

Datastead TVideoGrabber SDK

Table of contents

| | |
|--|----|
| General | 27 |
| Online Help | 27 |
| Features | 27 |
| New features | 29 |
| Development platforms - System requirements | 29 |
| Support - Contact | 29 |
| User guide | 30 |
| Licensing | 30 |
| Pre-requisties | 30 |
| Pre - requisties | 30 |
| Platform-dependent properties | 31 |
| Platform dependent properties | 31 |
| Device-dependent properties | 32 |
| Device dependent properties | 32 |
| Video sources | 33 |
| Video sources supported for preview and recording | 33 |
| USB and PCI video capture devices | 34 |
| Video capture devices | 34 |
| Preview | 36 |
| Video quality | 36 |
| Camera control | 37 |
| Detection of the video signal | 38 |
| Blackmagic Decklink cards | 38 |
| IP Cameras and other network streaming sources | 39 |
| The Datastead RTSP/RTMP/HTTP/ONVIF DirectShow source filter | 39 |
| ONVIF Cameras Discovery | 40 |
| ONVIF - Connecting to IP cameras through the ONVIF protocol | 40 |
| ONVIF - Retrieving camera information | 41 |
| ONVIF - PTZ | 41 |
| ONVIF Replay | 42 |
| ONVIF snapshot | 43 |
| Preview and recording of IP cameras and URL sources | 44 |
| Video stream of IP camera with audio of PC microphone or other audio capture device | 45 |
| IP cameras in MJPEG mode | 45 |
| IR Cut Filter of Axis cameras | 51 |
| Asynchronous vs Asynchronous connection | 52 |
| Auto-reconnection | 53 |
| NTP time of IP cameras | 53 |
| Using the TVideoGrabber http/mjpeg decoder when the Datastead RTSP filter is installed | 54 |
| Audio | 54 |
| Audio capture devices | 54 |
| Audio rendering | 55 |
| Audio levels and VU-meters | 56 |
| Datastead encoder | 57 |
| using the Datastead Encoder | 58 |

| | |
|--|-----|
| Recording | 59 |
| Recording methods and properties | 59 |
| Recording through the Datstead Encoder | 62 |
| Pause/resume during recording | 64 |
| Preroll recording (Backtimed recording) | 66 |
| Playing a video clip during recording | 67 |
| Dropped frames and audio/video sync problems | 68 |
| AES Encryption | 69 |
| AES Encryption Overview | 69 |
| Key | 70 |
| Realtime Encryption or Decryption | 70 |
| Batch Encryption or decryption | 70 |
| Disabling the encryption/decryption | 70 |
| Frame capture | 71 |
| Frame capture features | 71 |
| Graphic and text overlays | 74 |
| Setting and retrieving the overlay properties | 74 |
| Frame overlay vs window overlay | 75 |
| Overlays and aspect ratio | 76 |
| Image overlays | 76 |
| Text overlays | 79 |
| Graphic overlays | 80 |
| Chroma key | 82 |
| Overlays before or after transforms | 82 |
| To retrieve a pixel value | 82 |
| How to refresh the overlays while the video clip is paused | 83 |
| Mouse events | 83 |
| Player | 84 |
| Player features | 84 |
| Seamless playback | 87 |
| Current player state | 87 |
| Trackbar | 88 |
| Playing static images and animated GIFS | 89 |
| Playlist | 90 |
| Playing a clip from a TStream | 93 |
| StartPreview or StartRecording from video clips or URLs | 93 |
| AVI / ASF information and header attributes | 94 |
| Playing pictures | 94 |
| Synchronization of several player components | 95 |
| Video 360° | 96 |
| Decoding of 360° videos | 96 |
| PIP | 97 |
| PIP (Picture In Picture) | 97 |
| Streaming | 98 |
| Streaming through the Datstead Encoder | 98 |
| MMS streaming | 99 |
| Streaming using Newtek NDI | 101 |
| Video window - Display - Aspect Ratio - Dual display | 102 |
| Video window | 102 |
| Dual display | 104 |

| | |
|---|-----|
| Multiple video windows | 105 |
| Transparency - Color keying | 106 |
| madVR video renderer | 107 |
| Third-party external renderers | 107 |
| Screen recording - desktop capture | 108 |
| screen capture and recording with or without cursor | 108 |
| screen capture and recording of a specified window | 109 |
| Recording only a part of the screen | 110 |
| Desktop capture | 111 |
| Recording on the fly of video clips and live streams | 111 |
| video clips and live streams recorded on the fly | 112 |
| Video clips built on the fly from Bitmap handles, BMP or JPEG files | 112 |
| Video clips built on the fly by passing bitmap handles, BMP or JPEG files | 112 |
| Video clips from a fixed set of BMP files or JPEG files merged into a clip | 114 |
| Video clip built from a fixed set of BMP files or JPEG files | 114 |
| Video processing | 116 |
| Zoom | 116 |
| Rotation, mirroring, vertical and horizontal flip | 117 |
| Cropping and zooming | 117 |
| Image adjustments (brightness, contrast, etc) | 118 |
| Deinterlacing | 119 |
| Motion detection | 121 |
| Motion detection | 121 |
| Motion ratio | 122 |
| Color intensity | 122 |
| Grid structure / grid sensitivity | 123 |
| Color / Greyscale | 125 |
| Video noise | 125 |
| Recording only when motion is detected | 126 |
| Mixing several video sources | 127 |
| Mixing several video sources into a single one | 127 |
| Reencoding | 130 |
| Reencoding of clips in batch mode | 130 |
| Reencoding of clips in preview or recording mode | 133 |
| Merging or splitting video clips | 134 |
| Video capture devices with multiplexed inputs | 134 |
| Video capture devices having multiplexed inputs | 134 |
| Opening a clip or an IP URL from a background thread without blocking | 136 |
| Opening a clip or an IP URL from a background thread without blocking the main thread | 136 |
| Synchronization of TVideoGrabber components | 136 |
| Synchronization of several TVideoGrabber components | 136 |
| Logo | 137 |
| Logo displayed in the video window | 137 |
| WPF | 138 |
| VidGrabWPF:VideoGrabberWPF component | 138 |
| TVideoGrabber | 138 |
| Properties | 139 |
| AdjustOverlayAspectRatio | 146 |
| AdjustPixelAspectRatio | 147 |

| | |
|--|-----|
| Aero | 148 |
| AnalogVideoStandard | 149 |
| AnalogVideoStandards | 149 |
| AnalogVideoStandardsCount | 150 |
| ApplicationPriority | 150 |
| ASFAudioBitRate | 151 |
| ASFAudioChannels | 152 |
| ASFBufferWindow | 152 |
| ASFDeinterlaceMode | 153 |
| ASFFixedFrameRate | 153 |
| ASFMediaServerPublishingPoint | 154 |
| ASFMediaServerRemovePublishingPointAfterDisconnect | 155 |
| ASFMediaServerTemplatePublishingPoint | 155 |
| ASFNetworkMaxUsers | 156 |
| ASFNetworkPort | 156 |
| ASFProfile | 157 |
| ASFProfileFromCustomFile | 158 |
| ASFProfiles | 158 |
| ASFProfilesCount | 159 |
| ASFProfileVersion | 159 |
| ASFVideoBitRate | 160 |
| ASFVideoFrameRate | 161 |
| ASFVideoHeight | 161 |
| ASFVideoMaxKeyFrameSpacing | 162 |
| ASFVideoQuality | 163 |
| ASFVideoWidth | 163 |
| AspectRatioToUse | 164 |
| AssociateAudioAndVideoDevices | 164 |
| AudioBalance | 165 |
| AudioChannelRenderMode | 165 |
| AudioCompressor | 166 |
| AudioCompressorName | 167 |
| AudioCompressors | 167 |
| AudioCompressorsCount | 168 |
| AudioDevice | 169 |
| AudioDeviceName | 169 |
| AudioDeviceRendering | 170 |
| AudioDevices | 170 |
| AudioDevicesCount | 171 |
| AudioFormat | 172 |
| AudioFormats | 172 |
| AudioInput | 173 |
| AudioInputBalance | 173 |
| AudioInputLevel | 173 |
| AudioInputMono | 174 |
| AudioInputs | 174 |
| AudioInputsCount | 175 |
| AudioPeakEvent | 175 |
| AudioRecording | 176 |
| AudioRenderer | 176 |

| | |
|--|-----|
| AudioRendererName | 177 |
| AudioRenderers | 177 |
| AudioRenderersCount | 178 |
| AudioSource | 178 |
| AudioStreamNumber | 179 |
| AudioSyncAdjustment | 180 |
| AudioSyncAdjustmentEnabled | 180 |
| AudioVolume | 181 |
| AutoFileName | 181 |
| AutoFileNameDateTimeFormat | 182 |
| AutoFileNameMinDigits | 182 |
| AutoFilePrefix | 183 |
| AutoRefreshPreview | 184 |
| AutoStartPlayer | 184 |
| AVIDurationUpdated | 185 |
| AVIFormatOpenDML | 185 |
| AVIFormatOpenDMLCompatibilityIndex | 186 |
| BackgroundColor | 187 |
| BorderStyle | 187 |
| BurstCount | 187 |
| BurstInterval | 188 |
| BurstMode | 188 |
| BurstType | 189 |
| Busy | 190 |
| BusyCursor | 190 |
| CameraControlSettings | 190 |
| CaptureFileExt | 191 |
| ColorKey | 191 |
| ColorKeyEnabled | 192 |
| CompressionMode | 192 |
| CompressionType | 193 |
| Cropping_Enabled | 193 |
| Cropping_Height | 194 |
| Cropping_Outbounds | 194 |
| Cropping_Width | 195 |
| Cropping_X | 195 |
| Cropping_XMax | 196 |
| Cropping_Y | 196 |
| Cropping_YMax | 196 |
| Cropping_Zoom | 197 |
| CurrentFrameRate | 197 |
| CurrentState | 198 |
| DeliveredFrames | 198 |
| DirectShowFilters | 199 |
| DirectShowFiltersCount | 199 |
| Display_Active | 199 |
| Display_AlphaBlendEnabled | 200 |
| Display_AlphaBlendValue | 201 |
| Display_AspectRatio | 201 |
| Display_AutoSize | 202 |

| | |
|---|-----|
| Display_Embedded | 203 |
| Display_FullScreen | 203 |
| Display_Height | 204 |
| Display_Left | 205 |
| Display_Monitor | 205 |
| Display_MouseMovesWindow | 206 |
| Display_PanScanRatio | 207 |
| Display_StayOnTop | 207 |
| Display_Top | 208 |
| Display_TransparentColorEnabled | 209 |
| Display_TransparentColorValue | 209 |
| Display_VideoHeight | 210 |
| Display_VideoPortEnabled | 211 |
| Display_VideoWidth | 212 |
| Display_VideoWindowHandle | 212 |
| Display_Visible | 213 |
| Display_Width | 213 |
| DroppedFrames | 214 |
| DroppedFramesPollingInterval | 214 |
| DualDisplay_Active | 215 |
| DualDisplay_AlphaBlendEnabled | 215 |
| DualDisplay_AlphaBlendValue | 216 |
| DualDisplay_AspectRatio | 217 |
| DualDisplay_AutoSize | 217 |
| DualDisplay_Embedded | 218 |
| DualDisplay_FullScreen | 219 |
| DualDisplay_Height | 219 |
| DualDisplay_Left | 220 |
| DualDisplay_Monitor | 221 |
| DualDisplay_MouseMovesWindow | 221 |
| DualDisplay_PanScanRatio | 222 |
| DualDisplay_StayOnTop | 223 |
| DualDisplay_Top | 223 |
| DualDisplay_TransparentColorEnabled | 224 |
| DualDisplay_TransparentColorValue | 225 |
| DualDisplay_VideoHeight | 225 |
| DualDisplay_VideoPortEnabled | 226 |
| DualDisplay_VideoWidth | 226 |
| DualDisplay_VideoWindowHandle | 227 |
| DualDisplay_Visible | 228 |
| DualDisplay_Width | 228 |
| DVDateTimeEnabled | 229 |
| DVDiscontinuityMinimumInterval | 230 |
| DVDTitle | 230 |
| DVEncoder_VideoFormat | 231 |
| DVEncoder_VideoResolution | 231 |
| DVEncoder_VideoStandard | 232 |
| DVRecordingInNativeFormatSeparatesStreams | 232 |
| DVReduceFrameRate | 233 |
| DVRgb219 | 233 |

| | |
|---|-----|
| DVTimeCodeEnabled | 234 |
| EncryptionMethod | 235 |
| EventNotificationSynchronone | 235 |
| ExtraDLLPath | 235 |
| FixFlickerOrBlackCapture | 236 |
| FrameCaptureHeight | 237 |
| FrameCaptureWidth | 237 |
| FrameCaptureWithoutOverlay | 238 |
| FrameCaptureZoomSize | 239 |
| FrameGrabber | 239 |
| FrameGrabberCurrentRGBFormat | 240 |
| FrameGrabberRGBFormat | 241 |
| FrameNumberStartsFromZero | 241 |
| FrameRate | 242 |
| FramerateDivider | 242 |
| GetLastFrameWaitTimeoutMs | 243 |
| HoldRecording | 243 |
| ImageOverlay_AlphaBlend | 244 |
| ImageOverlay_AlphaBlendValue | 245 |
| ImageOverlay_ChromaKey | 246 |
| ImageOverlay_ChromaKeyLeewayPercent | 246 |
| ImageOverlay_ChromaKeyRGBColor | 247 |
| ImageOverlay_Height | 247 |
| ImageOverlay_LeftLocation | 248 |
| ImageOverlay_RotationAngle | 248 |
| ImageOverlay_StretchToVideoSize | 249 |
| ImageOverlay_TopLocation | 250 |
| ImageOverlay_Transparent | 250 |
| ImageOverlay_TransparentColorValue | 251 |
| ImageOverlay_UseTransparentColor | 252 |
| ImageOverlay_VideoAlignment | 252 |
| ImageOverlay_Width | 253 |
| ImageOverlayEnabled | 254 |
| ImageOverlaySelector | 254 |
| ImageRatio | 256 |
| InFrameProgressEvent | 257 |
| IPCameraURL | 257 |
| IsAnalogVideoDecoderAvailable | 258 |
| IsAudioCrossbarAvailable | 258 |
| IsAudioInputBalanceAvailable | 259 |
| IsCameraControlAvailable | 259 |
| IsDigitalVideoIn | 260 |
| IsDVCommandAvailable | 260 |
| IsHorizontalSyncLocked | 261 |
| IsMPEGStream | 261 |
| IsPlayerAudioStreamAvailable | 261 |
| IsPlayerVideoStreamAvailable | 262 |
| IsRecordingPaused | 263 |
| IsTimeCodeReaderAvailable | 263 |
| IsTVAudioAvailable | 264 |

| | |
|---|-----|
| IsTVAutoTuneRunning | 264 |
| IsTVTunerAvailable | 265 |
| IsVideoControlAvailable | 265 |
| IsVideoCrossbarAvailable | 265 |
| IsVideoInterlaced | 266 |
| IsVideoPortAvailable | 266 |
| IsVideoQualityAvailable | 267 |
| IsWDMVideoDriver | 267 |
| JPEGPerformance | 268 |
| JPEGProgressiveDisplay | 268 |
| JPEGQuality | 269 |
| Last_BurstFrameCapture_FileName | 269 |
| Last_CaptureFrameTo_FileName | 270 |
| Last_Clip_Played | 270 |
| Last_Recording_FileName | 271 |
| LogoDisplayed | 272 |
| LogoLayout | 272 |
| MixAudioSamples_CurrentSourceLevel | 273 |
| MixAudioSamples_ExternalSourceLevel | 273 |
| Mixer_MosaicColumns | 274 |
| Mixer_MosaicLines | 274 |
| MotionDetector_CompareBlue | 274 |
| MotionDetector_CompareGreen | 275 |
| MotionDetector_CompareRed | 276 |
| MotionDetector_Enabled | 276 |
| MotionDetector_GlobalMotionRatio | 277 |
| MotionDetector_GreyScale | 278 |
| MotionDetector_Grid | 278 |
| MotionDetector_GridXCount | 279 |
| MotionDetector_GridYCount | 280 |
| MotionDetector_IsGridValid | 280 |
| MotionDetector_MaxDetectionsPerSecond | 281 |
| MotionDetector_ReduceCPULoad | 282 |
| MotionDetector_ReduceVideoNoise | 282 |
| MotionDetector_Triggered | 283 |
| MouseWheelEventEnabled | 284 |
| MpegStreamType | 285 |
| MultiplexedInputEmulation | 285 |
| MultiplexedRole | 286 |
| MultiplexedStabilizationDelay | 286 |
| MultiplexedSwitchDelay | 287 |
| MuteAudioRendering | 287 |
| NDIFormatType | 288 |
| NDIName | 288 |
| NetworkStreaming | 288 |
| NetworkStreamingType | 289 |
| NormalCursor | 290 |
| NotificationMethod | 290 |
| NotificationPriority | 291 |
| NotificationSleepTime | 291 |

| | |
|--|-----|
| OnFrameBitmapEventSynchronise | 291 |
| OpenURLAsync | 292 |
| OverlayAfterTransform | 293 |
| ParentWindow | 294 |
| PlayerAudioCodec | 294 |
| PlayerAudioRendering | 295 |
| PlayerDuration | 295 |
| PlayerDVSize | 296 |
| PlayerFastSeekSpeedRatio | 297 |
| PlayerFileName | 297 |
| PlayerForcedCodec | 298 |
| PlayerFrameCount | 299 |
| PlayerFramePosition | 300 |
| PlayerFrameRate | 301 |
| PlayerHwAccel | 301 |
| PlayerOpenProgressPercent | 302 |
| PlayerRefreshPausedDisplay | 302 |
| PlayerRefreshPausedDisplayFrameRate | 303 |
| PlayerSpeedRatio | 303 |
| PlayerSpeedRatioConstantAudioPitch | 304 |
| PlayerState | 304 |
| PlayerTimePosition | 305 |
| PlayerTrackBar | 306 |
| PlayerTrackBarScale | 306 |
| PlayerTrackBarSynchronise | 307 |
| PlayerVideoCodec | 308 |
| PlaylistIndex | 308 |
| PreallocCapFileCopiedAfterRecording | 309 |
| PreallocCapFileEnabled | 310 |
| PreallocCapFileName | 310 |
| PreallocCapFileSizeInMB | 311 |
| PreviewZoomSize | 312 |
| RawAudioSampleCapture | 312 |
| RawCaptureAsyncEvent | 313 |
| RawSampleCaptureLocation | 313 |
| RawVideoSampleCapture | 314 |
| RecordingAudioBitRate | 314 |
| RecordingBacktimedFramesCount | 315 |
| RecordingCanPause | 315 |
| RecordingDuration | 316 |
| RecordingFileName | 316 |
| RecordingFileSizeMaxInMB | 317 |
| RecordingFourCC | 318 |
| RecordingHeight | 319 |
| RecordingInNativeFormat | 319 |
| RecordingMethod | 320 |
| RecordingOnMotion_Enabled | 321 |
| RecordingOnMotion_MotionThreshold | 322 |
| RecordingOnMotion_NoMotionPauseDelayMs | 322 |
| RecordingPauseCreatesNewFile | 323 |

| | |
|--|-----|
| RecordingSize | 324 |
| RecordingTimer | 324 |
| RecordingTimerInterval | 325 |
| RecordingVideoBitRate | 326 |
| RecordingWidth | 326 |
| Reencoding_IncludeAudioStream | 327 |
| Reencoding_IncludeVideoStream | 327 |
| Reencoding_Method | 328 |
| Reencoding_NewVideoClip | 329 |
| Reencoding_SourceVideoClip | 329 |
| Reencoding_StartFrame | 330 |
| Reencoding_StartTime | 330 |
| Reencoding_StopFrame | 331 |
| Reencoding_StopTime | 331 |
| Reencoding_UseAudioCompressor | 332 |
| Reencoding_UseFrameGrabber | 332 |
| Reencoding_UseVideoCompressor | 333 |
| Reencoding_WMVOutput | 333 |
| ScreenRecordingLayeredWindows | 334 |
| ScreenRecordingMonitor | 334 |
| ScreenRecordingNonVisibleWindows | 335 |
| ScreenRecordingSizePercent | 335 |
| ScreenRecordingThroughClipboard | 336 |
| ScreenRecordingWithCursor | 336 |
| SendToDV_DeviceIndex | 337 |
| ShapeOverlay | 337 |
| ShapeOverlayEnabled | 338 |
| SourceStream | 339 |
| SpeakerBalance | 340 |
| SpeakerControl | 340 |
| SpeakerVolume | 341 |
| StoragePath | 341 |
| StoreDeviceSettingsInRegistry | 342 |
| StreamingURL | 342 |
| StreamInterface_Format | 343 |
| StreamInterface_FrameRate | 343 |
| StreamInterface_IsRealTime | 344 |
| SyncCommands | 344 |
| SynchronizationRole | 345 |
| Synchronized | 345 |
| SyncPreview | 346 |
| SystemTempPath | 347 |
| TextOverlay_Align | 347 |
| TextOverlay_AlphaBlend | 348 |
| TextOverlay_AlphaBlendValue | 349 |
| TextOverlay_BkColor | 349 |
| TextOverlay_Enabled | 350 |
| TextOverlay_Font | 351 |
| TextOverlay_FontColor | 352 |
| TextOverlay_Left | 352 |

| | |
|---|-----|
| TextOverlay_Right | 353 |
| TextOverlay_Scrolling | 354 |
| TextOverlay_ScrollingSpeed | 355 |
| TextOverlay_Selector | 356 |
| TextOverlay_Shadow | 358 |
| TextOverlay_ShadowColor | 359 |
| TextOverlay_ShadowDirection | 360 |
| TextOverlay_String | 361 |
| TextOverlay_Top | 362 |
| TextOverlay_Transparent | 363 |
| TextOverlay_VideoAlignment | 364 |
| ThirdPartyDeinterlacer | 365 |
| TranslateMouseCoordinates | 365 |
| TunerFrequency | 366 |
| TunerMode | 366 |
| TVChannel | 367 |
| TVCountryCode | 367 |
| TVTunerInputType | 368 |
| TVUseFrequencyOverrides | 368 |
| UniqueID | 369 |
| UseClock | 369 |
| v360_AspectRatio | 370 |
| v360_Enabled | 370 |
| v360_MasterAngle | 371 |
| v360_MouseAction | 371 |
| v360_MouseActionPercent | 372 |
| VCRHorizontalLocking | 372 |
| Version | 373 |
| VideoCompression_DataRate | 373 |
| VideoCompression_KeyFrameRate | 374 |
| VideoCompression_PFramesPerKeyFrame | 374 |
| VideoCompression_Quality | 375 |
| VideoCompression_WindowSize | 375 |
| VideoCompressor | 376 |
| VideoCompressorName | 377 |
| VideoCompressors | 377 |
| VideoCompressorsCount | 378 |
| VideoControlSettings | 378 |
| VideoCursor | 379 |
| VideoDevice | 379 |
| VideoDeviceName | 380 |
| VideoDevices | 380 |
| VideoDevicesCount | 381 |
| VideoDevicesId | 382 |
| VideoDoubleBuffered | 382 |
| VideoFormat | 383 |
| VideoFormats | 384 |
| VideoFormatsCount | 384 |
| VideoFromImages_BitmapsSortedBy | 385 |
| VideoFromImages_RepeatIndefinitely | 385 |

| | |
|---|-----|
| VideoFromImages_SourceDirectory | 386 |
| VideoFromImages_TemporaryFile | 386 |
| VideoHeight | 387 |
| VideoHeight_PreferredAspectRatio | 387 |
| VideoInput | 388 |
| VideoInputs | 389 |
| VideoInputsCount | 389 |
| VideoPlayableWhileRecording | 390 |
| VideoProcessing_Brightness | 390 |
| VideoProcessing_Contrast | 391 |
| VideoProcessing_Deinterlacing | 391 |
| VideoProcessing_FlipHorizontal | 392 |
| VideoProcessing_FlipVertical | 392 |
| VideoProcessing_GrayScale | 393 |
| VideoProcessing_Hue | 393 |
| VideoProcessing_InvertColors | 394 |
| VideoProcessing_Pixellization | 394 |
| VideoProcessing_Rotation | 395 |
| VideoProcessing_RotationCustomAngle | 396 |
| VideoProcessing_Saturation | 396 |
| VideoQualitySettings | 397 |
| VideoRenderer | 397 |
| VideoRendererExternal | 398 |
| VideoRendererExternalIndex | 399 |
| VideoRendererPriority | 400 |
| VideoSize | 400 |
| VideoSizes | 401 |
| VideoSizesCount | 402 |
| VideoSource | 402 |
| VideoSource_FileOrURL | 403 |
| VideoSource_FileOrURL_StartTime | 404 |
| VideoSource_FileOrURL_StopTime | 404 |
| VideoSources | 405 |
| VideoSourcesCount | 405 |
| VideoStreamNumber | 406 |
| VideoSubtype | 406 |
| VideoSubtypes | 407 |
| VideoSubtypesCount | 407 |
| VideoVisibleWhenStopped | 407 |
| VideoWidth | 408 |
| VideoWidth_PreferredAspectRatio | 409 |
| Visible | 409 |
| VUMeter | 410 |
| WebcamStillCaptureButton | 410 |
| ZoomCoeff | 411 |
| ZoomXCenter | 412 |
| ZoomYCenter | 412 |
| Methods | 412 |
| About | 419 |
| AnalogVideoStandardIndex | 419 |

| | |
|---|-----|
| ASFStreaming_GetAuthorizationList | 420 |
| ASFStreaming_GetConnectedClients | 420 |
| ASFStreaming_GetConnectedClientsCount | 421 |
| ASFStreaming_ResetAuthorizations | 421 |
| ASFStreaming_SetAuthorization | 422 |
| AssociateMultiplexedSlave | 423 |
| AudioCompressorIndex | 423 |
| AudioDeviceIndex | 424 |
| AudioInputIndex | 424 |
| AudioRendererIndex | 425 |
| AVIDuration | 425 |
| AVIHeaderInfo | 426 |
| AVIInfo | 427 |
| AVIInfo2 | 429 |
| CameraControlAuto | 430 |
| CameraControlDefault | 430 |
| CameraControlMax | 430 |
| CameraControlMin | 431 |
| CameraControlStep | 431 |
| CameraControlValue | 432 |
| Cancel | 432 |
| CanProcessMessages | 433 |
| CaptureFrameRenderedTo | 433 |
| CaptureFrameSyncTo | 434 |
| CaptureFrameTo | 434 |
| ClearHeaderAttributes | 435 |
| ClosePlayer | 435 |
| Create | 436 |
| CreatePreallocCapFile | 436 |
| Decrypt_File | 437 |
| Destroy | 437 |
| Display_SetLocation | 437 |
| DrawBitmapOverFrame | 438 |
| DualDisplay_SetLocation | 439 |
| DVDInfo | 440 |
| EnableMultiplexedInput | 441 |
| EnableThreadMode | 441 |
| Encoder_CloseOutputFile | 442 |
| Encoder_GetInt | 442 |
| Encoder_NewOutputFile | 443 |
| Encoder_Pause | 443 |
| Encoder_Resume | 444 |
| Encoder_SetInt | 444 |
| Encoder_SetStr | 445 |
| Encoders_CreateInstanceForRecording | 445 |
| Encoders_CreateInstanceForStreaming | 445 |
| Encoders_RemoveAllInstances | 446 |
| Encoders_RemoveInstance | 446 |
| Encrypt_File | 447 |
| EnumerateWindows | 447 |

| | |
|--|-----|
| Facebook_GoLive_ReturnStreamURL | 448 |
| FastForwardPlayer | 449 |
| FindIndexInListByName | 449 |
| GetAudioCodec | 450 |
| GetCameraExposure | 451 |
| GetCameraExposureAsString | 451 |
| GetDisplayActive | 451 |
| GetDisplayAlphaBlendEnabled | 452 |
| GetDisplayAlphaBlendValue | 452 |
| GetDisplayAspectRatio | 452 |
| GetDisplayAutoSize | 453 |
| GetDisplayEmbedded | 453 |
| GetDisplayFullScreen | 453 |
| GetDisplayHeight | 454 |
| GetDisplayLeft | 454 |
| GetDisplayMonitor | 455 |
| GetDisplayMouseMoveWindow | 455 |
| GetDisplayPanScanRatio | 455 |
| GetDisplayStayOnTop | 456 |
| GetDisplayTop | 456 |
| GetDisplayTransparentColorEnabled | 456 |
| GetDisplayTransparentColorValue | 457 |
| GetDisplayVideoHeight | 457 |
| GetDisplayVideoPortEnabled | 457 |
| GetDisplayVideoWidth | 458 |
| GetDisplayVideoWindowHandle | 458 |
| GetDisplayVisible | 458 |
| GetDisplayWidth | 459 |
| GetFrameInfo | 459 |
| GetFrameInfoString | 460 |
| GetFWCam1394 | 461 |
| GetFWCam1394List | 461 |
| GetImageOverlay_AlphaBlend | 462 |
| GetImageOverlay_AlphaBlendValue | 462 |
| GetImageOverlay_ChromaKey | 462 |
| GetImageOverlay_ChromaKeyLeewayPercent | 463 |
| GetImageOverlay_ChromaKeyRGBColor | 463 |
| GetImageOverlay_Enabled | 463 |
| GetImageOverlay_Height | 464 |
| GetImageOverlay_LeftLocation | 464 |
| GetImageOverlay_RotationAngle | 464 |
| GetImageOverlay_StretchToVideoSize | 465 |
| GetImageOverlay_TargetDisplay | 465 |
| GetImageOverlay_TopLocation | 465 |
| GetImageOverlay_Transparent | 465 |
| GetImageOverlay_TransparentColorValue | 466 |
| GetImageOverlay_UseTransparentColor | 466 |
| GetImageOverlay_Width | 466 |
| GetItemNameFromList | 467 |
| GetLastAverageStreamValue | 467 |

| | |
|---------------------------------------|-----|
| GetLastErrorMessage | 467 |
| GetLastFrameAsHBITMAP | 468 |
| GetLastFrameAsTBitmap | 469 |
| GetLastFrameBitmapBits | 470 |
| GetLastFrameBitmapBits2 | 471 |
| GetLogString | 471 |
| GetMiscDeviceControl | 472 |
| GetNearestVideoHeight | 472 |
| GetNearestVideoSize | 473 |
| GetNearestVideoWidth | 473 |
| GetPixelsDistance | 474 |
| GetPlaylist | 474 |
| GetRGBPixelAt | 475 |
| GetTextOverlay_Align | 476 |
| GetTextOverlay_AlphaBlend | 477 |
| GetTextOverlay_AlphaBlendValue | 477 |
| GetTextOverlay_BkColor | 478 |
| GetTextOverlay_Enabled | 478 |
| GetTextOverlay_Font | 478 |
| GetTextOverlay_GradientColor | 479 |
| GetTextOverlay_GradientMode | 479 |
| GetTextOverlay_HighResFont | 479 |
| GetTextOverlay_Left | 479 |
| GetTextOverlay_Right | 480 |
| GetTextOverlay_Scrolling | 480 |
| GetTextOverlay_ScrollingSpeed | 480 |
| GetTextOverlay_Shadow | 481 |
| GetTextOverlay_ShadowColor | 481 |
| GetTextOverlay_ShadowDirection | 481 |
| GetTextOverlay_String | 482 |
| GetTextOverlay_TargetDisplay | 482 |
| GetTextOverlay_Top | 482 |
| GetTextOverlay_Transparent | 482 |
| GetTranslatedCoordinates | 483 |
| GetTVChannelInfo | 483 |
| GetVideoCodec | 484 |
| GetVideoCompressionSettings | 484 |
| GetVideoControlMode | 485 |
| GetVideoHeightFromIndex | 486 |
| GetVideoSizeFromIndex | 486 |
| GetVideoWidthFromIndex | 487 |
| GetVMR9ImageAdjustmentBounds | 487 |
| GetVUMeterSetting | 488 |
| GraphState | 488 |
| IsAudioDeviceASoundCard | 489 |
| IsAudioDeviceConnected | 489 |
| IsAudioRendererConnected | 490 |
| IsCameraControlSettingAvailable | 490 |
| IsDialogAvailable | 491 |
| IsDVDevice | 491 |

| | |
|--|-----|
| IsPlaylistActive | 491 |
| IsURLResponding | 492 |
| IsURLVideoStreamAvailable | 492 |
| IsVideoControlModeAvailable | 493 |
| IsVideoDeviceConnected | 493 |
| IsVideoQualitySettingAvailable | 494 |
| IsVideoSignalDetected | 494 |
| IsVMR9ImageAdjustmentAvailable | 495 |
| LoadCompressorSettingsFromDataString | 496 |
| LoadCompressorSettingsFromTextFile | 496 |
| MixAudioSamples | 497 |
| Mixer_Activation | 497 |
| Mixer_AddToMixer | 498 |
| Mixer_RemoveFromMixer | 499 |
| Mixer_SetOverlayRoundedCorner | 500 |
| Mixer_SetupPIPFromSource | 500 |
| Monitor_Primary_Index | 501 |
| MonitorBounds | 502 |
| MonitorsCount | 503 |
| MotionDetector_CellColorIntensity | 503 |
| MotionDetector_CellMotionRatio | 504 |
| MotionDetector_EnumGridDialogControls | 504 |
| MotionDetector_Get2DTextGrid | 506 |
| MotionDetector_Get2DTextMotion | 506 |
| MotionDetector_GetCellLocation | 507 |
| MotionDetector_GetCellSensitivity | 507 |
| MotionDetector_GetCellSize | 508 |
| MotionDetector_GlobalColorIntensity | 509 |
| MotionDetector_GloballyIncOrDecSensitivity | 509 |
| MotionDetector_Reset | 510 |
| MotionDetector_ResetGlobalSensitivity | 510 |
| MotionDetector_SetCellSensitivity | 511 |
| MotionDetector_SetGridSize | 511 |
| MotionDetector_ShowGridDialog | 512 |
| MotionDetector_TriggerNow | 513 |
| MotionDetector_UseThisReferenceSample | 513 |
| MP4NeedsReindexing | 514 |
| MPEGProgramSetting | 515 |
| MultipurposeEncoder_QuickConfigure_UDPStreaming_H264 | 515 |
| MultipurposeEncoder_ReindexClip | 516 |
| NotifyPlayerTrackbarAction | 516 |
| ONVIF_GetStr | 517 |
| ONVIF_SetStr | 517 |
| ONVIFCancelDiscovery | 518 |
| ONVIFDeviceInfo | 518 |
| ONVIFDiscoverCameras_IPRange | 518 |
| ONVIFDiscoverCameras_Multicast | 519 |
| ONVIFEnumCamerasDiscovered | 520 |
| ONVIFPTZGetLimits | 520 |
| ONVIFPTZGetPosition | 521 |

| | |
|--|-----|
| ONVIFPTZPreset | 521 |
| ONVIFPTZSendAuxiliaryCommand | 522 |
| ONVIFPTZSetPosition | 522 |
| ONVIFPTZStartMove | 523 |
| ONVIFPTZStopMove | 523 |
| ONVIFSnapShot | 524 |
| OpenDVD | 524 |
| OpenPlayer | 525 |
| OpenPlayerAtFramePositions | 526 |
| OpenPlayerAtTimePositions | 527 |
| OpenURLAsyncStatus | 528 |
| PausePlayer | 528 |
| PausePreview | 529 |
| PauseRecording | 529 |
| PlayerFrameStep | 530 |
| Playlist | 531 |
| PointGreyConfig | 531 |
| PutMiscDeviceControl | 532 |
| RecordingKBytesWrittenToDisk | 533 |
| RecordToNewFileNow | 533 |
| RefreshDevicesAndCompressorsLists | 534 |
| RefreshPlayerOverlays | 535 |
| ResetVideoDeviceSettings | 535 |
| ResumePreview | 536 |
| ResumeRecording | 536 |
| RetrieveInitialXYAfterRotation | 537 |
| RewindPlayer | 538 |
| RunPlayer | 539 |
| RunPlayerBackwards | 539 |
| SaveCompressorSettingsToDataString | 540 |
| SaveCompressorSettingsToTextFile | 541 |
| ScreenRecordingUsingCoordinates | 541 |
| SendCameraCommand | 542 |
| SendDVCommand | 543 |
| SendImageToVideoFromBitmaps | 543 |
| SendImageToVideoFromBitmaps2 | 544 |
| SendIPCameraCommand | 545 |
| SetAudioRendererAdditional | 545 |
| SetAuthentication | 546 |
| SetAVIMuxConfig | 546 |
| SetCameraControl | 547 |
| SetCameraExposure | 548 |
| SetDatasteadFilterDllName | 548 |
| SetDecryptionKey | 549 |
| SetDisplayActive | 549 |
| SetDisplayAlphaBlendEnabled | 549 |
| SetDisplayAlphaBlendValue | 550 |
| SetDisplayAspectRatio | 550 |
| SetDisplayAutoSize | 551 |
| SetDisplayEmbedded | 551 |

| | |
|--|-----|
| SetDisplayFullScreen | 551 |
| SetDisplayHeight | 552 |
| SetDisplayLeft | 552 |
| SetDisplayLocation | 552 |
| SetDisplayMonitor | 553 |
| SetDisplayMouseMovesWindow | 553 |
| SetDisplayPanScanRatio | 554 |
| SetDisplayParent | 554 |
| SetDisplayStayOnTop | 554 |
| SetDisplayTop | 555 |
| SetDisplayTransparentColorEnabled | 555 |
| SetDisplayTransparentColorValue | 555 |
| SetDisplayVideoPortEnabled | 556 |
| SetDisplayVisible | 556 |
| SetDisplayWidth | 556 |
| SetEncryptionKey | 557 |
| SetFFmpegAudioFilter | 557 |
| SetFFmpegFilter | 558 |
| SetFrameCaptureBounds | 558 |
| SetFWCam1394 | 559 |
| SetHeaderAttribute | 560 |
| SetImageOverlay_AlphaBlend | 561 |
| SetImageOverlay_AlphaBlendValue | 562 |
| SetImageOverlay_Attributes | 563 |
| SetImageOverlay_Attributes2 | 564 |
| SetImageOverlay_ChromaKey | 565 |
| SetImageOverlay_ChromaKeyLeewayPercent | 566 |
| SetImageOverlay_ChromaKeyRGBColor | 567 |
| SetImageOverlay_Enabled | 567 |
| SetImageOverlay_Height | 568 |
| SetImageOverlay_LeftLocation | 569 |
| SetImageOverlay_RotationAngle | 570 |
| SetImageOverlay_StretchToVideoSize | 571 |
| SetImageOverlay_TargetDisplay | 572 |
| SetImageOverlay_TopLocation | 573 |
| SetImageOverlay_Transparent | 573 |
| SetImageOverlay_TransparentColorValue | 574 |
| SetImageOverlay_UseTransparentColor | 575 |
| SetImageOverlay_Width | 576 |
| SetImageOverlayFromBMPFile | 577 |
| SetImageOverlayFromBMPFile2 | 578 |
| SetImageOverlayFromHBitmap | 578 |
| SetImageOverlayFromHBitmap2 | 579 |
| SetImageOverlayFromImageFile | 580 |
| SetImageOverlayFromImageFile2 | 581 |
| SetImageOverlayFromJPEGFile | 582 |
| SetImageOverlayFromJPEGFile2 | 583 |
| SetImageOverlayFromTBitmap | 583 |
| SetImageOverlayFromTBitmap2 | 584 |
| SetImageOverlayFromTImage | 585 |

| | |
|--------------------------------------|-----|
| SetImageOverlayFromTImage2 | 585 |
| SetIPCameraSetting | 586 |
| SetLocation | 587 |
| SetLogoFromBMPFile | 588 |
| SetLogoFromHBitmap | 588 |
| SetLogoFromJPEGFile | 589 |
| SetLogoFromTBitmap | 589 |
| SetLogoFromTImage | 589 |
| SetMultiplexerFilterByName | 590 |
| SetParentWindow | 590 |
| SetTextOverlay_Align | 591 |
| SetTextOverlay_AlphaBlend | 592 |
| SetTextOverlay_AlphaBlendValue | 592 |
| SetTextOverlay_BkColor | 593 |
| SetTextOverlay_CustomVar | 594 |
| SetTextOverlay_Enabled | 595 |
| SetTextOverlay_Font | 596 |
| SetTextOverlay_FontColor | 597 |
| SetTextOverlay_GradientColor | 598 |
| SetTextOverlay_GradientMode | 598 |
| SetTextOverlay_HighResFont | 599 |
| SetTextOverlay_Left | 600 |
| SetTextOverlay_Right | 601 |
| SetTextOverlay_Scrolling | 602 |
| SetTextOverlay_ScrollingSpeed | 602 |
| SetTextOverlay_Shadow | 603 |
| SetTextOverlay_ShadowColor | 604 |
| SetTextOverlay_ShadowDirection | 605 |
| SetTextOverlay_String | 606 |
| SetTextOverlay_TargetDisplay | 607 |
| SetTextOverlay_Top | 608 |
| SetTextOverlay_Transparent | 608 |
| SetVideoCompressionDefaults | 609 |
| SetVideoCompressionSettings | 610 |
| SetVideoControlMode | 611 |
| SetVideoControlMode2 | 611 |
| SetVideoQuality | 611 |
| SetVMR9ImageAdjustmentValue | 612 |
| SetVuMeter_Enabled | 613 |
| SetVUMeterSetting | 613 |
| SetWindowRecordingByHandle | 614 |
| SetWindowRecordingByName | 614 |
| SetWindowTransparency | 615 |
| ShapeOverlayList | 616 |
| ShowDebugWindow | 617 |
| ShowDialog | 617 |
| StartAudioRecording | 618 |
| StartAudioRendering | 619 |
| StartPreview | 620 |
| StartRecording | 620 |

| | |
|--|-----|
| StartReencoding | 621 |
| StartSynchronized | 621 |
| Stop | 622 |
| StopPlayer | 622 |
| StopPreview | 623 |
| StopRecording | 623 |
| StopReencoding | 624 |
| StreamInterface_PushData | 625 |
| TextOverlay_CreateCustomFont | 625 |
| TextOverlay_CreateCustomFont2 | 626 |
| ThirdPartyFilter_AddToList | 627 |
| ThirdPartyFilter_ClearList | 628 |
| ThirdPartyFilter_Enable | 628 |
| ThirdPartyFilter_RemoveFromList | 629 |
| ThirdPartyFilter_ShowDialog | 629 |
| TVClearFrequencyOverrides | 629 |
| TVGetMinMaxChannels | 630 |
| TVSetChannelFrequencyOverride | 630 |
| TVStartAutoScan | 631 |
| TVStopAutoScan | 632 |
| UseNearestVideoSize | 632 |
| v360_AddYawPitchRoll | 633 |
| v360_GetAngle | 633 |
| v360_GetYawPitchRoll | 634 |
| v360_ResetAnglesToDefault | 634 |
| v360_SetAngle | 634 |
| v360_SetInterpolation | 635 |
| v360_SetProjection | 635 |
| v360_SetStereoFormat | 636 |
| v360_SetTranspose | 636 |
| v360_SetYawPitchRoll | 637 |
| VideoCompressorIndex | 637 |
| VideoDeviceIndex | 638 |
| VideoDeviceIndexFromId | 638 |
| VideoFormatIndex | 639 |
| VideoFromImages_CreateSetOfBitmaps | 639 |
| VideoInputIndex | 640 |
| VideoQualityAuto | 640 |
| VideoQualityDefault | 641 |
| VideoQualityMax | 641 |
| VideoQualityMin | 641 |
| VideoQualityStep | 642 |
| VideoQualityValue | 642 |
| VideoSizeIndex | 643 |
| VideoSubtypeIndex | 643 |
| WriteScriptCommand | 644 |
| Events | 644 |
| OnAudioBufferNegotiation | 646 |
| OnAudioDeviceSelected | 646 |
| OnAudioPeak | 647 |

| | |
|---------------------------------------|-----|
| OnAuthenticationNeeded | 647 |
| OnAVIDurationUpdated | 648 |
| OnBacktimedFramesCountReached | 649 |
| OnBitmapsLoadingProgress | 650 |
| OnClick | 650 |
| OnClientConnection | 650 |
| OnColorKeyChange | 651 |
| OnCopyPreallocDataCompleted | 652 |
| OnCopyPreallocDataProgress | 652 |
| OnCopyPreallocDataStarted | 653 |
| OnCreatePreallocFileCompleted | 653 |
| OnCreatePreallocFileProgress | 654 |
| OnCreatePreallocFileStarted | 654 |
| OnDbClick | 654 |
| OnDeviceArrivalOrRemoval | 655 |
| OnDeviceLost | 656 |
| OnDeviceReconnected | 656 |
| OnDeviceReconnecting | 656 |
| OnDirectNetworkStreamingHostUrl | 657 |
| OnDiskFull | 657 |
| OnDragDropFiles | 658 |
| OnDVCommandCompleted | 658 |
| OnDVDDiscontinuity | 659 |
| OnEnumerateWindows | 659 |
| OnFilterSelected | 660 |
| OnFrameBitmap | 660 |
| OnFrameCaptureCompleted | 662 |
| OnFrameOverlayUsingDC | 663 |
| OnFrameOverlayUsingDIB | 664 |
| OnFrameOverlayUsingVIDEOHDR | 665 |
| OnFrameProgress | 666 |
| OnFrameProgress2 | 667 |
| OnGraphBuilt | 667 |
| OnInactive | 668 |
| OnKeyPress | 668 |
| OnLastCommandCompleted | 669 |
| OnLeavingFullScreen | 669 |
| OnLog | 670 |
| OnMotionDetected | 670 |
| OnMotionNotDetected | 671 |
| OnMouseDown | 671 |
| OnMouseDown_Video | 673 |
| OnMouseDown_Window | 673 |
| OnMouseMove | 673 |
| OnMouseMove_Video | 675 |
| OnMouseMove_Window | 675 |
| OnMouseUp | 675 |
| OnMouseUp_Video | 677 |
| OnMouseUp_Window | 677 |
| OnMouseWheel | 677 |

| | |
|---|-----|
| OnNoVideoDevices | 678 |
| OnNTPTimeStamp | 679 |
| OnONVIFDiscoveryCompletedNotification | 679 |
| OnOpenURLAsyncStatusChanged | 680 |
| OnPlayerBufferingData | 680 |
| OnPlayerEndOfPlaylist | 681 |
| OnPlayerEndOfStream | 681 |
| OnPlayerOpened | 682 |
| OnPlayerStateChanged | 682 |
| OnPlayerUpdateTrackbarPosition | 683 |
| OnPreviewStarted | 683 |
| OnRawAudioSample | 684 |
| OnRawVideoSample | 684 |
| OnRecordingCompleted | 685 |
| OnRecordingPaused | 685 |
| OnRecordingReadyToStart | 686 |
| OnRecordingStarted | 687 |
| OnReencodingCompleted | 688 |
| OnReencodingProgress | 688 |
| OnReencodingStarted | 689 |
| OnReinitializing | 689 |
| OnResizeVideo | 689 |
| OnTextOverlayScrollingCompleted | 690 |
| OnThirdPartyFilterConnected | 690 |
| OnThreadSync | 691 |
| OnTVChannelScanCompleted | 691 |
| OnTVChannelScanStarted | 692 |
| OnTVChannelSelected | 692 |
| OnVideoCompressionSettings | 693 |
| OnVideoDeviceSelected | 693 |
| OnVideoFromBitmapsNextFrameNeeded | 694 |
| Types | 695 |
| TAero | 695 |
| TApplicationPriority | 695 |
| TASFDeinterlaceMode | 696 |
| TASFProfileVersion | 696 |
| TAspectRatio | 696 |
| TAudioSource | 697 |
| TAuthenticationType | 697 |
| TAutoFileName | 698 |
| TAVIInfoType | 699 |
| TAVIMuxConfig | 699 |
| TCardinalDirection | 700 |
| TDialog | 701 |
| TDiscoveryCallbackStatus | 701 |
| TDVCommand | 702 |
| TDVDInfoType | 703 |
| TDVSize | 703 |
| TDVVideoFormat | 704 |
| TDVVideoStandard | 704 |

| | |
|--|-----|
| TEncoder_int | 705 |
| TEncoder_str | 706 |
| TEncryptionMethod | 706 |
| TEventNotification | 707 |
| TFileSort | 707 |
| TFormatType | 707 |
| TFrameBitmapInfo | 708 |
| TFrameCaptureDest | 708 |
| TFrameInfo | 709 |
| TFrameInfoId | 710 |
| TFrameInfoStringId | 711 |
| TGPUEncoder | 711 |
| TGraphState | 711 |
| THeaderAttribute | 712 |
| THwAccel | 712 |
| TIPCameraSetting | 712 |
| TLogoLayout | 713 |
| TLogType | 713 |
| TMiscDeviceControl | 718 |
| TMouseButton | 718 |
| TMPEGProgramSetting | 718 |
| TMpegStreamType | 719 |
| TMultiplexedRole | 719 |
| TNDIFormatType | 720 |
| TNetworkStreaming | 720 |
| TNetworkStreamingType | 721 |
| TNotificationMethod | 721 |
| TOnAudioBufferNegotiation | 722 |
| TOnAudioPeak | 722 |
| TOnAuthenticationNeeded | 722 |
| TOnAVIDurationUpdated | 723 |
| TOnClientConnection | 723 |
| TOnColorKeyChange | 724 |
| TOnCreatePreallocatedFileCompleted | 724 |
| TOnDeviceArrivalOrRemoval | 724 |
| TOnDirectNetworkStreamingHostUrl | 725 |
| TOnDragDropFiles | 726 |
| TOnDVCommandCompleted | 726 |
| TOnDVDDiscontinuity | 726 |
| TOnEnumerateWindows | 727 |
| TOnFileNotification | 727 |
| TOnFilterSelected | 728 |
| TOnFrameCaptureCompleted | 728 |
| TOnFrameOverlayUsingDC | 728 |
| TOnFrameOverlayUsingDIB | 729 |
| TOnFrameProgress | 731 |
| TOnLog | 731 |
| TOnMotionDetected | 732 |
| TOnMotionNotDetected | 733 |
| TOnMouseWheel | 733 |

| | |
|--|-----|
| TOnONVIFDiscoveryCompletedNotification | 734 |
| TOnOpenURLAsyncStatusChanged | 734 |
| TOnPlayerBufferingData | 735 |
| TOnPlayerStateChanged | 735 |
| TOnProgress | 736 |
| TOnProgressCommented | 736 |
| TOnRawSample | 737 |
| TOnRecordingCompleted | 737 |
| TOnRecordingReadyToStart | 738 |
| TOnResizeVideo | 739 |
| TOnSourceFileToDestFileCompleted | 739 |
| TOnSourceFileToDestFileStarted | 740 |
| TOnTextOverlayScrollingCompleted | 740 |
| TOnThirdPartyFilterConnected | 740 |
| TOnThreadSync | 741 |
| TOnTVChannelScanStarted | 741 |
| TOnTVChannelSelected | 742 |
| TOnVideoCompressionSettings | 743 |
| TOnVideoFromBitmapsNextFrameNeeded | 743 |
| TOnVideoKeyPress | 744 |
| TOnVideoMouseMove | 744 |
| TOnVideoMouseUpDown | 745 |
| TONVIFDeviceInfo | 745 |
| TOpenURLAsyncStatus | 746 |
| TPlayerState | 746 |
| TPlaylist | 747 |
| TPointGreyConfig | 747 |
| TRawSampleCaptureLocation | 748 |
| TRecordingMethod | 748 |
| TRecordingTimer | 749 |
| TRegistryRoot | 750 |
| TRGBSelector | 750 |
| TStreamType | 750 |
| TSynchronizationRole | 751 |
| TSyncPreview | 751 |
| TTextOverlayGradientMode | 752 |
| TThirdPartyFilterList | 752 |
| TThreadSyncPoint | 753 |
| TTrackbarAction | 754 |
| TTriState | 754 |
| TTunerMode | 754 |
| TTVChannelInfo | 755 |
| Tv360_Angle | 755 |
| Tv360_InOut | 756 |
| Tv360_Interpolation | 756 |
| TV360_MouseAction | 756 |
| Tv360_Projection | 757 |
| Tv360_StereoFormat | 758 |
| TVideoAlignment | 758 |
| TVideoDeinterlacing | 758 |

| | |
|---------------------------------|-----|
| TVideoRenderer | 759 |
| TVideoRendererPriority | 760 |
| TVideoRotation | 760 |
| TVideoSource | 761 |
| TVMR9ImageAdjustment | 761 |
| TVUMeter | 762 |
| TVUMeterSetting | 762 |
| TWebcamStillCaptureButton | 763 |

General

Created with the Standard Edition of HelpNDoc: [Converting Word Documents to eBooks: A Step-by-Step Guide with HelpNDoc](#)

Online Help

Online Help

[Prev](#)

[Next](#)

Datastead TVideoGrabber SDK 16.2 help Description

The latest version of the TVideoGrabber SDK 16.2 documentation is available here:

<https://www.datastead.com/video-sdk/>

The FAQ, including code snippets, is available here:

<https://www.datastead.com/faq/>

For a question feel free to contact us at contact@datastead.com or from our contact page:

<https://www.datastead.com/contact/>

Created with the Standard Edition of HelpNDoc: [Free Web Help generator](#)

Features

Features

[Prev](#)

[Next](#)

TVideoGrabber Video SDK v16.4

for C#, VB.NET, VC++, Delphi and C++Builder (all versions), and ActiveX compatible tools (PowerBuilder, LabView,...)

©2025 Datastead Software

Home page:

<https://www.datastead.com/>

Description

CAPTURE AND PREVIEW OF VIDEO SOURCES

The TVideoGrabber SDK supports most of the video sources and capture devices:

- ONVIF cameras with ONVIF discovery PTZ support (*)
- RTSP cameras and servers
- streaming URLs (*)
- webcams,
- USB video capture devices,
- PCIe video capture cards,
- Blackmagic Decklink cards,
- USB composite capture devices (e.g. Easycap),
- GigE cameras (e.g. PointGrey)

(*) requires our optional [Datastead RTSP/RTMP/HTTP/ONVIF Source Filter](#)

VIDEO RECORDING

- audio/video recording to various formats

through the [Datastead Encoder](#) after installing our optional [Datastead Multipurpose Encoder filter](#)

Most of the recording formats include MP4, MOV, FLV, AVI, Ogg/Theora, WebM, etc...

- audio/video recording through third-party DirectShow filters

- AVI recording,
 - WMV/ASF recording,
 - MP4/FLV recording (requires a third-party H264 or AAC DirectShow encoder)
- Timer-based recording (delayed start, timered stop, or create new file periodically)

Compression of the audio and video streams, "on-the-fly" or after recording,
Pause/resume during recording, with possibility to create a new clip for each pause/resume,
Preview while recording.

NETWORK STREAMING

- video/audio streaming through our optional [Datastead Multipurpose Encoder filter](#),
- MMS streaming,
- NDI streaming through our optional [Datastead NDI filters](#)

MEDIA PLAYER

- play most of the audio/video clips after installing the [LAV decoder filters](#).
- play static images: BMP, PNG, JPEG, GIF, ...
- trackbar control,
- playlist support,
- playback at higher or lower speed, forward or backward,
- fast seeking, forward or backward,
- synchronization of several TVideoGrabber player components,
- play streaming sources,
- opens static JPEG, BMP, PNG, GIF image files, allowing image processings and overlays

FRAME CAPTURE

- on the fly during preview, recording or playback, to memory bitmap, or to BMP, JPEG, PNG, TIFF files

IP CAMERAS / IP VIDEO SERVERS

- supports RTSP / ONVIF IP cameras and other URL sources: HTTP, RTMP, UDP/MPEG-TS, RTSPS, HTTPS (requires our [Datastead RTSP/RTMP/HTTP/ONVIF Source Filter](#))
- support for ONVIF PTZ control

MOTION DETECTION

- the video frame area can be divided in cells of motion detection,

MULTIPLE VIDEO WINDOWS

- assigned to different monitors (e.g. a small video window on a monitor and a full screen video on the 2nd and 3rd monitors)

VIDEO PROCESSING AND OVERLAYS

- multiple text and image overlays, over the video frame or directly over the video window(s),
- rotation, resizing, alpha blending, chroma keying,
- video transparency,
- chroma key,
- video rotation 90° 180° 270°,
- top-down and left-right,
- half size and full size deinterlacing,
- brightness, contrast, saturation, hue,
- inverted colors,
- greyscale
- video processing using FFmpeg filters with [SetFFmpegFilter](#) (requires the Datastead RTSP/RTMP/HTTP/ONVIF Source filter)

SCREEN RECORDING

- with or without cursor
- full screen recording (or one monitor when using several monitors),
- recording of the extended desktop across several monitors,
- recording of a given window identified by its name or handle

CROPPING AND ZOOMING

- recording of only a cropped area of the video source
- zooming within areas the video source

VIDEO CLIP BUILT FROM BITMAPS OR IMAGE FILES (BMP, JPEG, GIF, PNG, etc...)

- the final frame rate can be adjusted at the end of the recording

REENCODING OF VIDEO AND AUDIO CLIPS

- by using the installed audio and/or video compressors,
- by applying graphics and/or text overlays,
- by extracting sequences from a start and stop time.

AUDIO VU-METERS

- "needle analog" style or "digital bargraph" style

MULTIPLEXED INPUTS SUPPORT

- supports video cards with several inputs,
- accept 4 switched inputs and 16 switched inputs in 2x2 or 4x4 mosaic video or in master/slave modes,

New features

New features

[Prev](#)[Next](#)

New features

Description

NEW FEATURES

The changelog including new features and bug fixes can be found [here](#).

Development platforms - System requirements

Development platforms - System requirements

[Prev](#)[Next](#)

Development platforms - System requirements

Description

DEVELOPMENT TOOLS SUPPORTED

- Delphi and C++Builder
- Visual Studio .NET
- C++, QT
- ActiveX

SYSTEM REQUIREMENTS

Windows 11, Windows 10, Windows 8.1, Windows 8, Windows 7, Windows XP

Support - Contact

Support - Contact

[Prev](#)[Next](#)

Product support.

Description

Support

online documentation: <https://www.datastead.net/vidgrabhelp/Datastead%20TVideoGrabber%20SDK.html>

FAQ: <https://www.datastead.com/faq/>

Contact emails

Support questions: support@datastead.com

Sales questions: sales@datastead.com

Other questions: contact@datastead.com

Contact form

<https://www.datastead.com/contact/>

User guide

Created with the Standard Edition of HelpNDoc: [Experience the Power and Simplicity of HelpNDoc's User Interface](#)

Licensing

Licensing

[Prev](#)[Next](#)

Licensing

Description

Our license is a commercial, per-developer(*), royalty-free license.
The application developed can be distributed on as many PCs, as needed without having to pay additional end user runtime fees.

(*)

- if you are developing as contractor for another company, a license must be purchased in the name of this company.
- if several developers are working concurrently on this SDK, the corresponding number of licenses must be purchased.

The licenses can be purchased from our online store:

<http://www.datastead.com/purchase.html>

[License Agreement](#).

Created with the Standard Edition of HelpNDoc: [Bring your WinHelp HLP help files into the present with HelpNDoc's easy CHM conversion](#)

Pre-requisties

Created with the Standard Edition of HelpNDoc: [Streamline Your Documentation Process with HelpNDoc's Project Analyzer](#)

Pre - requisities

Pre - requisities

[Prev](#)[Next](#)

Pre-requisties

Description

IP CAMERAS, URL SOURCES

To be able to decode IP cameras and URL sources, and record them without decompression/recompression(*), the [Datastead RTSP/RTMP/HTTP/ONVIF Source filter](#) is required.

(*) when VideoGrabber: [RecordingInNativeFormat](#) is enabled

RECORDING

Webcams, URL sources, screen, PCIe HD capture cards, video mixer:

To be able to record by using the Datastead codecs (h264, hevc, etc...), including the NVidia/AMD/Intel GPU encoders, the [Datastead Multipurpose Encoder](#) is required (starting from the v2.01, it does not run external processes)

See the [Datastead Encoder](#) chapter for more information

PLAYER

For the decoding of video files we recommend to install the [LAV Filters](#) that can be downloaded here: <https://github.com/Nevcairiel/LAVFilters/releases/>

If not using the LAV Filters, note that the Microsoft H264 decoder installed on Windows may produce glitches/pixelation when running from the debugger. However there is no problem out of the debugger.

INSTALLATION

All the filters above can be installed:

A) either by running the .exe the installer

B) either by copying the x86 and x64 folders directly under the folder where is located the application's .exe:

Example of file layout:

c:/appfolder/application.exe

c:/appfolder/x86

c:/appfolder/x64

where x86 and x64 are the folders containing the respective LAV filters binaries.

In this case TVideoGrabber will find them automatically.

C) either by copying the x86 and x64 folders to another folder, that is specified in the VideoGrabber.ExtraDLLPath property:

Example:

VideoGrabber.ExtraDLLPath = c:/codecfolder

File layout:

c:/appfolder/application.exe

c:/codecfolder/x86

c:/codecfolder

Created with the Standard Edition of HelpNDoc: [Why Microsoft Word Isn't Cut Out for Documentation: The Benefits of a Help Authoring Tool](#)

Platform-dependent properties

Created with the Standard Edition of HelpNDoc: [Effortlessly create a professional-quality documentation website with HelpNDoc](#)

Platform dependent properties

Platform dependent properties

[Prev](#)

[Next](#)

List of platform-dependent properties.

Description

The functions below return platform-dependent information.

[VideoDevices](#) : list of the video capture devices available

[VideoDevicesCount](#) : count of the current video capture devices available

[AudioDevices](#) : list of the current audio capture devices available

[AudioDevicesCount](#) : count of the current audio capture devices available

[VideoCompressors](#) : list of the current video encoders available

[VideoCompressorsCount](#) : count of the current video encoders available

[AudioCompressors](#) : list of the current audio encoders available

[AudioCompressorsCount](#) : count of the current audio encoders available

TVideoGrabber exposes the following properties, that are the indexes (in the lists above) of the current devices or compressors used by the TVideoGrabber component :

[VideoDevice](#) : selects the video capture device to be used during preview or recording

[AudioDevice](#) : selects the audio capture device to be used during preview or recording

[VideoCompressor](#) : selects the video compressor to be used during recording

[AudioCompressor](#) : selects the audio compressor to be used during recording

[VideoDeviceName](#) : returns the name of the current video capture device

[AudioDeviceName](#) : returns the name of the current audio capture device
[VideoCompressorName](#) : returns the name of the current video compressor
[AudioCompressorName](#) : returns the name of the current audio compressor

Created with the Standard Edition of HelpNDoc: [Generate EPub eBooks with ease](#)

Device-dependent properties

Created with the Standard Edition of HelpNDoc: [Streamline your documentation process with HelpNDoc's WinHelp HLP to CHM conversion feature](#)

Device dependent properties

Device dependent properties

[Prev](#)

[Next](#)

Explanations about device-dependent properties.

Description

Most of the video capture device properties are device-dependent.

This means that their content changes when the current video capture device changes.

For this reason these properties are not published, because setting a value from the Object Inspector does allow to specify the concerned video device (selected by the [VideoDevice](#) property).

TVideoGrabber saves and retrieves these values for each capture device in the registry in HKEY_CURRENT_USER, under the key. if the "Datastead" key is destroyed it will be recreated automatically and the device settings will be reset to their default values.

The content of these properties is refreshed when the [OnVideoDeviceSelected](#) event occurs after selecting the video capture device by assigning a new value to [VideoDevice](#) .

The principle is the same for the audio device ([OnAudioDeviceSelected](#) and [AudioDevice](#)).

E.g., we have 2 video capture devices on our platform.

- the 1st video device has 3 inputs: "Tuner", "Composite" and "SVideo",
- the 2nd device has 2 composite inputs: "Composite 0" and "Composite 1".

a) on the first video capture device (VideoDevice = 0), the VideoInputs string list returns the following values: "Tuner", "Composite", "SVideo"

The index value VideoInput = 1 means that the Composite video input has been selected on this video capture device.

b) on the 2nd video capture device (VideoDevice = 1), the VideoInputs string list returns the following values: "Composite 0", "Composite 1".

Now the index value VideoInput = 1 means that the "Composite 1" input has been selected on this video capture device.

Moreover, a VideoInput = 2 value that means "SVideo" on the 1st device has no meaning and is out of range on the 2nd device.

Thus, the edit fields, listboxes and indexes of any application using TVideoGrabber MUST be refreshed when the [OnVideoDeviceSelected](#) or [OnAudioDeviceSelected](#) event occurs, otherwise their content will no longer reflect the current values of the TVideoGrabber properties.

We find mainly 2 types of properties:

- single values
- list / index based values, that can be easily identified as they are composed of a string list, an index in the list, and a list count. e.g.: VideoInput, VideoInputs and VideoInputsCount.

The main device-dependent properties related to the video capture device are:

[AnalogVideoStandard](#)

[AnalogVideoStandards](#)
[AnalogVideoStandardsCount](#)
[CameraControlAuto](#)
[FrameRate](#)
[IsAnalogVideoDecoderAvailable](#)
[IsCameraControlAvailable](#)
[IsDigitalVideoIn](#)
[IsHorizontalSyncLocked](#)
[IsTimeCodeReaderAvailable](#)
[IsTVAudioAvailable](#)
[IsTVTunerAvailable](#)
[IsDVCommandAvailable](#)
[IsVideoControlAvailable](#)
[IsVideoCrossbarAvailable](#)
[IsVideoInterlaced](#)
[IsVideoPortAvailable](#)
[IsVideoQualityAvailable](#)
[IsWDMVideoDriver](#)
[VCRHorizontalLocking](#)
[VideoInput](#)
[VideoInputs](#)
[VideoInputsCount](#)
[VideoQualityAuto](#)
[VideoSize](#)
[VideoSizes](#)
[VideoSizesCount](#)
[VideoSubtype](#)
[VideoSubtypes](#)
[VideoSubtypesCount](#)

The main device-dependent properties related to the audio capture device are:

[AudioInputBalance](#)
[AudioInputLevel](#)
[AudioInput](#)
[AudioInputs](#)
[AudioInputsCount](#)
[IsAudioCrossbarAvailable](#)

Created with the Standard Edition of HelpNDoc: [Benefits of a Help Authoring Tool](#)

Video sources

Created with the Standard Edition of HelpNDoc: [Create Professional CHM Help Files with HelpNDoc's Easy-to-Use Tool](#)

Video sources supported for preview and recording

| Video sources supported for preview and recording | Prev | Next |
|---|----------------------|----------------------|
|---|----------------------|----------------------|

Video sources supported for preview and recording

Description

Video sources supported for preview and recording

The [VideoSource](#) property selects the video source to use:

vs_VideoCaptureDevice:

selects the video capture devices available on the current platform (by default), then the video capture device is selected by the [VideoDevice](#) property,

vs_VideoFileOrURL:

selects a file, URL or a static playlist specified by the [VideoSource_FileOrURL](#) property, for preview or recording (for playback with trackbar handling use [OpenPlayer](#) and the [Player](#) functions).

vs_ScreenRecording:

selects the screen as video source (to perform screen recording),

vs_Mixer:

In this mode the component mixes several other TVideoGrabber components. See [How to mix several video sources into one a single one](#)

vs_JPEGsOrBitmaps:

the video source are bitmap handles, or BMP files or JPEG files of the same format passed to the [OnVideoFromBitmaps_NextFrameNeeded](#) event.

See [Video clips built on the fly by passing bitmap handles, BMP or JPEG files](#).

vs_VideoFromImages:

the video source is built from a set of bitmaps (BMP or JPEG files). See [Video clip from bitmaps: Overview](#).

See Also

[VideoSource](#) [VideoSources](#) [VideoSourcesCount](#) [TVideoSource](#)

Created with the Standard Edition of HelpNDoc: [Full-featured multi-format Help generator](#)

USB and PCI video capture devices

Created with the Standard Edition of HelpNDoc: [Transform Your Help Documentation Process with a Help Authoring Tool](#)

Video capture devices

Video capture devices

[Prev](#)
[Next](#)

Video capture devices overview.

Description

Selecting a video capture device

First of all set [VideoSource](#) = **vs_VideoCaptureDevice** to select the video capture devices as video source.

Then select the current video capture device by assigning the [VideoDevice](#) property, which is an index in the [VideoDevices](#) property, that contains [VideoDevicesCount](#) items.

It is possible to select the video capture device programmatically by its name (as it appears in the [VideoDevices](#) list), e.g.:

```
VideoGrabber.VideoDevice = VideoGrabber.VideoDeviceIndex ("Microsoft DV Camera and VCR -
```

The name of the current video capture device is reported by [VideoDeviceName](#) .

When a video capture device is selected, its [device-dependent properties](#) are reloaded from the registry and the [OnVideoDeviceSelected](#) event occurs.

Note: these registry settings can be reset to their default values by invoking [ResetVideoDeviceSettings](#) while the component is inactive. This is rarely needed, this may be useful if the current settings shows a black video.

Identifying the video capture device by its hardware identifier (instead of its manufacturer's name)

When using several video capture devices of the same brand and model that can't be distinguished in the VideoDevices list, use [VideoDeviceIndexFromId](#) that returns the index of the device in the [VideoDevicesId](#) list.

E.g.:

```
VideoGrabber.VideoDevice = VideoGrabber.VideoDeviceIndexFromId
( "devicepnpusbvid046dpid0826mi02739103c9a0000265e8773d-8f56-11d0-a3b9-
00a0c9223196globalHD Webcam C525R" );
```

Connecting/disconnecting a video capture device

When a video capture device is connected or disconnected, the [OnDeviceArrivalOrRemoval](#) event occurs. This event reports the name and index of the video capture device in the [VideoDevices](#) list.

After the video capture device has been disconnected, [IsVideoDeviceConnected](#) returns true, until the device is reconnected.

If the related device is used by preview or recording, the [OnDeviceLost](#) event occurs when the device is disconnected, and the preview or recording stops.

When a video capture device is disconnected, it is not removed from the [VideoDevices](#) list, in order to prevent the complexity of having to manage the shifted indexes of the video captures whose indexes are above the index of the device currently removed (because VideoDevice selects the current video device in the VideoDevices list).

When TVideoGrabber is running, the rules are the following:

1. when a device is turned on, the device is added at the bottom of the VideoDevices list. The OnDeviceArrivalOrRemoval event occurs. Its IsDeviceArrival parameter reports "true", and its DeviceIndex parameter reports the index of the video device added in the list.
2. when a device is turned off, the device remains in the VideoDevices list. The OnDeviceArrivalOrRemoval event occurs. Its IsDeviceArrival parameter reports "false", and its DeviceIndex parameter reports the index of the video device turned off.
3. if a device turned off (case 2. above) is turned on again, the VideoDevices list does not change. The OnDeviceArrivalOrRemoval event occurs. Its IsDeviceArrival parameter reports "true", and its DeviceIndex parameter reports the index the video device had when it was turned off.

You can retrieve current the state of any video capture devices (connected or not) by testing [IsVideoDeviceConnected](#) (DeviceIndex).

E.g. IsVideoDeviceConnected (3) will return "true" if the device having the index 3 in the VideoDevices list is active and "false" if the device has been turned off.

PID and VID identifiers of USB video capture devices

The PID and VID identifiers of the USB video capture devices are included in the [VideoDevicesId](#) list that has the same number of items than the [VideoDevices](#) list (the number of items returned by [VideoDevicesCount](#))

You can retrieve them as follows:

- assign the [VideoDevicesId](#) list to a StringList (e.g. IdList)
- read the IdList[VideoGrabber.VideoDevice] string value
- if the device is USB you will find the "pid..." and "vid..." in the string returned.

WDM vs VFW drivers

More information [here](#).

See Also

[WDM drivers](#) [TOnDeviceArrivalOrRemoval](#) [AutoConnectRelatedPins](#) [CurrentFrameRate](#) [DeliveredFrames](#) [DroppedFrames](#) [FixFlickerOrBlackCapture](#) [IsDVDevice](#) [IsVideoControlAvailable](#) [IsVideoCrossbarAvailable](#) [IsVideoDeviceConnected](#) [IsWDMVideoDriver](#) [OnDeviceArrivalOrRemoval](#) [OnDeviceLost](#) [OnNoVideoDevices](#) [OnVideoDeviceSelected](#) [RefreshDevicesAndCompressorsLists](#) [ResetVideoDeviceSettings](#) [ShowDialog](#) [VideoDevice](#) [VideoDeviceIndex](#) [VideoDeviceIndexFromId](#)

[VideoDeviceName](#) [VideoDevices](#) [VideoDevicesCount](#) [VideoDevicesId](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Easily create PDF Help documents](#)

Preview

Preview

[Prev](#)

[Next](#)

Live preview overview.

Description

Preview of video capture devices

1. select the video source

First of all, set [VideoSource](#) = vs_ **VideoCaptureDevice** to select the video capture devices as video source

2. select the video capture device (optional if you have only one video capture device)

By default the 1st video capture in the [VideoDevices](#) list will be used ([VideoDevice](#) index = 0).

So you may have to select the video capture device by assigning the [VideoDevice](#) property with the index of the video capture device in the [VideoDevices](#) list.

If you need to select it by its name, [VideoDeviceIndex](#) ("...name of your video capture device...") will return the index of the name in the [VideoDevices](#) list.

E.g.

VideoGrabber.VideoDevice = VideoGrabber.VideoDeviceIndex ("...name of your video capture device...")

3. Start / stop / pause / resume preview

Preview can be started by using [StartPreview](#) and stopped by using [StopPreview](#).

After starting preview, it can be paused with [PausePreview](#), and resumed with [ResumePreview](#).

Frame progress

During preview, information about each video frame is reported by the [OnFrameProgress](#) event.

It is possible to get information about the current frame by invoking [GetFrameInfo](#) or [GetFrameInfoString](#).

Frame capture

Frames can be captured to TBitmap, BMP files or JPEG files, one by one or automatically by using the [frame grabber](#). See the [Frame capture](#) chapter for more information.

Graphics and text overlays

It is possible to draw text, shapes, bitmaps or graphics over video frames. See the [Graphics and text overlays](#) chapter.

Display options

See [Video window](#), [Dual display](#), [Transparency](#), [Third-party video renderer filter](#)

Created with the Standard Edition of HelpNDoc: [Upgrade your help files and your workflow with HelpNDoc's WinHelp HLP to CHM conversion](#)

Video quality

Video quality

[Prev](#)

[Next](#)

Video quality overview.

Description

The video quality settings allow to set brightness, hue, saturation, etc...

They are available on the [current video capture device](#) whether [IsVideoQualityAvailable](#) returns true.

The possible video quality settings are enumerated by the [TVideoQuality](#) type.

For each video quality setting, it is possible to :

- retrieve the current value with [VideoQualityValue](#)
- retrieve the "auto" mode with [VideoQualityAuto](#)
- retrieve the default value with [VideoQualityDefault](#)
- retrieve the minimum value with [VideoQualityMin](#)
- retrieve the maximum value with [VideoQualityMax](#)
- retrieve the stepping value with [VideoQualityStep](#)
- set a new value with [SetVideoQuality](#).

Use of the [SetVideoQuality](#) parameters:

- if **SetAuto** is true, **SetDefault** and **SetValue** are ignored and the setting is switched in "auto" mode
- if **SetAuto** is false and **SetDefault** is true, **SetValue** is ignored and the setting is set to its default value,
- if **SetAuto** is false and **SetDefault** is false, the **SetValue** value is applied to the setting.

The default, minimum, maximum and stepping values are usually used to set trackbar ranges. See the *miscVideoQuality* form of the *MainDemo* project for sample code.

Note: these properties are [device-dependent](#) . Any form that uses these properties should be refreshed when the [OnVideoDeviceSelected](#) event occurs.

Look at the *MainDemo* project for sample code.

See Also

[IsVideoQualityAvailable](#) [IsVideoQualitySettingAvailable](#) [SetVideoQuality](#) [VideoQualityAuto](#)
[VideoQualityDefault](#) [VideoQualityMax](#) [VideoQualityMin](#) [VideoQualitySettings](#) [VideoQualityStep](#)
[VideoRendererPriority](#) [TVideoQuality](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly upgrade your WinHelp HLP help files to CHM with HelpNDoc](#)

Camera control

Camera control

[Prev](#)

[Next](#)

Camera control overview.

Description

General camera control settings

The camera control settings allow to control pan, tilt, roll, zoom, exposure, iris, etc...

They are available on the [current video capture device](#) whether [IsCameraControlAvailable](#) returns true.

The possible camera controls settings are enumerated by the [TCameraControl](#) type.

For each camera control setting, it is possible to :

- retrieve the current value with [CameraControlValue](#)
- retrieve the "auto" or "manual" mode with [CameraControlAuto](#)
- retrieve the default value with [CameraControlDefault](#)
- retrieve the minimum value with [CameraControlMin](#)
- retrieve the maximum value with [CameraControlMax](#)
- retrieve the stepping value with [CameraControlStep](#)

Use of the [SetCameraControl](#) parameters:

- if **SetAuto** is true, **SetDefault** and **SetValue** are ignored and the setting is switched in "auto" mode
- if **SetAuto** is false and **SetDefault** is true, **SetValue** is ignored and the setting is set to its default value,
- if **SetAuto** is false and **SetDefault** is false, the **SetValue** value is applied to the setting.

The default, minimum, maximum and stepping values are usually used to set trackbar ranges. See *the miscVideoQuality form of the MainDemo project* for sample code.

Note: these properties are [device-dependent](#). Any form that uses these properties should be refreshed when the [OnVideoDeviceSelected](#) event occurs.
See the MainDemo project for sample code.

Pan / Tilt position control

With some cameras it is possible to control the pan and tilt positioning by using [SendCameraCommand](#).

See Also

[TCameraControl](#) [CameraControlAuto](#) [CameraControlDefault](#) [CameraControlMax](#) [CameraControlMin](#) [CameraControlSettings](#) [CameraControlStep](#) [GetCameraExposure](#) [GetCameraExposureAsString](#) [IsCameraControlAvailable](#) [IsCameraControlSettingAvailable](#) [SendCameraCommand](#) [SetCameraControl](#) [SetCameraExposure](#)

Created with the Standard Edition of HelpNDoc: [Free CHM Help documentation generator](#)

Detection of the video signal

Detection of the video signal

[Prev](#)

[Next](#)

Detection of the video signal

Description

Detection of the video signal

Is it possible to detect if the video signal is present or not for each video frame, by invoking [IsVideoSignalDetected](#).

Created with the Standard Edition of HelpNDoc: [Maximize Your Documentation Efficiency with a Help Authoring Tool](#)

Blackmagic Decklink cards

Blackmagic Decklink cards

[Prev](#)

[Next](#)

Using BlackMagic Decklink cards

Description

Using BlackMagic Decklink cards

Declink specific settings

To activate the video capture of the Decklink card, the video VideoSize and VideoSubtype indexes must be set to 0, and the format must be specified by the VideoFormat index, e.g.:

```
VideoGrabber.VideoSubtype = 0 (default)
VideoGrabber.VideoSize = 0 (default)
VideoGrabber.VideoFormat = VideoGrabber.VideoFormatIndex ("HDYC 1920x1080 25 fps")
VideoGrabber.VideoInput = VideoGrabber.VideoInputIndex ("HDMI")
VideoGrabber.StartPreview()
```

Remark:

If there is no image (black video window), it is possible that the VideoFormat selected is not suitable.
To determine the proper video format, proceed as follows:

- run the pre-compiled MainDemo.exe
 - select the Declink card in the VideoDevices list
 - set VideoSubtype and VideoSize to default
 - select the video input in the "Video Inputs" list
 - select the video format in the "Video formats" list
 - start the preview
- then try each one of the video formats available in the "Video Formats" list, until you see the video capture.

Playout to the decklink renderer

The [VideoRendererExternal](#) property lets you activate the external renderer on the Blackmagic Declink card.

- to activate it, set [VideoRendererExternal](#) = **vre_BlackMagic_Decklink**
- to deactivate it, set [VideoRendererExternal](#) = **vre_None**

Note that the renderer requires a standard video size (e.g. 720x480 or 1920x1080).

By default TVideoGrabber resizes to 720x480.

To select the HD format, invoke [UseNearestVideoSize](#) (1920, 1080, true) before invoking OpenPlayer (to go back to the default setting invoke UseNearestVideoSize (0, 0, false))

Note that the Decklink renderer requires also a standard frame rate (e.g. 25 fps or 29.97 fps). If the clip has a non-standard frame rate it may fail to play on the Declink output.

Specifying the Decklink renderer to use when more than one Decklink card is installed in the same PC

If more than one Decklink card is used in the same PC, specify the card number to use to the [VideoRendererExternalIndex](#) property (in the 0..n-1 range).

Created with the Standard Edition of HelpNDoc: [Produce electronic books easily](#)

IP Cameras and other network streaming sources

Created with the Standard Edition of HelpNDoc: [Effortlessly create a professional-quality documentation website with HelpNDoc](#)

The Datastead RTSP/RTMP/HTTP/ONVIF DirectShow source filter

The Datastead RTSP/RTMP/HTTP/ONVIF DirectShow source filter

[Prev](#)

[Next](#)

Datastead RTSP/RTMP/HTTP/ONVIF DirectShow source filter Description

Datastead RTSP/RTMP/HTTP/ONVIF DirectShow source filter

Capturing live streams and IP cameras require to install our optional [RTSP/RTMP/HTTP/ONVIF DirectShow source filter](#).

It is not necessary to register the filter, you can just unzip the filter binaries (includede in x86 and x64 subfolders), either under the application .exe folder, either in any folder specified as [ExtraDLLPath](#) (see also the [installation information](#))

This filter is able to decode live streams through most of the common live protocols (ONVIF, RTSP, RTMP, UDP, RTP, SDP, HTTP, MMS, etc...) and at the same time save the native frames received (H264, H265/HEVC, AAC, etc...) to .MP4, .FLV, .AVI, .MKV or .MOV containers without transcoding.

If a new file name is specified during the recording, it closes the previous file and opens the new file without loosing frames between the 2 files.

When used in TVideoGrabber the configuration of the RTSP filter is done by TVideoGrabber, however it is possible to control most of the specific filter parameters by adding and combining them at the end of the

URL as follows:

```
VideoGrabber.VideoSource = vs_IPCamera
VideoGrabber.IPCameraURL = "rtsp://192.168.5.1/live.sdp>autoreconnect=0"
VideoGrabber.StartPreview();
Other example, preview/recording of any ONVIF URL:
VideoGrabber.VideoSource = vs_IPCamera
VideoGrabber.IPCameraURL = "onvif://192.168.5.1>buffer=0>autoreconnect=0"
VideoGrabber.SetAuthentication (at_IPCamera, "user", "password");
VideoGrabber.RecordingInNativeFormat = true
VideoGrabber.RecordingMethod = rm_MP4
VideoGrabber.StartRecording();
The possible optional parameters are listed in the "Parameter Identifiers" section of the help here:
http://www.datastead.com/products/rtsprtmpsrc/documentation.html
```

Created with the Standard Edition of HelpNDoc: [Easily create CHM Help documents](#)

ONVIF Cameras Discovery

ONVIF Cameras Discovery

[Prev](#)
[Next](#)

Discovering ONVIF cameras

Description

ONVIF Cameras Network Discovery

It is possible to discover the ONVIF cameras in Multicast mode, by scanning a range of IP addresses, or both by invoking both functions consecutively.

The discovery is performed in the background.

Starting the discovery

- to discover the ONVIF cameras through a multicast request, invoke:

[ONVIFDiscoverCameras_Multicast \(timeout_seconds\)](#)

- to discover the ONVIF cameras through a range of IP addresses, invoke:

[ONVIFDiscoverCameras_IPRange \(first_IP, last_IP, timeout_seconds\)](#)

Note: a timeout of 2 to 5 seconds should be sufficient.

Stopping the discovery before its completion

- eventually invoke [ONVIFCancelDiscovery\(\)](#) to stop the current background process, if needed.

Results of the discovery

- each camera discovered is notified by an event [OnONVIFDiscoveryCompletedNotification](#) (**dcs_CameraFound**, int CameraCount)

- the end of the multicast discovery is notified by an event [OnONVIFDiscoveryCompletedNotification](#) (**dcs_MulticastCompleted**, int CameraCount)

- the end of the IP range scanning is notified by an event [OnONVIFDiscoveryCompletedNotification](#) (**dcs_IPRangeCompleted**, int CameraCount)

From any of these events it is possible to invoke `ONVIFEnumCamerasDiscovered` for each camera whose index is specified by the `CameraIndex` parameter.

When `CameraIndex >=` the number of cameras the function returns false

The sample code is included in the Delphi and C# MainDemo projects.

Created with the Standard Edition of HelpNDoc: [Experience the Power and Ease of Use of a Help Authoring Tool](#)

ONVIF - Connecting to IP cameras through the ONVIF protocol

ONVIF - Connecting to IP cameras through the ONVIF protocol

[Prev](#)
[Next](#)

[Connecting to IP cameras through the ONVIF protocol](#)

Description

Connecting to IP cameras through the ONVIF protocol

Connecting to an ONVIF camera through the ONVIF protocol (instead of directly the rtsp://... URL) give access:

- to the ONVIF information of the camera
- to the PTZ and other ONVIF features

Quick ONVIF connection

- to quick connect (without support of PTZ and other ONVIF features), use the "onvifurl://" prefix, e.g.:

VideoGrabber.[IPCameraURL](#) = "onvifurl://192.168.5.1:8080"

- to connect with PTZ support and other ONVIF features, use the "onvif://" prefix, e.g.:

VideoGrabber.[IPCameraURL](#) = "onvif://192.168.5.1:8080"

RTSPS support

To connect through the RTSPS protocol (for IP cameras that implement it), use the "onvifs://" or "onvifsurl://" prefix.

- quick connect:

VideoGrabber.[IPCameraURL](#) = "onvifsurl://192.168.5.1:8080"

- normal connect:

VideoGrabber.[IPCameraURL](#) = "onvifs://192.168.5.1:8080"

Sample code

VideoGrabber.[VideoSource](#) = vs_IPCamera

VideoGrabber.[IPCameraURL](#) = "onvifurl://192.168.5.1:8080"

VideoGrabber.[SetAuthentication](#) (at_IPCamera, "user", "password");

VideoGrabber.StartPreview();

Created with the Standard Edition of HelpNDoc: [Upgrade Your Documentation Process with a Help Authoring Tool](#)

ONVIF - Retrieving camera information

ONVIF - Retrieving camera information

[Prev](#)[Next](#)

Retrieving ONVIF information

Description

Retrieving ONVIF information

The various ONVIF information can be retrieved as strings by invoking [ONVIFDeviceInfo](#) ([TONVIFDeviceInfo](#)):!

E.g.:

VideoGrabber.VideoSource = vs_IPCamera

VideoGrabber.IPCameraURL = "onvif://192.168.2.144"

VideoGrabber.SetAuthentication (at_IPCamera, "user", "password");

string SerialNumber = VideoGrabber.ONVIFDeviceInfo (onv_SerialNumber);

string PTZInfo = VideoGrabber.ONVIFDeviceInfo (onv_PTZInfo);

string PTZPresets = VideoGrabber.ONVIFDeviceInfo (onv_PTZPresets);

string XMLRecordings = VideoGrabber.ONVIFDeviceInfo (onv_XMLReplay);

string FullXMLInfo = VideoGrabber.ONVIFDeviceInfo (onv_XMLInfo);

Created with the Standard Edition of HelpNDoc: [Create iPhone web-based documentation](#)

ONVIF - PTZ

ONVIF - PTZ

[Prev](#)[Next](#)

ONVIF PTZ

Description

ONVIF PTZ

Absolute, relative and continuous Pan / Tilt / Zoom are supported, as well as Presets.

Note that some camera support only partial PTZ features (e.g. the continuous move but not the absolute move, etc...)

All the values of the PTZ functions are expressed as "Double" values.

Most of the positioning functions below include a SpeedRatio parameter, the SpeedRatio value is usually in the 0 .. 1.0 range.

Before invoking any of the functions below, first [start the preview of the IP camera by using an onvif:// URL](#).

Min and max values

To retrieve the min and max values of each PTZ axis, invoke [ONVIFPTZGetLimits](#)

The PTZ current positions are also returned as text by invoking [ONVIFDeviceInfo](#) (onv_PTZInfo)

Retrieving the current position

Invoke VideoGrabber.[ONVIFPTZGetPosition](#) to get the current pan, tilt, and zoom positions as double values

Continuous move

- to start a continuous invoke [ONVIFPTZStartMove](#), e.g.:

VideoGrabber.ONVIFPTZStartMove ("Pan", true, 0.5, 100)

. the 1st parameter can be "Pan", "Tilt" or "Zoom",

. the 2nd parameter specifies the direction or its opposite,

. the 3rd parameter specifies the speed,

. the 4th parameter specifies the duration of the continuous move. Note that some cameras do not implement it and go on moving until ONVIFPTZStopMove is invoked.

- to stop it, invoke [ONVIFPTZStopMove](#), e.g.:

VideoGrabber.ONVIFPTZStopMove ("Pan")

Absolute move

Invoke VideoGrabber.[ONVIFPTZSetPosition](#) (pan position, tilt position, zoom position, speed, **true**)

Relative move

Invoke VideoGrabber.[ONVIFPTZSetPosition](#) (relative pan position, relative tilt position, relative zoom position, speed, **false**)

Managing presets

The list of the existing presets is returned as a string by invoking [ONVIFDeviceInfo](#) (onv_PTZPresets)

To manage a preset, invoke VideoGrabber.[ONVIFPTZPreset](#) (PresetAction, PresetName)

- PresetAction can be "CREATE", "REMOVE" or "GOTO"

- PresetName can be any name (however some camera support only their own predefined preset names)

Note: to create a preset, first position the PTZ at the desired location with the positioning functions above, then invoke ONVIFPTZPreset ("CREATE", presetname) to create it.

Then, when needed, invoke ONVIFPTZPreset ("GOTO", presetname) to move the camera to the desired location.

Specific manufacturer commands

To send a such command, invoke VideoGrabber.[ONVIFPTZSendAuxiliaryCommand](#) (Command)

The supported commands are described in the manufacturer's user guide of the IP camera

Created with the Standard Edition of HelpNDoc: [Easy EPub and documentation editor](#)

ONVIF Replay

ONVIF Replay

[Prev](#)

[Next](#)

ONVIF REPLAY

Description

ONVIF REPLAY

To use the ONVIF Replay:

1) get the list of the recordings available on the ONVIF DVR or IP camera:

VideoGrabber.VideoSource = vs_IPCamera

VideoGrabber.IPCameraURL = "onvif://192.168.2.144"

VideoGrabber.SetAuthentication (at_IPCamera, "user", "password");

string XMLRecordings = VideoGrabber.ONVIFDeviceInfo (onv_XMLReplay);

2) extract the URL from XMLRecordings, corresponding to the recording that must be replayed ("uri" field)

3) pass the URL extracted (*) to IPCameraURL:

VideoGrabber.VideoSource = vs_IPCamera

VideoGrabber.IPCameraURL = "rtsp://..." // (*)

VideoGrabber.SetAuthentication (at_IPCamera, "user", "password");

VideoGrabber.StartPreview();

Note: if a seeking is required, add ">starttime=nnn" at the end of the replay URL (with nnn expressed in

milliseconds). E.g. to start the replay at 15 seconds (15000 ms):

rtsp://192.168.1.22/rtsp_tunnel?rec=1&vcd=2>**starttime=15000**

Example of XML data structure returned for 2 recordings:

```
<recordings>
<recording>
<id>Cam1Rec0</id>
<source>SRC_1_0</source>
<name>_____</name>
<description>Cam1Rec0</description>
<address>LOCAL_SRC_1_0</address>
<uri>rtsp://192.168.8.139/rtsp_tunnel?
rec=1&vcd=2&enableaudio=1&audio_mode=0&aacOut=1&silence=0</uri>
</recording>
<recording>
<id>Cam1Rec1</id>
<source>SRC_1_1</source>
<name>_____</name>
<description>Cam1Rec1</description>
<address>LOCAL_SRC_1_1</address>
<uri>rtsp://192.168.8.139/rtsp_tunnel?
rec=1&inst=2&vcd=2&enableaudio=1&audio_mode=0&aacOut=1&silence=0</uri>
</recording>
</recordings>
```

Created with the Standard Edition of HelpNDoc: [Benefits of a Help Authoring Tool](#)

ONVIF snapshot

ONVIF snapshot

[Prev](#)

[Next](#)

ONVIF snapshots

Description

ONVIF snapshots

To get just JPEG image from an IP camera, instead of starting the camera live stream, it is possible to request a JPEG snapshot, that is returned as a JPEG file or as a pointer to the JPEG memory buffer, or both.

Sample code to get a JPEG file:

VideoGrabber.[VideoSource](#) = vs_IPCamera

VideoGrabber.[OpenURLAsync](#) = true; // so the connection does not block the main thread (it's the default anyway)

VideoGrabber.[IPCameraURL](#) = "onvif://192.168.0.25:80" // if the port is not specified the default port 80 is used

VideoGrabber.[SetAuthentication](#) (at_IPCamera, "root", "admin")

VideoGrabber.[ONVIFSnapshot](#) (false, true, "c:/folder/snapshot.jpg")

Sample code to get an access to the JPEG snapshot buffer in memory:

VideoGrabber.[VideoSource](#) = vs_IPCamera

VideoGrabber.[OpenURLAsync](#) = true; // default, so the connection does not block the main thread

VideoGrabber.[IPCameraURL](#) = "onvif://192.168.0.25:80" // if the port is not specified the default port 80 is used

VideoGrabber.[SetAuthentication](#) (at_IPCamera, "root", "admin")

VideoGrabber.[OnRawVideoSample](#) = OnRawVideoSampleCallback

VideoGrabber.[ONVIFSnapshot](#) (true, false, "")

in the OnRawVideoSampleCallback event callback, the **pSampleBuffer** return a pointer to the JPEG snapshot in memory of a **SampleBufferSize** size.

Note: it is possible to get both the memory callback and the JPEG file as follows:

VideoGrabber.[ONVIFSnapshot](#) (true, true, "c:/folder/snapshot.jpg")

Preview and recording of IP cameras and URL sources

Preview and recording of IP cameras and URL sources

[Prev](#)

[Next](#)

Preview and recording of IP cameras and URL sources

Description

Preview and recording of IP cameras and URL sources

Note: with IP cameras supporting the ONVIF protocol no need to know RTSP URL syntax anymore, you can open the RTSP URL by specifying the onvif:// protocol following by the IP address or domain name and port

Preview only:

```
VideoGrabber.VideoSource = vs_IPCamera
VideoGrabber.OpenURLAsync = true; // default, so the connection does not block the main thread
VideoGrabber.IPCameraURL = "onvif://192.168.0.25:80" // if the port is not specified the default port 80 is used
VideoGrabber.SetAuthentication (at_IPCamera, "root", "admin")
VideoGrabber.AudioDeviceRendering = true // optional, if audio rendering is needed
VideoGrabber.FrameGrabber = fg_Disabled // if no overlay or frame capture is required, saves CPU
VideoGrabber.StartPreview()
```

Preview and recording:

```
VideoGrabber.VideoSource = vs_IPCamera
VideoGrabber.OpenURLAsync = true; // default, so the connection does not block the main thread
VideoGrabber.IPCameraURL = "onvif://192.168.0.25:80" // if the port is not specified the default port 80 is used
VideoGrabber.SetAuthentication (at_IPCamera, "root", "admin")
VideoGrabber.FrameGrabber = fg_Disabled // if no overlay or frame capture is required, saves CPU
VideoGrabber.RecordingInNativeFormat = true
VideoGrabber.RecordingMethod = rm_MP4
VideoGrabber.AudioRecording = true // optional, if the URL has audio and audio recording is needed
VideoGrabber.StartRecording()
Recording only (no decoding, saves CPU):
VideoGrabber.VideoSource = vs_IPCamera
VideoGrabber.OpenURLAsync = true; // default, so the connection does not block the main thread
VideoGrabber.IPCameraURL = "onvif://192.168.0.25:80" // if the port is not specified the default port 80 is used
VideoGrabber.SetAuthentication (at_IPCamera, "root", "admin")
VideoGrabber.RecordingInNativeFormat = true
VideoGrabber.RecordingMethod = rm_MP4
VideoGrabber.FrameGrabber = fg_Disabled
VideoGrabber.VideoRenderer = vr_None
VideoGrabber.StartRecording()
```

To close the current file and start recording to a new file while the recording is running, invoke [RecordToNewFileNow](#), this will generate a new file on the fly without losing frames.

Alternatively it is possible to enable the [RecordingTimer](#) to periodically close the current file and generate a new file.

In this case the file name is generated automatically, however the file names can be renamed on the fly from the [OnRecordingCompleted](#) event.

RTSP, RTMP, HTTP, UDP, MMS, RTP sources and other streaming protocols

The sample code is the same, e.g.:

```
IPCameraURL = "rtsp://192.168.5.25/axis-media/media.amp?videocodec=h264&audio=1"
IPCameraURL = "udp://192.168.0.25"
IPCameraURL = "mmsh://192.168.0.25"
IPCameraURL = "rtmp://192.168.0.25/live"
IPCameraURL = "rtp://192.168.0.25"
IPCameraURL = "http://192.168.0.25/url"
```


etc...

Created with the Standard Edition of HelpNDoc: [Free help authoring tool](#)

Video stream of IP camera with audio of PC microphone or other audio capture device

Video stream of IP camera with audio of PC microphone or other audio capture device

[Prev](#)

[Next](#)

Video stream of IP camera with audio of PC microphone or other audio capture device

Description

Capturing the video stream of an IP camera along with the audio of the soundboard microphone, or another audio capture device

This is possible starting from the version 7.1.4 of the [Datastead RTSP/RTMP/HTTP/ONVIF DirectShow Source filter](#) with the following sample code (the specific lines of code are in bold):

```
VideoGrabber.VideoSource = vs_IPCamera
```

```
VideoGrabber.AudioDevice = VideoGrabber.AudioDeviceIndex (...) (*)
```

```
VideoGrabber.AudioSource = as_UseExternalAudio
```

```
// VideoGrabber.AudioDeviceRendering = true // if needed to render
```

```
VideoGrabber.AudioRecording = true
```

```
VideoGrabber.RecordingMethod = ...
```

```
VideoGrabber....
```

```
VideoGrabber.StartRecording()
```

(*) replace (...) by the name of the audio device as it appears in the "audio devices" list of MainDemo.exe (the VideoGrabber.AudioDevice index is in the 0..n-1 range)

See Also

[IR Cut Filter of Axis cameras](#) [TDiscoveryCallbackStatus](#) [TOnONVIFDiscoveryCompletedNotification](#) [ExtraDLLPath](#) [GetLastErrorMessage](#) [IPCameraURL](#) [IsURLResponding](#) [IsURLVideoStreamAvailable](#) [OnONVIFDiscoveryCompletedNotification](#) [ONVIF_GetStr](#) [ONVIF_SetStr](#) [ONVIFCancelDiscovery](#) [ONVIFDiscoverCameras_IPRange](#) [ONVIFDiscoverCameras_Multicast](#) [ONVIFEnumCamerasDiscovered](#) [OpenURLAsync](#) [OpenURLAsyncStatus](#) [SendIPCameraCommand](#) [SetDatasteadFilterDllName](#) [SetIPCameraSetting](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly bring your documentation online with HelpNDoc](#)

IP cameras in MJPEG mode

IP cameras in MJPEG mode

[Prev](#)

[Next](#)

IP Cameras in MJPEG mode

Description

IP Cameras in MJPEG mode

TVideoGrabber supports natively the IP cameras in http Motion JPEG (MJPEG) or JPG mode listed on this page, with their specific URL to use.

Important note:

By default the HTTP URLs in MJPEG mode are handled by TVideoGrabber, and not by the Datastead RTSP filter.

If the Datastead RTSP/RTMP/HTTP filter is installed and you want to delegate it the processing of the HTTP/MJPEG URLs, set:

```
VideoGrabber.PlayerForcedCodec = "DTSTDRTSP"
```

before invoking StartPreview() or StartRecording().

(to go back to the default behavior set VideoGrabber.PlayerForcedCodec="")

Starting the IP camera preview or recording

To use an IP camera as video source:

- set [VideoSource](#) = vs_IPCamera
- set [IPCameraURL](#) = the URL of the IP camera (*the URL depends of the manufacturer*)
- invoke [SetAuthentication](#) (at_IPCamera, ".username..", "..password...") if needed
- then invoke [StartPreview](#)() or [StartRecording](#)()

E.g.:

```
VideoGrabber.VideoSource = vs_IPCamera
VideoGrabber.IPCameraURL = "http://192.168.0.25/axis-cgi/mjpg/video.cgi";
VideoGrabber.SetAuthentication (at_IPCamera, "root", "admin");
VideoGrabber.RecordingInNativeFormat = true
VideoGrabber.RecordingMethod = rm_MP4
VideoGrabber.AudioRecording = true
VideoGrabber.StartRecording()
```

All the TVideoGrabber properties and events related to the video capture devices apply to the IP cameras.

E.g.:

```
VideoGrabber.VideoSource = vs_IPCamera
VideoGrabber.IPCameraURL = "http://x.x.x.x/axis-cgi/mjpg/video.cgi?camera=&resolution=640x480"
VideoGrabber.StartPreview
```

Note:

- x.x.x.x must be replaced by the IP or the host name of your IP camera
- if the IP camera does not stream on the default 80 port, you have to specify the port after the IP, e.g. for the port 6015:
http://x.x.x.x:6015/axis-cgi/mjpg/video.cgi?camera=&resolution=640x480

Authentication

If an authentication is required to connect to the IP camera, there are 2 ways to set an username and password required:

- either by invoking [SetAuthentication](#) (at_IPCamera, "...username...", "...password...") before connecting to the IP camera,
- either through the [OnAuthenticationNeeded](#) event that will occur when connecting if [SetAuthentication](#) has not been invoked yet.

Frame rate

By default TVideoGrabber tries to stream the IP cameras at their maximal speed.
It uses a default frame rate of 30 fps that is therefore theoretical.

You can limit the frame rate by specifying a lower value to the [FrameRate](#) property (e.g. FrameRate = 1 will receive one frame by second).

Connection TimeOut

The connection timeout is set by default to 10000 (10 seconds), and the receive timeout to 5000 (5 seconds).

You can modify them as needed by invoking [SetIPCameraSetting](#).

PTZ control

You can send commands to IP cameras that include PTZ control by invoking [SendIPCameraCommand](#).

E.g.:

SendIPCameraCommand ("http://x.x.x.x/axis-cgi/com/ptz.cgi?move=home")

SendIPCameraCommand ("http://x.x.x.x/axis-cgi/com/ptz.cgi?move=up")

tc...

List of the IP camera URL formats to use, depending of the manufacturer

This list is not exhaustive. If you don't know the URL to use for your IP camera, feel free to contact us at support@datastead.com.

The URLs below must begin by http://ip:port or http://hostname:port

Examples for the axis camera below, with the following URL format: **/axis-cgi/mjpg/video.cgi**.

1) let's suppose the camera IP is 212.45.2.14, the URL to use is:

e.g. on the default port 80: http://212.45.2.14/axis-cgi/mjpg/video.cgi

e.g. if setup on the port 10120: http://212.45.2.14:10120/axis-cgi/mjpg/video.cgi

2) let's suppose the camera host name is myipcamera.net, the URL to use is:

e.g. on the default port 80: http://myipcamera.net/axis-cgi/mjpg/video.cgi

e.g. if setup on the port 9580: http://myipcamera.net:9580/axis-cgi/mjpg/video.cgi

A-Linking

/GetData.cgi

Airlink

/mjpeg.cgi

/cgi/mjpg/mjpeg.cgi

/cgi/jpg/image.cgi

Airlive

/video.mjpg

/mjpg/video.mjpg

Airwave

/cgi-bin/pusher.cgi

Arecont

/mjpeg?res=full&x0=0&y0=0&x1=100%&y1=100%&quality=12&doublescan=0&fps=1&ver=HTTP/1.1

/image?res=half&x0=0&y0=0&x1=1600&y1=1200&quality=15&doublescan=0

Aviosys

/GetData.cgi

Aviosys 9060-I

/cgi-bin/Stream?Video?Authorization=

Axis IP camera in MJPEG format

/axis-cgi/mjpg/video.cgi

/axis-cgi/mjpg/video.cgi?resolution=352x240 *(the resolution specified must be available on the IP camera, look at the camera video settings panel)*

Axis (IP video server, several cameras) in MJPEG format

/axis-cgi/mjpg/video.cgi?camera=1&resolution=320x240

/axis-cgi/mjpg/video.cgi?camera=2&resolution=320x240

/axis-cgi/mjpg/video.cgi?camera=3&resolution=320x240

/axis-cgi/mjpg/video.cgi?camera=4&resolution=320x240

/axis-cgi/mjpg/video.cgi?camera=quad&resolution=704x576 (4 x 4 layout of the 4 inputs)

Bowya ([here](#))
/video.cgi

Bosch
/snap.jpg

Canon (VB-C50, VB-C60, etc...)
/-wvhttp-01-/
/-wvhttp-01-/GetOneShot
/-wvhttp-01-/GetOneShot?frame_count=no_limit
/-wvhttp-01-/GetStillImage

Convision
/fullsize.push?camera=1&sleep=15

Digicom
/mjpeg.cgi

D-Link (DSC2121)
/video/mjpg.cgi

D-Link
/video/mjpg.cgi
/video.cgi
/mjpeg.cgi
/cgi-bin/video.jpg
/IMAGE.jpg
/cgi-bin/video.vam
/_gCVimage.jpg

EasyN
/video.cgi
/videostream.cgi
/videostream.cgi?resolution=8
/videostream.cgi?resolution=8&rate=13
/videostream.cgi?user=username&pwd=password

Edimax
/jpg/image.jpg
/mjpg/video.mjpg
/snapshot.cgi

Ego PT-200
/cgi-bin/sf.cgi

Foscam
/videostream.cgi
/snapshot.cgi

Fulicom FC-CR1060
/cgi-bin/sf.cgi

Gadspot
/Jpeg/CamImg.jpg
/GetData.cgi?Status=0

Goscam
/cgi-bin/Stream?Video?Acc=**USER**?Pwd=**PASSWORD**?webcamPWD=RootCookies00000
(replace *USER* and *PASSWORD* by their respective values)

Kingnow PT200
/cgi-bin/sf.cgi

Hamlet

http://www.hamletcom.com/ProductDetails.aspx?ProductCode_EQ_HNIPC30W/mjpeg.cgi

Intellinet

/jpg/image.jpg

or

/temp/image.jpg

(you may have to enable the direct HTTP image access in the setup of the camera)

Intellinet NCS18

/jpg/image.jpg

IP Cam waterproof Infrared IP Camera: <http://ipcam.en.ecplaza.net/11.asp>

/snapshot.cgi

IQeye

/now.jpg?snap=spush

JVC (e.g. VN-X35U/235U)

/api/video?encode=jpeg&framerate=15&boundary=on

Linksys

/img/snapshot.cgi?size=2

/img/video.mjpeg

/img/mjpeg.cgi

(MJPEG mode activated with SendIPCameraCommand ("/adm/file.cgi?h_videotype=mjpeg&todo=save"))

mms://x.x.x.x/img/video.asf

(ASF mode activated with SendIPCameraCommand ("/adm/file.cgi?h_videotype=mpeg4&todo=save"))

Linudix

/cgi-bin/nph-update_4ch.cgi?ch=1

Lumenera

/cgi-bin/nph-video

Marmitek

/cgi/mjpg/mjpeg.cgi

Predefined positions:

videograbber1.SendIPCameraCommand('http://[IPAddress]/cgi/admin/ptctrl.cgi?action=move&Cmd=Position1');

videograbber1.SendIPCameraCommand('http://[IPAddress]/cgi/admin/ptctrl.cgi?action=move&Cmd=Position2');

videograbber1.SendIPCameraCommand('http://[IPAddress]/cgi/admin/ptctrl.cgi?action=move&Cmd=home');

...

Pan/tilt control:

videograbber1.SendIPCameraCommand('http://[IPAddress]/cgi/admin/ptctrl.cgi?action=move&Cmd=up');

videograbber1.SendIPCameraCommand('http://[IPAddress]/cgi/admin/ptctrl.cgi?action=move&Cmd=right');

videograbber1.SendIPCameraCommand('http://[IPAddress]/cgi/admin/ptctrl.cgi?action=move&Cmd=down');

videograbber1.SendIPCameraCommand('http://[IPAddress]/cgi/admin/ptctrl.cgi?action=move&Cmd=left');

Mobotix

/record/current.jpg

/control/faststream.jpg?stream=full

/faststream.jpg?stream=full&fps=1.0 (1 fps)

/faststream.jpg?stream=full&fps=3.0 (1 fps)

/faststream.jpg?stream=full&fps=0 (max frame rate)

Moxa

//cgi-bin/video.jpg

Panasonic

/nphMotionJpeg?Resolution=640x480&Quality=Clarity
 /cgi-bin/nphContinuousServerPush
 /SnapshotJPEG?mode=Refresh
 /cgi-bin/camera

Pixord

/Getimage.cgi
 /Getimage?camera=1&fmt=full (full size)
 /Getimage?camera=1&fmt=qsif (half size)
 /Getimage?camera=1&fmt=sif (quarter size)

PLANET ICA-108

/jpg/image.jpg

Qnap

/cgi/mjpg/mjpeg.cgi

Samsung SNB

/video?submenu=mjpg
 /video?submenu=jpg

Sanyo

/liveimg.cgi?serverpush=1 (*MJPEG mode*)
 /liveimg.cgi (*Jpeg mode*)

Sharkx

/stream.jpg

Silicon

/snapshot.cgi

Skyway Security

/GetData.cgi?Status=0
 /Jpeg/CamImg.jpg

Sony

/image
 /image?speed=0
 /oneshotimage.jpg

Shenzen Sunsky S-NC-0201: <http://www.sun-usb.com/product/details/S-NC-0201.htm>

/cgi-bin/sf.cgi

Surecom

/mjpeg.cgi

Swann IP-3G ConnectCam 1000

/cgi/jpg/image.cgi

Topcom

http://www.topcom.net/fiche.asp?p_EQ_wirelessipcam2000
 /mjpeg.cgi

Toshiba

/__live.jpg?&&&
 getstream.cgi?10&10&&&10&0&0&0&0

TP-Link

/jpg/image.jpg
 /video.mjpg
 (the "/video.mjpg" MJPEG streaming is available only after disabling the RTSP streaming and rebooting)

the camera)

Trendnet: <http://trendnet.com/products/products.asp?cat=48>

/goform/video (e.g. TV-IP201W)
 /goform/video2 (e.g. TV-IP201W)
 /cgi/mjpg/mjpg.cgi (e.g. TV-IP410)
 /GetData.cgi
 /image.jpg

Vilar

/cgi-bin/sf.cgi

Vivotek

/video.mjpg (IP71.., IP 73.., VS71.. cameras. Be sure to enable the MJPG mode in the video settings)
 /cgi-bin/video.jpg
 /cgi-bin/viewer/video.jpg

Y-Cam

/stream.jpg

Zavio

/jpg/image.jpg

See Also

[Opening a clip or an IP URL from a background thread without blocking the main thread](#) [Player features](#)
[TOnThreadSync](#) [TThreadSyncPoint](#) [EnableThreadMode](#) [OnThreadSync](#)

Created with the Standard Edition of HelpNDoc: Modernize your help files with HelpNDoc's WinHelp HLP to CHM conversion tool

IR Cut Filter of Axis cameras

IR Cut Filter of Axis cameras

[Prev](#)

[Next](#)

IR Cut Filter of Axis cameras

Description

It is possible to control the IR Cut Filter of IP cameras through the [Datastead RTSP/RTMP/HTTP/ONVIF DirectShow Source Filter](#) v7.2.2.1 or newer, that must be installed first.

First start the preview of the Axis IP camera with the following sample code, e.g.:

```
VideoGrabber.VideoSource = vs_IPCamera
VideoGrabber.IPCameraURL = "rtsp://192.168.5.29/axis-media/media.amp?videocodec=h264&audio=1";
VideoGrabber.SetAuthentication (at_IPCamera, "root", "password");
```

then, once the preview is running:

- to set the IR Cut Filter state, invoke one of the following commands:

```
VideoGrabber.ONVIF_SetStr ("RTSP_Source_Axis_IrCutFilter_str", "enabled")
```

or

```
VideoGrabber.ONVIF_SetStr ("RTSP_Source_Axis_IrCutFilter_str", "disabled")
```

or

```
VideoGrabber.ONVIF_SetStr ("RTSP_Source_Axis_IrCutFilter_str", "auto")
```

- to retrieve the IR Cut Filter state, invoke:

```
function VideoGrabber.ONVIF_GetStr ("RTSP_Source_Axis_IrCutFilter_str", Value) : Boolean
```

Value returns one of the following values: "enabled", "disabled", or "auto"

See Also

[TDiscoveryCallbackStatus](#) [TOnONVIFDiscoveryCompletedNotification](#) [TONVIFDeviceInfo](#) [Video stream of IP camera with audio of PC microphone or other audio capture device](#) [ExtraDLLPath](#) [GetLastErrorMessage](#)
[IPCameraURL](#) [IsURLResponding](#) [IsURLVideoStreamAvailable](#) [OnONVIFDiscoveryCompletedNotification](#)
[ONVIF_GetStr](#) [ONVIF_SetStr](#) [ONVIFCancelDiscovery](#) [ONVIFDeviceInfo](#) [ONVIFDiscoverCameras](#) [IPRange](#)

[ONVIFDiscoverCameras](#) [Multicast ONVIFEnumCamerasDiscovered](#) [ONVIFPTZGetLimits](#) [ONVIFPTZGetPosition](#) [ONVIFPTZPreset](#) [ONVIFPTZSendAuxiliaryCommand](#) [ONVIFPTZSetPosition](#) [ONVIFPTZStartMove](#) [ONVIFPTZStopMove](#) [ONVIFSnapShot](#) [OpenURLAsync](#) [OpenURLAsyncStatus](#) [SendIPCameraCommand](#) [SetDasteadFilterDllName](#) [SetIPCameraSetting](#)

Created with the Standard Edition of HelpNDoc: [Experience a User-Friendly Interface with HelpNDoc's Documentation Tool](#)

Asynchronous vs Asynchronous connection

Asynchronous vs Asynchronous connection

[Prev](#)
[Next](#)

Synchronous vs Asynchronous connection

Description

TVideoGrabber can connect to the IP cameras and URLs:

- synchronously if OpenURLAsync is disabled

In this case when invoking StartPreview() or StartRecording(), the function blocks until the connection succeeds or fails.

- synchronously if OpenURLAsync is enabled (default)

In this case, when invoking StartPreview() or StartRecording(), the function returns immediately so the main thread is not blocked and the application continues to respond to key and mouse events.

Once the connection is completed, the OnPreviewStarted() or OnRecordingStarted() event occurs to notify the application that the component is running.

If the connection fails, the [OnLog](#) event occurs with a "e_failed_to_start_preview" or "e_failed_to_start_recording" [LogType](#) error.

Example of generic StartPreview() / StartRecording() set of functions that handles the asynchronous connection result through the OnPreviewStarted() / OnRecordingStarted() and OnLog() events:

```
function StartPreview()

    VideoGrabber.OpenURLAsync = true // default
    VideoGrabber.VideoSource = vs_IPCamera
    VideoGrabber.IPCameraURL =
rtsp://192.168.0.25/axis-media/media.amp?videocodec=h264

    VideoGrabber.SetAuthentication (at_IPCamera, "root", "admin");
    VideoGrabber.StartPreview()

function StartRecord()

    VideoGrabber.OpenURLAsync = true // default
    VideoGrabber.VideoSource = vs_IPCamera
    VideoGrabber.IPCameraURL =
rtsp://192.168.0.25/axis-media/media.amp?videocodec=h264&audio=1

    VideoGrabber.SetAuthentication (at_IPCamera, "root", "admin");
    VideoGrabber.RecordingInNativeFormat = true
    VideoGrabber.RecordingMethod = rm_MP4
    VideoGrabber.AudioRecording = true
    VideoGrabber.StartRecording()
```



```
event function OnRecordingStartedEvent(Sender, Filename)

    ShowMessage (filename + " recording started!")

event function OnPreviewStartedEvent(Sender)

    ShowMessage ("preview started!")

event function OnLogEvent(Sender, Logtype, Severity, Errormsg)

    if (LogType = e_failed_to_start_recording)
        ShowMessage ("error: recording failed to start")

    else if (LogType = e_failed_to_start_preview)
        ShowMessage ("error: preview failed to start")
```

Created with the Standard Edition of HelpNDoc: [Effortlessly Create Professional Documentation with HelpNDoc's Clean UI](#)

Auto-reconnection

Auto-reconnection

[Prev](#)

[Next](#)

Auto-Reconnection

Description

Auto-Reconnection of an URL

- 1) If the IP camera auto-reconnection is enabled (default setting):
 - when the connection is lost, the [OnDeviceReconnecting](#) event occurs and the RTSP filter tries to reconnect to the IP camera
 - when the camera is connected again, the [OnDeviceReconnected](#) event occurs
 - 2) If the IP camera autoreconnection is disabled:
 - when the connection is lost, the [OnDeviceLost](#) event occurs and TVideoGrabber stops.
- To disable the auto-reconnection, add:
 >autoreconnect=0
 at the end of the URL, e.g.:
 VideoGrabber.IPCameraURL = "rtsp://192.168.5.1/live.sdp>autoreconnect=0"

Created with the Standard Edition of HelpNDoc: [Free Kindle producer](#)

NTP time of IP cameras

NTP time of IP cameras

[Prev](#)

[Next](#)

NTP time of IP cameras

Description

If the camera streams expose the NTP time, the NTP time of the current frame being displayed can be obtained as 100ns units:

- at any time by invoking `int64 NTPTIME = VideoGrabber.GetFrameInfo (0, fi_NTPFrameTime)`
- though the [TFrameInfo](#) structure pointer of the [OnFrameBitmap](#) event: `FrameInfo->NTPFrameTime`

Note :

- if the NTP server is configured on the IP camera, the typical time value should look like the [current Unix epoch time](#).

- if the NTP server is not configured the value may start from a given value, but **the NTP time of the current frame minus the NTP time of the first frame received** can be anyway used to calculate the current stream duration since the preview started.

Created with the Standard Edition of HelpNDoc: [Experience a User-Friendly Interface with HelpNDoc's Documentation Tool](#)

Using the TVideoGrabber http/mjpeg decoder when the Datastead RTSP filter is installed

Using the TVideoGrabber http/mjpeg decoder when the Datastead RTSP filter is installed

[Prev](#)
[Next](#)

Using the TVideoGrabber http/mjpeg decoder when the Datastead RTSP filter is installed

Description

When the [Datastead RTSP/RTMP/HTTP/ONVIF DirectShow Source Filter](#) is installed, the capture/decoding of HTTP streams is delegated to it.

To force TVideoGrabber to use its embedded HTTP/MJPEG decoder anyway, set:

VideoGrabber.PlayerForcedCodec := "HTTP_TVIG";

before invoking StartPreview() or StartRecording()

(This can be cancelled by setting VideoGrabber.PlayerForcedCodec := "")

Created with the Standard Edition of HelpNDoc: [Experience a User-Friendly Interface with HelpNDoc's Documentation Tool](#)

Audio

Created with the Standard Edition of HelpNDoc: [How to Protect Your PDFs with Encryption and Passwords](#)

Audio capture devices

Audio capture devices

[Prev](#)
[Next](#)

Audio capture devices overview.

Description

Audio Source

If [AudioSource](#) = **as_Default** (default value) the audio output of the video capture device will be used for recording (if this device exposes an audio output, e.g. like a DV camcorder), otherwise the audio output of the current audio capture device will be used.

If [AudioSource](#) is set to **as_UseExternalAudio**, the current audio capture device will be used for recording, even if the video capture device has an audio out.

Audio capture device

The current audio capture device is selected by [AudioDevice](#), which is an index in the [AudioDevices](#) list, that contains [AudioDevicesCount](#) items.

It is possible to select the audiop capture device programmatically by name by using the [FindIndexInListByName](#) function, e.g.:

The name of the current audio capture device is reported by [AudioDeviceName](#).

```
VideoGrabber.AudioDevice := VideoGrabber.FindIndexInListByName (VideoGrabber.AudioDevices
```

Note: [AudioDevices](#) and AudioDevices count are also available as global variables in Delphi and C++Builder.

Device-dependent properties

When an audio capture device is selected, its [device-dependent properties](#) are reloaded from the registry and the [OnAudioDeviceSelected](#) event occurs, therefore any control that uses e.g. the audio inputs should be refreshed from this event.

Audio inputs

After selecting an audio capture device, the [AudioInputs](#) lists returns the list of the audio inputs available for this audio capture device. An audio input can be selected by assigning the index of audio input in the list to the [AudioInput](#) property (in the 0..[AudioInputsCount](#) - 1 range).

Audio input level

The audio input level can be adjusted with [AudioInputLevel](#) (in the 0..65535) range.

Audio input balance

The audio input balance can be adjusted with [AudioInputBalance](#) in the -32768...32767 range (the center point is 0).

The audio input balance can be available or not, depending of the audio input. When preview or recording is running, [IsAudioInputBalanceAvailable](#) returns ts_True if the audio balance is available, and ts_False if not. If preview and recording are stopped, it returns ts_Undefined.

Audio recording

The recording of the current audio capture device is activated when [AudioRecording](#) is enabled.

Audio rendering

The audio rendering is activated when [AudioDeviceRendering](#) is enabled.

The audio volume can be adjusted with [AudioVolume](#) and the audio balance with [AudioBalance](#).

To mute the audio volume enable/disable the [MuteAudioRendering](#) property.

These settings affect only the volume and balance of the audio rendering, not the volume and balance of the audio recording.

Device connection/disconnection

When an audio capture device is connected or disconnected, the [OnDeviceArrivalOrRemoval](#) event occurs. This event reports the name and index of the audio capture device in the global [AudioDevices](#) list.

After an audio capture device has been disconnected, [IsAudioDeviceConnected](#) returns true, until the device is reconnected.

If the related device is used by preview or recording, the [OnDeviceLost](#) event occurs when the device is disconnected, and the preview or recording stops.

See Also

[TAudioFormat](#) [AssociateAudioAndVideoDevices](#) [AudioBalance](#) [AudioDevice](#) [AudioDeviceIndex](#) [AudioDeviceName](#) [AudioDeviceRendering](#) [AudioDevices](#) [AudioDevicesCount](#) [AudioFormat](#) [AudioFormats](#) [AudioInput](#) [AudioInputIndex](#) [AudioInputLevel](#) [AudioInputMono](#) [AudioInputs](#) [AudioInputsCount](#) [AudioSource](#) [IsAudioCrossbarAvailable](#) [IsAudioDeviceASoundCard](#) [IsAudioDeviceConnected](#) [IsAudioInputBalanceAvailable](#) [OnAudioBufferNegotiation](#) [OnAudioDeviceSelected](#) [SpeakerBalance](#) [SpeakerControl](#) [SpeakerVolume](#)

Created with the Standard Edition of HelpNDoc: [Upgrade your help files and your workflow with HelpNDoc's WinHelp HLP to CHM conversion](#)

Audio rendering

Audio rendering

[Prev](#)

[Next](#)

Audio rendering
Description

Enabling the audio rendering

The audio rendering is delivered when [AudioDeviceRendering](#) = true, excepted for the player for which the audio is rendered when [PlayerAudioRendering](#) = true

Volume and balance

The volume is usually adjusted with [AudioVolume](#), and the balance with [AudioBalance](#). These settings does not interact with the default sound card, unless the default sound card has been selected as [AudioDevice](#) in the [AudioDevices](#) list.

It is possible to modify the global audio volume and balance of the default sound card with [SpeakerVolume](#) and [SpeakerBalance](#).

Choosing the audio renderer

By default, the default DirectSound Audio Renderer is used ([AudioRenderer](#) = -1). To select a different audio renderer, simply assign to [AudioRenderer](#) the index of the audio renderer to use, choosen in the [AudioRenderers](#) list.

Selecting the right or left channel

It is possible to convert the right channel or the left channel as "mono" and to mute either the left, right or both channels, or convert a mono channel in stereo with [AudioChannelRenderMode](#).

Selecting the audio stream

When the clip has multiple audio streams it is possible to select one audio stream, see [AudioStreamNumber](#)

Created with the Standard Edition of HelpNDoc: [Converting Word Documents to eBooks: A Step-by-Step Guide with HelpNDoc](#)

Audio levels and VU-meters

Audio levels and VU-meters

[Prev](#)

[Next](#)

Audio levels and VU-meters

Description

Audio levels

The audio levels are returned by the [OnAudioPeak](#) event (type [TOnAudioPeak](#)) when the [AudioPeakEvent](#) property is enabled.

The event returns the peak values of the left and right channels in percentage or in DB.

VU-Meters

VU-Meters are integrated in the component (analog or bargraph) and can be easily displayed.

Proceed as follows:

1. put 2 panel (or image) controls on the form, one for the left channel, one for the right channel. Let's suppose they are named Panel1 and Panel2.

2. set VUMeter = vu_Analog (or vu_Bargraph)

3. assign the handle of the panel (or the image) controls as follows:

```
VideoGrabber.SetVUMeterSetting(VU_LEFT, vu_Handle, Panel1.Handle)
VideoGrabber.SetVUMeterSetting(VU_RIGHT, vu_Handle, Panel2.Handle)
```

4. you can adjust the other VU-Meter settings (mainly colors), if needed, by invoking [SetVUMeterSetting](#).

Note:

to detach the VU-Meters from their panel (or image) controls, invoke:

```
VideoGrabber.SetVUMeterSetting(VU_LEFT, vu_Handle, 0)
```

```
VideoGrabber.SetVUMeterSetting(VU_RIGHT, vu_Handle, 0)
```

VU-meters and bargraphs overlayed over the video frames

Up to 6 vu-meters/bargraphs are available (from index 0 to 5).

(for backward compatibility the VUMeter property is kept but it activates only the VUMeters 0 and 1)

Additionally, now the VU-Meters and bargraphs can display custom values updated in real-time, the custom values must be in the 0..100 range.

To activate any of the 6 VU-meters, instead of setting the VUMeter property, invoke [SetVUMeter_Enabled](#).

E.g. to activate a bargraph, invoke SetVUMeter_Enabled(index, vu_BargraphOverlay)

So it is possible to use;

- the 2 originals VU-meters (index 0 and 1)
- 2 additional VU-meters or bargraphs (index 2 and 3) that can display audio or custom values
- 2 additional VU-meters or bargraphs (index 4 and 5) that can display custom values ONLY

Example to activate the bargraphs 2 and 3:

```
VideoGrabber.SetVUMeter_Enabled(2, vu_BargraphOverlay);
```

```
VideoGrabber.SetVUMeterSetting(2, vu_Transparent, 0);
```

```
VideoGrabber.SetVUMeterSetting(2, vu_OverlayLeft, 10);
```

```
VideoGrabber.SetVUMeterSetting(2, vu_OverlayTop, 10);
```

```
VideoGrabber.SetVUMeterSetting(2, vu_OverlayWidth, 100);
```

```
VideoGrabber.SetVUMeterSetting(2, vu_OverlayHeight, 60);
```

```
VideoGrabber.SetVUMeter_Enabled(3, vu_BargraphOverlay);
```

```
VideoGrabber.SetVUMeterSetting(3, vu_Transparent, 0);
```

```
VideoGrabber.SetVUMeterSetting(3, vu_OverlayLeft, 200);
```

```
VideoGrabber.SetVUMeterSetting(3, vu_OverlayTop, 500);
```

```
VideoGrabber.SetVUMeterSetting(3, vu_OverlayWidth, 100);
```

```
VideoGrabber.SetVUMeterSetting(3, vu_OverlayHeight, 60);
```

and to use the bargraph 3 as custom bargraph:

```
VideoGrabber.SetVUMeterSetting(3, vu_CustomPercentValue, 40); // e.g. for 40%
```

Note: to customize the VU-meter or bargraph foreground and background color, pass the color as RGB hexadecimal value, e.g.:

Delphi:

```
SetVUMeterSetting(2, VidGrab.TVUMeterSetting.vu_NormalColor, $FFFF00)
```

C#:

```
SetVUMeterSetting(2, VidGrab.TVUMeterSetting.vu_NormalColor, 0xFFFF00)
```

VB:

```
SetVUMeterSetting(2, VidGrab.TVUMeterSetting.vu_NormalColor, CType(&Hffff00, IntPtr))
```

See Also

[TVUMeter](#) [TVUMeterSetting](#) [AudioBalance](#) [AudioChannelRenderMode](#) [AudioVolume](#) [MuteAudioRendering](#) [SpeakerBalance](#) [SpeakerControl](#) [SpeakerVolume](#) [VUMeter](#)

Created with the Standard Edition of HelpNDoc: [Streamline Your CHM Help File Creation with HelpNDoc](#)

Datastead encoder

Datastead encoder

[Prev](#)
[Next](#)

Datastead Encoder

Description

The Datastead Encoder is part of the Datastead Multipurpose Encoder package.

Once installed, TVideoGrabber is able to use its codecs and muxers to perform recording and streaming.

Created with the Standard Edition of HelpNDoc: [Effortlessly Convert Your Markdown Content with HelpNDoc](#)

using the Datastead Encoder

using the Datastead Encoder

[Prev](#)
[Next](#)

Datastead Encoder

Description

The Datastead Encoder is a new set of codecs allowing to:

- compress video in various formats including MPEG4, MJPEG, H264, HEVC, eventually by using the GPU encoder if available (NVidia, AMD and Intel Quicksync supported),
- compress audio in various formats including AAC, Mpeg2
- record to file in various container formats (.avi, .mp4, .mkv, etc...),
- stream in various formats like UDP, RTMP, RTP, etc...

The Datastead Encoder requires the [Datastead Multipurpose Encoder](#) version 2 or newer to be installed,

however it does not launch FFmpegLGPL.exe, nor requires to run any external process.

One or more Datastead Encoder instances can be associated with a given TVideoGrabber component (e.g. an encoder instance performing the recording and another instance performing the streaming of the same TVideoGrabber instance)

A Datastead Encoder instance can record what is being displayed/rendered to a file, or send a live stream to the network.

When adding an Encoder instance, it returns an unique ID that is then used to configure this instance.

INSTALLATION

To install it:

- option 1:

run the .exe installer located in the "Install" folder of the Multipurpose Encoder package. Then TVideoGrabber will locate the binaries automatically.

- option 2:

copy the x86 and x64 folders containing the encoder binaries:

- either directly under the folder where is located the application's ".exe", e.g.:

Example of layout:

c:/appfolder/application.exe

c:/appfolder/x86

c:/appfolder/x64

- either to another folder, that you specify with VideoGrabber.[ExtraDLLPath](#)

Example:

VideoGrabber.ExtraDLLPath = "c:/AnotherFolder"

and the corresponding layout:

c:/appfolder/application.exe

c:/AnotherFolder/x86

c:/AnotherFolder/x64

ACTIVATION AND CONFIGURATION

Note: the instance and parameters must be configured before starting the video.

RECORDING

The recording is described in [Recording through the Datastead Encoder](#)

STREAMING

The streaming is described in [Streaming through the Datastead Encoder](#)

CONFIGURATION OF THE ENCODER

The configuration is achieved through the [Encoder_SetInt](#) and [Encoder_SetStr](#) functions.

Example of various values that can be passed:

- to select the codec:

Encoder_SetStr (ENCODER_RECORDING_ID, Enc_Video_Codec, "mjpeg");

Encoder_SetStr (ENCODER_RECORDING_ID, Enc_Audio_Codec, "aac");

Encoder_SetStr (ENCODER_RECORDING_ID, Enc_Video_Codec, "mpeg4");

Encoder_SetStr (ENCODER_RECORDING_ID, Enc_Video_Codec, "mpeg2");

Encoder_SetStr (ENCODER_RECORDING_ID, Enc_Video_Codec, "jpeg");

Encoder_SetStr (ENCODER_RECORDING_ID, Enc_Video_Codec, "h264");

Encoder_SetStr (ENCODER_RECORDING_ID, Enc_Video_Codec, "hevc");

- to configure the GPU encoder (usually for the "h264", "hevc" or "mjpeg" codecs:

Encoder_SetInt (ENCODER_RECORDING_ID, , (int) Enc_GPU_Intel_QSV);

(possible values include Enc_GPU_None, Enc_GPU_Auto, Enc_GPU_Intel_QSV,

Enc_GPU_NVidia_NVENC, Enc_GPU_AMD_AMF)

- **to set the bit rate:**

Encoder_SetInt (ENCODER_RECORDING_ID, Enc_Video_Bitrate_kb, 2000);

- **to configure the bitrate control:**

Encoder_SetInt (ENCODER_RECORDING_ID, Enc_Video_rc_BufferSize_kb, 1000);

Encoder_SetInt (ENCODER_RECORDING_ID, Enc_Video_MinBitrate_kb, 2000);

Encoder_SetInt (ENCODER_RECORDING_ID, Enc_Video_MaxBitrate_kb, 2000);

See Also

[Datastead Encoder Pause/resume during recording](#) [Recording methods and properties](#) [Encoder_GetInt](#) [Encoder_Pause](#) [Encoder_Resume](#) [Encoder_SetInt](#) [Encoder_SetStr](#) [Encoders_CreateInstanceForRecording](#) [Encoders_CreateInstanceForStreaming](#) [Encoders_RemoveAllInstances](#) [Encoders_RemoveInstance](#) [SetDatasteadFilterDllName](#)

Created with the Standard Edition of HelpNDoc: [Transform Your Help Documentation Process with a Help Authoring Tool](#)

Recording

Created with the Standard Edition of HelpNDoc: [Effortlessly bring your documentation online with HelpNDoc](#)

Recording methods and properties

Recording methods and properties

[Prev](#)

[Next](#)

Recording overview.

Description

Recording methods and properties

Video and audio capture devices

- if [AudioRecording](#) is disabled, only the [VideoDevice](#) video source (chosen in the [VideoDevices](#) list) will be recording to the file.
- if [AudioRecording](#) is enabled, the current [VideoDevice](#) and the current [AudioDevice](#) (chosen in the [AudioDevices](#) list) will be captured.
- if the audio and video capture devices are different and the audio of the video clip recorded is not in sync with the video, enable [AudioSyncAdjustmentEnabled](#) and adjust the number of audio samples to delay with [AudioSyncAdjustment](#).

Video format

The video format is determined by the extension of the [RecordingFileName](#)

E.g. to record a MP4 video set .mp4 as extension, to record a Matroska video set .mkv as extension.

If the file name is generated automatically, specify the video format with the RecordingMethod, e.g.

[RecordingMethod](#) = rm_MP4 will record a ".mp4" video clip.

RECORDING BY USING THE [DATASTEAD ENCODER](#) (recommended)

See [Recording through the Datastead Encoder](#)

RECORDING BY USING THIRD-PARTY CODECS

- select the third-party codec by its name, e.g.:
- set [VideoCompressor](#) = [VideoCompressorIndex](#) ("x264vfw - H.264/MPEG-4 AVC codec")
- set [CompressionMode](#) = cm_CompressOnTheFly

Starting the recording immediately

- once the settings above applied, invoke [StartRecording\(\)](#)

Starting the recording preview but not writing immediately to the file

- once the settings above applied:

- set [HoldRecording](#) = true

- invoke [StartRecording\(\)](#)

this starts the video preview and recording but does not write yet to the file.

- later, to begin writing to the file, invoke [ResumeRecording\(\)](#)

Stopping writing to the current file and starting writing to a new file while the recording is running

Invoke [RecordToNewFileNow](#)

Stopping the recording

- invoke [Stop\(\)](#)

Playing a video currently being recorded

This is possible when recording in .MP4 or .MKV format by enabling the [VideoPlayableWhileRecording](#) property.

Delayed start/stop

The RecordingTimer property lets you start, stop or restart periodically the recording after an interval of time:

[RecordingTimer](#) = rt_Disabled : normal start/stop

[RecordingTimer](#) = rt_RecordToNewFile : a new recording file is generated each [RecordingTimerInterval](#) seconds

[RecordingTimer](#) = rt_StopRecording : the recording will stop after [RecordingTimerInterval](#) seconds

[RecordingTimer](#) = rt_StartRecording : the recording starts in preview mode only, the real recording will begin after [RecordingTimerInterval](#) seconds

Creating a new video clip on the fly

While the recording is running, invoke [RecordToNewFileNow](#) (*NewFileName*, true) and a new video clip will be created.

- either pass the new file name to be created, either pass an empty string to let TVideoGrabber generate the new file name automatically, according to the [AutoFileName](#) settings.

To create small video clips having the same duration:

- set [RecordingTimerInterval](#) to the desired period (expressed in seconds)
- set [RecordingTimer](#) = rt_RecordToNewFileNow

*Note that this feature is **not compatible** with:*

- the recompression after recording (be sure that you use CompressionMode = cm_NoCompression or cm_CompressOnTheFly),
- the use of a preallocated recording file (be sure that PreallocCapfileEnabled = false)

Specifying a maximum file size

To limit the recording file size, specify a [RecordingFileSizeMaxInMB](#) value > 0.

During the recording, each time the size specified is exceeded, a new file is generated on the fly, according to the auto file name settings.

Holding the beginning of the recording for an accurate recording start time

By default, when invoking [StartRecording](#) the recording starts immediately, but building the recording graph requires a few seconds, so the moment when the recording starts after invoking [StartRecording](#) is never accurate.

To avoid this problem, enable the [HoldRecording](#) property before invoking [StartRecording](#).

When [HoldRecording](#) is enabled, the recording graph is built, and then the [OnRecordingReadyToStart](#) event occurs.

When this event occurs, you can invoke [ResumeRecording](#) to really start the recording, or [StopRecording](#) to cancel it.

E.g.:

```
...
procedure TForm1.Button1Click (Sender: TObject);
begin
  VideoGrabber1.HoldRecording := True;
  VideoGrabber1.StartRecording;
end;
...
procedure TForm1.VideoGrabberRecordingReadyToStart(Sender: TObject);
begin
  if ConditionToStartRecording then begin
    VideoGrabber.ResumeRecording;
  end
  else begin
```



```
VideoGrabber.StopRecording;
end;
end;
```

Recording file name

The recording file name can be chosen by assigning the [RecordingFileName](#) property before invoking [StartRecording](#).

If [RecordingFileName](#) is left blank, a file name is generated automatically according to the [AutoFileName](#), [AutoFileNameDateTimeFormat](#), [AutoFileNameMinDigits](#), [AutoFilePrefix](#) and [CaptureFileExt](#) properties.

By example if you customize the file name as follows:

```
VideoGrabber.AutoFileName = fn_DateTime;
VideoGrabber.AutoFileNameDateTimeFormat = "dd-mm-yy_hh-mm-ss"
VideoGrabber.AutoFilePrefix = "file_"
the recording file name will be file_13-02-13_15-27-45.avi
```

As soon as recording begins, this file name can be retrieved with [Last_Recording_FileName](#). When the recording ends, this file name is returned by the [OnRecordingCompleted](#) event.

Current file size of the video clip beeing recorded

The [RecordingKBytesWrittenToDisk](#) reports the current (growing) file size in KB while the recording is running.

Invoke this function periodically (e.g. every 10 seconds) to retrieve on the fly the current file size of the video clip being recorded.

When the recording ends, invoke [RecordingKBytesWrittenToDisk](#) from the [OnRecordingCompleted](#) event to get the final file size of the recorded clip.

You can find the corresponding sample code in the OnFrameProgress event of the MainDemo project.

Pausing / resuming the AVI recording

See [AVI pause/resume during recording](#)

Events

- if [HoldRecording](#) is enabled, when invoking [StartRecording](#) the [OnRecordingReadyToStart](#) event occurs, and then you can invoke [ResumeRecording](#) to really start to record.
- when the recording really begins, the [OnRecordingStarted](#) event occurs,
- when the recording ends, the [OnRecordingCompleted](#) event occurs.

Frame capture

It is possible to capture frames during recording when the [frame grabber](#) is enabled. See the [Frame Capture](#) chapter.

Frame overlay

It is possible to perform graphics or text overlay during recording when the [frame grabber](#) is enabled. See the [Frame Capture](#) chapter.

Quality of the video window during Recording

If the quality of the preview is not critical during recording, we recommend to set [VideoRenderer](#) = [vr_RecordingPriority](#) before invoking [StartRecording](#) to assign the resources prioritarily to the recording.

As the CPU load is critical during recording, in certain cases, preventing the video frames to be stretched to the video window size may save CPU, especially if the video frame is resized to a size larger than the original size of the video frame.

To prevent the video frames to be resized:

- if you want the control to be resized automatically, set [AdjustPixelAspectRatio](#) = false **and** [Display_AutoSize](#) = true
- if you want to specify yourself the size of the control, set [AdjustPixelAspectRatio](#) = false **and** [Display_AutoSize](#) = false **and** [Display_AspectRatio](#) = ar_NoResize

Back-timed recording (preroll)

It is possible to start recording with a preroll a few seconds before an event occurs.

E.g. when motion detection is used, this allows to capture the full video sequence by starting just before the motion detection event occurs.

See the [Back-timed recording \(preroll\)](#) chapter.

CPU and disk requirements

In order to minimize dropped frames, capturing to AVI in real time requires high CPU and disk availability, therefore:

- prefer fast disks / fast CPU platforms,
- prefer Windows 2000 / Windows XP platforms rather than Win98 or WinME platforms,
- prefer NTFS partitions rather than FAT32 partitions.

If several video clips are recorded at the same time and lot of dropped frames occur, we recommend to use SCSI disks for better disk performances.

Dropped frames

As video capture is synchrone, when the time required to record a given video frame becomes longer than the time between 2 video frames, frames are dropped.

An excessive amount of dropped frames cause jerky AVI playback and bad audio/video synchronization.

When dropped frames problems occur, act on the following points:

- check that [StoragePath](#) saves to a fast disk (avoid networked drives),
- disable the frame grabber on the AVI stream (set [FrameGrabber](#) = fg_PreviewStream), or disable it totally (set [FrameGrabber](#) = fg_Disabled)
- reduce the video size,
- reduce the frame rate ([FrameRate](#) property or [DVReduceFrameRate](#) for DV sources).

See Also

[Datastead Encoder](#) [Pause/resume during recording](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TCompressionType](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TOnVideoCompressionSettings](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioCompressor](#) [AudioCompressorIndex](#) [AudioCompressorName](#) [AudioCompressors](#) [AudioCompressorsCount](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [CompressionMode](#) [CompressionType](#) [Encoder](#) [GetInt](#) [Encoder](#) [Pause](#) [Encoder](#) [Resume](#) [Encoder](#) [SetInt](#) [Encoder](#) [SetStr](#) [Encoders](#) [CreateInstanceForRecording](#) [Encoders](#) [CreateInstanceForStreaming](#) [Encoders](#) [RemoveAllInstances](#) [Encoders](#) [RemoveInstance](#) [GetVideoCompressionSettings](#) [HoldRecording](#) [IsRecordingPaused](#) [Last](#) [Recording](#) [FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [OnVideoCompressionSettings](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [RefreshDevicesAndCompressorsLists](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetDatasteadFilterDllName](#) [SetMultiplexerFilterByName](#) [SetVideoCompressionDefaults](#) [SetVideoCompressionSettings](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoCompression](#) [KeyFrameRate](#) [VideoCompression](#) [PFramesPerKeyFrame](#) [VideoCompression](#) [Quality](#) [VideoCompression](#) [WindowSize](#) [VideoCompressor](#) [VideoCompressorIndex](#) [VideoCompressorName](#) [VideoCompressors](#) [VideoCompressorsCount](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: Full-featured Documentation generator

Recording through the Datastead Encoder

Recording through the Datastead Encoder

[Prev](#)

[Next](#)

Recording through the Datastead Encoder

Description

The installation and functions overview are described in the [Datastead Encoder](#) chapter.

It is possible:

1. either to record through the Datstead recording functions, allowing to use TVideoGrabber's OnRecording... events, in this case TVideoGrabber is started by invoking **StartRecording()**
2. either to record through an external Encoder instance (not managed by TVideoGrabber), in this case TVideoGrabber is started by invoking **StartPreview()**, that will send the audio/video samples to all the active Encoder instances.

SAMPLE CODE

1. sample code for a recording using the TVideoGrabber recording functions:
managed by TVideoGrabber, that controls it and raises the corresponding recording events

IMPORTANT NOTE:

The TVideoGrabber recording features (RecordingFileName / CompressionMode / StartRecording / PauseRecording / ResumeRecording / RecordToNewFileNow, etc...) invoke the internal Encoder instance with ID 0.

In this case, **do not invoke Encoders_CreateInstanceForRecording to pass the recording file name instead of setting RecordingFileName**, this is already controlled internally by TVideoGrabber.

Only Encoder_SetInt and Encoder_SetStr should be invoked with ID 0 to configure the parameters, as shown in the MainDemo project -> "multipurpose" tab.

If you prefer to configure and control the recording yourself by configuring a recorder instance with Encoders_CreateInstanceForRecording, run TVideoGrabber in preview mode only (by invoking StartPreview), it will pass the audio/video samples to the instance you have created, as described in the paragraph 2. below.

A) STARTING THE RECORDING

```
#define ENCODER_RECORDING_ID 0 // the encoder with ID 0 is the TVideoGrabber's
embedded encoder, this will never change
// in the MainDemo project, "video source" tab
VideoGrabber.VideoDevice = VideoGrabber.VideoDeviceIndex ("c922 Pro Stream
Webcam");
VideoGrabber.AudioDevice = VideoGrabber.AudioDeviceindex ("Microphone (C922
Pro Stream Webcam)");
VideoGrabber.VideoCompressor = VideoCompressorIndex ("Datstead Multipurpose
Encoder");
VideoGrabber.AudioCompressor = AudioCompressorIndex ("Datstead Multipurpose
Encoder");
// in the MainDemo project, "multipurpose" tab
VideoGrabber.Encoder_SetInt (ENCODER_RECORDING_ID, Enc_Video_Bitrate_kb,
2000);
VideoGrabber.Encoder_SetStr (ENCODER_RECORDING_ID, Enc_Video_Codec, "hevc");
VideoGrabber.Encoder_SetInt (ENCODER_RECORDING_ID, Enc_Video_Thread_Count, 4);
// default 1
// VideoGrabber.Encoder_SetInt (ENCODER_RECORDING_ID, Enc_Video_GPU_Encoder,
integer (Enc_GPU_NVidia_NVENC)); to enable the GPU encoding through NVidia (to
select any GPU available set Enc_GPU_Auto)
VideoGrabber.Encoder_SetStr (ENCODER_RECORDING_ID, Enc_Audio_Codec, "aac");
// in the MainDemo project, "recording" tab
VideoGrabber.RecordingMethod = rm_MP4; // to generate the file name
automatically,
// VideoGrabber.RecordingFileName = "VideoClip.mp4"; // if the file name is
specified, it determines the recording method, no need to set RecordingMethod
above
VideoGrabber.AudioRecording = true;
VideoGrabber.CompressionMode = cm_CompressOnTheFly;
VideoGrabber.StartRecording();
```

B) PAUSING THE RECORDING

```
VideoGrabber.PauseRecording();
```

C) RESUMING THE RECORDING

```
VideoGrabber.ResumeRecording();
```

D) RECORDING TO A NEW FILE

```
VideoGrabber.RecordToNewFileNow ("c:/folder/newvideoclip.mp4");
```

E) STOPPING THE RECORDING

```
VideoGrabber.Stop();
```

2. same sample code to recording using an external encoder instance

not managed by TVideoGrabber, that just sends the audio/video samples to the encoder(s)

A) STARTING THE RECORDING

```
int m_EncoderInstanceID = -1;
// in the MainDemo project, "video source" tab
VideoGrabber.VideoDevice = VideoGrabber.VideoDeviceIndex ("C922 Pro Stream Webcam");
VideoGrabber.AudioDevice = VideoGrabber.AudioDeviceIndex ("Microphone (C922 Pro Stream Webcam)");
m_EncoderInstanceID = VideoGrabber.Encoders_CreateInstanceForRecording ("VideoClip.mp4");
if (m_EncoderInstanceID > -1)
VideoGrabber.Encoder_SetInt (m_EncoderInstanceID, Enc_Video_Enabled_bool, 1);
VideoGrabber.Encoder_SetInt (m_EncoderInstanceID, Enc_Video_Bitrate_kb, 2000);
VideoGrabber.Encoder_SetStr (m_EncoderInstanceID, Enc_Video_Codec, "hevc");
VideoGrabber.Encoder_SetInt (m_EncoderInstanceID, Enc_Video_Thread_Count, 4); // default 1
// VideoGrabber.Encoder_SetInt (m_EncoderInstanceID, Enc_Video_GPU_Encoder, integer (Enc_GPU_NVidia_NVENC)); to enable the GPU encoding through NVidia (to select any GPU available set Enc_GPU_Auto)
VideoGrabber.Encoder_SetInt (m_EncoderInstanceID, Enc_Audio_Enabled_bool, 1);
VideoGrabber.Encoder_SetStr (m_EncoderInstanceID, Enc_Audio_Codec, "aac");
VideoGrabber.Encoder_SetInt (m_EncoderInstanceID, Enc_Audio_Bitrate_kb, 64);
VideoGrabber.AudioDeviceRendering = true;
VideoGrabber.StartPreview();
```

B) PAUSING THE RECORDING

```
VideoGrabber.Encoder_Pause (m_EncoderInstanceID);
```

C) RESUMING THE RECORDING

```
VideoGrabber.Encoder_Resume (m_EncoderInstanceID);
```

D) RECORDING TO A NEW FILE

bool OpenInPausedState = false; // to record to the new file immediately, otherwise the current file is closed and the recording to the next file will start when invoking Encoder_Resume

```
VideoGrabber.Encoder_NewOutputFile(m_EncoderInstanceID, "c:/folder/newvideoclip.mp4", OpenInPausedState);
```

D) STOPPING THE RECORDING ONLY

```
VideoGrabber.Encoder_NewOutputFile(m_EncoderInstanceID, "c:/folder/AnyNewFileName.mp4", true);
```

// this closes the current file and prepare for the next file. If you never invoke Encoder_Resume, the next file will

E) STOPPING THE PREVIEW AND RECORDING

```
VideoGrabber.Stop();
```

F) SETTING A NEW FILE NAME BEFORE RESTARTING THE VIDEO

bool OpenInPausedState = false; // to record to the new file immediately, otherwise the current file is closed and the recording to the next file will start when invoking Encoder_Resume

```
VideoGrabber.Encoder_NewOutputFile(m_EncoderInstanceID, "c:/folder/anothervideoclip.mp4", OpenInPausedState);
VideoGrabber.StartPreview();
```

Created with the Standard Edition of HelpNDoc: [Converting Word Docs to eBooks Made Easy with HelpNDoc](#)

Pause/resume during recording

Pause/resume during recording

[Prev](#)
[Next](#)

Pausing and resuming the recording

Description

Pause/resume during recording

It is possible to pause / resume the recording if [RecordingCanPause](#) has been enabled before invoking [StartRecording](#).

The recording can be paused by invoking [PauseRecording](#) , and resumed by invoking [ResumeRecording](#).

The [OnRecordingPaused](#) event notifies that the pause recording is completed (when the recording is resumed, a [OnRecordingStarted](#) event notification occurs).

Remark:

While the recording is paused the video source goes on streaming and previewing. If you need to pause the video source as well look at the "pausing/resuming the whole recording graph" chapter below.

Appending the video when resuming the recording or generating a new clip

- if [RecordingPauseCreatesNewFile](#) is **disabled**, the recording is appended to the current video clip.

Splitting the resume/pause sequences in several files

- if [RecordingPauseCreatesNewFile](#) is **enabled**, a new clip is generated automatically for each resume/pause sequence.

(if the [RecordingFileName](#) property is empty a new file name is generated automatically depending on the [AutoFileName](#) setting, otherwise the recording file name specified just before invoking [RecordingPauseCreatesNewFile](#) is used)

The new file is not created when resuming the recording, but before when the recording has previously been paused by invoking [PauseRecording](#).

Therefore, the file is already created and opened when invoking [ResumeRecording](#), "ready to go", so by this way frames will not be dropped by loosing time to create the new file.

So if the recording is stopped before invoking [ResumeRecording](#), the new empty file (that has just been created before, when [PauseRecording](#) has been invoked) is deleted.

Note: the pause / resume while compressing on the fly may not work properly with some video compressors.

Pausing/resuming the whole recording graph

During recording, when invoking [PauseRecording](#) / [ResumeRecording](#), the graph goes on running and previewing (only the recording is paused)

If you need to pause /resume the whole graph during the recording, invoke [PausePreview](#) / [ResumePreview](#) instead.

This can be useful e.g. to apply overlays on a video clip by recording it to a new file with the following settings:

```
VideoGrabber.VideoSource = vs_VideoFileOrURL
VideoGrabber.VideoSource_FileOrURL = name of the source clip
VideoGrabber.StartRecording ()
...
```

- then invoke later VideoGrabber.[PausePreview](#) to pause the recording of the video clip, so you can prepare and/or update the overlays,

- then invoke later VideoGrabber.[ResumePreview](#) to go on recording from the location where the clip has been paused.

See Also

[Datastead Encoder Recording methods and properties](#) [Encoder_GetInt](#) [Encoder_Pause](#) [Encoder_Resume](#) [Encoder_SetInt](#) [Encoder_SetStr](#) [Encoders_CreateInstanceForRecording](#) [Encoders_CreateInstanceForStreaming](#) [Encoders_RemoveAllInstances](#) [Encoders_RemoveInstance](#) [SetDatasteadFilterDllName](#)

Created with the Standard Edition of HelpNDoc: [Upgrade Your Documentation Process with a Help Authoring Tool](#)

Preroll recording (Backtimed recording)**Preroll recording (Backtimed recording)**[Prev](#)[Next](#)

Pre-roll recording (starting with a back-timed delay)

Description**Preroll recording**

It is possible to record the file with a back-timed amount of video frames (pre-roll). This property is activated by specifying a [RecordingBacktimedFramesCount](#) value (greater than 0).

This feature is useful when the recording is started by an event (e.g. motion detected) and you want to start the recording of the AVI file a few moments before this event occurs.

When [RecordingBacktimedFramesCount](#) has a value > 0, TVideoGrabber uses a FIFO (first in, first out) memory buffer to store the latest video frames.

This property can be used e.g. with the [motion detection](#) features. When a motion is detected and the recording starts, this property allows you to have at the beginning of the recording the few moments before the motion detection event occurs.

By this way recording is shifted back in time and starts at the "current - n" video frame.

When starting the recording, the normal video frames are streamed "normally" until the buffer is filled out. Then, when the buffer is filled out, the [OnBacktimedFrameCountReached](#) event occurs to let you know that now current video frames are replaced by back-timed video frames..

Note that when invoking StopRecording the recording stops "as is", this means that the latest video frames in the buffer will be missing (only the backtimed frames are recorded), unless you stop the recording at the desired time + the delay corresponding to the frames in the buffer.

Note: be careful when assigning a number of frames to this property, this means that a corresponding buffer of video frames will be created in memory during recording.
 E.g. 100 frames in 320 x 240 x 24 bits = 320 x 240 x 3 x 100 = 22,500 Kb = 22 Mb of memory required during recording.

Requirements

- the [Frame grabber](#) must be set to fg_BothStreams
- the [RecordingInNativeFormat](#) must be disabled.
- the [RecordingMethod](#) must be set to rm_AVI.

Implementation

- to preview the back-timed video stream, set [FrameGrabber](#) = fg_BothStream. If you prefer to preview the normal video stream, set [FrameGrabber](#) = fg_CaptureStream.
- set [RecordingInNativeFormat](#) = false,
- set [HoldRecording](#) to true to start recording in preview mode and hold AVI writing,
- call [StartRecording](#) to put the recording in preview mode and start to fill the buffer,
- when the [OnBacktimedFrameCountReached](#) event occurs (1), the buffer is filled out
- then invoke [ResumeRecording](#) to really start the AVI recording.

(1) when this event occurs, back-timed video frames start replacing normal video frames in the captured stream.

quick test code:

```

procedure TForm1.BtnPrepareRecordingClick(Sender: TObject);
begin
    VideoGrabber.OnRecordingReadyToStart := nil;
    VideoGrabber.FrameGrabber := fg_BothStreams;
    VideoGrabber.FrameRate := 10; // 10 fps
    VideoGrabber.RecordingBacktimedFramesCount := 30; // 30 frames = e.g. 3 seconds at 10
    VideoGrabber.RecordingInNativeFormat := False;
    VideoGrabber.HoldRecording := True;
    VideoGrabber.StartRecording;
end;

procedure TForm1.BtnStartRecordingClick(Sender: TObject); // wait for the buffer to fill
begin
    VideoGrabber.ResumeRecording;
end;

procedure TForm1.BtnStopRecordingClick(Sender: TObject);
begin
    VideoGrabber.StopRecording;
end;

```

Created with the Standard Edition of HelpNDoc: [Effortlessly upgrade your WinHelp HLP help files to CHM with HelpNDoc](#)

Playing a video clip during recording

Playing a video clip during recording

[Prev](#)

[Next](#)

How to play a video clip while it is recording?

Description

It is possible to play a video clip in a 2nd instance of TVideoGrabber while it is recording in ASF format a 1st instance as follows:

In the 1st instance of TVideoGrabber:

1.
 - set CompressionMode = cm_NoCompression
 - select the "ASF" recording method (RecordingMethod = rm_ASF),

2. invoke StartRecording.

The video clip starts recording, by using the name specified in the RecordingFileName property (unless you left it blank, in this case a recording file name is generated automatically and returned by the OnRecordingStarted event).

In the 2nd instance of TVideoGrabber:

1. set VideoSource_FileOrUrl = the path / name of the file currently recording the 1st instance
2. set VideoSource = vs_VideoFileOrURL
3. invoke StartPreview

and the video clip currently recording will start playing in the 2nd instance of TVideoGrabber.

This can be tested quickly by running MainDemo.exe as follows:

a)

- run a first instance of MainDemo.exe,
- click on the "Recording" tab,
- select "ASF" as recording method (on the right of the tab),
- click on "start recording"

The name of the file currently recording will appear in the memo on the left, copy it to the clipboard.

b)

- run a 2nd instance of MainDemo.exe,
- in the "video source = file or URL" group, click on the "open" button and paste the name of the file currently recording
- browse the "VIDEO SOURCE" list, and select "video file or URL"
- then click on the "Start" preview button

and the video clip currently recording in the 1st instance of MainDemo.exe will start playing in the 2nd instance.

Remark: the clip played while recording is not seekable, because the total duration will be determined only when the recording completes.

Created with the Standard Edition of HelpNDoc: [Full-featured EPub generator](#)

Dropped frames and audio/video sync problems

Dropped frames and audio/video sync problems

[Prev](#)

[Next](#)

Dropped frames and audio/video sync problems

Description

DROPPED FRAMES

The live video and audio sources stream data at a fixed frame rate, and this means that the processor must have finished processing the previous frame when the next frame arrives, otherwise the frame is dropped and replaced by a blank frame.

The dropped frame count can be retrieved by reading the [DroppedFrames](#) property.

Because reading the dropped frames needs more or less CPU (depending of the video capture device), the number of dropped frames is not updated for every frame, but refreshed periodically according to the [DroppedFramesPollingInterval](#) property (specified in seconds).

E.g:

[DroppedFramesPollingInterval](#) = 20 polls the dropped frames count every 20 seconds.

[DroppedFramesPollingInterval](#) = 0 disables the polling of dropped frames (recommended to save CPU).

Disk access and preallocated files

During recording, one major problem is the block allocation when the operating system extends the recording file. When allocating new blocks, the operating system places significant demands on the CPU/disk resources, and this can cause dropped frames.

This problem occurs mainly when capturing uncompressed video in AVI format. It is less significant when using a video compressor on the fly (or when recording in ASF format), because the file size is 10 times or more smaller.

The best solution to avoid this problem is to record to a SSD (if a lot of disk space is needed for multiple recording, once recorded move the video clip from the SSD to a normal HDD, so the SSD is used only as temporary disk just for the recording).

Another solution is to use an huge preallocated capture file, larger than the amount of data to capture. See the "**preallocated capture file**" section in the [AVI recording](#) chapter.

FAT 32 is not efficient for large capture files, generating large amounts of dropped frames when new disk blocs are allocated.

On Windows 2000 and Windows XP platforms, always choose NTFS partitions on disks with fast access time and cache memory.

Frame grabber

The frame grabber adds an additional step in the stream processing, and therefore requires more CPU. If you don't need to capture frames or perform text or graphics overlays during recording, simply disable it by setting [FrameGrabber](#) = fg_Disabled before invoking [StartRecording](#).

Audio rendering

During recording of video + audio streams, disable if possible [AudioDeviceRendering](#), that renders sound in the loudspeakers. This will let TVideoGrabber capture the video and audio streams without having to split the audio stream. Audio rendering must be disabled before invoking [StartRecording](#).

Video Display

In order to save CPU you can disable the video window during the recording. Simply set VideoRenderer = vr_None before invoking [StartRecording](#).

Video size and frame rate

Of course, larger is the video size and higher is the frame rate, higher is the CPU and disk consumption. Reducing the video size to 320x240 and the frame rate will help to fixing audio sync problems.

Created with the Standard Edition of HelpNDoc: [Elevate Your Help Documentation with a Help Authoring Tool](#)

AES Encryption

Created with the Standard Edition of HelpNDoc: [Transform Your Documentation Process with HelpNDoc's Project Analyzer](#)

AES Encryption Overview

AES Encryption Overview

[Prev](#)

[Next](#)

AES Encryption Overview

Description

It is possible to :

- encrypt a recording in real-time through AES with a 128-bit or 256-bit key to any video container (AVI, MP4, MKV, etc...)
- play and seek the encrypted video with [OpenPlayer\(\)](#). The decryption is performed in real-time.
- batch-encrypt a non-encrypted video clip to produce an encrypted video clip
- batch-decrypt an encrypted video clip to produce an unencrypted video clip

The encryption/decryption require the optionals Datastead RTSP/RTMP/HTTP/ONVIF Source 9.2 or above and/or Datastead Multipurpose Encoder 3.2 or above.

- **the encryption is activated** by passing an encryption key to [SetEncryptionKey](#).

- the decryption is activated by passing the the key used for encryption to [SetDecryptionKey](#).

Both features can be deactivated by passing an **empty key**.

The Datastead RTSP/RTMP/HTTP/ONVIF Source can:

- encrypt in real-time the recording IP cameras or other URL sources in native format (without decompression/recompression) by setting [SetEncryptionKey](#).
- play and seek in real-time any video previously encrypted through the player features. The decryption is enabled with [SetDecryptionKey](#).
- batch encrypt existing video files with [Encrypt_File](#)
- batch decrypt encrypted video files with [Decrypt_File](#)

The Datastead Multipurpose Encoder can:

- encrypt in real-time during recording the audio/video source being recorded, excepted the "recording of IP cameras and other URL sources in native format" case above that is done directly by the Datastead RTSP/RTMP/HTTP/ONVIF Source (described above)

- batch encrypt existing video files with [Encrypt_File](#)
- batch decrypt encrypted video files with [Decrypt_File](#)

Created with the Standard Edition of HelpNDoc: [Make Your PDFs More Secure with Encryption and Password Protection](#)

Key

Key

[Prev](#)

[Next](#)

Encryption/decryption Key

Description

The encryption/decryption key is set by passing a string of 16 characters length (128-bit key) or 32 characters length (256 bits).

For security reasons this key is not stored directly as a string in memory by [SetEncryptionKey](#) or [SetDecryptionKey](#).

Note: **do not loose the key**. Once the file encrypted, without the key it is impossible to recover the file.

Created with the Standard Edition of HelpNDoc: [Experience the Power and Simplicity of HelpNDoc's User Interface](#)

Realtime Encryption or Decryption

Realtime Encryption or Decryption

[Prev](#)

[Next](#)

Realtime Encryption or Decryption

Description

Enabling the realtime encryption

- to enable the **encryption** during recording, invoke [SetEncryptionKey](#) and pass an encryption key of your choice (**that MUST be 16 characters or 32 characters length**), then start the recording as usual
- to enable the **decryption** of an encrypted video during playback, invoke [SetDecryptionKey](#) and pass **the key that has been used for the encryption**, then open the clip for playback as usual

Note: the ASF recording does not yet support encryption for the moment.

See Also

[TEncryptionMethod](#) [Decrypt_File](#) [Encrypt_File](#) [EncryptionMethod](#) [SetDecryptionKey](#) [SetEncryptionKey](#)

Created with the Standard Edition of HelpNDoc: [Easily create Qt Help files](#)

Batch Encryption or decryption

Batch Encryption or decryption

[Prev](#)

[Next](#)

Batch Encryption or decryption

Description

- to batch-encrypt a video file, invoke [SetEncryptionKey](#), then invoke [Encrypt_File](#)
- to batch-decrypt a video file, invoke [SetDecryptionKey](#), then invoke [Decrypt_File](#)

Created with the Standard Edition of HelpNDoc: [Produce Kindle eBooks easily](#)

Disabling the encryption/decryption

Disabling the encryption/decryption

[Prev](#)

[Next](#)

Disabling the encryption/decryption

Description

- after finishing an encryption job (e.g. recording or encrypting an existing video clip), **to disable the encryption invoke [SetEncryptionKey](#)** and pass an empty string.
- after finishing an decryption job (e.g. playback or decryption of an encrypted video clip), **to disable the decryption invoke [SetDecryptionKey](#)** and pass an empty string.

Created with the Standard Edition of HelpNDoc: [Revolutionize Your Documentation Output with a Help Authoring Tool](#)

Frame capture

Created with the Standard Edition of HelpNDoc: [Quickly and Easily Convert Your Word Document to an ePub or Kindle eBook](#)

Frame capture features

Frame capture features

[Prev](#)

[Next](#)

Frame capture overview.

Description

Frame capture

It is possible to capture video frames during preview, recording and play back when the [frame grabber](#) is enabled.

The frame grabber can be inserted on the preview stream, capture stream, or both. See the [FrameGrabber](#) property.

Frames may be captured to:

- memory bitmaps,
- BMP files,
- JPEG files,
- clipboard (CF_BITMAP)

one by one or automatically in burst mode.

I. DIRECT COPY OF THE BITMAP HANDLE (OR DIRECTLY THE BITMAP BITS ARRAY) OF THE CURRENT VIDEO FRAME FROM THE [OnFrameBitmap](#) EVENT

For each video frame travelling through the graph, the [OnFrameBitmap](#) event occurs and allows you to capture directly the bitmap bits.

This is the method that requires the less CPU, however as the [OnFrameBitmap](#) event is called directly from a thread (1), the bitmap handle and the bitmap bits are valid only when the event occurs so you have to copy them to another buffer immediately.

Noet that this method can slow down the graph and/or produce a jerky preview, because the processing time added in the event is included in the frame processing time, so if the time to process 2 frames becomes longer than the frame rate frames will be dropped.

So it recommended rather when the graph is previewing (by invoking [StartPreview](#)), in this case dropped frames are less critical than during a recording (this could produce a jerky recording) or during playback (that may appear jerky).

Sample code can be found in the OnFrameBitmap event of the MainDemo project (seach "FrameBitmap" in the MainForm)

(1) the event can be synchronized with the Windows main thread if the [OnFrameBitmapEventSynchronone](#) property is enabled.

II. DIRECT FRAME CAPTURE OF THE CURRENT (OR PREVIOUS FRAMES)

You can invoke [GetLastFrameAsHBitmap](#) (all versions) or [GetLastFrameAsTBitmap](#) (only with the Delphi/C++Builder versions), to get a copy of the current (or previous) frame with the size of your choice, and with or without overlays.

This way does not slow down the graph and does not generate dropped frames.

A sample code is shown in the MotionDetected event of the MainDemo project, that displays a copy of the last frame detected with motion in the "Overlay" tab.

III. FRAME CAPTURE OF THE CURRENT FRAME (OR NEXT FRAMES AUTOMATICALLY IN BURST MODE)

by using [CaptureFrameTo](#) (or enabling the burst mode) + the [OnFrameCaptureCompleted](#) event

This way does not slow down the graph and does not generate dropped frames.

With or without overlays

If the application performs graphic or text overlays, by default the frames are captured with overlays. To capture the frames without overlays, enable the [FrameCaptureWithoutOverlay](#) property.

ASYNCHRONOUS VS. SYNCHRONOUS FRAME CAPTURE

A) Asynchronous frame capture

to capture frames one by one

Invoke [CaptureFrameTo](#) . The function send a "capture order" and exits immediately. Then the captured frame is returned by the [OnFrameCaptureCompleted](#) event.

to capture frames automatically in burst mode

The burst mode allows to capture frames automatically. It is activated by enabling the [BurstMode](#) property.

Frames can be captured to TBitmap, BMP file or JPEG file according to the [BurstType](#) property.

Frames are captured indefinitely if [BurstCount](#) = 0, otherwise frame capture stops when a [BurstCount](#) number of frames have been captured.

If [BurstInterval](#) > 0, a [BurstInterval](#) number of frames will be skipped between each frame captured.

The burst mode is activated by enabling [BurstMode](#) .

The burst mode is deactivated:

- by disabling [BurstMode](#) ,
- if [BurstCount](#) > 0, when a [BurstCount](#) number of frames have been captured.

When capturing video frames to BMP or JPEG files:

- if no file name is specified when invoking [CaptureFrameTo](#) or [CaptureFrameSyncTo](#), the file name is generated automatically (1).
- in burst mode file names are always generated automatically (1).

(1) according to [StoragePath](#) , [AutoFileName](#) and [AutoFilePrefix](#) .

Frame capture of unmodified frames when performing frame overlay

When drawing over video frames by using the [Frame Overlay](#) feature, by default the frame capture occurs AFTER frames have been overlayed (objects drawn over video frames appear on captured frames).

Enable the [FrameCaptureWithoutOverlay](#) property to capture unmodified frames before they are overlayed.

Example of asynchronous frame capture

The method is the same as described above, excepted that you have to set [UseClock](#) = false before opening the clip, to play it at the maximal speed.

E.g.:

```
VideoGrabber.BurstMode = true
VideoGrabber.BurstCount = 5
VideoGrabber.BurstInterval = 10
VideoGrabber.BurstType = fc_BmpFile
VideoGrabber.PlayerFileName = "... the file name of my video clip..."
VideoGrabber.UseClock = false
VideoGrabber.OpenPlayer
```

Size of the captured frames

Default size

- by default, the size of the captured frames is the current size of the video frames (returned by the [VideoWidth](#) and [VideoHeight](#) properties).

Zoomed size

- it is possible to zoom the captured frames by specifying a [FrameCaptureZoomSize](#) percentage.
The default value is 100 (100%)

Custom size

- it is possible to stretch the captured frames to specified width, height or both by assigning a value to [FrameCaptureWidth](#) and/or [FrameCaptureHeight](#).
The default value for both is -1 (disabled)

NB: if a value $\neq 100$ is assigned to [FrameCaptureZoomSize](#), [FrameCaptureWidth](#) and [FrameCaptureHeight](#) are ignored.

B) Synchronous frame capture through the frame grabber

The function will wait for the frame capture to be completed before returning.

invoke [CaptureFrameSyncTo](#) . The function waits for the frame to be captured. The frame is returned the frame through the [OnFrameCaptureCompleted](#) , and then function returns true upon success, exactly like for the asynchronous [CaptureFrameTo](#) described above.

C) Synchronous frame capture with the frame grabber disabled

It is possible to capture the frame being displayed in the video renderer by invoking [CaptureFrameRenderedTo](#), allowing to disable the frame grabber, to save CPU load and memory.

HOW TO CAPTURE A SUB-RECTANGLE OF THE VIDEO FRAME

To capture a sub-rectangle, invoke [SetFrameCaptureBounds](#) (Left, Top, Right, Bottom) before invoking [CaptureFrameTo](#) or [CaptureFrameSyncTo](#).
(to retrieve the current size of the video frame use [VideoWidth](#) and [VideoHeight](#))

HOW TO STRETCH THE CAPTURED VIDEO FRAME

To resize the video frame as desired, specify a [FrameCaptureWidth](#) and/or [FrameCaptureHeight](#) value

HOW TO ZOOM THE CAPTURED VIDEO FRAME

To resize the captured frame, specify to [FrameCaptureZoomSize](#) a zooming percentage.

TOP-DOWN AND LEFT-RIGHT FRAME CAPTURE

This can be easily achieved by invoking [SetFrameCaptureBounds](#) and specifying the bottom instead of the top, and/or the left instead of the right.
(to retrieve the current size of the video frame use [VideoWidth](#) and [VideoHeight](#))

See Also

[TAutoFileName](#) [TFrameCaptureDest](#) [TOnFrameCaptureCompleted](#) [BurstCount](#) [BurstInterval](#) [BurstMode](#) [BurstType](#) [CaptureFrameSyncTo](#) [CaptureFrameTo](#) [FrameCaptureHeight](#) [FrameCaptureWidth](#) [FrameCaptureWithoutOverlay](#) [FrameCaptureZoomSize](#) [GetFrameInfoString](#) [GetLastFrameAsHBITMAP](#) [GetLastFrameAsTBitmap](#) [GetLastFrameBitmapBits](#) [GetLastFrameBitmapBits2](#) [GetLastFrameWaitTimeoutMs](#) [JPEGPerformance](#) [JPEGProgressiveDisplay](#) [JPEGQuality](#) [Last](#) [BurstFrameCapture](#) [FileName](#) [Last](#) [CaptureFrameTo](#) [FileName](#) [OnDiskFull](#) [OnFrameBitmap](#) [SetFrameCaptureBounds](#) [StoragePath](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Upgrade your help files and your workflow with HelpNDoc's WinHelp HLP to CHM conversion](#)

Graphic and text overlays

Created with the Standard Edition of HelpNDoc: [Easily create CHM Help documents](#)

Setting and retrieving the overlay properties

Setting and retrieving the overlay properties

[Prev](#)

[Next](#)

Graphic and text overlays

Description

TVideoGrabber includes build-in functions to manipulate text and image overlays.

Several text overlays or image overlays can be applied concurrently on the same video stream, **each overlay is identified by its index, in the 0..n-1 range**.

The indexes of the overlays are independent, it is possible to use a text overlay with index 0 and an image overlay with index 0 at the same time.

The overlays can be set in 2 ways:

- either by invoking `SetOverlay_propertyname` (index, value) or `GetOverlay_propertyname` (Index)
- either by setting the overlay index selector first, then setting or reading the property name

1. THE -STRAIGHT- FUNCTION WAY (recommended) e.g.:

```
VideoGrabber.SetTextOverlay_String (0, "first overlay");
VideoGrabber.SetTextOverlay_Left (0, 10);
VideoGrabber.SetTextOverly_Enabled (0, true);
```

```
VideoGrabber.SetTextOverlay_String (1, "2nd overlay");
VideoGrabber.SetTextOverlay_Left (1, 200);
VideoGrabber.SetTextOverly_Enabled (1, true);
```

Text overlays

- the text overlays can be set by invoking `SetTextOverlay...` (text overlay index, value) functions, e.g. [SetTextOverlay_Enabled](#) (0, true) for the 1st text overlay
- the text overlay properties can be read by invoking `GetTextOverlay...` (text overlay index) functions, e.g. [GetTextOverlay_Left](#) (3) for the 4th text overlay

Image overlays

- the image overlay settings can be set by invoking `SetImageOverlay...` (image overlay index, value) function, e.g. [SetImageOverlay_Transparent](#) (2, true) for the 3rd image overlay
- the image overlay settings can be read by invoking `GetImageOverlay...` (image overlay index) functions, e.g. [GetImageOverlay_TopLocation](#) (1) for the 2nd image overlay

2. THE "PROPERTIES" WAY (not recommended when using many overlays) e.g.:

VideoGrabber.TextOverlay_Selector = 0

```
VideoGrabber.TextOverlay_String = "first overlay"
```

```
VideoGrabber.TextOverlay_Left = 10
```

VideoGrabber.TextOverlay_Enabled = true

VideoGrabber.TextOverlay_Selector = 1

VideoGrabber.TextOverlay_String = "2nd overlay"

VideoGrabber.TextOverlay_Left = 200

VideoGrabber.TextOverlay_Enabled = true

Text overlays

The text overlays can be set and retrieved:

- by first setting the [TextOverlay_Selector](#) property
- then by setting or reading the TextOverlay_... properties, e.g. [TextOverlay_Left](#)

Image overlays

The image overlays can be set and retrieved:

- by first setting the [ImageOverlaySelector](#) property
- then by setting or reading the ImageOverlay_... properties, e.g. [ImageOverlay_LeftLocation](#)

Created with the Standard Edition of HelpNDoc: [iPhone web sites made easy](#)

Frame overlay vs window overlay

Frame overlay vs window overlay

[Prev](#)
[Next](#)

Frame overlay vs Window Overlay

Description

FRAME OVERLAY vs WINDOW OVERLAY

It is possible to apply the overlays:

A) over the video frames

(the [frame grabber](#) must be enabled).

- if the video frames are stretched, the overlays are stretched too
- if the video source is being recorded and [RecordingInNativeFormat](#) is disabled, the overlays are applied to the recording.

To apply the overlay over the video frame, set the "target display" property to **-1** (*default value anyway*) by invoking [SetTextOverlay_TargetDisplay](#) (overlay index, **-1**), or [SetImageOverlay_TargetDisplay](#) (overlay index, **-1**)

B 1) over the display video window(s)

- the size of the overlays is fixed and not stretched when the video frame or video window is resized
- if 2 or more video windows are used for the same video stream (enabled by the DualDisplay... properties), it is possible to apply the overlay to only one of the video windows.

To apply the overlay over ALL the video windows, set the "target display" property to **-2** by invoking [SetTextOverlay_TargetDisplay](#) (overlay index, **-2**), or [SetImageOverlay_TargetDisplay](#) (overlay index, **-2**)

To apply the overlay over only 1 of the video windows, set the "target display" property to the index of the video window in the 0..n-1 range, by invoking e.g. [SetTextOverlay_TargetDisplay](#) (overlay index, **0**), or [SetImageOverlay_TargetDisplay](#) (overlay index, **1**)

The video windows indexes are assigned as follows:

- the **default video window** ("Display_..." settings) corresponds to the **index 0**
 - the **2nd video window** ("DualDisplay_..." settings) corresponds to the **index 1**
 - the 3rd video window corresponds to the index 2
- and so on...

Created with the Standard Edition of HelpNDoc: [Ensure High-Quality Documentation with HelpNDoc's Hyperlink and Library Item Reports](#)

Overlays and aspect ratio

Overlays and aspect ratio

[Prev](#)

[Next](#)

About overlays and aspect ratio

Description

About overlays and aspect ratio

Some video sources or video clips output a video size that does not match the display size.

E.g. the size of the video frames is 720x480 and the display aspect ratio is 16:9, so the video frames are stretched to 720x405 in the video window.

By default, the [AdjustOverlayAspectRatio](#) property is enabled and prevents the overlays to be distorted when the video frames are stretched into the video window to be displayed at their correct display aspect ratio.

Disabling this property may save CPU, if no matter the display aspect ratio of the overlays.

Created with the Standard Edition of HelpNDoc: [Generate Kindle eBooks with ease](#)

Image overlays

Image overlays

[Prev](#)

[Next](#)

Image overlays

Description

Image overlays over video frames

Overview

It is possible to overlay one or several images at the same time over the video frames (using more than one image overlay is enabled by the ImageOverlaySelector property described in this chapter).

The origin of the image can be:

- an image file
- a bitmap handle
- a TImage component (not available in OCX versions)
- a TBitmap component (not available in OCX versions)

For image files, the following format are supported:

BMP
GIF
JPEG
EXIF
PNG
TIFF
WMF
EMF
EMF+
EMF+ Dual

Unlike the [logo](#), that is displayed in the video window when the component is **not running**, the image will be overlayed over the video frames when the component is previewing, recording or playing back.

Source of the image overlay

The image can be loaded from various image source formats by invoking:

A) straight method (overlay index passed as parameter):

[SetImageOverlayFromImageFile2](#) (Index, filename): loads the overlay #index with the specified file image

[SetImageOverlayFromHBitmap2](#) (Index, bitmapHandle): loads the overlay #index with the specified bitmap handle

[SetImageOverlayFromTBitmap2](#): loads the overlay #index with the specified TBitmap Object (Delphi and C++Builder)

[SetImageOverlayFromTImage2](#): loads the overlay #index with the specified TImage Object (Delphi and C++Builder)

B) image overlay selector method: (see [Setting and retrieving the overlay properties](#))

- first set [ImageOverlaySelector](#) with the desired index to select the overlay

- then invoke one of the following functions:

[SetImageOverlayFromImageFile](#) (filename): loads the current overlay with the specified file image

[SetImageOverlayFromHBitmap](#) (bitmap handle): loads the current overlay with the specified bitmap handle

[SetImageOverlayFromTBitmap](#) (TBitmap object): loads the current overlay with the specified TBitmap Object (Delphi and C++Builder)

[SetImageOverlayFromTImage](#) (TImage object): loads the current overlay with the specified TImage Object (Delphi and C++Builder)

Activation/deactivation of the image overlay

The image overlay may be enabled or disabled on the fly by enabling or disabling

[SetImageOverlay_Enabled](#) (index, value).

Location of the image overlay

The x,y location of the overlay over the video frames is specified by [SetImageOverlay_LeftLocation](#) and [SetImageOverlay_TopLocation](#).

ImageOverlay_LeftLocation must be in the [0..video width] range.

ImageOverlay_TopLocation must be in the [0..video height] range.

Size of the image overlay

The size to which the image will be stretched can be specified (in pixels) by [SetImageOverlay_Width](#) and [SetImageOverlay_Height](#).

To use the real width of the image overlay set ImageOverlay_Width = -1

To use the real height of the image overlay set ImageOverlay_Height = -1

Stretching the image overlay to the video size

When [SetImageOverlay_StretchToVideoSize](#) is enabled, this property stretches the image overlay to the video size (in this case the image overlay location, width and height settings above are ignored)

Alpha-blending (semi-transparency)

The alphablending is enabled by [SetImageOverlay_AlphaBlend](#).

The value of the alpha blending (in the [0..255] range) must be specified by

[SetImageOverlay_AlphaBlendValue](#).

Color-based transparency

The color-based transparency is enabled by [SetImageOverlay_Transparent](#).

If [SetImageOverlay_UseTransparentColor](#) is disabled, the default color (background color) of the image is used.

If [SetImageOverlay_UseTransparentColor](#) is enabled, the color specified by

[SetImageOverlay_TransparentColorValue](#) will be used.

Specifying all the properties at the time

You can invoke [SetImageOverlayAttributes](#) (instead of setting the image overlay properties one by one) to set all the settings above from a single function call.

Applying several image overlays concurrently

The [ImageOverlaySelector](#) property lets you use up to 1000 image overlays concurrently. This is just a theoretical limit, the corresponding resources will be allocated only when you activate the corresponding selector.

You have just to specify the image overlay number before invoking SetImageOverlay... and ImageOverlayEnabled.

E.g.:

- loading and showing myimage1.gif:

```
ImageOverlaySelector = 0
SetImageOverlayFromAnyFile ("myimage1.gif")
ImageOverlayEnabled = true
```

- loading and showing myimage2.png at the (10, 20) location with a default size:

```
ImageOverlaySelector = 1
SetImageOverlayFromAnyFile ("myimage2.png")
ImageOverlay_LeftLocation = 10
ImageOverlay_TopLocation = 20
ImageOverlay_Width = -1
ImageOverlay_Height = -1
ImageOverlayEnabled = true
```

- hiding myimage1.gif

```
ImageOverlaySelector = 0
ImageOverlayEnabled = false
```

- modifying the location of myimage2.png at (40, 60) and resizing it at 120x90:

```
ImageOverlaySelector = 1
ImageOverlay_LeftLocation = 40
ImageOverlay_TopLocation = 50
ImageOverlay_Width = 120
ImageOverlay_Height = 90
```

- showing myimage1.gif semi-transparent

```
ImageOverlaySelector = 0
ImageOverlayEnabled = true
ImageOverlay_AlphaBlend = true
ImageOverlay_AlphaBlendValue = 140
```

See Also

[GetPixelsDistance](#) [GetRGBPixelAt](#) [ImageOverlay_AlphaBlend](#) [ImageOverlay_AlphaBlendValue](#) [ImageOverlay_Height](#) [ImageOverlay_LeftLocation](#) [ImageOverlay_TopLocation](#) [ImageOverlay_Transparent](#) [ImageOverlay_TransparentColorValue](#) [ImageOverlay_UseTransparentColor](#) [ImageOverlay_VideoAlignment](#) [ImageOverlay_Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#) [SetImageOverlay_AlphaBlend](#) [SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#) [SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#) [SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#) [SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#) [SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#) [SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)

Text overlays

Text overlays

[Prev](#)

[Next](#)

Text overlays

Description

Text over video frames

- set the [SetTextOverlay_String](#) property with the text string to write ([string options](#)),
- set the [SetTextOverlay_Font](#) color and font size,
- set the [SetTextOverlay_Transparent](#) text transparency.
- set the [SetTextOverlay_BkColor](#) background color (useful only when TextOverlay_Transparency is disabled)
- set the [SetTextOverlay_Align](#) text alignment,
- set the [SetTextOverlay_Left](#) , [SetTextOverlay_Right](#) and [SetTextOverlay_Top](#) text location,
- invoke [SetTextOverlay_Enabled](#) to activate the text overlay.

Remark: a shadow can be draw under the text overlay when [SetTextOverlay_Shadow](#) is enabled, according to the [SetTextOverlay_ShadowDirection](#) and [SetTextOverlay_ShadowColor](#) values.

Alpha blending

To enable the alpha blending of the text overlays:

- enable [SetTextOverlay_AlphaBlend](#)
- set [SetTextOverlay_AlphaBlendValue](#) with a transparency value in the 0..255 range

Text scrolling

The text scrolling can be activated by enabling the [SetTextOverlay_Scrolling](#) property.
The scrolling speed is adjusted with the TextOverlay_Scrolling_Speed:

TextOverlay_Scrolling_Speed > 0 -> scrolling from right to left
TextOverlay_Scrolling_Speed < 0 -> scrolling from left to right
TextOverlay_Scrolling_Speed = 0 -> stops the scrolling

When a scrolling is completed (when the last character disappear from the video frame) the [OnTextOverlayScrollingCompleted](#) event occurs.

The TextOverlayIndex parameter of the OnTextOverlayScrollingCompleted event returns the index of the related text overlay.

Custom fonts

You can create a fully customised font by invoking [TextOverlay_CreateCustomFont](#) or [TextOverlay_CreateCustomFont2](#)

E.g.:

```
VideoGrabber.TextOverlay_CreateCustomFont (16, 30, 0, 0, 200, true, false, false, 2, 0, 0, 0, 0, "Wingdings")
```

Using several text overlays, each one having its own set of font / color / background

You can define up to 1000 different groups of text overlays settings, that will be drawn at the same time over video frames, each one having different TextOverlay_... values.

Each group is selected by setting [TextOverlay_Selector](#) to the corresponding value (in the [0..999] range). By default the TextOverlay_... settings concern the 1st group (TextOverlay_Selector = 0).

At startup TVideoGrabber does not allocate the resources for 1000 sets of text overlays. By default only one font and one parameter sets are allocated (corresponding to TextOverlay_Selector = 0).

When using more than one text overlay, the required resource allocation is done when the `TextOverlay_Selector` property receives a greater value than the latest highest value (see *the remark below*).

To use more than one group simply set [TextOverlay_Selector](#) with the index of the group (in the 0..999 range) before reading or setting the corresponding `TextOverlay_...` properties.

Look at the [TextOverlay_Selector](#) property for same code.

Remark:

when using more text overlays, the additional memory and fonts required are allocated when a new `TextOverlay_Selector` value set (and greater than the highest value previously used).

E.g. if the previous value of `TextOverlay_Selector` was 5 and if you set now `TextOverlay_Selector` = 20, this will allocate 15 new fonts and parameter sets, one for each new text overlay (for a total of 20 fonts and parameter sets allocated).

So if you need e.g. only 5 different text overlays in your app (in this case `TextOverlay_Selector` will be used in the [0..4] range), be sure to NEVER set `TextOverlay_Selector` to a value GREATER THAN 4, to prevent allocating more resources than needed (by allocating useless text overlay fonts and parameter sets).

Created with the Standard Edition of HelpNDoc: [Full-featured EBook editor](#)

Graphic overlays

Graphic overlays

[Prev](#)
[Next](#)

[Example](#)

Graphics and text overlays.

Description

Frame overlay

When the [frame grabber](#) is enabled, it is possible to perform graphics or text overlay over uncompressed RGB video frames during preview, recording or playback.

For the graphics and text overlays to **be applied over the video frames recored in the AVI** during recording, the [RecordingInNativeFormat](#) property must be **disabled**.

Getting direct access to the current video frame bitmap

You can get direct access to each video frame by using the [OnFrameBitmap](#) event, that occurs for each video frame. Click [here](#) for more information about this event.

You will find the sample code in the "MainForm" of the MainDemo project included in te package -> search for "FrameBitmap"

From this event you should not perform any actions with the potential to block:

1. do not hold a critical section or wait on another thread,
2. do not call any GDI or USER32.DLL APIs that might cause a window to move,
3. do not invoke component properties that send messages, e.g. do not read the `ItemIndex` property of a `ListBox` component.

If you need to read a such property, set an intermediary variable when the property changes, and then read the intermediary variable from the `OnFrameOverlay...` event.

Restrictions

- a. when the video is recorded in "native" format ([RecordingInNativeFormat](#) = true), drawing over video frames will be **visible** on the preview stream but **not on the AVI stream** (native means "unmodified"),
- b. during preview or recording of video capture boards having **a video port** (e.g. like ATI Radeon), drawing over video frames will **not be visible** on the **preview stream** when the [video port](#) is **enabled**.

Therefore:

1. for the overlays to be visible on the preview window, disable the [video port](#) (but depending of the video capture device the preview could be jerky when previewing large video sizes (or in full screen mode),
2. in order to include the frame overlay in the recorded stream, disable [RecordingInNativeFormat](#) (however this requires higher CPU consumption during recording, because the video stream can have to be uncompressed before applying the overlays).

Drawing bitmaps over video frames

Bitmaps can be drawn "as is" or stretched over video frames by invoking the [DrawBitmapOverFrame](#) function.

This function includes the transparency and alpha blending features.

It MUST be called only from the [OnFrameBitmap](#) event, that occurs for each video frame.

Drawing a shape over the video frames *(not available in the OCX versions)*

To draw a shape over video frames:

- put a TShape component on the form,
- assign the TShape component to the ShapeOverlay property,
- disable the TShape's visible property to avoid the shape being visible at runtime,
- enable [ShapeOverlayEnabled](#) to activate drawing shapes over video frames,
- at runtime, define the TShape's Left, Top, Width and Height properties, that will be used by TVideoGrabber to position and resize the shape over video frames.

Drawing several shapes over video frames *(not available in the OCX versions)*

To draw several shapes over video frames:

- put TShape components on the form (e.g. Shape1 and Shape2),
- call VideoGrabber1.[ShapeOverlayList](#) (Shape1, true) to add the 1st shape to the TVideoGrabber's shape list,
- call VideoGrabber1.[ShapeOverlayList](#) (Shape2, true) to add the 2nd shape to the TVideoGrabber's shape list,
- disable the Shape1 and Shape2 visible properties to avoid the shape being visible at runtime,
- enable [ShapeOverlayEnabled](#) to activate drawing shapes over video frames,
- at runtime, define the Shape1 and Shape2's Left, Top, Width and Height properties, that will be used by TVideoGrabber to position and resize the shapes over video frames.

Note: Shapes are not drawn if their *Enabled* property is disabled.

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#) [SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#) [SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#) [SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#) [SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#) [TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#) [TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#) [TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#) [TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#) [TranslateMouseCoordinates](#)

Chroma key

Chroma key

[Prev](#)

[Next](#)

Chroma Key

Description

Chroma Key

The chroma key feature is activated as follows:

- load a background image with [SetImageOverlayFromImageFile](#) (or another SetImageOverlayFrom... function)
- enable the [ImageOverlayEnabled](#) property
- enable the [ImageOverlay_ChromaKey](#) property
- assign the chroma key RGB color to use to the [ImageOverlay_ChromaKeyRGBColor](#) property
- adjust the chroma key leeway % (optional) with the [ImageOverlay_ChromaKeyLeewayPercent](#) property (the default value is 25%)

E.g.:

```
VideoGrabber.PlayerFileName := 'my video clip.avi';
VideoGrabber.SetImageOverlayFromImageFile ('my background image.jpg');
VideoGrabber.ImageOverlay_ChromaKeyRGBColor := $FF0000; // the blue color
VideoGrabber.ImageOverlay_ChromaKeyLeewayPercent := 20; // 20%
VideoGrabber.ImageOverlay_ChromaKey := true;
VideoGrabber.ImageOverlayEnabled := true;
VideoGrabber.OpenPlayer;
```

To use several image overlay, see "**Applying several image overlays concurrently**" in the "[Graphic overlays](#)" chapter.

See Also

[ImageOverlay_ChromaKey](#) [ImageOverlay_ChromaKeyLeewayPercent](#)
[ImageOverlay_ChromaKeyRGBColor](#)

Created with the Standard Edition of HelpNDoc: [Easily create HTML Help documents](#)

Overlays before or after transforms

Overlays before or after transforms

[Prev](#)

[Next](#)

Overlays before or after transforms

Description

The [OverlayBeforeTransform](#) property allows to specify if the graphic and text overlays must be performed before or after the transforms (like video rotation, cropping, etc...).

Created with the Standard Edition of HelpNDoc: [Converting Word Documents to eBooks: A Step-by-Step Guide with HelpNDoc](#)

To retrieve a pixel value

To retrieve a pixel value

[Prev](#)

[Next](#)

How to retrieve a pixel value

Description

How to retrieve a pixel value

It is possible to retrieve the RGB value of a pixel at the (x, y) location in the video frame by invoking [GetRgBPixelAt](#) from the [OnFrameOverlayUsingDC](#) event.

Note: the video size is returned by the [VideoWidth](#) and [VideoHeight](#) functions.

Created with the Standard Edition of HelpNDoc: [5 Reasons Why a Help Authoring Tool is Better than Microsoft Word for Documentation](#)

How to refresh the overlays while the video clip is paused

How to refresh the overlays while the video clip is paused

[Prev](#)

[Next](#)

How to refresh the overlays while the video clip is paused

Description

When the video clip is paused, the overlays are not refreshed by default, so the OnFrameOverlayUsing... events do not occur.

Invoke [RefreshPlayerOverlays](#) and TVideoGrabber will raise the OnFrameOverlayUsing... events and refresh the video frame with the updated overlays.

Look at the "free hand drawing" sample code in the OnFrameOverlayUsingDC event of the MainDemo project.

Created with the Standard Edition of HelpNDoc: [Experience a User-Friendly Interface with HelpNDoc's Documentation Tool](#)

Mouse events

Mouse events

[Prev](#)

[Next](#)

Mouse events

Description

The following mouse events are available when the mouse is moved over the video window:

OnMouseDown: [TOnVideoMouseUpDown](#)

OnMouseMove: [TOnVideoMouseMove](#)

OnMouseUp: [TOnVideoMouseUpDown](#)

OnMouseWheel: [TOnMouseWheel](#)

(activated only if [MouseWheelEventEnabled](#) is set to true)

If [TranslateMouseCoordinates](#) is enabled (default), the mouse events return the real corresponding x, y positions on the video source, even if the video window is stretched.

Created with the Standard Edition of HelpNDoc: [Free PDF documentation generator](#)

Player

Created with the Standard Edition of HelpNDoc: [Don't be left in the past: convert your WinHelp HLP help files to CHM with HelpNDoc](#)

Player features

Player features

[Prev](#)
[Next](#)

About the player.

Description

Player features

TVideoGrabber includes an advanced player that allows to play video clips, [capture video frames](#) and perform [frame overlay](#) when the [frame grabber](#) is enabled. It is possible to play multiple video clips seamlessly with the [Seamless playback](#) feature.

Opening / pausing / running / closing the clip

- set the [PlayerFileName](#) property with the full qualified path or the URL of the video clip,
- open the clip with [OpenPlayer](#)

When the clip is opened the [OnPlayerOpened](#) event occurs. From this event you can read the [PlayerDuration](#) and [PlayerFrameCount](#) properties, e.g. to setup a trackbar (see below the trackbar chapter)

To open a clip in a paused state:

VideoGrabber.[PlayerFileName](#) = "myclip.avi"

VideoGrabber.[AutoStartPlayer](#) = **false**

VideoGrabber.[OpenPlayer\(\)](#)

To open and play a clip immediately:

VideoGrabber.[PlayerFileName](#) = "myclip.avi"

VideoGrabber.[AutoStartPlayer](#) = **true**

VideoGrabber.[OpenPlayer\(\)](#)

To pause a clip during playback:

VideoGrabber.[PausePlayer\(\)](#)

To resume the playback when the clip is paused:

VideoGrabber.[RunPlayer\(\)](#)

To close the clip:

VideoGrabber.[ClosePlayer\(\)](#)

Player seeking

Note:

- if a video clip is seeking slow, invoke [MultipurposeEncoder_ReindexClip](#) to reindex it (*requires the optional Datastead Multipurpose Encoder to be installed*)
- to determine if a MP4 video clip needs a reindexing, invoke [MP4NeedsReindexing](#)

After opening the video clip, you can set a new position by assigning:

- either the [PlayerFramePosition](#) property, in the 1..[PlayerFrameCount](#) range
- either the [PlayerTimePosition](#) property, in the 0..[PlayerDuration](#) range (expressed in 100ns units, e.g. 2 seconds = 20000000)

To set the current position at the frame #500:

VideoGrabber.[PlayerFramePosition](#) = 500

To specify a new position located at "15 seconds":

VideoGrabber.[PlayerTimePosition](#) = 150000000

Frame counting and frame seeking

By default the frame counting / frame seeking starts from 1 (the 1st frame is the frame #1).

Alternatively it is possible to let TVideoGrabber start from the frame #0 by enabling the [FrameNumberStartsFromZero](#) property.

Frame stepping

When the clip is paused, you can invoke [PlayerFrameStep](#) (1) to step to the next frame.

It is possible to specify a value > 1 to skip n-1 frames, e.g. [PlayerFrameStep](#) (10) will skip the next 9 frames and display the 10th frame.

*Note that with MPEG video clips [PlayerFramePosition](#) and [PlayerTimePosition](#) can reach only key frames, whereas [PlayerFrameStep](#) will step through all the frames (key frames **and** delta frames).*

Seeking

If the clip is seekable it can be played backwards using [RunPlayerBackwards](#) , however the play back is not as smooth as normal play back.

Note that you can get a more effective reverse playback by opening the clip through AVISynth, as explained in the [Advanced playback through AVISynth](#) chapter.

To scan the clip quickly:

- forwards use [FastForwardPlayer](#) ,
- backwards use [RewindPlayer](#) ,

The scan speed can be modified using [PlayerFastSeekSpeedRatio](#) .

Keeping the last video frame visible while opening the next video clip

By default after invoking [ClosePlayer](#), the video window is closed until the next clip is opened.

Enable the [VideoVisibleWhenStopped](#) property to keep the video window active after the previous video ends.

Then the last video frame will remain displayed until [OpenPlayer](#) is invoked and the new video clip starts playing.

Playback boundaries

Instead of invoking [OpenPlayer](#):

- invoke [OpenPlayerAtFramePositions](#) to open a clip at the specified start frame
- invoke [OpenPlayerAtTimePositions](#) to open a clip at the specified stop frame

If the KeepBound parameter is true, the clip is played within the start and stop boundaries specified.

When the clip is already opened, invoking [OpenPlayerAtFramePositions](#) or [OpenPlayerAtTimePositions](#) modifies the playback boundaries without reopening the clip.

Playback progress

During playback, information about each video frame is reported by the [OnFrameProgress](#) event.

End of stream

When the end of stream is reached the [OnPlayerEndOfStream](#) event occurs.

From this event you can invoke any task, e.g. you can invoke [RunPlayer](#) to restart the playback from the beginning.

Paused player and CPU consumption

When a video clip is paused and:

- the [PlayerRefreshPausedDisplay](#) property is enabled,
- the [FrameGrabber](#) property is enabled,

the display is refreshed periodically at a frame rate specified by [PlayerRefreshPausedDisplayFrameRate](#), in order to get the text and graphics overlays refreshed.

However this feature requires CPU. To reduce the CPU consumption when a video clip is paused, you can:

- specify a lower [PlayerRefreshPausedDisplayFrameRate](#) value,
- disable the [PlayerRefreshPausedDisplay](#) property.

Playback speed

A clip is played at its normal speed when [PlayerSpeedRatio](#) = 1.

It is possible to play video clips at different speeds, however the playback speed range is larger if the audio rendering is disabled. To play a clip at a different speed:

- with audio rendering, enable [PlayerAudioRendering](#) and modify [PlayerSpeedRatio](#) in the 0.5 to 2.0 range,
- without audio rendering, disabled [PlayerAudioRendering](#) and modify [PlayerSpeedRatio](#) in a larger range.

Playing clips at max speed without clock

It is possible to play the clips without clock, this guarantee e.g. that all the frames will be captured in burst mode (when [BurstMode](#) = true and [BurstCount](#) = 0).

To enable this feature, disable [UseClock](#) before invoking [OpenPlayer](#).

Player trackbar

TVideoGrabber handles automatically a player trackbar (Delphi and C++Builder versions ONLY).

Simply put a trackbar on the form, and then associate it by assigning it from the Object Inspector to the [PlayerTrackBar](#) property.

See the [PlayerTrackBar](#) and the [PlayerTrackBarScale](#) properties for more information.

Frame capture during playback

Video frames can be captured during playback. See the [Frame capture](#) chapter.

Audio playback

For video clips that has an audio stream (when [IsPlayerAudioStreamAvailable](#) returns ts_True), you can disable the rendering of the audio stream by setting [PlayerAudioRendering](#) to false. This can help to play video clips slower than 0.5 or faster than 2.

Selecting the audio sound card

You can specify the soundcard used for the playback rendering by assigning the [AudioRenderer](#) index with the index of the sound card in the [AudioRenderers](#) list.

e.g. if the index in the list is 2:

```
VideoGrabber.AudioRenderer = 2
```

OR

e.g. if the name in the list is "Speakers (Blackmagic Audio):"

```
VideoGrabber.AudioRenderer = VideoGrabber.AudioRendererIndex ("Speakers (Blackmagic Audio)")
```

Selecting only the right or left audio channel

It is possible to convert the right channel or the left channel as "mono" and to mute either the left, right or both channels, or to convert a mono channel into stereo with [AudioChannelRenderMode](#).

Selecting one audio stream when the clip has multiple audio streams

When the clip has multiple audio streams it is possible to select only one audio stream with [AudioStreamNumber](#)

Codec used during playback

It is possible to force the use of a given codec by assigning a codec name or a codec GUID to [PlayerForcedCodec](#) .

The video codec and audio codec using during playback are reported by the [PlayerVideoCodec](#) and [PlayerAudioCodec](#) properties.

Specifying the buffering time when playing a streaming URL

The buffering time can be adjusted by invoking [ShowDialog](#) (dlg_NetShowConfig)

Authentication

If an authentication is required to connect to a streaming url, there are 2 ways to set an username and password required:

- either by invoking [SetAuthentication](#) (at_StreamingUrl, "...username...", "...password...") before invoking OpenPlayer,
- either through the [OnAuthenticationNeeded](#) event that will occur when connecting if [SetAuthentication](#) has not been invoked yet.

See Also

[Opening a clip or an IP URL from a background thread without blocking the main thread](#)
[TOnPlayerStateChanged](#) [TOnThreadSync](#) [TPlayerState](#) [TThreadSyncPoint](#) [TOnPlayerBufferingData](#)
[AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#)
[AVIInfo2](#) [ClosePlayer](#) [EnableThreadMode](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#)
[IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#)
[OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OnThreadSync](#)
[OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#)
[PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#)
[PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#)
[PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#)
[PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchronise](#) [PlayerVideoCodec](#)
[RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#)
[SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#)
[VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Add an Extra Layer of Security to Your PDFs with Encryption](#)

Seamless playback

Seamless playback

[Prev](#)
[Next](#)

Seamless playback

Description

The seamless playback feature allows to play a list of several clips **having the same format** as a single clip:

```
VideoGrabber.Playlist (pl_Clear, "");
VideoGrabber.Playlist (pl_Add, 'c:/folder/videoclip1.mp4');
VideoGrabber.Playlist (pl_Add, 'c:/folder/videoclip2.mp4');
VideoGrabber.Playlist (pl_Add, 'c:/folder/videoclip3.mp4');
VideoGrabber.PlayerFileName := 'PLAYLIST';
VideoGrabber.OpenPlayer;
```

More information can be found in the [PlayList](#) chapter.

Created with the Standard Edition of HelpNDoc: [Powerful and User-Friendly Help Authoring Tool for Markdown Documents](#)

Current player state

Current player state

[Prev](#)
[Next](#)

Current player state

Description

Obtaining the current player state

At any time it is possible to know the current player state by reading the [PlayerState](#) property.

E.g., to know if the player is currently in normal playing state:

```
if PlayerState = ps_Playing then...
```

E.g. to know if the player is opened and playing normally, backwards or faster:

```
if PlayerState >= ps_Playing then...
```

E.g. to know if the player is paused:

```
if PlayerState = ps_Paused then...
```

Player state notification

Each time the player state changes, the [OnPlayerStateChanged](#) event and returns the old and the new state (of [TPlayerState](#) type).

Created with the Standard Edition of HelpNDoc: [Create iPhone web-based documentation](#)

Trackbar

Trackbar

[Prev](#)
[Next](#)

Implementation of a player trackbar

Description

Moving the position clip immediately while the trackbar position is moved

When [PlayerTrackBarSynchron](#) is enabled, moving the trackbar with the mouse **updates continuously** the position in the video clip, until the mouse button is released.

When [PlayerTrackBarSynchron](#) is disabled (**default**), the position is **updated only** when the mouse button is released.

This property requires the **full trackbar implementation** below (unless you are using the PlayerTrackbar property under Delphi or C++Builder, see remark below)

TRACKBAR IMPLEMENTATION

Remark: if you are using Delphi or C++Builder, you can put a TTrackbar component on your form, assign it to the [PlayerTrackbar](#) property, it will be handled automatically so you can just ignore the implementations below.

Minimal trackbar implementation

The minimal trackbar implementation consists to:

- place a trackbar on your form
- create a TVideoGrabber's [OnPlayerOpened](#) event, and set your trackbar max position by reading the [PlayerFrameCount](#) property from this event
- create a TVideoGrabber's [OnPlayerUpdateTrackbarPosition](#) event, and update your trackbar position with the FrameNumber parameter returned by this event
- create a trackbar's OnChange or OnValueChanged event, and update the [PlayerFramePosition](#) from this event with the current position of your trackbar

The problem with this minimal implementation is that you may notice a jerky trackbar behavior if you move the trackbar thumb while the clip is playing, because the trackbar position is upgraded periodically in the background by the [OnPlayerUpdateTrackbarPosition](#) event.

FULL TRACKBAR IMPLEMENTATION IN VISUAL STUDIO

1. put a trackbar component on your form

2. create a TVideoGrabber's [OnPlayerOpened](#) event that sets the min and max positions of the trackbar, e.g.:

```
private void axVideoGrabberNET1_OnPlayerOpened(object sender, System.EventArgs e)

tbrPlayer.Minimum = 0;
tbrPlayer.TickFrequency = 1;
tbrPlayer.Maximum = (int) axVideoGrabberNET1.PlayerFrameCount;
```

3. create a TVideoGrabber's [OnPlayerUpdateTrackbarPosition](#) event that updates your trackbar's position, e.g.:

```
private void axVideoGrabberNET1_OnPlayerUpdateTrackbarPosition(object sender,
Axvidgrab_NET.IVideoGrabberNETEvents_OnPlayerUpdateTrackbarPositionEvent e)

tbrPlayer.Value = (int) e.frameNumber;
```

4. create a trackbar's [OnChange](#) or [OnValueChanged](#) event that updates the TVideoGrabber's [PlayerFramePosition](#), e.g:

```
private void tbrPlayer_ValueChanged(object sender, System.EventArgs e)

axVideoGrabberNET1.PlayerFramePosition = tbrPlayer.Value;
```

5. create a trackbar's [OnMouseDown](#), [OnMouseUp](#), [OnKeyDown](#) and [OnKeyUp](#) event that invoke [NotifyPlayerTrackbarAction](#) with the corresponding TTrackbarAction parameter, e.g.:

```
private void tbrPlayer_MouseDown(object sender, System.Windows.Forms.MouseEventArgs e)

axVideoGrabberNET1.NotifyPlayerTrackbarAction (vidgrab_NET.TxTrackbarAction.tba_MouseDown);

private void tbrPlayer_MouseUp(object sender, System.Windows.Forms.MouseEventArgs e)

axVideoGrabberNET1.NotifyPlayerTrackbarAction (vidgrab_NET.TxTrackbarAction.tba_MouseUp);

private void tbrPlayer_KeyDown(object sender, System.Windows.Forms.KeyEventArgs e)

axVideoGrabberNET1.NotifyPlayerTrackbarAction (vidgrab_NET.TxTrackbarAction.tba_KeyDown);

private void tbrPlayer_KeyUp(object sender, System.Windows.Forms.KeyEventArgs e)

axVideoGrabberNET1.NotifyPlayerTrackbarAction (vidgrab_NET.TxTrackbarAction.tba_KeyUp);
```

Then run your app, the trackbar should work properly even when moving the thumb while the clip is playing.

Created with the Standard Edition of HelpNDoc: [Don't be left in the past: convert your WinHelp HLP help files to CHM with HelpNDoc](#)

Playing static images and animated GIFS

Playing static images and animated GIFS

[Prev](#)

[Next](#)

Playing static images

Description

Static images

Static images, like BMP and JPG files can be opened in the player.

They are opened in a paused state, so if [PlayerRefreshPausedDisplay](#) and [FrameGrabber](#) are enabled, it is possible to perform graphic processing and then recapture the frame.

The transform processings can also be applied (like [rotation](#), [brightness](#), [cropping](#), etc...)

Animated GIFs

Animated GIFs can be opened exactly as video clips. The transform processings and graphic overlays can be applied (like [rotation](#), [brightness](#), [cropping](#), etc...).

Created with the Standard Edition of HelpNDoc: [Produce online help for Qt applications](#)

Playlist

Playlist

[Prev](#)

[Next](#)

Using the playlist

Description

Overview

The playlist features lets you specify a list of video clips that have to be played automatically one after the other.

There are 2 ways to use the playlist:

statically: you specify a static set of clips, then play this set of clip like it was a single clip (the trackbar moves across the whole set of clips)

dynamically: you can dynamically add/remove clips while the playlist is playing by invoking `PlayList(pl_Add...)`. In this case the trackbar is active for the current clip being played.

- all the playlist actions are performed by using the [Playlist](#) function, detailed below.
- the playlist content can be retrieved by the [GetPlaylist](#) function, that returns a list of strings separated by CR/LF
- the current index in the playlist is returned by the [PlaylistIndex](#) property (if opened dynamically)
- when the end of the playlist is reached, the [OnPlayerEndOfPlaylist](#) event occurs.
- it is possible to know if the playlist is currently playing by testing [IsPlaylistActive](#).

The Playlist function

The playlist function accepts 2 parameters:

- a first parameter that specify the action
- a 2nd parameter that specify the video clip (used only for `pl_Add` and `pl_Remove`).

The first parameter can take one of the following values:

pl_Add: adds the video clip (specified as 2nd parameter) to the playlist

pl_Remove: removes the video clip (specified as 2nd parameter) from the playlist

pl_Clear: clears the playlist

pl_Loop: enables the loop mode (the playlist restart from the beginning when the end is reached)
pl_NoLoop: disables the loop mode
pl_Play: starts playing the playlist
pl_Stop: stops playing the playlist after the end of the current clip
pl_Next: starts playing the next clip (immediately)
pl_Previous: starts playing the previous clip (immediately)
pl_SortAlpha: sorts the playlist in the alphabetical order
pl_SortRevAlpha: sorts the playlist in the reverse alphabetical order
pl_Random: enable the "random" mode: the clips will be played in a random order.
pl_Sequential: disables the "random" mode: the clips will be played sequentially.
pl_SpecifyPositions: let specify a start position and/or a stop position

Remark: while the static playlist is playing, you can't control it with the playlist settings anymore, it behaves like a single video clip.

Building the playlist

Simply invoke Playlist (pl_Add, "... path of the video clip...") as many times as required. Pass an empty string for the "VideoClip" parameter when it is not required (it is required only for pl_Add and pl_Remove).
 E.g.:

```
videograbber.Playlist (pl_Clear, '')
videograbber.Playlist (pl_Add, 'vg000004.avi')
videograbber.Playlist (pl_Add, 'vg000002.avi')
videograbber.Playlist (pl_Add, 'StaticImage1.BMP')
videograbber.Playlist (pl_Add, 'vg000003.avi')
videograbber.Playlist (pl_Add, 'vg000005.avi')
videograbber.Playlist (pl_Add, 'vg000001.avi')
videograbber.Playlist (pl_Add, 'StaticImage2.JPG')
```

Starting the dynamic playlist

Be sure to set all the required options (like pl_Loop, pl_SortAlpha or pl_Random), then invoke pl_Play. E.g.:

```
videograbber.Playlist (pl_Clear, '')
videograbber.Playlist (pl_Add, 'vg000004.avi')
videograbber.Playlist (pl_Add, 'vg000002.avi')
videograbber.PlayerDuration = 5000000 // specifies 5 seconds of display time for Stat
videograbber.Playlist (pl_Add, 'StaticImage1.BMP');
videograbber.Playlist (pl_Add, 'vg000003.avi')
videograbber.Playlist (pl_Add, 'vg000005.avi')
videograbber.Playlist (pl_Add, 'vg000001.avi')
videograbber.PlayerDuration = 8000000 // specifies 8 seconds of display time for Stat
videograbber.Playlist (pl_Add, 'StaticImage2.JPG')
videograbber.PlaylistIndex := 0
videograbber.Playlist (pl_SortAlpha, '')
videograbber.Playlist (pl_Loop, '')
videograbber.Playlist (pl_Random, '')
videograbber.Playlist (pl_Play, '');
```

Starting the static playlist

```
videograbber.Playlist (pl_Clear, '');
videograbber.Playlist (pl_Add, 'vg000004.avi');
videograbber.Playlist (pl_Add, 'vg000002.avi');
videograbber.Playlist (pl_Add, 'vg000003.avi');
videograbber.Playlist (pl_Add, 'vg000005.avi');
videograbber.Playlist (pl_Add, 'vg000001.avi');
videograbber.PlayerFileName = 'PLAYLIST'
videograbber.OpenPlayer()
```


Specifying a custom video size and/or frame rate to the static playlist

By default the static playlist plays all the clips at the same video size (640x480) and at the same frame rate (29.97 fps)

- to customize the frame rate, set a [FrameRate](#) value > 0
- to customize the video size, invoke [UseNearestVideoSize](#) (width, height, false) before invoking `OpenPlayer`

E.g.:

```
...
VideoGrabber.PlayList (pl_Add, "vg0000006.avi")
VideoGrabber.PlayerFileName = 'PLAYLIST'
VideoGrabber.FrameRate = 25.0
VideoGrabber.UseNearestVideoSize (720, 480, false)
VideoGrabber.OpenPlayerPlayer()
```

You can later reset the video size by invoking:

```
UseNearestVideoSize (0, 0, false)
and the frame rate by setting:
FrameRate = 0
```

Pausing the playlist

Simply invoke [PausePlayer](#).

Stopping the playlist

- to stop the playlist at the end of the current clip, invoke `Playlist (pl_Stop)`
- to close the playlist immediately, invoke [ClosePlayer](#)

specifying start/stop positions for each clip

Invoke `VideoGrabber.PlayList (pl_SpecifyPositions,...)` and immediately after [OpenPlayerAtTimePositions](#) or [OpenPlayerAtFramePositions](#) (this will not open the clips but just register the positions)

```
VideoGrabber.PlayList (pl_Add, 'myvideoclip1.wmv');
VideoGrabber.PlayList (pl_SpecifyPositions, 'myvideoclip1.wmv');
VideoGrabber.OpenPlayerAtTimePositions (50000000, 120000000, true, true);

VideoGrabber.PlayList (pl_Add, 'myvideoclip2.wmv');
VideoGrabber.PlayList (pl_SpecifyPositions, 'myvideoclip2.wmv');
VideoGrabber.OpenPlayerTimePositions (30000000, 130000000, true, true);

VideoGrabber.PlayerFileName := 'PLAYLIST';
VideoGrabber.Openplayer();
```

In this example the first clip will play 7 seconds (from 7 seconds to 12 seconds) and then the second will play 10 seconds (from 3 seconds to 13 seconds) to let the clip play until the end specify -1 as stop position)

Using static images or animated GIFs in the playlist

Static images (like BMP, JPG files) can be opened alternatively with video clips in the dynamic playlists. To specify the display time for static images, assign the [PlayerDuration](#) with the desired value **BEFORE** the `PlayList (pl_Add...)` statement (the display time is expressed as 100ns units, e.g. 7 seconds = 70000000).

E.g.:


```

VideoGrabber.PlayerDuration = 30000000; // will set 3 seconds for myvideoclip1
VideoGrabber.PlayList (pl_Add, 'myvideoclip1.wmv');
VideoGrabber.PlayerDuration = 40000000; // will set 4 seconds for myvideoclip2
VideoGrabber.PlayList (pl_Add, 'myvideoclip2.wmv');
...
VideoGrabber.PlayList (pl_Play, "");

```

See Also

[TPlaylist](#) [Video formats](#) [GetPlaylist](#) [IsPlaylistActive](#) [OnPlayerEndOfPlaylist](#) [Playlist](#) [PlaylistIndex](#)

Created with the Standard Edition of HelpNDoc: [Converting Word Documents to eBooks: A Step-by-Step Guide with HelpNDoc](#)

Playing a clip from a TStream

Playing a clip from a TStream

[Prev](#)
[Next](#)

Playing a clip from a TStream.

Description

It is possible to play in the media player a video clip stored in a TMemoryStream, TFileStream or TStream descendent.

Simply assign your TStream descendent to the [SourceStream](#) property. When this property is assigned **before** invoking [OpenPlayer](#), the stream data is used as source by the player.

To go back to the "normal player behavior", set a **nil** (or NULL in C++) value to [SourceStream](#) **after** invoking [ClosePlayer](#), to let TVideoGrabber use for the next [OpenPlayer](#) the clip specified in the [PlayerFileName](#) property.

Created with the Standard Edition of HelpNDoc: [Effortlessly Support Your Windows Applications with HelpNDoc's CHM Generation](#)

StartPreview or StartRecording from video clips or URLs

StartPreview or StartRecording from video clips or URLs

[Prev](#)
[Next](#)

Previewing or recording video clips

Description

Previewing or recording video clips

It is possible to preview or record a video clip (or a streaming URL) by invoking StartPreview or StartRecording as follows:

- set VideoSource = vs_VideoFileOrURL
- set VideoSource_FileOrUrl = "http://.....YourStreamingUrl..." or ("mms://...yourStreamingUrl....")
- (optionally) invoke SetAuthentication (ap_StreamingUrl, "...username...", "...password...")
- then invoke StartPreview or StartRecording

Authentication

If an authentication is required to connecto a streaming url, there are 2 ways to set an username and password required:

- either by invoking [SetAuthentication](#) (at_StreamingUrl, "...username...", "...password...") before invoking OpenPlayer,
- either through the [OnAuthenticationNeeded](#) event that will occur when connecting if [SetAuthentication](#) has not been invoked yet.

Created with the Standard Edition of HelpNDoc: [Full-featured EPub generator](#)

AVI / ASF information and header attributes

AVI / ASF information and header attributes

[Prev](#)

[Next](#)

AVI / ASF information and header attributes of a video clip

Description

AVI information of a video clip

The [AVIInfo](#) function retrieves the following information:

- duration,
- frame count,
- video width
- video height,
- video frame rate in frames per second,
- audio bit rate in Hertz,
- name of the video codec,
- name of the audio codec.

This function requires the parameters to be passed by reference.

For languages that pass parameters only by value (e.g. JavaScript), use [AVIInfo2](#) instead.

AVI / ASF header attributes of a video clip

Invoke [AVIHeaderInfo](#) to retrieve the attributes of an existing video clip, if any.

Created with the Standard Edition of HelpNDoc: [Effortlessly create a professional-quality documentation website with HelpNDoc](#)

Playing pictures

Playing pictures

[Prev](#)

[Next](#)

Playing pictures

Description

The player can open a single picture, e.g.:

VideoGrabber.[PlayerFileName](#) = "mypicture.jpg"

VideoGrabber.[OpenPlayer](#)()

However you may need a picture to be displayed at a given frame rate, e.g. to perform graphics or text overlays that may change while the picture is displayed.

In this case proceed rather this way:

- set [VideoSource](#) = vs_JPEGsOrBitmaps,
- create a [OnVideoFromBitmapsNextFrameNeeded](#) event and pass the file name of your .JPG to the BMPorJPEGFile parameter of this event,
- set [FrameRate](#) with the frame rate needed
- then invoke [StartPreview](#)

Synchronization of several player components

Synchronization of several player components

[Prev](#)[Next](#)

Synchronization of several player components

Description

Synchronization of several player components

It is possible to synchronize several player components.

One component is declared as the master component, the other components are the slave components.

Example:

```
VideoGrabber2.Synchronized = true  
VideoGrabber2.SynchronizationRole = sr_Slave
```

```
VideoGrabber3.Synchronized = true  
VideoGrabber3.SynchronizationRole = sr_Slave
```

```
VideoGrabber1.Synchronized = true  
VideoGrabber1.SynchronizationRole = sr_Master  
VideoGrabber1.PlayerFileName = "myclip.avi"  
VideoGrabber1.AutoStartPlayer = false  
VideoGrabber1.OpenPlayer()  
VideoGrabber1.StartSynchronized\(\)
```

This will open the clip in the 3 components at the same time. Starting from now all the player actions below performed on VideoGrabber1 are applied at the same time on VideoGrabber2 and VideoGrabber3:
by invoking the following functions:

[OpenPlayer](#)
[ClosePlayer](#)
[RunPlayerBackwards](#)
[PausePlayer](#)
[StopPlayer](#)
[RunPlayer](#)
[RewindPlayer](#)
[FastForwardPlayer](#)

by assigning the following properties:

[PlayerTimePosition](#)
[PlayerFramePosition](#)

Shifting the synchronization position of one slave player

You can set a delta position on a slave player (after the clip is opened on all the components) as follows:

- disable temporarily the [Synchronized](#) property on this slave
- set a new position on the slave (with [PlayerFramePosition](#) or [PlayerTimePosition](#))
- enable the [Synchronized](#) property on this slave

Now all position changes on the master component will be performed on the slave by applying the delta position set while the Synchronized property was disabled.

Look at the [SynchronizedPlayers](#) demo project included in the package for sample code.

Video 360°

Created with the Standard Edition of HelpNDoc: [Make Documentation a Breeze with a Help Authoring Tool](#)

Decoding of 360° videos

Decoding of 360° videos

[Prev](#)
[Next](#)

360 Videos decoding

Description

The decoding of 360 videos requires the bundle of the TVideoGrabber SDK + Datastead RTSP/RTMP/HTTP/ONVIF Source Filter (v7.8.2.0 or higher).

To enable the decoding of 360° videos, enable the [v360_Enabled](#) property.

When [v360_MouseAction](#) is activated, the mouse motion controls the point of view, and the mouse wheel controls the angles (zoom), either when the mouse button is down, either when the mouse button is released, depending on the [Tv360_MouseAction](#) value.

When moving the mouse, the "quantity of displacement" is controlled by [v360_MouseActionPercent](#) (10% by default).

The mouse wheel controls the zoom-in or zoom-out.

To return to the default angles, invoke [v360_ResetAnglesToDefault](#)

The point of view can be controled by [v360_SetYawPitchRoll](#) or [v360_AddYawPitchRoll](#).

The current point of view is returned by [v360_GetYawPitchRoll](#)

The zoom can be controled directly by invoking [v360_SetAngle](#) (v360_fov_Horizontal, Value) or [v360_SetAngle](#) (v360_fov_Vertical, Value).

The master angle is the angle from which is calculated the opposite angle to maintain the aspect ratio, by default it is v360_fov_Horizontal but can be changed to v360_fov_Vertical by modifying the [v360_MasterAngle](#) property.

The aspect ratio is controled automatically when [v360_AspectRatio](#) = -1 (default).

This can be deactivated by setting a value > 0.0 (e.g. set 1.777778 to force a 16:9 aspect ratio, or 2.0 to set a 2:1 aspect ratio, etc...).

Supported stereo formats: ([v360_SetStereoFormat](#))

sf_2DMono,
sf_SideBySide,
sf_TopBottom

Supported projections: ([v360_SetProjection](#))

ipp_Equirectangular,
ipp_Cubemap_3_2,
ipp_Cubemap_6_1,
ipp_Equiangular,
ipp_Flat,
ipp_Dual_fisheye,
ipp_Barrel,
ipp_Cubemap_1_6,
ipp_Stereographic,
ipp_Mercator,
ipp_Ball,
ipp_Hammer,
ipp_Sinusoidal,
ipp_Fisheye,
ipp_Pannini,
ipp_Cylindrical,
ipp_Perspective,
ipp_Tetrahedron,
ipp_Barrel_split,
ipp_Tspyramid,
ipp_Hequirectangular,
ipp_Equisolid,
ipp_Orthographic,

ipp_Octahedron

Supported interpolations: ([v360_SetInterpolation](#))

ipl_Bilinear,

ipl_Nearest,

ipl_Lagrange9,

ipl_Bicubic,

ipl_Lanczos,

ipl_Spline16,

ipl_Gaussian,

ipl_Mitchell

Created with the Standard Edition of HelpNDoc: [Create cross-platform Qt Help files](#)

PIP

Created with the Standard Edition of HelpNDoc: [Upgrade your help files and your workflow with HelpNDoc's WinHelp HLP to CHM conversion](#)

PIP (Picture In Picture)

PIP (Picture In Picture)

[Prev](#)

[Next](#)

PIP (Picture In Picture)

Description

PIP (Picture In Picture) between several TVideoGrabber components

During preview, recording or playback, a TVideoGrabber component can receive and display the video from others TVideoGrabber components, and display their videos as rectangles in "PIP" mode (Picture in Picture)

This is set up by invoking [Mixer_SetupPIPFromSource](#) and passing as parameter the TVideoGrabber component used as source, the source rectangle that should be captured, and the destination rectangle where the PIP should be displayed.

The first boolean parameter specifies if the PIP must be activated or deactivated

The last boolean parameter specifies if the PIP must be located at the top (useful only if several PIPs are overlapped)

It is possible to make several PIPs on the same video frame.

E.g. let's suppose the whole VideoGrabber2 video frame should be displayed as a 80x60 rectangle at the x=10, y=10 location in VideoGrabber1, invoke:

```
VideoGrabber1.Mixer_SetupPIPFromSource (VideoGrabber2.UniqueID, 0, 0, 0, 0, true, 10, 10,
```

E.g. let's suppose a 160x120 rectangle of VideoGrabber2 located at x=40 y=60, should be displayed as a 80x60 rectangle at the 10, 10 location in VideoGrabber1, invoke:

```
VideoGrabber1.Mixer_SetupPIPFromSource (VideoGrabber2.UniqueID, 40, 60, 160, 120, true, 1
```

You can re-invoke this function on the fly several times, e.g. to activate/deactivate the PIP or to modify its location or size.

Created with the Standard Edition of HelpNDoc: [Modernize your help files with HelpNDoc's WinHelp HLP to CHM conversion tool](#)

Streaming

Created with the Standard Edition of HelpNDoc: [Full-featured EBook editor](#)

Streaming through the Datastead Encoder

Streaming through the Datastead Encoder

[Prev](#)

[Next](#)

Streaming through the Datastead Encoder

Declaration

procedure TfrmMainForm.btnMultipurposeUDPStreamingClick(Sender: TObject);

begin

if FMultipurposeUDPStreamingID=-1 **then begin**

FMultipurposeUDPStreamingID := VideoGrabber.Encoders_CreateInstanceForStreaming
('udp://239.255.0.1:10124');

end;

if FMultipurposeUDPStreamingID > -1 **then begin**

/// uncomment one of the lines below to use a specific codec, e.g.:

//VideoGrabber.Encoder_SetStr (FMultipurposeUDPStreamingID, Enc_Video_Codec, 'hevc');

//VideoGrabber.Encoder_SetStr (FMultipurposeUDPStreamingID, Enc_Video_Codec, 'h264');

VideoGrabber.Encoder_SetInt (FMultipurposeUDPStreamingID, Enc_Video_Enabled_bool, 1);

VideoGrabber.Encoder_SetInt (FMultipurposeUDPStreamingID, Enc_Audio_Enabled_bool, 1);

if not chkStreamingVideoEnabled.Checked **then begin**

VideoGrabber.Encoder_SetInt (FMultipurposeUDPStreamingID, Enc_Video_Enabled_bool, 0);

end;

if not chkStreamingAudioEnabled.Checked **then begin**

VideoGrabber.Encoder_SetInt (FMultipurposeUDPStreamingID, Enc_Audio_Enabled_bool, 0);

end;

VideoGrabber.Encoder_SetInt (FMultipurposeUDPStreamingID, Enc_Video_BitRate_kb, 2000);

VideoGrabber.Encoder_SetInt (FMultipurposeUDPStreamingID, Enc_Video_rc_MinBitRate_kb, 1800);

VideoGrabber.Encoder_SetInt (FMultipurposeUDPStreamingID, Enc_Video_rc_MaxBitRate_kb, 2200);

VideoGrabber.Encoder_SetInt (FMultipurposeUDPStreamingID, Enc_Video_IDR_Interval, 30);

/// uncomment to enable the GPU encoder, if available for h264 or hevc

// VideoGrabber.Encoder_SetInt (FStreamingEncoderId, Enc_Video_GPU_Encoder, LongInt
(Enc_GPU_Auto)); // or specify directly Enc_GPU_Intel_QSV, Enc_GPU_NVidia_NVENC or

Enc_GPU_AMD_AMF

VideoGrabber.Encoder_SetInt (FMultipurposeUDPStreamingID, Enc_IsActive_bool, 1);

end;

VideoGrabber.VideoSource := vs_ScreenRecording;

VideoGrabber.UseNearestVideoSize (1920, 1080, true);

VideoGrabber.StartPreview();

end;

private void btnMultipurposeUDPStreaming_Click(**object** sender, EventArgs e)

if (m_MultipurposeUDPStreamingID == -1)

m_MultipurposeUDPStreamingID=VideoGrabber.Encoders_CreateInstanceForStreaming("udp://239.255.0.1:10124");

if (m_MultipurposeUDPStreamingID > -1)

/// uncomment one of the lines below to use a specific codec, e.g.:

//VideoGrabber.Encoder_SetStr (FMultipurposeUDPStreamingID, TEncoder_str.Enc_Video_Codec,
"hevc");

//VideoGrabber.Encoder_SetStr (FMultipurposeUDPStreamingID, TEncoder_str.Enc_Video_Codec,
"h264");

```
VideoGrabber.Encoder_SetInt(m_MultipurposeUDPStreamingID, TEncoder_int.Enc_Video_Enabled, 1);
VideoGrabber.Encoder_SetInt(m_MultipurposeUDPStreamingID, TEncoder_int.Enc_Audio_Enabled, 1);
if (!chkVideoStreamingEnabled.Checked)
VideoGrabber.Encoder_SetInt(m_MultipurposeUDPStreamingID, TEncoder_int.Enc_Video_Enabled, 0);

if (!chkAudioStreamingEnabled.Checked)
VideoGrabber.Encoder_SetInt(m_MultipurposeUDPStreamingID, TEncoder_int.Enc_Audio_Enabled, 0);

VideoGrabber.Encoder_SetInt(m_MultipurposeUDPStreamingID, TEncoder_int.Enc_Video_BitRate_kb,
2000);
VideoGrabber.Encoder_SetInt(m_MultipurposeUDPStreamingID,
TEncoder_int.Enc_Video_rc_MinBitRate_kb, 1800);
VideoGrabber.Encoder_SetInt(m_MultipurposeUDPStreamingID,
TEncoder_int.Enc_Video_rc_MaxBitRate_kb, 2200);
VideoGrabber.Encoder_SetInt(m_MultipurposeUDPStreamingID, TEncoder_int.Enc_Video_IDR_Interval,
30);
/// uncomment to enable the GPU encoder, if (available for h264 or hevc
// VideoGrabber.Encoder_SetInt (FStreamingEncoderId, TEncoder_int.Enc_Video_GPU_Encoder, LongInt
(TEncoder_int.Enc_GPU_Auto)); // or specify directly Enc_GPU_Intel_QSV, Enc_GPU_NVidia_NVENC or
Enc_GPU_AMD_AMF
VideoGrabber.Encoder_SetInt(m_MultipurposeUDPStreamingID, TEncoder_int.Enc_IsActive_bool, 1);

VideoGrabber.VideoSource=vs_ScreenRecording;
VideoGrabber.UseNearestVideoSize (1920, 1080, true);
VideoGrabber.StartPreview();
```

Description

Creating and configuring an instance

To create an instance that will stream to the network, invoke [Encoders.CreateInstanceForStreaming](#)

The function return an unique ID that is attributed when creating the instance.

To configure then encoder, this ID must be passed to [Encoder_SetStr](#) and [Encoder_SetInt](#).

While the video is running:

- the instance can be paused with [Encoder_Pause](#) and resumed with [Encoder_Resume](#)

Once the video is stopped (VideoGrabber.Stop()):

- an encoder instance can be removed by invoking [Encoders.RemoveInstance](#), otherwise it will be active again when restarting the video.

- to remove all the encoders instances associated with a TVideoGrabber component invoke

[Encoders.RemoveAllInstances](#)

See Also

[TNetworkStreamingType](#) [TOnClientConnection](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#)
[ASF_AUDIO_BitRate](#) [ASF_AUDIO_Channels](#) [ASF_DeinterlaceMode](#) [ASF_FixedFrameRate](#)
[ASF_MediaServerPublishingPoint](#) [ASF_MediaServerRemovePublishingPointAfterDisconnect](#)
[ASF_MediaServerTemplatePublishingPoint](#) [ASF_NetworkMaxUsers](#) [ASF_NetworkPort](#) [ASF_Profile](#)
[ASF_ProfileFromCustomFile](#) [ASF_Profiles](#) [ASF_ProfilesCount](#) [ASF_ProfileVersion](#) [ASF_VideoBitRate](#)
[ASF_VideoFrameRate](#) [ASF_VideoHeight](#) [ASF_VideoMaxKeyFrameSpacing](#) [ASF_VideoQuality](#) [ASF_VideoWidth](#)
[NetworkStreaming](#) [NetworkStreamingType](#) [OnAuthenticationNeeded](#) [OnClientConnection](#)
[OnDirectNetworkStreamingHostUrl](#) [ShowDialog](#) [StartAudioRendering](#) [StreamingURL](#)

Created with the Standard Edition of HelpNDoc: Effortlessly create a professional-quality documentation website with HelpNDoc

MMS streaming

MMS streaming

[Prev](#)

[Next](#)

MMS Streaming

Description

The MMS streaming is activated when [NetworkStreaming](#) = ns_ASFDirectNetworkStreaming.

The TCP port used can be specified with [ASFNetworkPort](#).

If [ASFNetworkPort](#) = 0, an available port is automatically assigned. This port is returned by the [OnDirectNetworkStreamingHostUrl](#). This event returns also the **full URL** that can be given to the users to connect to this network stream.

The streaming URL that must be given to users to connect to the current stream is also returned by the [StreamingURL](#) property.

Increasing the default limitation of 5 concurrent clients

By default a standard ("non-server") Windows PC is limited to 5 concurrent connections.

This limit can be increased to a maximum of 50 connections by modifying the value of the HKEY_CLASSES_ROOTMedia Tools**MaxClientConnections** registry key.

Controlling the users connected

- the maximum number of users that can connect to this network stream is specified by the [ASFNetworkMaxUsers](#) property (default value : 5 users).
- the current number of users connected is returned by the [ASFStreaming_GetConnectedClientsCount](#) function.
- the list of the users connected is returned by the [ASFStreaming_GetConnectedClients](#) string (as lines separated by CR/LF)
- it is possible to block an IP address (or a group of IP addresses) by invoking [ASFStreaming_SetAuthorization](#)
- the list of the blocked IP addresses is returned by the [ASFStreaming_GetAuthorizationList](#) string (as lines separated by CR/LF)
- to clear the list of the blocked IP addresses, invoke [ASFStreaming_ResetAuthorizations](#).

SAMPLE CODE (server side)

Streaming a live video source (video only):

```
VideoGrabber1.VideoDevice := 0;
VideoGrabber1.NetworkStreaming := ns_ASFDirectNetworkStreaming;
VideoGrabber1.NetworkStreamingType := nst_VideoStreaming;
VideoGrabber1.ASFNetworkPort := 10500;
VideoGrabber1.StartPreview;
```

Streaming a live video source (audio + video):

```
VideoGrabber1.VideoDevice := 0;
VideoGrabber1.AudioDevice := 0;
VideoGrabber1.NetworkStreaming := ns_ASFDirectNetworkStreaming;
VideoGrabber1.NetworkStreamingType := nst_AudioVideoStreaming;
VideoGrabber1.ASFNetworkPort := 10500;
VideoGrabber1.StartPreview;
```

Streaming a video clip:

```
VideoGrabber1.VideoSource := vs_VideoFileOrURL;
VideoGrabber1.VideoSource_FileOrUrl := " replace by the file path to your
video clip"
VideoGrabber1.NetworkStreaming := ns_ASFDirectNetworkStreaming;
VideoGrabber1.NetworkStreamingType := nst_AudioVideoStreaming;
VideoGrabber1.ASFNetworkPort := 10500;
VideoGrabber1.StartPreview;
```

To view the streaming on the client, you need to open the server URL, that is built as follows:

mms:// IP : port

E.g. if the server IP is 192.168.0.123 and the streaming port is 10500 the URL will be
mms://192.168.0.123:10500

To know the IP address of the server.

- either run IPConfig on the server to show it,
- either use the OnDirectNetworkStreamingHostUrl event (on the server) that returns directly the URL to open in the client when starting the preview or recording (HostUrl parameter).

SAMPLE CODE (client side)

E.g. if the server IP is 192.168.0.123 and the port is 10500 the URL will be mms://192.168.0.123:10500

- Without the optional [Datastead RTSP DirectShow source filter](#):

```
VideoGrabber1.VideoSource := vs_VideoFileOrURL;
VideoGrabber1.VideoSource_FileOrUrl := 'mms://192.168.0.123:10500';
VideoGrabber1.StartPreview;
```

- With the optional [Datastead RTSP DirectShow source filter](#) installed on the client:

```
VideoGrabber1.VideoSource := vs_VideoFileOrURL;
VideoGrabber1.VideoSource_FileOrUrl := 'mmsh://192.168.0.123:10500';
VideoGrabber1.StartPreview;
```

Created with the Standard Edition of HelpNDoc: [Effortlessly upgrade your WinHelp HLP help files to CHM with HelpNDoc](#)

Streaming using Newtek NDI

Streaming using Newtek NDI

[Prev](#)
[Next](#)

Streaming using NEWTEK NDI

Description

The NDI streaming requires the Newtek NDI runtime, and the optional Datastead NDI Filters package that can be downloaded here :

<https://www.datastead.com/downloads/>

The installation of the Datastead NDI filters and the Newtek NDI runtime are described in the
INSTALL_README.rtf

Sample code to send a stream to the network:

```
VideoGrabber.VideoSource = ...
VideoGrabber.NDIName = "MYSTREAM";
VideoGrabber.NetworkStreaming = ns_NDI;
VideoGrabber.NetworkStreamingType = nst_VideoStreaming; // or nst_AudioVideoStreaming if audio
needed
VideoGrabber.StartPreview();
E.g. if the video source is a Logitech 930e webcam:
VideoGrabber.VideoSource = vs_VideoCaptureDevice;
VideoGrabber.VideoDevice = VideoGrabber.VideoDeviceIndex ("Logitech Webcam C930e");
VideoGrabber.AudioDevice = VideoGrabber.AudioDeviceIndex ("Microphone (Logitech Webcam C930e)");
VideoGrabber.NDIName = "MYSTREAM";
VideoGrabber.NetworkStreaming = ns_NDI;
VideoGrabber.NetworkStreamingType = nst_AudioVideoStreaming;
VideoGrabber.StartPreview();
```

Sample code to play a stream from the network:

```
VideoGrabber.VideoSource = vs_VideoFileOrURL;
VideoGrabber.VideoSource_FileOrUrl = "ndi://DESKTOP-5B66SBT (MYSTREAM)";
VideoGrabber.AudioDeviceRendering = true ;
VideoGrabber.StartPreview();
```

Notes:

- replace "DESKTOP-5B66SBT" by the name of the computer that is streaming to the network, and "MYSTREAM" by the real stream name to use.
- if the sender and receiver are on the same computer, the computer name can be replaced by "localhost", e.g.:
Videograbber.VideoSource_FileOrURL = "ndi://localhost (MYSTREAM)";

See Also

[TNDIFormatType](#) [NDIFormatType](#) [NDIName](#) [NetworkStreaming](#) [SetDatasteadFilterDllName](#)

Created with the Standard Edition of HelpNDoc: [Upgrade your help files and your workflow with HelpNDoc's WinHelp HLP to CHM conversion](#)

Video window - Display - Aspect Ratio - Dual display

Created with the Standard Edition of HelpNDoc: [Achieve Professional Documentation Results with a Help Authoring Tool](#)

Video window

Video window

[Prev](#)
[Next](#)

Video size overview.

Description

Video render

You can choose the video renderer through the [VideoRenderer](#) property:

vr_AutoSelect: selected automatically

vr_VMR9: video renderer 9 (VMR9) -> good quality, more CPU. It buffers 3 frames, this can cause positioning problems in the player.

vr_VMR7: video renderer 7 (VMR7) -> good quality, less CPU

vr_EVR: enhanced video renderer (EVR), best quality, more CPU

vr_StandardRenderer: standard renderer -> average quality, less CPU

vr_OverlayRenderer: overlay renderer -> good quality, low CPU

vr_RecordingPriority: recommended if the quality of the recording is more important than the quality of the preview.

vr_None: no renderer

Keeping the last video frame visible while opening the next video clip

By default after invoking [StopPreview](#), [StopRecording](#) or [ClosePlayer](#), the video window is closed until the next clip is opened.

Enable the [VideoVisibleWhenStopped](#) property to keep the video window active and the last frame displayed until [StartPreview](#), [StartRecording](#) or [OpenPlayer](#) start displaying video frames.

Show / hide the video window

Enable [Display_Active](#) to specify if the video window must be constructed when invoking StartPreview / StartRecording / OpenPlayer.

Then the video window can be shown or hidden when the preview/recording/playback is running by enabling or disabling [Display_Visible](#).

Size of the video window & auto resizing

The control is automatically resized by TVideoGrabber when [Display_AutoSize](#) is enabled, otherwise it depends of the Width and Height properties.

When [Display_AutoSize](#) is enabled:

- selecting a [VideoSize](#) in the [VideoSizes](#) list resizes the control to the selected size, remark: the size of the control can be stretched to a given percent of the native video size by using [PreviewZoomSize](#). This property does not affect the frame capture size and the recording size.

Display aspect ratio

Video clips can have a display aspect ratio different than the frame aspect ratio. TVideoGrabber detects the display aspect ratio and takes it in account when the [AdjustPixelAspectRatio](#) property is enabled (see below).

In this case, the [VideoWidth_PreferedAspectRatio](#) and the [VideoHeight_PreferedAspectRatio](#) properties return the preferred width and height that should be used to display the clip. These properties are useful if you want to control yourself the video size when [Display_AutoSize](#) is disabled.

To display the video in its correct display aspect ratio:

- if you want the control to be resized automatically, set [AdjustPixelAspectRatio](#) = true **and** [Display_AutoSize](#) = true

- if you want to specify yourself the size of the control, set [AdjustPixelAspectRatio](#) = true **and** [Display_AutoSize](#) = false, and read the [VideoWidth_PreferedAspectRatio](#) and [VideoHeight_PreferedAspectRatio](#) properties from the [OnPlayerOpened](#) event to get the correct video width and height of the video clip.

To display the video AS IS:

- if you want the control to be resized automatically, set [AdjustPixelAspectRatio](#) = false **and** [Display_AutoSize](#) = true

- if you want to specify yourself the size of the control, set [AdjustPixelAspectRatio](#) = false **and** [Display_AutoSize](#) = false **and** [Display_AspectRatio](#) = ar_NoResize and read the VideoWidth and VideoHeight properties from the [OnPlayerOpened](#), [OnPreviewStarted](#) or [OnRecordingStarted](#) events.

Location of the video frames within the video window

When [Display_AutoSize](#) = false, the aspect ratio is controlled by the [Display_AspectRatio](#) property. The possible settings are:

ar_Box: the video appears in full within the control in a letterbox (bars on the top and the bottom) or pillarbox (bars on the left and the right)

ar_NoResize: the video appears in its "native" size. If the native size is larger than then control, it appears centered within the control. If the native size is larger than the control, the video appears truncated.

ar_Stretch: the video is stretched as needed to fill the control

ar_PanScan: from pan to tilt, according to the [Display_PanScanRatio](#) value (in the 0..100 range, default value 50).

Zooming the frame capture

The size of captured frames can be stretched to a given percent of the native video size by using [FrameCaptureZoomSize](#) .

Analog video sources

For non-DV video sources, [VideoSizes](#) returns the list of video sizes really supported by the [current video capture device](#) (e.g. 320x240, 352x288, 640x480, etc...)/

DV video sources

DV video sizes are standard.

When the current video capture device is a DV source, the VideoSizes returns "dc", "quarter", "half" and "full". The resulting video size depends of the video signal (PAL or NTSC):

| VideoSizes list | NTSC | PAL |
|------------------------|-------------|------------|
| default | 360x240 | 360x288 |
| dc | 88x66 | 88x72 |
| quarter | 180x120 | 180x144 |
| half | 360x240 | 360x288 |
| full | 720x480 | 720x576 |

Recording size

The recording size depends of the [VideoSize](#) property, except when [UseNearestVideoSize](#) is used.

However, when capturing DV video sources and [RecordingInNativeFormat](#) is enabled, the video stream is always saved to AVI in "full" size, and [VideoSize](#) and [UseNearestVideoSize](#) apply in this case only to the

preview control size.

refreshing the video sizes

Any listbox that uses the video sizes of the current video capture device should be refreshed with [VideoSizes](#) ([VideoSize](#) is the current index in this list) when the [OnVideoDeviceSelected](#) event occurs. See the *MainDemo* project for sample code.

Video quality under Windows Vista / Windows 7 and later

It is possible to improve the quality of the video display (e.g. in full screen mode) when playing MPEG clips and the frame grabber is disabled, see the [Aero](#) property.

Custom renderer

If your video display board is shipped with a custom renderer, you can tell TVideoGrabber to use it. You have to invoke the `ThirdPartyFilter_AddToList` function before invoking `StartPreview`, and pass the CLSID of your third-party filter.

E.g.:

```
...
uses ComObj;
...
procedure TfrmMainForm.Button1Click(Sender: TObject);
const
  CLSID_MyRendererFilter: TGUID = '70e102b0-5556-11ce-97c0-00aa0055595a';
begin
  if VideoGrabber.ThirdPartyFilter_AddToList (tpf_VideoRenderer, GUIDToString (CLSID_MyRendererFilter)) then
    ShowMessage ('filter added');
    VideoGrabber.StartPreview;
  end;
end;
```

Feel free to contact us at support@datastead.com if you don't know how to retrieve the CLSID of your renderer filter.

See Also

[GetNearestVideoHeight](#) [GetNearestVideoSize](#) [GetNearestVideoWidth](#) [GetVideoHeightFromIndex](#) [GetVideoSizeFromIndex](#) [GetVideoWidthFromIndex](#) [UseNearestVideoSize](#) [VideoFormat](#) [VideoFormatIndex](#) [VideoSize](#) [VideoSizeIndex](#) [VideoSizes](#) [VideoSizesCount](#)

Created with the Standard Edition of HelpNDoc: [Easily create EBooks](#)

Dual display

Dual display

[Prev](#)

[Next](#)

Dual display.

Description

overview

It is possible to display 2 identical video windows, e.g. for a teacher to see the video preview on a 1st monitor, and for students to see the same video on a 2nd monitor in full screen mode.

The default video window is handled through the "Display_" properties and methods.

The 2nd video window is handled through the "DualDisplay_" identical properties and methods.

default video window

By default, only the 1st video window is active ([Display_Active](#) = true) and the 2nd video window is disabled ([DualDisplay_Active](#) = false).

By default, the 1st video window is embedded in the TVideoGrabber control ([Display_Embedded](#) = true).

activation/deactivation of the dual display

To activate the 2nd video window, enable [DualDisplay_Active](#) before calling [StartPreview](#) , [StartRecording](#) or [OpenPlayer](#) .

Then the 2nd video window can be shown or hidden when the preview/recording/playback is running by enabling or disabling [DualDisplay_Visible](#).

multi-monitors display

You can send each video window on the monitor of your choice. Simply assign the [Display_Monitor](#) or [DualDisplay_Monitor](#) property with the index of the desired monitor.

2 global functions help you to get information on the available monitors:

- 1) [MonitorsCount](#) returns the number of monitors available on the current platform.
Therefore [Display_Monitor](#) or [DualDisplay_Monitor](#) accept a value in the 0..[MonitorsCount](#) -1 range.
- 2) [MonitorBounds](#) returns the bounds of the specified monitor in the 0..[MonitorsCount](#) -1 range.
The bounds returned are intended to help you to position the video window within the screen bounds.

window location

When it is not embedded ([Display_Embedded](#) = false), the video window:

- can be placed on the screen by assigning [Display_Left](#) and [Display_Top](#) ,
- can be sized by assigning [Display_Width](#) and [Display_Height](#) .

However, rather than assigning these properties one by one, you can set all of them at the same time by using [Display_SetLocation](#) .

See Also

[Display_AlphaBlendEnabled](#) [Display_AlphaBlendValue](#) [Display_TransparentColorEnabled](#)
[Display_TransparentColorValue](#) [DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#)
[DualDisplay_AlphaBlendValue](#) [DualDisplay_AutoSize](#) [DualDisplay_Embedded](#) [DualDisplay_FullScreen](#)
[DualDisplay_Height](#) [DualDisplay_Left](#) [DualDisplay_Monitor](#) [DualDisplay_MouseMovesWindow](#)
[DualDisplay_PanScanRatio](#) [DualDisplay_SetLocation](#) [DualDisplay_StayOnTop](#) [DualDisplay_Top](#)
[DualDisplay_TransparentColorEnabled](#) [DualDisplay_TransparentColorValue](#) [DualDisplay_VideoHeight](#)
[DualDisplay_VideoPortEnabled](#) [DualDisplay_VideoWidth](#) [DualDisplay_VideoWindowHandle](#)
[DualDisplay_Visible](#) [DualDisplay_Width](#) [Monitor_Primary_Index](#) [MonitorBounds](#) [MonitorsCount](#)
[SetWindowTransparency](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#)

Created with the Standard Edition of HelpNDoc: [HelpNDoc's Project Analyzer: Incredible documentation assistant](#)

Multiple video windows

Multiple video windows

[Prev](#)

[Next](#)

Multiple video windows ([dual display](#) feature "extended" to more than 2 windows).

Description

The [Display_...](#) and [DualDisplay_...](#) features allow to control 2 video windows for the same source (e.g. a preview control embedded in the application, and a 2nd full screen window on another monitor)

It is possible to use more than 2 video windows by invoking directly the [SetDisplay...](#) functions, and by passing the index of the window, in the 2..8 range (0 and 1 are reserved respectively to the [Display...](#) and [DualDisplay...](#) properties)

Use the following sample code for 1 embedded preview (enabled by default, corresponding to the "Display... properties") and additional 2 full screen monitors:

```
Videograbber.SetDisplayActive (2, true);
Videograbber.SetDisplayActive (3, true);
```

Videograbber.StartPreview();

```
Videograbber.SetDisplayMonitor (2, 1);
Videograbber.SetDisplayMonitor (3, 2);
Videograbber.SetDisplayFullScreen (2, true);
```

Videograbber.SetDisplayFullScreen (3, true);

The functions related to the multiple windows are:

SetDisplayActive

SetDisplayAlphaBlendEnabled
 SetDisplayAlphaBlendValue
 SetDisplayAssociatedRenderer
 SetDisplayAutoSize
 SetDisplayAspectRatio
 SetDisplayEmbedded
 SetDisplayFullScreen
 SetDisplayMonitor
 SetDisplayMouseMovesWindow
 SetDisplayPanScanRatio
 SetDisplayParent
 SetDisplayStayOnTop
 SetDisplayTransparentColorEnabled
 SetDisplayTransparentColorValue
 SetDisplayVideoPortEnabled
 SetDisplayVisible

SetDisplayLeft
 SetDisplayTop
 SetDisplayWidth
 SetDisplayHeight

SetDisplayLocation

Created with the Standard Edition of HelpNDoc: [Bring your WinHelp HLP help files into the present with HelpNDoc's easy CHM conversion](#)

Transparency - Color keying

Transparency - Color keying

[Prev](#)

[Next](#)

Window transparency overview.

Description

Alphablending and color keying when the video window is detached from the control

When the video window is detached from the control ([Display_Embedded](#) = false), it is possible to superpose another video window and to apply semi transparency AND/OR color keying.

The semi transparency is enabled by [Display_AlphaBlendEnabled](#), and the semi transparent value is specified by [Display_AlphaBlendValue](#) (in the 0..255 range)

The color keying is enabled by [Display_TransparentColorEnabled](#), and the RGB color key value is specified by [Display_TransparentColorValue](#)

Transparency of another video window when the video window is embedded in the control

To use window transparency, simply enable the [ColorKeyEnabled](#) property, then invoke [SetWindowTransparency](#) to set the color key of your form (or window) to make it transparent when moved over the video window.

You have to pass to [SetWindowTransparency](#) the default TVideoGrabber ColorKey. (you can customize the ColorKey value if needed)

Optionally the color key can be defined by the [ColorKey](#) property.

Look at the transparency checkbox code in the "display" tab of the MainDemo project for sample code.

See Also

[TOnColorKeyChange](#) [ColorKeyEnabled](#) [OnColorKeyChange](#)

Created with the Standard Edition of HelpNDoc: [Easily create Web Help sites](#)

madVR video renderer

madVR video renderer

[Prev](#)

[Next](#)

madVR Video Renderer

Description

The madVR video renderer is supported by installing the package in a folder of your choice and setting VideoGrabber.VideoRenderer = vr_madVR.

The filter must be first installed by downloading the madVR package from <http://madvr.com>

After unzipping the package, is it possible to:

- either register the madVR renderer filter in DirectShow by running the install.bat
- either just specify the madVR folder to the [ExtraDLLPath](#) property. In this case no registration of the madVR renderer binaries is needed.

To configure the madVR renderer:

- connect the [OnThirdPartyFilterConnected](#) event, that passes the instance of the filter through the Intf parameter
- invoke Intf.QueryInterface (IMadVRSettings, MadVRSettings)
- invoke the MadVRSetting configuration functions

Created with the Standard Edition of HelpNDoc: [Elevate your documentation to new heights with HelpNDoc's built-in SEO](#)

Third-party external renderers

Third-party external renderers

[Prev](#)

[Next](#)

Third party external video renderers

Description

Other third-party video renderers

(if you are using a BlackMagic Decklink, see the [Blackmagic Decklink cards](#) chapter)

If your video card is shipped with a third-party custom renderer filter, it is possible to let TVideoGrabber use it instead of the default video renderers.

You must pass to TVideoGrabber the CLSID of the filter. You can retrieve it by searching the registry for the name of the filter.

Feel free to contact us at support@datastead.com if you have problems to find the CLSID.

Here is some sample code to use a custom renderer filter:

```
...
uses ComObj;
...
procedure TfrmMainForm.Button1Click(Sender: TObject);
const
  CLSID_MyRendererFilter: TGUID = '70e102b0-5556-11ce-97c0-00aa0055595a'; // replace the
```



```
begin
    if VideoGrabber.ThirdPartyFilter_AddToList (tpf_VideoRenderer, GUIDToString (CLSID_MyF
        ShowMessage ('filter added');
        VideoGrabber.StartPreview;
    end;
end;
```

Created with the Standard Edition of HelpNDoc: [Effortlessly upgrade your WinHelp HLP help files to CHM with HelpNDoc](#)

Screen recording - desktop capture

Created with the Standard Edition of HelpNDoc: [Experience the Power and Ease of Use of a Help Authoring Tool](#)

screen capture and recording with or without cursor

screen capture and recording with or without cursor

[Prev](#)

[Next](#)

Screen recording

Description

Screen recording

The screen recording is enabled when [VideoSource](#) = **vs_ScreenRecording**.

Then, all the preview and recording features of the video capture devices can be applied:

- the preview by invoking [StartPreview](#),
- the frame capture by invoking [CaptureFrameTo](#),
- the recording by invoking [StartRecording](#),
- the compression on the fly or after capture (see the **Recording** section)
- the [video processings](#) and [graphics and text overlays](#) when the frame grabber is enabled.

The recording of the cursor is enabled or disabled with [ScreenRecordingWithCursor](#).

The capture of layered windows (e.g. floating or transparent windows) can be enabled by the [ScreenRecordingLayeredWindows](#) property.

If some areas of the screen are not properly recorded, try to enable or disable the [ScreenRecordingThroughClipboard](#) property.

WMV screen recording compressed with the Windows Media screen codec

when recording in WMV, the following settings will automatically use the Microsoft's Windows Media screen codec:/line

```
VideoGrabber.VideoSource = vs_ScreenRecording
VideoGrabber.RecordingMethod = rm_ASF
VideoGrabber.ASFProfileVersion = apv_ProfileVersion_9
VideoGrabber.ASFVideoQuality = 95
VideoGrabber.StartRecording()
```

or

```
VideoGrabber.VideoSource = vs_ScreenRecording
VideoGrabber.RecordingMethod = rm_ASF
VideoGrabber.ASFProfileVersion = apv_ProfileVersion_9
VideoGrabber.ASFVideoBitRate = 2000000
```



```
VideoGrabber.StartRecording()
```

(note: ASFVideoBitRate and ASFVideoQuality are exclusive, ASFVideoQuality is mandatory)

If you are using a .prx profile, select the screen codec in the codec list when creating it with WMPProEdt.exe

AVI screen recording compressed by using third-party codecs

The recording will give better results (quality/size) if codecs specifically designed for this purpose are used for the recording.

E.g. by using the Innoheim ISCC codec:/line

```
VideoGrabber.VideoSource = vs_ScreenRecording
VideoGrabber.RecordingMethod = rm_AVI
VideoGrabber.VideoCompressor = VideoGrabber.VideoCompressorIndex ("inno Screen Capture Co
VideoGrabber.CompressionType = ct_Video
if VideoGrabber.VideoCompressor > -1 then
    VideoGrabber.StartRecording()
else
    MessageBox ("screen codec not installed")
end
```

Specifying the monitor(s) to record when more than 1 monitor is available

If more than one monitor is available, the number of the monitor to record is specified by the [ScreenRecordingMonitor](#) property.

- to record the default monitor set ScreenRecordingMonitor = 0 (default value).
- to record the 2nd monitor, set ScreenRecordingMonitor = 1, and so on...

Recording the extended desktop

Simply set [ScreenRecordingMonitor](#) = -1, this will record the whole extended desktop across all the monitors.

Remarks:

- the preview or recording frame rate of the screen display is specified by the [FrameRate](#) property.
- when recording the full screen, it is recommended to use a low frame rate to prevent dropped frames.
- it is possible to record only a part of the screen by specifying a [cropping area](#).
- when [ScreenRecordingThroughClipboard](#) is enabled, the clipboard is continuously flushed during preview and recording.

See Also

[TOnEnumerateWindows](#) [EnumerateWindows](#) [ScreenRecordingLayeredWindows](#) [ScreenRecordingMonitor](#) [ScreenRecordingNonVisibleWindows](#) [ScreenRecordingUsingCoordinates](#) [ScreenRecordingWithCursor](#) [SetWindowRecordingByHandle](#) [SetWindowRecordingByName](#)

Created with the Standard Edition of HelpNDoc: [Why Microsoft Word Isn't Cut Out for Documentation: The Benefits of a Help Authoring Tool](#)

screen capture and recording of a specified window

screen capture and recording of a specified window

[Prev](#)
[Next](#)

Screen recording of a window

Description

Screen recording of a window

It is possible to record only a window, instead of the whole screen.

There are 2 constraints:

- the window may not be resized after the recording starts,
- the window must not be overlapped by another window.

Identifying the video window to record

It is necessary to specify to the component the window to be recorded.
This can be achieved by 2 ways.

1. simple way

- identify the name of the window to record by looking at its title bar, and invoke [SetWindowRecordingByName](#) by passing the whole title bar, or only a substring that identifies it.
- then you can pass:
 - . either the full string to the WindowName parameter with ExactMatch = true
 - . either a substring picked up in the window name, with ExactMatch = false (in this case the string is not case-sensitive).

2. advanced way

Use [EnumerateWindows](#) to retrieve the window name.

After invoking [EnumerateWindows](#), the [OnEnumerateWindows](#) will occur for each visible window, and will return:

- the window name
- the name of the class of the window
- the window handle

Then, after choosing the window, you can specify it by invoking [SetWindowRecordingByHandle](#) and passing the window handle returned by the [OnEnumerateWindows](#) event.

Starting the window recording

- specify the window to record by using the 1. or 2. method above,
- set [VideoSource](#) = vs_ScreenRecording
- invoke [StartRecording](#) to record, or [StartPreview](#) and then [CaptureFrameTo](#) to make single window captures.

Window is destroyed during preview or recording

If the window is destroyed during preview or recording, the preview or recording stops.

Screen capture and recording of non-visible (hidden) windows

It is possible to enable the capture of non-visible windows by enabling the [ScreenRecordingNonVisibleWindows](#) property.

Note: this mode may give variable results depending on the content of the window

Created with the Standard Edition of HelpNDoc: [Simplify Your Help Documentation Process with a Help Authoring Tool](#)

Recording only a part of the screen

Recording only a part of the screen

[Prev](#)

[Next](#)

Recording only a part of the screen

Description

Recording only a part of the screen

It is possible to limit the recording to a part of the screen by specifying the width and height that must be recorded at the x, y location.

To activate this feature simply invoke `ScreenRecordingUsingCoordinates` before invoking `StartPreview` or `StartRecording`.

E.g.

```
VideoGrabber.VideoSource := vs_ScreenRecording;
VideoGrabber.ScreenRecordingUsingCoordinates (true, 10, 10, 300, 100);
VideoGrabber.StartPreview;
```

Remarks:

- the `scLeft` and `scTop` location may be modified dynamically while previewing or recording
- this feature is inactive if the recording of a window has been activated with [SetWindowRecordingByName](#) or [SetWindowRecordingByHandle](#) .

Created with the Standard Edition of HelpNDoc: [Modernize your help files with HelpNDoc's WinHelp HLP to CHM conversion tool](#)

Desktop capture

Desktop capture

[Prev](#)

[Next](#)

Desktop capture

Description

How to perform a desktop capture

- set [VideoSource](#) = `vs_ScreenRecording`
- invoke [StartPreview](#)
- minimizes the app (or make it non-visible)
- then it is possible to capture the desktop image to a memory bitmap or a BMP or JPEG file by invoking [CaptureFrameTo](#), or perform a screen recording as explained in the [recording the screen with or without cursor](#) chapter.

This can be quick tested in 2 clicks by using `MainDemo.exe` included in the package as follows:

- run `MainDemo.exe`
- in the "video source" tab, select video source = "screen recording"
- click `StartPreview`
- go to the "frame grabber" tab
- move the `MainDemo.exe` window at the bottom of the desktop (to hide most of the `MainDemo.exe` app), just keep the buttons of "one shot frame capture" group visible,
- then click "BMP file" or "JPEG file" to capture the desktop image to a file, or "TBitmap" to capture it to a new window.

Of course, the app must be kept minimized (or not visible) to capture the desktop without capturing your app's window.

Remark: if some special windows appear black in the captured image, enable the [ScreenRecordingThroughClipboard](#) property.

Created with the Standard Edition of HelpNDoc: [Simplify Your Help Documentation Process with a Help Authoring Tool](#)

Recording on the fly of video clips and live streams

Created with the Standard Edition of HelpNDoc: [Maximize Your Documentation Capabilities with a Help Authoring Tool](#)

video clips and live streams recorded on the fly

video clips and live streams recorded on the fly

[Prev](#)
[Next](#)

Recording "on the fly" of video clips and live streams

Description

Preview and recording "on the fly" of video clips and live streams

It is possible to record "on the fly" video clips and live streams.

- 1) first of all set [VideoSource](#) = [vs_VideoFileOrURL](#) to select a video clip or a live stream as video source.
- 2) then set the [VideoSource_FileOrUrl](#) property with the path or the URL of the video clip or the live stream. Then, all the preview and recording features of the video capture devices can be applied:

Default settings

- the audio rendering is enabled by [AudioDeviceRendering](#),
- the frame capture can be used by invoking [CaptureFrameTo](#) if the [frame grabber](#) is enabled.
- the [video processings](#) and [graphic overlays](#) can be applied if the [frame grabber](#) is enabled.

Preview

The preview is started by invoking [StartPreview](#).

Recording

- the (re)compression can be applied on the fly through the [Datastead Encoder](#)
- the recording is started by invoking [StartRecording](#)

Created with the Standard Edition of HelpNDoc: [Say Goodbye to Documentation Headaches with a Help Authoring Tool](#)

Video clips built on the fly from Bitmap handles, BMP or JPEG files

Created with the Standard Edition of HelpNDoc: [Make Help Documentation a Breeze with a Help Authoring Tool](#)

Video clips built on the fly by passing bitmap handles, BMP or JPEG files

Video clips built on the fly by passing bitmap handles, BMP or JPEG files

[Prev](#)
[Next](#)

How to create a video clip directly from bitmap handles.

Description

A video clip can be created on the fly by directly passing either directly memory bitmap handles, either the file name of BMP or JPEG files.

This feature is activated when [VideoSource](#) is set to [vs_JPEGsOrBitmaps](#).

The frames are passed by invoking [SendImageToVideoFromBitmaps](#) and passing as parameter:

- either the path to one or the BMP or JPEG files ([ImageFilePath](#) parameter).
- either a bitmap handle ([BitmapHandle](#) parameter, see remark (b) below),

1. BUILDING A VIDEO CLIP BY PASSING IMAGES WHEN THEY ARE RECEIVED, IN REAL TIME (e.g. the images are currently received from another live source at 15 fps)

- set [VideoSource](#) = [vs_JPEGsOrBitmaps](#)
- set [FrameRate](#) with the real frame rate
- invoke [SendImageToVideoFromBitmaps](#) a first time so the component can learn the video format (*if this*

step is omitted the next step will fail)

- invoke [StartPreview\(\)](#) or [StartRecording\(\)](#)
- then invoke periodically [SendImageToVideoFromBitmaps](#) or [SendImageToVideoFromBitmaps2](#) as soon as new frames are available to pass to the component.

I

2. BUILDING A VIDEO CLIP FROM AN EXISTING SET OF IMAGES (e.g. a set of Jpeg files located in a folder)

When [StartPreview](#) or [StartRecording](#) is invoked, the [OnVideoFromBitmapsNextFrameNeeded](#) occurs periodically (depending of the [FrameRate](#) property), requesting for the next image that will be used as a video frame to built the video stream.

From this event, invoke [SendImageToVideoFromBitmaps](#) and pass the bitmap handle or the file name of the next frame.

- set [VideoSource](#) = vs_JPEGsOrBitmaps,
- set [FrameRate](#) with the desired frame rate
- create the [OnVideoFromBitmapsNextFrameNeeded](#) event, and put the code required to pass the bitmap handles, or the the image files,
- from the [OnVideoFromBitmapsNextFrameNeeded](#) event, invoke [SendImageToVideoFromBitmaps](#) or [SendImageToVideoFromBitmaps2](#) and pass as parameter either the file name, either the bitmap handle
- invoke [StartPreview\(\)](#) or [StartRecording\(\)](#)

You can find sample code in the MainDemo project included in the package.

Remarks about [SendImageToVideoFromBitmaps](#):

a) BitmapHandle and **ImageFilePath** are mutually exclusive:

- if you pass a file path to **ImageFilePath**, pass 0 to the **BitmapHandle** paramter
- if you pass a bitmap handle to the **BitmapHandle** parameter, pass an empty string to do not pass a file path to the **ImageFilePath** parameter

b) if the you pass a **BitmapHandle**, enable **CanFreeBitmapHandle** if TVideoGrabber must free the bitmap when it is no longer needed, or disable **CanFreeBitmapHandle** if you need to reuse the bitmap later.

c) all the images passed to this event must have the same format.

Remark about the frame rate

When recording without audio to an AVI file, it is possible to record as fast as possible by setting a very high frame rate, e.g. [FrameRate](#) = 200) and then set the final frame rate when the recording ends from the [OnEndOfAVIRecording](#) event.

- * you can apply the [on the fly compression](#) before invoking StartRecording, if needed.
- * the **FirstSample** parameter is true for the first frame, and false for the others. You can test it e.g. to restart from your first bitmap.
- * set **EndOfData** to true to notify the end of stream. This is equivalent to invoking [StopRecording](#) or [StopPreview](#).

c. if you pass a bitmap handle, no not release it. If the bitmap handle is picked up from a TBitmap object (by passing TBitmap.Handle), be sure to release the handle before freeing the TBitmap.

E.g.:

```
procedure TForm1.VideoGrabberVideoFromBitmapsNextFrameNeeded(Sender: TObject; FirstSample: Boolean;
var
    Bitmap: TBitmap;
begin
    Bitmap := TBitmap.Create;
```

```

Bitmap.Assign (Image1.Picture);
BitmapHandle := Bitmap.Handle;
VideoGrabber. SendImageToVideoFromBitmaps ("", LongInt (BitmapHandle), true, false);
Bitmap.ReleaseHandle;
Bitmap.Free;
end;

```

See Also

[TOnVideoFromBitmapsNextFrameNeeded](#)

Created with the Standard Edition of HelpNDoc: [Transform Your Documentation Process with HelpNDoc's Project Analyzer](#)

Video clips from a fixed set of BMP files or JPEG files merged into a clip

Created with the Standard Edition of HelpNDoc: [Transform Your Word Doc into a Professional-Quality eBook with HelpNDoc](#)

Video clip built from a fixed set of BMP files or JPEG files

Video clip built from a fixed set of BMP files or JPEG files

[Prev](#)

[Next](#)

How to create a video clip from a set of bitmap or BMP files or JPEG files.

Description

How to create a video clip from a set of bitmap or BMP files or JPEG files.

First of all, a temporary file must be created from the set of images (step A).

Then, it is possible:

- to preview the video built from temporary file (step B)
- to record a video clip built from the temporary file to an AVI or ASF file (step C)

A. Create a temporary file from the BMP files or JPG files.

a1. create a set of images as BMP or JPEG files and put them in an empty directory.

These files must be all of the same format (all BMP or all JPG, same frame width and height, same color resolution).

The format used is the format of the first video frame found in the directory. If other video frames have not the same format, they will be ignored.

(e.g. set StoragePath with a new directory name, go to the "frame capture tab" and capture 20 frames to JPG files in burst mode).

a2. specify the location of the image files:

assign the directory containing the image files to "[VideoFromImages_SourceDirectory](#)". E.g.:

VideoGrabber.VideoFromImages_SourceDirectory = "c:/MyImagesFolder"

a3. specify a sorting method for the image files (optional)

choose a [VideoFromImages_BitmapSortedBy](#) value. E.g.:

VideoGrabber.VideoFromImages_BitmapSortedBy = fs_NameAsc

Note: numeric sequences must be "zero padded". E.g.:

001.bmp

002.bmp
 003.bmp
 ...
 009.bmp
 010.bmp
 011.bmp
 ...
 098.bmp
 099.bmp
 100.bmp
 101.bmp
 ...

a4. choose and specify the temporary file that will be created by using the image files

assign this file name to the [VideoFromImages_TemporaryFile](#) property. E.g.

VideoGrabber.VideoFromImages_TemporaryFile = "MyTempFile.dat"

a5. create the temporary file from the set of bitmaps:

Invoke [VideoFromImages_CreateSetOfBitmaps](#) .

This function will return true upon success and , that you can use immediately in the next steps below, or reuse later.

Sample code:

```
procedure TfrmMainForm.CreateTempFileClick(Sender: TObject);
begin
  VideoGrabber.VideoFromImages_BitmapSortedBy := fs_NameAsc;
  VideoGrabber.VideoFromImages_SourceDirectory := 'MyImagesFolder';
  VideoGrabber.VideoFromImages_TemporaryFile := 'MyTempFile.dat';
  VideoGrabber.VideoFromImages_CreateSetOfBitmaps;
end;
```

B. Preview the video built from temporary file

b1. Select the temporary file as live video source

Set [VideoSource](#) = vs_VideoFromImages

b2. specify the temporary file to use

e.g.: [VideoFromImages_TemporaryFile](#) = "MyTempFile.dat"

b3. specify if the preview must stop at the end of the image files or repeat from the beginning

e.g. [VideoFromImages_RepeatIndefinitely](#) = False

b4. Choose a frame rate

E.g. set [FrameRate](#) = 15

b5. Start the preview

Invoke [StartPreview](#)

Sample code:

```
procedure TfrmMainForm.startpreviewClick(Sender: TObject);
begin
  VideoGrabber.VideoSource := vs_VideoFromImages;
  VideoGrabber.VideoFromImages_TemporaryFile := 'MyTempFile.dat';
  VideoGrabber.VideoFromImages_RepeatIndefinitely := False;
  VideoGrabber.FrameRate := 10;
  VideoGrabber.StartPreview;
end;
```

C. Record a video clip built from the temporary file to an AVI or ASF file (step C)

c1. Select the temporary file as live video source

Set [VideoSource](#) = vs_VideoFromImages

c2. specify the temporary file to use

e.g.: [VideoFromImages_TemporaryFile](#) = "MyTempFile.dat"

c3. specify if the preview must stop at the end of the image files or repeat from the beginning

e.g. [VideoFromImages_RepeatIndefinitely](#) = False

c4. Choose a frame rate

E.g. set [FrameRate](#) = 15

c5. Choose the [recording method](#). Specify if you wish to use a [video and/or audio compressor](#)

c5. Start the recording

Invoke [StartRecording](#)

Sample code:

```
procedure TfrmMainForm.startrecordingClick(Sender: TObject);
begin
    VideoGrabber.VideoSource := vs_VideoFromImages;
    VideoGrabber.VideoFromImages_TemporaryFile := 'MyTempFile.dat';
    VideoGrabber.VideoFromImages_RepeatIndefinitely := False;
    VideoGrabber.StoragePath := 'VideoStorageFolder';
    VideoGrabber.CompressionMode := cm_NoCompression;
    VideoGrabber.FrameRate := 10;
    VideoGrabber.StartRecording;
end;
```

See Also

[OnBitmapsLoadingProgress](#) [SendImageToVideoFromBitmaps](#) [SendImageToVideoFromBitmaps2](#)
[VideoFromImages_BitmapSortedBy](#) [VideoFromImages_RepeatIndefinitely](#)
[VideoFromImages_SourceDirectory](#) [VideoFromImages_TemporaryFile](#)

Created with the Standard Edition of HelpNDoc: [Write eBooks for the Kindle](#)

Video processing

Created with the Standard Edition of HelpNDoc: [Free help authoring environment](#)

Zoom

Zoom

[Prev](#)

Zoom

Description

The zoom can be activated by specifying a [ZoomCoeff](#) value greater than 1000 (expressed as per thousand).

E.g.:

1000 means no zoom (default value)

2500 means a zoom of 2.5x

4000 means a zoom of 4.x

It is possible to move the X and Y center positions by specifying a [ZoomXCenter](#) and [ZoomYCenter](#) values greater or lower than 0 (expressed in pixels).

Note: the [frame grabber](#) must be enabled to use this feature.

See Also

Rotation, mirroring, vertical and horizontal flip

Rotation, mirroring, vertical and horizontal flip

[Prev](#)[Next](#)

Rotation ,mirror, vertical and horizontal flip

Description

Rotation ,mirror, vertical and horizontal flip

The [VideoProcessing_Rotation](#) and [VideoProcessing_RotationCustomAngle](#) properties offer any combination of rotation and mirror effects.

Flip

- the [VideoProcessing_FlipVertical](#) property flips the video vertically
- the [VideoProcessing_FlipHorizontal](#) property flips the video horizontally

Preventing the drawing to be rotated

When the rotation is used and [OverlayAfterTransform](#) is enabled (e.g. to rotate the text overlay with the video), any drawing performed from the [OnFrameOverlayUsingDC](#) event will be rotated as well.

To retrieve the initial coordinates and prevent the drawing to be rotated, use [RetrieveInitialXYAfterRotation](#).

See Also

[TVideoDeinterlacing](#) [MP4NeedsReindexing](#) [OnFrameBitmap](#) [RetrieveInitialXYAfterRotation](#) [VideoProcessing_Brightness](#) [VideoProcessing_Contrast](#) [VideoProcessing_Deinterlacing](#) [VideoProcessing_FlipHorizontal](#) [VideoProcessing_FlipVertical](#) [VideoProcessing_GrayScale](#) [VideoProcessing_Hue](#) [VideoProcessing_InvertColors](#) [VideoProcessing_Pixellization](#) [VideoProcessing_Rotation](#) [VideoProcessing_RotationCustomAngle](#) [VideoProcessing_Saturation](#) [TVideoRotation](#)

Cropping and zooming

Cropping and zooming

[Prev](#)[Next](#)

Cropping and zomming

Description

Cropping

It is possible to easily define an area of the video source that will be cropped.

- define a cropping size with [Cropping_Width](#) and [Cropping_Height](#),
- enable the cropping with [Cropping_Enabled](#),
- start the preview, recording or playback,
- now you can move the cropping area (even dynamically during preview, recording or playback) with [Cropping_X](#) and [Cropping_Y](#).

Trackbar bounds

When trackbars are used to move the cropped area over the video window, the maximum position for these

trackbars are returned by the [Cropping_XMax](#) and [Cropping_YMax](#) properties.
The best location to update the trackbar bounds is from the [OnResizeVideo](#) event. E.g.:

```
procedure TForm1.VideoGrabberResizeVideo(Sender: TObject; SourceWidth, SourceHeight: Integer)
begin
    TrackBarX.Min := 0;
    TrackBarX.Position := VideoGrabber.Cropping_X;
    TrackBarX.Max := VideoGrabber.Cropping_XMax;

    TrackBarY.Min := 0;
    TrackBarY.Position := VideoGrabber.Cropping_Y;
    TrackBarY.Max := VideoGrabber.Cropping_YMax;
end;
```

Zooming

It is possible to zoom a cropped area as follows:

- define a cropping size with [Cropping_Width](#) and [Cropping_Height](#),
- enable the cropping with [Cropping_Enabled](#),
- start the preview, recording or playback,
- now it is possible to specify dynamically a zooming coefficient with [Cropping_Zoom](#),
- it is also possible to move the cropping area with [Cropping_X](#) and [Cropping_Y](#).

Note: if the zooming area is near an edge of the video frame, when changing [Cropping_Zoom](#), the video image will "move" to the opposite direction.

To prevent this problem, enable the [Cropping_Outbounds](#) property. This will keep the zooming location fixed against the edges, however in counterpart a black border will appear when [Cropping_X](#) and [Cropping_Y](#) are near the edges.

See Also

[Cropping_Enabled](#) [Cropping_Height](#) [Cropping_Outbounds](#) [Cropping_Width](#) [Cropping_X](#) [Cropping_XMax](#) [Cropping_Y](#) [Cropping_YMax](#) [Cropping_Zoom](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your CHM Help File Capabilities with HelpNDoc](#)

Image adjustments (brightness, contrast, etc)

Image adjustments (brightness, contrast, etc)

[Prev](#)

[Next](#)

VMR9 image adjustments

Description

Video stream image adjustments

When the [frame grabber](#) is enabled, it is possible to adjust the brightness, contrast, hue and saturation of the video stream by using:

[VideoProcessing_Brightness](#)

[VideoProcessing_Contrast](#)

[VideoProcessing_Hue](#)

[VideoProcessing_Saturation](#)

These properties affect the whole video stream, therefore the captured frames and the recorded streams will be affected.

On the other hand, each of these image processings requires CPU when the corresponding property is "in use" (when its value is <> 0).

Greyscale and inverted colors

When the [frame grabber](#) is enabled, it is possible:

- to convert the video stream in greyscale mode by enabling [VideoProcessing_GrayScale](#)
- to invert the colors by enabling [VideoProcessing_InvertColors](#)

Pixellization

The video frames will be pixellized by assigning to [VideoProcessing_Pixellization](#) a value greater than 1.

VMR9 display image adjustments

Unlike the video stream image adjustments above that require CPU, it is possible to adjust the brightness, contrast, hue and saturation at the graphic card level by using the VMR9 video renderer. Of course, this processing affects only the display, and not the captured frames or the recorded streams.

The corresponding functions are:

[GetVMR9ImageAdjustmentBounds](#) -> returns the bounds of a setting (e.g. the brightness), useful e.g. to setup a trackbar

[IsVMR9ImageAdjustmentAvailable](#) -> returns true if a setting (e.g. the brightness) is available

[SetVMR9ImageAdjustmentValue](#) -> sets a setting value (e.g. the brightness).

See Also

[GetVMR9ImageAdjustmentBounds](#) [IsVMR9ImageAdjustmentAvailable](#) [SetVMR9ImageAdjustmentValue](#) [TVMR9ImageAdjustment](#)

Created with the Standard Edition of HelpNDoc: [Easily convert your WinHelp HLP help files to CHM with HelpNDoc's step-by-step guide](#)

Deinterlacing

Deinterlacing

[Prev](#)
[Next](#)

Deinterlacing

Description

DEINTERLACING

A) Deinterlacing through the VMR9 video renderer

If you don't need to capture frames or perform graphics or text overlays, set [FrameGrabber](#) = fg_Disabled and [VideoRenderer](#) = vr_VMR9, then the VMR9 will deinterlace the video automatically, to need to enable anything else.

B) Deinterlacing by enabling the VideoProcessing_Deinterlacing property

The deinterlacing is activated by the [VideoProcessing_Deinterlacing](#) property, that accepts the following values:

di_Disabled : deinterlacing is disabled (embedded)

di_HalfSize : built-in half-size deinterlacing (embedded)

di_FullSize : built-in full size deinterlacing (embedded)

di_DScaler : deinterlacing through the DScaler deinterlacer (third-party filter).

di_AVISynth : deinterlacing through AVISynth (third-party library, AVISynth must be installed)

di_FFDSHOW : deinterlacing through FFDSHOW (third-party library, FFDSHOW must be installed)

di_ThirdPartyDeinterlacer: deinterlacing by using a third-party deinterlacer specified by name to the [ThirdPartyDeinterlacer](#) property

[Deinterlacing through the built-in deinterlacer](#)

di_HalfSize: performs a half-size deinterlacing,
di_FullSize: performs a full size deinterlacing (the [frame grabber](#) must be enabled)

Deinterlacing through FFDSHOW

- FFDSHOW must be installed (see [Video and Audio decoders](#))
- set [VideoProcessing_Deinterlacing](#) = di_FFDSHOW

then start the preview, recording or playback.

Deinterlacing through AVISynth

The deinterlacing through AVISynth produces a very good quality, however it is available only for video clips, not for a video capture device being captured.

- download AVISynth 2.5 or higher (the .exe version) from <http://avisynth.org>
- install AVISynth with its default options
- set [VideoProcessing_Deinterlacing](#) = di_AVISynth

then open the clip a normal way, e.g.:

```
VideoGrabber.PlayerFileName = "...the video clip.avi"
VideoGrabber.OpenPlayer
```

and the video clip will be deinterlaced automatically in the background through this library.

Deinterlacing through a specified deinterlacer filter

Register the deinterlacer filter with regsvr32.exe, then it will appear in the [DirectShowFilters](#) list after restarting the application.

Then invoke:

```
VideoGrabber.ThirdPartyDeinterlacer = "....." // specify the deinterlace filter name as it appears in the
DirectShowFilters list
VideoGrabber.VideoProcessing_Deinterlacing = di_ThirdPartyDeinterlacer
```

Deinterlacing through DScaler

TVideoGrabber supports the DScaler deinterlacer, however we do not recommend it.

You have to use the **Deinterlace120ax.zip** package that can be downloaded here:
http://sourceforge.net/project/showfiles.php?group_id=7420

Simply unzip the package and register the filter by running **regsvr32.exe Deinterlace.ax**

The DScaler deinterlacer must be activated as follows:

- set [VideoProcessing_Deinterlacing](#) = di_DScaler
- invoke [ShowDialog \(dlg_DScaler\)](#) to adjust the deinterlacer settings.
- then invoke [StartPreview](#) to see to changes.

Remark: when using a DV source, if you get either:

- a interlacing that looks like colored snow in 8 bits colors
- frozen video frames (e.g. one each 2 seconds)

use the following settings:

- set DVReduceFrameRate = true
- you may have to check or uncheck the "Is Odd field first" checkbox
- you may have to choose another deinterlacing method than the default one.

Important notes:

1. with DV sources, be sure to always select the "full" video size in the [VideoSizes](#) list, whatever the deinterlacing method used.
2. if applying deinterlacing on the recording, be sure to disable it during playback, otherwise it will be applied 2 times, e.g.:
 - > you start DV preview in full size,
 - > you apply half-size deinterlacing during capture, OK.
 - > therefore the AVI clip contains half size video, OK.
 - > you forget to disable the half-size deinterlacing before opening the clip, the half-size deinterlacing is applied a 2nd time -> WRONG
 - > the AVI clip appears as quarter-size

Created with the Standard Edition of HelpNDoc: [Transform Your Word Document into a Professional eBook with HelpNDoc](#)

Motion detection

Created with the Standard Edition of HelpNDoc: [Free Kindle producer](#)

Motion detection

| | | |
|-------------------------|----------------------|----------------------|
| Motion detection | Prev | Next |
|-------------------------|----------------------|----------------------|

Motion detection.

Description

Motion detection overview

The motion detection is enabled when [MotionDetector.Enabled](#) is true. The [frame grabber](#) must be enabled to use this feature.

When motion detection is enabled, TVideoGrabber compares the last video frame received to the previous one, unless a reference sample has been specified with [MotionDetector.UseThisReferenceSample](#)

For each video frame, if motion is detected, the [OnMotionDetected](#) event occurs, that returns a global motion ratio, depending of the number of cells in which motion has been detected, and the level of motion in each cell.

If no motion is detected, the [OnMotionNotDetected](#) event occurs.

Areas in which motion can be detected

To let you limit the areas of the video frames in which motion will be detected, the video frame is divided in a **grid** composed of lines and rows, that defines cell having a motion level.

In each cell, the motion level can be set in a range from 1 (minimal motion sensitivity) to 9 (maximal motion sensitivity).

A value of 0 disables motion detection in the cell.

This is explained in the [Grid structure / grid sensitivity](#) chapter.

Triggered motion detection

By default, the motion detection occurs for each video frame.

If you want to trigger the motion detection manually, enable the [MotionDetector.Triggered](#) property.

Then, invoke [MotionDetector.TriggerNow](#) during preview, recording or playback to trigger the motion detection one time, until the next [MotionDetector.TriggerNow](#) call.

Detection of the video signal

Is it possible to detect if the video signal is present or not for each video frame, by invoking [IsVideoSignalDetected](#).

See Also

[Color / Greyscale Grid structure / grid sensitivity](#) [Motion ratio](#) [Recording only when motion is detected](#) [Video noise](#) [TOnMotionDetected](#) [TOnMotionNotDetected](#) [MotionDetector.CellMotionRatio](#)

[MotionDetector_CompareBlue](#)
[MotionDetector_CompareGreen](#)
[MotionDetector_CompareRed](#)
[MotionDetector_Enabled](#)
[MotionDetector_EnumGridDialogControls](#)
[MotionDetector_Get2DTextGrid](#)
[MotionDetector_Get2DTextMotion](#)
[MotionDetector_GetCellLocation](#)
[MotionDetector_GetCellSensitivity](#)
[MotionDetector_GetCellSize](#)
[MotionDetector_GloballyIncOrDecSensitivity](#)
[MotionDetector_GlobalMotionRatio](#)
[MotionDetector_GreyScale](#)
[MotionDetector_Grid](#)
[MotionDetector_GridXCount](#)
[MotionDetector_GridYCount](#)
[MotionDetector_IsGridValid](#)
[MotionDetector_MaxDetectionsPerSecond](#)
[MotionDetector_ReduceCPULoad](#)
[MotionDetector_ReduceVideoNoise](#)
[MotionDetector_Reset](#)
[MotionDetector_ResetGlobalSensitivity](#)
[MotionDetector_SetCellSensitivity](#)
[MotionDetector_SetGridSize](#)
[MotionDetector_ShowGridDialog](#)
[MotionDetector_Triggered](#)
[MotionDetector_UseThisReferenceSample](#)
[OnBacktimedFramesCountReached](#)
[OnMotionDetected](#)
[OnMotionNotDetected](#)
[RecordingOnMotion_Enabled](#)
[RecordingOnMotion_MotionThreshold](#)
[RecordingOnMotion_NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: Create HTML Help, DOC, PDF and print manuals from 1 single source

Motion ratio

Motion ratio

[Prev](#)
[Next](#)

Motion detection and motion ratio.

Description

Motion ratio

The motion ratio is the result of motion detection between 2 video frames. All motion ratios values are within the 0..1 range.

For each video frame received:

a) if motion is detected:

The [OnMotionDetected](#) event occurs. This event returns the and returns a global motion ratio. From this event you can also query cells using [CellMotionRatio \(x, y\)](#) to get their individual motion ratio.

b) if no motion is detected:

The [OnMotionNotDetected](#) event occurs.

See Also

[Color / Greyscale Grid structure / grid sensitivity](#)
[Recording only when motion is detected](#)
[Video noise](#)
[TOnMotionDetected](#)
[TOnMotionNotDetected](#)
[MotionDetector_CellMotionRatio](#)
[MotionDetector_CompareBlue](#)
[MotionDetector_CompareGreen](#)
[MotionDetector_CompareRed](#)
[MotionDetector_Enabled](#)
[MotionDetector_EnumGridDialogControls](#)
[MotionDetector_Get2DTextGrid](#)
[MotionDetector_Get2DTextMotion](#)
[MotionDetector_GetCellLocation](#)
[MotionDetector_GetCellSensitivity](#)
[MotionDetector_GetCellSize](#)
[MotionDetector_GloballyIncOrDecSensitivity](#)
[MotionDetector_GlobalMotionRatio](#)
[MotionDetector_GreyScale](#)
[MotionDetector_Grid](#)
[MotionDetector_GridXCount](#)
[MotionDetector_GridYCount](#)
[MotionDetector_IsGridValid](#)
[MotionDetector_MaxDetectionsPerSecond](#)
[MotionDetector_ReduceCPULoad](#)
[MotionDetector_ReduceVideoNoise](#)
[MotionDetector_Reset](#)
[MotionDetector_ResetGlobalSensitivity](#)
[MotionDetector_SetCellSensitivity](#)
[MotionDetector_SetGridSize](#)
[MotionDetector_ShowGridDialog](#)
[MotionDetector_Triggered](#)
[MotionDetector_UseThisReferenceSample](#)
[OnBacktimedFramesCountReached](#)
[OnMotionDetected](#)
[OnMotionNotDetected](#)
[RecordingOnMotion_Enabled](#)
[RecordingOnMotion_MotionThreshold](#)
[RecordingOnMotion_NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: Free help authoring environment

Color intensity

Color intensity

[Prev](#)
[Next](#)

Color intensity

Description

Color intensity of a cell

It is possible to retrieve the average color intensity of a cell by invoking [MotionDetector_CellColorIntensity \(RGBSelector, x, y\)](#).

The function returns the average intensity for the specified color in the specified cell.

The value returned is in the [0.255] range.

Global intensity of a color

The [MotionDetector_GlobalColorIntensity \(RGBSelector\)](#) function returns the global average intensity for the specified color, in the [0.255] range.

Created with the Standard Edition of HelpNDoc: [Easily create PDF Help documents](#)

Grid structure / grid sensitivity

Grid structure / grid sensitivity

[Prev](#)

[Next](#)

Grid structure.

Description

Overview

A sensitivity grid is applied on video frames. This grid allows to choose areas within the image where motion must be detected. The grid divides the image into rows and columns, defining cells. Each cell sensitivity can be set from 0 (no motion detection) to 9 (maximal motion sensitivity).

E.g. if the video frame is 320 x 240, and you specify a grid size of 32x24, each cell will have a 10x10 size.

One motion ratio is calculated per cell, and the [OnMotionDetected](#) event reports the X and Y coordinates of the cell that reported the higher motion ratio.

Common grids sizes span 2 x 2 to 30 x 30, but any grid size can be used, from 1 x 1 to the width x height size of the video frame.

The grid is defined by a simple string that can be set and retrieved by using the [Grid](#) property, allowing to easily save it and retrieve it using the registry (see Grid Structure below).

Grid setup

The sensitivity of each cell can be set programmatically by assigning the [Grid](#) property, or interactively through a dialog invoked by [ShowGridDialog](#). This dialog shows the grid superposed to the current video image, allowing to enable/disable the motion detection in each cell by clicking on the cell.

Grid structure

The grid corresponds to a division of the frame into rows and columns, equally spaced. The minimum size is 1 x 1, the maximum size is the size of the frame.

The grid divides the video frame in cells. A value associated to each cell represents the sensitivity of this cell, from 0 (no motion sensitivity) to 9 (maximal motion sensitivity).

Grids samples:

2 x 2 grid:

```
0 6
6 6
```

10 x 8 grid:

```
0 0 0 0 0 0 0 0 0 0
7 7 7 7 7 7 7 7 7 7
7 7 7 7 7 7 7 7 7 7
7 7 7 7 7 7 7 7 7 7
7 7 7 7 7 7 7 7 7 7
7 7 7 7 7 7 7 7 7 7
7 7 7 7 7 7 7 7 7 7
```


7 7 7 7 7 7 7 7 7 7

4 x 8 grid:

```
8 8 0 0
8 8 0 0
8 8 0 0
8 8 0 0
8 8 0 0
8 8 0 0
8 8 0 0
8 8 0 0
8 8 0 0
```

If the grid over sizes the frame (the grid has more columns than the frame width and/or more rows than the frame height, in pixels), the motion detection is done on the top/left part of the grid that covers the frame, the parts that oversize the frame are ignored, and the OnGridOversizesFrame event occurs returning the delta width and/or height in pixels that do not cover the frame.

Grid layout

The grid is set and retrieved as a linear string layout of each cell sensitivity. This layout allows to easily store the grid in the registry and retrieve it as a simple string.

The grid string begins by the 1st row, a blank separator, then the 2nd row, a blank separator, the 3rd, row, and so on. E.g., for a 10 x 10 grid :

```
"0000044441 5555555551 4444444441 5555555551 6666666661 6666666661 6666666661 3333333331
0000000000 0000000000"
```

corresponds to the following 2D grid:

```
0 0 0 0 0 4 4 4 4 1
5 5 5 5 5 5 5 5 1
4 4 4 4 4 4 4 4 1
5 5 5 5 5 5 5 5 1
6 6 6 6 6 6 6 6 1
6 6 6 6 6 6 6 6 1
6 6 6 6 6 6 6 6 1
6 6 6 6 6 6 6 6 1
3 3 3 3 3 3 3 3 1
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
```

In this sample, the 4 top/left cells and the 2 bottom rows will not detect motion, and the most sensitive part of the frame is located on the 5th, 6th and 7th rows.

See Also

[Color / Greyscale Motion ratio Recording only when motion is detected Video noise TOnMotionDetected TOnMotionNotDetected MotionDetector_CellMotionRatio MotionDetector_CompareBlue MotionDetector_CompareGreen MotionDetector_CompareRed MotionDetector_Enabled MotionDetector_EnumGridDialogControls MotionDetector_Get2DTextGrid MotionDetector_Get2DTextMotion MotionDetector_GetCellLocation MotionDetector_GetCellSensitivity MotionDetector_GetCellSize MotionDetector_GloballyIncOrDecSensitivity MotionDetector_GlobalMotionRatio MotionDetector_GreyScale MotionDetector_Grid MotionDetector_GridXCount MotionDetector_GridYCount MotionDetector_IsGridValid MotionDetector_MaxDetectionsPerSecond MotionDetector_ReduceCPULoad MotionDetector_ReduceVideoNoise MotionDetector_Reset MotionDetector_ResetGlobalSensitivity MotionDetector_SetCellSensitivity MotionDetector_SetGridSize MotionDetector_ShowGridDialog MotionDetector_Triggered MotionDetector_UseThisReferenceSample OnBacktimedFramesCountReached OnMotionDetected OnMotionNotDetected RecordingOnMotion_Enabled RecordingOnMotion_MotionThreshold RecordingOnMotion_NoMotionPauseDelayMs](#)

Color / Greyscale

Color / Greyscale

[Prev](#)[Next](#)

Color / greyscale.

Description

Color sensitivity

The motion detection is performed by default on a RGB basis.

Any of the 3 RGB colors can be enabled or disabled for motion detection by using [CompareBlue](#) , [CompareRed](#) or [CompareGreen](#) .

Greyscale comparison

To perform motion detection on a greyscale basis, enable the [GreyScale](#) property. In this case CompareBlue, CompareGreen and CompareRed are ignored.

See Also

[Grid structure / grid sensitivity](#) [Motion ratio](#) [Recording only when motion is detected](#) [Video noise](#) [TOnMotionDetected](#) [TOnMotionNotDetected](#) [MotionDetector](#) [CellMotionRatio](#) [MotionDetector](#) [CompareBlue](#) [MotionDetector](#) [CompareGreen](#) [MotionDetector](#) [CompareRed](#) [MotionDetector](#) [Enabled](#) [MotionDetector](#) [EnumGridDialogControls](#) [MotionDetector](#) [Get2DTextGrid](#) [MotionDetector](#) [Get2DTextMotion](#) [MotionDetector](#) [GetCellLocation](#) [MotionDetector](#) [GetCellSensitivity](#) [MotionDetector](#) [GetCellSize](#) [MotionDetector](#) [GloballyIncOrDecSensitivity](#) [MotionDetector](#) [GlobalMotionRatio](#) [MotionDetector](#) [GreyScale](#) [MotionDetector](#) [Grid](#) [MotionDetector](#) [GridXCount](#) [MotionDetector](#) [GridYCount](#) [MotionDetector](#) [IsGridValid](#) [MotionDetector](#) [MaxDetectionsPerSecond](#) [MotionDetector](#) [ReduceCPULoad](#) [MotionDetector](#) [ReduceVideoNoise](#) [MotionDetector](#) [Reset](#) [MotionDetector](#) [ResetGlobalSensitivity](#) [MotionDetector](#) [SetCellSensitivity](#) [MotionDetector](#) [SetGridSize](#) [MotionDetector](#) [ShowGridDialog](#) [MotionDetector](#) [Triggered](#) [MotionDetector](#) [UseThisReferenceSample](#) [OnBacktimedFramesCountReached](#) [OnMotionDetected](#) [OnMotionNotDetected](#) [RecordingOnMotion](#) [Enabled](#) [RecordingOnMotion](#) [MotionThreshold](#) [RecordingOnMotion](#) [NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: Bring your WinHelp HLP help files into the present with HelpNDoc's easy CHM conversion

Video noise

Video noise

[Prev](#)[Next](#)

Video noise.

Description

Video noise

When the camera captures video frames in a dark environment, it is possible that the electrical background noise of the CCD video cell creates fake motion detection events.

It is possible to reduce the sensitivity of to the video noise by enabling the [ReduceVideoNoise](#) property.

Usually this saves about one point of [grid sensitivity](#) (e.g. if motion detection occurred with a grid sensitivity starting from 9, when enabling this property it occurs only with a grid sensitivity starting from 8).

See Also

[Color / Greyscale](#) [Grid structure / grid sensitivity](#) [Motion ratio](#) [Recording only when motion is detected](#) [TOnMotionDetected](#) [TOnMotionNotDetected](#) [MotionDetector](#) [CellMotionRatio](#) [MotionDetector](#) [CompareBlue](#) [MotionDetector](#) [CompareGreen](#) [MotionDetector](#) [CompareRed](#) [MotionDetector](#) [Enabled](#) [MotionDetector](#) [EnumGridDialogControls](#) [MotionDetector](#) [Get2DTextGrid](#) [MotionDetector](#) [Get2DTextMotion](#) [MotionDetector](#) [GetCellLocation](#) [MotionDetector](#) [GetCellSensitivity](#) [MotionDetector](#) [GetCellSize](#) [MotionDetector](#) [GloballyIncOrDecSensitivity](#) [MotionDetector](#) [GlobalMotionRatio](#) [MotionDetector](#) [GreyScale](#) [MotionDetector](#) [Grid](#) [MotionDetector](#) [GridXCount](#) [MotionDetector](#) [GridYCount](#) [MotionDetector](#) [IsGridValid](#) [MotionDetector](#) [MaxDetectionsPerSecond](#) [MotionDetector](#) [ReduceCPULoad](#) [MotionDetector](#) [ReduceVideoNoise](#) [MotionDetector](#) [Reset](#) [MotionDetector](#) [ResetGlobalSensitivity](#)

[MotionDetector_SetCellSensitivity](#) [MotionDetector_SetGridSize](#) [MotionDetector_ShowGridDialog](#)
[MotionDetector_Triggered](#) [MotionDetector_UseThisReferenceSample](#) [OnBacktimedFramesCountReached](#)
[OnMotionDetected](#) [OnMotionNotDetected](#) [RecordingOnMotion_Enabled](#)
[RecordingOnMotion_MotionThreshold](#) [RecordingOnMotion_NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: [Full-featured EBook editor](#)

Recording only when motion is detected

Recording only when motion is detected

[Prev](#)

[Next](#)

How to record only when motion is detected.

Description

To start recording only when motion is detected:

- enable [MotionDetector_Enabled](#)
- enable [RecordingOnMotion_Enabled](#)
- adjust [RecordingOnMotion_MotionThreshold](#) and [RecordingOnMotion_NoMotionPauseDelayMs](#) if required,
- disable [RecordingPauseCreatesNewFile](#) to store all the recording in a single AVI file, or enable [RecordingPauseCreatesNewFile](#) to create a new AVI file each time no motion is detected,
- then invoke [StartRecording](#).

[RecordingOnMotion_Enabled](#): specifies to record only when motion is detected

[RecordingOnMotion_MotionThreshold](#): specifies the minimum motion ratio to (re)start the recording

[RecordingOnMotion_NoMotionPauseDelayMs](#): specifies the delay after which the recording is switched back in a paused state when no motion is detected

E.g.:

```
VideoGrabber1->MotionDetector_Enabled = true;
VideoGrabber1->RecordingCanPause = true;
VideoGrabber1->RecordingOnMotion_Enabled = true;
VideoGrabber1->RecordingOnMotion_NoMotionPauseDelayMs = 2000;
VideoGrabber1->RecordingOnMotion_MotionThreshold = 0.001;
VideoGrabber1->RecordingPauseCreatesNewFile := true;
VideoGrabber1->StartRecording();
```

See Also

[Color / Greyscale Grid structure / grid sensitivity](#) [Motion ratio](#) [Video noise](#) [TOnMotionDetected](#)
[TOnMotionNotDetected](#) [MotionDetector_CellMotionRatio](#) [MotionDetector_CompareBlue](#)
[MotionDetector_CompareGreen](#) [MotionDetector_CompareRed](#) [MotionDetector_Enabled](#)
[MotionDetector_EnumGridDialogControls](#) [MotionDetector_Get2DTextGrid](#) [MotionDetector_Get2DTextMotion](#)
[MotionDetector_GetCellLocation](#) [MotionDetector_GetCellSensitivity](#) [MotionDetector_GetCellSize](#)
[MotionDetector_GloballyIncOrDecSensitivity](#) [MotionDetector_GlobalMotionRatio](#) [MotionDetector_GreyScale](#)
[MotionDetector_Grid](#) [MotionDetector_GridXCount](#) [MotionDetector_GridYCount](#) [MotionDetector_IsGridValid](#)
[MotionDetector_MaxDetectionsPerSecond](#) [MotionDetector_ReduceCPULoad](#)
[MotionDetector_ReduceVideoNoise](#) [MotionDetector_Reset](#) [MotionDetector_ResetGlobalSensitivity](#)
[MotionDetector_SetCellSensitivity](#) [MotionDetector_SetGridSize](#) [MotionDetector_ShowGridDialog](#)
[MotionDetector_Triggered](#) [MotionDetector_UseThisReferenceSample](#) [OnBacktimedFramesCountReached](#)
[OnMotionDetected](#) [OnMotionNotDetected](#) [RecordingOnMotion_Enabled](#)
[RecordingOnMotion_MotionThreshold](#) [RecordingOnMotion_NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: [Full-featured multi-format Help generator](#)

Mixing several video sources

Created with the Standard Edition of HelpNDoc: [Transform your help documentation into a stunning website](#)

Mixing several video sources into a single one

Mixing several video sources into a single one

[Prev](#)
[Next](#)

Mixing several video sources into a single one

Description

Overview

It is possible to mix one or several TVideoGrabber components **used as "normal" video sources** (video capture devices and/or video clips) into a single TVideoGrabber component used as **mixer**.

This mixer component works independently of the source component, it can be stopped, previewed, recorded, paused, etc... while the 1st component sends it the video frames continuously.

The sources components may be displayed into the mixer component:

- as a basic "copy": a 2nd component in mixer mode receives and display the video from the 1st component
- by switching several sources into a single one when needed
- as a mosaic layout (e.g. 4 cameras displayed at the same time on a 2x2 layout),
- as an alternated display (e.g. the 4 cameras are displayed alternatively one after the other),
- in a mosaic/alternated layout (e.g. the 16 cameras are displayed in 4 alternated mosaic layouts of 2x2 cameras).

Combined mosaic / alternated mixing

It is possible to combine the alternated and mosaic modes, e.g. to display 16 cameras by groups of 4 cameras displayed alternatively into a 4 X 4 video window.

Parameters of the Mixer_AddToMixer component

- the 1st parameter is the Uniqueld of the source component
- the 2nd parameter is not used for the moment, just set it to 0
- the 3rd parameter is the mosaic line where the source will be displayed (set it to 0 for an "alternated only" use).
- the 4th parameter is the mosaic column where the source will be displayed (set it to 0 for an "alternated only" use).
- the 5th parameter is the display group number (set it to 0 for a "mosaic only" use)
- the 6th parameter is the display group duration in milliseconds (set it to 0 for a "mosaic only" use)
- the 7th parameter should be set to TRUE
- the 8th parameter should be set to TRUE

See [Mixer_AddToMixer](#).

Basic mixing (a 2nd component receives the video frames from the 1st component)

This mode lets you have a 2nd component that uses the first component as video source.
E.g. the first component makes the preview and the 2nd component starts/stops/pause/resume the recording independently of the 1st component that does a continuous preview.

E.g.:

```
VideoGrabber1.VideoSource = vs_VideoCaptureDevice
```

```
VideoGrabber1.StartPreview()
VideoGrabber2.VideoSource = vs_Mixer
VideoGrabber2.Mixer_AddToMixer (VideoGrabber1.UniqueID, 0, 0, 0, 0, 0, true, true);
VideoGrabber2.StartPreview()
```

and VideoGrabber2 will receive as video source the video displayed and sent by VideoGrabber1.

Choosing the mixer video size

By default, the mixer uses the following video size:

- if the source has been started before starting the mixer, the mixer uses the source size,
- if the source has not been started, the mixer starts in 320x240.

To choose the video size, invoke UseNearestVideoSize on the mixer before StartPreview or StartRecording.

E.g. to start the mixer in 640x480, invoke:

```
VideoGrabberMixer.VideoSource = vs_Mixer
VideoGrabberMixer.UseNearestVideoSize (640, 480, true)
VideoGrabberMixer.StartPreview()
```

Switching several sources into a single one when needed

Similar to the basic mixing with more than one video source, the sources can be switched by invoking [Mixer Activation](#) , e.g:

1. start the preview of the 1st capture device:

```
VideoGrabber1.VideoSource = vs_VideoCaptureDevice
VideoGrabber1.VideoDevice = 0
VideoGrabber1.StartPreview()
```

2. start the preview of the 2nd capture device:

```
VideoGrabber2.VideoSource = vs_VideoCaptureDevice
VideoGrabber2.VideoDevice = 1
VideoGrabber2.StartPreview()
```

3. start the 3rd component that will make the preview or recording in mixer mode:

```
VideoGrabber3.VideoSource = vs_Mixer
int MixerId1 = VideoGrabber3.Mixer_AddToMixer (VideoGrabber1.UniqueID, 0, 0, 0, 0, 0, true, true)
int MixerId2 = VideoGrabber3.Mixer_AddToMixer (VideoGrabber2.UniqueID, 0, 0, 0, 0, 0, true, true)
VideoGrabber3.Mixer_Activation (MixerId2, false) // let' start with MixerId1 activated
VideoGrabber3.StartPreview()
```

4. then to switch between the inputs, activate one and deactivate the other:

```
VideoGrabber3.Mixer_Activation (MixerId1, false)
VideoGrabber3.Mixer_Activation (MixerId2, true)
or
VideoGrabber3.Mixer_Activation (MixerId2, false)
VideoGrabber3.Mixer_Activation (MixerId1, true)
```

Activating the mixer component in automatic alternated mixing mode

Let's take an example where the mixer component is named "Mixer1", and the sources components "Source1", "Source2" and "Source3".

- set Mixer1.[VideoSource](#) = vs_Mixer

- set Mixer1.[Display_AutoSize](#) = false (if you want to control the Width and Height of the component and prevent it to be resized automatically)
- set Mixer1.[Mixer_MosaicLines](#) = 1
- set Mixer1.[Mixer_MosaicColumns](#) = 1
- invoke [Mixer_AddToMixer](#) (Source component, 0, 0, 0, **group number**, **group display duration**, true, true).

E.g if (group 1 = 1500 ms, group 2 = 2000 ms, group 3 = 2500 ms)

```
Mixer1.Mixer_AddToMixer (Source1.UniqueID, 0, 0, 0, 1, 1500, True, True)
Mixer1.Mixer_AddToMixer (Source2.UniqueID, 0, 0, 0, 2, 2000, True, True)
Mixer1.Mixer_AddToMixer (Source3.UniqueID, 0, 0, 0, 3, 2500, True, True)
```

then invoke e.g.:

```
Source1.StartPreview()
Source2.StartPreview()
Mixer1.StartPreview()
```

See the remark below.

Activating the mixer component in mosaic mixing mode

In this mode the destination component window is splitted into x lines and y columns, and each source is displayed at a predefined (x,y) location.

Let's take an example where the mixer component is named "Mixer1", and the sources components "Source1", "Source2", "Source3" and "Source4" will be displayed in a 2 x 2 layout.

- set Mixer1.[VideoSource](#) = vs_Mixer
- set Mixer1.[Display_AutoSize](#) = false (if you want to control the Width and Height of the component and prevent it to be resized automatically)
- set Mixer1.[Mixer_MosaicLines](#) = 2
- set Mixer1.[Mixer_MosaicColumns](#) = 2
- invoke [Mixer_AddToMixer](#) (Source component, 0, **line**, **column**, 0, 0, true, true).

```
Mixer1.Mixer_AddToMixer (Source1.UniqueID, 0, 1, 1, 0, 0, True, True)
Mixer1.Mixer_AddToMixer (Source2.UniqueID, 0, 1, 2, 0, 0, True, True)
Mixer1.Mixer_AddToMixer (Source3.UniqueID, 0, 2, 1, 0, 0, True, True)
Mixer1.Mixer_AddToMixer (Source4.UniqueID, 0, 2, 2, 0, 0, True, True)
```

then invoke e.g.:

```
Source1.StartPreview()
Source2.StartPreview()
Source3.StartPreview()
Source4.StartPreview()
Mixer1.StartPreview()
```

See the remark below.

Activating the mixer component in alternated/mosaic mixing mode

In this mode each source is displayed alternatively into a single video window.

Let's take an example where the mixer component is named "Mixer1", and the sources components "Source1", "Source2", "Source3" and "Source4" will be displayed alternatively in 2 layouts of 1 x 2 source components.

We will use 2 groups named "55" and "66" (the number is not significant, it must be simply different for

each group). The group 55 will be displayed with a duration of 1500 milliseconds, and the group 66 with a duration of 2500 milliseconds.

- set Mixer1.[VideoSource](#) = vs_Mixer
- set Mixer1.[Display_AutoSize](#) = false (if you want to control the Width and Height of the component and prevent it to be resized automatically)
- set Mixer1.[Mixer_MosaicLines](#) = 1
- set Mixer1.[Mixer_MosaicColumns](#) = 2
- invoke [Mixer_AddToMixer](#) (Source component, 0, line, column, group number, group duration, true, true).

```
Mixer1.Mixer_AddToMixer (Source1.UniqueID, 0, 1, 1, 55, 1500, True, True)
Mixer1.Mixer_AddToMixer (Source2.UniqueID, 0, 1, 2, 55, 1500, True, True)
Mixer1.Mixer_AddToMixer (Source3.UniqueID, 0, 1, 1, 66, 2500, True, True)
Mixer1.Mixer_AddToMixer (Source4.UniqueID, 0, 1, 2, 66, 2500, True, True)
```

then invoke e.g.:

```
Source1.StartPreview()
Source2.StartPreview()
Source3.StartPreview()
Source4.StartPreview()
Mixer1.StartPreview()
```

Remark: you can also invoke Mixer1.[StartPreview](#) before starting or associating the sources, in this case an empty window with a [BackgroundColor](#) will be displayed.

See Also

[Mixer_Activation](#) [Mixer_AddToMixer](#) [Mixer_MosaicColumns](#) [Mixer_MosaicLines](#) [Mixer_RemoveFromMixer](#) [Mixer_SetOverlayRoundedCorner](#) [Mixer_SetupPIPFromSource](#)

Created with the Standard Edition of HelpNDoc: [Upgrade your help files and your workflow with HelpNDoc's WinHelp HLP to CHM conversion](#)

Reencoding

Created with the Standard Edition of HelpNDoc: [Effortlessly Create High-Quality Documentation with a Help Authoring Tool](#)

Reencoding of clips in batch mode

Reencoding of clips in batch mode

[Prev](#)

[Next](#)

How to reencode video, audio/video or audio clips.

Description

How to reencode video audio/video or audio clips

TVideoGrabber lets you reencode clips:

- 1) you can simply cut video clips by specifying a start and stop time,
- 2) you can reencode a clip:
 - by specifying a start and or stop time,
 - by using the current audio and/or video compressor,
 - by applying any of the frame grabber features (text overlay, graphics overlay, video rotation, deinterlacing, etc...).

Reencoding properties

[Reencoding_SourceVideoClip](#): file name of the source video clip
[Reencoding_NewVideoClip](#): file name of the video clip to create
[Reencoding_StartTime](#): start time expressed in 100ns units, (default -1 = beginning),
[Reencoding_StartFrame](#): start frame (default -1 = beginning)
[Reencoding_StopTime](#): stop time expressed in 100ns units, (default -1 = end),
[Reencoding_StopFrame](#): stop frame (default -1 = end of the clip)
[Reencoding_IncludeAudioStream](#): if enabled, the audio stream is included in the new video clip,
[Reencoding_IncludeVideoStream](#): if enabled, the video stream is included in the new video clip,
[Reencoding_Method](#): rm_AVI to record in AVI format, or rm_ASF to record in ASF format,
[Reencoding_UseAudioCompressor](#): if enabled, the current [audio compressor](#) is used,
[Reencoding_UseVideoCompressor](#): if enabled, the current [video compressor](#) is used,
[Reencoding_UseFrameGrabber](#): if enabled, graphics, text overlays, cropping and rotation will be applied,
[Reencoding_WMVOutput](#): the clip will be created as .WMV

About the start/stop frames and times

- the default -1 value specifies "start from the beginning of the clip" or "stop when the end of the clip is reached".
- the times are specified in 100ns units, e.g. 3 seconds = 3000000
- if you specify a [Reencoding_StartTime](#) AND a [Reencoding_StartFrame](#), the [Reencoding_StartTime](#) will be ignored.
- if you specify a [Reencoding_StopTime](#) AND a [Reencoding_StopFrame](#), the [Reencoding_StopTime](#) will be ignored.

To start reencoding:

To start reencoding a video clip:

- set the "Reencoding_..." properties,
- invoke [StartReencoding](#)

Important remark:

when invoking [StartReencoding](#), the process starts the reencoding and **returns immediatly, it does not wait for the reencoding process to complete.**

If you are creating the component programmatically, be sure to wait for the OnReencodingCompleted before destroying the component, otherwise the reencoding process would be interrupted before finished.

E.g.:

```

VideoGrabber1.Reencoding_SourceVideoClip = "MyVideoClipToReencode.avi"
VideoGrabber1.Reencoding_NewVideoClip = "MyReencodedVideoClip.wmv"
VideoGrabber1.Reencoding_WMVOutput = true // output clip is wmv
VideoGrabber1.Reencoding_Method = rm_ASF
VideoGrabber1.Reencoding_StartFrame = -1 // -1 = beginning of the clip. E.g. if you se
VideoGrabber1.Reencoding_StopFrame = -1 // -1 = beginning of the clip. E.g. if you set
VideoGrabber1.Reencoding_StartTime = -1 // -1 = end of the clip. E.g. if you set 50000
VideoGrabber1.Reencoding_StopTime = -1 // -1 = end of the clip. E.g. if you set 100000
VideoGrabber1.Reencoding_IncludeAudioStream = true // if audio stream needed in the re
VideoGrabber1.Reencoding_IncludeVideoStream = true // if video stream needed in the re
VideoGrabber1.Reencoding_UseAudioCompressor = false
VideoGrabber1.Reencoding_UseVideoCompressor = false
VideoGrabber1.Reencoding_UseFrameGrabber = true
VideoGrabber1.StartReencoding // from now the progress will be returned periodically b
    
```

To stop the reencoding process before it ends, simply invoke [StopReencoding](#).

Reencoding progress

When the reencoding begins, the [OnReencodingStarted](#) event occurs.

During the reencoding, the [OnReencodingProgress](#) event occurs periodically, reporting the percentage of completion.

When the reencoding ends, the [OnReencodingCompleted](#) event occurs.

Reencoding an audio clip / Extracting the audio from a video clip

It is possible to reencode an audio clip, or to extract only the audio to a .WAV or .MP3 audio clip, just by specifying ".wav" or ".mp3" as file extension for the new clip.

To extract in MP3, first download and register (with regsvr32.exe) the LAME Audio Encoder (freeware) that you can download [here](#) (the direct download link is [here](#))

Sample code to extract 4 seconds of a WAV audio clip starting at 2 seconds (therefore stop time = 2s + 4s) and save them in MP3:

```
VideoGrabber.Reencoding_SourceVideoClip = "c:/myfolder/myvideoclip.wav"
VideoGrabber.Reencoding_Start_Frame = -1
VideoGrabber.Reencoding_Stop_Frame = -1
VideoGrabber.Reencoding_Start_Time = 20000000
VideoGrabber.Reencoding_Stop_Time = 60000000
VideoGrabber.Reencoding_NewVideoClip = "c:/myfolder/mynewclip.mp3"
VideoGrabber.StartReencoding()
```

Sample code to exact the whole audio from a video clip to a .WAV format:

```
VideoGrabber.Reencoding_SourceVideoClip = "c:/myfolder/myvideoclip.avi"
VideoGrabber.Reencoding_Start_Frame = -1
VideoGrabber.Reencoding_Stop_Frame = -1
VideoGrabber.Reencoding_Start_Time = -1
VideoGrabber.Reencoding_Stop_Time = -1
VideoGrabber.Reencoding_NewVideoClip = "c:/myfolder/mynewclip.wav"
VideoGrabber.StartReencoding()
```

Sample code to exact the first 3 seconds of a video clip to a .MP3 format:

```
VideoGrabber.Reencoding_SourceVideoClip = "c:/myfolder/myvideoclip.avi"
VideoGrabber.Reencoding_Start_Frame = -1
VideoGrabber.Reencoding_Stop_Frame = -1
VideoGrabber.Reencoding_Start_Time = -1
VideoGrabber.Reencoding_Stop_Time = 30000000
VideoGrabber.Reencoding_NewVideoClip = "c:/myfolder/mynewclip.mp3"
VideoGrabber.StartReencoding()
```

Remark:

when extracting audio, only the following Reencoding... settings are taken in account:

[Reencoding_StartTime](#)
[Reencoding_StopTime](#)
[Reencoding_StartFrame](#)
[Reencoding_StopFrame](#)

the other Reencoding... settings are ignored.

Converting a DVD to a M2V file

You can reencode a DVD into a M2V format through DGMPGDec as follows:

- download the DGMPGDec executables from <http://neuron2.net/dgmpgdec/dgmpgdec.html>, unzip in the folder of your choice
- use the following reencoding commands:

E.g. let's suppose:

- you have unzipped DGMPGDec into c:/myDGfolder
- your DVD is placed in the "D:" drive

- you want to generate the M2V to c:/myOutputFolder/myOutputVideo.m2v

```
VideoGrabber.Reencoding_SourceVideoClip = "c:/myDGfolder/DGIndex.exe D:/VIDEO_TS"
VideoGrabber.Reencoding_NewVideoClip "c:/myOutputFolder/myOutputVideo.m2v"
VideoGrabber.StartReencoding()
```

Remark about the reencoding progress: the values reported by the [OnReencodingProgress](#) event are not significant, because it is not possible to predict the time required to perform the reencoding with DGIndex.exe. The end of the reencoding will be notified by the [OnReencodingCompleted](#) event.

See Also

[OnReencodingCompleted](#) [OnReencodingProgress](#) [OnReencodingStarted](#) [Reencoding](#) [IncludeAudioStream](#) [Reencoding](#) [IncludeVideoStream](#) [Reencoding](#) [Method](#) [Reencoding](#) [NewVideoClip](#) [Reencoding](#) [SourceVideoClip](#) [Reencoding](#) [StartFrame](#) [Reencoding](#) [StartTime](#) [Reencoding](#) [StopFrame](#) [Reencoding](#) [StopTime](#) [Reencoding](#) [UseAudioCompressor](#) [Reencoding](#) [UseFrameGrabber](#) [Reencoding](#) [UseVideoCompressor](#) [Reencoding](#) [WMVOutput](#) [StartReencoding](#) [StopReencoding](#)

Created with the Standard Edition of HelpNDoc: [Easy to use tool to create HTML Help files and Help web sites](#)

Reencoding of clips in preview or recording mode

Reencoding of clips in preview or recording mode

[Prev](#)

[Next](#)

Video clips reencoding in preview/recording mode

Description

Video clips reencoding in preview/recording mode

The reencoding of video clips can be done as follows:

- set [VideoSource](#) = vs_VideoFileOrURL,
- set [VideoSource FileOrURL](#) with the file name or the URL of the video clip,
- set all the text overlays, graphics overlays, video effects, etc... you need,
- select the compression settings (see the [Software compression using codecs](#) chapter),
- invoke [StartRecording](#) to record to a new video clip, or [StartPreview](#) to just play it.

Note: a start time and/or stop time can be specified by [VideoSource FileOrURL StartTime](#) and [VideoSource FileOrURL StopTime](#).

Reencoding of a playlist in preview/recording mode

You can reencode a playlist as follows:

E.g.:

```
videograbber.Playlist (pl_Clear, '')
videograbber.Playlist (pl_Add, 'vg000004.avi')
videograbber.Playlist (pl_Add, 'vg000002.avi')
videograbber.Playlist (pl_Add, ...
videograbber.Playlist (pl_Add, ...
videograbber.VideoSource = vs_VideoFileOrURL
videograbber.VideoSource_FileOrURL = 'PLAYLIST'
videograbber.StartRecording()
```

You can invoke StartPreview instead of StartRecording to just play the playlist.

See Also

[VideoSource FileOrURL VideoSource FileOrURL StartTime VideoSource FileOrURL StopTime](#)

Merging or splitting video clips

Merging or splitting video clips

[Prev](#)[Next](#)

Merge or split video clips

Description

To merge clips

Clips can be merged by specifying a playlist, then invoking [StartReencoding](#) through the [Playlist](#) feature.

E.g.:

```
videograbber.Playlist (pl_Clear, '')
videograbber.Playlist (pl_Add, 'vg000004.avi')
videograbber.Playlist (pl_Add, 'vg000002.avi')
videograbber.Reencoding_SourceVideoClip = 'PLAYLIST'
videograbber.Reencoding_NewVideoClip = 'mynewclip.avi'
videograbber.StartReencoding()
```

Depending of the kind of clip reencoded, you can also specify a start/stop time:

```
videograbber.Playlist (pl_Clear, '')
videograbber.Playlist (pl_Add, 'vg000004.avi')
VideoGrabber.PlayList (pl_SpecifyPositions, 'vg000004.avi');
VideoGrabber.OpenPlayerAtTimePositions (50000000, 120000000, true, true);
videograbber.Playlist (pl_Add, 'vg000002.avi')
VideoGrabber.PlayList (pl_SpecifyPositions, 'vg000002.avi');
VideoGrabber.OpenPlayerAtTimePositions (50000000, 120000000, true, true);
videograbber.Reencoding_SourceVideoClip = 'PLAYLIST'
videograbber.Reencoding_NewVideoClip = 'mynewclip.avi'
videograbber.StartReencoding()
```

See the [Reencoding](#) chapter for the reencoding parameters (reencoding method, etc...)

To split video clips

To split a clip, use the [Reencoding](#) feature by specifying a start time and/or a stop time (or a start frame and/or stop frame).

Video capture devices with multiplexed inputs

Video capture devices having multiplexed inputs

Video capture devices having multiplexed inputs

[Prev](#)[Next](#)

Video capture cards with multiplexed inputs

Description

Video capture cards with multiplexed inputs

TVideoGrabber supports video capture boards having multiplexed inputs.

This mode is activated by the [MultiplexedRole](#) property. Each input can be enabled/disabled with [EnableMultiplexedInput](#).

2 cases:

a. the video capture card is switching its inputs automatically

If the inputs are switched by the card itself (you see the video frames of each channel alternatively in the same video window), disable the [MultiplexedInputEmulation](#) property.

It is possible that this feature is only available through one of the video inputs. In this case you have to select this [VideoInput](#) in the [VideoInputs](#) list.

b. the video capture card does not switch its inputs automatically

Whatever the [VideoInput](#) that you select in the [VideoInputs](#) list, you see only one video channel at the time. In this case enable the [MultiplexedInputEmulation](#) property to get the inputs switched programmatically by TVideoGrabber.

Each video input will be switched after a delay specified by the [MultiplexedSwitchDelay](#) (default value: 0 means the input is switched after each video frame received).

When an input is switched, the [MultiplexedStabilizationDelay](#) waits for the specified duration (expressed in milliseconds) to wait for the input stabilization (default value = 70 milliseconds, minimum value = 30 ms).

It is possible to work in 2 ways:

1) one single TVideoGrabber component shows a mosaic layout

In this mode a single TVideoGrabber component displays all inputs in a single video window as a "mosaic layout".

This mode is activated by setting:

- [MultiplexedRole](#) = mr_MultiplexedMosaic4 -> for a 4 inputs display
- [MultiplexedRole](#) = mr_MultiplexedMosaic16 -> for a 16 inputs display

Note: the frame rate is specified by the [FrameRate](#) property.

2) master/slaves: one master TVideoGrabber component distributes the inputs to several TVideoGrabber slaves components

- in this video capture board is managed in preview mode by the TVideoGrabber master component
- the slaves TVideoGrabber components are "virtual input" components associated to the TVideoGrabber master component.

To activate this mode for a 4 inputs board:

a)

- put a TVideoGrabber component on the form (e.g. VideoGrabber1)
- set its [MultiplexedRole](#) property to mr_MultiplexedMaster

b)

- put 4 TVideoGrabber components on the form (e.g. VideoGrabber2, VideoGrabber3, VideoGrabber4, VideoGrabber5).
- set their [MultiplexedRole](#) property to mr_MultiplexedSlave.

c)

- invoke programmatically [AssociateMultiplexedSlave](#) on the master component for each slave component. E.g. add the following code to associate the slaves components to the master component: (in the FormShow event, or before invoking StartPreview):

```
procedure TForm1.FormShow(Sender: TObject);
begin
    VideoGrabber1.AssociateMultiplexedSlave (0, VideoGrabber2.UniqueID);
    VideoGrabber1.AssociateMultiplexedSlave (1, VideoGrabber3.UniqueID);
    VideoGrabber1.AssociateMultiplexedSlave (2, VideoGrabber4.UniqueID);
```

```
VideoGrabber1.AssociateMultiplexedSlave (3, VideoGrabber5.UniqueID);
end;
```

Note: the first parameter is the number of the video input to associate to the slave component.

Then you can invoke [StartPreview](#) to start distributing the virtual inputs to the slaves components.

Notes:

- the frame rate of the master component is specified by the [FrameRate](#) property.
- the frame rate of the slaves components can be adjusted by setting their respective [FrameRate](#) property.

See Also

[TMultiplexedRole](#) [AssociateMultiplexedSlave](#) [EnableMultiplexedInput](#) [MultiplexedInputEmulation](#)
[MultiplexedRole](#) [MultiplexedStabilizationDelay](#) [MultiplexedSwitchDelay](#) [UniqueID](#)

Created with the Standard Edition of HelpNDoc: [Free help authoring tool](#)

Opening a clip or an IP URL from a background thread without blocking

Created with the Standard Edition of HelpNDoc: [Streamline your documentation process with HelpNDoc's WinHelp HLP to CHM conversion feature](#)

Opening a clip or an IP URL from a background thread without blocking the main thread

Opening a clip or an IP URL from a background thread without blocking the main thread

[Prev](#)

[Next](#)

Opening a clip or an IP URL from a background thread without blocking the main thread

Description

It is possible to open a clip or an URL without blocking the main thread.

C# and VB.NET:

- define a class derived from VidGrab.VideoGrabberThread
- put the TVideoGrabber initialization code in the VideoGrabberThreadFunc() class
- synchronize the TVideoGrabber event callbacks that refer to .NET controls with the Invoke method

The "AsynchronousStartFromThread" C# or VB demo project included in the package shows the required sample code to open an IP camera URL.

See Also

[Player features](#) [TOnThreadSync](#) [TThreadSyncPoint](#) [EnableThreadMode](#) [OnThreadSync](#)

Created with the Standard Edition of HelpNDoc: [Free iPhone documentation generator](#)

Synchronization of TVideoGrabber components

Created with the Standard Edition of HelpNDoc: [Say Goodbye to Documentation Headaches with a Help Authoring Tool](#)

Synchronization of several TVideoGrabber components

Synchronization of several TVideoGrabber components

[Prev](#)

[Next](#)

Synchronization of several TVideoGrabber components.

Description

SYNCHRONIZATION ACTIVATION

It is possible to synchronize several TVideoGrabber components placed on the same form, or on different forms, within the same application.

This feature is activated by enabling the [Synchronized](#) property on each TVideoGrabber component that must be synchronized with the others.

PREVIEW/RECORDING SYNCHRONIZATION

The preview and recording synchronization lets TVideoGrabber prepare the preview or recording on all the specified components, then the 3 components are started by just invoking [StartSynchronized\(\)](#) on one of them.

E.g. to start the preview concurrently on 3 TVideoGrabber components:

```
VideoGrabber1.VideoDevice = VideoGrabber.VideoDeviceIndex ("... name of your 1st video capture device ...")
VideoGrabber2.VideoDevice = VideoGrabber.VideoDeviceIndex ("... name of your 2nd video capture device ...")
VideoGrabber3.VideoDevice = VideoGrabber.VideoDeviceIndex ("... name of your 3rd video capture device ...")
```

```
VideoGrabber1.Synchronized = true
VideoGrabber2.Synchronized = true
VideoGrabber3.Synchronized = true
```

```
VideoGrabber1.StartPreview()
VideoGrabber2.StartPreview()
VideoGrabber3.StartPreview()
```

```
VideoGrabber1.StartSynchronized()
```

When invoking StartSynchronized on one of the 3 TVideoGrabber components (in this example VideoGrabber1) the preview will start simultaneously on the 3 components.

For the recording, replace StartPreview() by StartRecording()

PLAYER SYNCHRONIZATION

See [Synchronization of several player components chapter](#)

Created with the Standard Edition of HelpNDoc: [Streamline your documentation process with HelpNDoc's HTML5 template](#)

Logo

Created with the Standard Edition of HelpNDoc: [Maximize Your Documentation Capabilities with HelpNDoc's User-Friendly UI](#)

Logo displayed in the video window

Logo displayed in the video window

[Prev](#)
[Next](#)

Logo displayed in the inactive video window.

Description

It is possible to fill out the inactive video window with a logo or a static image.

First of all, setup a logo by using one of the following functions:

[SetLogoFromBMPFile](#) (FileName: string)
[SetLogoFromJPEGFile](#) (FileName: string)
[SetLogoFromHBitmap](#) (Bitmap: HBITMAP)
[SetLogoFromTBitmap](#) (Bitmap: TBitmap)
[SetLogoFromTImage](#) (Image: TImage)

Then set a layout (centered, stretched, boxed, repeated, etc...) with [LogoLayout](#)

Now the logo can be shown or hidden by enabling or disabling [LogoDisplayed](#)

See Also

[LogoDisplayed](#) [LogoLayout](#) [SetLogoFromBMPFile](#) [SetLogoFromHBitmap](#) [SetLogoFromJPEGFile](#)
[SetLogoFromTBitmap](#)

Created with the Standard Edition of HelpNDoc: [Easily create Help documents](#)

WPF

Created with the Standard Edition of HelpNDoc: [Streamline Your Documentation Creation with a Help Authoring Tool](#)

VidGrabWPF:VideoGrabberWPF component

VidGrabWPF:VideoGrabberWPF component

[Prev](#)
[Next](#)

New VidGrabWPF:VideoGrabberWPF component

Description

This VidGrabWPF:VideoGrabberWPF component is a WPF component compatible with the code of the VidGrab::VideoGrabber component of WinForms.

It has a dual mode that can be switched with the VideoGrabberToImage property listed in the "Common" properties of the component:

"VideoGrabberToImage" property disabled (default):

TVideoGrabber renders in a DirectShow renderer through a WindowsFormHost.

This saves CPU, but with the potential "airspace issue" if using WPF semi-transparent overlays

"VideoGrabberToImage" property enabled:

TVideoGrabber renders through an Image component, allowing WPF semi-transparent overlays

The sample code can be found in the "MainDemoWPF" project included in the package under CSharp_VB.NET

This demo is a WPF demo project similar to the Winform MainDemo project.

Created with the Standard Edition of HelpNDoc: [Create HTML Help, DOC, PDF and print manuals from 1 single source](#)

TVideoGrabber

TVideoGrabber

[Prev](#)
[Next](#)

[Properties](#) [Methods](#) [Events](#)

TVideoGrabber component.

Unit

[VidGrab](#)

Description

TVideoGrabber component events, methods and properties.

Created with the Standard Edition of HelpNDoc: [Easy EPub and documentation editor](#)

Properties

TVideoGrabber Properties

[TVideoGrabber](#)

Public

[AdjustOverlayAspectRatio](#)
[AdjustPixelAspectRatio](#)
[Aero](#)
[Align](#)
[Alignment](#)
[AnalogVideoStandard](#)
[AnalogVideoStandards](#)
[AnalogVideoStandardsCount](#)
[Anchors](#)
[ApplicationPriority](#)
[ASFAudioBitRate](#)
[ASFAudioChannels](#)
[ASFBufferWindow](#)
[ASFDeinterlaceMode](#)
[ASFFixedFrameRate](#)
[ASFMediaServerPublishingPoint](#)
[ASFMediaServerRemovePublishingPointAfterDisconnect](#)
[ASFMediaServerTemplatePublishingPoint](#)
[ASFNetworkMaxUsers](#)
[ASFNetworkPort](#)
[ASFProfile](#)
[ASFProfileFromCustomFile](#)
[ASFProfiles](#)
[ASFProfilesCount](#)
[ASFProfileVersion](#)
[ASFVideoBitRate](#)
[ASFVideoFrameRate](#)
[ASFVideoHeight](#)
[ASFVideoMaxKeyFrameSpacing](#)
[ASFVideoQuality](#)
[ASFVideoWidth](#)
[AspectRatioToUse](#)
[AssociateAudioAndVideoDevices](#)
[AudioBalance](#)
[AudioChannelRenderMode](#)
[AudioCompressor](#)
[AudioCompressorName](#)
[AudioCompressors](#)
[AudioCompressorsCount](#)
[AudioDevice](#)
[AudioDeviceName](#)
[AudioDeviceRendering](#)
[AudioDevices](#)
[AudioDevicesCount](#)
[AudioFormat](#)
[AudioFormats](#)

[AudioInput](#)
[AudioInputBalance](#)
[AudioInputLevel](#)
[AudioInputMono](#)
[AudioInputs](#)
[AudioInputsCount](#)
[AudioPeakEvent](#)
[AudioRecording](#)
[AudioRenderer](#)
[AudioRendererName](#)
[AudioRenderers](#)
[AudioRenderersCount](#)
[AudioSource](#)
[AudioStreamNumber](#)
[AudioSyncAdjustment](#)
[AudioSyncAdjustmentEnabled](#)
[AudioVolume](#)
[AutoFileName](#)
[AutoFileNameDateTimeFormat](#)
[AutoFileNameMinDigits](#)
[AutoFilePrefix](#)
[AutoRefreshPreview](#)
[AutoStartPlayer](#)
[AVIDurationUpdated](#)
[AVIFormatOpenDML](#)
[AVIFormatOpenDMLCompatibilityIndex](#)
[BackgroundColor](#)
[BorderStyle](#)
[BurstCount](#)
[BurstInterval](#)
[BurstMode](#)
[BurstType](#)
[Busy](#)
[BusyCursor](#)
[CameraControlSettings](#)
[Caption](#)
[CaptureFileExt](#)
[Color](#)
[ColorKey](#)
[ColorKeyEnabled](#)
[CompressionMode](#)
[CompressionType](#)
[Cropping_Enabled](#)
[Cropping_Height](#)
[Cropping_Outbounds](#)
[Cropping_Width](#)
[Cropping_X](#)
[Cropping_XMax](#)
[Cropping_Y](#)
[Cropping_YMax](#)
[Cropping_Zoom](#)
[CurrentFrameRate](#)
[CurrentState](#)
[DeliveredFrames](#)
[DirectShowFilters](#)
[DirectShowFiltersCount](#)
[Display_Active](#)
[Display_AlphaBlendEnabled](#)
[Display_AlphaBlendValue](#)
[Display_AspectRatio](#)
[Display_AutoSize](#)
[Display_Embedded](#)
[Display_FullScreen](#)

[Display_Height](#)
[Display_Left](#)
[Display_Monitor](#)
[Display_MouseMovesWindow](#)
[Display_PanScanRatio](#)
[Display_StayOnTop](#)
[Display_Top](#)
[Display_TransparentColorEnabled](#)
[Display_TransparentColorValue](#)
[Display_VideoHeight](#)
[Display_VideoPortEnabled](#)
[Display_VideoWidth](#)
[Display_VideoWindowHandle](#)
[Display_Visible](#)
[Display_Width](#)
[DroppedFrames](#)
[DroppedFramesPollingInterval](#)
[DualDisplay_Active](#)
[DualDisplay_AlphaBlendEnabled](#)
[DualDisplay_AlphaBlendValue](#)
[DualDisplay_AspectRatio](#)
[DualDisplay_AutoSize](#)
[DualDisplay_Embedded](#)
[DualDisplay_FullScreen](#)
[DualDisplay_Height](#)
[DualDisplay_Left](#)
[DualDisplay_Monitor](#)
[DualDisplay_MouseMovesWindow](#)
[DualDisplay_PanScanRatio](#)
[DualDisplay_StayOnTop](#)
[DualDisplay_Top](#)
[DualDisplay_TransparentColorEnabled](#)
[DualDisplay_TransparentColorValue](#)
[DualDisplay_VideoHeight](#)
[DualDisplay_VideoPortEnabled](#)
[DualDisplay_VideoWidth](#)
[DualDisplay_VideoWindowHandle](#)
[DualDisplay_Visible](#)
[DualDisplay_Width](#)
[DVDateTimeEnabled](#)
[DVDiscontinuityMinimumInterval](#)
[DVTitle](#)
[DVEncoder_VideoFormat](#)
[DVEncoder_VideoResolution](#)
[DVEncoder_VideoStandard](#)
[DVRecordingInNativeFormatSeparatesStreams](#)
[DVReduceFrameRate](#)
[DVrgb219](#)
[DVTimeCodeEnabled](#)
[Enabled](#)
[EncryptionMethod](#)
[EventNotificationSynchron](#)
[ExtraDLLPath](#)
[FixFlickerOrBlackCapture](#)
[FrameCaptureHeight](#)
[FrameCaptureWidth](#)
[FrameCaptureWithoutOverlay](#)
[FrameCaptureZoomSize](#)
[FrameGrabber](#)
[FrameGrabberCurrentRGBFormat](#)
[FrameGrabberRGBFormat](#)
[FrameNumberStartsFromZero](#)
[FrameRate](#)

[FramerateDivider](#)
[FullRepaint](#)
[GetLastFrameWaitTimeoutMs](#)
[Height](#)
[HoldRecording](#)
[ImageOverlay_AlphaBlend](#)
[ImageOverlay_AlphaBlendValue](#)
[ImageOverlay_ChromaKey](#)
[ImageOverlay_ChromaKeyLeewayPercent](#)
[ImageOverlay_ChromaKeyRGBColor](#)
[ImageOverlay_Height](#)
[ImageOverlay_LeftLocation](#)
[ImageOverlay_RotationAngle](#)
[ImageOverlay_StretchToVideoSize](#)
[ImageOverlay_TopLocation](#)
[ImageOverlay_Transparent](#)
[ImageOverlay_TransparentColorValue](#)
[ImageOverlay_UseTransparentColor](#)
[ImageOverlay_VideoAlignment](#)
[ImageOverlay_Width](#)
[ImageOverlayEnabled](#)
[ImageOverlaySelector](#)
[ImageRatio](#)
[InFrameProgressEvent](#)
[IPCameraURL](#)
[IsAnalogVideoDecoderAvailable](#)
[IsAudioCrossbarAvailable](#)
[IsAudioDeviceASoundCard](#)
[IsAudioInputBalanceAvailable](#)
[IsCameraControlAvailable](#)
[IsDigitalVideoIn](#)
[IsDVCommandAvailable](#)
[IsHorizontalSyncLocked](#)
[IsMPEGStream](#)
[IsPlayerAudioStreamAvailable](#)
[IsPlayerVideoStreamAvailable](#)
[IsRecordingPaused](#)
[IsTimeCodeReaderAvailable](#)
[IsTVAudioAvailable](#)
[IsTVAutoTuneRunning](#)
[IsTVTunerAvailable](#)
[IsVideoControlAvailable](#)
[IsVideoCrossbarAvailable](#)
[IsVideoInterlaced](#)
[IsVideoPortAvailable](#)
[IsVideoQualityAvailable](#)
[IsWDMVideoDriver](#)
[JPEGPerformance](#)
[JPEGProgressiveDisplay](#)
[JPEGQuality](#)
[Last_BurstFrameCapture_FileName](#)
[Last_CaptureFrameTo_FileName](#)
[Last_Clip_Played](#)
[Last_Recording_FileName](#)
[Left](#)
[LogoDisplayed](#)
[LogoLayout](#)
[MixAudioSamples_CurrentSourceLevel](#)
[MixAudioSamples_ExternalSourceLevel](#)
[Mixer_MosaicColumns](#)
[Mixer_MosaicLines](#)
[MotionDetector_CompareBlue](#)
[MotionDetector_CompareGreen](#)

[MotionDetector_CompareRed](#)
[MotionDetector_Enabled](#)
[MotionDetector_GlobalMotionRatio](#)
[MotionDetector_GreyScale](#)
[MotionDetector_Grid](#)
[MotionDetector_GridXCount](#)
[MotionDetector_GridYCount](#)
[MotionDetector_IsGridValid](#)
[MotionDetector_MaxDetectionsPerSecond](#)
[MotionDetector_ReduceCPULoad](#)
[MotionDetector_ReduceVideoNoise](#)
[MotionDetector_Triggered](#)
[MouseWheelEventEnabled](#)
[MpegStreamType](#)
[MultiplexedInputEmulation](#)
[MultiplexedRole](#)
[MultiplexedStabilizationDelay](#)
[MultiplexedSwitchDelay](#)
[MuteAudioRendering](#)
[NDIFormatType](#)
[NDIName](#)
[NetworkStreaming](#)
[NetworkStreamingType](#)
[NewProperty2](#)
[NormalCursor](#)
[NotificationMethod](#)
[NotificationPriority](#)
[NotificationSleepTime](#)
[OnFrameBitmapEventSynchron](#)
[OpenURLAsync](#)
[OverlayAfterTransform](#)
[ParentColor](#)
[ParentShowHint](#)
[ParentWindow](#)
[PlayerAudioCodec](#)
[PlayerAudioRendering](#)
[PlayerDuration](#)
[PlayerDVSize](#)
[PlayerFastSeekSpeedRatio](#)
[PlayerFileName](#)
[PlayerForcedCodec](#)
[PlayerFrameCount](#)
[PlayerFramePosition](#)
[PlayerFrameRate](#)
[PlayerHwAccel](#)
[PlayerOpenProgressPercent](#)
[PlayerRefreshPausedDisplay](#)
[PlayerRefreshPausedDisplayFrameRate](#)
[PlayerSpeedRatio](#)
[PlayerSpeedRatioConstantAudioPitch](#)
[PlayerState](#)
[PlayerTimePosition](#)
[PlayerTrackBar](#)
[PlayerTrackBarScale](#)
[PlayerTrackBarSynchron](#)
[PlayerVideoCodec](#)
[PlaylistIndex](#)
[PopupMenu](#)
[PreallocCapFileCopiedAfterRecording](#)
[PreallocCapFileEnabled](#)
[PreallocCapFileName](#)
[PreallocCapFileSizeInMB](#)
[PreviewZoomSize](#)

[RawAudioSampleCapture](#)
[RawCaptureAsyncEvent](#)
[RawSampleCaptureLocation](#)
[RawVideoSampleCapture](#)
[RecordingAudioBitRate](#)
[RecordingBacktimedFramesCount](#)
[RecordingCanPause](#)
[RecordingDuration](#)
[RecordingFileName](#)
[RecordingFileSizeMaxInMB](#)
[RecordingFourCC](#)
[RecordingHeight](#)
[RecordingInNativeFormat](#)
[RecordingMethod](#)
[RecordingOnMotion_Enabled](#)
[RecordingOnMotion_MotionThreshold](#)
[RecordingOnMotion_NoMotionPauseDelayMs](#)
[RecordingPauseCreatesNewFile](#)
[RecordingSize](#)
[RecordingTimer](#)
[RecordingTimerInterval](#)
[RecordingVideoBitRate](#)
[RecordingWidth](#)
[Reencoding_IncludeAudioStream](#)
[Reencoding_IncludeVideoStream](#)
[Reencoding_Method](#)
[Reencoding_NewVideoClip](#)
[Reencoding_SourceVideoClip](#)
[Reencoding_StartFrame](#)
[Reencoding_StartTime](#)
[Reencoding_StopFrame](#)
[Reencoding_StopTime](#)
[Reencoding_UseAudioCompressor](#)
[Reencoding_UseFrameGrabber](#)
[Reencoding_UseVideoCompressor](#)
[Reencoding_WMVOutput](#)
[ScreenRecordingLayeredWindows](#)
[ScreenRecordingMonitor](#)
[ScreenRecordingNonVisibleWindows](#)
[ScreenRecordingSizePercent](#)
[ScreenRecordingThroughClipboard](#)
[ScreenRecordingWithCursor](#)
[SendToDV_DeviceIndex](#)
[ShapeOverlay](#)
[ShapeOverlayEnabled](#)
[SourceStream](#)
[SpeakerBalance](#)
[SpeakerControl](#)
[SpeakerVolume](#)
[StoragePath](#)
[StoreDeviceSettingsInRegistry](#)
[StreamingURL](#)
[StreamInterface_Format](#)
[StreamInterface_FrameRate](#)
[StreamInterface_IsRealTime](#)
[SyncCommands](#)
[SynchronizationRole](#)
[Synchronized](#)
[SyncPreview](#)
[SystemTempPath](#)
[Tag](#)
[TextOverlay_Align](#)
[TextOverlay_AlphaBlend](#)

[TextOverlay_AlphaBlendValue](#)
[TextOverlay_BkColor](#)
[TextOverlay_Enabled](#)
[TextOverlay_Font](#)
[TextOverlay_FontColor](#)
[TextOverlay_Left](#)
[TextOverlay_Right](#)
[TextOverlay_Scrolling](#)
[TextOverlay_ScrollingSpeed](#)
[TextOverlay_Selector](#)
[TextOverlay_Shadow](#)
[TextOverlay_ShadowColor](#)
[TextOverlay_ShadowDirection](#)
[TextOverlay_String](#)
[TextOverlay_Top](#)
[TextOverlay_Transparent](#)
[TextOverlay_VideoAlignment](#)
[ThirdPartyDeinterlacer](#)
[Top](#)
[TranslateMouseCoordinates](#)
[TunerFrequency](#)
[TunerMode](#)
[TVChannel](#)
[TVCountryCode](#)
[TVTunerInputType](#)
[TVUseFrequencyOverrides](#)
[UniqueID](#)
[UseClock](#)
[v360_AspectRatio](#)
[v360_Enabled](#)
[v360_MasterAngle](#)
[v360_MouseAction](#)
[v360_MouseActionPercent](#)
[VCRHorizontalLocking](#)
[Version](#)
[VideoCompression_DataRate](#)
[VideoCompression_KeyFrameRate](#)
[VideoCompression_PFramesPerKeyFrame](#)
[VideoCompression_Quality](#)
[VideoCompression_WindowSize](#)
[VideoCompressor](#)
[VideoCompressorName](#)
[VideoCompressors](#)
[VideoCompressorsCount](#)
[VideoControlSettings](#)
[VideoCursor](#)
[VideoDevice](#)
[VideoDeviceName](#)
[VideoDevices](#)
[VideoDevicesCount](#)
[VideoDevicesId](#)
[VideoDoubleBuffered](#)
[VideoFormat](#)
[VideoFormats](#)
[VideoFormatsCount](#)
[VideoFromImages_BitmapsSortedBy](#)
[VideoFromImages_RepeatIndefinitely](#)
[VideoFromImages_SourceDirectory](#)
[VideoFromImages_TemporaryFile](#)
[VideoHeight](#)
[VideoHeight_PreferredAspectRatio](#)
[VideoInput](#)
[VideoInputs](#)

[VideoInputsCount](#)
[VideoPlayableWhileRecording](#)
[VideoProcessing_Brightness](#)
[VideoProcessing_Contrast](#)
[VideoProcessing_Deinterlacing](#)
[VideoProcessing_FlipHorizontal](#)
[VideoProcessing_FlipVertical](#)
[VideoProcessing_GrayScale](#)
[VideoProcessing_Hue](#)
[VideoProcessing_InvertColors](#)
[VideoProcessing_Pixellization](#)
[VideoProcessing_Rotation](#)
[VideoProcessing_RotationCustomAngle](#)
[VideoProcessing_Saturation](#)
[VideoQualitySettings](#)
[VideoRenderer](#)
[VideoRendererExternal](#)
[VideoRendererExternalIndex](#)
[VideoRendererPriority](#)
[VideoSize](#)
[VideoSizes](#)
[VideoSizesCount](#)
[VideoSource](#)
[VideoSource_FileOrURL](#)
[VideoSource_FileOrURL_StartTime](#)
[VideoSource_FileOrURL_StopTime](#)
[VideoSources](#)
[VideoSourcesCount](#)
[VideoStreamNumber](#)
[VideoSubtype](#)
[VideoSubtypes](#)
[VideoSubtypesCount](#)
[VideoVisibleWhenStopped](#)
[VideoWidth](#)
[VideoWidth_PREFERREDASPECTRATIO](#)
[Visible](#)
[VUMeter](#)
[WebcamStillCaptureButton](#)
[Width](#)
[ZoomCoeff](#)
[ZoomXCenter](#)
[ZoomYCenter](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Support Your Windows Applications with HelpNDoc's CHM Generation](#)

AdjustOverlayAspectRatio

TVideoGrabber.AdjustOverlayAspectRatio

[Next](#)

[TVideoGrabber](#) [Properties](#)

Adjusts the aspect ratio of the video frames before applying the overlays

Declaration

property AdjustOverlayAspectRatio: Boolean **read** GetAdjustOverlayAspectRatio **write** SetAdjustOverlayAspectRatio **default** DEF_AdjustOverlayAspectRatio;

__property bool AdjustOverlayAspectRatio=read=GetAdjustOverlayAspectRatio, write=SetAdjustOverlayAspectRatio, **default**=1

Property AdjustOverlayAspectRatio As Boolean

Description

Used to adjust the aspect ratio of the video frames before applying the overlays.

This property applies only when the aspect ratio of the video source does not match the size of the video frames (e.g. a 16x9 aspect ratio for a 720x480 native video size)

AdjustOverlayAspectRatio enabled (requires a few CPU more):

The video frames are resized to their display aspect ratio **before** the overlays are applied, so the aspect ratio of the overlays is correct.

AdjustOverlayAspectRatio disabled (saves CPU):

The video frames are not resized, the overlays are applied to the native video size.

If the video is displayed to its correct aspect ratio (e.g. when [Display AspectRatio](#) = ar_Box), the video frames will be stretched, including the overlays that may appear distorted.

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEONHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlay](#) [FromBMPFile](#) [SetImageOverlay](#) [FromHBitmap](#) [SetImageOverlay](#) [FromHBitmap2](#) [SetImageOverlay](#) [FromImageFile](#) [SetImageOverlay](#) [FromImageFile2](#) [SetImageOverlay](#) [FromJPEGFile](#) [SetImageOverlay](#) [FromTBitmap](#) [SetImageOverlay](#) [FromTBitmap2](#) [SetImageOverlay](#) [FromTImage](#) [SetImageOverlay](#) [FromTImage2](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#) [SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#) [SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#) [SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#) [SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#) [TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#) [TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#) [TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#) [TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Say Goodbye to Documentation Headaches with a Help Authoring Tool](#)

AdjustPixelAspectRatio

TVideoGrabber.AdjustPixelAspectRatio

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Enable/disable the correction of non-square pixels

Declaration

property AdjustPixelAspectRatio: Boolean **read** GetAdjustPixelAspectRatio **write** SetAdjustPixelAspectRatio
default DEF_AdjustPixelAspectRatio;

__**property bool** AdjustPixelAspectRatio=read=GetAdjustPixelAspectRatio,
write=SetAdjustPixelAspectRatio, **default=1**

Property AdjustPixelAspectRatio As Boolean

Description

Used to enable / disable the correction of non-square pixels, e.g. when the video source is a PAL source or a NTSC source.

When [AdjustPixelAspectRatio](#) is enabled (by default) the pixel aspect ratio is corrected when a PAL or NTSC format is detected, and also according to the monitor aspect ratio. This concerns the video window and the frame capture size.

When [AdjustPixelAspectRatio](#) is disabled, the video frame will be processed "as is", but it can be stretched to the size of the video window if [Display_AutoSize](#) = false.

To display the video AS IS (without stretching the video frame to the video window size):

- if you want the control to be resized automatically, set [AdjustPixelAspectRatio](#) = false **and**

[Display_AutoSize](#) = true

- if you want to specify yourself the size of the control, set [AdjustPixelAspectRatio](#) = false **and**

[Display_AutoSize](#) = false **and** [Display_AspectRatio](#) = ar_NoResize

See Also

[TVideoRenderer Display_Active Display_AlphaBlendEnabled Display_AlphaBlendValue Display_AutoSize Display_Embedded Display_FullScreen Display_Height Display_Left Display_Monitor Display_MouseMovesWindow Display_PanScanRatio Display_SetLocation Display_StayOnTop Display_Top Display_TransparentColorEnabled Display_TransparentColorValue Display_VideoHeight Display_VideoPortEnabled Display_VideoWidth Display_VideoWindowHandle Display_Width DualDisplay_Active DualDisplay_AlphaBlendEnabled DualDisplay_AlphaBlendValue DualDisplay_AutoSize DualDisplay_Embedded DualDisplay_FullScreen DualDisplay_Height DualDisplay_Left DualDisplay_Monitor DualDisplay_MouseMovesWindow DualDisplay_PanScanRatio DualDisplay_StayOnTop DualDisplay_Top DualDisplay_TransparentColorEnabled DualDisplay_TransparentColorValue DualDisplay_VideoHeight DualDisplay_VideoPortEnabled DualDisplay_VideoWidth DualDisplay_VideoWindowHandle DualDisplay_Visible DualDisplay_Width IsVideoPortAvailable Monitor_Primary Index MonitorBounds MonitorsCount OnLeavingFullScreen SetParentWindow SetWindowTransparency VideoDoubleBuffered VideoHeight PreferredAspectRatio VideoRendererExternal VideoRendererExternalIndex VideoRendererPriority VideoVisibleWhenStopped VideoWidth PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Revolutionize Your CHM Help File Output with HelpNDoc](#)

Aero

TVideoGrabber.Aero

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Temporarily disables/enables the Aero mode of Vista/Windows 7

Declaration

property Aero: TAero **read** GetAero **write** SetAero **default** DEF_Aero;

__**property** TAero Aero=read=GetAero, write=SetAero, **default=1**

Property Aero As TAero

Description

This [TAero](#) property is used to disable/enable the Windows Aero mode (the transparency of the window titles in Vista / Windows 7)

By default the Windows Vista and Windows 7 desktop use the "Aero" mode (in this mode you can see the background under the transparent borders and title of the windows).

In this mode the overlay rendering can't be used.

To get a better video quality when playing MPEG clips in full screen mode with the VMR7 / VMR9 and the frame grabber is disabled, you can temporarily disable the Aero mode then the video starts playing by setting Aero = **ae_ForceOffWhenStartingVideo**

If you prefer to disable the Aero mode immediately set Aero = ae_ForceOffImmediately (then set Aero = ae_ForceOnImmediately to restore it).

Note that it is not possible to enable the Aero mode if it was disabled when starting the application (unless changing the properties of the desktop), it is only possible to disable the Aero mode if it was enabled, then re-enable it and re-disable as needed.

Created with the Standard Edition of HelpNDoc: [Transform Your Documentation Process with HelpNDoc's Project Analyzer](#)

AnalogVideoStandard

TVideoGrabber.AnalogVideoStandard

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Selects an analog video standard.

Declaration

property AnalogVideoStandard: LongInt **read** GetAnalogVideoStandard **write** SetAnalogVideoStandard;

__property **int** AnalogVideoStandard=read=GetAnalogVideoStandard, write=SetAnalogVideoStandard, **nodefault**

Property AnalogVideoStandard As Long

Description

Used to select an analog video standard for the [current video capture device](#) in the [AnalogVideoStandards](#) list.

Important:

The analog standard is not the same from a video capture device to the other. It is reloaded when the video capture device is selected by assigning the [VideoDevice](#) property.

- any control displaying this value should re-read it from the [OnVideoDeviceSelected](#) event (that occurs when the video device is selected)
- be sure that the video capture device has been selected (by assigning the [VideoDevice](#) property) before setting this property

See [device-dependent properties](#)

See Also

[Analog Video standards](#) [AnalogVideoStandardIndex](#) [AnalogVideoStandards](#) [AnalogVideoStandardsCount](#) [IsAnalogVideoDecoderAvailable](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Reach: Convert Your Word Document to an ePub or Kindle eBook](#)

AnalogVideoStandards

TVideoGrabber.AnalogVideoStandards

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

List of analog video standards available for the current video capture device.

Declaration

property AnalogVideoStandards: **string read** GetAnalogVideoStandards;

__property wchar_t *AnalogVideoStandards=read=GetAnalogVideoStandards

Property AnalogVideoStandards As String

Description

Used to retrieve a string that contains the list of the video standards available on the current video capture device when when [IsAnalogVideoDecoderAvailable](#) returns true.

The [AnalogVideoStandard](#) property is an index in this list, used to select the current analog video standard.

This list is updated when the [OnVideoDeviceSelected](#) event occurs (when a video capture device is selected with [VideoDevice](#)).

This list can be assigned to list based controls. E.g.:

ComboBox1.Items.Text := VideoGrabber1.AnalogVideoStandards

ComboBox1.ItemIndex := VideoGrabber1.AnalogVideoStandard

See Also

[Analog Video standards](#) [AnalogVideoStandard](#) [AnalogVideoStandardIndex](#) [AnalogVideoStandardsCount](#) [IsAnalogVideoDecoderAvailable](#)

Created with the Standard Edition of HelpNDoc: [Produce electronic books easily](#)

AnalogVideoStandardsCount

TVideoGrabber.AnalogVideoStandardsCount

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Number of analog video standards.

Declaration

property AnalogVideoStandardsCount: LongInt **read** GetAnalogVideoStandardsCount;

__property **int** AnalogVideoStandardsCount=read=GetAnalogVideoStandardsCount, **nodefault**

Property AnalogVideoStandardsCount As Long

Description

Number of analog video standards in the [AnalogVideoStandards](#) list for the current video capture device.

The [AnalogVideoStandard](#) property is an index used to select the current analog video standard, in the **0 ... AnalogVideoStandardsCount - 1** range.

See Also

[Analog Video standards](#) [AnalogVideoStandard](#) [AnalogVideoStandardIndex](#) [AnalogVideoStandards](#) [IsAnalogVideoDecoderAvailable](#)

Created with the Standard Edition of HelpNDoc: [Why Microsoft Word Isn't Cut Out for Documentation: The Benefits of a Help Authoring Tool](#)

ApplicationPriority

TVideoGrabber.ApplicationPriority

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Sets the priority of the application.

Declaration

property ApplicationPriority: TApplicationPriority **read** GetApplicationPriority **write** SetApplicationPriority
default DEF_ApplicationPriority;

__property TApplicationPriority ApplicationPriority=read=GetApplicationPriority,
 write=SetApplicationPriority, **default**=0

Description

Used to set the priority of the application.

| Values | normal priority | idle priority | normal priority | high priority | real time priority |
|-------------|-----------------|---------------|-----------------|---------------|--|
| ap_default: | ap_idle: | ap_normal: | ap_high: | ap_realtime: | |
| | | | | | Although it looks like threading priority, this property is used to select the accuracy of the frame counting by the OnFrameProgress2 event. |
| | | | | | It is recommend to leave it to its default value. |
| | | | | | Setting it to ap_idle may reduce the CPU load but may not report all the frames. |

Created with the Standard Edition of HelpNDoc: [Easy to use tool to create HTML Help files and Help web sites](#)

ASFAudioBitRate

TVideoGrabber.ASFAudioBitRate

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the bit rate of the audio stream in bits per second.

Declaration

property ASFAudioBitRate: LongInt **read** GetASFAudioBitRate **write** SetASFAudioBitRate **default** DEF_ASFAudioBitRate;

__property int ASFAudioBitRate=read=GetASFAudioBitRate, write=SetASFAudioBitRate, **default**=- 1

Property ASFAudioBitRate As Long

Description

Specifies the bit rate of the audio stream in bits per second.

If the specified value is -1, the default ASF value is used, or the current [ASF_profile](#) value if an ASF profile is used.

E.g. a value of 40000 specifies about 40 Kbits per second.

See Also

[TNetworkStreamingType](#) [TOnClientConnection](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#) [ASFAudioChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#) [ASFMediaServerPublishingPoint](#)

[ASFMediaServerRemovePublishingPointAfterDisconnect](#) [ASFMediaServerTemplatePublishingPoint](#)
[ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#) [ASFProfileFromCustomFile](#) [ASFProfiles](#)
[ASFProfilesCount](#) [ASFProfileVersion](#) [ASFStreaming](#) [GetAuthorizationList](#)
[ASFStreaming](#) [GetConnectedClients](#) [ASFStreaming](#) [GetConnectedClientsCount](#)
[ASFStreaming](#) [ResetAuthorizations](#) [ASFStreaming](#) [SetAuthorization](#) [ASFVideoBitRate](#)
[ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#)
[NetworkStreaming](#) [NetworkStreamingType](#) [OnAuthenticationNeeded](#) [OnClientConnection](#)
[OnDirectNetworkStreamingHostUrl](#) [ShowDialog](#) [StartAudioRendering](#) [StreamingURL](#)

Created with the Standard Edition of HelpNDoc: [Streamline your documentation process with HelpNDoc's HTML5 template](#)

ASFAudioChannels

TVideoGrabber.ASFAudioChannels

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the number of channels of the audio stream.

Declaration

property ASFAudioChannels: LongInt **read** GetASFAudioChannels **write** SetASFAudioChannels **default** DEF_ASFAudioChannels;

__property **int** ASFAudioChannels=read=GetASFAudioChannels, write=SetASFAudioChannels, **default**=- 1

Property ASFAudioChannels As Long

Description

Specifies the number of channels of the audio stream.

If the specified value is -1, the default ASF value is used, or the current [ASF profile](#) value if an ASF profile is used.

1: 1 channel (mono)

2: 2 channels (stereo)

See Also

[TNetworkStreamingType](#) [TOnClientConnection](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#)
[ASFAudioBitRate](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#) [ASFMediaServerPublishingPoint](#)
[ASFMediaServerRemovePublishingPointAfterDisconnect](#) [ASFMediaServerTemplatePublishingPoint](#)
[ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#) [ASFProfileFromCustomFile](#) [ASFProfiles](#)
[ASFProfilesCount](#) [ASFProfileVersion](#) [ASFStreaming](#) [GetAuthorizationList](#)
[ASFStreaming](#) [GetConnectedClients](#) [ASFStreaming](#) [GetConnectedClientsCount](#)
[ASFStreaming](#) [ResetAuthorizations](#) [ASFStreaming](#) [SetAuthorization](#) [ASFVideoBitRate](#)
[ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#)
[NetworkStreaming](#) [NetworkStreamingType](#) [OnAuthenticationNeeded](#) [OnClientConnection](#)
[OnDirectNetworkStreamingHostUrl](#) [ShowDialog](#) [StartAudioRendering](#) [StreamingURL](#)

Created with the Standard Edition of HelpNDoc: [Write eBooks for the Kindle](#)

ASFBufferWindow

TVideoGrabber.ASFBufferWindow

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Increases the streaming latency

Declaration

property ASFBufferWindow: LongInt **read** GetASFBufferWindow **write** SetASFBufferWindow **default** DEF_ASFBufferWindow;

__property **int** ASFBufferWindow=read=GetASFBufferWindow, write=SetASFBufferWindow, **default**=- 1

Property ASFBufferWindow As Long

Description

Used to increase the streaming latency, expressed in milliseconds.

Created with the Standard Edition of HelpNDoc: [What is a Help Authoring tool?](#)

ASFDeinterlaceMode

TVideoGrabber.ASFDeinterlaceMode

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies if the video stream must be deinterlaced.

Declaration

property ASFDeinterlaceMode: TASFDeinterlaceMode **read** GetASFDeinterlaceMode **write** SetASFDeinterlaceMode **default** DEF_ASFDeinterlaceMode;

__property TASFDeinterlaceMode ASFDeinterlaceMode=read=GetASFDeinterlaceMode, write=SetASFDeinterlaceMode, **default**=0

Property ASFDeinterlaceMode As TxASFDeinterlaceMode

Description

Specifies if the video stream must be deinterlaced before being recorded.
The value is chosen in the [TASFDeinterlaceMode](#) type.

See Also

[TNetworkStreamingType](#) [TOnClientConnection](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#) [ASFAudioBitRate](#) [ASFAudioChannels](#) [ASFFixedFrameRate](#) [ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#) [ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#) [ASFProfileFromCustomFile](#) [ASFProfiles](#) [ASFProfilesCount](#) [ASFProfileVersion](#) [ASFStreaming](#) [GetAuthorizationList](#) [ASFStreaming_GetConnectedClients](#) [ASFStreaming_GetConnectedClientsCount](#) [ASFStreaming_ResetAuthorizations](#) [ASFStreaming_SetAuthorization](#) [ASFVideoBitRate](#) [ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#) [NetworkStreaming](#) [NetworkStreamingType](#) [OnAuthenticationNeeded](#) [OnClientConnection](#) [OnDirectNetworkStreamingHostUrl](#) [ShowDialog](#) [StartAudioRendering](#) [StreamingURL](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Convert Your Word Doc to an eBook: A Step-by-Step Guide](#)

ASFFixedFrameRate

TVideoGrabber.ASFFixedFrameRate

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Forces the ASF recording to use a fixed frame rate.

Declaration

property ASFFixedFrameRate: Boolean **read** GetASFFixedFrameRate **write** SetASFFixedFrameRate **default** DEF_ASFFixedFrameRate;

__property **bool** ASFFixedFrameRate=read=GetASFFixedFrameRate, write=SetASFFixedFrameRate, **default**=0

Property ASFFixedFrameRate As Boolean

Description

Specifies if a fixed frame rate must be used to record the video stream.
By default the ASF frame rate is not constant.

See Also

[TNetworkStreamingType](#) [TOnClientConnection](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#)
[ASFVideoBitRate](#) [ASFVideoChannels](#) [ASFDeinterlaceMode](#) [ASFMediaServerPublishingPoint](#)
[ASFMediaServerRemovePublishingPointAfterDisconnect](#) [ASFMediaServerTemplatePublishingPoint](#)
[ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#) [ASFProfileFromCustomFile](#) [ASFProfiles](#)
[ASFProfilesCount](#) [ASFProfileVersion](#) [ASFStreaming](#) [GetAuthorizationList](#)
[ASFStreaming](#) [GetConnectedClients](#) [ASFStreaming](#) [GetConnectedClientsCount](#)
[ASFStreaming](#) [ResetAuthorizations](#) [ASFStreaming](#) [SetAuthorization](#) [ASFVideoBitRate](#)
[ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#)
[NetworkStreaming](#) [NetworkStreamingType](#) [OnAuthenticationNeeded](#) [OnClientConnection](#)
[OnDirectNetworkStreamingHostUrl](#) [ShowDialog](#) [StartAudioRendering](#) [StreamingURL](#)

Created with the Standard Edition of HelpNDoc: Effortlessly optimize your documentation website for search engines

ASFMediaServerPublishingPoint

TVideoGrabber.ASFMediaServerPublishingPoint

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies a publishing point on a Windows Media Server.

Declaration

property ASFMediaServerPublishingPoint: **string** **read** GetASFMediaServerPublishingPoint **write** SetASFMediaServerPublishingPoint;

```
__property wchar_t *ASFMediaServerPublishingPoint=read=GetASFMediaServerPublishingPoint,
write=SetASFMediaServerPublishingPoint
```

Property ASFMediaServerPublishingPoint As String

Description

Used to specify a publishing point on a Windows Media Server, to which the audio/video stream will be sent.

This property is used when [Networking](#) = ns_ASFStreamingToPublishingPoint.

E.g.: http://your_WM_server.com/your_publishing_point

See Also

[TNetworkStreamingType](#) [TOnClientConnection](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#)
[ASFVideoBitRate](#) [ASFVideoChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#)
[ASFMediaServerRemovePublishingPointAfterDisconnect](#) [ASFMediaServerTemplatePublishingPoint](#)
[ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#) [ASFProfileFromCustomFile](#) [ASFProfiles](#)
[ASFProfilesCount](#) [ASFProfileVersion](#) [ASFStreaming](#) [GetAuthorizationList](#)
[ASFStreaming](#) [GetConnectedClients](#) [ASFStreaming](#) [GetConnectedClientsCount](#)
[ASFStreaming](#) [ResetAuthorizations](#) [ASFStreaming](#) [SetAuthorization](#) [ASFVideoBitRate](#)
[ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#)
[NetworkStreaming](#) [NetworkStreamingType](#) [OnAuthenticationNeeded](#) [OnClientConnection](#)
[OnDirectNetworkStreamingHostUrl](#) [ShowDialog](#) [StartAudioRendering](#) [StreamingURL](#)

Created with the Standard Edition of HelpNDoc: Generate Kindle eBooks with ease

ASFMediaServerRemovePublishingPointAfterDisconnect

TVideoGrabber.ASFMediaServerRemovePublishingPointAfterDisconnect

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Properties](#)

Specifies if a publishing point created dynamically must be destroyed.

Declaration

property ASFMediaServerTemplatePublishingPoint: **string read**

GetASFMediaServerTemplatePublishingPoint **write** SetASFMediaServerTemplatePublishingPoint;

```
__property wchar_t *ASFMediaServerPublishingPoint=read=GetASFMediaServerTemplatePublishingPoint,
write=SetASFMediaServerTemplatePublishingPoint
```

Property ASFMediaServerTemplatePublishingPoint As String

Description

Used to specify if the publishing point created dynamically on the Windows Media server must be destroyed when disconnecting.

See Also

[TNetworkStreamingType](#) [TOnClientConnection](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#)
[ASFAudioBitRate](#) [ASFAudioChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#)
[ASFMediaServerPublishingPoint](#) [ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkMaxUsers](#)
[ASFNetworkPort](#) [ASFProfile](#) [ASFProfileFromCustomFile](#) [ASFProfiles](#) [ASFProfilesCount](#) [ASFProfileVersion](#)
[ASFStreaming](#) [GetAuthorizationList](#) [ASFStreaming](#) [GetConnectedClients](#)
[ASFStreaming](#) [GetConnectedClientsCount](#) [ASFStreaming](#) [ResetAuthorizations](#)
[ASFStreaming](#) [SetAuthorization](#) [ASFVideoBitRate](#) [ASFVideoFrameRate](#) [ASFVideoHeight](#)
[ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#) [NetworkStreaming](#)
[NetworkStreamingType](#) [OnAuthenticationNeeded](#) [OnClientConnection](#) [OnDirectNetworkStreamingHostUrl](#)
[ShowDialog](#) [StartAudioRendering](#) [StreamingURL](#)

Created with the Standard Edition of HelpNDoc: [Free Qt Help documentation generator](#)

ASFMediaServerTemplatePublishingPoint

TVideoGrabber.ASFMediaServerTemplatePublishingPoint

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Properties](#)

Specifies a template publishing point on a Windows Media Server.

Declaration

property ASFMediaServerRemovePublishingPointAfterDisconnect: **boolean read**

GetASFMediaServerRemovePublishingPointAfterDisconnect **write**

SetASFMediaServerRemovePublishingPointAfterDisconnect;

```
__property bool
```

```
ASFMediaServerRemovePublishingPointAfterDisconnect=read=GetASFMediaServerRemovePublishingPointAfterDisconnect, write=SetASFMediaServerRemovePublishingPointAfterDisconnect
```

Property ASFMediaServerRemovePublishingPointAfterDisconnect As Bool

Description

Used to specify the name of a template publishing point on the Windows Media Server, that will be used to create automatically the publishing point specified by [ASFMediaServerPublishingPoint](#) .

This property is used when [Networking](#) = ns_ASFStreamingToPublishingPoint.

See Also

[TNetworkStreamingType](#) [TOnClientConnection](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#)
[ASFAudioBitRate](#) [ASFAudioChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#)
[ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#)
[ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#) [ASFProfileFromCustomFile](#) [ASFProfiles](#)
[ASFProfilesCount](#) [ASFProfileVersion](#) [ASFStreaming](#) [GetAuthorizationList](#)
[ASFStreaming](#) [GetConnectedClients](#) [ASFStreaming](#) [GetConnectedClientsCount](#)
[ASFStreaming](#) [ResetAuthorizations](#) [ASFStreaming](#) [SetAuthorization](#) [ASFVideoBitRate](#)
[ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#)
[NetworkStreaming](#) [NetworkStreamingType](#) [OnAuthenticationNeeded](#) [OnClientConnection](#)
[OnDirectNetworkStreamingHostUrl](#) [ShowDialog](#) [StartAudioRendering](#) [StreamingURL](#)

Created with the Standard Edition of HelpNDoc: [iPhone web sites made easy](#)

ASFNetworkMaxUsers

TVideoGrabber.ASFNetworkMaxUsers

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the maximum number of users for direct network streaming.

Declaration

property ASFNetworkMaxUsers: LongInt **read** GetASFNetworkMaxUsers **write** SetASFNetworkMaxUsers
default DEF_ASFNetworkMaxUsers;

__property int ASFNetworkMaxUsers=read=GetASFNetworkMaxUsers, write=SetASFNetworkMaxUsers,
default=5

Property ASFNetworkMaxUsers As Long

Description

Used to specify the maximum number of users that can connect to the current platform when network streaming is enabled by [Networking](#) = ns_ASFDirectNetworkStreaming

See Also

[TNetworkStreamingType](#) [TOnClientConnection](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#)
[ASFAudioBitRate](#) [ASFAudioChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#)
[ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#)
[ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkPort](#) [ASFProfile](#) [ASFProfileFromCustomFile](#)
[ASFProfiles](#) [ASFProfilesCount](#) [ASFProfileVersion](#) [ASFStreaming](#) [GetAuthorizationList](#)
[ASFStreaming](#) [GetConnectedClients](#) [ASFStreaming](#) [GetConnectedClientsCount](#)
[ASFStreaming](#) [ResetAuthorizations](#) [ASFStreaming](#) [SetAuthorization](#) [ASFVideoBitRate](#)
[ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#)
[NetworkStreaming](#) [NetworkStreamingType](#) [OnAuthenticationNeeded](#) [OnClientConnection](#)
[OnDirectNetworkStreamingHostUrl](#) [ShowDialog](#) [StartAudioRendering](#) [StreamingURL](#)

Created with the Standard Edition of HelpNDoc: [Full-featured EPub generator](#)

ASFNetworkPort

TVideoGrabber.ASFNetworkPort

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the network port for direct network streaming.

Declaration

property ASFNetworkPort: LongInt **read** GetASFNetworkPort **write** SetASFNetworkPort **default** DEF_ASFNetworkPort;

__property int ASFNetworkPort=read=GetASFNetworkPort, write=SetASFNetworkPort, **default**=0

Property ASFNetworkPort As Long

Description

Used to specify the network port used to perform network streaming from the when network streaming is enabled by [Networking](#) = ns_ASFDirectNetworkStreaming

If this property is set to 0, a valid port number is automatically assigned and returned by the [OnDirectNetworkStreamingHostUrl](#) event.

See Also

[TNetworkStreamingType](#) [TOnClientConnection](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#) [ASFAudioBitRate](#) [ASFAudioChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#) [ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#) [ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkMaxUsers](#) [ASFProfile](#) [ASFProfileFromCustomFile](#) [ASFProfiles](#) [ASFProfilesCount](#) [ASFProfileVersion](#) [ASFStreaming](#) [GetAuthorizationList](#) [ASFStreaming](#) [GetConnectedClients](#) [ASFStreaming](#) [GetConnectedClientsCount](#) [ASFStreaming](#) [ResetAuthorizations](#) [ASFStreaming](#) [SetAuthorization](#) [ASFVideoBitRate](#) [ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#) [NetworkStreaming](#) [NetworkStreamingType](#) [OnAuthenticationNeeded](#) [OnClientConnection](#) [OnDirectNetworkStreamingHostUrl](#) [ShowDialog](#) [StartAudioRendering](#) [StreamingURL](#)

Created with the Standard Edition of HelpNDoc: [Produce Kindle eBooks easily](#)

ASFProfile

TVideoGrabber.ASFProfile

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Index of the current ASF profile.

Declaration

property ASFProfile: Integer **read** GetASFProfile **write** SetASFProfile **default** DEF_ASFProfile;

__property int ASFProfile=read=GetASFProfile, write=SetASFProfile, **default**=- 1

Property ASFProfile As Long

Description

Index of the current ASF profile in the [ASFProfiles](#) list.

Set this value to -1 to disable ASF profiles and use default ASF streaming values.

See Also

[TNetworkStreamingType](#) [TOnClientConnection](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#) [ASFAudioBitRate](#) [ASFAudioChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#) [ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#) [ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfileFromCustomFile](#) [ASFProfiles](#) [ASFProfilesCount](#) [ASFProfileVersion](#) [ASFStreaming](#) [GetAuthorizationList](#) [ASFStreaming](#) [GetConnectedClients](#) [ASFStreaming](#) [GetConnectedClientsCount](#) [ASFStreaming](#) [ResetAuthorizations](#) [ASFStreaming](#) [SetAuthorization](#) [ASFVideoBitRate](#) [ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#) [NetworkStreaming](#) [NetworkStreamingType](#) [OnAuthenticationNeeded](#) [OnClientConnection](#) [OnDirectNetworkStreamingHostUrl](#) [ShowDialog](#) [StartAudioRendering](#) [StreamingURL](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Productivity with a Help Authoring Tool](#)

ASFProfileFromCustomFile

TVideoGrabber.ASFProfileFromCustomFile

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Selects a file that contains a custom profile

Declaration

property ASFProfileFromCustomFile: **string read** GetASFProfileFromCustomFile **write** SetASFProfileFromCustomFile;

__property wchar_t *ASFProfileFromCustomFile=read=GetASFProfileFromCustomFile, write=SetASFProfileFromCustomFile

Property ASFProfileFromCustomFile As String

Description

Used to select a file that contains a custom profile (usually a .prx file).

When a file is assigned to this property, the [ASFProfile](#) property is ignored and this profile is used instead.

E.g. ASFProfileFromCustomFile = "myprofile.prx"

See Also

[TNetworkStreamingType](#) [TOnClientConnection](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#) [ASFAudioBitRate](#) [ASFAudioChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#) [ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#) [ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#) [ASFProfiles](#) [ASFProfilesCount](#) [ASFProfileVersion](#) [ASFStreaming](#) [GetAuthorizationList](#) [ASFStreaming](#) [GetConnectedClients](#) [ASFStreaming](#) [GetConnectedClientsCount](#) [ASFStreaming](#) [ResetAuthorizations](#) [ASFStreaming](#) [SetAuthorization](#) [ASFVideoBitRate](#) [ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#) [NetworkStreaming](#) [NetworkStreamingType](#) [OnAuthenticationNeeded](#) [OnClientConnection](#) [OnDirectNetworkStreamingHostUrl](#) [ShowDialog](#) [StartAudioRendering](#) [StreamingURL](#)

Created with the Standard Edition of HelpNDoc: Make Your PDFs More Secure with Encryption and Password Protection

ASFProfiles

TVideoGrabber.ASFProfiles

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Returns the ASF profiles available on the current platform.

Declaration

property ASFProfiles: **String read** GetASFProfiles;

__property wchar_t *ASFProfiles=read=GetASFProfiles

Property ASFProfiles As String

Description

Used to retrieve the ASF profiles available on the current platform.

The [ASFProfile](#) property is the index in this list of the current ASF profile.

Selecting an ASF profile let you choose the streaming format (video size, bit rate, etc...).

This list can be assigned to a TStringList, e.g. :

```
...
StringList := TStringList.Create;
StringList.Text := VideoGrabber.ASFProfiles;
...
```

See Also

[TNetworkStreamingType](#) [TOnClientConnection](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#)
[ASFAudioBitRate](#) [ASFAudioChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#)
[ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#)
[ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#)
[ASFProfileFromCustomFile](#) [ASFProfilesCount](#) [ASFProfileVersion](#) [ASFStreaming](#) [GetAuthorizationList](#)
[ASFStreaming](#) [GetConnectedClients](#) [ASFStreaming](#) [GetConnectedClientsCount](#)
[ASFStreaming](#) [ResetAuthorizations](#) [ASFStreaming](#) [SetAuthorization](#) [ASFVideoBitRate](#)
[ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#)
[NetworkStreaming](#) [NetworkStreamingType](#) [OnAuthenticationNeeded](#) [OnClientConnection](#)
[OnDirectNetworkStreamingHostUrl](#) [ShowDialog](#) [StartAudioRendering](#) [StreamingURL](#)

Created with the Standard Edition of HelpNDoc: [Single source CHM, PDF, DOC and HTML Help creation](#)

ASFProfilesCount

TVideoGrabber.ASFProfilesCount

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Number of ASF profiles available on the current platform.

Declaration

property ASFProfilesCount: LongInt **read** GetASFProfilesCount;

___property **int** ASFProfilesCount=read=GetASFProfilesCount, **nodefault**

Property ASFProfilesCount As Long

Description

Used to retrieve the number of ASF profiles available on the current platform.

The [ASFProfile](#) index must be in the 0...[ASFProfilesCount](#) range.

See Also

[TNetworkStreamingType](#) [TOnClientConnection](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#)
[ASFAudioBitRate](#) [ASFAudioChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#)
[ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#)
[ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#)
[ASFProfileFromCustomFile](#) [ASFProfiles](#) [ASFProfileVersion](#) [ASFStreaming](#) [GetAuthorizationList](#)
[ASFStreaming](#) [GetConnectedClients](#) [ASFStreaming](#) [GetConnectedClientsCount](#)
[ASFStreaming](#) [ResetAuthorizations](#) [ASFStreaming](#) [SetAuthorization](#) [ASFVideoBitRate](#)
[ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#)
[NetworkStreaming](#) [NetworkStreamingType](#) [OnAuthenticationNeeded](#) [OnClientConnection](#)
[OnDirectNetworkStreamingHostUrl](#) [ShowDialog](#) [StartAudioRendering](#) [StreamingURL](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly upgrade your WinHelp HLP help files to CHM with HelpNDoc](#)

ASFProfileVersion

TVideoGrabber.ASFProfileVersion

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the Windows Media Format mode (WMV8 or WMV9).

Declaration

property ASFProfileVersion: TASFPProfileVersion **read** GetASFProfileVersion **write** SetASFProfileVersion
default DEF_ASFProfileVersion;

__property TASFPProfileVersion ASFProfileVersion==GetASFProfileVersion, write=SetASFProfileVersion,
 default=0;

property ASFProfileVersion as TxASFProfileVersion

Description

Used to specify the Windows Media profiles version (8 or 9), that determines the Windows Media Format mode used (WMV8 or WMV9).

apv_ProfileVersion_8: uses the Windows Media 8 encoders and profiles

apv_ProfileVersion_9: uses the Windows Media 9 encoders and profiles

See Also

[TNetworkStreamingType](#) [TOnClientConnection](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#)
[ASFAudioBitRate](#) [ASFAudioChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#)
[ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#)
[ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#)
[ASFProfileFromCustomFile](#) [ASFProfiles](#) [ASFProfilesCount](#) [ASFStreaming](#) [GetAuthorizationList](#)
[ASFStreaming](#) [GetConnectedClients](#) [ASFStreaming](#) [GetConnectedClientsCount](#)
[ASFStreaming](#) [ResetAuthorizations](#) [ASFStreaming](#) [SetAuthorization](#) [ASFVideoBitRate](#)
[ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#)
[NetworkStreaming](#) [NetworkStreamingType](#) [OnAuthenticationNeeded](#) [OnClientConnection](#)
[OnDirectNetworkStreamingHostUrl](#) [ShowDialog](#) [StartAudioRendering](#) [StreamingURL](#)

Created with the Standard Edition of HelpNDoc: Leave the tedious WinHelp HLP to CHM conversion process behind with HelpNDoc

ASFVideoBitRate**TVideoGrabber.ASFVideoBitRate**
[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the bit rate of the video stream in bits per second.

Declaration

property ASFVideoBitRate: LongInt **read** GetASFVideoBitRate **write** SetASFVideoBitRate **default**
 DEF_ASFVideoBitRate;

__property int ASFVideoBitRate=read=GetASFVideoBitRate, write=SetASFVideoBitRate, **default=- 1**

Property ASFVideoBitRate As Long

Description

Specifies the bit rate of the video stream in bits per second.

If the specified value is -1, the default ASF value is used, or the current [ASF profile](#) value if an ASF profile is used.

E.g. a value of 185000 specifies about 185 Kbits per second.

See Also

[TNetworkStreamingType](#) [TOnClientConnection](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#)
[ASFAudioBitRate](#) [ASFAudioChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#)
[ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#)
[ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#)
[ASFProfileFromCustomFile](#) [ASFProfiles](#) [ASFProfilesCount](#) [ASFProfileVersion](#)
[ASFStreaming](#) [GetAuthorizationList](#) [ASFStreaming](#) [GetConnectedClients](#)

[ASFStreaming_GetConnectedClientsCount](#) [ASFStreaming_ResetAuthorizations](#)
[ASFStreaming_SetAuthorization](#) [ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#)
[ASFVideoQuality](#) [ASFVideoWidth](#) [NetworkStreaming](#) [NetworkStreamingType](#) [OnAuthenticationNeeded](#)
[OnClientConnection](#) [OnDirectNetworkStreamingHostUrl](#) [ShowDialog](#) [StartAudioRendering](#) [StreamingURL](#)

Created with the Standard Edition of HelpNDoc: [Don't Let Unauthorized Users View Your PDFs: Learn How to Set Passwords](#)

ASFVideoFrameRate

TVideoGrabber.ASFVideoFrameRate

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the frame rate of the video stream in frames per second.

Declaration

property ASFVideoFrameRate: Double **read** GetASFVideoFrameRate **write** SetASFVideoFrameRate
default DEF_ASFVideoFrameRate;

__**property** double ASFVideoFrameRate=read=GetASFVideoFrameRate, write=SetASFVideoFrameRate,
default=- 1

Property ASFVideoFrameRate As Double

Description

Specifies the frame rate of the video stream expressed in frames per second.

If the specified value is -1, the frame rate of the video source is used, or the current [ASF profile](#) value if an ASF profile is used.

E.g. a value of 29.97 specifies 29.97 frames per second

See Also

[TNetworkStreamingType](#) [TOnClientConnection](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#)
[ASFAudioBitRate](#) [ASFAudioChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#)
[ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#)
[ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#)
[ASFProfileFromCustomFile](#) [ASFProfiles](#) [ASFProfilesCount](#) [ASFProfileVersion](#)
[ASFStreaming_GetAuthorizationList](#) [ASFStreaming_GetConnectedClients](#)
[ASFStreaming_GetConnectedClientsCount](#) [ASFStreaming_ResetAuthorizations](#)
[ASFStreaming_SetAuthorization](#) [ASFVideoBitRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#)
[ASFVideoQuality](#) [ASFVideoWidth](#) [NetworkStreaming](#) [NetworkStreamingType](#) [OnAuthenticationNeeded](#)
[OnClientConnection](#) [OnDirectNetworkStreamingHostUrl](#) [ShowDialog](#) [StartAudioRendering](#) [StreamingURL](#)

Created with the Standard Edition of HelpNDoc: [Make the switch to CHM with HelpNDoc's hassle-free WinHelp HLP to CHM conversion tool](#)

ASFVideoHeight

TVideoGrabber.ASFVideoHeight

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Video height for the ASF video stream.

Declaration

property ASFVideoHeight: LongInt **read** GetASFVideoHeight **write** SetASFVideoHeight **default**
DEF_ASFVideoHeight;

__**property** int ASFVideoHeight=read=GetASFVideoHeight, write=SetASFVideoHeight, **default**=240

Property ASFVideoHeight As Long

Description

Specifies the video height for the ASF video stream.

If the specified value is -1, the default ASF value is used, or the current [ASF profile](#) value if an ASF profile is used.

See Also

[TNetworkStreamingType](#) [TOnClientConnection](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#) [ASFAudioBitRate](#) [ASFAudioChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#) [ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#) [ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#) [ASFProfileFromCustomFile](#) [ASFProfiles](#) [ASFProfilesCount](#) [ASFProfileVersion](#) [ASFStreaming](#) [GetAuthorizationList](#) [ASFStreaming](#) [GetConnectedClients](#) [ASFStreaming](#) [GetConnectedClientsCount](#) [ASFStreaming](#) [ResetAuthorizations](#) [ASFStreaming](#) [SetAuthorization](#) [ASFVideoBitRate](#) [ASFVideoFrameRate](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#) [NetworkStreaming](#) [NetworkStreamingType](#) [OnAuthenticationNeeded](#) [OnClientConnection](#) [OnDirectNetworkStreamingHostUrl](#) [ShowDialog](#) [StartAudioRendering](#) [StreamingURL](#)

Created with the Standard Edition of HelpNDoc: [Easy Qt Help documentation editor](#)

ASFVideoMaxKeyFrameSpacing

TVideoGrabber.ASFVideoMaxKeyFrameSpacing

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Maximum interval between key frames.

Declaration

property ASFVideoMaxKeyFrameSpacing: LongInt **read** GetASFVideoMaxKeyFrameSpacing **write** SetASFVideoMaxKeyFrameSpacing **default** DEF_ASFVideoMaxKeyFrameSpacing;

__property int ASFVideoMaxKeyFrameSpacing=read=GetASFVideoMaxKeyFrameSpacing, write=SetASFVideoMaxKeyFrameSpacing, **default**=- 1

Property ASFVideoMaxKeyFrameSpacing As Long

Description

Specifies the maximum interval between key frames for the current ASF video stream.

This is the maximum key-frame spacing in 100-nanosecond units.

E.g. 20000 = 1 key frame every 2 seconds.

If the specified value is -1, the default ASF value is used, or the current [ASF profile](#) value if an ASF profile is used.

See Also

[TNetworkStreamingType](#) [TOnClientConnection](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#) [ASFAudioBitRate](#) [ASFAudioChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#) [ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#) [ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#) [ASFProfileFromCustomFile](#) [ASFProfiles](#) [ASFProfilesCount](#) [ASFProfileVersion](#) [ASFStreaming](#) [GetAuthorizationList](#) [ASFStreaming](#) [GetConnectedClients](#) [ASFStreaming](#) [GetConnectedClientsCount](#) [ASFStreaming](#) [ResetAuthorizations](#) [ASFStreaming](#) [SetAuthorization](#) [ASFVideoBitRate](#) [ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoQuality](#) [ASFVideoWidth](#) [NetworkStreaming](#) [NetworkStreamingType](#) [OnAuthenticationNeeded](#) [OnClientConnection](#) [OnDirectNetworkStreamingHostUrl](#) [ShowDialog](#) [StartAudioRendering](#) [StreamingURL](#)

Created with the Standard Edition of HelpNDoc: [Easily create iPhone documentation](#)

ASFVideoQuality

TVideoGrabber.ASFVideoQuality

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Properties](#)

Quality setting for the video stream.

Declaration

property ASFVideoQuality: LongInt **read** GetASFVideoQuality **write** SetASFVideoQuality **default** DEF_ASFVideoQuality;

__property **int** ASFVideoQuality=read=GetASFVideoQuality, write=SetASFVideoQuality, **default**=- 1

Property ASFVideoQuality As Long

Description

Specifies the quality setting for the video stream, in the 0 ... 100 range (100 = maximum quality). If the specified value is -1, the default ASF value is used, or the current [ASF_profile](#) value if an ASF profile is used.

See Also

[TNetworkStreamingType](#) [TOnClientConnection](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#) [ASFAudioBitRate](#) [ASFAudioChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#) [ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#) [ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#) [ASFProfileFromCustomFile](#) [ASFProfiles](#) [ASFProfilesCount](#) [ASFProfileVersion](#) [ASFStreaming](#) [GetAuthorizationList](#) [ASFStreaming](#) [GetConnectedClients](#) [ASFStreaming](#) [GetConnectedClientsCount](#) [ASFStreaming](#) [ResetAuthorizations](#) [ASFStreaming](#) [SetAuthorization](#) [ASFVideoBitRate](#) [ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoWidth](#) [NetworkStreaming](#) [NetworkStreamingType](#) [OnAuthenticationNeeded](#) [OnClientConnection](#) [OnDirectNetworkStreamingHostUrl](#) [ShowDialog](#) [StartAudioRendering](#) [StreamingURL](#)

Created with the Standard Edition of HelpNDoc: Elevate Your Documentation with HelpNDoc's Project Analyzer Features

ASFVideoWidth

TVideoGrabber.ASFVideoWidth

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Properties](#)

Video width for the ASF video stream.

Declaration

property ASFVideoWidth: LongInt **read** GetASFVideoWidth **write** SetASFVideoWidth **default** DEF_ASFVideoQuality;

__property **int** ASFVideoWidth=read=GetASFVideoWidth, write=SetASFVideoWidth, **default**=- 1

Property ASFVideoWidth As Long

Description

Specifies the video width for the ASF video stream. If the specified value is -1, the default ASF value is used, or the current [ASF_profile](#) value if an ASF profile is used.

See Also

[TNetworkStreamingType](#) [TOnClientConnection](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#)

[ASFVideoMaxKeyFrameSpacing](#)
[ASFVideoQuality](#)
[NetworkStreaming](#)
[NetworkStreamingType](#)
[OnAuthenticationNeeded](#)
[OnClientConnection](#)
[OnDirectNetworkStreamingHostUrl](#)
[ShowDialog](#)
[StartAudioRendering](#)
[StreamingURL](#)

Created with the Standard Edition of HelpNDoc: [Make CHM Help File Creation a Breeze with HelpNDoc](#)

AspectRatioToUse

TVideoGrabber.AspectRatioToUse [Prev](#) [Next](#)

[TVideoGrabber](#) [Properties](#)

Forces the aspect ratio of the video clip played

Declaration

property AspectRatioToUse: Double **read** GetAspectRatioToUse **write** SetAspectRatioToUse **default** DEF_AspectRatioToUse;

__**property double** AspectRatioToUse=read=GetAspectRatioToUse, write=SetAspectRatioToUse, **default**=1

property AspectRatioToUse as Double

Description

Used to specify an aspect ratio for the video clip that will be played.

By default the proper aspect ratio is applied when the [AdjustPixelAspectRatio](#) is enabled.

It can be overridden by specifying an AspectRatioToUse value > 0 (e.g. 1.3333 for 4:3, 1666667 for 16:9, etc...)

Created with the Standard Edition of HelpNDoc: [Maximize Your Productivity with a Help Authoring Tool](#)

AssociateAudioAndVideoDevices

TVideoGrabber.AssociateAudioAndVideoDevices [Prev](#) [Next](#)

[TVideoGrabber](#) [Properties](#)

Used to associate the current audio and video capture devices.

Declaration

property AssociateAudioAndVideoDevices: Boolean **read** GetAssociateAudioAndVideoDevices **write** SetAssociateAudioAndVideoDevices **default** DEF_AssociateAudioAndVideoDevices;

__**property bool** AssociateAudioAndVideoDevices=read=GetAssociateAudioAndVideoDevices, write=SetAssociateAudioAndVideoDevices, **default**=0

Property AssociateAudioAndVideoDevices As Boolean

Description

When enabled:

- when selecting an audio capture device by assigning a value to [AudioDevice](#), this device, the current audio states (1) are associated to the current [video capture device](#).
- therefore, when selecting later this video capture device by assigning a value to the [VideoDevice](#)

property, the associated audio capture device, its audio input and the audio input values are automatically reloaded.

The audio properties associated are:

[AudioDevice](#)
[AudioInput](#)
[AudioInputLevel](#)
[SpeakerBalance](#)
[SpeakerVolume](#)

See Also

[TAudioFormat](#) [AudioBalance](#) [AudioDevice](#) [AudioDeviceIndex](#) [AudioDeviceName](#) [AudioDeviceRendering](#)
[AudioDevices](#) [AudioDevicesCount](#) [AudioFormat](#) [AudioFormats](#) [AudioInput](#) [AudioInputIndex](#) [AudioInputLevel](#)
[AudioInputMono](#) [AudioInputs](#) [AudioInputsCount](#) [AudioSource](#) [IsAudioCrossbarAvailable](#)
[IsAudioDeviceASoundCard](#) [IsAudioDeviceConnected](#) [IsAudioInputBalanceAvailable](#)
[OnAudioBufferNegotiation](#) [OnAudioDeviceSelected](#) [SpeakerBalance](#) [SpeakerControl](#) [SpeakerVolume](#)

Created with the Standard Edition of HelpNDoc: [Protect Your Confidential PDFs with These Simple Security Measures](#)

AudioBalance

TVideoGrabber.AudioBalance

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Audio left-right balance of the current [audio capture device](#) .

Declaration

property AudioBalance: LongInt **read** GetAudioBalance **write** SetAudioBalance;

___property **int** AudioBalance=read=GetAudioBalance, write=SetAudioBalance, **nodefault**

Property AudioBalance As Long

Description

Used to set/retrieve the audio left-right balance of the current [audio capture device](#) when [AudioDeviceRendering](#) is enabled.

The valid range is from -32767 to 32767.

0 selects the center point.

This a rendering audio property only, it does not apply to the audio recording.

See Also

[TVUMeter](#) [TVUMeterSetting](#) [AudioChannelRenderMode](#) [AudioVolume](#) [MuteAudioRendering](#)
[SpeakerBalance](#) [SpeakerControl](#) [SpeakerVolume](#) [VUMeter](#)

Created with the Standard Edition of HelpNDoc: [Bring your WinHelp HLP help files into the present with HelpNDoc's easy CHM conversion](#)

AudioChannelRenderMode

TVideoGrabber.AudioChannelRenderMode

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies how the left and/or right audio channels are rendered

Declaration

property AudioChannelRenderMode: TAudioChannelRenderMode **read** GetAudioChannelRenderMode
write SetAudioChannelRenderMode **default** acrm_Normal;

__property TAudioChannelRenderMode AudioChannelRenderMode=read=GetAudioChannelRenderMode, write=SetAudioChannelRenderMode, **default** 0

property AudioChannelRenderMode as TAudioChannelRenderMode

Description

Used to specify the way the right and left audio channels of the current audio stream are rendered

To be taken in account this property must be set before invoking [OpenPlayer](#), [StartPreview](#), [StartRecording](#), etc...

Possible values:

acrm_Normal (default) : normal rendering of the left and right audio channels

acrm_RenderLeft : the left channel is converted as "mono" and rendered on both sides (the right channel is discarded)

acrm_RenderRight : the right channel is converted as "mono" and rendered on both sides (the left channel is discarded)

acrm_MuteLeft : the left channel is muted

acrm_MuteRight : the right channel is muted

acrm_Mute_All : both channels are muted

acrm_MixLeftAndRight : the left and right channels are mixed

acrm_PassThru : like acrm_Normal, just activates the AudioChannelRenderMode processing

Remark: **acrm_PassThru** must be used to activate the AudioChannelRenderMode processing without modifying the audio samples immediately (e.g. to be able to mute the audio later with acrm_Mute_All), otherwise the AudioChannelRenderMode feature will be disabled if the preview, recording or playback starts with AudioChannelRenderMode = acrm_Normal.

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TVUMeter](#) [TVUMeterSetting](#) [TOnPlayerBufferingData](#) [AudioBalance](#) [AudioStreamNumber](#) [AudioVolume](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip](#) [Played](#) [MP4NeedsReindexing](#) [MuteAudioRendering](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchrone](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [SpeakerBalance](#) [SpeakerControl](#) [SpeakerVolume](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#) [VUMeter](#)

Created with the Standard Edition of HelpNDoc: Transform Your Documentation Process with HelpNDoc's Project Analyzer

AudioCompressor

TVideoGrabber.AudioCompressor

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Selects the current audio compressor.

Declaration

property AudioCompressor: LongInt **read** GetAudioCompressor **write** SetAudioCompressor;

__property int AudioCompressor=read=GetAudioCompressor, write=SetAudioCompressor, **nodefault**

Property AudioCompressor As Long

Description

Used to select the current audio compressor in the global [AudioCompressors](#) list.

See Also

[Recording methods and properties](#) [TCompressionType](#) [TOnVideoCompressionSettings](#) [AudioCompressorIndex](#) [AudioCompressorName](#) [AudioCompressors](#) [AudioCompressorsCount](#) [CompressionMode](#) [CompressionType](#) [GetVideoCompressionSettings](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [OnVideoCompressionSettings](#) [RefreshDevicesAndCompressorsLists](#) [SaveCompressorSettingsToDataString](#) [SetVideoCompressionDefaults](#) [SetVideoCompressionSettings](#) [VideoCompression](#) [KeyFrameRate](#) [VideoCompression](#) [PFramesPerKeyFrame](#) [VideoCompression](#) [Quality](#) [VideoCompression](#) [WindowSize](#) [VideoCompressor](#) [VideoCompressorIndex](#) [VideoCompressorName](#) [VideoCompressors](#) [VideoCompressorsCount](#)

Created with the Standard Edition of HelpNDoc: [Transform Your Word Doc into a Professional-Quality eBook with HelpNDoc](#)

AudioCompressorName

TVideoGrabber.AudioCompressorName

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Name of the current audio compressor.

Declaration

property AudioCompressorName: **string** read GetAudioCompressorName;

___property wchar_t *AudioCompressorName=read=GetAudioCompressorName

Property AudioCompressorName As String

Description

Used to retrieve the name of the audio compressor selected by [AudioCompressor](#) in the global [AudioCompressors](#) list.

See Also

[Recording methods and properties](#) [TCompressionType](#) [TOnVideoCompressionSettings](#) [AudioCompressor](#) [AudioCompressorIndex](#) [AudioCompressors](#) [AudioCompressorsCount](#) [CompressionMode](#) [CompressionType](#) [GetVideoCompressionSettings](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [OnVideoCompressionSettings](#) [RefreshDevicesAndCompressorsLists](#) [SaveCompressorSettingsToDataString](#) [SetVideoCompressionDefaults](#) [SetVideoCompressionSettings](#) [VideoCompression](#) [KeyFrameRate](#) [VideoCompression](#) [PFramesPerKeyFrame](#) [VideoCompression](#) [Quality](#) [VideoCompression](#) [WindowSize](#) [VideoCompressor](#) [VideoCompressorIndex](#) [VideoCompressorName](#) [VideoCompressors](#) [VideoCompressorsCount](#)

Created with the Standard Edition of HelpNDoc: [Full-featured Kindle eBooks generator](#)

AudioCompressors

TVideoGrabber.AudioCompressors

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

List of the video compressors (audio codecs) available on the current platform.

Declaration

function AudioCompressors: **string**;

```
__property wchar_t * __fastcall AudioCompressors();
```

Property AudioCompressors as String

Description

Used to retrieve a string that contains the list of the audio compressors (audio codecs) available on the current platform.

This list can be assigned to list based controls. E.g.:

```
ComboBox1.Items.Text := VideoGrabber1.AudioCompressors;
```

```
ComboBox1.ItemIndex := VideoGrabber1.AudioCompressor;
```

It is possible to retrieve programmatically the index of a Audio compressor by using the [FindIndexInListByName](#) function as follows:

```
VideoGrabber.AudioCompressor := AudioGrabber.FindIndexInListByName (VideoGrabber.AudioCompressor, VideoGrabber.AudioCompressors[0]);
```

Note:

when the application starts the [VideoCompressors](#) and [AudioCompressors](#) lists are populated with the codecs currently installed. To refresh the compressor lists after installing/uninstalling codecs without exiting/restarting your application, invoke [RefreshDevicesAndCompressorsLists](#).

See Also

[Recording methods and properties TCompressionType TOnVideoCompressionSettings AudioCompressor AudioCompressorIndex AudioCompressorName AudioCompressorsCount CompressionMode CompressionType GetVideoCompressionSettings OnReencodingCompleted OnReencodingStarted OnVideoCompressionSettings RefreshDevicesAndCompressorsLists SaveCompressorSettingsToDataString SetVideoCompressionDefaults SetVideoCompressionSettings VideoCompression KeyFrameRate VideoCompression PFramesPerKeyFrame VideoCompression Quality VideoCompression WindowSize VideoCompressor VideoCompressorIndex VideoCompressorName VideoCompressors VideoCompressorsCount](#)

Created with the Standard Edition of HelpNDoc: Free Kindle producer

AudioCompressorsCount

TVideoGrabber.AudioCompressorsCount

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Number of audio compressors (codecs) available on the current platform.

Declaration

```
property AudioCompressorsCount: LongInt read GetGlobal_AudioCompressorsCount;
```

```
__property int AudioCompressorsCount=read=GetGlobal_AudioCompressorsCount, nodefault
```

Function AudioCompressorsCount as Long

Description

Number of audio compressors (codecs) in the [AudioCompressors](#) list.

Note: in Delphi and C++Builder versions of the component, this property is available also as a [AudioCompressorsCount](#) global variable.

See Also

[Recording methods and properties TCompressionType TOnVideoCompressionSettings AudioCompressor AudioCompressorIndex AudioCompressorName AudioCompressors CompressionMode CompressionType GetVideoCompressionSettings OnReencodingCompleted OnReencodingStarted OnVideoCompressionSettings RefreshDevicesAndCompressorsLists SaveCompressorSettingsToDataString SetVideoCompressionDefaults SetVideoCompressionSettings VideoCompression KeyFrameRate VideoCompression PFramesPerKeyFrame VideoCompression Quality VideoCompression WindowSize VideoCompressor VideoCompressorIndex VideoCompressorName VideoCompressors VideoCompressorsCount](#)

AudioDevice

TVideoGrabber.AudioDevice

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Selects the current audio capture device.

Declaration

property AudioDevice: LongInt **read** GetAudioDevice **write** SetAudioDevice **default** DEF_AudioDevice;

__property **int** AudioDevice=read=GetAudioDevice, write=SetAudioDevice, **default**=0

Property AudioDevice As Long

Description

Used to select the current audio capture device in the [AudioDevices](#) list.

When a new value is assigned to this property, the related [device-dependent values](#) are loaded from the registry and the [OnAudioDeviceSelected](#) event occurs.

It is possible to select the audio capture device programmatically by its name (as it appears in the [AudioDevices](#) list) by using the [FindIndexInListByName](#) function, e.g.:

```
procedure TfrmMainForm.Button1Click(Sender: TObject);
var
    i: LongInt;
begin
    i := VideoGrabber.FindIndexInListByName (VideoGrabber.AudioDevices, 'Realtek AC97 Au
    if i > -1 then begin // if this Device exists...
        VideoGrabber.AudioDevice := i;
    end;
end;
end;
```

See Also

[TAudioFormat](#) [AssociateAudioAndVideoDevices](#) [AudioBalance](#) [AudioDeviceIndex](#) [AudioDeviceName](#) [AudioDeviceRendering](#) [AudioDevices](#) [AudioDevicesCount](#) [AudioFormat](#) [AudioFormats](#) [AudioInput](#) [AudioInputIndex](#) [AudioInputLevel](#) [AudioInputMono](#) [AudioInputs](#) [AudioInputsCount](#) [AudioSource](#) [IsAudioCrossbarAvailable](#) [IsAudioDeviceASoundCard](#) [IsAudioDeviceConnected](#) [IsAudioInputBalanceAvailable](#) [OnAudioBufferNegotiation](#) [OnAudioDeviceSelected](#) [SpeakerBalance](#) [SpeakerControl](#) [SpeakerVolume](#)

AudioDeviceName

TVideoGrabber.AudioDeviceName

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Name of the current audio capture device.

Declaration

property AudioDeviceName: **string** **read** GetAudioDeviceName;

__property wchar_t *AudioDeviceName=read=GetAudioDeviceName

Property AudioDeviceName As String

Description

Used to retrieve the name of the audio capture device selected by [AudioDevice](#) in the [AudioDevices](#) list.

See Also

[TAudioFormat](#) [AssociateAudioAndVideoDevices](#) [AudioBalance](#) [AudioDevice](#) [AudioDeviceIndex](#) [AudioDeviceRendering](#) [AudioDevices](#) [AudioDevicesCount](#) [AudioFormat](#) [AudioFormats](#) [AudioInput](#) [AudioInputIndex](#) [AudioInputLevel](#) [AudioInputMono](#) [AudioInputs](#) [AudioInputsCount](#) [AudioSource](#) [IsAudioCrossbarAvailable](#) [IsAudioDeviceASoundCard](#) [IsAudioDeviceConnected](#) [IsAudioInputBalanceAvailable](#) [OnAudioBufferNegotiation](#) [OnAudioDeviceSelected](#) [SpeakerBalance](#) [SpeakerControl](#) [SpeakerVolume](#)

Created with the Standard Edition of HelpNDoc: [Easily share your documentation with the world through a beautiful website](#)

AudioDeviceRendering

TVideoGrabber.AudioDeviceRendering

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

If enabled, the audio stream is rendered during preview or recording.

Declaration

property AudioDeviceRendering: Boolean **read** GetAudioDeviceRendering **write** SetAudioDeviceRendering **default** DEF_AudioDeviceRendering;

__property **bool** AudioDeviceRendering=read=GetAudioDeviceRendering, write=SetAudioDeviceRendering, **default**=0

Property AudioDeviceRendering As Boolean

Description

If enabled, the current [AudioDevice](#) audio capture device is added to the preview or recording graph and the audio stream is rendered on the default sound device.

See Also

[TAudioFormat](#) [AssociateAudioAndVideoDevices](#) [AudioBalance](#) [AudioDevice](#) [AudioDeviceIndex](#) [AudioDeviceName](#) [AudioDevices](#) [AudioDevicesCount](#) [AudioFormat](#) [AudioFormats](#) [AudioInput](#) [AudioInputIndex](#) [AudioInputLevel](#) [AudioInputMono](#) [AudioInputs](#) [AudioInputsCount](#) [AudioSource](#) [IsAudioCrossbarAvailable](#) [IsAudioDeviceASoundCard](#) [IsAudioDeviceConnected](#) [IsAudioInputBalanceAvailable](#) [OnAudioBufferNegotiation](#) [OnAudioDeviceSelected](#) [SpeakerBalance](#) [SpeakerControl](#) [SpeakerVolume](#)

Created with the Standard Edition of HelpNDoc: [Revolutionize Your Documentation Review with HelpNDoc's Project Analyzer](#)

AudioDevices

TVideoGrabber.AudioDevices

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

List of the audio capture devices available on the current platform.

Declaration

function AudioDevices: **string**;

```
__property wchar_t * __fastcall AudioDevices();
```

Property AudioDevices as string

Description

Used to retrieve a string that contains the list of audio capture devices available on the current platform. This list is updated when an audio capture device is connected or removed (when the [OnDeviceArrivalOrRemoval](#) event occurs).

This list can be assigned to list based controls. E.g.:

```
ComboBox1.Items.Text := AudioDevices;
```

```
ComboBox1.ItemIndex := TVideoGrabber1.AudioDevice; (* index in the AudioDevices list. *)
```

It is possible to retrieve programmatically the index of an audio capture device by using the [FindIndexInListByName](#) function as follows:

```
TVideoGrabber.AudioDevice := TVideoGrabber.FindIndexInListByName (TVideoGrabber.AudioDevices
```

Note: in Delphi and C++Builder versions of the component, this property is available also as a [AudioDevices](#) global variable.

See Also

[TAudioFormat](#) [AssociateAudioAndVideoDevices](#) [AudioBalance](#) [AudioDevice](#) [AudioDeviceIndex](#) [AudioDeviceName](#) [AudioDeviceRendering](#) [AudioDevicesCount](#) [AudioFormat](#) [AudioFormats](#) [AudioInput](#) [AudioInputIndex](#) [AudioInputLevel](#) [AudioInputMono](#) [AudioInputs](#) [AudioInputsCount](#) [AudioSource](#) [IsAudioCrossbarAvailable](#) [IsAudioDeviceASoundCard](#) [IsAudioDeviceConnected](#) [IsAudioInputBalanceAvailable](#) [OnAudioBufferNegotiation](#) [OnAudioDeviceSelected](#) [SpeakerBalance](#) [SpeakerControl](#) [SpeakerVolume](#)

Created with the Standard Edition of HelpNDoc: [Make CHM Help File Creation a Breeze with HelpNDoc](#)

AudioDevicesCount

TVideoGrabber.AudioDevicesCount

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Number of audio capture devices available on the current platform.

Declaration

```
property AudioDevicesCount: LongInt read GetGlobal_AudioDevicesCount;
```

```
__property int AudioDevicesCount=read=GetGlobal_AudioDevicesCount, nodefault
```

Function AudioDevicesCount as Long

Description

Number of audio capture devices in the [AudioDevices](#) list.

Note: in Delphi and C++Builder versions of the component, this property is available also as a [AudioDevicesCount](#) global variable.

See Also

[TAudioFormat](#) [AssociateAudioAndVideoDevices](#) [AudioBalance](#) [AudioDevice](#) [AudioDeviceIndex](#) [AudioDeviceName](#) [AudioDeviceRendering](#) [AudioDevices](#) [AudioFormat](#) [AudioFormats](#) [AudioInput](#) [AudioInputIndex](#) [AudioInputLevel](#) [AudioInputMono](#) [AudioInputs](#) [AudioInputsCount](#) [AudioSource](#) [IsAudioCrossbarAvailable](#) [IsAudioDeviceASoundCard](#) [IsAudioDeviceConnected](#) [IsAudioInputBalanceAvailable](#) [OnAudioBufferNegotiation](#) [OnAudioDeviceSelected](#) [SpeakerBalance](#) [SpeakerControl](#) [SpeakerVolume](#)

Created with the Standard Edition of HelpNDoc: [What is a Help Authoring tool?](#)

AudioFormat

TVideoGrabber.AudioFormat

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Properties](#)

Selects an audio format.

Declaration

property AudioFormat: TAudioFormat **read** GetAudioFormat **write** SetAudioFormat **default** DEF_AudioFormat;

__property TAudioFormat AudioFormat=read=GetAudioFormat, write=SetAudioFormat, **default**=0

Property AudioFormat As TxAudioFormat

Description

Used to select an audio format in the [AudioFormats](#) list.

See Also

[TAudioFormat](#) [AssociateAudioAndVideoDevices](#) [AudioBalance](#) [AudioDevice](#) [AudioDeviceIndex](#) [AudioDeviceName](#) [AudioDeviceRendering](#) [AudioDevices](#) [AudioDevicesCount](#) [AudioFormats](#) [AudioInput](#) [AudioInputIndex](#) [AudioInputLevel](#) [AudioInputMono](#) [AudioInputs](#) [AudioInputsCount](#) [AudioSource](#) [IsAudioCrossbarAvailable](#) [IsAudioDeviceASoundCard](#) [IsAudioDeviceConnected](#) [IsAudioInputBalanceAvailable](#) [OnAudioBufferNegotiation](#) [OnAudioDeviceSelected](#) [SpeakerBalance](#) [SpeakerControl](#) [SpeakerVolume](#)

Created with the Standard Edition of HelpNDoc: Streamline your documentation process with HelpNDoc's WinHelp HLP to CHM conversion feature

AudioFormats

TVideoGrabber.AudioFormats

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Properties](#)

List of audio formats.

Declaration

property AudioFormats: **string** **read** GetAudioFormats;

__property wchar_t *AudioFormats=read=GetAudioFormats

Property AudioFormats As String

Description

List of audio formats that can be used with the [current audio capture device](#) .

See Also

[TAudioFormat](#) [AssociateAudioAndVideoDevices](#) [AudioBalance](#) [AudioDevice](#) [AudioDeviceIndex](#) [AudioDeviceName](#) [AudioDeviceRendering](#) [AudioDevices](#) [AudioDevicesCount](#) [AudioFormat](#) [AudioInput](#) [AudioInputIndex](#) [AudioInputLevel](#) [AudioInputMono](#) [AudioInputs](#) [AudioInputsCount](#) [AudioSource](#) [IsAudioCrossbarAvailable](#) [IsAudioDeviceASoundCard](#) [IsAudioDeviceConnected](#) [IsAudioInputBalanceAvailable](#) [OnAudioBufferNegotiation](#) [OnAudioDeviceSelected](#) [SpeakerBalance](#) [SpeakerControl](#) [SpeakerVolume](#)

Created with the Standard Edition of HelpNDoc: Streamline Your Documentation Process with HelpNDoc's Intuitive Interface

AudioInput

TVideoGrabber.AudioInput

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Properties](#)

Selects the current audio input.

Declaration

property AudioInput: LongInt **read** GetAudioInput **write** SetAudioInput;

__property **int** AudioInput=read=GetAudioInput, write=SetAudioInput, **nodefault**

Property AudioInput As Long

Description

Used to select/retrieve the current audio input in the [AudioInputs](#) list.

See Also

[TAudioFormat](#) [AssociateAudioAndVideoDevices](#) [AudioBalance](#) [AudioDevice](#) [AudioDeviceIndex](#) [AudioDeviceName](#) [AudioDeviceRendering](#) [AudioDevices](#) [AudioDevicesCount](#) [AudioFormat](#) [AudioFormats](#) [AudioInputIndex](#) [AudioInputLevel](#) [AudioInputMono](#) [AudioInputs](#) [AudioInputsCount](#) [AudioSource](#) [IsAudioCrossbarAvailable](#) [IsAudioDeviceASoundCard](#) [IsAudioDeviceConnected](#) [IsAudioInputBalanceAvailable](#) [OnAudioBufferNegotiation](#) [OnAudioDeviceSelected](#) [SpeakerBalance](#) [SpeakerControl](#) [SpeakerVolume](#)

Created with the Standard Edition of HelpNDoc: [Enhance Your Documentation with HelpNDoc's Advanced Project Analyzer](#)

AudioInputBalance

TVideoGrabber.AudioInputBalance

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Properties](#)

Sets/retrieves the audio input balance.

Declaration

property AudioInputBalance: LongInt **read** GetAudioInputBalance **write** SetAudioInputBalance;

__property **int** AudioInputBalance=read=GetAudioInputBalance, write=SetAudioInputBalance, **nodefault**

Property AudioInputBalance As Long

Description

Used to set/retrieve the audio input balance of the current [audio capture device](#).
The valid range is -32768...32767.
0 is the center point.

Created with the Standard Edition of HelpNDoc: [Transform Your Help Documentation Process with a Help Authoring Tool](#)

AudioInputLevel

TVideoGrabber.AudioInputLevel

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Properties](#)

Sets/retrieves the audio input level.

Declaration

property AudioInputLevel: LongInt **read** GetAudioInputLevel **write** SetAudioInputLevel;

__property **int** AudioInputLevel=read=GetAudioInputLevel, write=SetAudioInputLevel, **nodefault**

Property AudioInputLevel As Long

Description

Used to set/retrieve the audio input level of the current [audio capture device](#).

The valid range is from 0 to 65535.

0 indicates that the recording level is off; the value 65535 indicates that the recording level is at full volume.

See Also

[TAudioFormat](#) [AssociateAudioAndVideoDevices](#) [AudioBalance](#) [AudioDevice](#) [AudioDeviceIndex](#) [AudioDeviceName](#) [AudioDeviceRendering](#) [AudioDevices](#) [AudioDevicesCount](#) [AudioFormat](#) [AudioFormats](#) [AudioInput](#) [AudioInputIndex](#) [AudioInputMono](#) [AudioInputs](#) [AudioInputsCount](#) [AudioSource](#) [IsAudioCrossbarAvailable](#) [IsAudioDeviceASoundCard](#) [IsAudioDeviceConnected](#) [IsAudioInputBalanceAvailable](#) [OnAudioBufferNegotiation](#) [OnAudioDeviceSelected](#) [SpeakerBalance](#) [SpeakerControl](#) [SpeakerVolume](#)

Created with the Standard Edition of HelpNDoc: Full-featured Help generator

AudioInputMono**TVideoGrabber.AudioInputMono**

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Disables the stereo mode of the audio input.

Declaration

property AudioInputMono: Boolean **read** GetAudioInputMono **write** SetAudioInputMono;

__property **bool** AudioInputMono=read=GetAudioInputMono, write=SetAudioInputMono, **nodefault**

Property AudioInputMono As Boolean

Description

Used to switch the audio input in "mono" mode of the current [audio capture device](#).

When disabled, the audio input is in "stereo" mode.

See Also

[TAudioFormat](#) [AssociateAudioAndVideoDevices](#) [AudioBalance](#) [AudioDevice](#) [AudioDeviceIndex](#) [AudioDeviceName](#) [AudioDeviceRendering](#) [AudioDevices](#) [AudioDevicesCount](#) [AudioFormat](#) [AudioFormats](#) [AudioInput](#) [AudioInputIndex](#) [AudioInputLevel](#) [AudioInputs](#) [AudioInputsCount](#) [AudioSource](#) [IsAudioCrossbarAvailable](#) [IsAudioDeviceASoundCard](#) [IsAudioDeviceConnected](#) [IsAudioInputBalanceAvailable](#) [OnAudioBufferNegotiation](#) [OnAudioDeviceSelected](#) [SpeakerBalance](#) [SpeakerControl](#) [SpeakerVolume](#)

Created with the Standard Edition of HelpNDoc: Maximize Your Documentation Output with HelpNDoc's Advanced Project Analyzer

AudioInputs**TVideoGrabber.AudioInputs**

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

List of audio inputs available.

Declaration

property AudioInputs: **string read** GetAudioInputs;

__property wchar_t *AudioInputs=read=GetAudioInputs

Property AudioInputs As String

Description

Used to retrieve the list of the audio inputs available on the [current audio capture device](#) .

This list is updated when the [OnAudioDeviceSelected](#) event occurs (when an audio capture device is selected with [AudioDevice](#)).

This list can be assigned to list based controls. E.g.:

ComboBox1.Items.Text := VideoGrabber1.AudioInputs;

ComboBox1.ItemIndex := VideoGrabber1.AudioInput;

See Also

[TAudioFormat](#) [AssociateAudioAndVideoDevices](#) [AudioBalance](#) [AudioDevice](#) [AudioDeviceIndex](#) [AudioDeviceName](#) [AudioDeviceRendering](#) [AudioDevices](#) [AudioDevicesCount](#) [AudioFormat](#) [AudioFormats](#) [AudioInput](#) [AudioInputIndex](#) [AudioInputLevel](#) [AudioInputMono](#) [AudioInputsCount](#) [AudioSource](#) [IsAudioCrossbarAvailable](#) [IsAudioDeviceASoundCard](#) [IsAudioDeviceConnected](#) [IsAudioInputBalanceAvailable](#) [OnAudioBufferNegotiation](#) [OnAudioDeviceSelected](#) [SpeakerBalance](#) [SpeakerControl](#) [SpeakerVolume](#)

Created with the Standard Edition of HelpNDoc: Effortlessly Create High-Quality Documentation with a Help Authoring Tool

AudioInputsCount

TVideoGrabber.AudioInputsCount

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Number of audio inputs.

Declaration

property AudioInputsCount: LongInt **read** GetAudioInputsCount;

__property **int** AudioInputsCount=read=GetAudioInputsCount, **nodefault**

Property AudioInputsCount As Long

Description

Number of audio inputs in the [AudioInputs](#) list.

The AudioInput index (used to select the current audio input) is in the **0 .. AudioInputsCount - 1** range.

See Also

[TAudioFormat](#) [AssociateAudioAndVideoDevices](#) [AudioBalance](#) [AudioDevice](#) [AudioDeviceIndex](#) [AudioDeviceName](#) [AudioDeviceRendering](#) [AudioDevices](#) [AudioDevicesCount](#) [AudioFormat](#) [AudioFormats](#) [AudioInput](#) [AudioInputIndex](#) [AudioInputLevel](#) [AudioInputMono](#) [AudioInputs](#) [AudioSource](#) [IsAudioCrossbarAvailable](#) [IsAudioDeviceASoundCard](#) [IsAudioDeviceConnected](#) [IsAudioInputBalanceAvailable](#) [OnAudioBufferNegotiation](#) [OnAudioDeviceSelected](#) [SpeakerBalance](#) [SpeakerControl](#) [SpeakerVolume](#)

Created with the Standard Edition of HelpNDoc: Free CHM Help documentation generator

AudioPeakEvent

TVideoGrabber.AudioPeakEvent

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Activates the [OnAudioPeak](#) event.

Declaration

property AudioPeakEvent: Boolean **read** GetAudioPeakEvent **write** SetAudioPeakEvent;

__property **bool** AudioPeakEvent=read=GetAudioPeakEvent, write=SetAudioPeakEvent, **default false**

Property AudioPeakEvent As Boolean

Description

Used to activate the [OnAudioPeak](#) event, that returns PCM audio peaks.

Created with the Standard Edition of HelpNDoc: [Easily create Qt Help files](#)

AudioRecording

TVideoGrabber.AudioRecording

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables/disables audio stream capture during [recording](#) .

Declaration

property AudioRecording: Boolean **read** GetAudioRecording **write** SetAudioRecording;

__property **bool** AudioRecording=read=GetAudioRecording, write=SetAudioRecording, **nodefault**

Property AudioRecording As Boolean

Description

Used to enable/disable audio stream capture during [recording](#) .
If disabled, only the video stream is be captured.

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last](#) [Recording](#) [FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [Converting Word Documents to eBooks: A Step-by-Step Guide with HelpNDoc](#)

AudioRenderer

TVideoGrabber.AudioRenderer

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Selects the current audio renderer.

Declaration

property AudioRenderer: LongInt **read** GetAudioRenderer **write** SetAudioRenderer **default** DEF_AudioRenderer;

__**property** int AudioRenderer==GetAudioRenderer, write=SetAudioRenderer, **default**=-1;

Description

Used to select the current audio capture device in the [AudioRenderers](#) list.

The default value -1 uses the default renderer.

It is possible to select the audio renderer programmatically by its name (as it appears in the [AudioRenderers](#) list) by using the [FindIndexInListByName](#) function, e.g.:

```
procedure TfrmMainForm.Button1Click(Sender: TObject);
var
  i: LongInt;
begin
  i := VideoGrabber.FindIndexInListByName (VideoGrabber.AudioRenderers, 'AC97', true,
  if i > -1 then begin
    VideoGrabber.AudioRenderers := i;
  end;
end;
end;
```

Created with the Standard Edition of HelpNDoc: [Easy to use tool to create HTML Help files and Help web sites](#)

AudioRendererName**TVideoGrabber.AudioRendererName**[Prev](#)[Next](#)

[TVideoGrabber](#) [Properties](#)

Name of the current audio renderer.

Declaration

property AudioRendererName: **string** **read** GetAudioRendererName;

__**property** wchar_t *AudioRendererName==GetAudioRendererName;

Description

Used to retrieve the name of the audio capture device selected by [AudioRenderer](#) in the [AudioRenderers](#) list.

Created with the Standard Edition of HelpNDoc: [Don't Let Unauthorized Users View Your PDFs: Learn How to Set Passwords](#)

AudioRenderers**TVideoGrabber.AudioRenderers**[Prev](#)[Next](#)

[TVideoGrabber](#) [Properties](#)

List of the audio renderers available on the current platform.

Declaration

property AudioRenderers: **string** **read** GetGlobal_AudioRenderers;

```
__property wchar_t *AudioRenderers==GetGlobal_AudioRenderers;
```

Description

Used to retrieve a string that contains the list of the audio renderers available on the current platform.

This list can be assigned to list based controls. E.g.:

```
ComboBox1.Items.Text := AudioRenderers;
```

```
ComboBox1.ItemIndex := TVideoGrabber1.AudioRenderer; (* index in the AudioRenderers list. *)
```

It is possible to retrieve programmatically the index of an audio renderer by using the [FindIndexInListByName](#) function as follows:

```
TVideoGrabber.AudioRenderer := TVideoGrabber.FindIndexInListByName (TVideoGrabber.AudioRenderers, AudioRendererName);
```

Note: in Delphi and C++Builder versions of the component, this property is available also as a [AudioRenderers](#) global variable.

Created with the Standard Edition of HelpNDoc: [Say Goodbye to Documentation Headaches with a Help Authoring Tool](#)

AudioRenderersCount

TVideoGrabber.AudioRenderersCount

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Number of audio renderers available on the current platform

Declaration

```
property AudioRenderersCount: LongInt read GetGlobal_AudioRenderersCount;
```

```
__property int AudioRenderersCount==GetGlobal_AudioRenderersCount, ndefault};
```

Description

Number of audio renderers in the [AudioRenderers](#) list.

Note: in Delphi and C++Builder versions of the component, this property is available also as a [AudioRenderersCount](#) global variable.

Created with the Standard Edition of HelpNDoc: [Create cross-platform Qt Help files](#)

AudioSource

TVideoGrabber.AudioSource

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Source used for the audio recording.

Declaration

```
property AudioSource: TAudioSource read GetAudioSource write SetAudioSource default DEF_AudioSource;
```

```
__property TAudioSource AudioSource=read=GetAudioSource, write=SetAudioSource, default=0
```

Property AudioSource as TxAudioSource

Description

Specifies the source used for the audio recording:

as_Default:
uses the audio output of the [video capture device](#) if any, otherwise uses the current [audio capture device](#).
always uses the current [audio capture device](#) for the audio recording, even if the video capture device exposes an audio output.

as_UseExternalAudio:

See Also

[TAudioFormat](#) [AssociateAudioAndVideoDevices](#) [AudioBalance](#) [AudioDevice](#) [AudioDeviceIndex](#) [AudioDeviceName](#) [AudioDeviceRendering](#) [AudioDevices](#) [AudioDevicesCount](#) [AudioFormat](#) [AudioFormats](#) [AudioInput](#) [AudioInputIndex](#) [AudioInputLevel](#) [AudioInputMono](#) [AudioInputs](#) [AudioInputsCount](#) [IsAudioCrossbarAvailable](#) [IsAudioDeviceASoundCard](#) [IsAudioDeviceConnected](#) [IsAudioInputBalanceAvailable](#) [OnAudioBufferNegotiation](#) [OnAudioDeviceSelected](#) [SpeakerBalance](#) [SpeakerControl](#) [SpeakerVolume](#)

Created with the Standard Edition of HelpNDoc: [Powerful and User-Friendly Help Authoring Tool for Markdown Documents](#)

AudioStreamNumber

TVideoGrabber.AudioStreamNumber

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the audio stream to use

Declaration

property AudioStreamNumber: LongInt **read** GetAudioStreamNumber **write** SetAudioStreamNumber **default** -1;

__property int AudioStreamNumber==GetAudioStreamNumber, write=SetAudioStreamNumber, **default**== -1;

property AudioStreamNumber as Long

Description

If the clip contains more than one audio stream, use this property to select the audio stream to render

E.g.:

AudioStreamNumber = -1 (default) -> all the audio streams are rendered

AudioStreamNumber = 0 -> only the 1st audio stream is rendered

AudioStreamNumber = 1 -> only the 2nd audio stream is rendered

...

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchron](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Make Documentation a Breeze with a Help Authoring](#)

AudioSyncAdjustment

TVideoGrabber.AudioSyncAdjustment

[Prev](#)[Next](#)

[TVideoGrabber](#) [Properties](#)

Introduces an audio delay.

Declaration

property AudioSyncAdjustment: LongInt **read** GetAudioSyncAdjustment **write** SetAudioSyncAdjustment
default DEF_AudioSyncAdjustment;

__property int AudioSyncAdjustment==GetAudioSyncAdjustment, write=SetAudioSyncAdjustment,
default=0;

property AudioSyncAdjustment as Long

Description

Used to introduce an audio delay, when the video and audio are not synchronized.
 Useful mainly when the audio and video streams do not come out from the same capture device.
 This property specifies a number of audio samples that are delayed, in order to get the audio and video in sync.

To use this property:

- enable [AudioSyncAdjustmentEnabled](#)
- set AudioSyncAdjustment with a number of audio samples to delay, greater than 0

You can make a quick test of this feature with MainDemo.exe -> "audio tab"

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last_Recording_FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [5 Reasons Why a Help Authoring Tool is Better than Microsoft Word for Documentation](#)

AudioSyncAdjustmentEnabled

TVideoGrabber.AudioSyncAdjustmentEnabled

[Prev](#)[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables the audio sync adjustment.

Declaration

property AudioSyncAdjustmentEnabled: Boolean **read** GetAudioSyncAdjustmentEnabled **write**

SetAudioSyncAdjustmentEnabled **default** DEF_AudioSyncAdjustmentEnabled;

__**property bool** AudioSyncAdjustmentEnabled==GetAudioSyncAdjustmentEnabled,
write=SetAudioSyncAdjustmentEnabled, **default**=0;

property AudioSyncAdjustmentEnabled as Boolean

Description

Used to enable the audio sync adjustment.
See [AudioSyncAdjustment](#).

Created with the Standard Edition of HelpNDoc: [Transform Your Word Document into a Professional eBook with HelpNDoc](#)

AudioVolume

TVideoGrabber.AudioVolume

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Audio volume of the current [audio capture device](#) .

Declaration

property AudioVolume: LongInt **read** GetAudioVolume **write** SetAudioVolume;

__property **int** AudioVolume=read=GetAudioVolume, write=SetAudioVolume, **nodefault**

Property AudioVolume As Long

Description

Used to set/retrieve the audio volume of the current [audio capture device](#) when [AudioDeviceRendering](#) is enabled.

The valid range is from 0 (min level) to 65535 (max level).

This is a rendering audio property only, it does not apply to the audio recording.

See Also

[TVUMeter](#) [TVUMeterSetting](#) [AudioBalance](#) [AudioChannelRenderMode](#) [MuteAudioRendering](#)
[SpeakerBalance](#) [SpeakerControl](#) [SpeakerVolume](#) [VUMeter](#)

Created with the Standard Edition of HelpNDoc: [Easily convert your WinHelp HLP help files to CHM with HelpNDoc's step-by-step guide](#)

AutoFileName

TVideoGrabber.AutoFileName

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Format of file names generated automatically.

Declaration

property AutoFileName: TAutoFileName **read** GetAutoFileName **write** SetAutoFileName **default**
DEF_AutoFileName;

__property TAutoFileName AutoFileName=read=GetAutoFileName, write=SetAutoFileName, **default**=0

Property AutoFileName As TAutoFileName

Description

Sets or retrieves the format (sequential or date/time) used to generate file names automatically during

[frame capture](#) or [recording](#) .

See Also

[AutoFileNameDateTimeFormat](#) [AutoFilePrefix](#) [CaptureFileExt](#) [Last_BurstFrameCapture_FileName](#)
[Last_CaptureFrameTo_FileName](#) [RecordingFileName](#)

Created with the Standard Edition of HelpNDoc: [Create iPhone web-based documentation](#)

AutoFileNameDateTimeFormat

TVideoGrabber.AutoFileNameDateTimeFormat

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specify the date/time format for recording file name

Declaration

property AutoFileNameDateTimeFormat: **string** **read** GetAutoFileNameDateTimeFormat **write** SetAutoFileNameDateTimeFormat;

__**property** wchar_t *AutoFileNameDateTimeFormat=read=GetAutoFileNameDateTimeFormat, write=SetAutoFileNameDateTimeFormat

Property AutoFileNameDateTimeFormat As String

Description

Lets customize the date/time format when the recording file names are generated automatically and [AutoFileName](#) is set to **fn_DateTime**

By default the date/time format of the file names is:

yymmdd_hhmmss_zzz

By default [AutoFilePrefix](#) is set to "vg" (default), so the recording file name is e.g.

vg130213_151847_352.avi

By example if you customize it as follows:

```
VideoGrabber.AutoFileName = fn_DateTime;
VideoGrabber.AutoFileNameDateTimeFormat = "dd-mm-yy_hh-mm-ss"
VideoGrabber.AutoFilePrefix = "file_"
```

the recording file name will be **file_13-02-13_15-27-45.avi**

See Also

[AutoFileName](#) [AutoFilePrefix](#) [CaptureFileExt](#) [Last_BurstFrameCapture_FileName](#)
[Last_CaptureFrameTo_FileName](#) [RecordingFileName](#)

Created with the Standard Edition of HelpNDoc: [Single source CHM, PDF, DOC and HTML Help creation](#)

AutoFileNameMinDigits

TVideoGrabber.AutoFileNameMinDigits

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the number of 0 to pad auto-generated file names.

Declaration

property AutoFileNameMinDigits: **LongInt** **read** GetAutoFileNameMinDigits **write** SetAutoFileNameMinDigits **default** DEF_AutoFileNameMinDigits;

__**property** **int** AutoFileNameMinDigits=read=GetAutoFileNameMinDigits, write=SetAutoFileNameMinDigits,

default=6

Property AutoFileNameMinDigits As Long

Description

Used to specify the number of padding "0" used to generate a file name when [AutoFileName](#) = [fn_Sequential](#).

E.g.:

- if AutoFileNameMinDigits = 6, the file name will be generated e.g. as

vg000001.avi

vg000002.avi

vg000003.avi

...

- if AutoFileNameMinDigits = 3, the file name will be generated e.g. as

vg001.avi

vg002.avi

vg003.avi

...

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last_Recording_FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [Enhance Your Documentation with HelpNDoc's Advanced Project Analyzer](#)

AutoFilePrefix

TVideoGrabber.AutoFilePrefix

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Prefix of file names generated automatically.

Declaration

property AutoFilePrefix: **string** **read** GetAutoFilePrefix **write** SetAutoFilePrefix;

__property wchar_t *AutoFilePrefix=read=GetAutoFilePrefix, write=SetAutoFilePrefix

Property AutoFilePrefix As String

Description

Sets or retrieves the prefix added at the beginning of file names generated automatically during [frame capture](#) or [recording](#) .

See Also

[AutoFileName](#) [AutoFileNameDateTimeFormat](#) [CaptureFileExt](#) [Last](#) [BurstFrameCapture](#) [FileName](#)
[Last](#) [CaptureFrameTo](#) [FileName](#) [RecordingFileName](#)

Created with the Standard Edition of HelpNDoc: [Easily convert your WinHelp HLP help files to CHM with HelpNDoc's step-by-step guide](#)

AutoRefreshPreview

TVideoGrabber.AutoRefreshPreview

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

If enabled, preview is automatically restarted each time a property that requires to restart preview is modified.

Declaration

property AutoRefreshPreview: Boolean **read** GetAutoRefreshPreview **write** SetAutoRefreshPreview
default DEF_AutoRefreshPreview;

__property **bool** AutoRefreshPreview=read=GetAutoRefreshPreview, write=SetAutoRefreshPreview,
default=1

Property AutoRefreshPreview As Boolean

Description

If enabled, preview is automatically restarted each time a property that requires the preview to be restarted (to reflect the new value) is modified.

See Also

[TOnFrameProgress](#) [TOnResizeVideo](#) [BackgroundColor](#) [Display](#) [AutoSize](#) [Display](#) [FullScreen](#)
[Display](#) [SetLocation](#) [Display](#) [VideoPortEnabled](#) [ImageRatio](#) [InFrameProgressEvent](#) [IsVideoInterlaced](#)
[IsVideoPortAvailable](#) [OnFrameProgress](#) [OnKeyPress](#) [OnPreviewStarted](#) [OnResizeVideo](#) [PausePreview](#)
[PreviewZoomSize](#) [ResumePreview](#) [StartPreview](#) [StartSynchronized](#) [StopPreview](#) [Synchronized](#)
[UseNearestVideoSize](#) [VideoHeight](#) [VideoRenderer](#) [VideoRendererPriority](#) [VideoWidth](#) [Visible](#)

Created with the Standard Edition of HelpNDoc: [Benefits of a Help Authoring Tool](#)

AutoStartPlayer

TVideoGrabber.AutoStartPlayer

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Used to specify whether a clip must start playback automatically when [OpenPlayer](#) is called.

Declaration

property AutoStartPlayer: Boolean **read** GetAutoStartPlayer **write** SetAutoStartPlayer **default**
DEF_AutoStartPlayer;

__property **bool** AutoStartPlayer=read=GetAutoStartPlayer, write=SetAutoStartPlayer, **default**=1

Property AutoStartPlayer As Boolean

Description

If enabled, clips start playing automatically when they are opened.

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#)
[AudioStreamNumber](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#)
[IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last](#) [Clip](#) [Played](#) [MP4NeedsReindexing](#)
[OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#)

[OpenPlayer](#)
[OpenPlayerAtFramePositions](#)
[OpenPlayerAtTimePositions](#)
[PausePlayer](#)
[PlayerAudioCodec](#)
[PlayerAudioRendering](#)
[PlayerDuration](#)
[PlayerDVSize](#)
[PlayerFastSeekSpeedRatio](#)
[PlayerFileName](#)
[PlayerForcedCodec](#)
[PlayerFrameCount](#)
[PlayerFramePosition](#)
[PlayerFrameRate](#)
[PlayerFrameStep](#)
[PlayerRefreshPausedDisplay](#)
[PlayerRefreshPausedDisplayFrameRate](#)
[PlayerSpeedRatio](#)
[PlayerTimePosition](#)
[PlayerTrackBar](#)
[PlayerTrackBarScale](#)
[PlayerTrackBarSynchronise](#)
[PlayerVideoCodec](#)
[RewindPlayer](#)
[RunPlayer](#)
[RunPlayerBackwards](#)
[ShowDialog](#)
[SourceStream](#)
[StopPlayer](#)
[SynchronizationRole](#)
[Synchronized](#)
[VideoHeight](#)
[PreferredAspectRatio](#)
[VideoPlayableWhileRecording](#)
[VideoWidth](#)
[PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Leave the tedious WinHelp HLP to CHM conversion process behind with HelpNDoc](#)

AVIDurationUpdated

TVideoGrabber.AVIDurationUpdated

[Prev](#)
[Next](#)

[TVideoGrabber](#)
[Properties](#)

Activates the [OnAVIDurationUpdated](#) event.

Declaration

property AVIDurationUpdated: Boolean **read** GetAVIDurationUpdated **write** SetAVIDurationUpdated;

__property **bool** AVIDurationUpdated=read=GetAVIDurationUpdated, write=SetAVIDurationUpdated, **default false**

Property AVIDurationUpdated As Boolean

Description

When activated and the AVI recording ends:

- TVideoGrabber fixes a possible wrong AVI duration in the AVI file format, that can be caused e.g. by a large amount of dropped frames.
- the [OnAVIDurationUpdated](#) event occurs, that gives more control on the duration and frame rate interval.

See Also

[Recording methods and properties](#)
[TAVIMuxConfig](#)
[TASFDeinterlaceMode](#)
[TAutoFileName](#)
[TOnRecordingCompleted](#)
[TOnRecordingReadyToStart](#)
[TRecordingMethod](#)
[TSyncPreview](#)
[AudioRecording](#)
[AudioSyncAdjustment](#)
[AutoFileNameMinDigits](#)
[AVIFormatOpenDML](#)
[AVIHeaderInfo](#)
[AVIInfo](#)
[Encoder](#)
[SetInt](#)
[HoldRecording](#)
[IsRecordingPaused](#)
[Last_Recording_FileName](#)
[OnBacktimedFramesCountReached](#)
[OnCopyPreallocDataCompleted](#)
[OnCopyPreallocDataProgress](#)
[OnCopyPreallocDataStarted](#)
[OnCreatePreallocFileCompleted](#)
[OnCreatePreallocFileStarted](#)
[OnDiskFull](#)
[OnRecordingCompleted](#)
[OnRecordingPaused](#)
[OnRecordingReadyToStart](#)
[OnRecordingStarted](#)
[OnReencodingCompleted](#)
[OnReencodingStarted](#)
[PauseRecording](#)
[PreallocCapFileCopiedAfterRecording](#)
[PreallocCapFileEnabled](#)
[PreallocCapFileName](#)
[PreallocCapFileSizeInMB](#)
[RecordingBacktimedFramesCount](#)
[RecordingCanPause](#)
[RecordingDuration](#)
[RecordingFileName](#)
[RecordingFileSizeMaxInMB](#)
[RecordingFourCC](#)
[RecordingHeight](#)
[RecordingInNativeFormat](#)
[RecordingKBytesWrittenToDisk](#)
[RecordingMethod](#)
[RecordingSize](#)
[RecordingTimer](#)
[RecordingTimerInterval](#)
[RecordingWidth](#)
[ResumeRecording](#)
[SaveCompressorSettingsToDataString](#)
[SetMultiplexerFilterByName](#)
[StartAudioRecording](#)
[StartRecording](#)
[StartSynchronized](#)
[StopRecording](#)
[StoragePath](#)
[Synchronized](#)
[SyncPreview](#)
[VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [Make Documentation a Breeze with a Help Authoring Tool](#)

AVIFormatOpenDML

TVideoGrabber.AVIFormatOpenDML

[Prev](#)
[Next](#)

[TVideoGrabber](#)
[Properties](#)

Specifies the index format of the AVI file (AVI 1.0 or AVI 2.0 OpenDML).

Declaration

property AVIFormatOpenDML: Boolean **read** GetAVIFormatOpenDML **write** SetAVIFormatOpenDML
default true;

__property **bool** AVIFormatOpenDML=read=GetAVIFormatOpenDML, write=SetAVIFormatOpenDML,
default=1

Property AVIFormatOpenDML As Boolean

Description

Used to specify the index format of the AVI file created during [recording](#) .

false: indicates an AVI 1.0 compatible index format.

true: indicates an AVI 2.0 index format.

AVI 2.0 index format allows for increased AVI file size (greater than 1 GB), hierarchical indexing, incremental growth of files, and minimal disk seeks.

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#)
[TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#)
[AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#)
[HoldRecording](#) [IsRecordingPaused](#) [Last_Recording_FileName](#) [OnBacktimedFramesCountReached](#)
[OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#)
[OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#)
[OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#)
[OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#)
[PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#)
[RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#)
[RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#)
[RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#)
[SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#)
[StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [Upgrade Your Documentation Process with a Help Authoring Tool](#)

AVIFormatOpenDMLCompatibilityIndex**TVideoGrabber.AVIFormatOpenDMLCompatibilityIndex**
[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies if the AVI 2.0 OpenDML file includes an AVI 1.0 index.

Declaration

property AVIFormatOpenDMLCompatibilityIndex: Boolean **read** GetAVIFormatOpenDMLCompatibilityIndex
write SetAVIFormatOpenDMLCompatibilityIndex **default** true;

__property **bool** AVIFormatOpenDMLCompatibilityIndex=read=GetAVIFormatOpenDMLCompatibilityIndex,
write=SetAVIFormatOpenDMLCompatibilityIndex, **default**=1

Property AVIFormatOpenDMLCompatibilityIndex As Boolean

Description

When [AVIFormatOpenDML](#) is enabled:

true : specifies that both an AVI 2.0 index format **AND** an AVI 1.0 index is included in the AVI 2.0 format (for Video for Windows compatibility, allows to get information about the video clip by displaying the file properties)

false: specifies that only an AVI 2.0 index is included.

Created with the Standard Edition of HelpNDoc: [Effortlessly Create High-Quality Documentation with a](#)

[Help Authoring Tool](#)

BackgroundColor**TVideoGrabber.BackgroundColor**[Prev](#)[Next](#)
[TVideoGrabber](#) [Properties](#)

Color of the background when the video size is smaller than the video window.

Declaration

property BackgroundColor: TColor **read** GetBackgroundColor **write** SetBackgroundColor **default** DEF_BackgroundColor;

__property Graphics::TColor BackgroundColor=read=GetBackgroundColor, write=SetBackgroundColor, **default**=0

Property BackgroundColor As OLE_COLOR

Description

Color of the background when the video size is smaller than the video window.

Used when the video window is not embedded and KeepAspectRatio is enabled.

See Also

[TOnFrameProgress](#) [TOnResizeVideo](#) [AutoRefreshPreview](#) [Display](#) [AutoSize](#) [Display_FullScreen](#) [Display_SetLocation](#) [Display_VideoPortEnabled](#) [ImageRatio](#) [InFrameProgressEvent](#) [IsVideoInterlaced](#) [IsVideoPortAvailable](#) [OnFrameProgress](#) [OnKeyPress](#) [OnPreviewStarted](#) [OnResizeVideo](#) [PausePreview](#) [PreviewZoomSize](#) [ResumePreview](#) [StartPreview](#) [StartSynchronized](#) [StopPreview](#) [Synchronized](#) [UseNearestVideoSize](#) [VideoHeight](#) [VideoRenderer](#) [VideoRendererPriority](#) [VideoWidth](#) [Visible](#)

Created with the Standard Edition of HelpNDoc: [Free help authoring tool](#)

BorderStyle**TVideoGrabber.BorderStyle**[Prev](#)[Next](#)
[TVideoGrabber](#) [Properties](#)

Enable or disable border.

Declaration

property BorderStyle: TBorderStyle **read** GetBorderStyle **write** SetBorderStyle **default** DEF_BorderStyle;

__property Forms::TBorderStyle BorderStyle=read=GetBorderStyle, write=SetBorderStyle, **default**=1

Property BorderStyle As TxBorderStyle

Description

Used to enable or disable the control border.

Created with the Standard Edition of HelpNDoc: [Create HTML Help, DOC, PDF and print manuals from 1 single source](#)

BurstCount**TVideoGrabber.BurstCount**[Prev](#)[Next](#)
[TVideoGrabber](#) [Properties](#)

Number of frames to capture.

Declaration

property BurstCount: LongInt **read** GetBurstCount **write** SetBurstCount **default** DEF_BurstCount;

__property **int** BurstCount=read=GetBurstCount, write=SetBurstCount, **default**=3

Property BurstCount As Long

Description

Used to set or retrieve the number of frames to capture when [BurstMode](#) will be enabled.
A "0" value means "don't stop capturing frames".

See Also

[TAutoFileName](#) [TFrameCaptureDest](#) [TOnFrameCaptureCompleted](#) [BurstInterval](#) [BurstMode](#) [BurstType](#) [CaptureFrameSyncTo](#) [CaptureFrameTo](#) [FrameCaptureHeight](#) [FrameCaptureWidth](#) [FrameCaptureWithoutOverlay](#) [FrameCaptureZoomSize](#) [GetFrameInfoString](#) [GetLastFrameAsHBITMAP](#) [GetLastFrameAsTBitmap](#) [GetLastFrameBitmapBits](#) [GetLastFrameBitmapBits2](#) [GetLastFrameWaitTimeoutMs](#) [JPEGPerformance](#) [JPEGProgressiveDisplay](#) [JPEGQuality](#) [Last](#) [BurstFrameCapture](#) [FileName](#) [Last](#) [CaptureFrameTo](#) [FileName](#) [OnDiskFull](#) [OnFrameBitmap](#) [SetFrameCaptureBounds](#) [StoragePath](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Streamline Your Documentation Process with a Help Authoring Tool](#)

BurstInterval

TVideoGrabber.BurstInterval

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Number of frames to skip between 2 captured frames.

Declaration

property BurstInterval: LongInt **read** GetBurstInterval **write** SetBurstInterval **default** DEF_BurstInterval;

__property **int** BurstInterval=read=GetBurstInterval, write=SetBurstInterval, **default**=3

Property BurstInterval As Long

Description

Sets or retrieves the number of frames to skip between 2 captured frames.

A "0" value captures all the frames (no frame skipped).

E.g.: BurstInterval = 20 means 1 frame captured every 20 frames.

See Also

[TAutoFileName](#) [TFrameCaptureDest](#) [TOnFrameCaptureCompleted](#) [BurstCount](#) [BurstMode](#) [BurstType](#) [CaptureFrameSyncTo](#) [CaptureFrameTo](#) [FrameCaptureHeight](#) [FrameCaptureWidth](#) [FrameCaptureWithoutOverlay](#) [FrameCaptureZoomSize](#) [GetFrameInfoString](#) [GetLastFrameAsHBITMAP](#) [GetLastFrameAsTBitmap](#) [GetLastFrameBitmapBits](#) [GetLastFrameBitmapBits2](#) [GetLastFrameWaitTimeoutMs](#) [JPEGPerformance](#) [JPEGProgressiveDisplay](#) [JPEGQuality](#) [Last](#) [BurstFrameCapture](#) [FileName](#) [Last](#) [CaptureFrameTo](#) [FileName](#) [OnDiskFull](#) [OnFrameBitmap](#) [SetFrameCaptureBounds](#) [StoragePath](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [iPhone web sites made easy](#)

BurstMode

TVideoGrabber.BurstMode

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

ber

Enables/disables the automated frame capture process.

Declaration

property BurstMode: Boolean **read** GetBurstMode **write** SetBurstMode **default** DEF_BurstMode;

__property **bool** BurstMode=read=GetBurstMode, write=SetBurstMode, **default**=0

Property BurstMode As Boolean

Description

Used to enable/disable the automated frame capture process.

The number of frames captured depends of the [BurstCount](#) and [BurstInterval](#) parameters.

The destination of the captured frame depends of the [BurstType](#) parameter.

See Also

[TAutoFileName](#) [TFrameCaptureDest](#) [TOnFrameCaptureCompleted](#) [BurstCount](#) [BurstInterval](#) [BurstType](#) [CaptureFrameSyncTo](#) [CaptureFrameTo](#) [FrameCaptureHeight](#) [FrameCaptureWidth](#) [FrameCaptureWithoutOverlay](#) [FrameCaptureZoomSize](#) [GetFrameInfoString](#) [GetLastFrameAsHBITMAP](#) [GetLastFrameAsTBitmap](#) [GetLastFrameBitmapBits](#) [GetLastFrameBitmapBits2](#) [GetLastFrameWaitTimeoutMs](#) [JPEGPerformance](#) [JPEGProgressiveDisplay](#) [JPEGQuality](#) [Last](#) [BurstFrameCapture](#) [FileName](#) [Last](#) [CaptureFrameTo](#) [FileName](#) [OnDiskFull](#) [OnFrameBitmap](#) [SetFrameCaptureBounds](#) [StoragePath](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Easily Add Encryption and Password Protection to Your PDFs](#)

BurstType**TVideoGrabber.BurstType**
[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Destination of captured frames.

Declaration

property BurstType: TFrameCaptureDest **read** GetBurstType **write** SetBurstType **default** DEF_BurstType;

__property TFrameCaptureDest BurstType=read=GetBurstType, write=SetBurstType, **default**=0

Property BurstType As TFrameCaptureDest

Description

Used to set or retrieve the destination of captured frames:

- **fc_TBitmap**: to memory bitmap,
- **fc_BMPFile**: to a BMP file,
- **fc_JPEGFile**: to a JPEG file,
- **fc_Clipboard**: to the clipboard (CF_BITMAP format)

The type of this property is [TFrameCaptureDest](#).

See Also

[TAutoFileName](#) [TFrameCaptureDest](#) [TOnFrameCaptureCompleted](#) [BurstCount](#) [BurstInterval](#) [BurstMode](#) [CaptureFrameSyncTo](#) [CaptureFrameTo](#) [FrameCaptureHeight](#) [FrameCaptureWidth](#) [FrameCaptureWithoutOverlay](#) [FrameCaptureZoomSize](#) [GetFrameInfoString](#) [GetLastFrameAsHBITMAP](#) [GetLastFrameAsTBitmap](#) [GetLastFrameBitmapBits](#) [GetLastFrameBitmapBits2](#) [GetLastFrameWaitTimeoutMs](#) [JPEGPerformance](#) [JPEGProgressiveDisplay](#) [JPEGQuality](#) [Last](#) [BurstFrameCapture](#) [FileName](#) [Last](#) [CaptureFrameTo](#) [FileName](#) [OnDiskFull](#) [OnFrameBitmap](#) [SetFrameCaptureBounds](#) [StoragePath](#) [WebcamStillCaptureButton](#)

Busy

TVideoGrabber.Busy

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Returns true if TVideoGrabber is in an intermediary state.

Declaration

property Busy: Boolean **read** GetBusy;

__property **bool** Busy=read=GetBusy, **nodefault**

Property Busy As Boolean

Description

Returns true if TVideoGrabber is in an intermediary state (e.g. during preview or recording startup) and cannot execute commands immediately.

However, commands and property changes received by the component during a busy state are queued and executed as soon as the component becomes available.

BusyCursor

TVideoGrabber.BusyCursor

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Cursor displayed when TVideoGrabber is in a [busy](#) state.

Declaration

property BusyCursor: TCursor **read** GetBusyCursor **write** SetBusyCursor **default** DEF_BusyCursor;

__property Controls::TCursor BusyCursor=read=GetBusyCursor, write=SetBusyCursor, **default=- 11**

Property BusyCursor As TxCursors

Description

Sets or retrieves the cursor displayed while the component processes significant tasks (like reencoding or copying captured data, restarting preview or capture).

The BusyCursor is displayed when the task begins, and then, the [NormalCursor](#) is displayed when the task ends.

Set BusyCursor to **crDefault** to disable this feature. In this case NormalCursor is ignored.

CameraControlSettings

TVideoGrabber.CameraControlSettings

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Used to enable or disable the camera control settings.

Declaration

property CameraControlSettings: Boolean **read** GetCameraControlSettings **write** SetCameraControlSettings **default** DEF_CameraControlSettings;

__property **bool** CameraControlSettings=read=GetCameraControlSettings, write=SetCameraControlSettings, **default**=1

Property CameraControlSettings As Boolean

Description

Enables or disables the automatic saving and restoring of [camera control](#) settings.

See Also

[TCameraControl](#) [CameraControlAuto](#) [CameraControlDefault](#) [CameraControlMax](#) [CameraControlMin](#) [CameraControlStep](#) [GetCameraExposure](#) [GetCameraExposureAsString](#) [IsCameraControlAvailable](#) [IsCameraControlSettingAvailable](#) [SendCameraCommand](#) [SetCameraControl](#) [SetCameraExposure](#)

Created with the Standard Edition of HelpNDoc: [Enhance Your Documentation with HelpNDoc's Advanced Project Analyzer](#)

CaptureFileExt

TVideoGrabber.CaptureFileExt

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Extension of AVI files names generated automatically.

Declaration

property CaptureFileExt: **string** **read** GetCaptureFileExt **write** SetCaptureFileExt;

__property wchar_t *CaptureFileExt=read=GetCaptureFileExt, write=SetCaptureFileExt

Property CaptureFileExt As String

Description

Default extension used to create video clips when the file name is generated [automatically](#) .

Leave it blank if you want TVideoGrabber to generate it automatically. In this case it assigns:

- "avi" for standard AVI files,
- "mpg" for MPEG files.

See Also

[AutoFileName](#) [AutoFileNameDateTimeFormat](#) [AutoFilePrefix](#) [Last](#) [BurstFrameCapture](#) [FileName](#) [Last](#) [CaptureFrameTo](#) [FileName](#) [RecordingFileName](#)

Created with the Standard Edition of HelpNDoc: [Step-by-Step Guide: How to Turn Your Word Document into an eBook](#)

ColorKey

TVideoGrabber.ColorKey

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Used to modify the default color key used for window transparency.

Declaration

property ColorKey: TColorKey **read** GetColorKey **write** SetColorKey **default** DEF_ColorKey;

__property bool ColorKey=read=GetColorKey, write=SetColorKey, **default**=0

Property ColorKey As OLECOLOR

Description

Used to change the default color keys used for window transparency and related effects.
Enable the window transparency with [ColorKeyEnabled](#), then invoke [SetWindowTransparency](#) to set your form's or window transparency.

You can find sample code in the transparency checkbox code of the "display" tab of the MainDemo project.

Created with the Standard Edition of HelpNDoc: [Say Goodbye to Documentation Headaches with a Help Authoring Tool](#)

ColorKeyEnabled

TVideoGrabber.ColorKeyEnabled

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables color-key based effects, like windows transparency.

Declaration

property ColorKeyEnabled: Boolean **read** GetColorKeyEnabled **write** SetColorKeyEnabled **default** DEF_ColorKeyEnabled;

__property bool ColorKeyEnabled=read=GetColorKeyEnabled, write=SetColorKeyEnabled, **default**=0

Property ColorKeyEnabled As Boolean

Description

Used to enable color-key based effects, like window transparency.
The default color keys can be changed with the [ColorKey](#) property.
Then, when the preview, recording or play back starts, the [OnColorKeyChange](#) event returns the effective color key to use.
See the VideoGrabber1ColorKeyChange event of the MainDemo demo project for sample code.

See Also

[TOnColorKeyChange](#) [OnColorKeyChange](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Documentation Capabilities with a Help Authoring Tool](#)

CompressionMode

TVideoGrabber.CompressionMode

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Compression mode applied to recording.

Declaration

property CompressionMode: TCompressionMode **read** GetCompressionMode **write** SetCompressionMode **default** DEF_CompressionMode;

__property TCompressionMode CompressionMode=read=GetCompressionMode, write=SetCompressionMode, **default**=0

Property CompressionMode As TxCompressionMode

Description

Used to set/retrieve the [TCompressionMode](#) compression mode applied to recording.

See Also

[Recording methods and properties](#) [TCompressionType](#) [TOnVideoCompressionSettings](#) [AudioCompressor](#) [AudioCompressorIndex](#) [AudioCompressorName](#) [AudioCompressors](#) [AudioCompressorsCount](#) [CompressionType](#) [GetVideoCompressionSettings](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [OnVideoCompressionSettings](#) [RefreshDevicesAndCompressorsLists](#) [SaveCompressorSettingsToDataString](#) [SetVideoCompressionDefaults](#) [SetVideoCompressionSettings](#) [VideoCompression](#) [KeyFrameRate](#) [VideoCompression](#) [PFramesPerKeyFrame](#) [VideoCompression](#) [Quality](#) [VideoCompression](#) [WindowSize](#) [VideoCompressor](#) [VideoCompressorIndex](#) [VideoCompressorName](#) [VideoCompressors](#) [VideoCompressorsCount](#)

Created with the Standard Edition of HelpNDoc: [Easily convert your WinHelp HLP help files to CHM with HelpNDoc's step-by-step guide](#)

CompressionType

TVideoGrabber.CompressionType

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Compression type applied to recording.

Declaration

property CompressionType: TCompressionType **read** GetCompressionType **write** SetCompressionType
default DEF_CompressionType;

__property TCompressionType CompressionType=read=GetCompressionType,
write=SetCompressionType, **default**=0

Property CompressionType As TxCompressionType

Description

Used to set/retrieve the [TCompressionType](#) compression type applied to recording.

See Also

[Recording methods and properties](#) [TCompressionType](#) [TOnVideoCompressionSettings](#) [AudioCompressor](#) [AudioCompressorIndex](#) [AudioCompressorName](#) [AudioCompressors](#) [AudioCompressorsCount](#) [CompressionMode](#) [GetVideoCompressionSettings](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [OnVideoCompressionSettings](#) [RefreshDevicesAndCompressorsLists](#) [SaveCompressorSettingsToDataString](#) [SetVideoCompressionDefaults](#) [SetVideoCompressionSettings](#) [VideoCompression](#) [KeyFrameRate](#) [VideoCompression](#) [PFramesPerKeyFrame](#) [VideoCompression](#) [Quality](#) [VideoCompression](#) [WindowSize](#) [VideoCompressor](#) [VideoCompressorIndex](#) [VideoCompressorName](#) [VideoCompressors](#) [VideoCompressorsCount](#)

Created with the Standard Edition of HelpNDoc: [Easily create iPhone documentation](#)

Cropping_Enabled

TVideoGrabber.Cropping_Enabled

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables the video cropping

Declaration

property Cropping_Enabled: Boolean **read** GetCropping_Enabled **write** SetCropping_Enabled **default**

DEF_Cropping_Enabled;

__property **bool** Cropping_Enabled=read=GetCropping_Enabled, write=SetCropping_Enabled, **default**=0

Property Cropping_Enabled As Boolean

Description

Used to enable/disable the cropping of the video stream.

See Also

[Cropping and zooming](#) [Cropping_Height](#) [Cropping_Outbounds](#) [Cropping_Width](#) [Cropping_X](#) [Cropping_XMax](#) [Cropping_Y](#) [Cropping_YMax](#) [Cropping_Zoom](#)

Created with the Standard Edition of HelpNDoc: [Free Kindle producer](#)

Cropping_Height

TVideoGrabber.Cropping_Height

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Height of the cropping area.

Declaration

property Cropping_Height: LongInt **read** GetCropping_Height **write** SetCropping_Height **default** DEF_Cropping_Height;

__property **int** Cropping_Height=read=GetCropping_Height, write=SetCropping_Height, **default**=120

Property Cropping_Height As Long

Description

Specifies the height of the cropping area. The value must be lower or equal to the [VideoHeight](#) property.

See Also

[Cropping and zooming](#) [Cropping_Enabled](#) [Cropping_Outbounds](#) [Cropping_Width](#) [Cropping_X](#) [Cropping_XMax](#) [Cropping_Y](#) [Cropping_YMax](#) [Cropping_Zoom](#)

Created with the Standard Edition of HelpNDoc: [Free iPhone documentation generator](#)

Cropping_Outbounds

TVideoGrabber.Cropping_Outbounds

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Keeps the cropping area centered when zooming near the edges.

Declaration

property Cropping_Outbounds: Boolean **read** GetCropping_Outbounds **write** SetCropping_Outbounds **default** DEF_Cropping_Outbounds;

__property **bool** Cropping_Outbounds=read=GetCropping_Outbounds, write=SetCropping_Outbounds, **default**=0

Property Cropping_Outbounds As Boolean

Description

Useful when zooming near the edges of the video window.

When this property is enabled, moving the [Cropping_X](#) and [Cropping_Y](#) values against the edge of the video window makes the edge displayed at the center of the control (with a background border), allowing to zoom near the edge while keeping the image centered.

See Also

[Cropping and zooming](#) [Cropping_Enabled](#) [Cropping_Height](#) [Cropping_Width](#) [Cropping_X](#) [Cropping_XMax](#) [Cropping_Y](#) [Cropping_YMax](#) [Cropping_Zoom](#)

Created with the Standard Edition of HelpNDoc: [Create iPhone web-based documentation](#)

Cropping_Width

TVideoGrabber.Cropping_Width

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Width of the cropping area.

Declaration

property Cropping_Width: LongInt **read** GetCropping_Width **write** SetCropping_Width **default** DEF_Cropping_Width;

__property **int** Cropping_Width=read=GetCropping_Width, write=SetCropping_Width, **default**=160

Property Cropping_Width As Long

Description

Specifies the width of the cropping area. The value must be lower or equal to the [VideoWidth](#) property.

See Also

[Cropping and zooming](#) [Cropping_Enabled](#) [Cropping_Height](#) [Cropping_Outbounds](#) [Cropping_X](#) [Cropping_XMax](#) [Cropping_Y](#) [Cropping_YMax](#) [Cropping_Zoom](#)

Created with the Standard Edition of HelpNDoc: [Easy to use tool to create HTML Help files and Help web sites](#)

Cropping_X

TVideoGrabber.Cropping_X

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

X axis location of the cropping area.

Declaration

property Cropping_X: LongInt **read** GetCropping_X **write** SetCropping_X **default** DEF_Cropping_X;

__property **int** Cropping_X=read=GetCropping_X, write=SetCropping_X, **default**=0

Property Cropping_X As Long

Description

Used to set/retrieve the X axis location of the cropping area.

See Also

[Cropping and zooming](#) [Cropping_Enabled](#) [Cropping_Height](#) [Cropping_Outbounds](#) [Cropping_Width](#) [Cropping_XMax](#) [Cropping_Y](#) [Cropping_YMax](#) [Cropping_Zoom](#)

Created with the Standard Edition of HelpNDoc: [Free help authoring environment](#)

Cropping_XMax

TVideoGrabber.Cropping_XMax

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Maximum [Cropping_X](#) position allowed, depending of the current video size.

Declaration

property Cropping_XMax: LongInt **read** GetCropping_XMax;

__property **int** Cropping_XMax=read=GetCropping_XMax, **nodefault**

Property Cropping_XMax As Long

Description

Returns the maximum [Cropping_X](#) position allowed, depending of the video size.
This property is refreshed when the [OnResizeVideo](#) event occurs.

See Also

[Cropping and zooming](#) [Cropping_Enabled](#) [Cropping_Height](#) [Cropping_Outbounds](#) [Cropping_Width](#) [Cropping_X](#) [Cropping_Y](#) [Cropping_YMax](#) [Cropping_Zoom](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Documentation Capabilities with HelpNDoc's Project Analyzer](#)

Cropping_Y

TVideoGrabber.Cropping_Y

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Y axis location of the cropping area.

Declaration

property Cropping_Y: LongInt **read** GetCropping_Y **write** SetCropping_Y **default** DEF_Cropping_Y;

__property **int** Cropping_Y=read=GetCropping_Y, write=SetCropping_Y, **default=0**

Property Cropping_Y As Long

Description

Used to set/retrieve the Y axis location of the cropping area.

See Also

[Cropping and zooming](#) [Cropping_Enabled](#) [Cropping_Height](#) [Cropping_Outbounds](#) [Cropping_Width](#) [Cropping_X](#) [Cropping_XMax](#) [Cropping_YMax](#) [Cropping_Zoom](#)

Created with the Standard Edition of HelpNDoc: [Upgrade Your Documentation Process with a Help Authoring Tool](#)

Cropping_YMax

TVideoGrabber.Cropping_YMax

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Maximum [Cropping_Y](#) position allowed, depending of the current video size.

Declaration

property Cropping_YMax: LongInt **read** GetCropping_YMax;

__property **int** Cropping_YMax=read=GetCropping_YMax, **nodefault**

Property Cropping_YMax As Long

Description

Returns the maximum [Cropping_Y](#) position allowed, depending of the video size.
This property is refreshed when the [OnResizeVideo](#) event occurs.

See Also

[Cropping and zooming](#) [Cropping_Enabled](#) [Cropping_Height](#) [Cropping_Outbounds](#) [Cropping_Width](#) [Cropping_X](#) [Cropping_XMax](#) [Cropping_Y](#) [Cropping_Zoom](#)

Created with the Standard Edition of HelpNDoc: [Save time and frustration with HelpNDoc's WinHelp HLP to CHM conversion feature](#)

Cropping_Zoom

TVideoGrabber.Cropping_Zoom

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Zooming coefficient of the cropping area displayed.

Declaration

property Cropping_Zoom: Double **read** GetCropping_Zoom **write** SetCropping_Zoom;

__property **double** Cropping_Zoom=read=GetCropping_Zoom, write=SetCropping_Zoom

Property Cropping_Zoom As Double

Description

Sets/retrieves the zooming coefficient of the cropping area displayed.

See Also

[Cropping and zooming](#) [Cropping_Enabled](#) [Cropping_Height](#) [Cropping_Outbounds](#) [Cropping_Width](#) [Cropping_X](#) [Cropping_XMax](#) [Cropping_Y](#) [Cropping_YMax](#)

Created with the Standard Edition of HelpNDoc: [Generate EPub eBooks with ease](#)

CurrentFrameRate

TVideoGrabber.CurrentFrameRate

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Effective frame rate.

Declaration

property CurrentFrameRate: Double **read** GetCurrentFrameRate;

__property **double** CurrentFrameRate=read=GetCurrentFrameRate

Property CurrentFrameRate As Double

Description

Retrieves the effective frame rate, that can differ from the [FrameRate](#) that has been set.
Useful to know when the FrameRate property cannot be set, e.g. with video capture devices that have a

fixed frame rate depending of the video format (e.g. like DV sources).

Note that some video capture devices accept any stupid value and return this stupid value even if they limit this value to an acceptable value.

See Also

[WDM drivers](#) [AutoConnectRelatedPins](#) [DeliveredFrames](#) [DroppedFrames](#) [FixFlickerOrBlackCapture](#) [IsDVDevice](#) [IsVideoControlAvailable](#) [IsVideoCrossbarAvailable](#) [IsVideoDeviceConnected](#) [IsWDMVideoDriver](#) [OnNoVideoDevices](#) [OnVideoDeviceSelected](#) [RefreshDevicesAndCompressorsLists](#) [ResetVideoDeviceSettings](#) [ShowDialog](#) [VideoDevice](#) [VideoDeviceIndex](#) [VideoDeviceIndexFromId](#) [VideoDeviceName](#) [VideoDevices](#) [VideoDevicesCount](#) [VideoDevicesId](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Documentation Efficiency with a Help Authoring Tool](#)

CurrentState

TVideoGrabber.CurrentState

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Retrieves the [TCurrentState](#) current state of TVideoGrabber.

Declaration

property CurrentState: TCurrentState **read** GetCurrentState;

__property TCurrentState CurrentState=read=GetCurrentState, **nodefault**

Property CurrentState As TCurrentState

Description

Used to retrieve the [TCurrentState](#) current state of TVideoGrabber (down, previewing, capturing to AVI, reencodinging or playing back).

Created with the Standard Edition of HelpNDoc: [Free EBook and documentation generator](#)

DeliveredFrames

TVideoGrabber.DeliveredFrames

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Current number of delivered frames.

Declaration

property DeliveredFrames: LargeInteger **read** GetDeliveredFrames;

__property __int64 DeliveredFrames=read=GetDeliveredFrames

Property DeliveredFrames As Double

Description

Used to retrieve the current number of delivered frames since [preview](#) or [recording](#) started.

See Also

[WDM drivers](#) [AutoConnectRelatedPins](#) [CurrentFrameRate](#) [DroppedFrames](#) [FixFlickerOrBlackCapture](#) [IsDVDevice](#) [IsVideoControlAvailable](#) [IsVideoCrossbarAvailable](#) [IsVideoDeviceConnected](#) [IsWDMVideoDriver](#) [OnNoVideoDevices](#) [OnVideoDeviceSelected](#) [RefreshDevicesAndCompressorsLists](#) [ResetVideoDeviceSettings](#) [ShowDialog](#) [VideoDevice](#) [VideoDeviceIndex](#) [VideoDeviceIndexFromId](#) [VideoDeviceName](#) [VideoDevices](#) [VideoDevicesCount](#) [VideoDevicesId](#) [WebcamStillCaptureButton](#)

DirectShowFilters

TVideoGrabber.DirectShowFilters

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

List of the DirectShow filters available

Declaration

function DirectShowFilters: **string**;

__**property** wchar_t * __**fastcall** DirectShowFilters();

property DirectShowFilters as string

Description

Returns the list of the DirectShow filters available on the current platform.

DirectShowFiltersCount

TVideoGrabber.DirectShowFiltersCount

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Number of DirectShow filters available

Declaration

property DirectShowFiltersCount: LongInt read GetGlobal_DirectShowFiltersCount;

__property int DirectShowFiltersCount=read=GetGlobal_DirectShowFiltersCount, nodefault

Function DirectShowFiltersCount as Long

Description

Returns the number of DirectShow filters listed in the [DirectShowFilters](#) list.

Display_Active

TVideoGrabber.Display_Active

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables/disables the video window.

Declaration

property Display_Active: Boolean **index** vwActive **read** GetDisplayBoolProperties **write** SetDisplayBoolProperties **default** DEF_Display_Active;

__property **bool** Display_Active=read=GetDisplayBoolProperties, write=SetDisplayBoolProperties, index=0,

default=1

Property Display_Active As Boolean

Description

Used to enable/disable the video window.

See Also

[TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display_AlphaBlendEnabled](#) [Display_AlphaBlendValue](#) [Display_AutoSize](#) [Display_Embedded](#) [Display_FullScreen](#) [Display_Height](#) [Display_Left](#) [Display_Monitor](#) [Display_MouseMovesWindow](#) [Display_PanScanRatio](#) [Display_SetLocation](#) [Display_StayOnTop](#) [Display_Top](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [Display_VideoHeight](#) [Display_VideoPortEnabled](#) [Display_VideoWidth](#) [Display_VideoWindowHandle](#) [Display_Width](#) [DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#) [DualDisplay_AlphaBlendValue](#) [DualDisplay_AutoSize](#) [DualDisplay_Embedded](#) [DualDisplay_FullScreen](#) [DualDisplay_Height](#) [DualDisplay_Left](#) [DualDisplay_Monitor](#) [DualDisplay_MouseMovesWindow](#) [DualDisplay_PanScanRatio](#) [DualDisplay_StayOnTop](#) [DualDisplay_Top](#) [DualDisplay_TransparentColorEnabled](#) [DualDisplay_TransparentColorValue](#) [DualDisplay_VideoHeight](#) [DualDisplay_VideoPortEnabled](#) [DualDisplay_VideoWidth](#) [DualDisplay_VideoWindowHandle](#) [DualDisplay_Visible](#) [DualDisplay_Width](#) [IsVideoPortAvailable](#) [Monitor_Primary](#) [Index](#) [MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#) [SetParentWindow](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Easily create EBooks](#)

Display_AlphaBlendEnabled**TVideoGrabber.Display_AlphaBlendEnabled**[Prev](#)[Next](#)
[TVideoGrabber](#) [Properties](#)

Activates the alpha blending of the video window

Declaration

property Display_AlphaBlendEnabled: Boolean **index** vwAlphaBlendEnabled **read**
 GetDisplayBoolProperties **write** SetDisplayBoolProperties **default** DEF_Display_AlphaBlendEnabled;

__property bool Display_AlphaBlendEnabled=read=GetDisplayBoolProperties,
 write=SetDisplayBoolProperties, index=0, **default=1**

Property Display_AlphaBlendEnabled As Boolean

Description

Used to activate the alpha blending of the current video window with another video window when it is detached from the control (when [Display_Embedded](#) = false)

See Also

[Dual display](#) [TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display_Active](#) [Display_AlphaBlendValue](#) [Display_AutoSize](#) [Display_Embedded](#) [Display_FullScreen](#) [Display_Height](#) [Display_Left](#) [Display_Monitor](#) [Display_MouseMovesWindow](#) [Display_PanScanRatio](#) [Display_SetLocation](#) [Display_StayOnTop](#) [Display_Top](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [Display_VideoHeight](#) [Display_VideoPortEnabled](#) [Display_VideoWidth](#) [Display_VideoWindowHandle](#) [Display_Width](#) [DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#) [DualDisplay_AlphaBlendValue](#) [DualDisplay_AutoSize](#) [DualDisplay_Embedded](#) [DualDisplay_FullScreen](#) [DualDisplay_Height](#) [DualDisplay_Left](#) [DualDisplay_Monitor](#) [DualDisplay_MouseMovesWindow](#) [DualDisplay_PanScanRatio](#) [DualDisplay_SetLocation](#) [DualDisplay_StayOnTop](#) [DualDisplay_Top](#) [DualDisplay_TransparentColorEnabled](#) [DualDisplay_TransparentColorValue](#) [DualDisplay_VideoHeight](#) [DualDisplay_VideoPortEnabled](#) [DualDisplay_VideoWidth](#) [DualDisplay_VideoWindowHandle](#) [DualDisplay_Visible](#) [DualDisplay_Width](#) [IsVideoPortAvailable](#) [Monitor_Primary](#) [Index](#) [MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#) [SetParentWindow](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#)

[VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Easily create Web Help sites](#)

Display_AlphaBlendValue

TVideoGrabber.Display_AlphaBlendValue

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Value of the alpha blending of the video window

Declaration

property Display_AlphaBlendValue: LongInt **index** vwAlphaBlendValue **read** GetDisplayLongIntProperties **write** SetDisplayLongIntProperties **default** DEF_Display_AlphaBlendValue;

__property int Display_AlphaBlendValue=read=GetDisplayLongIntProperties, write=SetDisplayLongIntProperties, index=12, **default**=240

Property Display_AlphaBlendValue As Long

Description

Value of the alpha blending of the video window, in the 0..255 range (e.g. 128 = semi-transparent)
This feature can be used when [Display_AlphaBlendEnabled](#) = true and [Display_Embedded](#) = false

See Also

[Dual display](#) [TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display_Active](#) [Display_AlphaBlendEnabled](#) [Display_AutoSize](#) [Display_Embedded](#) [Display_FullScreen](#) [Display_Height](#) [Display_Left](#) [Display_Monitor](#) [Display_MouseMovesWindow](#) [Display_PanScanRatio](#) [Display_SetLocation](#) [Display_StayOnTop](#) [Display_Top](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [Display_VideoHeight](#) [Display_VideoPortEnabled](#) [Display_VideoWidth](#) [Display_VideoWindowHandle](#) [Display_Width](#) [DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#) [DualDisplay_AlphaBlendValue](#) [DualDisplay_AutoSize](#) [DualDisplay_Embedded](#) [DualDisplay_FullScreen](#) [DualDisplay_Height](#) [DualDisplay_Left](#) [DualDisplay_Monitor](#) [DualDisplay_MouseMovesWindow](#) [DualDisplay_PanScanRatio](#) [DualDisplay_SetLocation](#) [DualDisplay_StayOnTop](#) [DualDisplay_Top](#) [DualDisplay_TransparentColorEnabled](#) [DualDisplay_TransparentColorValue](#) [DualDisplay_VideoHeight](#) [DualDisplay_VideoPortEnabled](#) [DualDisplay_VideoWidth](#) [DualDisplay_VideoWindowHandle](#) [DualDisplay_Visible](#) [DualDisplay_Width](#) [IsVideoPortAvailable](#) [Monitor_Primary_Index](#) [MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#) [SetParentWindow](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Spot and Fix Problems in Your Documentation with HelpNDoc's Project Analyzer](#)

Display_AspectRatio

TVideoGrabber.Display_AspectRatio

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the aspect ratio to use within the video window.

Declaration

property Display_AspectRatio: TAspectRatio **index** 0 **read** GetDisplayAspectRatio **write** SetDisplayAspectRatio **default** ar_Box;

__property TAspectRatio Display_AspectRatio=read=GetDisplayAspectRatio, write=SetDisplayAspectRatio, index=0, **default**=0

Property `Display_AspectRatio` As `TAspectRatio`

Description

Used when [Display_AutoSize](#) = false to specify the aspect ratio method that must be used within the video window.

The possible values are described in the [TAspectRatio](#) type.

See Also

[TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display_Active](#) [Display_AlphaBlendEnabled](#) [Display_AlphaBlendValue](#) [Display_AutoSize](#) [Display_Embedded](#) [Display_FullScreen](#) [Display_Height](#) [Display_Left](#) [Display_Monitor](#) [Display_MouseMovesWindow](#) [Display_PanScanRatio](#) [Display_SetLocation](#) [Display_StayOnTop](#) [Display_Top](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [Display_VideoHeight](#) [Display_VideoPortEnabled](#) [Display_VideoWidth](#) [Display_VideoWindowHandle](#) [Display_Width](#) [DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#) [DualDisplay_AlphaBlendValue](#) [DualDisplay_AutoSize](#) [DualDisplay_Embedded](#) [DualDisplay_FullScreen](#) [DualDisplay_Height](#) [DualDisplay_Left](#) [DualDisplay_Monitor](#) [DualDisplay_MouseMovesWindow](#) [DualDisplay_PanScanRatio](#) [DualDisplay_StayOnTop](#) [DualDisplay_Top](#) [DualDisplay_TransparentColorEnabled](#) [DualDisplay_TransparentColorValue](#) [DualDisplay_VideoHeight](#) [DualDisplay_VideoPortEnabled](#) [DualDisplay_VideoWidth](#) [DualDisplay_VideoWindowHandle](#) [DualDisplay_Visible](#) [DualDisplay_Width](#) [IsVideoPortAvailable](#) [Monitor_Primary](#) [Index](#) [MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#) [SetParentWindow](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: Powerful and User-Friendly Help Authoring Tool for Markdown Documents

Display_AutoSize

TVideoGrabber.Display_AutoSize

[Prev](#)
[Next](#)

[TVideoGrabber](#) Properties

If enabled, the control is resized automatically according to the current video size.

Declaration

property `Display_AutoSize`: Boolean **index** `vwAutosize` **read** `GetDisplayBoolProperties` **write** `SetDisplayBoolProperties` **default** `DEF_Display_AutoSize`;

`__property bool Display_AutoSize=``read=GetDisplayBoolProperties, write=SetDisplayBoolProperties, index=1, default=1`

Property `Display_AutoSize` As Boolean

Description

Specifies whether the control must be resized automatically according to the current video size.

disabled: the control size depends of the Width and Height properties, and the video window is stretched inside.

enabled: the control size is automatically modified according to [VideoSize](#) or [UseNearestVideoSize](#) .

See Also

[TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display_Active](#) [Display_AlphaBlendEnabled](#) [Display_AlphaBlendValue](#) [Display_Embedded](#) [Display_FullScreen](#) [Display_Height](#) [Display_Left](#) [Display_Monitor](#) [Display_MouseMovesWindow](#) [Display_PanScanRatio](#) [Display_SetLocation](#) [Display_StayOnTop](#) [Display_Top](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [Display_VideoHeight](#) [Display_VideoPortEnabled](#) [Display_VideoWidth](#) [Display_VideoWindowHandle](#) [Display_Width](#) [DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#) [DualDisplay_AlphaBlendValue](#) [DualDisplay_AutoSize](#) [DualDisplay_Embedded](#) [DualDisplay_FullScreen](#) [DualDisplay_Height](#) [DualDisplay_Left](#) [DualDisplay_Monitor](#) [DualDisplay_MouseMovesWindow](#) [DualDisplay_PanScanRatio](#) [DualDisplay_StayOnTop](#) [DualDisplay_Top](#) [DualDisplay_TransparentColorEnabled](#)

[DualDisplay_TransparentColorValue](#) [DualDisplay_VideoHeight](#) [DualDisplay_VideoPortEnabled](#) [DualDisplay_VideoWidth](#) [DualDisplay_VideoWindowHandle](#) [DualDisplay_Visible](#) [DualDisplay_Width](#) [IsVideoPortAvailable](#) [Monitor_Primary_Index](#) [MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#) [SetParentWindow](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Transform your help documentation into a stunning website](#)

Display_Embedded

TVideoGrabber.Display_Embedded

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Used to detach/attach the video window from the TVideoGrabber control.

Declaration

property Display_Embedded: Boolean **index** vwEmbedded **read** GetDisplayBoolProperties **write** SetDisplayBoolProperties **default** DEF_Display_Embedded;

__property **bool** Display_Embedded=read=GetDisplayBoolProperties, write=SetDisplayBoolProperties, index=2, **default**=1

Property Display_Embedded As Boolean

Description

Used to detach/attach the video window from the TVideoGrabber control.

- **enabled**: the video window is embedded into the TVideoGrabber control,
- **disabled**: the video window is located on the desktop at the [Display_Left](#) and [Display_Top](#) positions (on the [Display_Monitor](#) monitor if more than 1 monitor is installed).

See Also

[TVideoRenderer_AdjustPixelAspectRatio](#) [Display_Active](#) [Display_AlphaBlendEnabled](#) [Display_AlphaBlendValue](#) [Display_AutoSize](#) [Display_FullScreen](#) [Display_Height](#) [Display_Left](#) [Display_Monitor](#) [Display_MouseMovesWindow](#) [Display_PanScanRatio](#) [Display_SetLocation](#) [Display_StayOnTop](#) [Display_Top](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [Display_VideoHeight](#) [Display_VideoPortEnabled](#) [Display_VideoWidth](#) [Display_VideoWindowHandle](#) [Display_Width](#) [DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#) [DualDisplay_AlphaBlendValue](#) [DualDisplay_AutoSize](#) [DualDisplay_Embedded](#) [DualDisplay_FullScreen](#) [DualDisplay_Height](#) [DualDisplay_Left](#) [DualDisplay_Monitor](#) [DualDisplay_MouseMovesWindow](#) [DualDisplay_PanScanRatio](#) [DualDisplay_StayOnTop](#) [DualDisplay_Top](#) [DualDisplay_TransparentColorEnabled](#) [DualDisplay_TransparentColorValue](#) [DualDisplay_VideoHeight](#) [DualDisplay_VideoPortEnabled](#) [DualDisplay_VideoWidth](#) [DualDisplay_VideoWindowHandle](#) [DualDisplay_Visible](#) [DualDisplay_Width](#) [IsVideoPortAvailable](#) [Monitor_Primary_Index](#) [MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#) [SetParentWindow](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Full-featured EPub generator](#)

Display_FullScreen

TVideoGrabber.Display_FullScreen

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Displays the preview window in full screen mode.

Declaration

property Display_FullScreen: Boolean **index** vwFullScreen **read** GetDisplayBoolProperties **write** SetDisplayBoolProperties **default** DEF_Display_FullScreen;

__property **bool** Display_FullScreen=read=GetDisplayBoolProperties, write=SetDisplayBoolProperties, index=3, **default**=0

Property Display_FullScreen As Boolean

Description

If enabled, the preview window is displayed in full screen mode when the preview, recording or play back starts.

To exit from the full screen mode disable this property or press the <ESC> key.

In full screen mode all keystrokes are returned by the [OnKeyPress](#) event.

See Also

[TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display_Active](#) [Display_AlphaBlendEnabled](#) [Display_AlphaBlendValue](#) [Display_AutoSize](#) [Display_Embedded](#) [Display_Height](#) [Display_Left](#) [Display_Monitor](#) [Display_MouseMovesWindow](#) [Display_PanScanRatio](#) [Display_SetLocation](#) [Display_StayOnTop](#) [Display_Top](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [Display_VideoHeight](#) [Display_VideoPortEnabled](#) [Display_VideoWidth](#) [Display_VideoWindowHandle](#) [Display_Width](#) [DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#) [DualDisplay_AlphaBlendValue](#) [DualDisplay_AutoSize](#) [DualDisplay_Embedded](#) [DualDisplay_FullScreen](#) [DualDisplay_Height](#) [DualDisplay_Left](#) [DualDisplay_Monitor](#) [DualDisplay_MouseMovesWindow](#) [DualDisplay_PanScanRatio](#) [DualDisplay_StayOnTop](#) [DualDisplay_Top](#) [DualDisplay_TransparentColorEnabled](#) [DualDisplay_TransparentColorValue](#) [DualDisplay_VideoHeight](#) [DualDisplay_VideoPortEnabled](#) [DualDisplay_VideoWidth](#) [DualDisplay_VideoWindowHandle](#) [DualDisplay_Visible](#) [DualDisplay_Width](#) [IsVideoPortAvailable](#) [Monitor_Primary_Index](#) [MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#) [SetParentWindow](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: Transform Your Documentation Process with HelpNDoc's Project Analyzer

Display_Height

TVideoGrabber.Display_Height

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the height of the video window, when it is not embedded in the TVideoGrabber control.

Declaration

property Display_Height: LongInt **index** vwHeight **read** GetDisplayLongIntProperties **write** SetDisplayLongIntProperties **default** DEF_Display_Height;

__property **int** Display_Height=read=GetDisplayLongIntProperties, write=SetDisplayLongIntProperties, index=12, **default**=240

Property Display_Height As Long

Description

Used to specify the height of the video window, when it is not embedded in the TVideoGrabber control.

Note: the left, top, width and height properties can be set at the same time by using [Display_SetLocation](#) .

See Also

[TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display_Active](#) [Display_AlphaBlendEnabled](#) [Display_AlphaBlendValue](#) [Display_AutoSize](#) [Display_Embedded](#) [Display_FullScreen](#) [Display_Left](#) [Display_Monitor](#) [Display_MouseMovesWindow](#) [Display_PanScanRatio](#) [Display_SetLocation](#) [Display_StayOnTop](#) [Display_Top](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [Display_VideoHeight](#) [Display_VideoPortEnabled](#) [Display_VideoWidth](#) [Display_VideoWindowHandle](#)

[Display_Width](#)
[DualDisplay_Active](#)
[DualDisplay_AlphaBlendEnabled](#)
[DualDisplay_AlphaBlendValue](#)
[DualDisplay_AutoSize](#)
[DualDisplay_Embedded](#)
[DualDisplay_FullScreen](#)
[DualDisplay_Height](#)
[DualDisplay_Left](#)
[DualDisplay_Monitor](#)
[DualDisplay_MouseMovesWindow](#)
[DualDisplay_PanScanRatio](#)
[DualDisplay_StayOnTop](#)
[DualDisplay_Top](#)
[DualDisplay_TransparentColorEnabled](#)
[DualDisplay_TransparentColorValue](#)
[DualDisplay_VideoHeight](#)
[DualDisplay_VideoPortEnabled](#)
[DualDisplay_VideoWidth](#)
[DualDisplay_VideoWindowHandle](#)
[DualDisplay_Visible](#)
[DualDisplay_Width](#)
[IsVideoPortAvailable](#)
[Monitor_Primary](#)
[Index](#)
[MonitorBounds](#)
[MonitorsCount](#)
[OnLeavingFullScreen](#)
[SetParentWindow](#)
[SetWindowTransparency](#)
[VideoDoubleBuffered](#)
[VideoHeight](#)
[PreferredAspectRatio](#)
[VideoRendererExternal](#)
[VideoRendererExternalIndex](#)
[VideoRendererPriority](#)
[VideoVisibleWhenStopped](#)
[VideoWidth](#)
[PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: Effortlessly create a professional-quality documentation website with HelpNDoc

Display_Left

TVideoGrabber.Display_Left

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the left position of the video window, when it is not embedded in the TVideoGrabber control.

Declaration

property Display_Left: LongInt **index** vwLeft **read** GetDisplayLongIntProperties **write** SetDisplayLongIntProperties **default** DEF_Display_Left;

__property **int** Display_Left=read=GetDisplayLongIntProperties, write=SetDisplayLongIntProperties, index=13, **default**=0

Property Display_Left As Long

Description

Specifies the left position of the video window, when it is not embedded in the TVideoGrabber control.

Note: the left, top, width and height properties can be set at the same time by using [Display_SetLocation](#) .

See Also

[TVideoRenderer](#)
[AdjustPixelAspectRatio](#)
[Display_Active](#)
[Display_AlphaBlendEnabled](#)
[Display_AlphaBlendValue](#)
[Display_AutoSize](#)
[Display_Embedded](#)
[Display_FullScreen](#)
[Display_Height](#)
[Display_Monitor](#)
[Display_MouseMovesWindow](#)
[Display_PanScanRatio](#)
[Display_SetLocation](#)
[Display_StayOnTop](#)
[Display_Top](#)
[Display_TransparentColorEnabled](#)
[Display_TransparentColorValue](#)
[Display_VideoHeight](#)
[Display_VideoPortEnabled](#)
[Display_VideoWidth](#)
[Display_VideoWindowHandle](#)
[Display_Width](#)
[DualDisplay_Active](#)
[DualDisplay_AlphaBlendEnabled](#)
[DualDisplay_AlphaBlendValue](#)
[DualDisplay_AutoSize](#)
[DualDisplay_Embedded](#)
[DualDisplay_FullScreen](#)
[DualDisplay_Height](#)
[DualDisplay_Left](#)
[DualDisplay_Monitor](#)
[DualDisplay_MouseMovesWindow](#)
[DualDisplay_PanScanRatio](#)
[DualDisplay_StayOnTop](#)
[DualDisplay_Top](#)
[DualDisplay_TransparentColorEnabled](#)
[DualDisplay_TransparentColorValue](#)
[DualDisplay_VideoHeight](#)
[DualDisplay_VideoPortEnabled](#)
[DualDisplay_VideoWidth](#)
[DualDisplay_VideoWindowHandle](#)
[DualDisplay_Visible](#)
[DualDisplay_Width](#)
[IsVideoPortAvailable](#)
[Monitor_Primary](#)
[Index](#)
[MonitorBounds](#)
[MonitorsCount](#)
[OnLeavingFullScreen](#)
[SetParentWindow](#)
[SetWindowTransparency](#)
[VideoDoubleBuffered](#)
[VideoHeight](#)
[PreferredAspectRatio](#)
[VideoRendererExternal](#)
[VideoRendererExternalIndex](#)
[VideoRendererPriority](#)
[VideoVisibleWhenStopped](#)
[VideoWidth](#)
[PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: News and information about help authoring tools and software

Display_Monitor

TVideoGrabber.Display_Monitor

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the monitor used to display the video window.

Declaration

property Display_Monitor: LongInt **index** vwMonitor **read** GetDisplayLongIntProperties **write** SetDisplayLongIntProperties **default** DEF_Display_Monitor;

__property **int** Display_Monitor=read=GetDisplayLongIntProperties, write=SetDisplayLongIntProperties, index=8, **default**=0

Property Display_Monitor As Long

Description

Used to specify the monitor used to display the video window.
The value is in the (0..[MonitorsCount](#) -1) range.

See Also

[TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display_Active](#) [Display_AlphaBlendEnabled](#) [Display_AlphaBlendValue](#) [Display_AutoSize](#) [Display_Embedded](#) [Display_FullScreen](#) [Display_Height](#) [Display_Left](#) [Display_MouseMovesWindow](#) [Display_PanScanRatio](#) [Display_SetLocation](#) [Display_StayOnTop](#) [Display_Top](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [Display_VideoHeight](#) [Display_VideoPortEnabled](#) [Display_VideoWidth](#) [Display_VideoWindowHandle](#) [Display_Width](#) [DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#) [DualDisplay_AlphaBlendValue](#) [DualDisplay_AutoSize](#) [DualDisplay_Embedded](#) [DualDisplay_FullScreen](#) [DualDisplay_Height](#) [DualDisplay_Left](#) [DualDisplay_Monitor](#) [DualDisplay_MouseMovesWindow](#) [DualDisplay_PanScanRatio](#) [DualDisplay_StayOnTop](#) [DualDisplay_Top](#) [DualDisplay_TransparentColorEnabled](#) [DualDisplay_TransparentColorValue](#) [DualDisplay_VideoHeight](#) [DualDisplay_VideoPortEnabled](#) [DualDisplay_VideoWidth](#) [DualDisplay_VideoWindowHandle](#) [DualDisplay_Visible](#) [DualDisplay_Width](#) [IsVideoPortAvailable](#) [Monitor_Primary_Index](#) [MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#) [SetParentWindow](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Write eBooks for the Kindle](#)

Display_MouseMovesWindow

TVideoGrabber.Display_MouseMovesWindow

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies whether the mouse moves the video window or returns mouse events for this window.

Declaration

property Display_MouseMovesWindow: Boolean **index** vwMouseMovesWindow **read** GetDisplayBoolProperties **write** SetDisplayBoolProperties **default** DEF_Display_MouseMovesWindow;

__property **bool** Display_MouseMovesWindow=read=GetDisplayBoolProperties, write=SetDisplayBoolProperties, index=6, **default**=1

Property Display_MouseMovesWindow As Boolean

Description

When the video window is not embedded in the TVideoGrabber control:

- **if enabled:** the mouse moves the video window,
- **if disabled:** the mouse returns mouse events, e.g. to draw graphic objects over the video window.

See Also

[TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display_Active](#) [Display_AlphaBlendEnabled](#) [Display_AlphaBlendValue](#) [Display_AutoSize](#) [Display_Embedded](#) [Display_FullScreen](#) [Display_Height](#) [Display_Left](#) [Display_Monitor](#) [Display_PanScanRatio](#) [Display_SetLocation](#) [Display_StayOnTop](#) [Display_Top](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [Display_VideoHeight](#)

[Display_VideoPortEnabled](#) [Display_VideoWidth](#) [Display_VideoWindowHandle](#) [Display_Width](#)
[DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#) [DualDisplay_AlphaBlendValue](#) [DualDisplay_AutoSize](#)
[DualDisplay_Embedded](#) [DualDisplay_FullScreen](#) [DualDisplay_Height](#) [DualDisplay_Left](#)
[DualDisplay_Monitor](#) [DualDisplay_MouseMovesWindow](#) [DualDisplay_PanScanRatio](#)
[DualDisplay_StayOnTop](#) [DualDisplay_Top](#) [DualDisplay_TransparentColorEnabled](#)
[DualDisplay_TransparentColorValue](#) [DualDisplay_VideoHeight](#) [DualDisplay_VideoPortEnabled](#)
[DualDisplay_VideoWidth](#) [DualDisplay_VideoWindowHandle](#) [DualDisplay_Visible](#) [DualDisplay_Width](#)
[IsVideoPortAvailable](#) [Monitor_Primary](#) [Index_MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#)
[SetParentWindow](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#)
[VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#)
[VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Easily create Help documents](#)

Display_PanScanRatio

TVideoGrabber.Display_PanScanRatio

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Adjust the Pan/Scan ratio.

Declaration

property Display_PanScanRatio: LongInt **index** vwPanScanRatio **read** GetDisplayLongIntProperties **write** SetDisplayLongIntProperties **default** DEF_Display_PanScanRatio;

__property int Display_PanScanRatio==GetDisplayLongIntProperties, write=SetDisplayLongIntProperties, index=11, **default**=50;

Property Display_PanScanRatio as Long

Description

When [Display_AspectRatio](#) = **ar_PanScan**, this property is used to adjust the Pan/Scan ratio (in the 0..100 range, default value 50)

See Also

[TVideoRenderer_AdjustPixelAspectRatio](#) [Display_Active](#) [Display_AlphaBlendEnabled](#)
[Display_AlphaBlendValue](#) [Display_AutoSize](#) [Display_Embedded](#) [Display_FullScreen](#) [Display_Height](#)
[Display_Left](#) [Display_Monitor](#) [Display_MouseMovesWindow](#) [Display_SetLocation](#) [Display_StayOnTop](#)
[Display_Top](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [Display_VideoHeight](#)
[Display_VideoPortEnabled](#) [Display_VideoWidth](#) [Display_VideoWindowHandle](#) [Display_Width](#)
[DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#) [DualDisplay_AlphaBlendValue](#) [DualDisplay_AutoSize](#)
[DualDisplay_Embedded](#) [DualDisplay_FullScreen](#) [DualDisplay_Height](#) [DualDisplay_Left](#)
[DualDisplay_Monitor](#) [DualDisplay_MouseMovesWindow](#) [DualDisplay_PanScanRatio](#)
[DualDisplay_StayOnTop](#) [DualDisplay_Top](#) [DualDisplay_TransparentColorEnabled](#)
[DualDisplay_TransparentColorValue](#) [DualDisplay_VideoHeight](#) [DualDisplay_VideoPortEnabled](#)
[DualDisplay_VideoWidth](#) [DualDisplay_VideoWindowHandle](#) [DualDisplay_Visible](#) [DualDisplay_Width](#)
[IsVideoPortAvailable](#) [Monitor_Primary](#) [Index_MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#)
[SetParentWindow](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#)
[VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#)
[VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Full-featured Help generator](#)

Display_StayOnTop

TVideoGrabber.Display_StayOnTop

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies whether the video window must stay over other windows, when not embedded.

Declaration

property Display_StayOnTop: Boolean **index** vwStayOnTop **read** GetDisplayBoolProperties **write** SetDisplayBoolProperties **default** DEF_Display_StayOnTop;

__property **bool** Display_StayOnTop=read=GetDisplayBoolProperties, write=SetDisplayBoolProperties, index=5, **default**=0

Property Display_StayOnTop As Boolean

Description

Specifies whether the video window must stay over other windows, when it is not embedded in the TVideoGrabber control.

See Also

[TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display_Active](#) [Display_AlphaBlendEnabled](#) [Display_AlphaBlendValue](#) [Display_AutoSize](#) [Display_Embedded](#) [Display_FullScreen](#) [Display_Height](#) [Display_Left](#) [Display_Monitor](#) [Display_MouseMovesWindow](#) [Display_PanScanRatio](#) [Display_SetLocation](#) [Display_Top](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [Display_VideoHeight](#) [Display_VideoPortEnabled](#) [Display_VideoWidth](#) [Display_VideoWindowHandle](#) [Display_Width](#) [DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#) [DualDisplay_AlphaBlendValue](#) [DualDisplay_AutoSize](#) [DualDisplay_Embedded](#) [DualDisplay_FullScreen](#) [DualDisplay_Height](#) [DualDisplay_Left](#) [DualDisplay_Monitor](#) [DualDisplay_MouseMovesWindow](#) [DualDisplay_PanScanRatio](#) [DualDisplay_StayOnTop](#) [DualDisplay_Top](#) [DualDisplay_TransparentColorEnabled](#) [DualDisplay_TransparentColorValue](#) [DualDisplay_VideoHeight](#) [DualDisplay_VideoPortEnabled](#) [DualDisplay_VideoWidth](#) [DualDisplay_VideoWindowHandle](#) [DualDisplay_Visible](#) [DualDisplay_Width](#) [IsVideoPortAvailable](#) [Monitor_Primary_Index](#) [MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#) [SetParentWindow](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [News and information about help authoring tools and software](#)

Display_Top**TVideoGrabber.Display_Top**
[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the top position of the video window, when it is not embedded in the TVideoGrabber control.

Declaration

property Display_Top: LongInt **index** vwTop **read** GetDisplayLongIntProperties **write** SetDisplayLongIntProperties **default** DEF_Display_Top;

__property **int** Display_Top=read=GetDisplayLongIntProperties, write=SetDisplayLongIntProperties, index=14, **default**=0

Property Display_Top As Long

Description

Specifies the top position of the video window, when it is not embedded in the TVideoGrabber control.

Note: the left, top, width and height properties can be set at the same time by using [Display_SetLocation](#) .

See Also

[TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display_Active](#) [Display_AlphaBlendEnabled](#) [Display_AlphaBlendValue](#) [Display_AutoSize](#) [Display_Embedded](#) [Display_FullScreen](#) [Display_Height](#) [Display_Left](#) [Display_Monitor](#) [Display_MouseMovesWindow](#) [Display_PanScanRatio](#) [Display_SetLocation](#) [Display_StayOnTop](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [Display_VideoHeight](#) [Display_VideoPortEnabled](#) [Display_VideoWidth](#) [Display_VideoWindowHandle](#)

[Display_Width](#)
[DualDisplay_Active](#)
[DualDisplay_AlphaBlendEnabled](#)
[DualDisplay_AlphaBlendValue](#)
[DualDisplay_AutoSize](#)
[DualDisplay_Embedded](#)
[DualDisplay_FullScreen](#)
[DualDisplay_Height](#)
[DualDisplay_Left](#)
[DualDisplay_Monitor](#)
[DualDisplay_MouseMovesWindow](#)
[DualDisplay_PanScanRatio](#)
[DualDisplay_StayOnTop](#)
[DualDisplay_Top](#)
[DualDisplay_TransparentColorEnabled](#)
[DualDisplay_TransparentColorValue](#)
[DualDisplay_VideoHeight](#)
[DualDisplay_VideoPortEnabled](#)
[DualDisplay_VideoWidth](#)
[DualDisplay_VideoWindowHandle](#)
[DualDisplay_Visible](#)
[DualDisplay_Width](#)
[IsVideoPortAvailable](#)
[Monitor_Primary_Index](#)
[MonitorBounds](#)
[MonitorsCount](#)
[OnLeavingFullScreen](#)
[SetParentWindow](#)
[SetWindowTransparency](#)
[VideoDoubleBuffered](#)
[VideoHeight](#)
[PreferredAspectRatio](#)
[VideoRendererExternal](#)
[VideoRendererExternalIndex](#)
[VideoRendererPriority](#)
[VideoVisibleWhenStopped](#)
[VideoWidth](#)
[PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Keep Your Sensitive PDFs Safe with These Easy Security Measures](#)

Display_TransparentColorEnabled

TVideoGrabber.Display_TransparentColorEnabled

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Activate the color keying of the current video window

Declaration

property Display_TransparentColorEnabled: Boolean **index** vwTransparentColorEnabled **read** GetDisplayBoolProperties **write** SetDisplayBoolProperties **default** DEF_Display_TransparentColorEnabled;

__property bool Display_TransparentColorEnabled=read=GetDisplayBoolProperties, write=SetDisplayBoolProperties, index=0, **default**=1

Property Display_TransparentColorEnabled As Boolean

Description

Used to activate the color keying of the current video window with another video window when it is detached from the control (when [Display_Embedded](#) = false)

See Also

[Dual display](#)
[TVideoRenderer](#)
[AdjustPixelAspectRatio](#)
[Display_Active](#)
[Display_AlphaBlendEnabled](#)
[Display_AlphaBlendValue](#)
[Display_AutoSize](#)
[Display_Embedded](#)
[Display_FullScreen](#)
[Display_Height](#)
[Display_Left](#)
[Display_Monitor](#)
[Display_MouseMovesWindow](#)
[Display_PanScanRatio](#)
[Display_SetLocation](#)
[Display_StayOnTop](#)
[Display_Top](#)
[Display_TransparentColorValue](#)
[Display_VideoHeight](#)
[Display_VideoPortEnabled](#)
[Display_VideoWidth](#)
[Display_VideoWindowHandle](#)
[Display_Width](#)
[DualDisplay_Active](#)
[DualDisplay_AlphaBlendEnabled](#)
[DualDisplay_AlphaBlendValue](#)
[DualDisplay_AutoSize](#)
[DualDisplay_Embedded](#)
[DualDisplay_FullScreen](#)
[DualDisplay_Height](#)
[DualDisplay_Left](#)
[DualDisplay_Monitor](#)
[DualDisplay_MouseMovesWindow](#)
[DualDisplay_PanScanRatio](#)
[DualDisplay_SetLocation](#)
[DualDisplay_StayOnTop](#)
[DualDisplay_Top](#)
[DualDisplay_TransparentColorEnabled](#)
[DualDisplay_TransparentColorValue](#)
[DualDisplay_VideoHeight](#)
[DualDisplay_VideoPortEnabled](#)
[DualDisplay_VideoWidth](#)
[DualDisplay_VideoWindowHandle](#)
[DualDisplay_Visible](#)
[DualDisplay_Width](#)
[IsVideoPortAvailable](#)
[Monitor_Primary_Index](#)
[MonitorBounds](#)
[MonitorsCount](#)
[OnLeavingFullScreen](#)
[SetParentWindow](#)
[SetWindowTransparency](#)
[VideoDoubleBuffered](#)
[VideoHeight](#)
[PreferredAspectRatio](#)
[VideoRendererExternal](#)
[VideoRendererExternalIndex](#)
[VideoRendererPriority](#)
[VideoVisibleWhenStopped](#)
[VideoWidth](#)
[PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Support Your Windows Applications with HelpNDoc's CHM Generation](#)

Display_TransparentColorValue

TVideoGrabber.Display_TransparentColorValue

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Value of the color keying of the video window

Declaration

property Display_TransparentColorValue: LongInt **index** vwTransparentColorValue **read** GetDisplayLongIntProperties **write** SetDisplayLongIntProperties **default** DEF_Display_TransparentColorValue;

__**property int** Display_TransparentColorValue=read=GetDisplayLongIntProperties, write=SetDisplayLongIntProperties, index=12, **default**=240

Property Display_TransparentColorValue As Long

Description

Value of the color keying of the video window, expressed in RGB value.

This feature can be used when [Display_TransparentColorEnabled](#) = true and [Display_Embedded](#) = false

See Also

[Dual display TVideoRenderer AdjustPixelAspectRatio Display_Active Display_AlphaBlendEnabled Display_AlphaBlendValue Display_AutoSize Display_Embedded Display_FullScreen Display_Height Display_Left Display_Monitor Display_MouseMovesWindow Display_PanScanRatio Display_SetLocation Display_StayOnTop Display_Top Display_TransparentColorEnabled Display_VideoHeight Display_VideoPortEnabled Display_VideoWidth Display_VideoWindowHandle Display_Width DualDisplay_Active DualDisplay_AlphaBlendEnabled DualDisplay_AlphaBlendValue DualDisplay_AutoSize DualDisplay_Embedded DualDisplay_FullScreen DualDisplay_Height DualDisplay_Left DualDisplay_Monitor DualDisplay_MouseMovesWindow DualDisplay_PanScanRatio DualDisplay_SetLocation DualDisplay_StayOnTop DualDisplay_Top DualDisplay_TransparentColorEnabled DualDisplay_TransparentColorValue DualDisplay_VideoHeight DualDisplay_VideoPortEnabled DualDisplay_VideoWidth DualDisplay_VideoWindowHandle DualDisplay_Visible DualDisplay_Width IsVideoPortAvailable Monitor_Primary_Index MonitorBounds MonitorsCount OnLeavingFullScreen SetParentWindow SetWindowTransparency VideoDoubleBuffered VideoHeight PreferredAspectRatio VideoRendererExternal VideoRendererExternalIndex VideoRendererPriority VideoVisibleWhenStopped VideoWidth PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Make Documentation a Breeze with a Help Authoring Tool](#)

Display_VideoHeight

TVideoGrabber.Display_VideoHeight

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Returns the height of the 1st video window.

Declaration

property Display_VideoHeight: LongInt **read** GetDisplay_VideoHeight;

__**property int** Display_VideoHeight=read=GetDisplay_VideoHeight, **nodefault**

Property Display_VideoHeight As Long

Description

Used to retrieve the height of the 1st video window.

See Also

[TVideoRenderer AdjustPixelAspectRatio Display_Active Display_AlphaBlendEnabled Display_AlphaBlendValue Display_AutoSize Display_Embedded Display_FullScreen Display_Height Display_Left Display_Monitor Display_MouseMovesWindow Display_PanScanRatio Display_SetLocation Display_StayOnTop Display_Top Display_TransparentColorEnabled Display_TransparentColorValue Display_VideoPortEnabled Display_VideoWidth Display_VideoWindowHandle Display_Width DualDisplay_Active DualDisplay_AlphaBlendEnabled DualDisplay_AlphaBlendValue DualDisplay_AutoSize](#)

[DualDisplay_Embedded](#)
[DualDisplay_FullScreen](#)
[DualDisplay_Height](#)
[DualDisplay_Left](#)
[DualDisplay_Monitor](#)
[DualDisplay_MouseMovesWindow](#)
[DualDisplay_PanScanRatio](#)
[DualDisplay_StayOnTop](#)
[DualDisplay_Top](#)
[DualDisplay_TransparentColorEnabled](#)
[DualDisplay_TransparentColorValue](#)
[DualDisplay_VideoHeight](#)
[DualDisplay_VideoPortEnabled](#)
[DualDisplay_VideoWidth](#)
[DualDisplay_VideoWindowHandle](#)
[DualDisplay_Visible](#)
[DualDisplay_Width](#)
[IsVideoPortAvailable](#)
[Monitor_Primary](#)
[Index](#)
[MonitorBounds](#)
[MonitorsCount](#)
[OnLeavingFullScreen](#)
[SetParentWindow](#)
[SetWindowTransparency](#)
[VideoDoubleBuffered](#)
[VideoHeight](#)
[PreferredAspectRatio](#)
[VideoRendererExternal](#)
[VideoRendererExternalIndex](#)
[VideoRendererPriority](#)
[VideoVisibleWhenStopped](#)
[VideoWidth_PREFERREDASPECTRATIO](#)

Created with the Standard Edition of HelpNDoc: [Free EPub producer](#)

Display_VideoPortEnabled

TVideoGrabber.Display_VideoPortEnabled

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables/disables the video port, if available on the current video capture device.

Declaration

property Display_VideoPortEnabled: Boolean **index** vwVideoPortEnabled **read** GetDisplayBoolProperties **write** SetDisplayBoolProperties **default** DEF_Display_VideoPortEnabled;

__property **bool** Display_VideoPortEnabled=read=GetDisplayBoolProperties, write=SetDisplayBoolProperties, index=7, **default**=1

Property Display_VideoPortEnabled As Boolean

Description

If a video port is available on the current video capture device ([IsVideoPortAvailable](#) returns true), and this property is true, the video capture device writes directly to video memory on the preview stream. This provides a smoother display, however it is not possible to draw over video frames on the preview stream, even if the [frame grabber](#) is inserted on the preview stream.

If you need to draw over video frames and get the results visible on the preview window, disable this property.

See Also

[TVideoRenderer](#)
[AdjustPixelAspectRatio](#)
[Display_Active](#)
[Display_AlphaBlendEnabled](#)
[Display_AlphaBlendValue](#)
[Display_AutoSize](#)
[Display_Embedded](#)
[Display_FullScreen](#)
[Display_Height](#)
[Display_Left](#)
[Display_Monitor](#)
[Display_MouseMovesWindow](#)
[Display_PanScanRatio](#)
[Display_SetLocation](#)
[Display_StayOnTop](#)
[Display_Top](#)
[Display_TransparentColorEnabled](#)
[Display_TransparentColorValue](#)
[Display_VideoHeight](#)
[Display_VideoWidth](#)
[Display_VideoWindowHandle](#)
[Display_Width](#)
[DualDisplay_Active](#)
[DualDisplay_AlphaBlendEnabled](#)
[DualDisplay_AlphaBlendValue](#)
[DualDisplay_AutoSize](#)
[DualDisplay_Embedded](#)
[DualDisplay_FullScreen](#)
[DualDisplay_Height](#)
[DualDisplay_Left](#)
[DualDisplay_Monitor](#)
[DualDisplay_MouseMovesWindow](#)
[DualDisplay_PanScanRatio](#)
[DualDisplay_StayOnTop](#)
[DualDisplay_Top](#)
[DualDisplay_TransparentColorEnabled](#)
[DualDisplay_TransparentColorValue](#)
[DualDisplay_VideoHeight](#)
[DualDisplay_VideoPortEnabled](#)
[DualDisplay_VideoWidth](#)
[DualDisplay_VideoWindowHandle](#)
[DualDisplay_Visible](#)
[DualDisplay_Width](#)
[IsVideoPortAvailable](#)
[Monitor_Primary](#)
[Index](#)
[MonitorBounds](#)
[MonitorsCount](#)
[OnLeavingFullScreen](#)
[SetParentWindow](#)
[SetWindowTransparency](#)
[VideoDoubleBuffered](#)
[VideoHeight](#)
[PreferredAspectRatio](#)
[VideoRendererExternal](#)
[VideoRendererExternalIndex](#)
[VideoRendererPriority](#)
[VideoVisibleWhenStopped](#)
[VideoWidth_PREFERREDASPECTRATIO](#)

Created with the Standard Edition of HelpNDoc: [Transform Your Documentation Workflow with HelpNDoc's Intuitive UI](#)

Display_VideoWidth

TVideoGrabber.Display_VideoWidth

[Prev](#)[Next](#)[TVideoGrabber](#) [Properties](#)

Returns the width of the 1st video window.

Declaration

property Display_VideoWidth: LongInt **read** GetDisplay_VideoWidth;

__property **int** Display_VideoWidth=read=GetDisplay_VideoWidth, **nodefault**

Property Display_VideoWidth As Long

Description

Used to retrieve the width of the 1st video window.

See Also

[TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display_Active](#) [Display_AlphaBlendEnabled](#) [Display_AlphaBlendValue](#) [Display_AutoSize](#) [Display_Embedded](#) [Display_FullScreen](#) [Display_Height](#) [Display_Left](#) [Display_Monitor](#) [Display_MouseMovesWindow](#) [Display_PanScanRatio](#) [Display_SetLocation](#) [Display_StayOnTop](#) [Display_Top](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [Display_VideoHeight](#) [Display_VideoPortEnabled](#) [Display_VideoWindowHandle](#) [Display_Width](#) [DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#) [DualDisplay_AlphaBlendValue](#) [DualDisplay_AutoSize](#) [DualDisplay_Embedded](#) [DualDisplay_FullScreen](#) [DualDisplay_Height](#) [DualDisplay_Left](#) [DualDisplay_Monitor](#) [DualDisplay_MouseMovesWindow](#) [DualDisplay_PanScanRatio](#) [DualDisplay_StayOnTop](#) [DualDisplay_Top](#) [DualDisplay_TransparentColorEnabled](#) [DualDisplay_TransparentColorValue](#) [DualDisplay_VideoHeight](#) [DualDisplay_VideoPortEnabled](#) [DualDisplay_VideoWidth](#) [DualDisplay_VideoWindowHandle](#) [DualDisplay_Visible](#) [DualDisplay_Width](#) [IsVideoPortAvailable](#) [Monitor_Primary_Index](#) [MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#) [SetParentWindow](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your CHM Help File Capabilities with HelpNDoc](#)

Display_VideoWindowHandle

TVideoGrabber.Display_VideoWindowHandle

[Prev](#)[Next](#)[TVideoGrabber](#) [Properties](#)

Returns the window handle of the 1st video window.

Declaration

property Display_VideoWindowHandle: Hwnd **read** GetDisplay_VideoWindowHandle;

__property **HWND** Display_VideoWindowHandle=read=GetDisplay_VideoWindowHandle, **nodefault**

Property Display_VideoWindowHandle As Long

Description

Used to retrieve the window handle of the 1st video window.
Returns 0 if the 1st video window does not exist.

See Also

[TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display_Active](#) [Display_AlphaBlendEnabled](#) [Display_AlphaBlendValue](#) [Display_AutoSize](#) [Display_Embedded](#) [Display_FullScreen](#) [Display_Height](#) [Display_Left](#) [Display_Monitor](#) [Display_MouseMovesWindow](#) [Display_PanScanRatio](#) [Display_SetLocation](#)

[Display_StayOnTop](#) [Display_Top](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [Display_VideoHeight](#) [Display_VideoPortEnabled](#) [Display_VideoWidth](#) [Display_Width](#) [DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#) [DualDisplay_AlphaBlendValue](#) [DualDisplay_AutoSize](#) [DualDisplay_Embedded](#) [DualDisplay_FullScreen](#) [DualDisplay_Height](#) [DualDisplay_Left](#) [DualDisplay_Monitor](#) [DualDisplay_MouseMovesWindow](#) [DualDisplay_PanScanRatio](#) [DualDisplay_StayOnTop](#) [DualDisplay_Top](#) [DualDisplay_TransparentColorEnabled](#) [DualDisplay_TransparentColorValue](#) [DualDisplay_VideoHeight](#) [DualDisplay_VideoPortEnabled](#) [DualDisplay_VideoWidth](#) [DualDisplay_VideoWindowHandle](#) [DualDisplay_Visible](#) [DualDisplay_Width](#) [IsVideoPortAvailable](#) [Monitor_Primary_Index](#) [MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#) [SetParentWindow](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Streamline your documentation process with HelpNDoc's WinHelp HLP to CHM conversion feature](#)

Display_Visible

TVideoGrabber.Display_Visible

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Shows / hide the video window

Declaration

property Display_Visible: Boolean **index** vwVisible **read** GetDisplayBoolProperties **write** SetDisplayBoolProperties **default** DEF_Display_Visible;

__property bool Display_Visible=read=GetDisplayBoolProperties, write=SetDisplayBoolProperties, index=1, **default**=1

Property Display_Visible As Boolean

Description

Used to show / hide the video window.
Enabled by default.

Created with the Standard Edition of HelpNDoc: [Maximize Your Productivity with a Help Authoring Tool](#)

Display_Width

TVideoGrabber.Display_Width

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the width of the video window, when it is not embedded in the TVideoGrabber control.

Declaration

property Display_Width: LongInt **index** vwWidth **read** GetDisplayLongIntProperties **write** SetDisplayLongIntProperties **default** DEF_Display_Width;

__property int Display_Width=read=GetDisplayLongIntProperties, write=SetDisplayLongIntProperties, index=11, **default**=320

Property Display_Width As Long

Description

Used to specify the width of the video window, when it is not embedded in the TVideoGrabber control.
Note: the left, top, width and height properties can be set at the same time by using [Display_SetLocation](#) .

See Also

[TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display](#) [Active](#) [Display](#) [AlphaBlendEnabled](#) [Display](#) [AlphaBlendValue](#) [Display](#) [AutoSize](#) [Display](#) [Embedded](#) [Display](#) [FullScreen](#) [Display](#) [Height](#) [Display](#) [Left](#) [Display](#) [Monitor](#) [Display](#) [MouseMovesWindow](#) [Display](#) [PanScanRatio](#) [Display](#) [SetLocation](#) [Display](#) [StayOnTop](#) [Display](#) [Top](#) [Display](#) [TransparentColorEnabled](#) [Display](#) [TransparentColorValue](#) [Display](#) [VideoHeight](#) [Display](#) [VideoPortEnabled](#) [Display](#) [VideoWidth](#) [Display](#) [VideoWindowHandle](#) [DualDisplay](#) [Active](#) [DualDisplay](#) [AlphaBlendEnabled](#) [DualDisplay](#) [AlphaBlendValue](#) [DualDisplay](#) [AutoSize](#) [DualDisplay](#) [Embedded](#) [DualDisplay](#) [FullScreen](#) [DualDisplay](#) [Height](#) [DualDisplay](#) [Left](#) [DualDisplay](#) [Monitor](#) [DualDisplay](#) [MouseMovesWindow](#) [DualDisplay](#) [PanScanRatio](#) [DualDisplay](#) [StayOnTop](#) [DualDisplay](#) [Top](#) [DualDisplay](#) [TransparentColorEnabled](#) [DualDisplay](#) [TransparentColorValue](#) [DualDisplay](#) [VideoHeight](#) [DualDisplay](#) [VideoPortEnabled](#) [DualDisplay](#) [VideoWidth](#) [DualDisplay](#) [VideoWindowHandle](#) [DualDisplay](#) [Visible](#) [DualDisplay](#) [Width](#) [IsVideoPortAvailable](#) [Monitor](#) [Primary](#) [Index](#) [MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#) [SetParentWindow](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Free Qt Help documentation generator](#)

DroppedFrames

TVideoGrabber.DroppedFrames

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Current number of dropped frames.

Declaration

property DroppedFrames: LongInt **read** GetDroppedFrameCount;

__property **int** DroppedFrames=read=GetDroppedFrameCount, **nodefault**

Property DroppedFrames As Long

Description

Used to retrieve the current number of dropped frames since [preview](#) or [recording](#) started.

See Also

[WDM drivers](#) [AutoConnectRelatedPins](#) [CurrentFrameRate](#) [DeliveredFrames](#) [FixFlickerOrBlackCapture](#) [IsDVDevice](#) [IsVideoControlAvailable](#) [IsVideoCrossbarAvailable](#) [IsVideoDeviceConnected](#) [IsWDMVideoDriver](#) [OnNoVideoDevices](#) [OnVideoDeviceSelected](#) [RefreshDevicesAndCompressorsLists](#) [ResetVideoDeviceSettings](#) [ShowDialog](#) [VideoDevice](#) [VideoDeviceIndex](#) [VideoDeviceIndexFromId](#) [VideoDeviceName](#) [VideoDevices](#) [VideoDevicesCount](#) [VideoDevicesId](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Create help files for the Qt Help Framework](#)

DroppedFramesPollingInterval

TVideoGrabber.DroppedFramesPollingInterval

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the polling interval of dropped frames

Declaration

property DroppedFramesPollingInterval: LongInt **read** GetDroppedFramesPollingInterval **write** SetDroppedFramesPollingInterval **default** DEF_DroppedFramesPollingInterval;

__property **int** DroppedFramesPollingInterval=read=GetDroppedFramesPollingInterval, write=SetDroppedFramesPollingInterval, **default**=10

Property DVReduceFrameRate As Long

Description

Used to specify the polling interval of the dropped frames count, expressed in seconds.

Remarks:

DroppedFramesPollingInterval = -1 reads the dropped frames count each time a new frame is received (default value)

DroppedFramesPollingInterval = 0 disables the polling of dropped frames

E.g. DroppedFramesPollingInterval = 30 polls the dropped frames count every 30 seconds.

Created with the Standard Edition of HelpNDoc: [Make CHM Help File Creation a Breeze with HelpNDoc](#)

DualDisplay_Active

TVideoGrabber.DualDisplay_Active

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Same as [Display_Active](#) , but concerns the 2nd video window.

Declaration

property DualDisplay_Active: Boolean **index** vwActive **read** GetDisplay2BoolProperties **write** SetDisplay2BoolProperties **default** DEF_DualDisplay_Active;

__property **bool** DualDisplay_Active=read=GetDisplay2BoolProperties, write=SetDisplay2BoolProperties, index=0, **default**=0

Property DualDisplay_Active As Boolean

Description

Same as [Display_Active](#) , but concerns the 2nd video window.

See Also

[Dual display TVideoRenderer AdjustPixelAspectRatio Display_Active Display_AlphaBlendEnabled Display_AlphaBlendValue Display_AutoSize Display_Embedded Display_FullScreen Display_Height Display_Left Display_Monitor Display_MouseMovesWindow Display_PanScanRatio Display_SetLocation Display_StayOnTop Display_Top Display_TransparentColorEnabled Display_TransparentColorValue Display_VideoHeight Display_VideoPortEnabled Display_VideoWidth Display_VideoWindowHandle Display_Width DualDisplay_AlphaBlendEnabled DualDisplay_AlphaBlendValue DualDisplay_AutoSize DualDisplay_Embedded DualDisplay_FullScreen DualDisplay_Height DualDisplay_Left DualDisplay_Monitor DualDisplay_MouseMovesWindow DualDisplay_PanScanRatio DualDisplay_SetLocation DualDisplay_StayOnTop DualDisplay_Top DualDisplay_TransparentColorEnabled DualDisplay_TransparentColorValue DualDisplay_VideoHeight DualDisplay_VideoPortEnabled DualDisplay_VideoWidth DualDisplay_VideoWindowHandle DualDisplay_Visible DualDisplay_Width IsVideoPortAvailable Monitor_Primary_Index MonitorBounds MonitorsCount OnLeavingFullScreen SetParentWindow SetWindowTransparency VideoDoubleBuffered VideoHeight PreferredAspectRatio VideoRendererExternal VideoRendererExternalIndex VideoRendererPriority VideoVisibleWhenStopped VideoWidth PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Edit and Export Markdown Documents](#)

DualDisplay_AlphaBlendEnabled

TVideoGrabber.DualDisplay_AlphaBlendEnabled

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Same as [Display_AlphaBlendEnabled](#), for the 2nd video window.

Declaration

property DualDisplay_AlphaBlendEnabled: Boolean **index** vwAlphaBlendEnabled **read** GetDualDisplayBoolProperties **write** SetDualDisplayBoolProperties **default** DEF_DualDisplay_AlphaBlendEnabled;

__property bool DualDisplay_AlphaBlendEnabled=read=GetDualDisplayBoolProperties, write=SetDualDisplayBoolProperties, index=0, **default**=1

Property DualDisplay_AlphaBlendEnabled As Boolean

Description

Same as [Display_AlphaBlendEnabled](#), for the 2nd video window.

See Also

[Dual display TVideoRenderer AdjustPixelAspectRatio Display Active Display AlphaBlendEnabled Display AlphaBlendValue Display AutoSize Display Embedded Display FullScreen Display Height Display Left Display Monitor Display MouseMovesWindow Display PanScanRatio Display SetLocation Display StayOnTop Display Top Display TransparentColorEnabled Display TransparentColorValue Display VideoHeight Display VideoPortEnabled Display VideoWidth Display VideoWindowHandle Display Width DualDisplay Active DualDisplay AlphaBlendValue DualDisplay AutoSize DualDisplay Embedded DualDisplay FullScreen DualDisplay Height DualDisplay Left DualDisplay Monitor DualDisplay MouseMovesWindow DualDisplay PanScanRatio DualDisplay SetLocation DualDisplay StayOnTop DualDisplay Top DualDisplay TransparentColorEnabled DualDisplay TransparentColorValue DualDisplay VideoHeight DualDisplay VideoPortEnabled DualDisplay VideoWidth DualDisplay VideoWindowHandle DualDisplay Visible DualDisplay Width IsVideoPortAvailable Monitor Primary Index MonitorBounds MonitorsCount OnLeavingFullScreen SetParentWindow SetWindowTransparency VideoDoubleBuffered VideoHeight PreferredAspectRatio VideoRendererExternal VideoRendererExternalIndex VideoRendererPriority VideoVisibleWhenStopped VideoWidth PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: Effortlessly Create High-Quality Documentation with a Help Authoring Tool

DualDisplay_AlphaBlendValue**TVideoGrabber.DualDisplay_AlphaBlendValue**
[Prev](#)
[Next](#)

[TVideoGrabber](#) **Properties**

Same as [Display_AlphaBlendValue](#), for the 2nd video window.

Declaration

property DualDisplay_AlphaBlendValue: LongInt **index** vwAlphaBlendValue **read** GetDualDisplayLongIntProperties **write** SetDualDisplayLongIntProperties **default** DEF_DualDisplay_AlphaBlendValue;

__property int DualDisplay_AlphaBlendValue=read=GetDualDisplayLongIntProperties, write=SetDualDisplayLongIntProperties, index=12, **default**=240

Property DualDisplay_AlphaBlendValue As Long

Description

Same as [Display_AlphaBlendValue](#), for the 2nd video window.

See Also

[Dual display TVideoRenderer AdjustPixelAspectRatio Display Active Display AlphaBlendEnabled Display AlphaBlendValue Display AutoSize Display Embedded Display FullScreen Display Height Display Left Display Monitor Display MouseMovesWindow Display PanScanRatio Display SetLocation Display StayOnTop Display Top Display TransparentColorEnabled Display TransparentColorValue Display VideoHeight Display VideoPortEnabled Display VideoWidth Display VideoWindowHandle Display Width DualDisplay Active DualDisplay AlphaBlendEnabled DualDisplay AutoSize](#)

[DualDisplay_Embedded](#)
[DualDisplay_FullScreen](#)
[DualDisplay_Height](#)
[DualDisplay_Left](#)
[DualDisplay_Monitor](#)
[DualDisplay_MouseMovesWindow](#)
[DualDisplay_PanScanRatio](#)
[DualDisplay_SetLocation](#)
[DualDisplay_StayOnTop](#)
[DualDisplay_Top](#)
[DualDisplay_TransparentColorEnabled](#)
[DualDisplay_TransparentColorValue](#)
[DualDisplay_VideoHeight](#)
[DualDisplay_VideoPortEnabled](#)
[DualDisplay_VideoWidth](#)
[DualDisplay_VideoWindowHandle](#)
[DualDisplay_Visible](#)
[DualDisplay_Width](#)
[IsVideoPortAvailable](#)
[Monitor_Primary](#)
[Index](#)
[MonitorBounds](#)
[MonitorsCount](#)
[OnLeavingFullScreen](#)
[SetParentWindow](#)
[SetWindowTransparency](#)
[VideoDoubleBuffered](#)
[VideoHeight](#)
[PreferredAspectRatio](#)
[VideoRendererExternal](#)
[VideoRendererExternalIndex](#)
[VideoRendererPriority](#)
[VideoVisibleWhenStopped](#)
[VideoWidth_PreferedAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Documentation Capabilities with HelpNDoc's Project Analyzer](#)

DualDisplay_AspectRatio

TVideoGrabber.DualDisplay_AspectRatio

[Prev](#)
[Next](#)

[TVideoGrabber](#)
[Properties](#)

Same as [Display_AspectRatio](#) , for the 2nd video window.

Declaration

property DualDisplay_AspectRatio: TAspectRatio **index** 1 **read** GetDisplayAspectRatio **write** SetDisplayAspectRatio **default** ar_Box;

__property TAspectRatio DualDisplay_AspectRatio=read=GetDisplayAspectRatio, write=SetDisplayAspectRatio, index=1, **default**=0

Property DualDisplay_AspectRatio As TAspectRatio

Description

Same as [Display_AspectRatio](#) , for the the 2nd video window.

See Also

[Dual display](#)
[TVideoRenderer_AdjustPixelAspectRatio](#)
[Display_Active](#)
[Display_AlphaBlendEnabled](#)
[Display_AlphaBlendValue](#)
[Display_AutoSize](#)
[Display_Embedded](#)
[Display_FullScreen](#)
[Display_Height](#)
[Display_Left](#)
[Display_Monitor](#)
[Display_MouseMovesWindow](#)
[Display_PanScanRatio](#)
[Display_SetLocation](#)
[Display_StayOnTop](#)
[Display_Top](#)
[Display_TransparentColorEnabled](#)
[Display_TransparentColorValue](#)
[Display_VideoHeight](#)
[Display_VideoPortEnabled](#)
[Display_VideoWidth](#)
[Display_VideoWindowHandle](#)
[Display_Width](#)
[DualDisplay_Active](#)
[DualDisplay_AlphaBlendEnabled](#)
[DualDisplay_AlphaBlendValue](#)
[DualDisplay_AutoSize](#)
[DualDisplay_Embedded](#)
[DualDisplay_FullScreen](#)
[DualDisplay_Height](#)
[DualDisplay_Left](#)
[DualDisplay_Monitor](#)
[DualDisplay_MouseMovesWindow](#)
[DualDisplay_PanScanRatio](#)
[DualDisplay_SetLocation](#)
[DualDisplay_StayOnTop](#)
[DualDisplay_Top](#)
[DualDisplay_TransparentColorEnabled](#)
[DualDisplay_TransparentColorValue](#)
[DualDisplay_VideoHeight](#)
[DualDisplay_VideoPortEnabled](#)
[DualDisplay_VideoWidth](#)
[DualDisplay_VideoWindowHandle](#)
[DualDisplay_Visible](#)
[DualDisplay_Width](#)
[IsVideoPortAvailable](#)
[Monitor_Primary](#)
[Index](#)
[MonitorBounds](#)
[MonitorsCount](#)
[OnLeavingFullScreen](#)
[SetParentWindow](#)
[SetWindowTransparency](#)
[VideoDoubleBuffered](#)
[VideoHeight](#)
[PreferredAspectRatio](#)
[VideoRendererExternal](#)
[VideoRendererExternalIndex](#)
[VideoRendererPriority](#)
[VideoVisibleWhenStopped](#)
[VideoWidth_PreferedAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Easily create CHM Help documents](#)

DualDisplay_AutoSize

TVideoGrabber.DualDisplay_AutoSize

[Prev](#)
[Next](#)

[TVideoGrabber](#)
[Properties](#)

Same as [Display_Autosize](#) , but concerns the 2nd video window.

Declaration

property DualDisplay_AutoSize: Boolean **index** vwAutosize **read** GetDisplay2BoolProperties **write** SetDisplay2BoolProperties **default** DEF_DualDisplay_AutoSize;

__property **bool** DualDisplay_AutoSize=read=GetDisplay2BoolProperties,
write=SetDisplay2BoolProperties, index=1, **default**=1

Property DualDisplay_AutoSize As Boolean

Description

Same as [Display_Autosize](#) , but concerns the 2nd video window.

See Also

[Dual display](#) [TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display_Active](#) [Display_AlphaBlendEnabled](#) [Display_AlphaBlendValue](#) [Display_AutoSize](#) [Display_Embedded](#) [Display_FullScreen](#) [Display_Height](#) [Display_Left](#) [Display_Monitor](#) [Display_MouseMovesWindow](#) [Display_PanScanRatio](#) [Display_SetLocation](#) [Display_StayOnTop](#) [Display_Top](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [Display_VideoHeight](#) [Display_VideoPortEnabled](#) [Display_VideoWidth](#) [Display_VideoWindowHandle](#) [Display_Width](#) [DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#) [DualDisplay_AlphaBlendValue](#) [DualDisplay_Embedded](#) [DualDisplay_FullScreen](#) [DualDisplay_Height](#) [DualDisplay_Left](#) [DualDisplay_Monitor](#) [DualDisplay_MouseMovesWindow](#) [DualDisplay_PanScanRatio](#) [DualDisplay_SetLocation](#) [DualDisplay_StayOnTop](#) [DualDisplay_Top](#) [DualDisplay_TransparentColorEnabled](#) [DualDisplay_TransparentColorValue](#) [DualDisplay_VideoHeight](#) [DualDisplay_VideoPortEnabled](#) [DualDisplay_VideoWidth](#) [DualDisplay_VideoWindowHandle](#) [DualDisplay_Visible](#) [DualDisplay_Width](#) [IsVideoPortAvailable](#) [Monitor_Primary](#) [Index_MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#) [SetParentWindow](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Free HTML Help documentation generator](#)

DualDisplay_Embedded

TVideoGrabber.DualDisplay_Embedded

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Same as [Display_Embedded](#) , but concerns the 2nd video window.

Declaration

property DualDisplay_Embedded: Boolean **index** vwEmbedded **read** GetDisplay2BoolProperties **write** SetDisplay2BoolProperties **default** DEF_DualDisplay_Embedded;

__property **bool** DualDisplay_Embedded=read=GetDisplay2BoolProperties,
write=SetDisplay2BoolProperties, index=2, **default**=0

Property DualDisplay_Embedded As Boolean

Description

Same as [Display_Embedded](#) , but concerns the 2nd video window.

See Also

[Dual display](#) [TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display_Active](#) [Display_AlphaBlendEnabled](#) [Display_AlphaBlendValue](#) [Display_AutoSize](#) [Display_Embedded](#) [Display_FullScreen](#) [Display_Height](#) [Display_Left](#) [Display_Monitor](#) [Display_MouseMovesWindow](#) [Display_PanScanRatio](#) [Display_SetLocation](#) [Display_StayOnTop](#) [Display_Top](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [Display_VideoHeight](#) [Display_VideoPortEnabled](#) [Display_VideoWidth](#) [Display_VideoWindowHandle](#) [Display_Width](#) [DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#) [DualDisplay_AlphaBlendValue](#) [DualDisplay_AutoSize](#) [DualDisplay_FullScreen](#) [DualDisplay_Height](#) [DualDisplay_Left](#) [DualDisplay_Monitor](#) [DualDisplay_MouseMovesWindow](#) [DualDisplay_PanScanRatio](#) [DualDisplay_SetLocation](#) [DualDisplay_StayOnTop](#) [DualDisplay_Top](#) [DualDisplay_TransparentColorEnabled](#) [DualDisplay_TransparentColorValue](#) [DualDisplay_VideoHeight](#) [DualDisplay_VideoPortEnabled](#)

[DualDisplay_VideoWidth](#) [DualDisplay_VideoWindowHandle](#) [DualDisplay_Visible](#) [DualDisplay_Width](#) [IsVideoPortAvailable](#) [Monitor_Primary_Index](#) [MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#) [SetParentWindow](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Easily create Qt Help files](#)

DualDisplay_FullScreen

TVideoGrabber.DualDisplay_FullScreen

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Same as [Display_FullScreen](#) , but concerns the 2nd video window.

Declaration

property DualDisplay_FullScreen: Boolean **index** vwFullScreen **read** GetDisplay2BoolProperties **write** SetDisplay2BoolProperties **default** DEF_DualDisplay_FullScreen;

__property **bool** DualDisplay_FullScreen=read=GetDisplay2BoolProperties,
write=SetDisplay2BoolProperties, index=3, **default**=0

Property DualDisplay_FullScreen As Boolean

Description

Same as [Display_FullScreen](#) , but concerns the 2nd video window.

See Also

[Dual display](#) [TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display_Active](#) [Display_AlphaBlendEnabled](#) [Display_AlphaBlendValue](#) [Display_AutoSize](#) [Display_Embedded](#) [Display_FullScreen](#) [Display_Height](#) [Display_Left](#) [Display_Monitor](#) [Display_MouseMovesWindow](#) [Display_PanScanRatio](#) [Display_SetLocation](#) [Display_StayOnTop](#) [Display_Top](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [Display_VideoHeight](#) [Display_VideoPortEnabled](#) [Display_VideoWidth](#) [Display_VideoWindowHandle](#) [Display_Width](#) [DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#) [DualDisplay_AlphaBlendValue](#) [DualDisplay_AutoSize](#) [DualDisplay_Embedded](#) [DualDisplay_Height](#) [DualDisplay_Left](#) [DualDisplay_Monitor](#) [DualDisplay_MouseMovesWindow](#) [DualDisplay_PanScanRatio](#) [DualDisplay_SetLocation](#) [DualDisplay_StayOnTop](#) [DualDisplay_Top](#) [DualDisplay_TransparentColorEnabled](#) [DualDisplay_TransparentColorValue](#) [DualDisplay_VideoHeight](#) [DualDisplay_VideoPortEnabled](#) [DualDisplay_VideoWidth](#) [DualDisplay_VideoWindowHandle](#) [DualDisplay_Visible](#) [DualDisplay_Width](#) [IsVideoPortAvailable](#) [Monitor_Primary_Index](#) [MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#) [SetParentWindow](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Converting Word Documents to eBooks: A Step-by-Step Guide with HelpNDoc](#)

DualDisplay_Height

TVideoGrabber.DualDisplay_Height

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Same as [Display_Height](#) , but concerns the 2nd video window.

Declaration

property DualDisplay_Height: LongInt **index** vwHeight **read** GetDisplay2LongIntProperties **write** SetDisplay2LongIntProperties **default** DEF_DualDisplay_Height;

__property **int** DualDisplay_Height=read=GetDisplay2LongIntProperties,

write=SetDisplay2LongIntProperties, index=12, **default**=240

Property DualDisplay_Height As Long

Description

Same as [Display_Height](#) , but concerns the 2nd video window.

See Also

[Dual display TVideoRenderer AdjustPixelAspectRatio Display Active Display AlphaBlendEnabled Display AlphaBlendValue Display AutoSize Display Embedded Display FullScreen Display Height Display Left Display Monitor Display MouseMovesWindow Display PanScanRatio Display SetLocation Display StayOnTop Display Top Display TransparentColorEnabled Display TransparentColorValue Display VideoHeight Display VideoPortEnabled Display VideoWidth Display VideoWindowHandle Display Width DualDisplay Active DualDisplay AlphaBlendEnabled DualDisplay AlphaBlendValue DualDisplay AutoSize DualDisplay Embedded DualDisplay FullScreen DualDisplay Left DualDisplay Monitor DualDisplay MouseMovesWindow DualDisplay PanScanRatio DualDisplay SetLocation DualDisplay StayOnTop DualDisplay Top DualDisplay TransparentColorEnabled DualDisplay TransparentColorValue DualDisplay VideoHeight DualDisplay VideoPortEnabled DualDisplay VideoWidth DualDisplay VideoWindowHandle DualDisplay Visible DualDisplay Width IsVideoPortAvailable Monitor Primary Index MonitorBounds MonitorsCount OnLeavingFullScreen SetParentWindow SetWindowTransparency VideoDoubleBuffered VideoHeight PreferredAspectRatio VideoRendererExternal VideoRendererExternalIndex VideoRendererPriority VideoVisibleWhenStopped VideoWidth PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [iPhone web sites made easy](#)

DualDisplay_Left

TVideoGrabber.DualDisplay_Left

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Same as [Display_Left](#) , but concerns the 2nd video window.

Declaration

property DualDisplay_Left: LongInt **index** vwLeft **read** GetDisplay2LongIntProperties **write** SetDisplay2LongIntProperties **default** DEF_DualDisplay_Left;

__property **int** DualDisplay_Left=read=GetDisplay2LongIntProperties, write=SetDisplay2LongIntProperties, index=13, **default**=0

Property DualDisplay_Left As Long

Description

Same as [Display_Left](#) , but concerns the 2nd video window.

See Also

[Dual display TVideoRenderer AdjustPixelAspectRatio Display Active Display AlphaBlendEnabled Display AlphaBlendValue Display AutoSize Display Embedded Display FullScreen Display Height Display Left Display Monitor Display MouseMovesWindow Display PanScanRatio Display SetLocation Display StayOnTop Display Top Display TransparentColorEnabled Display TransparentColorValue Display VideoHeight Display VideoPortEnabled Display VideoWidth Display VideoWindowHandle Display Width DualDisplay Active DualDisplay AlphaBlendEnabled DualDisplay AlphaBlendValue DualDisplay AutoSize DualDisplay Embedded DualDisplay FullScreen DualDisplay Height DualDisplay Monitor DualDisplay MouseMovesWindow DualDisplay PanScanRatio DualDisplay SetLocation DualDisplay StayOnTop DualDisplay Top DualDisplay TransparentColorEnabled DualDisplay TransparentColorValue DualDisplay VideoHeight DualDisplay VideoPortEnabled DualDisplay VideoWidth DualDisplay VideoWindowHandle DualDisplay Visible DualDisplay Width IsVideoPortAvailable Monitor Primary Index MonitorBounds MonitorsCount OnLeavingFullScreen SetParentWindow SetWindowTransparency VideoDoubleBuffered VideoHeight PreferredAspectRatio VideoRendererExternal VideoRendererExternalIndex VideoRendererPriority VideoVisibleWhenStopped](#)

[VideoWidth PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Experience the power of a responsive website for your documentation](#)

DualDisplay_Monitor

TVideoGrabber.DualDisplay_Monitor

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Same as [Display_Monitor](#) , but concerns the 2nd video window.

Declaration

property DualDisplay_Monitor: LongInt **index** vwMonitor **read** GetDisplay2LongIntProperties **write** SetDisplay2LongIntProperties **default** DEF_DualDisplay_Monitor;

__property **int** DualDisplay_Monitor=read=GetDisplay2LongIntProperties, write=SetDisplay2LongIntProperties, index=8, **default**=0

Property DualDisplay_Monitor As Long

Description

Same as [Display_Monitor](#) , but concerns the 2nd video window.

See Also

[Dual display TVideoRenderer AdjustPixelAspectRatio Display Active Display AlphaBlendEnabled Display AlphaBlendValue Display AutoSize Display Embedded Display FullScreen Display Height Display Left Display Monitor Display MouseMovesWindow Display PanScanRatio Display SetLocation Display StayOnTop Display Top Display TransparentColorEnabled Display TransparentColorValue Display VideoHeight Display VideoPortEnabled Display VideoWidth Display VideoWindowHandle Display Width DualDisplay Active DualDisplay AlphaBlendEnabled DualDisplay AlphaBlendValue DualDisplay AutoSize DualDisplay Embedded DualDisplay FullScreen DualDisplay Height DualDisplay Left DualDisplay MouseMovesWindow DualDisplay PanScanRatio DualDisplay SetLocation DualDisplay StayOnTop DualDisplay Top DualDisplay TransparentColorEnabled DualDisplay TransparentColorValue DualDisplay VideoHeight DualDisplay VideoPortEnabled DualDisplay VideoWidth DualDisplay VideoWindowHandle DualDisplay Visible DualDisplay Width IsVideoPortAvailable Monitor Primary Index MonitorBounds MonitorsCount OnLeavingFullScreen SetParentWindow SetWindowTransparency VideoDoubleBuffered VideoHeight PreferredAspectRatio VideoRendererExternal VideoRendererExternalIndex VideoRendererPriority VideoVisibleWhenStopped VideoWidth PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your CHM Help File Capabilities with HelpNDoc](#)

DualDisplay_MouseMovesWindow

TVideoGrabber.DualDisplay_MouseMovesWindow

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Same as [Display_MouseMovesWindow](#) , but concerns the 2nd video window.

Declaration

property DualDisplay_MouseMovesWindow: Boolean **index** vwMouseMovesWindow **read** GetDisplay2BoolProperties **write** SetDisplay2BoolProperties **default** DEF_DualDisplay_MouseMovesWindow;

__property **bool** DualDisplay_MouseMovesWindow=read=GetDisplay2BoolProperties, write=SetDisplay2BoolProperties, index=6, **default**=1

Property DualDisplay_MouseMovesWindow As Boolean

Description

Same as [Display_MouseMovesWindow](#) , but concerns the 2nd video window.

See Also

[Dual display TVideoRenderer AdjustPixelAspectRatio Display Active Display AlphaBlendEnabled Display AlphaBlendValue Display AutoSize Display Embedded Display FullScreen Display Height Display Left Display Monitor Display MouseMovesWindow Display PanScanRatio Display SetLocation Display StayOnTop Display Top Display TransparentColorEnabled Display TransparentColorValue Display VideoHeight Display VideoPortEnabled Display VideoWidth Display VideoWindowHandle Display Width DualDisplay Active DualDisplay AlphaBlendEnabled DualDisplay AlphaBlendValue DualDisplay AutoSize DualDisplay Embedded DualDisplay FullScreen DualDisplay Height DualDisplay Left DualDisplay Monitor DualDisplay PanScanRatio DualDisplay SetLocation DualDisplay StayOnTop DualDisplay Top DualDisplay TransparentColorEnabled DualDisplay TransparentColorValue DualDisplay VideoHeight DualDisplay VideoPortEnabled DualDisplay VideoWidth DualDisplay VideoWindowHandle DualDisplay Visible DualDisplay Width IsVideoPortAvailable Monitor Primary Index MonitorBounds MonitorsCount OnLeavingFullScreen SetParentWindow SetWindowTransparency VideoDoubleBuffered VideoHeight PreferredAspectRatio VideoRendererExternal VideoRendererExternalIndex VideoRendererPriority VideoVisibleWhenStopped VideoWidth PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Write eBooks for the Kindle](#)

DualDisplay_PanScanRatio

TVideoGrabber.DualDisplay_PanScanRatio

[Prev](#)

[Next](#)

[TVideoGrabber](#) **Properties**

Same as [Display_AspectRatio](#), but concerns the 2nd video window.

Declaration

property DualDisplay_PanScanRatio: LongInt **index** vwPanScanRatio **read** GetDisplay2LongIntProperties **write** SetDisplay2LongIntProperties **default** DEF_Display_PanScanRatio;

__property int DualDisplay_PanScanRatio==GetDisplay2LongIntProperties, write=SetDisplay2LongIntProperties, index=11, **default**=50;

Property DualDisplay_PanScanRatio as Long

Description

Same as [Display_AspectRatio](#), but concerns the 2nd video window.

See Also

[Dual display TVideoRenderer AdjustPixelAspectRatio Display Active Display AlphaBlendEnabled Display AlphaBlendValue Display AutoSize Display Embedded Display FullScreen Display Height Display Left Display Monitor Display MouseMovesWindow Display PanScanRatio Display SetLocation Display StayOnTop Display Top Display TransparentColorEnabled Display TransparentColorValue Display VideoHeight Display VideoPortEnabled Display VideoWidth Display VideoWindowHandle Display Width DualDisplay Active DualDisplay AlphaBlendEnabled DualDisplay AlphaBlendValue DualDisplay AutoSize DualDisplay Embedded DualDisplay FullScreen DualDisplay Height DualDisplay Left DualDisplay Monitor DualDisplay MouseMovesWindow DualDisplay SetLocation DualDisplay StayOnTop DualDisplay Top DualDisplay TransparentColorEnabled DualDisplay TransparentColorValue DualDisplay VideoHeight DualDisplay VideoPortEnabled DualDisplay VideoWidth DualDisplay VideoWindowHandle DualDisplay Visible DualDisplay Width IsVideoPortAvailable Monitor Primary Index MonitorBounds MonitorsCount OnLeavingFullScreen SetParentWindow SetWindowTransparency VideoDoubleBuffered VideoHeight PreferredAspectRatio VideoRendererExternal VideoRendererExternalIndex VideoRendererPriority VideoVisibleWhenStopped VideoWidth PreferredAspectRatio](#)

DualDisplay_StayOnTop

TVideoGrabber.DualDisplay_StayOnTop

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Same as [Display_StayOnTop](#) , but concerns the 2nd video window.

Declaration

property DualDisplay_StayOnTop: Boolean **index** vwStayOnTop **read** GetDisplay2BoolProperties **write** SetDisplay2BoolProperties **default** DEF_DualDisplay_StayOnTop;

__property **bool** DualDisplay_StayOnTop=read=GetDisplay2BoolProperties, write=SetDisplay2BoolProperties, index=5, **default**=0

Property DualDisplay_StayOnTop As Boolean

Description

Same as [Display_StayOnTop](#) , but concerns the 2nd video window.

See Also

[Dual display TVideoRenderer AdjustPixelAspectRatio Display Active Display AlphaBlendEnabled Display AlphaBlendValue Display AutoSize Display Embedded Display FullScreen Display Height Display Left Display Monitor Display MouseMovesWindow Display PanScanRatio Display SetLocation Display StayOnTop Display Top Display TransparentColorEnabled Display TransparentColorValue Display VideoHeight Display VideoPortEnabled Display VideoWidth Display VideoWindowHandle Display Width DualDisplay Active DualDisplay AlphaBlendEnabled DualDisplay AlphaBlendValue DualDisplay AutoSize DualDisplay Embedded DualDisplay FullScreen DualDisplay Height DualDisplay Left DualDisplay Monitor DualDisplay MouseMovesWindow DualDisplay PanScanRatio DualDisplay SetLocation DualDisplay Top DualDisplay TransparentColorEnabled DualDisplay TransparentColorValue DualDisplay VideoHeight DualDisplay VideoPortEnabled DualDisplay VideoWidth DualDisplay VideoWindowHandle DualDisplay Visible DualDisplay Width IsVideoPortAvailable Monitor Primary Index MonitorBounds MonitorsCount OnLeavingFullScreen SetParentWindow SetWindowTransparency VideoDoubleBuffered VideoHeight PreferredAspectRatio VideoRendererExternal VideoRendererExternalIndex VideoRendererPriority VideoVisibleWhenStopped VideoWidth PreferredAspectRatio](#)

DualDisplay_Top

TVideoGrabber.DualDisplay_Top

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Same as [Display_Top](#) , but concerns the 2nd video window.

Declaration

property DualDisplay_Top: LongInt **index** vwTop **read** GetDisplay2LongIntProperties **write** SetDisplay2LongIntProperties **default** DEF_DualDisplay_Top;

__property **int** DualDisplay_Top=read=GetDisplay2LongIntProperties, write=SetDisplay2LongIntProperties, index=14, **default**=0

Property DualDisplay_Top As Long

Description

Same as [Display_Top](#) , but concerns the 2nd video window.

See Also

[Dual display TVideoRenderer AdjustPixelAspectRatio Display Active Display AlphaBlendEnabled Display AlphaBlendValue Display AutoSize Display Embedded Display FullScreen Display Height Display Left Display Monitor Display MouseMovesWindow Display PanScanRatio Display SetLocation Display StayOnTop Display Top Display TransparentColorEnabled Display TransparentColorValue Display VideoHeight Display VideoPortEnabled Display VideoWidth Display VideoWindowHandle Display Width DualDisplay Active DualDisplay AlphaBlendEnabled DualDisplay AlphaBlendValue DualDisplay AutoSize DualDisplay Embedded DualDisplay FullScreen DualDisplay Height DualDisplay Left DualDisplay Monitor DualDisplay MouseMovesWindow DualDisplay PanScanRatio DualDisplay SetLocation DualDisplay StayOnTop DualDisplay TransparentColorEnabled DualDisplay TransparentColorValue DualDisplay VideoHeight DualDisplay VideoPortEnabled DualDisplay VideoWidth DualDisplay VideoWindowHandle DualDisplay Visible DualDisplay Width IsVideoPortAvailable Monitor Primary Index MonitorBounds MonitorsCount OnLeavingFullScreen SetParentWindow SetWindowTransparency VideoDoubleBuffered VideoHeight PreferredAspectRatio VideoRendererExternal VideoRendererExternalIndex VideoRendererPriority VideoVisibleWhenStopped VideoWidth PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Write eBooks for the Kindle](#)

DualDisplay_TransparentColorEnabled

TVideoGrabber.DualDisplay_TransparentColorEnabled

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Same as [Display_TransparentColorEnabled](#), but for the 2nd video window.

Declaration

property DualDisplay_TransparentColorEnabled: Boolean **index** vwTransparentColorEnabled **read** GetDualDisplayBoolProperties **write** SetDualDisplayBoolProperties **default** DEF_DualDisplay_TransparentColorEnabled;

__property bool DualDisplay_TransparentColorEnabled=read=GetDualDisplayBoolProperties, write=SetDualDisplayBoolProperties, index=0, **default**=1

Property DualDisplay_TransparentColorEnabled As Boolean

Description

Same as [Display_TransparentColorEnabled](#), but for the 2nd video window.

See Also

[Dual display TVideoRenderer AdjustPixelAspectRatio Display Active Display AlphaBlendEnabled Display AlphaBlendValue Display AutoSize Display Embedded Display FullScreen Display Height Display Left Display Monitor Display MouseMovesWindow Display PanScanRatio Display SetLocation Display StayOnTop Display Top Display TransparentColorEnabled Display TransparentColorValue Display VideoHeight Display VideoPortEnabled Display VideoWidth Display VideoWindowHandle Display Width DualDisplay Active DualDisplay AlphaBlendEnabled DualDisplay AlphaBlendValue DualDisplay AutoSize DualDisplay Embedded DualDisplay FullScreen DualDisplay Height DualDisplay Left DualDisplay Monitor DualDisplay MouseMovesWindow DualDisplay PanScanRatio DualDisplay SetLocation DualDisplay StayOnTop DualDisplay Top DualDisplay TransparentColorValue DualDisplay VideoHeight DualDisplay VideoPortEnabled DualDisplay VideoWidth DualDisplay VideoWindowHandle DualDisplay Visible DualDisplay Width IsVideoPortAvailable Monitor Primary Index MonitorBounds MonitorsCount OnLeavingFullScreen SetParentWindow SetWindowTransparency VideoDoubleBuffered VideoHeight PreferredAspectRatio VideoRendererExternal VideoRendererExternalIndex VideoRendererPriority VideoVisibleWhenStopped VideoWidth PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [HelpNDoc's Project Analyzer: Incredible documentation assistant](#)

DualDisplay_TransparentColorValue

TVideoGrabber.DualDisplay_TransparentColorValue

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Same as [Display_TransparentColorValue](#), but for the 2nd video window.

Declaration

property DualDisplay_TransparentColorValue: LongInt **index** vwTransparentColorValue **read** GetDualDisplayLongIntProperties **write** SetDualDisplayLongIntProperties **default** DEF_DualDisplay_TransparentColorValue;

__**property int** DualDisplay_TransparentColorValue=read=GetDualDisplayLongIntProperties, write=SetDualDisplayLongIntProperties, index=12, **default**=240

Property DualDisplay_TransparentColorValue As Long

Description

Same as [Display_TransparentColorValue](#), but for the 2nd video window.

See Also

[Dual display](#) [TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display_Active](#) [Display_AlphaBlendEnabled](#) [Display_AlphaBlendValue](#) [Display_AutoSize](#) [Display_Embedded](#) [Display_FullScreen](#) [Display_Height](#) [Display_Left](#) [Display_Monitor](#) [Display_MouseMovesWindow](#) [Display_PanScanRatio](#) [Display_SetLocation](#) [Display_StayOnTop](#) [Display_Top](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [Display_VideoHeight](#) [Display_VideoPortEnabled](#) [Display_VideoWidth](#) [Display_VideoWindowHandle](#) [Display_Width](#) [DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#) [DualDisplay_AlphaBlendValue](#) [DualDisplay_AutoSize](#) [DualDisplay_Embedded](#) [DualDisplay_FullScreen](#) [DualDisplay_Height](#) [DualDisplay_Left](#) [DualDisplay_Monitor](#) [DualDisplay_MouseMovesWindow](#) [DualDisplay_PanScanRatio](#) [DualDisplay_SetLocation](#) [DualDisplay_StayOnTop](#) [DualDisplay_Top](#) [DualDisplay_TransparentColorEnabled](#) [DualDisplay_VideoHeight](#) [DualDisplay_VideoPortEnabled](#) [DualDisplay_VideoWidth](#) [DualDisplay_VideoWindowHandle](#) [DualDisplay_Visible](#) [DualDisplay_Width](#) [IsVideoPortAvailable](#) [Monitor_Primary](#) [Index](#) [MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#) [SetParentWindow](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Easy EPub and documentation editor](#)

DualDisplay_VideoHeight

TVideoGrabber.DualDisplay_VideoHeight

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Returns the height of the 2nd video window.

Declaration

property DualDisplay_VideoHeight: LongInt **read** GetDualDisplay_VideoHeight;

__**property int** DualDisplay_VideoHeight=read=GetDualDisplay_VideoHeight, **nodefault**

Property DualDisplay_VideoHeight As Long

Description

Used to retrieve the height of the 2nd video window when [dual display](#) is used.

See Also

[Dual display](#) [TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display_Active](#) [Display_AlphaBlendEnabled](#)

[Display_AlphaBlendValue](#) [Display_AutoSize](#) [Display_Embedded](#) [Display_FullScreen](#) [Display_Height](#) [Display_Left](#) [Display_Monitor](#) [Display_MouseMovesWindow](#) [Display_PanScanRatio](#) [Display_SetLocation](#) [Display_StayOnTop](#) [Display_Top](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [Display_VideoHeight](#) [Display_VideoPortEnabled](#) [Display_VideoWidth](#) [Display_VideoWindowHandle](#) [Display_Width](#) [DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#) [DualDisplay_AlphaBlendValue](#) [DualDisplay_AutoSize](#) [DualDisplay_Embedded](#) [DualDisplay_FullScreen](#) [DualDisplay_Height](#) [DualDisplay_Left](#) [DualDisplay_Monitor](#) [DualDisplay_MouseMovesWindow](#) [DualDisplay_PanScanRatio](#) [DualDisplay_SetLocation](#) [DualDisplay_StayOnTop](#) [DualDisplay_Top](#) [DualDisplay_TransparentColorEnabled](#) [DualDisplay_TransparentColorValue](#) [DualDisplay_VideoPortEnabled](#) [DualDisplay_VideoWidth](#) [DualDisplay_VideoWindowHandle](#) [DualDisplay_Visible](#) [DualDisplay_Width](#) [IsVideoPortAvailable](#) [Monitor_Primary](#) [Index_MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#) [SetParentWindow](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth_PREFERREDASPECTRATIO](#)

Created with the Standard Edition of HelpNDoc: [Make Documentation a Breeze with HelpNDoc's Clean and Efficient User Interface](#)

DualDisplay_VideoPortEnabled

TVideoGrabber.DualDisplay_VideoPortEnabled

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Same as [Display_VideoPortEnabled](#) , but concerns the 2nd video window.

Declaration

property DualDisplay_VideoPortEnabled: Boolean **index** vwVideoPortEnabled **read** GetDisplay2BoolProperties **write** SetDisplay2BoolProperties **default** DEF_DualDisplay_VideoPortEnabled;

__property **bool** DualDisplay_VideoPortEnabled=read=GetDisplay2BoolProperties, write=SetDisplay2BoolProperties, index=7, **default**=1

Property DualDisplay_VideoPortEnabled As Boolean

Description

Same as [Display_VideoPortEnabled](#) , but concerns the 2nd video window.

See Also

[Dual display TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display_Active](#) [Display_AlphaBlendEnabled](#) [Display_AlphaBlendValue](#) [Display_AutoSize](#) [Display_Embedded](#) [Display_FullScreen](#) [Display_Height](#) [Display_Left](#) [Display_Monitor](#) [Display_MouseMovesWindow](#) [Display_PanScanRatio](#) [Display_SetLocation](#) [Display_StayOnTop](#) [Display_Top](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [Display_VideoHeight](#) [Display_VideoPortEnabled](#) [Display_VideoWidth](#) [Display_VideoWindowHandle](#) [Display_Width](#) [DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#) [DualDisplay_AlphaBlendValue](#) [DualDisplay_AutoSize](#) [DualDisplay_Embedded](#) [DualDisplay_FullScreen](#) [DualDisplay_Height](#) [DualDisplay_Left](#) [DualDisplay_Monitor](#) [DualDisplay_MouseMovesWindow](#) [DualDisplay_PanScanRatio](#) [DualDisplay_SetLocation](#) [DualDisplay_StayOnTop](#) [DualDisplay_Top](#) [DualDisplay_TransparentColorEnabled](#) [DualDisplay_TransparentColorValue](#) [DualDisplay_VideoHeight](#) [DualDisplay_VideoWidth](#) [DualDisplay_VideoWindowHandle](#) [DualDisplay_Visible](#) [DualDisplay_Width](#) [IsVideoPortAvailable](#) [Monitor_Primary](#) [Index_MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#) [SetParentWindow](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth_PREFERREDASPECTRATIO](#)

Created with the Standard Edition of HelpNDoc: [Easily create Qt Help files](#)

DualDisplay_VideoWidth

TVideoGrabber.DualDisplay_VideoWidth

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

ber

Returns the width of the 2nd video window.

Declaration

property DualDisplay_VideoWidth: LongInt **read** GetDualDisplay_VideoWidth;

__property int DualDisplay_VideoWidth=read=GetDualDisplay_VideoWidth, **nodefault**

Property Display_VideoWidth As Long

Description

Used to retrieve the width of the 2nd video window when [dual display](#) is used.

See Also

[Dual display](#) [TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display](#) [Active](#) [Display](#) [AlphaBlendEnabled](#) [Display](#) [AlphaBlendValue](#) [Display](#) [AutoSize](#) [Display](#) [Embedded](#) [Display](#) [FullScreen](#) [Display](#) [Height](#) [Display](#) [Left](#) [Display](#) [Monitor](#) [Display](#) [MouseMovesWindow](#) [Display](#) [PanScanRatio](#) [Display](#) [SetLocation](#) [Display](#) [StayOnTop](#) [Display](#) [Top](#) [Display](#) [TransparentColorEnabled](#) [Display](#) [TransparentColorValue](#) [Display](#) [VideoHeight](#) [Display](#) [VideoPortEnabled](#) [Display](#) [VideoWidth](#) [Display](#) [VideoWindowHandle](#) [Display](#) [Width](#) [DualDisplay](#) [Active](#) [DualDisplay](#) [AlphaBlendEnabled](#) [DualDisplay](#) [AlphaBlendValue](#) [DualDisplay](#) [AutoSize](#) [DualDisplay](#) [Embedded](#) [DualDisplay](#) [FullScreen](#) [DualDisplay](#) [Height](#) [DualDisplay](#) [Left](#) [DualDisplay](#) [Monitor](#) [DualDisplay](#) [MouseMovesWindow](#) [DualDisplay](#) [PanScanRatio](#) [DualDisplay](#) [SetLocation](#) [DualDisplay](#) [StayOnTop](#) [DualDisplay](#) [Top](#) [DualDisplay](#) [TransparentColorEnabled](#) [DualDisplay](#) [TransparentColorValue](#) [DualDisplay](#) [VideoHeight](#) [DualDisplay](#) [VideoPortEnabled](#) [DualDisplay](#) [VideoWindowHandle](#) [DualDisplay](#) [Visible](#) [DualDisplay](#) [Width](#) [IsVideoPortAvailable](#) [Monitor](#) [Primary](#) [Index](#) [MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#) [SetParentWindow](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: Full-featured Documentation generator

DualDisplay_VideoWindowHandle**TVideoGrabber.DualDisplay_VideoWindowHandle**

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Returns the window handle of the 2nd video window.

Declaration

property DualDisplay_VideoWindowHandle: Hwnd **read** GetDualDisplay_VideoWindowHandle;

__property HWND DualDisplay_VideoWindowHandle=read=GetDualDisplay_VideoWindowHandle, **nodefault**

Property Display_VideoWindowHandle As Long

Description

Used to retrieve the window handle of the 2nd video window.
Returns 0 if the 2nd video window does not exist.

See Also

[Dual display](#) [TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display](#) [Active](#) [Display](#) [AlphaBlendEnabled](#) [Display](#) [AlphaBlendValue](#) [Display](#) [AutoSize](#) [Display](#) [Embedded](#) [Display](#) [FullScreen](#) [Display](#) [Height](#) [Display](#) [Left](#) [Display](#) [Monitor](#) [Display](#) [MouseMovesWindow](#) [Display](#) [PanScanRatio](#) [Display](#) [SetLocation](#) [Display](#) [StayOnTop](#) [Display](#) [Top](#) [Display](#) [TransparentColorEnabled](#) [Display](#) [TransparentColorValue](#) [Display](#) [VideoHeight](#) [Display](#) [VideoPortEnabled](#) [Display](#) [VideoWidth](#) [Display](#) [VideoWindowHandle](#) [Display](#) [Width](#) [DualDisplay](#) [Active](#) [DualDisplay](#) [AlphaBlendEnabled](#) [DualDisplay](#) [AlphaBlendValue](#) [DualDisplay](#) [AutoSize](#) [DualDisplay](#) [Embedded](#) [DualDisplay](#) [FullScreen](#) [DualDisplay](#) [Height](#)

[DualDisplay_Left](#)
[DualDisplay_Monitor](#)
[DualDisplay_MouseMovesWindow](#)
[DualDisplay_PanScanRatio](#)
[DualDisplay_SetLocation](#)
[DualDisplay_StayOnTop](#)
[DualDisplay_Top](#)
[DualDisplay_TransparentColorEnabled](#)
[DualDisplay_TransparentColorValue](#)
[DualDisplay_VideoHeight](#)
[DualDisplay_VideoPortEnabled](#)
[DualDisplay_VideoWidth](#)
[DualDisplay_Visible](#)
[DualDisplay_Width](#)
[IsVideoPortAvailable](#)
[Monitor_Primary_Index](#)
[MonitorBounds](#)
[MonitorsCount](#)
[OnLeavingFullScreen](#)
[SetParentWindow](#)
[SetWindowTransparency](#)
[VideoDoubleBuffered](#)
[VideoHeight](#)
[PreferredAspectRatio](#)
[VideoRendererExternal](#)
[VideoRendererExternalIndex](#)
[VideoRendererPriority](#)
[VideoVisibleWhenStopped](#)
[VideoWidth](#)
[PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly bring your documentation online with HelpNDoc](#)

DualDisplay_Visible

TVideoGrabber.DualDisplay_Visible

[Prev](#)
[Next](#)

[TVideoGrabber](#)
[Properties](#)

Shows / hide the 2nd video window

Declaration

property DualDisplay_Visible: Boolean **index** vwVisible **read** GetDisplay2BoolProperties **write** SetDisplay2BoolProperties **default** DEF_DualDisplay_Visible;

__property bool DualDisplay_Visible=read=GetDisplay2BoolProperties, write=SetDisplay2BoolProperties, index=1, **default**=1

Property DualDisplay_Visible As Boolean

Description

Used to show / hide the 2nd video window when the [dual display](#) feature is used. Enabled by default.

See Also

[Dual display](#)
[TVideoRenderer_AdjustPixelAspectRatio](#)
[Display_Active](#)
[Display_AlphaBlendEnabled](#)
[Display_AlphaBlendValue](#)
[Display_AutoSize](#)
[Display_Embedded](#)
[Display_FullScreen](#)
[Display_Height](#)
[Display_Left](#)
[Display_Monitor](#)
[Display_MouseMovesWindow](#)
[Display_PanScanRatio](#)
[Display_SetLocation](#)
[Display_StayOnTop](#)
[Display_Top](#)
[Display_TransparentColorEnabled](#)
[Display_TransparentColorValue](#)
[Display_VideoHeight](#)
[Display_VideoPortEnabled](#)
[Display_VideoWidth](#)
[Display_VideoWindowHandle](#)
[Display_Width](#)
[DualDisplay_Active](#)
[DualDisplay_AlphaBlendEnabled](#)
[DualDisplay_AlphaBlendValue](#)
[DualDisplay_AutoSize](#)
[DualDisplay_Embedded](#)
[DualDisplay_FullScreen](#)
[DualDisplay_Height](#)
[DualDisplay_Left](#)
[DualDisplay_Monitor](#)
[DualDisplay_MouseMovesWindow](#)
[DualDisplay_PanScanRatio](#)
[DualDisplay_SetLocation](#)
[DualDisplay_StayOnTop](#)
[DualDisplay_Top](#)
[DualDisplay_TransparentColorEnabled](#)
[DualDisplay_TransparentColorValue](#)
[DualDisplay_VideoHeight](#)
[DualDisplay_VideoPortEnabled](#)
[DualDisplay_VideoWidth](#)
[DualDisplay_VideoWindowHandle](#)
[DualDisplay_Width](#)
[IsVideoPortAvailable](#)
[Monitor_Primary_Index](#)
[MonitorBounds](#)
[MonitorsCount](#)
[OnLeavingFullScreen](#)
[SetParentWindow](#)
[SetWindowTransparency](#)
[VideoDoubleBuffered](#)
[VideoHeight](#)
[PreferredAspectRatio](#)
[VideoRendererExternal](#)
[VideoRendererExternalIndex](#)
[VideoRendererPriority](#)
[VideoVisibleWhenStopped](#)
[VideoWidth](#)
[PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Easily create Qt Help files](#)

DualDisplay_Width

TVideoGrabber.DualDisplay_Width

[Prev](#)
[Next](#)

[TVideoGrabber](#)
[Properties](#)

Same as [Display_Width](#) , but concerns the 2nd video window.

Declaration

property DualDisplay_Width: LongInt **index** vwWidth **read** GetDisplay2LongIntProperties **write** SetDisplay2LongIntProperties **default** DEF_DualDisplay_Width;

__property **int** DualDisplay_Width=read=GetDisplay2LongIntProperties, write=SetDisplay2LongIntProperties, index=11, **default**=320

Property DualDisplay_Width As Long

Description

Same as [Display_Width](#) , but concerns the 2nd video window.

See Also

[Dual display](#) [TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display_Active](#) [Display_AlphaBlendEnabled](#) [Display_AlphaBlendValue](#) [Display_AutoSize](#) [Display_Embedded](#) [Display_FullScreen](#) [Display_Height](#) [Display_Left](#) [Display_Monitor](#) [Display_MouseMovesWindow](#) [Display_PanScanRatio](#) [Display_SetLocation](#) [Display_StayOnTop](#) [Display_Top](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [Display_VideoHeight](#) [Display_VideoPortEnabled](#) [Display_VideoWidth](#) [Display_VideoWindowHandle](#) [Display_Width](#) [DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#) [DualDisplay_AlphaBlendValue](#) [DualDisplay_AutoSize](#) [DualDisplay_Embedded](#) [DualDisplay_FullScreen](#) [DualDisplay_Height](#) [DualDisplay_Left](#) [DualDisplay_Monitor](#) [DualDisplay_MouseMovesWindow](#) [DualDisplay_PanScanRatio](#) [DualDisplay_SetLocation](#) [DualDisplay_StayOnTop](#) [DualDisplay_Top](#) [DualDisplay_TransparentColorEnabled](#) [DualDisplay_TransparentColorValue](#) [DualDisplay_VideoHeight](#) [DualDisplay_VideoPortEnabled](#) [DualDisplay_VideoWidth](#) [DualDisplay_VideoWindowHandle](#) [DualDisplay_Visible](#) [IsVideoPortAvailable](#) [Monitor_Primary](#) [Index](#) [MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#) [SetParentWindow](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Revolutionize Your Documentation Review with HelpNDoc's Project Analyzer](#)

DVDateTimeEnabled

TVideoGrabber.DVDateTimeEnabled

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables the reporting of the DV date/time

Declaration

property DVDateTimeEnabled: Boolean **read** GetDVDateTimeEnabled **write** SetDVDateTimeEnabled **default** DEF_DVDateTimeEnabled;

__property **bool** DVDateTimeEnabled=read=GetDVDateTimeEnabled, write=SetDVDateTimeEnabled, **default**=1

Property DVDateTimeEnabled as Boolean

Description

Used to enable/disable the reporting of the date/time for DV sources (e.g. like a camcorder)

This property is enabled by default. Disabling it may save a few CPU load (non-DV sources are not concerned by this setting)

See Also

[TDVSize](#) [TDVVideoFormat](#) [TDVVideoStandard](#) [TOnDVCommandCompleted](#) [DVDDiscontinuityMinimumInterval](#) [DVRecordingInNativeFormatSeparatesStreams](#) [DVReduceFrameRate](#) [DVRgb219](#) [DVTimeCodeEnabled](#) [IsDigitalVideoIn](#) [IsDVCommandAvailable](#) [IsTimeCodeReaderAvailable](#) [OnDVCommandCompleted](#) [SendDVCommand](#)

Created with the Standard Edition of HelpNDoc: [Eliminate the Struggles of Documentation with a Help](#)

DVDDiscontinuityMinimumInterval

TVideoGrabber.DVDDiscontinuityMinimumInterval

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies a minimum interval of time between DV discontinuity notifications.

Declaration

property DVDDiscontinuityMinimumInterval: LongInt **read** GetDVDDiscontinuityMinimumInterval **write** SetDVDDiscontinuityMinimumInterval **default** DEF_DVDDiscontinuityMinimumInterval;

__property **int** DVDDiscontinuityMinimumInterval=read=GetDVDDiscontinuityMinimumInterval, write=SetDVDDiscontinuityMinimumInterval, **default**=3

Property DVDDiscontinuityMinimumInterval As Long

Description

This property specifies a minimum interval of time between the notifications of DV discontinuities detected in the date/time and reported by the [OnDVDDiscontinuity](#) event.

The end of a recording sequence having a duration shorter than this interval will not be notified by the [OnDVDDiscontinuity](#) event.

The value is expressed in seconds.

Default value: 3 seconds

Minimum value: 1 second

See Also

[TDVSize](#) [TDVVideoFormat](#) [TDVVideoStandard](#) [TOnDVCommandCompleted](#) [DVDDateTimeEnabled](#) [DVRRecordingInNativeFormatSeparatesStreams](#) [DVRReduceFrameRate](#) [DVRgb219](#) [DVTimeCodeEnabled](#) [IsDigitalVideoIn](#) [IsDVCommandAvailable](#) [IsTimeCodeReaderAvailable](#) [OnDVCommandCompleted](#) [SendDVCommand](#)

Created with the Standard Edition of HelpNDoc: [Make Help Documentation a Breeze with a Help Authoring Tool](#)

DVDTitle

TVideoGrabber.DVDTitle

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the DVD title to play

Declaration

property DVDTitle: LongInt **read** GetDVDTitle **write** SetDVDTitle **default** DEF_DVDTitle;

__property **int** DVDTitle==GetDVDTitle, write=SetDVDTitle, **default**=0;

Property DVDTitle as Long

Description

Used to specify a DVD title to play, starting from 1.

If DVDTitle = 0 (default) the DVD menu is shown when invoking [OpenPlayer](#).

E.g.:

```
VideoGrabber.PlayerFileName = "E:
VideoGrabber.DVDTitle = 0
VideoGrabber.OpenPlayer()
```

If DVDTitle > 0 the corresponding DVD title is played immediately.

E.g.:

```
VideoGrabber.PlayerFileName = "E:
VideoGrabber.DVDTitle = 1
VideoGrabber.OpenPlayer()
```

See Also

[DVDInfo](#) [OpenDVD](#) [PlayerFileName](#)

Created with the Standard Edition of HelpNDoc: [Bring your WinHelp HLP help files into the present with HelpNDoc's easy CHM conversion](#)

DVEncoder_VideoFormat

TVideoGrabber.DVEncoder_VideoFormat

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the DV encoding format.

Declaration

property DVEncoder_VideoFormat: TDVVideoFormat **read** GetDVEncoder_VideoFormat **write** SetDVEncoder_VideoFormat;

__property TDVVideoFormat DVEncoder_VideoFormat=read=GetDVEncoder_VideoFormat, write=SetDVEncoder_VideoFormat, **nodefault**

Property DVEncoder_VideoFormat As TxDVVideoFormat

Description

Used to specify the DV encoding format when sending the audio/video streams to a DV device (when [RecordingMethod](#) is rm_SendToDV).
The value is a [TDVVideoFormat](#) type.

See Also

[Send to DV](#) [DVEncoder](#) [VideoResolution](#) [DVEncoder](#) [VideoStandard](#) [SendToDV](#) [DeviceIndex](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Productivity with a Help Authoring Tool](#)

DVEncoder_VideoResolution

TVideoGrabber.DVEncoder_VideoResolution

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the DV encoding resolution.

Declaration

property DVEncoder_VideoResolution: TDVSize **read** GetDVEncoder_VideoResolution **write** SetDVEncoder_VideoResolution;

__property TDVSize DVEncoder_VideoResolution=read=GetDVEncoder_VideoResolution,
write=SetDVEncoder_VideoResolution, **nodefault**

Property DVEncoder_VideoResolution As TxDVSize

Description

Used to specify the DV encoding resolution when sending the audio/video streams to a DV device (when [RecordingMethod](#) is rm_SendToDV).

The value is [TDVSize](#) a type.

See Also

[Send to DV](#) [DVEncoder_VideoFormat](#) [DVEncoder_VideoStandard](#) [SendToDV_DeviceIndex](#)

Created with the Standard Edition of HelpNDoc: [Transform Your CHM Help File Creation Process with HelpNDoc](#)

DVEncoder_VideoStandard

TVideoGrabber.DVEncoder_VideoStandard

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the DV encoding standard.

Declaration

property DVEncoder_VideoStandard: TDVVideoStandard **read** GetDVEncoder_VideoStandard **write** SetDVEncoder_VideoStandard;

__property TDVVideoStandard DVEncoder_VideoStandard=read=GetDVEncoder_VideoStandard,
write=SetDVEncoder_VideoStandard, **nodefault**

Property DVEncoder_VideoStandard As TxDVVideoStandard

Description

Used to specify the DV encoding standard (PAL or NTSC) when sending the audio/video streams to a DV device (when [RecordingMethod](#) is rm_SendToDV).

The value is a [TDVVideoStandard](#) type.

See Also

[Send to DV](#) [DVEncoder_VideoFormat](#) [DVEncoder_VideoResolution](#) [SendToDV_DeviceIndex](#)

Created with the Standard Edition of HelpNDoc: [Transform Your Word Doc into a Professional-Quality eBook with HelpNDoc](#)

DVRecordingInNativeFormatSeparatesStreams

TVideoGrabber.DVRecordingInNativeFormatSeparatesStreams

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specify if the DV audio and video streams must be stored as separated streams.

Declaration

property DVRecordingInNativeFormatSeparatesStreams: Boolean **read** GetDVRecordingInNativeFormatSeparatesStreams **write** SetDVRecordingInNativeFormatSeparatesStreams **default** true;

__property **bool** DVRecordingInNativeFormatSeparatesStreams=read=GetDVRecordingInNativeFormatSeparatesStreams,

write=SetDVRecordingInNativeFormatSeparatesStreams, **default=1**

Property DVRecordingInNativeFormatSeparatesStreams As Boolean

Description

When recording DV audio+video in native format in an AVI file ([RecordingInNativeFormat](#) enabled and [RecordingMethod](#) = rm_AVI), this property specifies if the DV audio and video streams must be stored as 2 separated audio and video streams, or as a single A/V interleaved stream:

- if enabled, the DV audio and video are stored within the AVI as 2 separated streams: a video stream of type "dvsd", and an audio stream of type "auds".
- if disabled the DV audio and video are stored as a single interleaved audio and video stream of type "iavs".

See Also

[TDVSize](#) [TDVVideoFormat](#) [TDVVideoStandard](#) [TOnDVCommandCompleted](#) [DVDateTimeEnabled](#) [DVDDiscontinuityMinimumInterval](#) [DVReduceFrameRate](#) [DVRgb219](#) [DVTimeCodeEnabled](#) [IsDigitalVideoIn](#) [IsDVCommandAvailable](#) [IsTimeCodeReaderAvailable](#) [OnDVCommandCompleted](#) [SendDVCommand](#)

Created with the Standard Edition of HelpNDoc: [News and information about help authoring tools and software](#)

DVReduceFrameRate

TVideoGrabber.DVReduceFrameRate

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Reduces the frame rate of the DV video stream.

Declaration

property DVReduceFrameRate: Boolean **read** GetDVReduceFrameRate **write** SetDVReduceFrameRate **default** DEF_DVReduceFrameRate;

__property bool DVReduceFrameRate=read=GetDVReduceFrameRate, write=SetDVReduceFrameRate, **default=0**

Property DVReduceFrameRate As Boolean

Description

Used to reduce the frame rate of the DV video stream.

If enabled, half of the frames are discarded in the video stream:

- for NTSC, the frame rate is reduced from 30 frames per second (fps) to 15 fps,
- for PAL, the frame rate is reduced from 25 fps to 12.5 fps.

This property can be used only if the current video capture device source is DV ([IsDigitalVideoIn](#) returns true).

See Also

[Frame rate](#) [TDVSize](#) [TDVVideoFormat](#) [TDVVideoStandard](#) [TOnDVCommandCompleted](#) [DVDateTimeEnabled](#) [DVDDiscontinuityMinimumInterval](#) [DVRecordingInNativeFormatSeparatesStreams](#) [DVRgb219](#) [DVTimeCodeEnabled](#) [FrameRate](#) [FramerateDivider](#) [IsDigitalVideoIn](#) [IsDVCommandAvailable](#) [IsTimeCodeReaderAvailable](#) [OnDVCommandCompleted](#) [SendDVCommand](#)

Created with the Standard Edition of HelpNDoc: [Transform Your Documentation Process with HelpNDoc's Project Analyzer](#)

DVRgb219

TVideoGrabber.DVRgb219

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Disables the DV color range stretching.

Declaration

property DVRgb219: Boolean **read** GetDVRgb219 **write** SetDVRgb219 **default** DEF_DVRgb219;

__property **bool** DVRgb219=read=GetDVRgb219, write=SetDVRgb219, **default**=0

Property DVRgb219 As Boolean

Description

Used to disable the color range stretching of DV video streams going through the DV decoder. The DV video format has a dynamic range of 16-235. By default, when the DV Video Decoder decodes to 24-bit or 32-bit RGB, by stretching the color range from 0-255. In RGB-219 mode, the decoder does not stretch the color range. This property can be used only if the current video capture device source is DV ([IsDigitalVideoIn](#) returns true).

See Also

[TDVSize](#) [TDVVideoFormat](#) [TDVVideoStandard](#) [TOnDVCommandCompleted](#) [DVDateTimeEnabled](#) [DVDContinuityMinimumInterval](#) [DVRecordingInNativeFormatSeparatesStreams](#) [DVReduceFrameRate](#) [DVTimeCodeEnabled](#) [IsDigitalVideoIn](#) [IsDVCommandAvailable](#) [IsTimeCodeReaderAvailable](#) [OnDVCommandCompleted](#) [SendDVCommand](#)

Created with the Standard Edition of HelpNDoc: [Easily create Web Help sites](#)

DVTimeCodeEnabled

TVideoGrabber.DVTimeCodeEnabled

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables/disable the reporting of the DV TimeCode

Declaration

property DVTimeCodeEnabled: Boolean **read** GetDVTimeCodeEnabled **write** SetDVTimeCodeEnabled **default** DEF_DVTimeCodeEnabled;

__property **bool** DVTimeCodeEnabled=read=GetDVTimeCodeEnabled, write=SetDVTimeCodeEnabled, **default**=1

Property DVTimeCodeEnabled as Boolean

Description

Used to enable/disable the reporting of the DV TimeCode for DV sources (e.g. like a camcorder)

This property is enabled by default. Disabling it may save a few CPU load (non-DV sources are not concerned by this setting)

See Also

[TDVSize](#) [TDVVideoFormat](#) [TDVVideoStandard](#) [TOnDVCommandCompleted](#) [DVDateTimeEnabled](#) [DVDContinuityMinimumInterval](#) [DVRecordingInNativeFormatSeparatesStreams](#) [DVReduceFrameRate](#) [DVRgb219](#) [IsDigitalVideoIn](#) [IsDVCommandAvailable](#) [IsTimeCodeReaderAvailable](#) [OnDVCommandCompleted](#) [SendDVCommand](#)

Created with the Standard Edition of HelpNDoc: [Elevate Your CHM Help Files with HelpNDoc's Advanced Customization Options](#)

EncryptionMethod

TVideoGrabber.EncryptionMethod

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Properties](#)

Specifies the encryption method

Declaration

property EncryptionMethod: TEncryptionMethod **read** GetEncryptionMethod **write** SetEncryptionMethod
default DEF_EncryptionMethod;

__**property** TEncryptionMethod EncryptionMethod=read=GetEncryptionMethod,
 write=SetEncryptionMethod, **default**=0

Description

Specifies the encryption method enabled by [SetEncryptionKey](#)

See Also

[TEncryptionMethod](#) [Decrypt File](#) [Encrypt File](#) [SetDecryptionKey](#) [SetEncryptionKey](#)

Created with the Standard Edition of HelpNDoc: [Free EBook and documentation generator](#)

EventNotificationSynchronone

TVideoGrabber.EventNotificationSynchronone

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Properties](#)

Speficies if the events are notified synchronously.

Declaration

property EventNotificationSynchronone: Boolean **read** GetEventNotificationSynchronone **write**
 SetEventNotificationSynchronone **default** DEF_EventNotificationSynchronone;

__**property** **bool** EventNotificationSynchronone=read=GetEventNotificationSynchronone,
 write=SetEventNotificationSynchronone, **default**=0

Property EventNotificationSynchronone As Boolean

Description

Used to specify if the events occur synchronously.

If you invoke TVideoGrabber from a thread you can disable this property.

See also [SyncCommands](#).

Created with the Standard Edition of HelpNDoc: [Write EPub books for the iPad](#)

ExtraDLLPath

TVideoGrabber.ExtraDLLPath

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Properties](#)

Path of the binaries of the external Datastead DirectShow filters

Declaration

property ExtraDLLPath: **string** **read** GetExtraDLLPath **write** SetExtraDLLPath;

`__property wchar_t *ExtraDLLPath==GetExtraDLLPath, write=SetExtraDLLPath;`

Description

Specifies the location of the x86/x64 folders of the binaries of the Datastead DirectShow filters.

It is possible to invoke the [Datastead RTSP/RTMP/HTTP/ONVIF DirectShow source filter](#) or [Datastead Multipurpose Encoder](#) from the TVideoGrabber SDK just by specifying to TVideoGrabber the location of the binaries, **without having to run the .exe installer, and without having to register the filter(s) with regsvr32.exe.**

Just copy the x86 and x64 folders (containing the binaries of the filter) to a folder of your choice, and set ExtraDLLPath = this folder.

E.g. if the "x86" and "x64" folders have been copied to c:/datasteadfilter, set VideoGrabber.ExtraDLLPath = "c:/datasteadfilter"

Example:

```
VideoGrabber.VideoSource = vs_IPCamera
VideoGrabber.IPCameraURL = "rtsp://..."
VideoGrabber.ExtraDLLPath = "c:/datasteadfilter"
VideoGrabber.StartPreview();
```

It is also possible to specify a different DLL path per filter, by adding the **[RTSP]** or **[MPE]** prefix at the beginning, e.g.:

```
VideoGrabber.ExtraDLLPath = "[RTSP]c:/RTSPfolder1
VideoGrabber.ExtraDLLPath = "[MPE]c:/MPEfolder2
```

See Also

[IR Cut Filter of Axis cameras](#) [TDiscoveryCallbackStatus](#)
[TOnONVIFDiscoveryCompletedNotification](#) [TONVIFDeviceInfo](#) [Video stream of IP camera with audio of PC microphone or other audio capture device](#)
[GetLastError](#) [IPCameraURL](#) [IsURLResponding](#) [IsURLVideoStreamAvailable](#)
[OnONVIFDiscoveryCompletedNotification](#) [ONVIF_GetStr](#) [ONVIF_SetStr](#)
[ONVIFCancelDiscovery](#) [ONVIFDeviceInfo](#) [ONVIFDiscoverCameras_IPRange](#)
[ONVIFDiscoverCameras_Multicast](#) [ONVIFEnumCamerasDiscovered](#) [ONVIFPTZGetLimits](#)
[ONVIFPTZGetPosition](#) [ONVIFPTZPreset](#) [ONVIFPTZSendAuxiliaryCommand](#)
[ONVIFPTZSetPosition](#) [ONVIFPTZStartMove](#) [ONVIFPTZStopMove](#) [ONVIFSnapShot](#)
[OpenURLAsync](#) [OpenURLAsyncStatus](#) [SendIPCameraCommand](#) [SetDatasteadFilterDllName](#)
[SetIPCameraSetting](#)

Created with the Standard Edition of HelpNDoc: [Don't be left in the past: convert your WinHelp HLP help files to CHM with HelpNDoc](#)

FixFlickerOrBlackCapture

TVideoGrabber.FixFlickerOrBlackCapture

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Used to fix display problems with some video capture devices.

Declaration

property FixFlickerOrBlackCapture: Boolean **read** GetFixFlickerOrBlackCapture **write** SetFixFlickerOrBlackCapture **default** DEF_FixFlickerOrBlackCapture;

`__property bool FixFlickerOrBlackCapture=read=GetFixFlickerOrBlackCapture, write=SetFixFlickerOrBlackCapture, default=0`

Property FixFlickerOrBlackCapture As Boolean

Description

This property is used to fix the possible following problems with some video capture devices:

- flickering,
- black frame capture when invoking CaptureFrameTo,
- excessive apparent frame rate.

Try to enable this property if you get this kind of problem.

Enabling this property requires a bit more CPU, so it should be kept enabled **only** if it fixes these kind of problems.

See Also

[WDM drivers](#) [AutoConnectRelatedPins](#) [CurrentFrameRate](#) [DeliveredFrames](#) [DroppedFrames](#) [IsDVDevice](#) [IsVideoControlAvailable](#) [IsVideoCrossbarAvailable](#) [IsVideoDeviceConnected](#) [IsWDMVideoDriver](#) [OnNoVideoDevices](#) [OnVideoDeviceSelected](#) [RefreshDevicesAndCompressorsLists](#) [ResetVideoDeviceSettings](#) [ShowDialog](#) [VideoDevice](#) [VideoDeviceIndex](#) [VideoDeviceIndexFromId](#) [VideoDeviceName](#) [VideoDevices](#) [VideoDevicesCount](#) [VideoDevicesId](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Bring your WinHelp HLP help files into the present with HelpNDoc's easy CHM conversion](#)

FrameCaptureHeight

TVideoGrabber.FrameCaptureHeight

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies a height for the captured frame.

Declaration

property FrameCaptureHeight: LongInt read GetFrameCaptureHeight write SetFrameCaptureHeight default DEF_FrameCaptureHeight;

__property int FrameCaptureHeight==GetFrameCaptureHeight, write=SetFrameCaptureHeight, default=-1;

Property FrameCaptureHeight as Long

Description

Used to specify a height for the captured frame when capturing frames with [CaptureFrameTo](#) or the [burst mode](#).

Default value = -1 (disabled)

If a value > 0 is specified, the captured frame will be stretched to the specified height, otherwise the height of the video frame will be used.

See Also

[TAutoFileName](#) [TFrameCaptureDest](#) [TOnFrameCaptureCompleted](#) [BurstCount](#) [BurstInterval](#) [BurstMode](#) [BurstType](#) [CaptureFrameSyncTo](#) [CaptureFrameTo](#) [FrameCaptureWidth](#) [FrameCaptureWithoutOverlay](#) [FrameCaptureZoomSize](#) [GetFrameInfoString](#) [GetLastFrameAsHBITMAP](#) [GetLastFrameAsTBitmap](#) [GetLastFrameBitmapBits](#) [GetLastFrameBitmapBits2](#) [GetLastFrameWaitTimeoutMs](#) [JPEGPerformance](#) [JPEGProgressiveDisplay](#) [JPEGQuality](#) [Last](#) [BurstFrameCapture](#) [FileName](#) [Last](#) [CaptureFrameTo](#) [FileName](#) [OnDiskFull](#) [OnFrameBitmap](#) [SetFrameCaptureBounds](#) [StoragePath](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Full-featured Documentation generator](#)

FrameCaptureWidth

TVideoGrabber.FrameCaptureWidth

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies a width for the captured frame.

Declaration

property FrameCaptureWidth: LongInt read GetFrameCaptureWidth write SetFrameCaptureWidth default DEF_FrameCaptureWidth;

```
__property int FrameCaptureWidth==GetFrameCaptureWidth, write=SetFrameCaptureWidth, default=-1;
```

Property FrameCaptureWidth as Long

Description

Used to specify a width for the captured frame when capturing frames with [CaptureFrameTo](#) or the [burst mode](#).

Default value = -1 (disabled)

If a value > 0 is specified, the captured frame will be stretched to the specified width, otherwise the width of the video frame will be used.

See Also

[TAutoFileName](#) [TFrameCaptureDest](#) [TOnFrameCaptureCompleted](#) [BurstCount](#) [BurstInterval](#) [BurstMode](#) [BurstType](#) [CaptureFrameSyncTo](#) [CaptureFrameTo](#) [FrameCaptureHeight](#) [FrameCaptureWithoutOverlay](#) [FrameCaptureZoomSize](#) [GetFrameInfoString](#) [GetLastFrameAsHBITMAP](#) [GetLastFrameAsTBitmap](#) [GetLastFrameBitmapBits](#) [GetLastFrameBitmapBits2](#) [GetLastFrameWaitTimeoutMs](#) [JPEGPerformance](#) [JPEGProgressiveDisplay](#) [JPEGQuality](#) [Last](#) [BurstFrameCapture](#) [FileName](#) [Last](#) [CaptureFrameTo](#) [FileName](#) [OnDiskFull](#) [OnFrameBitmap](#) [SetFrameCaptureBounds](#) [StoragePath](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Streamline Your Documentation Process with HelpNDoc's Project Analyzer](#)

FrameCaptureWithoutOverlay

TVideoGrabber.FrameCaptureWithoutOverlay

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies whether [frame capture](#) functions must capture unmodified frames when drawing over video frames.

Declaration

property FrameCaptureWithoutOverlay: Boolean **read** GetFrameCaptureWithoutOverlay **write** SetFrameCaptureWithoutOverlay **default** DEF_FrameCaptureWithoutOverlay;

```
__property bool FrameCaptureWithoutOverlay=read=GetFrameCaptureWithoutOverlay, write=SetFrameCaptureWithoutOverlay, default=0
```

Property FrameCaptureWithoutOverlay As Boolean

Description

Used to specify whether [frame capture](#) must be performed before drawing over video frames (unmodified video frames) or after drawing over video frames (modified video frames).

- if FrameCaptureBeforeDrawing is disabled (default), frame overlay affects the captured frames as well as preview and recording files.

-if FrameCaptureBeforeDrawing is enabled, the frames captured by [CaptureFrameTo](#) or [in burst mode](#) concerns unmodified native frames.

In this case text, shapes, lines or bitmap drawn over video frames appear only on the preview window or recording files.

See Also

[TAutoFileName](#) [TFrameCaptureDest](#) [TOnFrameCaptureCompleted](#) [BurstCount](#) [BurstInterval](#) [BurstMode](#) [BurstType](#) [CaptureFrameSyncTo](#) [CaptureFrameTo](#) [FrameCaptureHeight](#) [FrameCaptureWidth](#) [FrameCaptureZoomSize](#) [GetFrameInfoString](#) [GetLastFrameAsHBITMAP](#) [GetLastFrameAsTBitmap](#) [GetLastFrameBitmapBits](#) [GetLastFrameBitmapBits2](#) [GetLastFrameWaitTimeoutMs](#) [JPEGPerformance](#) [JPEGProgressiveDisplay](#) [JPEGQuality](#) [Last](#) [BurstFrameCapture](#) [FileName](#) [Last](#) [CaptureFrameTo](#) [FileName](#) [OnDiskFull](#) [OnFrameBitmap](#) [SetFrameCaptureBounds](#) [StoragePath](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Elevate Your CHM Help Files with HelpNDoc's Advanced Customization Options](#)

FrameCaptureZoomSize

TVideoGrabber.FrameCaptureZoomSize

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the size of the captured video frames, in percent of the [video source size](#) .

Declaration

property FrameCaptureZoomSize: LongInt **read** GetFrameCaptureZoomSize **write** SetFrameCaptureZoomSize **default** DEF_FrameCaptureZoomSize;

__property **int** FrameCaptureZoomSize=read=GetFrameCaptureZoomSize, write=SetFrameCaptureZoomSize, **default**=100

Property FrameCaptureZoomSize As Long

Description

Used modify the size of video frames captured using the [frame capture](#) .

The captured frames will be stretched to the specified percentage of the [video source width](#) and [video source height](#) .

Default value: 100 (percent).

See Also

[TAutoFileName](#) [TFrameCaptureDest](#) [TOnFrameCaptureCompleted](#) [BurstCount](#) [BurstInterval](#) [BurstMode](#) [BurstType](#) [CaptureFrameSyncTo](#) [CaptureFrameTo](#) [FrameCaptureHeight](#) [FrameCaptureWidth](#) [FrameCaptureWithoutOverlay](#) [GetFrameInfoString](#) [GetLastFrameAsHBITMAP](#) [GetLastFrameAsTBitmap](#) [GetLastFrameBitmapBits](#) [GetLastFrameBitmapBits2](#) [GetLastFrameWaitTimeoutMs](#) [JPEGPerformance](#) [JPEGProgressiveDisplay](#) [JPEGQuality](#) [Last](#) [BurstFrameCapture](#) [FileName](#) [Last](#) [CaptureFrameTo](#) [FileName](#) [OnDiskFull](#) [OnFrameBitmap](#) [SetFrameCaptureBounds](#) [StoragePath](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Free CHM Help documentation generator](#)

FrameGrabber

TVideoGrabber.FrameGrabber

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables and configure or disables the frame grabber, that allows to capture frames and draw over frames.

Declaration

property FrameGrabber: TFrameGrabber **read** GetFrameGrabber **write** SetFrameGrabber **default** DEF_FrameGrabber;

__property TFrameGrabber FrameGrabber=read=GetFrameGrabber, write=SetFrameGrabber, **default**=0

Property FrameGrabber As TFrameGrabber

Description

Used to enable or disable the frame grabber.

Values:

fg_BothStreams: the frame grabber is applied on both preview stream and capture streams,

fg_PreviewStream: the frame grabber is applied only on the preview stream,

fg_CaptureStream: the frame grabber is applied only on the capture stream,

fg_Disabled: the frame grabber is disabled.

Overview

The frame grabber is a RGB filter that lets you capture frames and/or perform graphic and text overlays.

The frame grabber allows to:

- capture video frames: see the [Frame capture](#) ,
- draw (or write text) over video frames: see [Drawing over video frames](#) .

- if **FrameGrabber = fg_BothStreams**, the overlays are applied to the video window (preview) as well as to the captured frames and the video clip being recorded)

- if **FrameGrabber = fg_PreviewStream**, the overlays are applied only on the video window (but not on the captured frames or the video clip being recorded)

- if **FrameGrabber = fg_CaptureStream**, the overlay are applied to the captured frames and the video clip being recorded, but are not visible on the video window.

If you want to capture frames without applying the overlays (but you want the overlays in the preview and/or the recorded video clip), enable the [FrameCaptureWithoutOverlay](#) property.

Recording

When it is used on a stream (preview, capture or both), this stream is converted into RGB format.

By default the recording is made in the native format of the video source when [RecordingInNativeFormat](#) is enabled (default)

If you need to apply the overlays to the recorded clip, disable the [RecordingInNativeFormat](#). Then the recording will performed in uncompressed RGB with the overlays applied.

You can compress the video stream with the overlays applied by selecting an audio or video compressor, according to the [CompressionMode](#) and [CompressionType](#) settings.

During recording, the frame grabber can be applied on the recording stream, the preview stream, or both.

If you want to preserve the recording format and need however to capture frames, you can either:

- apply the frame grabber only on the preview stream (FrameGrabber = fg_PreviewStream).
- enable the [RecordingInNativeFormat](#) that will save the clip in the native (unmodified) format coming out of the video source.

Note: using the frame grabber requires more CPU, so it should be disabled if it is not required to capture video frames or draw over them during recording.

See Also

[TFrameGrabberRGBFormat](#) [FrameGrabberCurrentRGBFormat](#) [FrameGrabberRGBFormat](#) [FramerateDivider](#) [GetFrameInfo](#) [GetFrameInfoString](#) [InFrameProgressEvent](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Revolutionize Your Documentation Output with a Help Authoring Tool](#)

FrameGrabberCurrentRGBFormat

TVideoGrabber.FrameGrabberCurrentRGBFormat

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Returns the current RGB format of the frame grabber.

Declaration

property FrameGrabberCurrentRGBFormat: TFrameGrabberRGBFormat **read** GetFrameGrabberCurrentRGBFormat;

__property TFrameGrabberRGBFormat
FrameGrabberCurrentRGBFormat=read=GetFrameGrabberCurrentRGBFormat, **nodefault**

Property FrameGrabberCurrentRGBFormat As TFrameGrabberRGBFormat

Description

Used to retrieve the current RGB format of the frame grabber.
Useful mainly when the frame grabber has no specified RGB format ([FrameGrabberRGBFormat](#) = fgf_Default), to retrieve the current RGB format in use after invoking [StartPreview](#), [StartRecording](#) or [OpenPlayer](#).

See Also

[TFrameGrabberRGBFormat](#) [FrameGrabber](#) [FrameGrabberRGBFormat](#) [FramerateDivider](#) [GetFrameInfo](#) [GetFrameInfoString](#) [InFrameProgressEvent](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Easy EPub and documentation editor](#)

FrameGrabberRGBFormat

TVideoGrabber.FrameGrabberRGBFormat

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Format of the video frames captured by [CaptureFrameTo](#) or in [burst mode](#) .

Declaration

property FrameGrabberRGBFormat: TFrameGrabberRGBFormat **read** GetFrameGrabberRGBFormat **write** SetFrameGrabberRGBFormat **default** DEF_FrameGrabberRGBFormat;

__property TFrameGrabberRGBFormat FrameGrabberRGBFormat=read=GetFrameGrabberRGBFormat, write=SetFrameGrabberRGBFormat, **default**=0

Property FrameGrabberRGBFormat As TFrameGrabberRGBFormat

Description

Used to specify the RGB format (see [TFrameGrabberRGBFormat](#)) of the frame grabber.
This is the RGB format used for frame capture, and also for text and graphics overlays.
the format is fgf_Default (you let TVideoGrabber choose the best RGB format), the real RGB format in use will be returned by the [FrameGrabberCurrentRGBFormat](#) property.

See Also

[TFrameGrabberRGBFormat](#) [FrameGrabber](#) [FrameGrabberCurrentRGBFormat](#) [FramerateDivider](#) [GetFrameInfo](#) [GetFrameInfoString](#) [InFrameProgressEvent](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Transform Your Documentation Process with HelpNDoc's Project Analyzer](#)

FrameNumberStartsFromZero

TVideoGrabber.FrameNumberStartsFromZero

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies if the frame counting must start from 0

Declaration

property FrameNumberStartsFromZero: Boolean **read** GetFrameNumberStartsFromZero **write** SetFrameNumberStartsFromZero **default** DEF_FrameNumberStartsFromZero;

__**property bool** FrameNumberStartsFromZero=read=GetFrameNumberStartsFromZero, write=SetFrameNumberStartsFromZero, **default**=0

property FrameNumberStartsFromZero As Boolean

Description

Used to specify if the frame counting must start from 0.

(disabled by default, in this case the 1st frame number is 1)

Created with the Standard Edition of HelpNDoc: [Full-featured Kindle eBooks generator](#)

FrameRate

TVideoGrabber.FrameRate

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Desired frame rate, expressed in frame per second.

Declaration

property FrameRate: Double **read** GetFrameRate **write** SetFrameRate;

__**property double** FrameRate=read=GetFrameRate, write=SetFrameRate

Property FrameRate As Double

Description

Used to set or retrieve the desired frame rate, expressed in frames per second.

If FrameRate = 0, the default frame rate of the video capture device is used.

This property does not apply to DV (see the [Frame rate](#) chapter).

The effective frame rate (for analog as well as DV devices) can be retrieved by using the [CurrentFrameRate](#) property.

See Also

[Frame rate](#) [DVReduceFrameRate](#) [FramerateDivider](#)

Created with the Standard Edition of HelpNDoc: [Benefits of a Help Authoring Tool](#)

FramerateDivider

TVideoGrabber.FramerateDivider

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Divides the frame rate by the value specified

Declaration

property FrameRateDivider: LongInt **read** GetFrameRateDivider **write** SetFrameRateDivider **default** DEF_FrameRateDivider;

__**property System::LongInt** FrameRateDivider==GetFrameRateDivider, write=SetFrameRateDivider, **default**=0;

Description

Divides the frame rate by the value specified.

This property requires the [frame grabber](#) to be enabled.

Example ("f" = frame delivered):

FrameRateDivider = 1: f f f f f f f f f f f f f f f

FrameRateDivider = 2: f . f . f . f . f . f . f . f . f

FrameRateDivider = 3: f . . f . . f . . f . . f . . f . .

See Also

[Frame rate](#) [TFrameGrabberRGBFormat](#) [DVReduceFrameRate](#) [FrameGrabber](#) [FrameGrabberCurrentRGBFormat](#) [FrameGrabberRGBFormat](#) [FrameRate](#) [GetFrameInfo](#) [GetFrameInfoString](#) [InFrameProgressEvent](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Transform Your CHM Help File Creation Process with HelpNDoc](#)

GetLastFrameWaitTimeoutMs

TVideoGrabber.GetLastFrameWaitTimeoutMs

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

When specified the GetLastFrames... functions wait for the next frame

Declaration

property GetLastFrameWaitTimeoutMs: LongInt **read** GetGetLastFrameWaitTimeoutMs **write** SetGetLastFrameWaitTimeoutMs;

__property int ZoomYCenter=read=GetGetLastFrameWaitTimeoutMs, write=SetGetLastFrameWaitTimeoutMs, **nodefault**

Property GetLastFrameWaitTimeoutMs As Long

Description

By default the GetLastFrame... functions return the current video frame.

When GetLastFrameWaitTimeoutMs is set to a duration expressed in milliseconds (e.g. 50), the GetLastFrame... functions wait for the next frame, eventually return it, or fail if the timeout specified expires.

- for live sources the maximum duration is not critical, it is possible to set e.g. 1000 milliseconds
- during playback, when paused, we recommend to set a maximum value of 1.5 times the average time between 2 frames, to be sure to catch the next frame but avoid a potential deadlock when the clip is paused and/or when seeking.

E.g. at 30 fps the average time between 2 frames is 33ms, so GetLastFrameWaitTimeoutMs = 50ms should work.

See Also

[TAutoFileName](#) [TFrameCaptureDest](#) [TOnFrameCaptureCompleted](#) [BurstCount](#) [BurstInterval](#) [BurstMode](#) [BurstType](#) [CaptureFrameSyncTo](#) [CaptureFrameTo](#) [FrameCaptureHeight](#) [FrameCaptureWidth](#) [FrameCaptureWithoutOverlay](#) [FrameCaptureZoomSize](#) [GetFrameInfoString](#) [GetLastFrameAsHBITMAP](#) [GetLastFrameAsTBitmap](#) [GetLastFrameBitmapBits](#) [GetLastFrameBitmapBits2](#) [JPEGPerformance](#) [JPEGProgressiveDisplay](#) [JPEGQuality](#) [Last](#) [BurstFrameCapture](#) [FileName](#) [Last](#) [CaptureFrameTo](#) [FileName](#) [OnDiskFull](#) [OnFrameBitmap](#) [SetFrameCaptureBounds](#) [StoragePath](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Easy CHM and documentation editor](#)

HoldRecording

TVideoGrabber.HoldRecording

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Retains the beginning of the recording.

Declaration

property HoldRecording: Boolean **read** GetHoldRecording **write** SetHoldRecording **default** DEF_HoldRecording;

__property **bool** HoldRecording=read=GetHoldRecording, write=SetHoldRecording, **default**=0

Property HoldRecording As Boolean

Description

Used to start the recording in preview mode and retain the beginning of AVI writing.

When **HoldRecording** is set to true before calling [StartRecording](#) , the recording graph is build and started in preview mode. Then, as soon as [ResumeRecording](#) is invoked, AVI writing begins.

When calling StartRecording directly without using **HoldRecording**, a few seconds are necessary to build the recording graph, so it is not possible to have the AVI file starting at the moment where [StartRecording](#) was called.

By enabling **HoldRecording** before calling [StartRecording](#) , the recording is started in preview mode, and the AVI writing begins exactly when [ResumeRecording](#) is invoked.

After invoking StartRecording, the [OnRecordingReadyToStart](#) event occurs to let you know that you can invoke [ResumeRecording](#) to start the recording.

From this event the recording can be cancelled by invoking [StopRecording](#).

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [IsRecordingPaused](#) [Last_Recording_FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [News and information about help authoring tools and software](#)

ImageOverlay_AlphaBlend

TVideoGrabber.ImageOverlay_AlphaBlend

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables the alpha blending of the current image overlay selected by [ImageOverlaySelector](#)

Declaration

property ImageOverlay_AlphaBlend: Boolean **read** GetImageOverlay_AlphaBlend **write** SetImageOverlay_AlphaBlend **default** DEF_ImageOverlay_AlphaBlend;

__property **bool** ImageOverlay_AlphaBlend==GetImageOverlay_AlphaBlend, write=SetImageOverlay_AlphaBlend, **default**=0;

property ImageOverlay_AlphaBlend as Boolean

Description

Enables the alpha blending of an overlayed image.
The alpha blending value must be specified with [ImageOverlay_AlphaBlendValue](#)

Look at the [Image Overlays](#) chapter for more information.

See Also

[Image overlays](#) [GetPixelsDistance](#) [GetRGBPixelAt](#) [ImageOverlay_AlphaBlendValue](#) [ImageOverlay_Height](#) [ImageOverlay_LeftLocation](#) [ImageOverlay_TopLocation](#) [ImageOverlay_Transparent](#) [ImageOverlay_TransparentColorValue](#) [ImageOverlay_UseTransparentColor](#) [ImageOverlay_VideoAlignment](#) [ImageOverlay_Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#) [SetImageOverlay_AlphaBlend](#) [SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#) [SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#) [SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#) [SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#) [SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#) [SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)

Created with the Standard Edition of HelpNDoc: Create HTML Help, DOC, PDF and print manuals from 1 single source

ImageOverlay_AlphaBlendValue

TVideoGrabber.ImageOverlay_AlphaBlendValue

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the the alpha blending value of the current image overlay selected by [ImageOverlaySelector](#)

Declaration

property ImageOverlay_AlphaBlendValue: LongInt **read** GetImageOverlay_AlphaBlendValue **write** SetImageOverlay_AlphaBlendValue **default** DEF_ImageOverlay_AlphaBlendValue;

__property int ImageOverlay_AlphaBlendValue==GetImageOverlay_AlphaBlendValue, write=SetImageOverlay_AlphaBlendValue, **default**=180;

property ImageOverlay_AlphaBlendValue as Long

Description

Specifies the alpha blending value of an overlayed image.
The alpha blending must be enabled by [ImageOverlay_AlphaBlend](#)

Look at the [Image Overlays](#) chapter for more information.

See Also

[Image overlays](#) [GetPixelsDistance](#) [GetRGBPixelAt](#) [ImageOverlay_AlphaBlend](#) [ImageOverlay_Height](#) [ImageOverlay_LeftLocation](#) [ImageOverlay_TopLocation](#) [ImageOverlay_Transparent](#) [ImageOverlay_TransparentColorValue](#) [ImageOverlay_UseTransparentColor](#) [ImageOverlay_VideoAlignment](#) [ImageOverlay_Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#) [SetImageOverlay_AlphaBlend](#) [SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#) [SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#) [SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#) [SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#) [SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#) [SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)

[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)

Created with the Standard Edition of HelpNDoc: [Create HTML Help, DOC, PDF and print manuals from 1 single source](#)

ImageOverlay_ChromaKey

TVideoGrabber.ImageOverlay_ChromaKey

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables/disables the chroma key feature of the current image overlay selected by [ImageOverlaySelector](#)

Declaration

property ImageOverlay_ChromaKey: Boolean **read** GetImageOverlay_ChromaKey **write** SetImageOverlay_ChromaKey **default** DEF_ImageOverlay_ChromaKey;

__property bool ImageOverlay_ChromaKey==GetImageOverlay_ChromaKey, write=SetImageOverlay_ChromaKey, **default**=0;

property ImageOverlay_ChromaKey as Boolean

Description

Used to enable or disable the chroma key feature of the image overlay.

See the [Chroma Key](#) chapter.

See Also

[Chroma key](#) [ImageOverlay_ChromaKeyLeewayPercent](#) [ImageOverlay_ChromaKeyRGBColor](#)

Created with the Standard Edition of HelpNDoc: [Free iPhone documentation generator](#)

ImageOverlay_ChromaKeyLeewayPercent

TVideoGrabber.ImageOverlay_ChromaKeyLeewayPercent

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the percentage of leeway of the chroma key of the current image overlay selected by [ImageOverlaySelector](#)

Declaration

property ImageOverlay_ChromaKeyLeewayPercent: LongInt **read** GetImageOverlay_ChromaKeyLeewayPercent **write** SetImageOverlay_ChromaKeyLeewayPercent **default** DEF_ImageOverlay_ChromaKeyLeewayPercent;

__property int ImageOverlay_ChromaKeyLeewayPercent==GetImageOverlay_ChromaKeyLeewayPercent, write=SetImageOverlay_ChromaKeyLeewayPercent, **default**=25;

property ImageOverlay_ChromaKeyLeewayPercent as Long

Description

Used to Specify the percentage of leeway of the chroma key.

See the [Chroma Key](#) chapter.

See Also

[Chroma key](#) [ImageOverlay_ChromaKey](#) [ImageOverlay_ChromaKeyRGBColor](#)

ImageOverlay_ChromaKeyRGBColor

TVideoGrabber.ImageOverlay_ChromaKeyRGBColor

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the RGB color used as chroma key for the current image overlay selected by [ImageOverlaySelector](#)

Declaration

property ImageOverlay_ChromaKeyRGBColor: LongInt **read** GetImageOverlay_ChromaKeyRGBColor **write** SetImageOverlay_ChromaKeyRGBColor **default** DEF_ImageOverlay_ChromaKeyRGBColor;

__property int ImageOverlay_ChromaKeyRGBColor==GetImageOverlay_ChromaKeyRGBColor, write=SetImageOverlay_ChromaKeyRGBColor, **default**=0;

property ImageOverlay_ChromaKeyRGBColor as Long

Description

Specifies the RGB color used as chroma key.

See the [Chroma Key](#) chapter.

See Also

[Chroma key](#) [ImageOverlay_ChromaKey](#) [ImageOverlay_ChromaKeyLeewayPercent](#)

ImageOverlay_Height

TVideoGrabber.ImageOverlay_Height

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the height of the current image overlay selected by [ImageOverlaySelector](#)

Declaration

property ImageOverlay_Height: LongInt **read** GetImageOverlay_Height **write** SetImageOverlay_Height **default** DEF_ImageOverlay_Height;

__property int ImageOverlay_Height==GetImageOverlay_Height, write=SetImageOverlay_Height, **default**=-1;

property ImageOverlay_Height as Long

Description

Specifies the height to which the overlaid image will be stretched.

Set ImageOverlay_Height = -1 to use the real height of the loaded image.

Look at the [Image Overlays](#) chapter for more information.

See Also

[Image overlays](#) [GetPixelsDistance](#) [GetRGBPixelAt](#) [ImageOverlay_AlphaBlend](#) [ImageOverlay_AlphaBlendValue](#) [ImageOverlay_LeftLocation](#) [ImageOverlay_TopLocation](#) [ImageOverlay_Transparent](#) [ImageOverlay_TransparentColorValue](#) [ImageOverlay_UseTransparentColor](#) [ImageOverlay_VideoAlignment](#) [ImageOverlay_Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#)

[SetImageOverlay_AlphaBlend](#)
[SetImageOverlay_AlphaBlendValue](#)
[SetImageOverlay_ChromaKey](#)
[SetImageOverlay_ChromaKeyLeewayPercent](#)
[SetImageOverlay_ChromaKeyRGBColor](#)
[SetImageOverlay_Enabled](#)
[SetImageOverlay_Height](#)
[SetImageOverlay_LeftLocation](#)
[SetImageOverlay_RotationAngle](#)
[SetImageOverlay_StretchToVideoSize](#)
[SetImageOverlay_TargetDisplay](#)
[SetImageOverlay_TopLocation](#)
[SetImageOverlay_Transparent](#)
[SetImageOverlay_TransparentColorValue](#)
[SetImageOverlay_UseTransparentColor](#)
[SetImageOverlay_Width](#)
[SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#)
[SetImageOverlayFromHBitmap2](#)
[SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#)
[SetImageOverlayFromJPEGFile](#)
[SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#)
[SetImageOverlayFromTImage](#)
[SetImageOverlayFromTImage2](#)

Created with the Standard Edition of HelpNDoc: [Streamline Your Documentation Process with HelpNDoc's Intuitive Interface](#)

ImageOverlay_LeftLocation

TVideoGrabber.ImageOverlay_LeftLocation

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the left location of the current image overlay selected by [ImageOverlaySelector](#)

Declaration

property ImageOverlay_LeftLocation: LongInt **read** GetImageOverlay_LeftLocation **write** SetImageOverlay_LeftLocation **default** DEF_ImageOverlay_LeftLocation;

__property int ImageOverlay_LeftLocation==GetImageOverlay_LeftLocation, write=SetImageOverlay_LeftLocation, **default**=10;

property ImageOverlay_LeftLocation as Long

Description

Specifies the left location of the overlaid image over the video frames.

The value must be between 0 and the current video width.

Look at the [Image Overlays](#) chapter for more information.

See Also

[Image overlays](#)
[GetPixelsDistance](#)
[GetRGBPixelAt](#)
[ImageOverlay_AlphaBlend](#)
[ImageOverlay_AlphaBlendValue](#)
[ImageOverlay_Height](#)
[ImageOverlay_TopLocation](#)
[ImageOverlay_Transparent](#)
[ImageOverlay_TransparentColorValue](#)
[ImageOverlay_UseTransparentColor](#)
[ImageOverlay_VideoAlignment](#)
[ImageOverlay_Width](#)
[ImageOverlayEnabled](#)
[ImageOverlaySelector](#)
[SetImageOverlay_AlphaBlend](#)
[SetImageOverlay_AlphaBlendValue](#)
[SetImageOverlay_ChromaKey](#)
[SetImageOverlay_ChromaKeyLeewayPercent](#)
[SetImageOverlay_ChromaKeyRGBColor](#)
[SetImageOverlay_Enabled](#)
[SetImageOverlay_Height](#)
[SetImageOverlay_LeftLocation](#)
[SetImageOverlay_RotationAngle](#)
[SetImageOverlay_StretchToVideoSize](#)
[SetImageOverlay_TargetDisplay](#)
[SetImageOverlay_TopLocation](#)
[SetImageOverlay_Transparent](#)
[SetImageOverlay_TransparentColorValue](#)
[SetImageOverlay_UseTransparentColor](#)
[SetImageOverlay_Width](#)
[SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#)
[SetImageOverlayFromHBitmap2](#)
[SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#)
[SetImageOverlayFromJPEGFile](#)
[SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#)
[SetImageOverlayFromTImage](#)
[SetImageOverlayFromTImage2](#)

Created with the Standard Edition of HelpNDoc: [Full-featured Kindle eBooks generator](#)

ImageOverlay_RotationAngle

TVideoGrabber.ImageOverlay_RotationAngle

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies a rotation angle for the current image overlay selected by [ImageOverlaySelector](#)

Declaration

property ImageOverlay_RotationAngle: Double **read** GetImageOverlay_RotationAngle **write** SetImageOverlay_RotationAngle **default** DEF_ImageOverlay_RotationAngle

__property double ImageOverlay_RotationAngle==GetImageOverlay_RotationAngle, write=SetImageOverlay_RotationAngle, **default**=0;

property _ImageOverlay_RotationAngle as Double

Description

Used to specify a rotation angle for the current image overlay.

By default the rotation angle is 0.0, it can be any value between 0.0 and 360.0

Created with the Standard Edition of HelpNDoc: [Streamline Your Documentation Process with a Help Authoring Tool](#)

ImageOverlay_StretchToVideoSize**TVideoGrabber.ImageOverlay_StretchToVideoSize**
[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Stretches the current image overlay selected by [ImageOverlaySelector](#) to the video size

Declaration

property ImageOverlay_StretchToVideoSize: Boolean **read** GetImageOverlay_StretchToVideoSize **write** SetImageOverlay_StretchToVideoSize **default** DEF_ImageOverlay_StretchToVideoSize;

__property bool ImageOverlay_StretchToVideoSize=read=GetImageOverlay_StretchToVideoSize, write=SetImageOverlay_StretchToVideoSize, **default**=0

Property ImageOverlay_StretchToVideoSize as Boolean

Description

When enabled this property stretches the image overlay to the video size, (in this case the image overlay location and size settings are ignored)

See [Graphic overlays](#)

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay_AlphaBlend](#) [SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#) [SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#) [SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#) [SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#) [SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#) [SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay_Align](#) [SetTextOverlay_BkColor](#) [SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#) [SetTextOverlay_GradientColor](#) [SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#) [SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#) [SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#)

[SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay_BkColor](#) [TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#) [TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#) [TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#) [TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Quickly and Easily Convert Your Word Document to an ePub or Kindle eBook](#)

ImageOverlay_TopLocation

TVideoGrabber.ImageOverlay_TopLocation

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the top location of the current image overlay selected by [ImageOverlaySelector](#)

Declaration

property ImageOverlay_TopLocation: LongInt **read** GetImageOverlay_TopLocation **write** SetImageOverlay_TopLocation **default** DEF_ImageOverlay_TopLocation;

__property int ImageOverlay_TopLocation==GetImageOverlay_TopLocation, write=SetImageOverlay_TopLocation, **default**=10;

property ImageOverlay_TopLocation as Long

Description

Specifies the top location of the overlaid image over the video frames.

The value must be between 0 and the current video height.

Look at the [Image Overlays](#) chapter for more information.

See Also

[Image overlays](#) [GetPixelsDistance](#) [GetRGBPixelAt](#) [ImageOverlay_AlphaBlend](#) [ImageOverlay_AlphaBlendValue](#) [ImageOverlay_Height](#) [ImageOverlay_LeftLocation](#) [ImageOverlay_Transparent](#) [ImageOverlay_TransparentColorValue](#) [ImageOverlay_UseTransparentColor](#) [ImageOverlay_VideoAlignment](#) [ImageOverlay_Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#) [SetImageOverlay_AlphaBlend](#) [SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#) [SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#) [SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#) [SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#) [SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#) [SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)

Created with the Standard Edition of HelpNDoc: [Free Kindle producer](#)

ImageOverlay_Transparent

TVideoGrabber.ImageOverlay_Transparent

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables the transparency of the current image overlay selected by [ImageOverlaySelector](#)

Declaration

property ImageOverlay_Transparent: Boolean **read** GetImageOverlay_Transparent **write** SetImageOverlay_Transparent **default** DEF_ImageOverlay_Transparent;

__**property bool** ImageOverlay_Transparent==GetImageOverlay_Transparent, write=SetImageOverlay_Transparent, **default**=0;

property ImageOverlay_Transparent as Boolean

Description

Used to enable the transparency of the image overlay.

Look at the [Image Overlays](#) chapter for more information.

See Also

[Image overlays](#) [GetPixelsDistance](#) [GetRGBPixelAt](#) [ImageOverlay_AlphaBlend](#) [ImageOverlay_AlphaBlendValue](#) [ImageOverlay_Height](#) [ImageOverlay_LeftLocation](#) [ImageOverlay_TopLocation](#) [ImageOverlay_TransparentColorValue](#) [ImageOverlay_UseTransparentColor](#) [ImageOverlay_VideoAlignment](#) [ImageOverlay_Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#) [SetImageOverlay_AlphaBlend](#) [SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#) [SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#) [SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#) [SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#) [SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#) [SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)

Created with the Standard Edition of HelpNDoc: Experience the Power and Ease of Use of HelpNDoc for CHM Help File Generation

ImageOverlay_TransparentColorValue

TVideoGrabber.ImageOverlay_TransparentColorValue

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the transparency color of the current image overlay selected by [ImageOverlaySelector](#)

Declaration

property ImageOverlay_TransparentColorValue: LongInt **read** GetImageOverlay_TransparentColorValue **write** SetImageOverlay_TransparentColorValue **default** DEF_ImageOverlay_TransparentColorValue;

__**property int** ImageOverlay_TransparentColorValue==GetImageOverlay_TransparentColorValue, write=SetImageOverlay_TransparentColorValue, **default**=0;

property ImageOverlay_TransparentColorValue as Long

Description

Used to specify the value of the color used for transparency when [ImageOverlay_Transparent](#) and [ImageOverlay_UseTransparentColor](#) are enabled.

Look at the [Image Overlays](#) chapter for more information.

See Also

[Image overlays](#) [GetPixelsDistance](#) [GetRGBPixelAt](#) [ImageOverlay_AlphaBlend](#) [ImageOverlay_AlphaBlendValue](#) [ImageOverlay_Height](#) [ImageOverlay_LeftLocation](#) [ImageOverlay_TopLocation](#) [ImageOverlay_Transparent](#) [ImageOverlay_UseTransparentColor](#) [ImageOverlay_VideoAlignment](#) [ImageOverlay_Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#) [SetImageOverlay_AlphaBlend](#) [SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#) [SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#)

[SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#)
[SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#)
[SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#)
[SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)

Created with the Standard Edition of HelpNDoc: [Transform Your Word Doc into a Professional-Quality eBook with HelpNDoc](#)

ImageOverlay_UseTransparentColor

TVideoGrabber.ImageOverlay_UseTransparentColor

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables the use of a transparency color for the current image overlay selected by [ImageOverlaySelector](#)

Declaration

property ImageOverlay_UseTransparentColor: Boolean **read** GetImageOverlay_UseTransparentColor **write** SetImageOverlay_UseTransparentColor **default** DEF_ImageOverlay_UseTransparentColor;

__property bool ImageOverlay_UseTransparentColor==GetImageOverlay_UseTransparentColor,
write=SetImageOverlay_UseTransparentColor, **default**=0;

property ImageOverlay_UseTransparentColor as Boolean

Description

Used to specify if a given color must be used for transparency when [ImageOverlay_Transparent](#) is enabled.

If enabled, the color must be specified by [ImageOverlay_TransparentColorValue](#).

If disabled, the default color transparency (background color) of the image will be used.

Look at the [Image Overlays](#) chapter for more information.

See Also

[Image overlays](#) [GetPixelsDistance](#) [GetRGBPixelAt](#) [ImageOverlay_AlphaBlend](#)
[ImageOverlay_AlphaBlendValue](#) [ImageOverlay_Height](#) [ImageOverlay_LeftLocation](#)
[ImageOverlay_TopLocation](#) [ImageOverlay_Transparent](#) [ImageOverlay_TransparentColorValue](#)
[ImageOverlay_VideoAlignment](#) [ImageOverlay_Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#)
[SetImageOverlay_AlphaBlend](#) [SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#)
[SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#)
[SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#)
[SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#)
[SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#)
[SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)

Created with the Standard Edition of HelpNDoc: [Elevate Your Documentation with HelpNDoc's Project Analyzer Features](#)

ImageOverlay_VideoAlignment

TVideoGrabber.ImageOverlay_VideoAlignment

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Relative alignment of the image overlay within the video frame

Declaration

property ImageOverlay_VideoAlignment: TVideoAlignment **read** GetImageOverlayVideoAlignment **write** SetImageOverlayVideoAlignment;

__property TVideoAlignment ImageOverlay_VideoAlignment==GetImageOverlayVideoAlignment, write=SetImageOverlayVideoAlignment, **nodefault**};

Description

[TVideoAlignment](#) relative alignment of the image overlay within the video frame.
Default: oa_LeftTop

See Also

[Image overlays](#) [GetPixelsDistance](#) [GetRGBPixelAt](#) [ImageOverlay_AlphaBlend](#) [ImageOverlay_AlphaBlendValue](#) [ImageOverlay_Height](#) [ImageOverlay_LeftLocation](#) [ImageOverlay_TopLocation](#) [ImageOverlay_Transparent](#) [ImageOverlay_TransparentColorValue](#) [ImageOverlay_UseTransparentColor](#) [ImageOverlay_Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#) [SetImageOverlay_AlphaBlend](#) [SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#) [SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#) [SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#) [SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#) [SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#) [SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)

Created with the Standard Edition of HelpNDoc: Don't be left in the past: convert your WinHelp HLP help files to CHM with HelpNDoc

ImageOverlay_Width

TVideoGrabber.ImageOverlay_Width

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the width of the current image overlay selected by [ImageOverlaySelector](#)

Declaration

property ImageOverlay_Width: LongInt **read** GetImageOverlay_Width **write** SetImageOverlay_Width
default DEF_ImageOverlay_Width;

__property int ImageOverlay_Width==GetImageOverlay_Width, write=SetImageOverlay_Width, **default**=-1;

property ImageOverlay_Width as Long

Description

Specifies the width to which the overlaid image will be stretched.

Set ImageOverlay_Width = -1 to use the real width of the loaded image.

Look at the [Image Overlays](#) chapter for more information.

See Also

[Image overlays](#) [GetPixelsDistance](#) [GetRGBPixelAt](#) [ImageOverlay_AlphaBlend](#) [ImageOverlay_AlphaBlendValue](#) [ImageOverlay_Height](#) [ImageOverlay_LeftLocation](#) [ImageOverlay_TopLocation](#) [ImageOverlay_Transparent](#) [ImageOverlay_TransparentColorValue](#) [ImageOverlay_UseTransparentColor](#) [ImageOverlay_VideoAlignment](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#) [SetImageOverlay_AlphaBlend](#) [SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#) [SetImageOverlay_ChromaKeyLeewayPercent](#)

[SetImageOverlay_ChromaKeyRGBColor](#) [SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#) [SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#) [SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#) [SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)

Created with the Standard Edition of HelpNDoc: [Streamline Your Documentation Process with a Help Authoring Tool](#)

ImageOverlayEnabled

TVideoGrabber.ImageOverlayEnabled

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables / disables the image overlay currently loaded selected by [ImageOverlaySelector](#)

Declaration

property ImageOverlayEnabled: Boolean **read** GetImageOverlayEnabled **write** SetImageOverlayEnabled
default DEF_ImageOverlayEnabled;

__property **bool** ImageOverlayEnabled=read=GetImageOverlayEnabled, write=SetImageOverlayEnabled, **default**=0

Property ImageOverlayEnabled As Boolean

Description

Used to enable disable the image overlay currently loaded.

Note: see the [Image overlays](#) chapter for global information about this feature.

See Also

[Image overlays](#) [GetPixelsDistance](#) [GetRGBPixelAt](#) [ImageOverlay_AlphaBlend](#) [ImageOverlay_AlphaBlendValue](#) [ImageOverlay_Height](#) [ImageOverlay_LeftLocation](#) [ImageOverlay_TopLocation](#) [ImageOverlay_Transparent](#) [ImageOverlay_TransparentColorValue](#) [ImageOverlay_UseTransparentColor](#) [ImageOverlay_VideoAlignment](#) [ImageOverlay_Width](#) [ImageOverlaySelector](#) [SetImageOverlay_AlphaBlend](#) [SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#) [SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#) [SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#) [SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#) [SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#) [SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)

Created with the Standard Edition of HelpNDoc: [What is a Help Authoring tool?](#)

ImageOverlaySelector

TVideoGrabber.ImageOverlaySelector

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Select a group of image overlay settings

Declaration

property ImageOverlaySelector: LongInt **read** GetImageOverlaySelector **write** SetImageOverlaySelector;

__property int ImageOverlaySelector==GetImageOverlaySelector, write=SetImageOverlaySelector, **nodefault**};

property ImageOverlaySelector as Long

Description

Used to select a group of image overlay settings.

Up to 1000 image overlay groups of settings may be used, allowing to perform different image overlays (with different location, image format, etc...) at the same time.

The default image overlay settings correspond to the group 0 ([ImageOverlaySelector](#) = 0).

Note that TVideoGrabber does not allocate the resources for 1000 image overlays, by default it allocates the settings corresponding to ImageOverlaySelector = 0 (the resource allocation is made when ImageOverlaySelector receives a new value never used before, see the [Image overlays](#) chapter for more explanations).

Set [ImageOverlaySelector](#) to the index of the image overlay properties to read or write before reading or writing their values.

It is applied to all the ImageOverlay... properties and functions;

For example, the following code overlays 2 images at 2 different location, the 1st is overlayed in its current size (width = -1 and height = -1), the second is stretched (because the width and height are specified)

```
procedure TfrmMainForm.Button1Click(Sender: TObject);
begin
    VideoGrabber.ImageOverlaySelector := 0;
    VideoGrabber.SetImageOverlayFromAnyImageFile ('myimage1.gif');
    VideoGrabber.SetImageOverlayAttributes (10, 10, -1, -1, false, false, 0, false, 0);
    VideoGrabber.ImageOverlayEnabled := True;

    VideoGrabber.ImageOverlaySelector := 1;
    VideoGrabber.SetImageOverlayFromAnyImageFile ('myimage2.png');
    VideoGrabber.SetImageOverlayAttributes (100, 100, 30, 20, false, false, 0, false, 0);
    VideoGrabber.ImageOverlayEnabled := True;
end;
```

in C++:

```
void __fastcall TfrmMainForm::Button1Click(TObject *Sender)

void __fastcall TForm1::Button1Click(TObject *Sender)

    VideoGrabber->ImageOverlaySelector = 0;
    VideoGrabber->SetImageOverlayFromAnyImageFile ("myimage1.gif");
    VideoGrabber->SetImageOverlayAttributes (10, 10, -1, -1, false, false, 0, false, 0);
    VideoGrabber->ImageOverlayEnabled = true;

    VideoGrabber->ImageOverlaySelector = 1;
    VideoGrabber->SetImageOverlayFromAnyImageFile ("myimage2.png");
    VideoGrabber->SetImageOverlayAttributes (100, 100, 30, 20, false, false, 0, false, 0);
    VideoGrabber->ImageOverlayEnabled = true;
```

in VB:

```
Private Sub Command1_Click()
    VideoGrabber.ImageOverlaySelector = 0
    VideoGrabber.SetImageOverlayFromAnyImageFile ("myimage1.gif")
    VideoGrabber.SetImageOverlayAttributes (10, 10, -1, -1, false, false, 0, false, 0)
    VideoGrabber.ImageOverlayEnabled = true

    VideoGrabber.ImageOverlaySelector = 1
    VideoGrabber.SetImageOverlayFromAnyImageFile ("myimage2.png")
    VideoGrabber.SetImageOverlayAttributes (100, 100, 30, 20, false, false, 0, false, 0)
    VideoGrabber.ImageOverlayEnabled = true
End Sub
```

See Also

[Image overlays](#) [GetPixelsDistance](#) [GetRGBPixelAt](#) [ImageOverlay](#) [AlphaBlend](#) [ImageOverlay](#) [AlphaBlendValue](#) [ImageOverlay](#) [Height](#) [ImageOverlay](#) [LeftLocation](#) [ImageOverlay](#) [TopLocation](#) [ImageOverlay](#) [Transparent](#) [ImageOverlay](#) [TransparentColorValue](#) [ImageOverlay](#) [UseTransparentColor](#) [ImageOverlay](#) [VideoAlignment](#) [ImageOverlay](#) [Width](#) [ImageOverlayEnabled](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)

Created with the Standard Edition of HelpNDoc: [What is a Help Authoring tool?](#)

ImageRatio

TVideoGrabber.ImageRatio

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Ratio between the width and height of the native video size.

Declaration

property ImageRatio: Double **read** GetImageRatio;

__property **double** ImageRatio=read=GetImageRatio

Property ImageRatio As Double

Description

Used to retrieve the ratio between the width and the height of the native video size.
E.g. for a 4 / 3 image, this property will return 0.75

See Also

[TOnFrameProgress](#) [TOnResizeVideo](#) [AutoRefreshPreview](#) [BackgroundColor](#) [Display](#) [AutoSize](#) [Display](#) [FullScreen](#) [Display](#) [SetLocation](#) [Display](#) [VideoPortEnabled](#) [InFrameProgressEvent](#) [IsVideoInterlaced](#) [IsVideoPortAvailable](#) [OnFrameProgress](#) [OnKeyPress](#) [OnPreviewStarted](#) [OnResizeVideo](#) [PausePreview](#) [PreviewZoomSize](#) [ResumePreview](#) [StartPreview](#) [StartSynchronized](#) [StopPreview](#) [Synchronized](#) [UseNearestVideoSize](#) [VideoHeight](#) [VideoRenderer](#) [VideoRendererPriority](#) [VideoWidth](#)

[Visible](#)Created with the Standard Edition of HelpNDoc: [Free CHM Help documentation generator](#)

InFrameProgressEvent

TVideoGrabber.InFrameProgressEvent

[Prev](#)[Next](#)[TVideoGrabber](#) [Properties](#)

Returns true if the [OnFrameProgress](#) event is currently occurring.

Declaration

property InFrameProgressEvent: Boolean **read** GetInFrameProgressEvent;

__property **bool** InFrameProgressEvent=read=GetInFrameProgressEvent, **ndefault**

Property InFrameProgressEvent As Boolean

Description

Used to know if the [OnFrameProgress](#) event is currently occurring.

This property is useful e.g. when using a trackbar to track the player position. In this case the TrackBar's OnChange event must not update the TVideoGrabber's player position if this event occurs while the trackbar is updated from the OnFrameProgress event, otherwise a loop back update occurs and the trackbar motion is "jerky".

Sample code:

```
procedure TForm1.TrackBar1Change(Sender: TObject);
begin
    if not VideoGrabber1.InFrameProgressEvent then begin
        VideoGrabber1.PlayerFramePosition := TrackBar1.Position;
    end;
end;
```

See Also

[TFrameGrabberRGBFormat](#) [FrameGrabber](#) [FrameGrabberCurrentRGBFormat](#) [FrameGrabberRGBFormat](#) [FramerateDivider](#) [GetFrameInfo](#) [GetFrameInfoString](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Save time and frustration with HelpNDoc's WinHelp HLP to CHM conversion feature](#)

IPCameraURL

TVideoGrabber.IPCameraURL

[Prev](#)[Next](#)[TVideoGrabber](#) [Properties](#)

Sets the URL of the IP Camera

Declaration

property IPCameraURL: **String** **read** GetIPCameraURL **write** SetIPCameraURL

__property wchar_t *IPCameraURL==GetIPCameraURL, write=SetIPCameraURL

Property IPCameraURL as String

Description

Used to specify the URL of the IP Camera when [VideoSource](#) = vs_IPCamera

E.g. for an Axis camera:

(*x.x.x.x is the IP of the camera*)

VideoGrabber.VideoSource = vs_IPCamera

VideoGrabber.IPCameraURL = "http://x.x.x.x/axis-cgi/mjpg/video.cgi?resolution=640x480

VideoGrabber.StartPreview

See Also

[IR Cut Filter of Axis cameras](#) [TDiscoveryCallbackStatus](#) [TOnONVIFDiscoveryCompletedNotification](#) [Video stream of IP camera with audio of PC microphone or other audio capture device](#) [ExtraDLLPath](#) [GetLastErrorMessages](#) [IsURLResponding](#) [IsURLVideoStreamAvailable](#) [OnONVIFDiscoveryCompletedNotification](#) [ONVIF](#) [GetStr](#) [ONVIF](#) [SetStr](#) [ONVIFCancelDiscovery](#) [ONVIFDiscoverCameras](#) [IPRange](#) [ONVIFDiscoverCameras](#) [Multicast](#) [ONVIFEnumCamerasDiscovered](#) [OpenURLAsync](#) [OpenURLAsyncStatus](#) [SendIPCCommand](#) [SetDatasteadFilterDllName](#) [SetIPCSetting](#)

Created with the Standard Edition of HelpNDoc: [Revolutionize Your Documentation Review with HelpNDoc's Project Analyzer](#)

IsAnalogVideoDecoderAvailable

TVideoGrabber.IsAnalogVideoDecoderAvailable

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Returns true if an [analog video decoder](#) is available.

Declaration

property IsAnalogVideoDecoderAvailable: Boolean **read** GetIsAnalogVideoDecoderAvailable;

__property **bool** IsAnalogVideoDecoderAvailable=read=GetIsAnalogVideoDecoderAvailable, **nodefault**

Property IsAnalogVideoDecoderAvailable As Boolean

Description

Returns true if the current [video capture device](#) exposes an [analog video decoder](#) .

See Also

[Analog Video standards](#) [AnalogVideoStandard](#) [AnalogVideoStandardIndex](#) [AnalogVideoStandards](#) [AnalogVideoStandardsCount](#)

Created with the Standard Edition of HelpNDoc: [Qt Help documentation made easy](#)

IsAudioCrossbarAvailable

TVideoGrabber.IsAudioCrossbarAvailable

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Returns true if the audio crossbar is available.

Declaration

property IsAudioCrossbarAvailable: Boolean **read** GetIsAudioCrossbarAvailable;

__property **bool** IsAudioCrossbarAvailable=read=GetIsAudioCrossbarAvailable, **nodefault**

Property IsAudioCrossbarAvailable As Boolean

Description

Returns true if the current audio capture device exposes an audio crossbar.

See Also

[TAudioFormat](#) [AssociateAudioAndVideoDevices](#) [AudioBalance](#) [AudioDevice](#) [AudioDeviceIndex](#) [AudioDeviceName](#) [AudioDeviceRendering](#) [AudioDevices](#) [AudioDevicesCount](#) [AudioFormat](#) [AudioFormats](#) [AudioInput](#) [AudioInputIndex](#) [AudioInputLevel](#) [AudioInputMono](#) [AudioInputs](#) [AudioInputsCount](#) [AudioSource](#) [IsAudioDeviceASoundCard](#) [IsAudioDeviceConnected](#) [IsAudioInputBalanceAvailable](#) [OnAudioBufferNegotiation](#) [OnAudioDeviceSelected](#) [SpeakerBalance](#) [SpeakerControl](#) [SpeakerVolume](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Create Professional Documentation with HelpNDoc's Clean UI](#)

IsAudioInputBalanceAvailable

TVideoGrabber.IsAudioInputBalanceAvailable

[Prev](#)[Next](#)

[TVideoGrabber](#) [Properties](#)

Availability of the audio input balance.

Declaration

property IsAudioInputBalanceAvailable: TTriState **read** GetIsAudioInputBalanceAvailable;

__property TTriState IsAudioInputBalanceAvailable=read=GetIsAudioInputBalanceAvailable, **nodefault**

Property IsAudioInputBalanceAvailable As TTriState

Description

This property lets you know if audio balance is available for the current [audio input](#) selected. This is a tri-state property, that returns ts_True if available, or ts_False if not available. The [preview](#) or [recording](#) must be running, otherwise this property returns ts_Undefined.

See Also

[TAudioFormat](#) [AssociateAudioAndVideoDevices](#) [AudioBalance](#) [AudioDevice](#) [AudioDeviceIndex](#) [AudioDeviceName](#) [AudioDeviceRendering](#) [AudioDevices](#) [AudioDevicesCount](#) [AudioFormat](#) [AudioFormats](#) [AudioInput](#) [AudioInputIndex](#) [AudioInputLevel](#) [AudioInputMono](#) [AudioInputs](#) [AudioInputsCount](#) [AudioSource](#) [IsAudioCrossbarAvailable](#) [IsAudioDeviceASoundCard](#) [IsAudioDeviceConnected](#) [OnAudioBufferNegotiation](#) [OnAudioDeviceSelected](#) [SpeakerBalance](#) [SpeakerControl](#) [SpeakerVolume](#)

Created with the Standard Edition of HelpNDoc: [Easily create Web Help sites](#)

IsCameraControlAvailable

TVideoGrabber.IsCameraControlAvailable

[Prev](#)[Next](#)

[TVideoGrabber](#) [Properties](#)

Returns true if the camera controls are available.

Declaration

property IsCameraControlAvailable: Boolean **read** GetIsCameraControlAvailable;

__property **bool** IsCameraControlAvailable=read=GetIsCameraControlAvailable, **nodefault**

Property IsCameraControlAvailable As Boolean

Description

Returns true if the current video capture device exposes camera control settings.

See Also

[TCameraControl](#) [CameraControlAuto](#) [CameraControlDefault](#) [CameraControlMax](#) [CameraControlMin](#) [CameraControlSettings](#) [CameraControlStep](#) [GetCameraExposure](#) [GetCameraExposureAsString](#) [IsCameraControlSettingAvailable](#) [SendCameraCommand](#) [SetCameraControl](#) [SetCameraExposure](#)

Created with the Standard Edition of HelpNDoc: [Revolutionize Your Documentation Output with HelpNDoc's Stunning User Interface](#)

IsDigitalVideoIn

TVideoGrabber.IsDigitalVideoIn

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Returns true if the current video capture device is a [DV](#) source.

Declaration

property IsDigitalVideoIn: Boolean **read** GetIsDigitalVideoIn;

__property **bool** IsDigitalVideoIn=read=GetIsDigitalVideoIn, **nodefault**

Property IsDigitalVideoIn As Boolean

Description

Returns true if the current [video capture device](#) is a [DV](#) source.

See Also

[TDVSize](#) [TDVVideoFormat](#) [TDVVideoStandard](#) [TOnDVCommandCompleted](#) [DVDateTimeEnabled](#) [DVDDiscontinuityMinimumInterval](#) [DVRecordingInNativeFormatSeparatesStreams](#) [DVReduceFrameRate](#) [DVRgb219](#) [DVTimeCodeEnabled](#) [IsDVCommandAvailable](#) [IsTimeCodeReaderAvailable](#) [OnDVCommandCompleted](#) [SendDVCommand](#)

Created with the Standard Edition of HelpNDoc: [Transform Your Documentation Process with HelpNDoc's Project Analyzer](#)

IsDVCommandAvailable

TVideoGrabber.IsDVCommandAvailable

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Returns true if [DV commands](#) are available.

Declaration

property IsDVCommandAvailable: Boolean **read** GetIsDVCommandAvailable;

__property **bool** IsDVCommandAvailable=read=GetIsDVCommandAvailable, **nodefault**

Property IsDVCommandAvailable As Boolean

Description

Returns true if [DV commands](#) are available on the current video capture device.

See Also

[TDVSize](#) [TDVVideoFormat](#) [TDVVideoStandard](#) [TOnDVCommandCompleted](#) [DVDateTimeEnabled](#) [DVDDiscontinuityMinimumInterval](#) [DVRecordingInNativeFormatSeparatesStreams](#) [DVReduceFrameRate](#) [DVRgb219](#) [DVTimeCodeEnabled](#) [IsDigitalVideoIn](#) [IsTimeCodeReaderAvailable](#) [OnDVCommandCompleted](#) [SendDVCommand](#)

Created with the Standard Edition of HelpNDoc: [Save time and frustration with HelpNDoc's WinHelp HLP](#)

[to CHM conversion feature](#)

IsHorizontalSyncLocked

TVideoGrabber.IsHorizontalSyncLocked

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Returns true upon locked horizontal sync.

Declaration

property IsHorizontalSyncLocked: Boolean **read** GetIsHorizontalSyncLocked;

__property **bool** IsHorizontalSyncLocked=read=GetIsHorizontalSyncLocked, **nodefault**

Property IsHorizontalSyncLocked As Boolean

Description

Returns true if the horizontal synchronization of the analog video decoder is locked.

See Also

[TDVCommand](#) [IsDVCommandAvailable](#) [IsTimeCodeReaderAvailable](#) [SendDVCommand](#) [VCRHorizontalLocking](#)

Created with the Standard Edition of HelpNDoc: [Easily convert your WinHelp HLP help files to CHM with HelpNDoc's step-by-step guide](#)

IsMPEGStream

TVideoGrabber.IsMPEGStream

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Returns true if the video capture devices outputs an MPEG stream.

Declaration

property IsMpegStream: Boolean **read** GetIsMpegStream;

__property **bool** IsMpegStream=read=GetIsMpegStream, **nodefault**

Property IsMpegStream As Boolean

Description

Returns true if the video capture devices outputs an MPEG stream.

Created with the Standard Edition of HelpNDoc: [Easily create Help documents](#)

IsPlayerAudioStreamAvailable

TVideoGrabber.IsPlayerAudioStreamAvailable

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Availability of the audio stream in a clip.

Declaration

property IsPlayerAudioStreamAvailable: TTriState **read** GetIsPlayerAudioStreamAvailable;

__property TTriState IsPlayerAudioStreamAvailable=read=GetIsPlayerAudioStreamAvailable, **nodefault**

Property IsPlayerAudioStreamAvailable As TxTriState

Description

This property lets you know if a clip includes an audio stream.

This is a tri-state property, that returns **ts_True** if an audio stream is available, or **ts_False** if there is no audio in the clip.

The clip must have been opened with [OpenPlayer](#), otherwise this property returns **ts_Undefined**.

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchronone](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: Streamline your documentation process with HelpNDoc's WinHelp HLP to CHM conversion feature

IsPlayerVideoStreamAvailable

TVideoGrabber.IsPlayerVideoStreamAvailable

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Availability of the video stream in a clip.

Declaration

property IsPlayerVideoStreamAvailable: TTriState **read** GetIsPlayerVideoStreamAvailable;

__property TTriState IsPlayerVideoStreamAvailable=read=GetIsPlayerVideoStreamAvailable, **nodefault**

Property IsPlayerVideoStreamAvailable As TxTriState

Description

This property lets you know if a clip includes a video stream.

This is a tri-state property, that returns **ts_True** if a video stream is available, or **ts_False** if there no video stream.

The clip must have been opened with [OpenPlayer](#), otherwise this property returns **ts_Undefined**.

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchronone](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

IsRecordingPaused

TVideoGrabber.IsRecordingPaused

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Returns true if the current recording is in a paused state.

Declaration

property IsRecordingPaused: Boolean **read** GetIsRecordingPaused;

__property **bool** IsRecordingPaused=read=GetIsRecordingPaused, **nodefault**

Property IsRecordingPaused As Boolean

Description

Used to retrieve whether the current recording has been paused by [PauseRecording](#) .

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [Last](#) [Recording](#) [FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

IsTimeCodeReaderAvailable

TVideoGrabber.IsTimeCodeReaderAvailable

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Returns true if time code information is available.

Declaration

property IsTimeCodeReaderAvailable: Boolean **read** GetTimeCodeReaderAvailable;

__property **bool** IsTimeCodeReaderAvailable=read=GetTimeCodeReaderAvailable, **nodefault**

Property IsTimeCodeReaderAvailable As Boolean

Description

Returns true if time code information is available for the current [video capture device](#) .

See Also

[TDVSize](#) [TDVVideoFormat](#) [TDVVideoStandard](#) [TOnDVCommandCompleted](#) [DVDateTimeEnabled](#) [DVDiscontinuityMinimumInterval](#) [DVRecordingInNativeFormatSeparatesStreams](#) [DVReduceFrameRate](#) [DVRgb219](#) [DVTimeCodeEnabled](#) [IsDigitalVideoIn](#) [IsDVCommandAvailable](#) [OnDVCommandCompleted](#)

[SendDVCommand](#)

Created with the Standard Edition of HelpNDoc: [Make Documentation a Breeze with HelpNDoc's Clean and Efficient User Interface](#)

IsTVAudioAvailable

TVideoGrabber.IsTVAudioAvailable

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Returns true if the TV audio is available.

Declaration

property IsTVAudioAvailable: Boolean **read** GetIsTVAudioAvailable;

___property **bool** IsTVAudioAvailable=read=GetIsTVAudioAvailable, **nodefault**

Property IsTVAudioAvailable As Boolean

Description

Returns true if the TV audio filter is available for the current video capture device.

See Also

[TTVChannelInfo](#) [TOnTVChannelScanStarted](#) [TOnTVChannelSelected](#) [TTunerInputType](#) [GetTVChannelInfo](#) [IsHorizontalSyncLocked](#) [IsTVAutoTuneRunning](#) [IsTVTunerAvailable](#) [OnTVChannelScanCompleted](#) [OnTVChannelScanStarted](#) [OnTVChannelSelected](#) [TunerFrequency](#) [TunerMode](#) [TVChannel](#) [TVClearFrequencyOverrides](#) [TVCountryCode](#) [TVGetMinMaxChannels](#) [TVSetChannelFrequencyOverride](#) [TVStartAutoScan](#) [TVStopAutoScan](#) [TVTunerInputType](#) [TVUseFrequencyOverrides](#) [VCRHorizontalLocking](#)

Created with the Standard Edition of HelpNDoc: [Free help authoring environment](#)

IsTVAutoTuneRunning

TVideoGrabber.IsTVAutoTuneRunning

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Returns true when the [TV tuner](#) automatic tuning is currently running.

Declaration

property IsTVAutoTuneRunning: Boolean **read** GetIsTVAutoTuneRunning;

___property **bool** IsTVAutoTuneRunning=read=GetIsTVAutoTuneRunning, **nodefault**

Property IsTVAutoTuneRunning As Boolean

Description

Used to retrieve whether the [TV tuner](#) automatic tuning is currently running.

See Also

[TTVChannelInfo](#) [TOnTVChannelScanStarted](#) [TOnTVChannelSelected](#) [TTunerInputType](#) [GetTVChannelInfo](#) [IsHorizontalSyncLocked](#) [IsTVAudioAvailable](#) [IsTVTunerAvailable](#) [OnTVChannelScanCompleted](#) [OnTVChannelScanStarted](#) [OnTVChannelSelected](#) [TunerFrequency](#) [TunerMode](#) [TVChannel](#) [TVClearFrequencyOverrides](#) [TVCountryCode](#) [TVGetMinMaxChannels](#) [TVSetChannelFrequencyOverride](#) [TVStartAutoScan](#) [TVStopAutoScan](#) [TVTunerInputType](#) [TVUseFrequencyOverrides](#) [VCRHorizontalLocking](#)

Created with the Standard Edition of HelpNDoc: [5 Reasons Why a Help Authoring Tool is Better than Microsoft Word for Documentation](#)

IsTVTunerAvailable

TVideoGrabber.IsTVTunerAvailable

[Prev](#)[Next](#)[TVideoGrabber](#) [Properties](#)

Returns true if the [TV tuner](#) is available.

Declaration

property IsTVTunerAvailable: Boolean **read** GetIsTVTunerAvailable;

__property **bool** IsTVTunerAvailable=read=GetIsTVTunerAvailable, **nodefault**

Property IsTVTunerAvailable As Boolean

Description

Returns true if the [TV tuner](#) is available for the current video capture device.

See Also

[TTVChannelInfo](#) [TOnTVChannelScanStarted](#) [TOnTVChannelSelected](#) [TTunerInputType](#) [GetTVChannelInfo](#) [IsHorizontalSyncLocked](#) [IsTVAudioAvailable](#) [IsTVAutoTuneRunning](#) [OnTVChannelScanCompleted](#) [OnTVChannelScanStarted](#) [OnTVChannelSelected](#) [TunerFrequency](#) [TunerMode](#) [TVChannel](#) [TVClearFrequencyOverrides](#) [TVCountryCode](#) [TVGetMinMaxChannels](#) [TVSetChannelFrequencyOverride](#) [TVStartAutoScan](#) [TVStopAutoScan](#) [TVTunerInputType](#) [TVUseFrequencyOverrides](#) [VCRHorizontalLocking](#)

Created with the Standard Edition of HelpNDoc: [Eliminate the Struggles of Documentation with a Help Authoring Tool](#)

IsVideoControlAvailable

TVideoGrabber.IsVideoControlAvailable

[Prev](#)[Next](#)[TVideoGrabber](#) [Properties](#)

Returns true if [video control modes](#) are available.

Declaration

property IsVideoControlAvailable: Boolean **read** GetIsVideoControlAvailable;

__property **bool** IsVideoControlAvailable=read=GetIsVideoControlAvailable, **nodefault**

Property IsVideoControlAvailable As Boolean

Description

Returns true if [video control modes](#) can be set on the current video capture device.

See Also

[TVideoControl](#) [GetVideoControlMode](#) [IsVideoControlModeAvailable](#) [SetVideoControlMode](#) [SetVideoControlMode2](#) [VideoControlSettings](#)

Created with the Standard Edition of HelpNDoc: [Single source CHM, PDF, DOC and HTML Help creation](#)

IsVideoCrossbarAvailable

TVideoGrabber.IsVideoCrossbarAvailable

[Prev](#)[Next](#)[TVideoGrabber](#) [Properties](#)

Returns true if a video crossbar is available.

Declaration

property IsVideoCrossbarAvailable: Boolean **read** GetIsVideoCrossbarAvailable;

___property **bool** IsVideoCrossbarAvailable=read=GetIsVideoCrossbarAvailable, **nodefault**

Property IsVideoCrossbarAvailable As Boolean

Description

Returns true if a video crossbar is available for the current video capture device.

See Also

[WDM drivers](#) [AutoConnectRelatedPins](#) [CurrentFrameRate](#) [DeliveredFrames](#) [DroppedFrames](#) [FixFlickerOrBlackCapture](#) [IsDVDevice](#) [IsVideoControlAvailable](#) [IsVideoDeviceConnected](#) [IsWDMVideoDriver](#) [OnNoVideoDevices](#) [OnVideoDeviceSelected](#) [RefreshDevicesAndCompressorsLists](#) [ResetVideoDeviceSettings](#) [ShowDialog](#) [VideoDevice](#) [VideoDeviceIndex](#) [VideoDeviceIndexFromId](#) [VideoDeviceName](#) [VideoDevices](#) [VideoDevicesCount](#) [VideoDevicesId](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Free help authoring environment](#)

IsVideoInterlaced

TVideoGrabber.IsVideoInterlaced

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Returns true if the current video capture device source is interlaced.

Declaration

property IsVideoInterlaced: Boolean **read** GetIsVideoInterlaced;

___property **bool** IsVideoInterlaced=read=GetIsVideoInterlaced, **nodefault**

Property IsVideoInterlaced As Boolean

Description

Returns true if the current [video capture device](#) video source is interlaced.

See Also

[TOnFrameProgress](#) [TOnResizeVideo](#) [AutoRefreshPreview](#) [BackgroundColor](#) [Display](#) [AutoSize](#) [Display_FullScreen](#) [Display_SetLocation](#) [Display_VideoPortEnabled](#) [ImageRatio](#) [InFrameProgressEvent](#) [IsVideoPortAvailable](#) [OnFrameProgress](#) [OnKeyPress](#) [OnPreviewStarted](#) [OnResizeVideo](#) [PausePreview](#) [PreviewZoomSize](#) [ResumePreview](#) [StartPreview](#) [StartSynchronized](#) [StopPreview](#) [Synchronized](#) [UseNearestVideoSize](#) [VideoHeight](#) [VideoRenderer](#) [VideoRendererPriority](#) [VideoWidth](#) [Visible](#)

Created with the Standard Edition of HelpNDoc: [Experience the Power and Ease of Use of HelpNDoc for CHM Help File Generation](#)

IsVideoPortAvailable

TVideoGrabber.IsVideoPortAvailable

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Returns true if the video capture device has a video port.

Declaration

property IsVideoPortAvailable: Boolean **read** GetIsVideoPortAvailable;

___property **bool** IsVideoPortAvailable=read=GetIsVideoPortAvailable, **nodefault**

Property IsVideoPortAvailable As Boolean

Description

Used to know if the video capture device has a video port.
See [Display_VideoPortEnabled](#) .

See Also

[TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display_Active](#) [Display_AlphaBlendEnabled](#) [Display_AlphaBlendValue](#) [Display_AutoSize](#) [Display_Embedded](#) [Display_FullScreen](#) [Display_Height](#) [Display_Left](#) [Display_Monitor](#) [Display_MouseMovesWindow](#) [Display_PanScanRatio](#) [Display_SetLocation](#) [Display_StayOnTop](#) [Display_Top](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [Display_VideoHeight](#) [Display_VideoPortEnabled](#) [Display_VideoWidth](#) [Display_VideoWindowHandle](#) [Display_Width](#) [DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#) [DualDisplay_AlphaBlendValue](#) [DualDisplay_AutoSize](#) [DualDisplay_Embedded](#) [DualDisplay_FullScreen](#) [DualDisplay_Height](#) [DualDisplay_Left](#) [DualDisplay_Monitor](#) [DualDisplay_MouseMovesWindow](#) [DualDisplay_PanScanRatio](#) [DualDisplay_StayOnTop](#) [DualDisplay_Top](#) [DualDisplay_TransparentColorEnabled](#) [DualDisplay_TransparentColorValue](#) [DualDisplay_VideoHeight](#) [DualDisplay_VideoPortEnabled](#) [DualDisplay_VideoWidth](#) [DualDisplay_VideoWindowHandle](#) [DualDisplay_Visible](#) [DualDisplay_Width](#) [Monitor_Primary](#) [Index](#) [MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#) [SetParentWindow](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Easily create PDF Help documents](#)

IsVideoQualityAvailable

TVideoGrabber.IsVideoQualityAvailable

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Returns true if [video quality settings](#) are available.

Declaration

property IsVideoQualityAvailable: Boolean **read** GetIsVideoQualityAvailable;

___property **bool** IsVideoQualityAvailable=read=GetIsVideoQualityAvailable, **nodefault**

Property IsVideoQualityAvailable As Boolean

Description

Returns true if [video quality settings](#) are available on the current video capture device.

See Also

[IsVideoQualitySettingAvailable](#) [SetVideoQuality](#) [VideoQualityAuto](#) [VideoQualityDefault](#) [VideoQualityMax](#) [VideoQualityMin](#) [VideoQualitySettings](#) [VideoQualityStep](#) [VideoRendererPriority](#) [TVideoQuality](#)

Created with the Standard Edition of HelpNDoc: [Free iPhone documentation generator](#)

IsWDMVideoDriver

TVideoGrabber.IsWDMVideoDriver

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Returns true if the video capture driver is WDM compliant.

Declaration

property IsWDMVideoDriver: Boolean **read** GetIsWDMVideoDriver;

__property **bool** IsWDMVideoDriver=read=GetIsWDMVideoDriver, **nodefault**

Property IsWDMVideoDriver As Boolean

Description

Returns:

- true if the current video capture device driver is a WDM driver,
- false if the current video capture device driver is an "old" VFW driver.

See Also

[WDM drivers](#) [AutoConnectRelatedPins](#) [CurrentFrameRate](#) [DeliveredFrames](#) [DroppedFrames](#) [FixFlickerOrBlackCapture](#) [IsDVDevice](#) [IsVideoControlAvailable](#) [IsVideoCrossbarAvailable](#) [IsVideoDeviceConnected](#) [OnNoVideoDevices](#) [OnVideoDeviceSelected](#) [RefreshDevicesAndCompressorsLists](#) [ResetVideoDeviceSettings](#) [ShowDialog](#) [VideoDevice](#) [VideoDeviceIndex](#) [VideoDeviceIndexFromId](#) [VideoDeviceName](#) [VideoDevices](#) [VideoDevicesCount](#) [VideoDevicesId](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Edit and Export Markdown Documents](#)

JPEGPerformance

TVideoGrabber.JPEGPerformance

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Controls the trade-off between color quality and speed of decompression of JPEG files created by the frame grabber.

Declaration

property JPEGPerformance: TJPEGPerformance **read** GetJPEGPerformance **write** SetJPEGPerformance **default** DEF_JPEGPerformance;

__property Jpeg::TJPEGPerformance JPEGPerformance=read=GetJPEGPerformance, write=SetJPEGPerformance, **default**=0

Property JPEGPerformance As TxJPEGPerformance

Description

Use JPEGPerformance to control the trade-off between color quality and speed of decompression of JPEG files created during [frame capture](#).

See Also

[TAutoFileName](#) [TFrameCaptureDest](#) [TOnFrameCaptureCompleted](#) [BurstCount](#) [BurstInterval](#) [BurstMode](#) [BurstType](#) [CaptureFrameSyncTo](#) [CaptureFrameTo](#) [FrameCaptureHeight](#) [FrameCaptureWidth](#) [FrameCaptureWithoutOverlay](#) [FrameCaptureZoomSize](#) [GetFrameInfoString](#) [GetLastFrameAsHBITMAP](#) [GetLastFrameAsTBitmap](#) [GetLastFrameBitmapBits](#) [GetLastFrameBitmapBits2](#) [GetLastFrameWaitTimeoutMs](#) [JPEGProgressiveDisplay](#) [JPEGQuality](#) [Last](#) [BurstFrameCapture](#) [FileName](#) [Last](#) [CaptureFrameTo](#) [FileName](#) [OnDiskFull](#) [OnFrameBitmap](#) [SetFrameCaptureBounds](#) [StoragePath](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Benefits of a Help Authoring Tool](#)

JPEGProgressiveDisplay

TVideoGrabber.JPEGProgressiveDisplay

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Turns on or off the incremental display of an image when reading in JPEG files created by the frame grabber.

Declaration

property JPEGProgressiveDisplay: Boolean **read** GetJPEGProgressiveDisplay **write** SetJPEGProgressiveDisplay **default** DEF_JPEGProgressiveDisplay;

__property **bool** JPEGProgressiveDisplay=read=GetJPEGProgressiveDisplay, write=SetJPEGProgressiveDisplay, **default**=0

Property JPEGProgressiveDisplay As Boolean

Description

Use ProgressiveDisplay to turn on or off the incremental display of an image when reading in JPEG files created during [frame capture](#).

See Also

[TAutoFileName](#) [TFrameCaptureDest](#) [TOnFrameCaptureCompleted](#) [BurstCount](#) [BurstInterval](#) [BurstMode](#) [BurstType](#) [CaptureFrameSyncTo](#) [CaptureFrameTo](#) [FrameCaptureHeight](#) [FrameCaptureWidth](#) [FrameCaptureWithoutOverlay](#) [FrameCaptureZoomSize](#) [GetFrameInfoString](#) [GetLastFrameAsHBITMAP](#) [GetLastFrameAsTBitmap](#) [GetLastFrameBitmapBits](#) [GetLastFrameBitmapBits2](#) [GetLastFrameWaitTimeoutMs](#) [JPEGPerformance](#) [JPEGQuality](#) [Last](#) [BurstFrameCapture](#) [FileName](#) [Last](#) [CaptureFrameTo](#) [FileName](#) [OnDiskFull](#) [OnFrameBitmap](#) [SetFrameCaptureBounds](#) [StoragePath](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Produce electronic books easily](#)

JPEGQuality

TVideoGrabber.JPEGQuality

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Sets the compression quality of a JPEG image created by the frame grabber.

Declaration

property JPEGQuality: TJPEGQualityRange **read** GetJPEGQuality **write** SetJPEGQuality **default** DEF_JPEGQuality;

__property Jpeg::TJPEGQualityRange JPEGQuality=read=GetJPEGQuality, write=SetJPEGQuality, **default**=100

Property JPEGQuality As Integer

Description

Use CompressionQuality to set the compression quality of a JPEG image during [frame capture](#). Higher compression results in a poorer picture quality, but a smaller file size.

See Also

[TAutoFileName](#) [TFrameCaptureDest](#) [TOnFrameCaptureCompleted](#) [BurstCount](#) [BurstInterval](#) [BurstMode](#) [BurstType](#) [CaptureFrameSyncTo](#) [CaptureFrameTo](#) [FrameCaptureHeight](#) [FrameCaptureWidth](#) [FrameCaptureWithoutOverlay](#) [FrameCaptureZoomSize](#) [GetFrameInfoString](#) [GetLastFrameAsHBITMAP](#) [GetLastFrameAsTBitmap](#) [GetLastFrameBitmapBits](#) [GetLastFrameBitmapBits2](#) [GetLastFrameWaitTimeoutMs](#) [JPEGPerformance](#) [JPEGProgressiveDisplay](#) [Last](#) [BurstFrameCapture](#) [FileName](#) [Last](#) [CaptureFrameTo](#) [FileName](#) [OnDiskFull](#) [OnFrameBitmap](#) [SetFrameCaptureBounds](#) [StoragePath](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Streamline Your Documentation Creation with a Help Authoring Tool](#)

Last_BurstFrameCapture_FileName

TVideoGrabber.Last_BurstFrameCapture_FileName

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Name of the last BMP or JPEG file created by the frame capture in burst mode

Declaration

property Last_BurstFrameCapture_FileName: **string read** GetLast_BurstFrameCapture_FileName;

__property wchar_t *Last_BurstFrameCapture_FileName=read=GetLast_BurstFrameCapture_FileName

Property Last_BurstFrameCapture_FileName As String

Description

Retrieves the name of the last BMP or JPEG file created automatically by the frame grabber in burst mode. This property should be invoked from the [OnFrameCaptureCompleted](#) event.

See Also

[TAutoFileName](#) [TFrameCaptureDest](#) [TOnFrameCaptureCompleted](#) [BurstCount](#) [BurstInterval](#) [BurstMode](#) [BurstType](#) [CaptureFrameSyncTo](#) [CaptureFrameTo](#) [FrameCaptureHeight](#) [FrameCaptureWidth](#) [FrameCaptureWithoutOverlay](#) [FrameCaptureZoomSize](#) [GetFrameInfoString](#) [GetLastFrameAsHBITMAP](#) [GetLastFrameAsTBitmap](#) [GetLastFrameBitmapBits](#) [GetLastFrameBitmapBits2](#) [GetLastFrameWaitTimeoutMs](#) [JPEGPerformance](#) [JPEGProgressiveDisplay](#) [JPEGQuality](#) [Last_CaptureFrameTo_FileName](#) [OnDiskFull](#) [OnFrameBitmap](#) [SetFrameCaptureBounds](#) [StoragePath](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Reach: Convert Your Word Document to an ePub or Kindle eBook](#)

Last_CaptureFrameTo_FileName

TVideoGrabber.Last_CaptureFrameTo_FileName

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Name of the last BMP or JPEG file created by CaptureFrameTo.

Declaration

property Last_CaptureFrameTo_FileName: **string read** GetLast_CaptureFrameTo_FileName;

__property wchar_t *Last_CaptureFrameTo_FileName=read=GetLast_CaptureFrameTo_FileName

Property Last_CaptureFrameTo_FileName As String

Description

Retrieves the name of the last BMP or JPEG file created automatically by [CaptureFrameTo](#). This property should be invoked from the [OnFrameCaptureCompleted](#) event.

See Also

[TAutoFileName](#) [TFrameCaptureDest](#) [TOnFrameCaptureCompleted](#) [BurstCount](#) [BurstInterval](#) [BurstMode](#) [BurstType](#) [CaptureFrameSyncTo](#) [CaptureFrameTo](#) [FrameCaptureHeight](#) [FrameCaptureWidth](#) [FrameCaptureWithoutOverlay](#) [FrameCaptureZoomSize](#) [GetFrameInfoString](#) [GetLastFrameAsHBITMAP](#) [GetLastFrameAsTBitmap](#) [GetLastFrameBitmapBits](#) [GetLastFrameBitmapBits2](#) [GetLastFrameWaitTimeoutMs](#) [JPEGPerformance](#) [JPEGProgressiveDisplay](#) [JPEGQuality](#) [Last_BurstFrameCapture_FileName](#) [OnDiskFull](#) [OnFrameBitmap](#) [SetFrameCaptureBounds](#) [StoragePath](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Create Professional CHM Help Files with HelpNDoc's Easy-to-Use Tool](#)

Last_Clip_Played

TVideoGrabber.Last_Clip_Played

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Path/file name of the last video clip played.

Declaration

property Last_Clip_Played: **string read** GetLast_Clip_Played;

__property wchar_t *Last_Clip_Played=read=GetLast_Clip_Played

Property Last_Clip_Played As String

Description

Retrieves the path/file name of the last video clip played.

E.g.:

```
procedure TfrmMainForm.Button1Click(Sender: TObject);
begin
    VideoGrabber.PlayerFileName := VideoGrabber.Last_Clip_Played;
    VideoGrabber.OpenPlayer;
end;
```

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchronise](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: Effortlessly create a professional-quality documentation website with HelpNDoc

Last_Recording_FileName

TVideoGrabber.Last_Recording_FileName

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Name of the last AVI file created

Declaration

property Last_Recording_FileName: **string read** GetLast_Recording_FileName;

__property wchar_t *Last_Recording_FileName=read=GetLast_Recording_FileName

Property Last_Recording_FileName As String

Description

Retrieves the name of the last AVI file created (useful when file name are created automatically, see [recording](#)).

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDDeinterlaceMode](#) [TAutoFileName](#)

[TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: Save time and frustration with HelpNDoc's WinHelp HLP to CHM conversion feature

LogoDisplayed

TVideoGrabber.LogoDisplayed

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables / disables the video window logo.

Declaration

property LogoDisplayed: Boolean **read** GetLogoDisplayed **write** SetLogoDisplayed **default** DEF_LogoDisplayed;

__property **bool** LogoDisplayed=read=GetLogoDisplayed, write=SetLogoDisplayed, **default**=0

Property LogoDisplayed As Boolean

Description

Used to enable or disable the logo displayed in the inactive video window.

See Also

[Logo displayed in the video window](#) [LogoLayout](#) [SetLogoFromBMPFile](#) [SetLogoFromHBitmap](#) [SetLogoFromJPEGFile](#) [SetLogoFromTBitmap](#)

Created with the Standard Edition of HelpNDoc: Full-featured multi-format Help generator

LogoLayout

TVideoGrabber.LogoLayout

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies how the logo must be displayed.

Declaration

property LogoLayout: TLogoLayout **read** GetLogoLayout **write** SetLogoLayout **default** DEF_LogoLayout;

__property TLogoLayout LogoLayout=read=GetLogoLayout, write=SetLogoLayout, **default**=1

Property LogoLayout As TLogoLayout

Description

This [TLogoLayout](#) property is used to specify if the logo must be:

- centered (lg_Centered)

- stretched to fit the video window (lg_Stretched)
- repeated to fit the video window (lg_Repeated).

See Also

[Logo displayed in the video window](#) [LogoDisplayed](#) [SetLogoFromBMPFile](#) [SetLogoFromHBitmap](#) [SetLogoFromJPEGFile](#) [SetLogoFromTBitmap](#)

Created with the Standard Edition of HelpNDoc: [Add an Extra Layer of Security to Your PDFs with Encryption](#)

MixAudioSamples_CurrentSourceLevel

| | | |
|--|----------------------|----------------------|
| TVideoGrabber.MixAudioSamples_CurrentSourceLevel | Prev | Next |
| TVideoGrabber Properties | | |

specifies the level of the current source

Declaration

property MixAudioSamples_CurrentSourceLevel: LongInt **index** AUDMIX_CURRENT **read** GetMixAudioSamplesLevel **write** SetMixAudioSamplesLevel **default** 100;

__property int MixAudioSamples_CurrentSourceLevel==GetMixAudioSamplesLevel, write=SetMixAudioSamplesLevel, index=0, **default**=100;

property MixAudioSamples_CurrentSourceLevel as Long

Description

Used to specify the level of the current source, in percentage, when mixing audio samples by using [MixAudioSamples](#).

See Also

[MixAudioSamples](#) [MixAudioSamples_ExternalSourceLevel](#) [Mixer_SetupPIPFromSource](#)

Created with the Standard Edition of HelpNDoc: [Transform Your Word Document into a Professional eBook with HelpNDoc](#)

MixAudioSamples_ExternalSourceLevel

| | | |
|--|----------------------|----------------------|
| TVideoGrabber.MixAudioSamples_ExternalSourceLevel | Prev | Next |
| TVideoGrabber Properties | | |

specifies the level of the external source

Declaration

property MixAudioSamples_ExternalSourceLevel: LongInt **index** AUDMIX_EXTERN **read** GetMixAudioSamplesLevel **write** SetMixAudioSamplesLevel **default** DEF_MixAudioSamplesLevel;

__property int MixAudioSamples_ExternalSourceLevel==GetMixAudioSamplesLevel, write=SetMixAudioSamplesLevel, index=1, **default**=100;

property MixAudioSamples_ExternalSourceLevel as Long

Description

Used to specify the level of the external source, in percentage, when mixing audio samples by using [MixAudioSamples](#).

See Also

[MixAudioSamples](#) [MixAudioSamples_CurrentSourceLevel](#) [Mixer_SetupPIPFromSource](#)

Mixer_MosaicColumns

TVideoGrabber.Mixer_MosaicColumns

[Prev](#)[Next](#)[TVideoGrabber](#) [Properties](#)

Specifies the number of mosaic columns in Mixer mode.

Declaration

property Mixer_MosaicColumns: LongInt **read** GetMixer_MosaicColumns **write** SetMixer_MosaicColumns
default DEF_Mixer_MosaicColumns;

__property int Mixer_MosaicColumns==GetMixer_MosaicColumns, write=SetMixer_MosaicColumns,
default=1;

Property Mixer_MosaicColumns as Long

Description

Used to specify the number of columns when the component is used to mix several video sources into a single one in a mosaic layout (see [How to mix several video sources into one a single one](#)).

See Also

[Mixer_Activation](#) [Mixer_AddToMixer](#) [Mixer_MosaicLines](#) [Mixer_RemoveFromMixer](#)
[Mixer_SetOverlayRoundedCorner](#) [Mixer_SetupPIPFromSource](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly upgrade your WinHelp HLP help files to CHM with HelpNDoc](#)

Mixer_MosaicLines

TVideoGrabber.Mixer_MosaicLines

[Prev](#)[Next](#)[TVideoGrabber](#) [Properties](#)

Specifies the number of mosaic lines in Mixer mode.

Declaration

property Mixer_MosaicLines: LongInt **read** GetMixer_MosaicLines **write** SetMixer_MosaicLines **default**
DEF_Mixer_MosaicLines;

__property int Mixer_MosaicLines==GetMixer_MosaicLines, write=SetMixer_MosaicLines, **default**=1;

Property Mixer_MosaicLines as Long

Description

Used to specify the number of lines when the component is used to mix several video sources into a single one in a mosaic layout (see [How to mix several video sources into one a single one](#)).

See Also

[Mixer_Activation](#) [Mixer_AddToMixer](#) [Mixer_MosaicColumns](#) [Mixer_RemoveFromMixer](#)
[Mixer_SetOverlayRoundedCorner](#) [Mixer_SetupPIPFromSource](#)

Created with the Standard Edition of HelpNDoc: [Generate EPub eBooks with ease](#)

MotionDetector_CompareBlue

TVideoGrabber.MotionDetector_CompareBlue

[Prev](#)[Next](#)

TVideoGrabber **Properties**

Enables/disables the use of the blue color to detect motion.

Declaration

property MotionDetector_CompareBlue: Boolean **read** GetMotionDetector_CompareBlue **write** SetMotionDetector_CompareBlue **default** DEF_MotionDetector_CompareBlue;

__property **bool** MotionDetector_CompareBlue=read=GetMotionDetector_CompareBlue, write=SetMotionDetector_CompareBlue, **default**=1

Property MotionDetector_CompareBlue As Boolean

Description

Used to enable/disable the use of the blue color to detect motion.

If [MotionDetector_GreyScale](#) is enabled this property is ignored.

See Also

[Color / Greyscale Grid structure / grid sensitivity Motion ratio Recording only when motion is detected Video noise TOnMotionDetected TOnMotionNotDetected MotionDetector_CellMotionRatio MotionDetector_CompareGreen MotionDetector_CompareRed MotionDetector_Enabled MotionDetector_EnumGridDialogControls MotionDetector_Get2DTextGrid MotionDetector_Get2DTextMotion MotionDetector_GetCellLocation MotionDetector_GetCellSensitivity MotionDetector_GetCellSize MotionDetector_GloballyIncOrDecSensitivity MotionDetector_GlobalMotionRatio MotionDetector_GreyScale MotionDetector_Grid MotionDetector_GridXCount MotionDetector_GridYCount MotionDetector_IsGridValid MotionDetector_MaxDetectionsPerSecond MotionDetector_ReduceCPULoad MotionDetector_ReduceVideoNoise MotionDetector_Reset MotionDetector_ResetGlobalSensitivity MotionDetector_SetCellSensitivity MotionDetector_SetGridSize MotionDetector_ShowGridDialog MotionDetector_Triggered MotionDetector_UseThisReferenceSample OnBacktimedFramesCountReached OnMotionDetected OnMotionNotDetected RecordingOnMotion_Enabled RecordingOnMotion_MotionThreshold RecordingOnMotion_NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: [Easily create EPub books](#)

MotionDetector_CompareGreen

TVideoGrabber.MotionDetector_CompareGreen

[Prev](#)

[Next](#)

TVideoGrabber **Properties**

Enables/disables the use of the green color to detect motion.

Declaration

property MotionDetector_CompareGreen: Boolean **read** GetMotionDetector_CompareGreen **write** SetMotionDetector_CompareGreen **default** DEF_MotionDetector_CompareGreen;

__property **bool** MotionDetector_CompareGreen=read=GetMotionDetector_CompareGreen, write=SetMotionDetector_CompareGreen, **default**=1

Property MotionDetector_CompareGreen As Boolean

Description

Used to enable/disable the use of the green color to detect motion.

If [MotionDetector_GreyScale](#) is enabled this property is ignored.

See Also

[Color / Greyscale Grid structure / grid sensitivity Motion ratio Recording only when motion is detected Video noise TOnMotionDetected TOnMotionNotDetected MotionDetector_CellMotionRatio MotionDetector_CompareBlue MotionDetector_CompareRed MotionDetector_Enabled MotionDetector_EnumGridDialogControls MotionDetector_Get2DTextGrid MotionDetector_Get2DTextMotion](#)

[MotionDetector_GetCellLocation](#)
[MotionDetector_GetCellSensitivity](#)
[MotionDetector_GetCellSize](#)
[MotionDetector_GloballyIncOrDecSensitivity](#)
[MotionDetector_GlobalMotionRatio](#)
[MotionDetector_GreyScale](#)
[MotionDetector_Grid](#)
[MotionDetector_GridXCount](#)
[MotionDetector_GridYCount](#)
[MotionDetector_IsGridValid](#)
[MotionDetector_MaxDetectionsPerSecond](#)
[MotionDetector_ReduceCPULoad](#)
[MotionDetector_ReduceVideoNoise](#)
[MotionDetector_Reset](#)
[MotionDetector_ResetGlobalSensitivity](#)
[MotionDetector_SetCellSensitivity](#)
[MotionDetector_SetGridSize](#)
[MotionDetector_ShowGridDialog](#)
[MotionDetector_Triggered](#)
[MotionDetector_UseThisReferenceSample](#)
[OnBacktimedFramesCountReached](#)
[OnMotionDetected](#)
[OnMotionNotDetected](#)
[RecordingOnMotion_Enabled](#)
[RecordingOnMotion_MotionThreshold](#)
[RecordingOnMotion_NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: [Easily create PDF Help documents](#)

MotionDetector_CompareRed

TVideoGrabber.MotionDetector_CompareRed

[Prev](#)

[Next](#)

[TVideoGrabber](#)
[Properties](#)

Enables/disables the use of the red color to detect motion.

Declaration

property MotionDetector_CompareRed: Boolean **read** GetMotionDetector_CompareRed **write** SetMotionDetector_CompareRed **default** DEF_MotionDetector_CompareRed;

__property **bool** MotionDetector_CompareRed=read=GetMotionDetector_CompareRed, write=SetMotionDetector_CompareRed, **default**=1

Property MotionDetector_CompareRed As Boolean

Description

Used to enable/disable the use of the red color to detect motion.

If [MotionDetector_GreyScale](#) is enabled this property is ignored.

See Also

[Color / Greyscale Grid structure / grid sensitivity](#)
[Motion ratio](#)
[Recording only when motion is detected](#)
[Video noise](#)
[TOnMotionDetected](#)
[TOnMotionNotDetected](#)
[MotionDetector_CellMotionRatio](#)
[MotionDetector_CompareBlue](#)
[MotionDetector_CompareGreen](#)
[MotionDetector_Enabled](#)
[MotionDetector_EnumGridDialogControls](#)
[MotionDetector_Get2DTextGrid](#)
[MotionDetector_Get2DTextMotion](#)
[MotionDetector_GetCellLocation](#)
[MotionDetector_GetCellSensitivity](#)
[MotionDetector_GetCellSize](#)
[MotionDetector_GloballyIncOrDecSensitivity](#)
[MotionDetector_GlobalMotionRatio](#)
[MotionDetector_GreyScale](#)
[MotionDetector_Grid](#)
[MotionDetector_GridXCount](#)
[MotionDetector_GridYCount](#)
[MotionDetector_IsGridValid](#)
[MotionDetector_MaxDetectionsPerSecond](#)
[MotionDetector_ReduceCPULoad](#)
[MotionDetector_ReduceVideoNoise](#)
[MotionDetector_Reset](#)
[MotionDetector_ResetGlobalSensitivity](#)
[MotionDetector_SetCellSensitivity](#)
[MotionDetector_SetGridSize](#)
[MotionDetector_ShowGridDialog](#)
[MotionDetector_Triggered](#)
[MotionDetector_UseThisReferenceSample](#)
[OnBacktimedFramesCountReached](#)
[OnMotionDetected](#)
[OnMotionNotDetected](#)
[RecordingOnMotion_Enabled](#)
[RecordingOnMotion_MotionThreshold](#)
[RecordingOnMotion_NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: [Free EPub producer](#)

MotionDetector_Enabled

TVideoGrabber.MotionDetector_Enabled

[Prev](#)

[Next](#)

[TVideoGrabber](#)
[Properties](#)

Enables / disables motion detection.

Declaration

property MotionDetector_Enabled: Boolean **read** GetMotionDetector_Enabled **write** SetMotionDetector_Enabled **default** DEF_MotionDetector_Enabled;

__property **bool** MotionDetector_Enabled=read=GetMotionDetector_Enabled,
write=SetMotionDetector_Enabled, **default**=0

Property MotionDetector_Enabled As Boolean

Description

Used to enable / disable motion detection.

- when disabled, all the motion detection features are not used.
- when enabled, motion detection features are in use and the [OnMotionDetected](#) event occurs upon motion detection.

See Also

[Color / Greyscale Grid structure / grid sensitivity Motion ratio Recording only when motion is detected](#)
[Video noise TOnMotionDetected TOnMotionNotDetected MotionDetector_CellMotionRatio](#)
[MotionDetector_CompareBlue MotionDetector_CompareGreen MotionDetector_CompareRed](#)
[MotionDetector_EnumGridDialogControls MotionDetector_Get2DTextGrid MotionDetector_Get2DTextMotion](#)
[MotionDetector_GetCellLocation MotionDetector_GetCellSensitivity MotionDetector_GetCellSize](#)
[MotionDetector_GloballyIncOrDecSensitivity MotionDetector_GlobalMotionRatio MotionDetector_GreyScale](#)
[MotionDetector_Grid MotionDetector_GridXCount MotionDetector_GridYCount MotionDetector_IsGridValid](#)
[MotionDetector_MaxDetectionsPerSecond MotionDetector_ReduceCPULoad](#)
[MotionDetector_ReduceVideoNoise MotionDetector_Reset MotionDetector_ResetGlobalSensitivity](#)
[MotionDetector_SetCellSensitivity MotionDetector_SetGridSize MotionDetector_ShowGridDialog](#)
[MotionDetector_Triggered MotionDetector_UseThisReferenceSample OnBacktimedFramesCountReached](#)
[OnMotionDetected OnMotionNotDetected RecordingOnMotion_Enabled](#)
[RecordingOnMotion MotionThreshold RecordingOnMotion_NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: [Elevate Your Help Documentation with a Help Authoring Tool](#)

MotionDetector_GlobalMotionRatio

TVideoGrabber.MotionDetector_GlobalMotionRatio

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Retrieves the last motion ratio.

Declaration

property MotionDetector_GlobalMotionRatio: Double **read** GetMotionDetector_GlobalMotionRatio;

__property **double** MotionDetector_GlobalMotionRatio=read=GetMotionDetector_GlobalMotionRatio

Function MotionDetector_CellMotionRatio(x As Long, y As Long) As Double

Description

Used to retrieve the last motion ratio (moved pixels count / total pixels count) of the grid's cells having their sensitivity enabled (value between 1 and 9).

The return value will be in the **0.0** to **1.0** range (e.g. 0.5775).

This value is the result of the motion detection of the last video frame received.

This value is also returned by the [OnMotionDetected](#) event.

See Also

[Color / Greyscale Grid structure / grid sensitivity Motion ratio Recording only when motion is detected](#)
[Video noise TOnMotionDetected TOnMotionNotDetected MotionDetector_CellMotionRatio](#)
[MotionDetector_CompareBlue MotionDetector_CompareGreen MotionDetector_CompareRed](#)
[MotionDetector_Enabled MotionDetector_EnumGridDialogControls MotionDetector_Get2DTextGrid](#)
[MotionDetector_Get2DTextMotion MotionDetector_GetCellLocation MotionDetector_GetCellSensitivity](#)
[MotionDetector_GetCellSize MotionDetector_GloballyIncOrDecSensitivity MotionDetector_GreyScale](#)
[MotionDetector_Grid MotionDetector_GridXCount MotionDetector_GridYCount MotionDetector_IsGridValid](#)
[MotionDetector_MaxDetectionsPerSecond MotionDetector_ReduceCPULoad](#)

[MotionDetector_ReduceVideoNoise](#) [MotionDetector_Reset](#) [MotionDetector_ResetGlobalSensitivity](#) [MotionDetector_SetCellSensitivity](#) [MotionDetector_SetGridSize](#) [MotionDetector_ShowGridDialog](#) [MotionDetector_Triggered](#) [MotionDetector_UseThisReferenceSample](#) [OnBacktimedFramesCountReached](#) [OnMotionDetected](#) [OnMotionNotDetected](#) [RecordingOnMotion_Enabled](#) [RecordingOnMotion_MotionThreshold](#) [RecordingOnMotion_NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: Effortlessly Spot and Fix Problems in Your Documentation with HelpNDoc's Project Analyzer

MotionDetector_GreyScale

TVideoGrabber.MotionDetector_GreyScale

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables / disables motion detection on a greyscale basis.

Declaration

property MotionDetector_GreyScale: Boolean **read** GetMotionDetector_GreyScale **write** SetMotionDetector_GreyScale **default** DEF_MotionDetector_GreyScale;

__property **bool** MotionDetector_GreyScale=read=GetMotionDetector_GreyScale, write=SetMotionDetector_GreyScale, **default**=0

Property MotionDetector_GreyScale As Boolean

Description

If enabled, motion detection is performed on a greyscale basis, instead of the default RGB basis. In this case, the [MotionDetector_CompareRed](#), [MotionDetector_CompareBlue](#) and [MotionDetector_CompareGreen](#) properties are ignored.

See Also

[Color / Greyscale Grid structure / grid sensitivity](#) [Motion ratio](#) [Recording only when motion is detected](#) [Video noise](#) [TOnMotionDetected](#) [TOnMotionNotDetected](#) [MotionDetector_CellMotionRatio](#) [MotionDetector_CompareBlue](#) [MotionDetector_CompareGreen](#) [MotionDetector_CompareRed](#) [MotionDetector_Enabled](#) [MotionDetector_EnumGridDialogControls](#) [MotionDetector_Get2DTextGrid](#) [MotionDetector_Get2DTextMotion](#) [MotionDetector_GetCellLocation](#) [MotionDetector_GetCellSensitivity](#) [MotionDetector_GetCellSize](#) [MotionDetector_GloballyIncOrDecSensitivity](#) [MotionDetector_GlobalMotionRatio](#) [MotionDetector_Grid](#) [MotionDetector_GridXCount](#) [MotionDetector_GridYCount](#) [MotionDetector_IsGridValid](#) [MotionDetector_MaxDetectionsPerSecond](#) [MotionDetector_ReduceCPULoad](#) [MotionDetector_ReduceVideoNoise](#) [MotionDetector_Reset](#) [MotionDetector_ResetGlobalSensitivity](#) [MotionDetector_SetCellSensitivity](#) [MotionDetector_SetGridSize](#) [MotionDetector_ShowGridDialog](#) [MotionDetector_Triggered](#) [MotionDetector_UseThisReferenceSample](#) [OnBacktimedFramesCountReached](#) [OnMotionDetected](#) [OnMotionNotDetected](#) [RecordingOnMotion_Enabled](#) [RecordingOnMotion_MotionThreshold](#) [RecordingOnMotion_NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: Easy EPub and documentation editor

MotionDetector_Grid

TVideoGrabber.MotionDetector_Grid

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Sets or retrieves the grid structure, expressed as a simple string.

Declaration

property MotionDetector_Grid: **string** **read** GetMotionDetector_Grid **write** SetMotionDetector_Grid;

__property wchar_t *MotionDetector_Grid=read=GetMotionDetector_Grid, write=SetMotionDetector_Grid

Property MotionDetector_Grid As String

Description

Used to set or retrieve the sensitivity grid structure, expressed as a simple string.
After modifying the grid structure, test [MotionDetector_IsGridValid](#) to check if the grid has been accepted.

By example, the sensitivity grid:

```
0 0 0 0 4 4 4 4 1
5 5 5 5 5 5 5 5 1
4 4 4 4 4 4 4 4 1
5 5 5 5 5 5 5 5 1
6 6 6 6 6 6 6 6 1
6 6 6 6 6 6 6 6 1
6 6 6 6 6 6 6 6 1
3 3 3 3 3 3 3 3 1
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
```

has for MotionDetector_Grid value:

```
"0000044441 555555551 444444441 555555551 666666661 666666661 666666661 333333331
0000000000 0000000000"
```

See Also

[Color / Greyscale Grid structure / grid sensitivity](#) [Motion ratio](#) [Recording only when motion is detected](#)
[Video noise](#) [TOnMotionDetected](#) [TOnMotionNotDetected](#) [MotionDetector_CellMotionRatio](#)
[MotionDetector_CompareBlue](#) [MotionDetector_CompareGreen](#) [MotionDetector_CompareRed](#)
[MotionDetector_Enabled](#) [MotionDetector_EnumGridDialogControls](#) [MotionDetector_Get2DTextGrid](#)
[MotionDetector_Get2DTextMotion](#) [MotionDetector_GetCellLocation](#) [MotionDetector_GetCellSensitivity](#)
[MotionDetector_GetCellSize](#) [MotionDetector_GloballyIncOrDecSensitivity](#)
[MotionDetector_GlobalMotionRatio](#) [MotionDetector_GreyScale](#) [MotionDetector_GridXCount](#)
[MotionDetector_GridYCount](#) [MotionDetector_IsGridValid](#) [MotionDetector_MaxDetectionsPerSecond](#)
[MotionDetector_ReduceCPULoad](#) [MotionDetector_ReduceVideoNoise](#) [MotionDetector_Reset](#)
[MotionDetector_ResetGlobalSensitivity](#) [MotionDetector_SetCellSensitivity](#) [MotionDetector_SetGridSize](#)
[MotionDetector_ShowGridDialog](#) [MotionDetector_Triggered](#) [MotionDetector_UseThisReferenceSample](#)
[OnBacktimedFramesCountReached](#) [OnMotionDetected](#) [OnMotionNotDetected](#) [RecordingOnMotion_Enabled](#)
[RecordingOnMotion_MotionThreshold](#) [RecordingOnMotion_NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: Streamline your documentation process with HelpNDoc's
WinHelp HLP to CHM conversion feature

MotionDetector_GridXCount

TVideoGrabber.MotionDetector_GridXCount

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Retrieves the number of columns of the grid.

Declaration

property MotionDetector_GridXCount: LongInt **read** GetMotionDetector_GridXCount;

___property **int** MotionDetector_GridXCount=read=GetMotionDetector_GridXCount, **nodefault**

Property MotionDetector_GridXCount As Long

Description

Used to retrieve the number of columns of the grid.

See Also

[Color / Greyscale Grid structure / grid sensitivity](#) [Motion ratio](#) [Recording only when motion is detected](#)
[Video noise](#) [TOnMotionDetected](#) [TOnMotionNotDetected](#) [MotionDetector_CellMotionRatio](#)

[MotionDetector_CompareBlue](#)
[MotionDetector_CompareGreen](#)
[MotionDetector_CompareRed](#)
[MotionDetector_Enabled](#)
[MotionDetector_EnumGridDialogControls](#)
[MotionDetector_Get2DTextGrid](#)
[MotionDetector_Get2DTextMotion](#)
[MotionDetector_GetCellLocation](#)
[MotionDetector_GetCellSensitivity](#)
[MotionDetector_GetCellSize](#)
[MotionDetector_GloballyIncOrDecSensitivity](#)
[MotionDetector_GlobalMotionRatio](#)
[MotionDetector_GreyScale](#)
[MotionDetector_Grid](#)
[MotionDetector_GridYCount](#)
[MotionDetector_IsGridValid](#)
[MotionDetector_MaxDetectionsPerSecond](#)
[MotionDetector_ReduceCPULoad](#)
[MotionDetector_ReduceVideoNoise](#)
[MotionDetector_Reset](#)
[MotionDetector_ResetGlobalSensitivity](#)
[MotionDetector_SetCellSensitivity](#)
[MotionDetector_SetGridSize](#)
[MotionDetector_ShowGridDialog](#)
[MotionDetector_Triggered](#)
[MotionDetector_UseThisReferenceSample](#)
[OnBacktimedFramesCountReached](#)
[OnMotionDetected](#)
[OnMotionNotDetected](#)
[RecordingOnMotion_Enabled](#)
[RecordingOnMotion_MotionThreshold](#)
[RecordingOnMotion_NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: Experience a User-Friendly Interface with HelpNDoc's Documentation Tool

MotionDetector_GridYCount

TVideoGrabber.MotionDetector_GridYCount

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Retrieves the number of rows of the grid.

Declaration

property MotionDetector_GridYCount: LongInt **read** GetMotionDetector_GridYCount;

__property **int** MotionDetector_GridYCount=read=GetMotionDetector_GridYCount, **nodefault**

Property MotionDetector_GridYCount As Long

Description

Used to retrieve the number of rows of the grid.

See Also

[Color / Greyscale Grid structure / grid sensitivity](#)
[Motion ratio](#)
[Recording only when motion is detected](#)
[Video noise](#)
[TOnMotionDetected](#)
[TOnMotionNotDetected](#)
[MotionDetector_CellMotionRatio](#)
[MotionDetector_CompareBlue](#)
[MotionDetector_CompareGreen](#)
[MotionDetector_CompareRed](#)
[MotionDetector_Enabled](#)
[MotionDetector_EnumGridDialogControls](#)
[MotionDetector_Get2DTextGrid](#)
[MotionDetector_Get2DTextMotion](#)
[MotionDetector_GetCellLocation](#)
[MotionDetector_GetCellSensitivity](#)
[MotionDetector_GetCellSize](#)
[MotionDetector_GloballyIncOrDecSensitivity](#)
[MotionDetector_GlobalMotionRatio](#)
[MotionDetector_GreyScale](#)
[MotionDetector_Grid](#)
[MotionDetector_GridXCount](#)
[MotionDetector_IsGridValid](#)
[MotionDetector_MaxDetectionsPerSecond](#)
[MotionDetector_ReduceCPULoad](#)
[MotionDetector_ReduceVideoNoise](#)
[MotionDetector_Reset](#)
[MotionDetector_ResetGlobalSensitivity](#)
[MotionDetector_SetCellSensitivity](#)
[MotionDetector_SetGridSize](#)
[MotionDetector_ShowGridDialog](#)
[MotionDetector_Triggered](#)
[MotionDetector_UseThisReferenceSample](#)
[OnBacktimedFramesCountReached](#)
[OnMotionDetected](#)
[OnMotionNotDetected](#)
[RecordingOnMotion_Enabled](#)
[RecordingOnMotion_MotionThreshold](#)
[RecordingOnMotion_NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: Streamline Your Documentation Creation with a Help Authoring Tool

MotionDetector_IsGridValid

TVideoGrabber.MotionDetector_IsGridValid

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Returns true if the grid string is correct, false otherwise.

Declaration

property MotionDetector_IsGridValid: Boolean **read** GetMotionDetector_IsGridValid;

__property **bool** MotionDetector_IsGridValid=read=GetMotionDetector_IsGridValid, **ndefault**

Property MotionDetector_IsGridValid As Boolean

Description

Returns true if the grid string is correct, false otherwise.
See [Grid structure](#) .

See Also

[Color / Greyscale Grid structure / grid sensitivity Motion ratio Recording only when motion is detected Video noise TOnMotionDetected TOnMotionNotDetected MotionDetector_CellMotionRatio MotionDetector_CompareBlue MotionDetector_CompareGreen MotionDetector_CompareRed MotionDetector_Enabled MotionDetector_EnumGridDialogControls MotionDetector_Get2DTextGrid MotionDetector_Get2DTextMotion MotionDetector_GetCellLocation MotionDetector_GetCellSensitivity MotionDetector_GetCellSize MotionDetector_GloballyIncOrDecSensitivity MotionDetector_GlobalMotionRatio MotionDetector_GreyScale MotionDetector_Grid MotionDetector_GridXCount MotionDetector_GridYCount MotionDetector_MaxDetectionsPerSecond MotionDetector_ReduceCPULoad MotionDetector_ReduceVideoNoise MotionDetector_Reset MotionDetector_ResetGlobalSensitivity MotionDetector_SetCellSensitivity MotionDetector_SetGridSize MotionDetector_ShowGridDialog MotionDetector_Triggered MotionDetector_UseThisReferenceSample OnBacktimedFramesCountReached OnMotionDetected OnMotionNotDetected RecordingOnMotion_Enabled RecordingOnMotion_MotionThreshold RecordingOnMotion_NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: [Easy EBook and documentation generator](#)

MotionDetector_MaxDetectionsPerSecond

TVideoGrabber.MotionDetector_MaxDetectionsPerSecond

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Maximum motion detections per second.

Declaration

property MotionDetector_MaxDetectionsPerSecond: Double **read**
GetMotionDetector_MaxDetectionsPerSecond **write** SetMotionDetector_MaxDetectionsPerSecond;

__property **double**

MotionDetector_MaxDetectionsPerSecond=read=GetMotionDetector_MaxDetectionsPerSecond,
write=SetMotionDetector_MaxDetectionsPerSecond

Property MotionDetector_MaxDetectionsPerSecond As Double

Description

Used to specify the maximum number of motion detections per second.
0 = no limit (a motion detection is performed on each video frame).
E.g. 5 = 5 detections per second.

See Also

[Color / Greyscale Grid structure / grid sensitivity Motion ratio Recording only when motion is detected Video noise TOnMotionDetected TOnMotionNotDetected MotionDetector_CellMotionRatio MotionDetector_CompareBlue MotionDetector_CompareGreen MotionDetector_CompareRed MotionDetector_Enabled MotionDetector_EnumGridDialogControls MotionDetector_Get2DTextGrid MotionDetector_Get2DTextMotion MotionDetector_GetCellLocation MotionDetector_GetCellSensitivity MotionDetector_GetCellSize MotionDetector_GloballyIncOrDecSensitivity MotionDetector_GlobalMotionRatio MotionDetector_GreyScale MotionDetector_Grid MotionDetector_GridXCount MotionDetector_GridYCount MotionDetector_IsGridValid MotionDetector_ReduceCPULoad MotionDetector_ReduceVideoNoise MotionDetector_Reset MotionDetector_ResetGlobalSensitivity MotionDetector_SetCellSensitivity MotionDetector_SetGridSize](#)

[MotionDetector_ShowGridDialog](#) [MotionDetector_Triggered](#) [MotionDetector_UseThisReferenceSample](#) [OnBacktimedFramesCountReached](#) [OnMotionDetected](#) [OnMotionNotDetected](#) [RecordingOnMotion_Enabled](#) [RecordingOnMotion_MotionThreshold](#) [RecordingOnMotion_NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: [Easy EPub and documentation editor](#)

MotionDetector_ReduceCPULoad

TVideoGrabber.MotionDetector_ReduceCPULoad

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Reduces the CPU load when detecting motion on large video frames.

Declaration

property MotionDetector_ReduceCPULoad: LongInt **read** GetMotionDetector_ReduceCPULoad **write** SetMotionDetector_ReduceCPULoad **default** DEF_MotionDetector_ReduceCPULoad;

__property **int** MotionDetector_ReduceCPULoad=read=GetMotionDetector_ReduceCPULoad, write=SetMotionDetector_ReduceCPULoad, **default**=1

Property MotionDetector_ReduceCPULoad As Long

Description

Used to reduces the CPU load when detecting motion on large video frames.
Allowed values:

1: full processing, maximum accuracy

full motion detection processing is applied, whatever the video frame size.

2 to 9: "lighter" processing

Higher the value, "lighter" the processing applied to detect motion, but lower is the motion detection accuracy.

See Also

[Color / Greyscale Grid structure / grid sensitivity](#) [Motion ratio](#) [Recording only when motion is detected](#) [Video noise](#) [TOnMotionDetected](#) [TOnMotionNotDetected](#) [MotionDetector_CellMotionRatio](#) [MotionDetector_CompareBlue](#) [MotionDetector_CompareGreen](#) [MotionDetector_CompareRed](#) [MotionDetector_Enabled](#) [MotionDetector_EnumGridDialogControls](#) [MotionDetector_Get2DTextGrid](#) [MotionDetector_Get2DTextMotion](#) [MotionDetector_GetCellLocation](#) [MotionDetector_GetCellSensitivity](#) [MotionDetector_GetCellSize](#) [MotionDetector_GloballyIncOrDecSensitivity](#) [MotionDetector_GlobalMotionRatio](#) [MotionDetector_GreyScale](#) [MotionDetector_Grid](#) [MotionDetector_GridXCount](#) [MotionDetector_GridYCount](#) [MotionDetector_IsGridValid](#) [MotionDetector_MaxDetectionsPerSecond](#) [MotionDetector_ReduceVideoNoise](#) [MotionDetector_Reset](#) [MotionDetector_ResetGlobalSensitivity](#) [MotionDetector_SetCellSensitivity](#) [MotionDetector_SetGridSize](#) [MotionDetector_ShowGridDialog](#) [MotionDetector_Triggered](#) [MotionDetector_UseThisReferenceSample](#) [OnBacktimedFramesCountReached](#) [OnMotionDetected](#) [OnMotionNotDetected](#) [RecordingOnMotion_Enabled](#) [RecordingOnMotion_MotionThreshold](#) [RecordingOnMotion_NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: [Produce electronic books easily](#)

MotionDetector_ReduceVideoNoise

TVideoGrabber.MotionDetector_ReduceVideoNoise

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Reduces the sensitivity to the video noise.

Declaration

property MotionDetector_ReduceVideoNoise: Boolean **read** GetMotionDetector_ReduceVideoNoise **write**

SetMotionDetector_ReduceVideoNoise **default** DEF_MotionDetector_ReduceVideoNoise;

__property **bool** MotionDetector_ReduceVideoNoise=read=GetMotionDetector_ReduceVideoNoise,
write=SetMotionDetector_ReduceVideoNoise, **default**=0

Property MotionDetector_ReduceVideoNoise As Boolean

Description

When enabled, reduces the sensitivity to the video noise.

Enabling this property limits the risk of motion detected because of the video noise.

This processing increases the CPU load.

See Also

[Color / Greyscale Grid structure / grid sensitivity](#) [Motion ratio](#) [Recording only when motion is detected](#)

[Video noise](#) [TOnMotionDetected](#) [TOnMotionNotDetected](#) [MotionDetector_CellMotionRatio](#)

[MotionDetector_CompareBlue](#) [MotionDetector_CompareGreen](#) [MotionDetector_CompareRed](#)

[MotionDetector_Enabled](#) [MotionDetector_EnumGridDialogControls](#) [MotionDetector_Get2DTextGrid](#)

[MotionDetector_Get2DTextMotion](#) [MotionDetector_GetCellLocation](#) [MotionDetector_GetCellSensitivity](#)

[MotionDetector_GetCellSize](#) [MotionDetector_GloballyIncOrDecSensitivity](#)

[MotionDetector_GlobalMotionRatio](#) [MotionDetector_GreyScale](#) [MotionDetector_Grid](#)

[MotionDetector_GridXCount](#) [MotionDetector_GridYCount](#) [MotionDetector_IsGridValid](#)

[MotionDetector_MaxDetectionsPerSecond](#) [MotionDetector_ReduceCPULoad](#) [MotionDetector_Reset](#)

[MotionDetector_ResetGlobalSensitivity](#) [MotionDetector_SetCellSensitivity](#) [MotionDetector_SetGridSize](#)

[MotionDetector_ShowGridDialog](#) [MotionDetector_Triggered](#) [MotionDetector_UseThisReferenceSample](#)

[OnBacktimedFramesCountReached](#) [OnMotionDetected](#) [OnMotionNotDetected](#) [RecordingOnMotion_Enabled](#)

[RecordingOnMotion_MotionThreshold](#) [RecordingOnMotion_NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: Full-featured EBook editor

MotionDetector_Triggered

TVideoGrabber.MotionDetector_Triggered

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables the triggered motion detection.

Declaration

property MotionDetector_Triggered: Boolean **read** GetMotionDetector_Triggered **write**
SetMotionDetector_Triggered **default** DEF_MotionDetector_Triggered;

__property **bool** MotionDetector_Triggered=read=GetMotionDetector_Triggered,
write=SetMotionDetector_Triggered, **default**=0

Property MotionDetector_Triggered As Boolean

Description

Used to enable the triggered motion detection.

By default the motion detection occurs for each video frame.

Unlikely, when this property is enabled, the motion detection will occur only one time when
[MotionDetector_TriggerNow](#) is invoked, until the next [MotionDetector_TriggerNow](#) call.

See Also

[Color / Greyscale Grid structure / grid sensitivity](#) [Motion ratio](#) [Recording only when motion is detected](#)

[Video noise](#) [TOnMotionDetected](#) [TOnMotionNotDetected](#) [MotionDetector_CellMotionRatio](#)

[MotionDetector_CompareBlue](#) [MotionDetector_CompareGreen](#) [MotionDetector_CompareRed](#)

[MotionDetector_Enabled](#) [MotionDetector_EnumGridDialogControls](#) [MotionDetector_Get2DTextGrid](#)

[MotionDetector_Get2DTextMotion](#) [MotionDetector_GetCellLocation](#) [MotionDetector_GetCellSensitivity](#)

[MotionDetector_GetCellSize](#) [MotionDetector_GloballyIncOrDecSensitivity](#)

[MotionDetector](#) [GlobalMotionRatio](#) [MotionDetector](#) [GreyScale](#) [MotionDetector](#) [Grid](#) [MotionDetector](#) [GridXCount](#) [MotionDetector](#) [GridYCount](#) [MotionDetector](#) [IsGridValid](#) [MotionDetector](#) [MaxDetectionsPerSecond](#) [MotionDetector](#) [ReduceCPULoad](#) [MotionDetector](#) [ReduceVideoNoise](#) [MotionDetector](#) [Reset](#) [MotionDetector](#) [ResetGlobalSensitivity](#) [MotionDetector](#) [SetCellSensitivity](#) [MotionDetector](#) [SetGridSize](#) [MotionDetector](#) [ShowGridDialog](#) [MotionDetector](#) [UseThisReferenceSample](#) [OnBacktimedFramesCountReached](#) [OnMotionDetected](#) [OnMotionNotDetected](#) [RecordingOnMotion](#) [Enabled](#) [RecordingOnMotion](#) [MotionThreshold](#) [RecordingOnMotion](#) [NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: [Produce Kindle eBooks easily](#)

MouseWheelEventEnabled

TVideoGrabber.MouseWheelEventEnabled

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Activates the OnMouseWheel event

Declaration

property MouseWheelEventEnabled: Boolean **read** GetMouseWheelEventEnabled **write** SetMouseWheelEventEnabled **default** DEF_MouseWheelEventEnabled;

__property bool MouseWheelEventEnabled==GetMouseWheelEventEnabled,
write=SetMouseWheelEventEnabled

Property MouseWheelEventEnabled as Boolean

Description

Enable it to activate the OnMouseWheel event when the user turns the mouse wheel up or down over the video window.

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#) [SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#) [SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#) [SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#) [SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#) [TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#) [TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#) [TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#) [TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Elevate Your CHM Help Files with HelpNDoc's Advanced](#)

Customization Options

MpegStreamType

TVideoGrabber.MpegStreamType

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Type of the stream.

Declaration

property MpegStreamType: TMpegStreamType **read** GetMpegStreamType **write** SetMpegStreamType;

__property TMpegStreamType MpegStreamType=read=GetMpegStreamType, write=SetMpegStreamType, **nodefault**

Property MpegStreamType As TMpegStreamType

Description

[TMpegStreamType](#) type of the stream when the video capture device outputs an MPEG stream (when IsMPEGStream returns true).

This property is meaningful only when the video capture device is a MPEG device IsMPEGStream IsMPEGDevice returns true.

Type of the Mpeg stream. The current [video capture device](#) must be [previewing](#) or [capturing](#) .

Created with the Standard Edition of HelpNDoc: [Maximize Your PDF Protection with These Simple Steps](#)

MultiplexedInputEmulation

TVideoGrabber.MultiplexedInputEmulation

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Automatically switches inputs periodically.

Declaration

property MultiplexedInputEmulation: Boolean **read** GetMultiplexedInputEmulation **write** SetMultiplexedInputEmulation;

__property **bool** MultiplexedInputEmulation =read=GetMultiplexedInputEmulation, write=SetMultiplexedInputEmulation, **nodefault**

Property MultiplexedInputEmulation As Boolean

Description

Used to switch inputs periodically in multiplexed mode.

If the video capture board does not switch the inputs automatically, enable this property to get the inputs switched programmatically in the following frame order:

1...2...3...4...1...2...3...4...1...2...3...4... and so on.

See the [Video capture devices having multiplexed inputs](#) chapter for more information.

See Also

[TMultiplexedRole](#) [AssociateMultiplexedSlave](#) [EnableMultiplexedInput](#) [MultiplexedRole](#) [MultiplexedStabilizationDelay](#) [MultiplexedSwitchDelay](#) [UniqueID](#)

Created with the Standard Edition of HelpNDoc: [Full-featured EPub generator](#)

MultiplexedRole

TVideoGrabber.MultiplexedRole

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the component role with multiplexed video inputs.

Declaration

property MultiplexedRole: TMultiplexedRole **read** GetMultiplexedRole **write** SetMultiplexedRole **default** DEF_MultiplexedRole;

__property TMultiplexedRole MultiplexedRole=read=GetMultiplexedRole, write=SetMultiplexedRole, **default**=0

Property MultiplexedRole As TMultiplexedRole

Description

This [TMultiplexedRole](#) property is used to specify the role of the component with BT878-based video cards having multiplexed inputs.

mr_NotMultiplexed : normal mode

mr_MultiplexedMosaic4 : displays the 4 inputs in a "mosaic layout", the video size is 2 times the normal size

mr_MultiplexedMosaic16 : displays the 16 inputs in a "mosaic layout", the video size is 4 times the normal size

mr_MultiplexedMaster : sets the TVideoGrabber as master (it will distribute the inputs to the slaves)

mr_MultiplexedSlave : sets the TVideoGrabber component as slave (it will receive a virtual input from the master).

See the [multiplexed video inputs](#) chapter.

See Also

[TMultiplexedRole](#) [AssociateMultiplexedSlave](#) [EnableMultiplexedInput](#) [MultiplexedInputEmulation](#) [MultiplexedStabilizationDelay](#) [MultiplexedSwitchDelay](#) [UniqueID](#)

Created with the Standard Edition of HelpNDoc: [Free EPub producer](#)

MultiplexedStabilizationDelay

TVideoGrabber.MultiplexedStabilizationDelay

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the delay to wait for the image stabilization.

Declaration

property MultiplexedStabilizationDelay: LongInt **read** GetMultiplexedStabilizationDelay **write** SetMultiplexedStabilizationDelay **default** DEF_MultiplexedStabilizationDelay;

__property **int** MultiplexedStabilizationDelay =**read**=GetMultiplexedStabilizationDelay, **write**=SetMultiplexedStabilizationDelay

Property MultiplexedStabilizationDelay As Long

Description

Used to set/retrieve the delay to wait for the image stabilization, expressed in milliseconds.

Depending of the video capture board, a various time is required for the input to be properly switched and the image stabilized.

If you get unstable video display cross-channel mixing, try to increase this value (default value = 70

milliseconds)

If too low value may cause jumping frames or cross-input mixed frames.

See Also

[TMultiplexedRole](#) [AssociateMultiplexedSlave](#) [EnableMultiplexedInput](#) [MultiplexedInputEmulation](#)
[MultiplexedRole](#) [MultiplexedSwitchDelay](#) [UniqueID](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly optimize your documentation website for search engines](#)

MultiplexedSwitchDelay

TVideoGrabber.MultiplexedSwitchDelay

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the time each input is active, before switching to the next input.

Declaration

property MultiplexedSwitchDelay: LongInt **read** GetMultiplexedSwitchDelay **write** SetMultiplexedSwitchDelay **default** DEF_MultiplexedSwitchDelay;

__property **int** MultiplexedSwitchDelay = **read**=GetMultiplexedSwitchDelay, **write**=SetMultiplexedSwitchDelay

Property MultiplexedSwitchDelay As Long

Description

Used to specify/retrieve the time each input is active, before switching to the next input, expressed in number of frames.

Default value: 0 (means the input is switched after each video frame).

See Also

[TMultiplexedRole](#) [AssociateMultiplexedSlave](#) [EnableMultiplexedInput](#) [MultiplexedInputEmulation](#)
[MultiplexedRole](#) [MultiplexedStabilizationDelay](#) [UniqueID](#)

Created with the Standard Edition of HelpNDoc: [Generate EPub eBooks with ease](#)

MuteAudioRendering

TVideoGrabber.MuteAudioRendering

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Mutes the audio speakers.

Declaration

property MuteAudioRendering: Boolean **read** GetMuteAudioRendering **write** SetMuteAudioRendering **default** DEF_MuteAudioRendering;

__property **bool** MuteAudioRendering = **read**=GetMuteAudioRendering, **write**=SetMuteAudioRendering, **default**=0

Property MuteAudioRendering As Boolean

Description

Used to mute the audio speakers.

See Also

[TVUMeter](#) [TVUMeterSetting](#) [AudioBalance](#) [AudioChannelRenderMode](#) [AudioVolume](#) [SpeakerBalance](#)

[SpeakerControl](#) [SpeakerVolume](#) [VUMeter](#)

Created with the Standard Edition of HelpNDoc: [What is a Help Authoring tool?](#)

NDIFormatType

TVideoGrabber.NDIFormatType

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Type of NDI streaming

Declaration

property NDIFormatType: TNDIFormatType **read** GetNDIFormatType **write** SetNDIFormatType **default** DEF_NDIFormatType;

__**property** TNDIFormatType NDIFormatType==GetNDIFormatType, write=SetNDIFormatType, **default**=1;

Description

Type of the NDI streaming format

See Also

[TNDIFormatType](#) [NDIName](#) [NetworkStreaming](#) [SetDatasteadFilterDllName](#)

Created with the Standard Edition of HelpNDoc: [Transform Your Documentation Workflow with HelpNDoc's Intuitive UI](#)

NDIName

TVideoGrabber.NDIName

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Name of the NDI streaming

Declaration

property NDIName: **string** **read** GetNDIName **write** SetNDIName;

__**property** System::UnicodeString NDIName==GetNDIName, write=SetNDIName;

Description

Name of the NDI streaming

See [Streaming using Newtek NDI](#)

See Also

[TNDIFormatType](#) [NDIFormatType](#) [NetworkStreaming](#) [SetDatasteadFilterDllName](#)

Created with the Standard Edition of HelpNDoc: [Leave the tedious WinHelp HLP to CHM conversion process behind with HelpNDoc](#)

NetworkStreaming

TVideoGrabber.NetworkStreaming

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables the network streaming.

Declaration

property NetworkStreaming: TNetworkStreaming **read** GetNetworkStreaming **write** SetNetworkStreaming **default** DEF_NetworkStreaming;

```
__property TNetworkStreaming NetworkStreaming=read=GetNetworkStreaming,
write=SetNetworkStreaming, default=0
```

Property NetworkStreaming As TxNetworkStreaming

Description

This [TNetworkStreaming](#) property is used to enable the live network streaming when the [recording](#) is running.

Allowed values:

ns_Disabled network streaming is disabled

ns_ASFDirectNetworkStreaming direct network streaming from the platform running TVideoGrabber

ns_ASFStreamingToPublishingPoint TVideoGrabber sends the streaming media to the Windows Media Server specified by [ASFMediaServerPublishingPoint](#).

ns_NDI Streaming to NDI (requires the NDI Runtime 6 or newer to be installed)

See Also

[TNDIFormatType](#) [TNetworkStreamingType](#) [TOnClientConnection](#) [TNetworkStreaming](#)
[TOnDirectNetworkStreamingHostUrl](#) [ASFAudioBitRate](#) [ASFAudioChannels](#) [ASFDeinterlaceMode](#)
[ASFFixedFrameRate](#) [ASFMediaServerPublishingPoint](#)
[ASFMediaServerRemovePublishingPointAfterDisconnect](#) [ASFMediaServerTemplatePublishingPoint](#)
[ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#) [ASFProfileFromCustomFile](#) [ASFProfiles](#)
[ASFProfilesCount](#) [ASFProfileVersion](#) [ASFVideoBitRate](#) [ASFVideoFrameRate](#) [ASFVideoHeight](#)
[ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#) [NDIFormatType](#) [NDIName](#)
[NetworkStreamingType](#) [OnAuthenticationNeeded](#) [OnClientConnection](#) [OnDirectNetworkStreamingHostUrl](#)
[SetDatasteadFilterDllName](#) [ShowDialog](#) [StartAudioRendering](#) [StreamingURL](#)

Created with the Standard Edition of HelpNDoc: Effortlessly Create High-Quality Help Documentation
with a Help Authoring Tool

NetworkStreamingType

TVideoGrabber.NetworkStreamingType

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the type of streaming (audio, video or both)

Declaration

property NetworkStreamingType: TNetworkStreamingType **read** GetNetworkStreamingType **write** SetNetworkStreamingType **default** DEF_NetworkStreamingType;

```
__property TNetworkStreamingType NetworkStreamingType=read=GetNetworkStreamingType,
write=SetNetworkStreamingType, default=0
```

Property NetworkStreamingType As TxNetworkStreamingType

Description

Used to specify the type of streaming when [NetworkStreaming](#) <> ns_Disabled.

Values:

nst_AudioVideoStreaming : audio + video streaming

nst_VideoStreaming : video streaming only

nst_AudioStreaming : audio streaming only

See Also

[TNetworkStreamingType](#) [TOnClientConnection](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#)
[ASFAudioBitRate](#) [ASFAudioChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#)
[ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#)
[ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#)

[ASFProfileFromCustomFile](#) [ASFProfiles](#) [ASFProfilesCount](#) [ASFProfileVersion](#) [ASFVideoBitRate](#) [ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#) [NetworkStreaming](#) [OnAuthenticationNeeded](#) [OnClientConnection](#) [OnDirectNetworkStreamingHostUrl](#) [ShowDialog](#) [StartAudioRendering](#) [StreamingURL](#)

Created with the Standard Edition of HelpNDoc: [Free PDF documentation generator](#)

NormalCursor

TVideoGrabber.NormalCursor

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Cursor displayed when a TVideoGrabber's task ends.

Declaration

property NormalCursor: TCursor **read** GetNormalCursor **write** SetNormalCursor **default** DEF_NormalCursor;

__property Controls::TCursor NormalCursor=read=GetNormalCursor, write=SetNormalCursor, **default**=0

Property NormalCursor As TxCursors

Description

Cursor displayed when a TVideoGrabber's task ends when [BusyCursor](#) <> crDefault.
If [BusyCursor](#) = crDefault, NormalCursor is ignored.

Created with the Standard Edition of HelpNDoc: [Experience the Power and Ease of Use of a Help Authoring Tool](#)

NotificationMethod

TVideoGrabber.NotificationMethod

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies if the events are notified by timer or by thread.

Declaration

property NotificationMethod: TNotificationMethod **read** GetNotificationMethod **write** SetNotificationMethod **default** DEF_NotificationMethod;

__property TNotificationMethod NotificationMethod=read=GetNotificationMethod, write=SetNotificationMethod, **default**=0

Property NotificationMethod As TxNotificationMethod

Description

DEPRECATED

This [TNotificationMethod](#) property is used to specify if the events must be notified by timer or by thread.

The default method is by timer (nm_Timer), however depending of your application environment, you can prefer using a notification by thread (nm_Thread).

You can switch the method on the fly, event during preview, recording or playback.

Created with the Standard Edition of HelpNDoc: [Full-featured multi-format Help generator](#)

NotificationPriority

TVideoGrabber.NotificationPriority

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the thread priority when the events are notified by thread.

Declaration

property NotificationPriority: TThreadPriority **read** GetNotificationPriority **write** SetNotificationPriority
default DEF_NotificationPriority;

__property Classes::TThreadPriority NotificationPriority=read=GetNotificationPriority,
write=SetNotificationPriority, **default**=3

Property NotificationPriority As TThreadPriority

Description

Used to Specify the thread priority when the events are notified by thread (when [NotificationMethod](#) = nm_Thread).

Created with the Standard Edition of HelpNDoc: [Easily create PDF Help documents](#)

NotificationSleepTime

TVideoGrabber.NotificationSleepTime

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the sleep time between notifications.

Declaration

property NotificationSleepTime: LongInt **read** GetNotificationSleepTime **write** SetNotificationSleepTime
default DEF_NotificationSleepTime;

__property **int** NotificationSleepTime=read=GetNotificationSleepTime, write=SetNotificationSleepTime,
default-- 1

Property NotificationSleepTime As Long

Description

Used to specify the sleep time between event notifications (in milliseconds).
-1 = default value.

E.g. you can assign e.g. 200 to limit the notifications to 5 per second. In this case the [OnFrameProgress](#) event will occur only 5 times per second, even if the frame rate is 30 fps.

Created with the Standard Edition of HelpNDoc: [Upgrade Your Documentation Process with a Help Authoring Tool](#)

OnFrameBitmapEventSynchronone

TVideoGrabber.OnFrameBitmapEventSynchronone

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Make the OnFrameBitmap event callback synchronone

Declaration

property OnFrameBitmapEventSynchronone: Boolean **read** GetOnFrameBitmapEventSynchronone **write**

SetOnFrameBitmapEventSynchronone **default** DEF_OnFrameBitmapEventSynchronone;

__**property bool** OnFrameBitmapEventSynchronone=read=GetOnFrameBitmapEventSynchronone, write=SetOnFrameBitmapEventSynchronone, **default**=0

Property OnFrameBitmapEventSynchronone As Boolean

Description

When enabled, synchronize the OnFrameBitmap event callback with the main thread, allowing to update Windows controls or perform blocking operations from the event.

Warning: enabling this event can slow down the application or introduce latency in the video stream.

See Also

[TCardinalDirection](#) [TFrameGrabberRGBFormat](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [FrameGrabber](#) [FrameGrabberCurrentRGBFormat](#) [FrameGrabberRGBFormat](#) [FramerateDivider](#) [GetFrameInfo](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#) [InFrameProgressEvent](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#) [SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#) [SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#) [SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#) [SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#) [TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#) [TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#) [TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#) [TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#) [TranslateMouseCoordinates](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: Say Goodbye to Documentation Headaches with a Help Authoring Tool

OpenURLAsync

TVideoGrabber.OpenURLAsync

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

enables / disables the asynchronous URL connection

Declaration

property OpenURLAsync: Boolean **read** GetOpenURLAsync **write** SetOpenURLAsync **default** DEF_OpenURLAsync;

__**property bool** OpenURLAsync=read=GetOpenURLAsync, write=SetOpenURLAsync, **default**=0

Property OpenURLAsync As Boolean

Description

Specifies if the component must connect to IP cameras or URLs asynchronously or synchronously.

See [Asynchronous vs Asynchronous connection](#)

See Also

[IR Cut Filter of Axis cameras](#) [TDiscoveryCallbackStatus](#) [TOnONVIFDiscoveryCompletedNotification](#) [Video stream of IP camera with audio of PC microphone or other audio capture device](#) [ExtraDLLPath](#) [GetLastErrorMessage](#) [IPCameraURL](#) [IsURLResponding](#) [IsURLVideoStreamAvailable](#) [OnONVIFDiscoveryCompletedNotification](#) [ONVIF_GetStr](#) [ONVIF_SetStr](#) [ONVIFCancelDiscovery](#) [ONVIFDiscoverCameras](#) [IPRange](#) [ONVIFDiscoverCameras](#) [Multicast](#) [ONVIFEnumCamerasDiscovered](#) [OpenURLAsyncStatus](#) [SendIPCameraCommand](#) [SetDatsteadFilterDllName](#) [SetIPCameraSetting](#)

Created with the Standard Edition of HelpNDoc: Write Epub books for the iPad

OverlayAfterTransform

TVideoGrabber.OverlayAfterTransform

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies if the overlays are performed before or after the transforms

Declaration

property OverlayAfterTransform: Boolean **read** GetOverlayAfterTransform **write** SetOverlayAfterTransform **default** DEF_OverlayAfterTransform;

__property bool OverlayAfterTransform=read=GetOverlayAfterTransform, write=SetOverlayAfterTransform, default=0

Property OverlayAfterTransform As Boolean

Description

Used to specify if the graphics and text overlays are performed before or after the transforms (like video rotation, cropping, etc...).

Default value : false (overlays before transforms).

When this property is enabled and [VideoProcessing_Rotation](#) is used, any drawing performed from the [OnFrameOverlayUsingDC](#) event will be rotated.

To retrieve the initial coordinates and prevent the drawing to be rotated, use the [RetrievalInitialXYAfterRotation](#).

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEODHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#) [SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#) [SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#)

[SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#) [SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#) [TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#) [TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#) [TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#) [TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Protect Your Confidential PDFs with These Simple Security Measures](#)

ParentWindow

TVideoGrabber.ParentWindow

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the window handle that underlies the parent control.

Declaration

property ParentWindow: HWND **read** GetParentWindow **write** SetParentWindow;

__property HWND ParentWindow=read=GetParentWindow, write=SetParentWindow, **nodefault**

n/a

Description

ParentWindow refers to the window handle that underlies the parent control. To designate a non-VCL control as a parent, assign that control's handle to ParentWindow.

Setting ParentWindow has no effect if the **Parent** property is not nil.

Created with the Standard Edition of HelpNDoc: [Full-featured EBook editor](#)

PlayerAudioCodec

TVideoGrabber.PlayerAudioCodec

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Audio codec currently used by the player.

Declaration

property PlayerAudioCodec: **String** **read** GetPlayerAudioCodec;

__property wchar_t *PlayerAudioCodec=read=GetPlayerAudioCodec

Property PlayerAudioCodec As String

Description

Returns the name of the audio codec currently used by the player, if any.
The clip must have been opened by [OpenPlayer](#).

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#)

[PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#)
[PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#)
[PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchronone](#) [PlayerVideoCodec](#)
[RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#)
[SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#)
[VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: Elevate Your CHM Help Files with HelpNDoc's Advanced Customization Options

PlayerAudioRendering

TVideoGrabber.PlayerAudioRendering

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Disables the audio rendering during playback, and allows to use wider speed ratio ranges.

Declaration

property PlayerAudioRendering: Boolean **read** GetPlayerAudioRendering **write** SetPlayerAudioRendering
default DEF_PlayerAudioRendering;

__property **bool** PlayerAudioRendering=read=GetPlayerAudioRendering, write=SetPlayerAudioRendering,
default=1

Property PlayerAudioRendering As Boolean

Description

This property allows to enable/disable the audio rendering during playback of video clips.

When audio is disabled, it is possible to use wider speed ratio ranges. See [PlayerSpeedRatio](#) .

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#)
[AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#)
[FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#)
[MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#)
[OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#)
[PausePlayer](#) [PlayerAudioCodec](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#)
[PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#)
[PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#)
[PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchronone](#) [PlayerVideoCodec](#)
[RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#)
[SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#)
[VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: Make Documentation a Breeze with a Help Authoring Tool

PlayerDuration

TVideoGrabber.PlayerDuration

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Retrieves the total duration of an opened video clip.

Declaration

property PlayerDuration: LargeInteger **read** GetPlayerDuration **write** SetPlayerDuration;

__property __int64 PlayerDuration=read=GetPlayerDuration write=SetPlayerDuration

Property PlayerDuration As Double

Description

Used to retrieve the total duration of a video clip opened by [OpenPlayer](#) (expressed in 100-nanosecond units, divide it by 10000000 to convert in seconds).

This property is theoretically a read-only property, however you can assign a value to this property to specify the display time when opening static images in the player (useful mainly when opening static images in the [Playlist](#)).

E.g. if the playlist must display each static image during 4 seconds, set PlayerDuration = 40000000 before invoking [Playlist](#) (pl_Play, "").

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchronise](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Free help authoring tool](#)

PlayerDVSize

TVideoGrabber.PlayerDVSize

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the video size when playing back DV-encoded video clips.

Declaration

property PlayerDVSize: TDVSize **read** GetPlayerDVSize **write** SetPlayerDVSize **default** DEF_PlayerDVSize;

__property TDVSize PlayerDVSize=read=GetPlayerDVSize, write=SetPlayerDVSize, **default**=0

Property PlayerDVSize As TxDVSize

Description

Used to specify the video size when playing back DV-encoded video clips.
The value is a [TDVSize](#) type.

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#)

[PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchronone](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: Streamline Your Documentation Process with HelpNDoc's Intuitive Interface

PlayerFastSeekSpeedRatio

TVideoGrabber.PlayerFastSeekSpeedRatio

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Speed ratio when playing a video clip at different speeds, forwards or backwards.

Declaration

property PlayerFastSeekSpeedRatio: LongInt **read** GetPlayerFastSeekSpeedRatio **write** SetPlayerFastSeekSpeedRatio **default** DEF_PlayerFastSeekSpeedRatio;

__property **int** PlayerFastSeekSpeedRatio=read=GetPlayerFastSeekSpeedRatio, write=SetPlayerFastSeekSpeedRatio, **default**=4

Property PlayerFastSeekSpeedRatio As Long

Description

Set or retrieves the speed ratio used to play a video clip at different speeds forwards or backwards using [RewindPlayer](#) or [FastForwardPlayer](#) .

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchronone](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: Eliminate the Struggles of Documentation with a Help Authoring Tool

PlayerFileName

TVideoGrabber.PlayerFileName

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Sets or retrieves the full qualified path and file name of a video clip.

Declaration

property PlayerFileName: **string** **read** GetPlayerFileName **write** SetPlayerFileName;

__property wchar_t *PlayerFileName=read=GetPlayerFileName, write=SetPlayerFileName

Property PlayerFileName As String

Description

Used to set or retrieve the full qualified path and file name of a video clip that will be opened then by [OpenPlayer](#) .

E.g.:

```
VideoGrabber.PlayerFileName = "myvideoclip.avi"
VideoGrabber.AutoStartPlayer = false
VideoGrabber.OpenPlayer()
```

Remarks:

- to open a DVD at a specified location, set location of the the VIDEO_TS folder, e.g.:

```
VideoGrabber.PlayerFileName = "E:
VideoGrabber.OpenPlayer()
```

- to open the default DVD, set an empty string and invoke OpenDVD:

```
VideoGrabber.PlayerFileName = ""
VideoGrabber.OpenDVD()
```

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [DVDInfo](#) [DVDTitle](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenDVD](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchron](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: Leave the tedious WinHelp HLP to CHM conversion process behind with HelpNDoc

PlayerForcedCodec

TVideoGrabber.PlayerForcedCodec

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Force the use of a given codec during playback.

Declaration

property PlayerForcedCodec: **string read** GetPlayerForcedCodec **write** SetPlayerForcedCodec;

__property wchar_t *PlayerForcedCodec=read=GetPlayerForcedCodec, write=SetPlayerForcedCodec

Property PlayerForcedCodec As String

Description

Used to specify particular codecs to be used by TVideoGrabber.

TVideoGrabber automatically selects the default appropriate codec to play a given clip.

However in some cases a different codec can be preferred. E.g. to play MPEG2 clips you can prefer the

Elecard/Moonlight video decoder instead of the Intervideo video decoder.

Note that adding codecs to the graph can be achieved also by invoking [ThirdPartyFilter_AddToList](#) ([tpf_AddToGraph](#)...

To use the default codecs, leave **PlayerForcedCodec** empty.

To force the use of a given codec, simply assign to this property:

- either a significant part of the codec name,
- either the CLSID string of the codec.

E.g.:

```
VideoGrabber1.PlayerForcedCodec = "Elecard MPEG2 Video Decoder"
```

or

```
VideoGrabber1.PlayerForcedCodec = "F50B3F13-19C4-11CF-AA9A-02608C9BABA2"
```

PlayerForcedCodec keywords

The following keywords are supported by the PlayerForcedCodec property:

PlayerForcedCodec = "**NOFFDSHOW**"

Prevents TVideoGrabber to use the FFDSHOW codecs when they are enabled by default on the current platform.

PlayerForcedCodec = "**NOMPC**"

Prevents TVideoGrabber to use the MPC (Media Player Classic) codecs when they are enabled by default on the current platform.

PlayerForcedCodec = "**AVS**"

enables the playback through AVISynth when [AVISynth](#) is installed

PlayerForcedCodec = "**FFDSHOW**"

enables the playback through FFDSHOW is the FFDSHOW video decoder is configured to decode the format to be played.

All these non-default settings can be restored to their default value by setting PlayerForcedCodec = ""

Note that reading the PlayerForcedCodec string does not reflect the state of these keywords, only codec names or codec CLSIDs appear in the PlayerForcedCodec string.

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchronise](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Transform Your CHM Help File Creation Process with HelpNDoc](#)

PlayerFrameCount

TVideoGrabber.PlayerFrameCount

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Retrieves the total number of video frames of an opened video clip.

Declaration

property PlayerFrameCount: LargeInteger **read** GetPlayerFrameCount;

__property __int64 PlayerFrameCount=read=GetPlayerFrameCount

Property PlayerFrameCount As Double

Description

Used to retrieve the total number of video frames of a clip opened by [OpenPlayer](#) .

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchronone](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Easily create Help documents](#)

PlayerFramePosition

TVideoGrabber.PlayerFramePosition

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Retrieves the current frame position of an opened video clip.

Declaration

property PlayerFramePosition: LargeInteger **read** GetPlayerFramePosition **write** SetPlayerFramePosition;

__property __int64 PlayerFramePosition=read=GetPlayerFramePosition, write=SetPlayerFramePosition

Property PlayerFramePosition As Double

Description

Used to retrieve the current frame position of a clip opened by [OpenPlayer](#) .

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchronone](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Don't be left in the past: convert your WinHelp HLP help](#)

PlayerFrameRate

TVideoGrabber.PlayerFrameRate

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Retrieves the frame rate of a video clip.

Declaration

property PlayerFrameRate: Double **read** GetPlayerFrameRate;

__property double PlayerFrameRate=read=GetPlayerFrameRate

Property PlayerFrameRate As Double

Description

Used to retrieves the frame rate (expressed in frames per second) of a video clip opened with [OpenPlayer](#) .

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip](#) [Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchron](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Achieve Professional Documentation Results with a Help Authoring Tool](#)

PlayerHwAccel

TVideoGrabber.PlayerHwAccel

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Hardware video acceleration

Declaration

property PlayerHwAccel: THwAccel **read** GetPlayerHwAccel **write** SetPlayerHwAccel **default** DEF_PlayerHwAccel;

__property THwAccel PlayerHwAccel=read=GetPlayerHwAccel, write=SetPlayerHwAccel, **default**=hw_None

Property PlayerHwAccel As THwAccel

Description

Enables the hardware acceleration of video decoding during the playback
This feature is supported through the [LAV filters](#) that must be installed first.
The possible values are enumerated here: [THwAccel](#)

PlayerOpenProgressPercent

TVideoGrabber.PlayerOpenProgressPercent

[Prev](#)[Next](#)[TVideoGrabber](#) [Properties](#)

Percentage of the video being opened in the background

Declaration

property PlayerOpenProgressPercent: LongInt **read** GetPlayerOpenProgressPercent: LongInt;

__property int PlayerOpenProgressPercent==GetPlayerOpenProgressPercent, **nodefault**};

property PlayerOpenProgressPercent as Long

Description

Used to retrieve the percentage of the video being opened in the background

Created with the Standard Edition of HelpNDoc: [Quickly and Easily Convert Your Word Document to an ePub or Kindle eBook](#)

PlayerRefreshPausedDisplay

TVideoGrabber.PlayerRefreshPausedDisplay

[Prev](#)[Next](#)[TVideoGrabber](#) [Properties](#)

Specifies if the video window must be refreshed when the video clip is paused.

Declaration

property PlayerRefreshPausedDisplay: THwAccel **read** GetPlayerRefreshPausedDisplay **write** SetPlayerRefreshPausedDisplay **default** DEF_PlayerRefreshPausedDisplay;

__property bool PlayerRefreshPausedDisplay=read=GetPlayerRefreshPausedDisplay, write=SetPlayerRefreshPausedDisplay, **default**=0

Property PlayerRefreshPausedDisplay As Boolean

Description

If enabled, when the video clip is pause the video window is refreshed at a frame rate specified by [PlayerRefreshPausedDisplayFrameRate](#) property.

Enable this feature to get the graphics and text overlays refreshed when the video clip is paused.

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip](#) [Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchronone](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

PlayerRefreshPausedDisplayFrameRate

TVideoGrabber.PlayerRefreshPausedDisplayFrameRate

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the frame rate used to refresh the display when a video clip is paused.

Declaration

property PlayerRefreshPausedDisplayFrameRate: Double **read** GetPlayerRefreshPausedDisplayFrameRate **write** SetPlayerRefreshPausedDisplayFrameRate;

__property **double**

PlayerRefreshPausedDisplayFrameRate=read=GetPlayerRefreshPausedDisplayFrameRate, write=SetPlayerRefreshPausedDisplayFrameRate

Property PlayerRefreshPausedDisplayFrameRate As Double

Description

Specifies the frame rate used to refresh periodically the display when a video clip is in a paused state and the [PlayerRefreshPausedDisplay](#) property is enabled.

If PlayerRefreshPausedDisplayFrameRate = 0, the display is refreshed at about 30 fps.

E.g.:

PlayerRefreshPausedDisplayFrameRate = 1 refreshes the paused display every second

PlayerRefreshPausedDisplayFrameRate = 5 refreshes the paused display 5 times per second

PlayerRefreshPausedDisplayFrameRate = 0.25 refreshes the paused display every 4 seconds.

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchron](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

PlayerSpeedRatio

TVideoGrabber.PlayerSpeedRatio

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Sets or retrieves the playback speed ratio.

Declaration

property PlayerSpeedRatio: Double **read** GetPlayerSpeedRatio **write** SetPlayerSpeedRatio;

__property **double** PlayerSpeedRatio=read=GetPlayerSpeedRatio, write=SetPlayerSpeedRatio

Property PlayerSpeedRatio As Double

Description

Used to set or retrieve the player's speed ratio when playing a clip by using [OpenPlayer](#) and/or [RunPlayer](#) .

The value is expressed as a ratio of the normal speed, e.g. 0.5 is half speed and 2 is twice speed.

- if the player audio rendering is enabled ([PlayerAudioRendering](#) = true), the speed ratio can be usually used in the 0.5 to 2 range.
- if the player audio rendering is disabled ([PlayerAudioRendering](#) = false), a larger speed ratio range can be used (from 0.1 to 5 or more).

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchronise](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Save time and frustration with HelpNDoc's WinHelp HLP to CHM conversion feature](#)

PlayerSpeedRatioConstantAudioPitch

TVideoGrabber.PlayerSpeedRatioConstantAudioPitch

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Keeps a constant audio pitch for speed ratio <> 1

Declaration

property PlayerSpeedRatioConstantAudioPitch: Double **read** GetPlayerSpeedRatioConstantAudioPitch **write** SetPlayerSpeedRatioConstantAudioPitch;

__property double PlayerSpeedRatioConstantAudioPitch=read=GetPlayerSpeedRatioConstantAudioPitch, write=SetPlayerSpeedRatioConstantAudioPitch

Property PlayerSpeedRatioConstantAudioPitch As Double

Description

When [PlayerSpeedRatio](#) <> 1:

- if PlayerSpeedRatioConstantAudioPitch is enabled (default), a constant audio pitch is maintained (audio samples are dropped)
- if PlayerSpeedRatioConstantAudioPitch is disabled, the audio pitch is increased or decreased according to the player speed.

Created with the Standard Edition of HelpNDoc: [Why Microsoft Word Isn't Cut Out for Documentation: The Benefits of a Help Authoring Tool](#)

PlayerState

TVideoGrabber.PlayerState

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

ber

Returns the current player state.

Declaration

property PlayerState: TPlayerState **read** GetPlayerState;

__property TPlayerState PlayerState==GetPlayerState, **nodefault**};

property PlayerState as TPlayerState

Description

Used to retrieve the current player [state](#).

The possible values are:

ps_Closed
ps_Stopped
ps_Paused
ps_Playing
ps_PlayingBackward
ps_FastForwarding
ps_FastRewinding

Each value is greater than the previous one, so it is possible to test e.g. if the clip is playing in any direction by testing PlayerState >= ps_Playing.

Created with the Standard Edition of HelpNDoc: [Full-featured multi-format Help generator](#)

PlayerTimePosition**TVideoGrabber.PlayerTimePosition**

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Sets or retrieves the current time position of a video clip.

Declaration

property PlayerTimePosition: LargeInteger **read** GetPlayerTimePosition **write** SetPlayerTimePosition;

__property __int64 PlayerTimePosition=read=GetPlayerTimePosition, write=SetPlayerTimePosition

Property PlayerTimePosition As Double

Description

Used to set or retrieve the current time position of a video clip opened with [OpenPlayer](#) (expressed in in 100-nanosecond units, divide it by 10000000 to convert in seconds).

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchronone](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

PlayerTrackBar

TVideoGrabber.PlayerTrackBar

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Associates a trackbar to the player (Delphi and C++Builder versions only).

Declaration

property PlayerTrackBar: TTrackBar **read** GetPlayerTrackBar **write** SetPlayerTrackBar **default** DEF_PlayerTrackBar;

__property ComCtrls::TTrackBar *PlayerTrackBar=read=GetPlayerTrackBar, write=SetPlayerTrackBar, default=0

n/a

Description

Used to associate a trackbar to the player (Delphi and C++Builder versions only).

To use it:

- put a trackbar component on the form,
- select the TVideoGrabber component with the object inspector,
- browse the PlayerTrackBar property and select the trackbar you put on the form.

At runtime, when opening a video clip, the trackbar frequency is set to the number of frames of the video clip (1).

During playback, the trackbar cursor moves automatically, according to the current position of the video clip played.

When the clip is paused, you can set a new position simply by moving the trackbar cursor.

(1) note: the frequency of the trackbar is limited to 32768, see [PlayerTrackBarScale](#) for more information.

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchronone](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

PlayerTrackBarScale

TVideoGrabber.PlayerTrackBarScale

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Scale of the trackbar cursor frequency (Delphi and C++Builder versions only).

Declaration

property PlayerTrackBarScale: Double **read** GetPlayerTrackBarScale;

__property **double** PlayerTrackBarScale =read=GetPlayerTrackBarScale

n/a

Description

Scale of the trackbar cursor frequency, when the number of frames of the video clip is greater than 32768 (Delphi and C++Builder versions only).

When opening a video clip:

- if the number of frames of the video clip is lower or equal to 32768, the frequency of the trackbar is set to the number of frames of the video clip, and PlayerTrackBarScale = 1
- if the number of frames of the video clip is greater than 32768, the frequency of the trackbar is set to 32768, and PlayerTrackBarScale is set to the number of frames / 32768.

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarSynchronone](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Easily create Help documents](#)

PlayerTrackBarSynchronone**TVideoGrabber.PlayerTrackBarSynchronone**

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enable the synchronone behavior of the trackbar

Declaration

property PlayerTrackBarSynchronone: Boolean **read** GetPlayerTrackBarSynchronone **write** SetPlayerTrackBarSynchronone **default** DEF_PlayerTrackBarSynchronone;

__property **bool** PlayerTrackBarSynchronone=read=GetPlayerTrackBarSynchronone, write=SetPlayerTrackBarSynchronone, **default**=0

Property PlayerTrackBarSynchronone as Boolean

Description

When this property is enabled, moving the trackbar with the mouse moves also the position in the video clip immediately.

When this property is disabled, the position is updated only after moving the mouse on the trackbar, when the mouse button is released.

Important:

For this feature to work property you must inform TVideoGrabber of the mouse and keyboard actions on your trackbar by invoking [NotifyPlayerTrackbarAction](#) from your trackbar's event.

Look at the *trackbar's* code of the *MainDemo* project for sample code.

When using Delphi or C++Builder, if a TTrackbar component is assigned to the [PlayerTrackbar](#) property this is handled automatically.

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip](#) [Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Why Microsoft Word Isn't Cut Out for Documentation: The Benefits of a Help Authoring Tool](#)

PlayerVideoCodec

TVideoGrabber.PlayerVideoCodec

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Video codec currently used by the player.

Declaration

property PlayerVideoCodec: **String** **read** GetPlayerVideoCodec;

```
___property wchar_t *PlayerVideoCodec=read=GetPlayerVideoCodec
```

Property PlayerVideoCodec As String

Description

Returns the name of the video codec currently used by the player, if any.
The clip must have been opened by [OpenPlayer](#).

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip](#) [Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchron](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Full-featured Help generator](#)

PlaylistIndex

TVideoGrabber.PlaylistIndex

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Returns the current position in the playlist.

Declaration

function GetPlaylistIndex: LongInt;

int GetPlaylistIndex(**void**);

Function GetPlaylist as Long

Description

Returns the position of the clip currently played in the playlist.

See the "[Using the playlist](#)" chapter for more information about the playlist feature.

See Also

[TPlaylist](#) [Video formats](#) [GetPlaylist](#) [IsPlaylistActive](#) [OnPlayerEndOfPlaylist](#) [Playlist](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly upgrade your WinHelp HLP help files to CHM with HelpNDoc](#)

PreallocCapFileCopiedAfterRecording

TVideoGrabber.PreallocCapFileCopiedAfterRecording

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables recopy of the preallocated file immediately after capture.

Declaration

property PreallocCapFileCopiedAfterRecording: Boolean **read** GetPreallocCapFileCopiedAfterRecording **write** SetPreallocCapFileCopiedAfterRecording **default** DEF_PreallocCapFileCopiedAfterRecording;

__property **bool** PreallocCapFileCopiedAfterRecording=read=GetPreallocCapFileCopiedAfterRecording, write=SetPreallocCapFileCopiedAfterRecording, **default**=1

Property PreallocCapFileCopiedAfterRecording As Boolean

Description

This property is significant only when the preallocated file is used ("[PreallocCapFileEnabled](#)" is true).

- when this property is **enabled**, the recopy of the preallocated file occurs immediately after [StopRecording](#) is invoked. **This is the normal default value.**

- when this property is **disabled**, the preallocated file is not recopied.

This option is useful only if you prefer to keep the video clip "as is" in the preallocated capture file and if you plan to recopy or reencode it later by using [StartReencoding](#).

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last_Recording_FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#)

[SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [Single source CHM, PDF, DOC and HTML Help creation](#)

PreallocCapFileEnabled

TVideoGrabber.PreallocCapFileEnabled

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Used to pre-allocate the recording file.

Declaration

property PreallocCapFileEnabled: Boolean **read** GetPreallocCapFileEnabled **write** SetPreallocCapFileEnabled **default** DEF_PreallocCapFileEnabled;

__property bool PreallocCapFileEnabled=read=GetPreallocCapFileEnabled, write=SetPreallocCapFileEnabled, **default**=0

Property PreallocCapFileEnabled As Boolean

Description

If enabled, a preallocated capture file is created before starting recording.

See the "**preallocated capture file**" section in the [AVI recording](#) chapter for more information.

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last_Recording_FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [Easy to use tool to create HTML Help files and Help web sites](#)

PreallocCapFileName

TVideoGrabber.PreallocCapFileName

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Sets or retrieves the full qualified path and name of the pre-allocated recording file.

Declaration

property PreallocCapFileName: **string** **read** GetPreallocCapFileName **write** SetPreallocCapFileName;

__property wchar_t *PreallocCapFileName=read=GetPreallocCapFileName, write=SetPreallocCapFileName

Property PreallocCapFileName As String

Description

Used to set or retrieve the full qualified path and name of the pre-allocated recording file. This property is ignored if [PreallocCapFileEnabled](#) is disabled. If left blank, a "prealloc.avi" file will be created in the [StoragePath](#) directory.

See the "**preallocated capture file**" section in the [AVI recording](#) chapter for more information.

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last_Recording_FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [Generate Kindle eBooks with ease](#)

PreallocCapFileSizeInMB

TVideoGrabber.PreallocCapFileSizeInMB

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Size of the pre-allocated recording file, in Mb.

Declaration

property PreallocCapFileSizeInMB: LongInt **read** GetPreallocCapFileSizeInMB **write** SetPreallocCapFileSizeInMB **default** DEF_PreallocCapFileSizeInMB;

__property **int** PreallocCapFileSizeInMB=read=GetPreallocCapFileSizeInMB, write=SetPreallocCapFileSizeInMB, **default**=100

Property PreallocCapFileSizeInMB As Long

Description

Used to set or retrieve the size of the pre-allocated recording file, in Mb. For best capture results, capture to a unfragmented, pre-allocated capture file that is larger than the size of the capture data. This property is ignored if [PreallocCapFileEnabled](#) is disabled.

See the "**preallocated capture file**" section in the [AVI recording](#) chapter for more information.

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last_Recording_FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#)

[RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [iPhone web sites made easy](#)

PreviewZoomSize

TVideoGrabber.PreviewZoomSize

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Modifies the size of the preview window (in percent of the [native video width](#) and [height](#)).

Declaration

property PreviewZoomSize: LongInt **read** GetPreviewZoomSize **write** SetPreviewZoomSize **default** DEF_PreviewZoomSize;

__property **int** PreviewZoomSize=read=GetPreviewZoomSize, write=SetPreviewZoomSize, **default**=100

Property PreviewZoomSize As Long

Description

Used to modify the size of the preview window (in percent of the [native video width](#) and [height](#)).
Default value: 100 (percent).

See Also

[TOnFrameProgress](#) [TOnResizeVideo](#) [AutoRefreshPreview](#) [BackgroundColor](#) [Display](#) [AutoSize](#) [Display_FullScreen](#) [Display_SetLocation](#) [Display_VideoPortEnabled](#) [ImageRatio](#) [InFrameProgressEvent](#) [IsVideoInterlaced](#) [IsVideoPortAvailable](#) [OnFrameProgress](#) [OnKeyPress](#) [OnPreviewStarted](#) [OnResizeVideo](#) [PausePreview](#) [ResumePreview](#) [StartPreview](#) [StartSynchronized](#) [StopPreview](#) [Synchronized](#) [UseNearestVideoSize](#) [VideoHeight](#) [VideoRenderer](#) [VideoRendererPriority](#) [VideoWidth](#) [Visible](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Productivity with HelpNDoc's CHM Help File Creation Features](#)

RawAudioSampleCapture

TVideoGrabber.RawAudioSampleCapture

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables/disables the capture of the raw audio samples.

Declaration

property RawAudioSampleCapture: Boolean **read** GetRawAudioSampleCapture **write** SetRawAudioSampleCapture **default** DEF_RawAudioSampleCapture;

__property **bool** RawAudioSampleCapture=read=GetRawAudioSampleCapture, write=SetRawAudioSampleCapture, **default**=0

Property RawAudioSampleCapture As Boolean

Description

Used to enable / disable the capture of each raw audio sample coming out of the audio source.
Each audio sample will be returned by the [OnRawAudioSample](#) event.

See Also

[TOnRawSample](#) [OnRawAudioSample](#) [OnRawVideoSample](#) [RawCaptureAsyncEvent](#) [RawVideoSampleCapture](#)

RawCaptureAsyncEvent

TVideoGrabber.RawCaptureAsyncEvent

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies whether sample captures are returned synchronously or asynchronously.

Declaration

property RawCaptureAsyncEvent: Boolean **read** GetRawCaptureAsyncEvent **write** SetRawCaptureAsyncEvent **default** DEF_RawCaptureAsyncEvent;

__property **bool** RawCaptureAsyncEvent=read=GetRawCaptureAsyncEvent, write=SetRawCaptureAsyncEvent, **default**=1

Property RawCaptureAsyncEvent As Boolean

Description

Used to specify whether sample captures are returned synchronously or asynchronously by the [OnRawVideoSample](#) or [OnRawAudioSample](#) events.

By default, this property is enabled and the samples are returned asynchronously. It means that samples are captured internally when they travel the video stream and returned later by the events.

If you disable this property, the samples are returned synchronously. This means that the video stream waits for the capture events to complete.

In this case you must process the samples as fast as possible and you should not perform any actions with the potential to block, such as holding a critical section or waiting on another thread. Also, do not call any GDI or USER32.DLL APIs that might cause a window to move.

See Also

[TOnRawSample](#) [OnRawAudioSample](#) [OnRawVideoSample](#) [RawAudioSampleCapture](#) [RawVideoSampleCapture](#)

RawSampleCaptureLocation

TVideoGrabber.RawSampleCaptureLocation

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the insertion point of the raw sample capture.

Declaration

property RawSampleCaptureLocation: TRawSampleCaptureLocation **read** GetRawSampleCaptureLocation **write** SetRawSampleCaptureLocation **default** DEF_RawSampleCaptureLocation;

__property TRawSampleCaptureLocation RawSampleCaptureLocation=read=GetRawSampleCaptureLocation, write=SetRawSampleCaptureLocation, **default**=0

Property RawSampleCaptureLocation As TxRawSampleCaptureLocation

Description

This [TRawSampleCaptureLocation](#) property is used to specify the insertion point of the raw sample capture.

nm_rl_SourceFormat the sample capture is inserted on the capture device output.

nm_rl_AfterCompression the sample capture is inserted after the audio or video compressor, when recording and compressing on the fly.

Created with the Standard Edition of HelpNDoc: [Bring your WinHelp HLP help files into the present with HelpNDoc's easy CHM conversion](#)

RawVideoSampleCapture

TVideoGrabber.RawVideoSampleCapture

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables/disables the capture of the raw video samples.

Declaration

property RawVideoSampleCapture: Boolean **read** GetRawVideoSampleCapture **write** SetRawVideoSampleCapture **default** DEF_RawVideoSampleCapture;

__property **bool** RawVideoSampleCapture=read=GetRawVideoSampleCapture, write=SetRawVideoSampleCapture, **default**=0

Property RawVideoSampleCapture As Boolean

Description

Used to enable / disable the capture of each raw video sample coming out of the video source. Each video sample will be returned by the [OnRawVideoSample](#) event.

See Also

[TOnRawSample](#) [OnRawAudioSample](#) [OnRawVideoSample](#) [RawAudioSampleCapture](#) [RawCaptureAsyncEvent](#)

Created with the Standard Edition of HelpNDoc: [Experience the Power and Simplicity of HelpNDoc's User Interface](#)

RecordingAudioBitRate

TVideoGrabber.RecordingAudioBitRate

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the audio bit rate used for the recording

Declaration

RecordingAudioBitRate RecordingTimerInterval: LongInt **read** GetRecordingTimerInterval **write** SetRecordingTimerInterval **default** DEF_RecordingTimerInterval;

__RecordingAudioBitRate **int** RecordingTimerInterval==GetRecordingTimerInterval, write=SetRecordingTimerInterval, **default**=-1;

RecordingAudioBitRate RecordingTimerInterval as Long

Description

Used to specify the audio bit rate used for the recording, expressed in Kb/sec. Default = -1 (no bit rate specified)

This value is used by the Datastead RTSP DirectShow source filter.

Created with the Standard Edition of HelpNDoc: [Maximize Your Productivity with HelpNDoc's Efficient User Interface](#)

RecordingBacktimedFramesCount

TVideoGrabber.RecordingBacktimedFramesCount

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Properties](#)

Used to record the AVI file starting with a back-timed amount of video frames.

Declaration

property RecordingBacktimedFramesCount: LongInt **read** GetRecordingBacktimedFramesCount **write** SetRecordingBacktimedFramesCount;

__property **int** RecordingBacktimedFramesCount=read=GetRecordingBacktimedFramesCount, write=SetRecordingBacktimedFramesCount, **nodefault**

Property RecordingBacktimedFramesCount As Long

Description

Specifies the number of video frames to back-time (preroll) in the buffer.

E.g. if you want to preroll 3 seconds and the frame rate is 15 fps, set RecordingBacktimedFramesCount = 3 x 15 = 45.

See the [Back-timed recording \(preroll\)](#) chapter for more information.

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last_Recording_FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [Easily create CHM Help documents](#)

RecordingCanPause

TVideoGrabber.RecordingCanPause

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Properties](#)

Enables/disables the pause/resume functions during recording.

Declaration

property RecordingCanPause: Boolean **read** GetRecordingCanPause **write** SetRecordingCanPause;

__property **bool** RecordingCanPause=read=GetRecordingCanPause, write=SetRecordingCanPause, **nodefault**

Property RecordingCanPause As Boolean

Description

Used to enable or disable the pause/resume feature during recording.

This property activates the [PauseRecording](#) and [ResumeRecording](#) features during recording.

Disable this property if you get audio / video synchronisation problems.

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last_Recording_FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: Effortlessly Spot and Fix Problems in Your Documentation with HelpNDoc's Project Analyzer

RecordingDuration

TVideoGrabber.RecordingDuration

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Duration of last AVI file captured.

Declaration

property RecordingDuration: Double **read** GetRecordingDuration;

___property **double** RecordingDuration=read=GetRecordingDuration

Property RecordingDuration As Double

Description

Retrieves the duration of the last AVI file captured since the application started.

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last_Recording_FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: Easily convert your WinHelp HLP help files to CHM with HelpNDoc's step-by-step guide

RecordingFileName

TVideoGrabber.RecordingFileName

[Prev](#)

[Next](#)

TVideoGrabber **Properties**

File name used to create the recording file.

Declaration

property RecordingFileName: **string** **read** GetRecordingFileName **write** SetRecordingFileName;

__property wchar_t *RecordingFileName=read=GetRecordingFileName, write=SetRecordingFileName

Property RecordingFileName As String

Description

Used to set or retrieve the full qualified path and name of the recording file created by [StartRecording](#) . If the file exists, it will be overwritten when the capture starts.

If **RecordingFileName** is left blank, when invoking [StartRecording](#) or [RecordToNewFileNow](#) a file name is generated automatically according to the [StoragePath](#) , [AutoFileName](#) , [AutoFilePrefix](#) and [CaptureFileExt](#) properties.

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last](#) [Recording_FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [Add an Extra Layer of Security to Your PDFs with Encryption](#)

RecordingFileSizeMaxInMB

TVideoGrabber.RecordingFileSizeMaxInMB

[Prev](#)

[Next](#)

TVideoGrabber **Properties**

Specifies a maximum recording file size

Declaration

property RecordingFileSizeMaxInMB: LongInt **read** GetRecordingFileSizeMaxInMB **write** SetRecordingFileSizeMaxInMB **default** DEF_RecordingFileSizeMaxInMB;

__property Int RecordingFileSizeMaxInMB=read=GetRecordingFileSizeMaxInMB, write=SetRecordingFileSizeMaxInMB, **default**=0 ;

Property RecordingFileSizeMaxInMB As Integer

Description

If set to a non-zero value before invoking StartRecording, during the recording, each time the file size being recorded exceeds the specified size expressed in Mb, a new file is generated on the fly, depending on the auto file name settings.

See Also

[Recording methods and properties](#)
[TAVIMuxConfig](#)
[TASFDeinterlaceMode](#)
[TAutoFileName](#)
[TOnRecordingCompleted](#)
[TOnRecordingReadyToStart](#)
[TRecordingMethod](#)
[TSyncPreview](#)
[AudioRecording](#)
[AudioSyncAdjustment](#)
[AutoFileNameMinDigits](#)
[AVIDurationUpdated](#)
[AVIFormatOpenDML](#)
[AVIHeaderInfo](#)
[AVIInfo](#)
[Encoder](#)
[SetInt](#)
[HoldRecording](#)
[IsRecordingPaused](#)
[Last_Recording_FileName](#)
[OnBacktimedFramesCountReached](#)
[OnCopyPreallocDataCompleted](#)
[OnCopyPreallocDataProgress](#)
[OnCopyPreallocDataStarted](#)
[OnCreatePreallocFileCompleted](#)
[OnCreatePreallocFileStarted](#)
[OnDiskFull](#)
[OnRecordingCompleted](#)
[OnRecordingPaused](#)
[OnRecordingReadyToStart](#)
[OnRecordingStarted](#)
[OnReencodingCompleted](#)
[OnReencodingStarted](#)
[PauseRecording](#)
[PreallocCapFileCopiedAfterRecording](#)
[PreallocCapFileEnabled](#)
[PreallocCapFileName](#)
[PreallocCapFileSizeInMB](#)
[RecordingBacktimedFramesCount](#)
[RecordingCanPause](#)
[RecordingDuration](#)
[RecordingFileName](#)
[RecordingFourCC](#)
[RecordingHeight](#)
[RecordingInNativeFormat](#)
[RecordingKBytesWrittenToDisk](#)
[RecordingMethod](#)
[RecordingSize](#)
[RecordingTimer](#)
[RecordingTimerInterval](#)
[RecordingWidth](#)
[ResumeRecording](#)
[SaveCompressorSettingsToDataString](#)
[SetMultiplexerFilterByName](#)
[StartAudioRecording](#)
[StartRecording](#)
[StartSynchronized](#)
[StopRecording](#)
[StoragePath](#)
[Synchronized](#)
[SyncPreview](#)
[VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: Transform Your Word Document into a Professional eBook with HelpNDoc

RecordingFourCC

TVideoGrabber.RecordingFourCC

[Prev](#)
[Next](#)

[TVideoGrabber](#)
[Properties](#)

Returns the FourCC used to record the video stream

Declaration

property RecordingFourCC: **String** read GetRecordingFourCC;

__property wchar_t *RecordingFourCC=read=GetRecordingFourCC

Property RecordingFourCC As String

Description

Returns the FourCC that corresponds to the video format used to record the video stream.

E.g.:
YUY2
RGB24
dvsd

Note:

When the frame grabber is enabled on the recording stream ([FrameGrabber](#) = fg_BothStreams or fg_CaptureStream), the recorded format will be a RGB... format corresponding to the [FrameGrabberRGBFormat](#) property.

To record in the format specified by the [VideoSubtype](#) property, you can

- disable the frame grabber on place it on the preview stream only (([FrameGrabber](#) = fg_Disabled or fg_PreviewStream),
- force the recording in the native format by enabling the [RecordingInNativeFormat](#) property.

See Also

[Recording methods and properties](#)
[TAVIMuxConfig](#)
[TASFDeinterlaceMode](#)
[TAutoFileName](#)
[TOnRecordingCompleted](#)
[TOnRecordingReadyToStart](#)
[TRecordingMethod](#)
[TSyncPreview](#)
[AudioRecording](#)
[AudioSyncAdjustment](#)
[AutoFileNameMinDigits](#)
[AVIDurationUpdated](#)
[AVIFormatOpenDML](#)
[AVIHeaderInfo](#)
[AVIInfo](#)
[Encoder](#)
[SetInt](#)
[HoldRecording](#)
[IsRecordingPaused](#)
[Last_Recording_FileName](#)
[OnBacktimedFramesCountReached](#)
[OnCopyPreallocDataCompleted](#)
[OnCopyPreallocDataProgress](#)
[OnCopyPreallocDataStarted](#)
[OnCreatePreallocFileCompleted](#)
[OnCreatePreallocFileStarted](#)
[OnDiskFull](#)
[OnRecordingCompleted](#)
[OnRecordingPaused](#)
[OnRecordingReadyToStart](#)
[OnRecordingStarted](#)
[OnReencodingCompleted](#)
[OnReencodingStarted](#)
[PauseRecording](#)
[PreallocCapFileCopiedAfterRecording](#)
[PreallocCapFileEnabled](#)
[PreallocCapFileName](#)
[PreallocCapFileSizeInMB](#)
[RecordingBacktimedFramesCount](#)
[RecordingCanPause](#)
[RecordingDuration](#)
[RecordingFileName](#)
[RecordingFileSizeMaxInMB](#)
[RecordingHeight](#)

[RecordingInNativeFormat](#)
[RecordingKBytesWrittenToDisk](#)
[RecordingMethod](#)
[RecordingSize](#)
[RecordingTimer](#)
[RecordingTimerInterval](#)
[RecordingWidth](#)
[ResumeRecording](#)
[SaveCompressorSettingsToDataString](#)
[SetMultiplexerFilterByName](#)
[StartAudioRecording](#)
[StartRecording](#)
[StartSynchronized](#)
[StopRecording](#)
[StoragePath](#)
[Synchronized](#)
[SyncPreview](#)
[VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [Easily create HTML Help documents](#)

RecordingHeight

TVideoGrabber.RecordingHeight

[Prev](#)

[Next](#)

[TVideoGrabber](#)
[Properties](#)

Returns the height of video frames in the video stream saved to the AVI file during [recording](#) .

Declaration

property RecordingHeight: LongInt **read** GetRecordingHeight;

__property **int** RecordingHeight=read=GetRecordingHeight, **nodefault**

Property RecordingHeight As Long

Description

Used to retrieve the height of video frames in the video stream saved to the AVI file during [recording](#) .

The returned value is usually the same as [VideoHeight](#) , except when capturing DV in native interleaved mode (when [RecordingInNativeFormat](#) enabled). In this case, the captured size is always the full DV resolution (720x576 in PAL and 720x480 in NTSC), whatever the [video size](#) used for preview.

See Also

[Recording methods and properties](#)
[TAVIMuxConfig](#)
[TASFDeinterlaceMode](#)
[TAutoFileName](#)
[TOnRecordingCompleted](#)
[TOnRecordingReadyToStart](#)
[TRecordingMethod](#)
[TSyncPreview](#)
[AudioRecording](#)
[AudioSyncAdjustment](#)
[AutoFileNameMinDigits](#)
[AVIDurationUpdated](#)
[AVIFormatOpenDML](#)
[AVIHeaderInfo](#)
[AVIInfo](#)
[Encoder](#)
[SetInt](#)
[HoldRecording](#)
[IsRecordingPaused](#)
[Last_Recording_FileName](#)
[OnBacktimedFramesCountReached](#)
[OnCopyPreallocDataCompleted](#)
[OnCopyPreallocDataProgress](#)
[OnCopyPreallocDataStarted](#)
[OnCreatePreallocFileCompleted](#)
[OnCreatePreallocFileStarted](#)
[OnDiskFull](#)
[OnRecordingCompleted](#)
[OnRecordingPaused](#)
[OnRecordingReadyToStart](#)
[OnRecordingStarted](#)
[OnReencodingCompleted](#)
[OnReencodingStarted](#)
[PauseRecording](#)
[PreallocCapFileCopiedAfterRecording](#)
[PreallocCapFileEnabled](#)
[PreallocCapFileName](#)
[PreallocCapFileSizeInMB](#)
[RecordingBacktimedFramesCount](#)
[RecordingCanPause](#)
[RecordingDuration](#)
[RecordingFileName](#)
[RecordingFileSizeMaxInMB](#)
[RecordingFourCC](#)
[RecordingInNativeFormat](#)
[RecordingKBytesWrittenToDisk](#)
[RecordingMethod](#)
[RecordingSize](#)
[RecordingTimer](#)
[RecordingTimerInterval](#)
[RecordingWidth](#)
[ResumeRecording](#)
[SaveCompressorSettingsToDataString](#)
[SetMultiplexerFilterByName](#)
[StartAudioRecording](#)
[StartRecording](#)
[StartSynchronized](#)
[StopRecording](#)
[StoragePath](#)
[Synchronized](#)
[SyncPreview](#)
[VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [Make Your PDFs More Secure with Encryption and Password Protection](#)

RecordingInNativeFormat

TVideoGrabber.RecordingInNativeFormat

[Prev](#)

[Next](#)

[TVideoGrabber](#)
[Properties](#)

Specifies that the video stream must be saved as it comes out of the video device (no overlay, no compression).

Declaration

property RecordingInNativeFormat: Boolean **read** GetRecordingInNativeFormat **write** SetRecordingInNativeFormat **default** DEF_RecordingInNativeFormat;

__property **bool** RecordingInNativeFormat=read=GetRecordingInNativeFormat, write=SetRecordingInNativeFormat, **default**=1

Property RecordingInNativeFormat As Boolean

Description

Used to specify if the video stream must be saved "as is" into the AVI file, or if it can be modified (compression, frame overlay, etc...)

enabled (default): the video stream is saved as it comes out of the video capture device, it cannot be modified,

disabled: the video stream is saved after going through the frame grabber (e.g. for text or graphics overlay) and/or a video compressor (for "on the fly" compression).

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last](#) [Recording](#) [FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [Achieve Professional Documentation Results with a Help Authoring Tool](#)

RecordingMethod

TVideoGrabber.RecordingMethod

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the recording method.

Declaration

property RecordingMethod: TRecordingMethod **read** GetRecordingMethod **write** SetRecordingMethod **default** DEF_RecordingMethod;

__property TRecordingMethod RecordingMethod=read=GetRecordingMethod, write=SetRecordingMethod, **default**=0

Property RecordingMethod As TxRecordingMethod

Description

This [TRecordingMethod](#) property is used to specify the recording method:

rm_AVI : records the video and audio streams to an AVI file.

rm_ASF : records the video and audio streams to an ASF file.

rm_SendToDV : sends the output to a DV device, like a DV camcorder. See the [Send to DV](#) chapter.

rm_MKV: similar to rm_AVI, but the recording will create a MKV file (the Matroska muxer must be installed, it can be downloaded at <http://www.matroska.org/>)

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last](#) [Recording](#) [FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [News and information about help authoring tools and software](#)

RecordingOnMotion_Enabled

TVideoGrabber.RecordingOnMotion_Enabled

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Activates the recording only when motion is detected.

Declaration

property RecordingOnMotion_Enabled: Boolean **read** GetRecordingOnMotion_Enabled **write** SetRecordingOnMotion_Enabled **default** DEF_RecordingOnMotion_Enabled;

__property **bool** RecordingOnMotion_Enabled=read=GetRecordingOnMotion_Enabled, write=SetRecordingOnMotion_Enabled, **default**=0

Property RecordingOnMotion_Enabled As Boolean

Description

Used to activate the recording only when motion is detected. Then the recording is paused until motion is detected again.

Must be activated before invoking [StartRecording](#).

See Also

[Color / Greyscale](#) [Grid structure / grid sensitivity](#) [Motion ratio](#) [Recording only when motion is detected](#) [Video noise](#) [TOnMotionDetected](#) [TOnMotionNotDetected](#) [MotionDetector](#) [CellMotionRatio](#) [MotionDetector](#) [CompareBlue](#) [MotionDetector](#) [CompareGreen](#) [MotionDetector](#) [CompareRed](#) [MotionDetector](#) [Enabled](#) [MotionDetector](#) [EnumGridDialogControls](#) [MotionDetector](#) [Get2DTextGrid](#) [MotionDetector](#) [Get2DTextMotion](#) [MotionDetector](#) [GetCellLocation](#) [MotionDetector](#) [GetCellSensitivity](#) [MotionDetector](#) [GetCellSize](#) [MotionDetector](#) [GloballyIncOrDecSensitivity](#) [MotionDetector](#) [GlobalMotionRatio](#) [MotionDetector](#) [GreyScale](#) [MotionDetector](#) [Grid](#) [MotionDetector](#) [GridXCount](#) [MotionDetector](#) [GridYCount](#) [MotionDetector](#) [IsGridValid](#) [MotionDetector](#) [MaxDetectionsPerSecond](#) [MotionDetector](#) [ReduceCPULoad](#) [MotionDetector](#) [ReduceVideoNoise](#) [MotionDetector](#) [Reset](#) [MotionDetector](#) [ResetGlobalSensitivity](#) [MotionDetector](#) [SetCellSensitivity](#) [MotionDetector](#) [SetGridSize](#) [MotionDetector](#) [ShowGridDialog](#) [MotionDetector](#) [Triggered](#) [MotionDetector](#) [UseThisReferenceSample](#) [OnBacktimedFramesCountReached](#) [OnMotionDetected](#) [OnMotionNotDetected](#) [RecordingOnMotion](#) [MotionThreshold](#) [RecordingOnMotion](#) [NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: [Free CHM Help documentation generator](#)

RecordingOnMotion_MotionThreshold

TVideoGrabber.RecordingOnMotion_MotionThreshold

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Motion threshold that activates the recording.

Declaration

property RecordingOnMotion_MotionThreshold: Double **read** GetRecordingOnMotion_MotionThreshold
write SetRecordingOnMotion_MotionThreshold;

__property **double** RecordingOnMotion_MotionThreshold=read=GetRecordingOnMotion_MotionThreshold,
write=SetRecordingOnMotion_MotionThreshold

Property RecordingOnMotion_MotionThreshold As Double

Description

When [RecordingOnMotion_Enabled](#) is activated, this property specifies a minimal motion ratio above which the recording is (re-)activated.

See Also

[Color / Greyscale Grid structure / grid sensitivity Motion ratio Recording only when motion is detected Video noise TOnMotionDetected TOnMotionNotDetected MotionDetector_CellMotionRatio MotionDetector_CompareBlue MotionDetector_CompareGreen MotionDetector_CompareRed MotionDetector_Enabled MotionDetector_EnumGridDialogControls MotionDetector_Get2DTextGrid MotionDetector_Get2DTextMotion MotionDetector_GetCellLocation MotionDetector_GetCellSensitivity MotionDetector_GetCellSize MotionDetector_GloballyIncOrDecSensitivity MotionDetector_GlobalMotionRatio MotionDetector_GreyScale MotionDetector_Grid MotionDetector_GridXCount MotionDetector_GridYCount MotionDetector_IsGridValid MotionDetector_MaxDetectionsPerSecond MotionDetector_ReduceCPULoad MotionDetector_ReduceVideoNoise MotionDetector_Reset MotionDetector_ResetGlobalSensitivity MotionDetector_SetCellSensitivity MotionDetector_SetGridSize MotionDetector_ShowGridDialog MotionDetector_Triggered MotionDetector_UseThisReferenceSample OnBacktimedFramesCountReached OnMotionDetected OnMotionNotDetected RecordingOnMotion_Enabled RecordingOnMotion_NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: [Streamline Your CHM Help File Creation with HelpNDoc](#)

RecordingOnMotion_NoMotionPauseDelayMs

TVideoGrabber.RecordingOnMotion_NoMotionPauseDelayMs

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

"no motion" delay after which the recording is paused.

Declaration

property RecordingOnMotion_NoMotionPauseDelayMs: LongInt **read** GetRecordingOnMotion_NoMotionPauseDelayMs **write** SetRecordingOnMotion_NoMotionPauseDelayMs
default DEF_RecordingOnMotion_NoMotionPauseDelayMs;

__property **int** RecordingOnMotion_NoMotionPauseDelayMs=read=GetRecordingOnMotion_NoMotionPauseDelayMs,
write=SetRecordingOnMotion_NoMotionPauseDelayMs, **default**=2000

Property RecordingOnMotion_NoMotionPauseDelayMs As Long

Description

When [RecordingOnMotion_Enabled](#) is activated and no motion is detected, this property specifies a delay after which the recording is switched back in a paused state, waiting for the next motion to occur. The value is expressed in ms (e.g. 2000 = 2 seconds).

See Also

[Color / Greyscale Grid structure / grid sensitivity Motion ratio Recording only when motion is detected](#)
[Video noise TOnMotionDetected TOnMotionNotDetected MotionDetector_CellMotionRatio](#)
[MotionDetector_CompareBlue MotionDetector_CompareGreen MotionDetector_CompareRed](#)
[MotionDetector_Enabled MotionDetector_EnumGridDialogControls MotionDetector_Get2DTextGrid](#)
[MotionDetector_Get2DTextMotion MotionDetector_GetCellLocation MotionDetector_GetCellSensitivity](#)
[MotionDetector_GetCellSize MotionDetector_GloballyIncOrDecSensitivity](#)
[MotionDetector_GlobalMotionRatio MotionDetector_GreyScale MotionDetector_Grid](#)
[MotionDetector_GridXCount MotionDetector_GridYCount MotionDetector_IsGridValid](#)
[MotionDetector_MaxDetectionsPerSecond MotionDetector_ReduceCPULoad](#)
[MotionDetector_ReduceVideoNoise MotionDetector_Reset MotionDetector_ResetGlobalSensitivity](#)
[MotionDetector_SetCellSensitivity MotionDetector_SetGridSize MotionDetector_ShowGridDialog](#)
[MotionDetector_Triggered MotionDetector_UseThisReferenceSample OnBacktimedFramesCountReached](#)
[OnMotionDetected OnMotionNotDetected RecordingOnMotion_Enabled](#)
[RecordingOnMotion_MotionThreshold](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Productivity with HelpNDoc's Efficient User Interface](#)

RecordingPauseCreatesNewFile

TVideoGrabber.RecordingPauseCreatesNewFile

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Generates a new recording file each time the recording is paused.

Declaration

property RecordingPauseCreatesNewFile: Boolean **read** GetRecordingPauseCreatesNewFile **write** SetRecordingPauseCreatesNewFile **default** DEF_RecordingOnMotion_CreateNewFiles;

__property **bool** RecordingPauseCreatesNewFile=read=GetRecordingPauseCreatesNewFile, write=SetRecordingPauseCreatesNewFile, **default**=0

Property RecordingPauseCreatesNewFile As Boolean

Description

When pausing the recording and this property is enabled, a new recording file name is generated automatically.

The new file is not created when resuming the recording, but rather when the recording is paused by invoking [PauseRecording](#).

By this way, the file is already created and opened when invoking [ResumeRecording](#), so no frames are lost at this time.

If the recording is stopped before invoking [ResumeRecording](#), the new empty file (that has just been created before, when [PauseRecording](#) has been invoked) is deleted.

See Also

[Recording methods and properties TAVIMuxConfig TASFDDeinterlaceMode TAutoFileName](#)
[TOnRecordingCompleted TOnRecordingReadyToStart TRecordingMethod TSyncPreview AudioRecording](#)
[AudioSyncAdjustment AutoFileNameMinDigits AVIDurationUpdated AVIFormatOpenDML AVIHeaderInfo](#)
[AVIInfo Encoder SetInt HoldRecording IsRecordingPaused Last_Recording_FileName](#)
[OnBacktimedFramesCountReached OnCopyPreallocDataCompleted OnCopyPreallocDataProgress](#)
[OnCopyPreallocDataStarted OnCreatePreallocFileCompleted OnCreatePreallocFileStarted OnDiskFull](#)
[OnRecordingCompleted OnRecordingPaused OnRecordingReadyToStart OnRecordingStarted](#)
[OnReencodingCompleted OnReencodingStarted PauseRecording PreallocCapFileCopiedAfterRecording](#)
[PreallocCapFileEnabled PreallocCapFileName PreallocCapFileSizeInMB RecordingBacktimedFramesCount](#)
[RecordingCanPause RecordingDuration RecordingFileName RecordingFileSizeMaxInMB](#)

[RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#)
[RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#)
[ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#)
[StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#)
[VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [Free iPhone documentation generator](#)

RecordingSize

TVideoGrabber.RecordingSize

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Used to record at a smaller size than the preview size

Declaration

property RecordingSize: TRecordingSize **read** GetRecordingSize **write** SetRecordingSize **default** DEF_RecordingSize;

__property TRecordingSize RecordingSize=read=GetRecordingSize, write=SetRecordingSize, **default**=0 ;

Property RecordingSize As TxRecordingSize

Description

Used to record in AVI at a smaller size than the preview size.

By default, the recording is made at the same size than the preview size.

Possible values:

rs_Default : records at the current video size

rs_HalfSize : records at half size

rs_QuarterSize : records at quarter size

This property lets you record smaller at a smaller size, while keeping an higher resolution for frame capture.

E.g. if the current preview size is 640x480 and RecordingSize = HalfSize, the AVI recording will be mad in 320x240.

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#)
[TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#)
[AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#)
[AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last_Recording_FileName](#)
[OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#)
[OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#)
[OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#)
[OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#)
[PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#)
[RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#)
[RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#)
[RecordingMethod](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#)
[SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#)
[StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [Easily create Help documents](#)

RecordingTimer

TVideoGrabber.RecordingTimer

[Prev](#)

[Next](#)

TVideoGrabber Properties

Used to enable the timed recording

Declaration

property RecordingTimer: TRecordingTimer **read** GetRecordingTimer **write** SetRecordingTimer **default** DEF_RecordingTimer;

__property TRecordingTimer RecordingTimer==GetRecordingTimer, write=SetRecordingTimer, **default**=0;

property RecordingTimer as TxRecordingTimer

Description

Used to enable the timed recording:

rt_Disabled : disabled

rt_RecordToNewFile : a new recording file is generated each [RecordingTimerInterval](#) seconds

rt_StopRecording : the recording stops after [RecordingTimerInterval](#) seconds

rt_StartRecording : the recording starts in preview mode only, the recording will begin after [RecordingTimerInterval](#) seconds

rt_FrameCapture : Captures periodically the current frame, the type of capture (memory bitmap, bmp file, etc) is determined by the BurstType property. The interval is determined by RecordingTimerInterval, expressed in seconds

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last_Recording_FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [Free iPhone documentation generator](#)

RecordingTimerInterval

TVideoGrabber.RecordingTimerInterval

[Prev](#)

[Next](#)

TVideoGrabber Properties

Interval of time for RecordingTimer, in seconds

Declaration

property RecordingTimerInterval: LongInt **read** GetRecordingTimerInterval **write** SetRecordingTimerInterval **default** DEF_RecordingTimerInterval;

__property int RecordingTimerInterval==GetRecordingTimerInterval, write=SetRecordingTimerInterval, **default**=0;

property RecordingTimerInterval as Long

Description

Specifies the Interval of time between each [RecordingTimer](#) action (generates new file or stops recording), expressed in seconds.

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last](#) [Recording](#) [FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: Effortlessly Create High-Quality Help Documentation with a Help Authoring Tool

RecordingVideoBitRate

TVideoGrabber.RecordingVideoBitRate

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the video bit rate used for the recording

Declaration

RecordingVideoBitRate RecordingTimerInterval: LongInt **read** GetRecordingTimerInterval **write** SetRecordingTimerInterval **default** DEF_RecordingTimerInterval;

__RecordingVideoBitRate **int** RecordingTimerInterval==GetRecordingTimerInterval, write=SetRecordingTimerInterval, **default**=-1;

RecordingVideoBitRate RecordingTimerInterval as Long

Description

Used to specify the video bit rate used for the recording, expressed in Kb/sec.
Default = -1 (no bit rate specified)

Deprecated

Created with the Standard Edition of HelpNDoc: Full-featured multi-format Help generator

RecordingWidth

TVideoGrabber.RecordingWidth

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Returns the width of video frames in the video stream saved to the AVI file during [recording](#) .

Declaration

property RecordingWidth: LongInt **read** GetRecordingWidth;

__**property** **int** RecordingWidth=read=GetRecordingWidth, **nodefault**

Property RecordingWidth As Long

Description

Used to retrieve the width of video frames in the video stream saved to the AVI file during [recording](#) .

The returned value is usually the same as [VideoWidth](#) , except when capturing DV in native interleaved mode (when [RecordingInNativeFormat](#) is true). In this case, the captured size is always the full DV resolution (720x576 in PAL and 720x480 in NTSC), whatever the [video size](#) used for preview.

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last_Recording_FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [iPhone web sites made easy](#)

Reencoding_IncludeAudioStream

TVideoGrabber.Reencoding_IncludeAudioStream

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies if the audio stream must be included.

Declaration

property Reencoding_IncludeAudioStream: Boolean **read** GetReencodingIncludeAudioStream **write** SetReencodingIncludeAudioStream **default** DEF_Reencoding_IncludeAudioStream;

__property **bool** Reencoding_IncludeAudioStream=read=GetReencodingIncludeAudioStream, write=SetReencodingIncludeAudioStream, **default**=1

Property Reencoding_IncludeAudioStream As Boolean

Description

Used to specify if the audio stream must be included in the reencoded video clip.
If false, the new video clip will contain only the video stream.

See Also

[OnReencodingCompleted](#) [OnReencodingProgress](#) [OnReencodingStarted](#) [Reencoding_IncludeVideoStream](#) [Reencoding_Method](#) [Reencoding_NewVideoClip](#) [Reencoding_SourceVideoClip](#) [Reencoding_StartFrame](#) [Reencoding_StartTime](#) [Reencoding_StopFrame](#) [Reencoding_StopTime](#) [Reencoding_UseAudioCompressor](#) [Reencoding_UseFrameGrabber](#) [Reencoding_UseVideoCompressor](#) [Reencoding_WMVOutput](#) [StartReencoding](#) [StopReencoding](#)

Created with the Standard Edition of HelpNDoc: [Free EPub producer](#)

Reencoding_IncludeVideoStream

TVideoGrabber.Reencoding_IncludeVideoStream

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

ber

Specifies if the video stream must be included in the reencoded video clip.

Declaration

property Reencoding_IncludeVideoStream: Boolean **read** GetReencodingIncludeVideoStream **write** SetReencodingIncludeVideoStream **default** DEF_Reencoding_IncludeVideoStream;

__property bool Reencoding_IncludeVideoStream=read=GetReencodingIncludeVideoStream, write=SetReencodingIncludeVideoStream, **default**=1

Property Reencoding_IncludeVideoStream As Boolean

Description

Used to specify if the video stream must be included in the reencoded video clip. If false, the new video clip will contain only the audio stream.

See Also

[OnReencodingCompleted](#) [OnReencodingProgress](#) [OnReencodingStarted](#) [Reencoding_IncludeAudioStream](#) [Reencoding_Method](#) [Reencoding_NewVideoClip](#) [Reencoding_SourceVideoClip](#) [Reencoding_StartFrame](#) [Reencoding_StartTime](#) [Reencoding_StopFrame](#) [Reencoding_StopTime](#) [Reencoding_UseAudioCompressor](#) [Reencoding_UseFrameGrabber](#) [Reencoding_UseVideoCompressor](#) [Reencoding_WMVOutput](#) [StartReencoding](#) [StopReencoding](#)

Created with the Standard Edition of HelpNDoc: [Don't Let Unauthorized Users View Your PDFs: Learn How to Set Passwords](#)

Reencoding_Method**TVideoGrabber.Reencoding_Method**
[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the reencoding format: AVI or ASF

Declaration

property Reencoding_Method: TRecordingMethod **read** GetReencodingMethod **write** SetReencodingMethod **default** DEF_Reencoding_Method;

__property TRecordingMethod Reencoding_Method=read=GetReencodingMethod, write=SetReencodingMethod, **default**=0

Property Reencoding_Method As TxRecordingMethod

Description

Used to specify the reencoding format:

AVI = rm_AVI

ASF = rm_ASF

(other [TRecordingMethod](#) values are not supported for this property).

See Also

[OnReencodingCompleted](#) [OnReencodingProgress](#) [OnReencodingStarted](#) [Reencoding_IncludeAudioStream](#) [Reencoding_IncludeVideoStream](#) [Reencoding_NewVideoClip](#) [Reencoding_SourceVideoClip](#) [Reencoding_StartFrame](#) [Reencoding_StartTime](#) [Reencoding_StopFrame](#) [Reencoding_StopTime](#) [Reencoding_UseAudioCompressor](#) [Reencoding_UseFrameGrabber](#) [Reencoding_UseVideoCompressor](#) [Reencoding_WMVOutput](#) [StartReencoding](#) [StopReencoding](#)

Created with the Standard Edition of HelpNDoc: [Full-featured Help generator](#)

Reencoding_NewVideoClip

TVideoGrabber.Reencoding_NewVideoClip

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the file name of the new video clip.

Declaration

property Reencoding_NewVideoClip: **string read** GetReencodingNewVideoClip **write** SetReencodingNewVideoClip;

__property wchar_t *Reencoding_NewVideoClip=read=GetReencodingNewVideoClip,
write=SetReencodingNewVideoClip

Property Reencoding_NewVideoClip As String

Description

Used to specify the file name of the video clip that will be created when invoking [StartReencoding](#).

See Also

[OnReencodingCompleted](#) [OnReencodingProgress](#) [OnReencodingStarted](#) [Reencoding_IncludeAudioStream](#) [Reencoding_IncludeVideoStream](#) [Reencoding_Method](#) [Reencoding_SourceVideoClip](#) [Reencoding_StartFrame](#) [Reencoding_StartTime](#) [Reencoding_StopFrame](#) [Reencoding_StopTime](#) [Reencoding_UseAudioCompressor](#) [Reencoding_UseFrameGrabber](#) [Reencoding_UseVideoCompressor](#) [Reencoding_WMVOutput](#) [StartReencoding](#) [StopReencoding](#)

Created with the Standard Edition of HelpNDoc: Powerful and User-Friendly Help Authoring Tool for Markdown Documents

Reencoding_SourceVideoClip

TVideoGrabber.Reencoding_SourceVideoClip

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

File name of the video clip to reencode.

Declaration

property Reencoding_SourceVideoClip: **String read** GetReencodingSourceVideoClip **write** SetReencodingSourceVideoClip;

__property wchar_t *Reencoding_SourceVideoClip=read=GetReencodingSourceVideoClip,
write=SetReencodingSourceVideoClip

Property Reencoding_SourceVideoClip As String

Description

Used to specify the file name of the video clip that will be reencoded when invoking [StartReencoding](#).

See Also

[OnReencodingCompleted](#) [OnReencodingProgress](#) [OnReencodingStarted](#) [Reencoding_IncludeAudioStream](#) [Reencoding_IncludeVideoStream](#) [Reencoding_Method](#) [Reencoding_NewVideoClip](#) [Reencoding_StartFrame](#) [Reencoding_StartTime](#) [Reencoding_StopFrame](#) [Reencoding_StopTime](#) [Reencoding_UseAudioCompressor](#) [Reencoding_UseFrameGrabber](#) [Reencoding_UseVideoCompressor](#) [Reencoding_WMVOutput](#) [StartReencoding](#) [StopReencoding](#)

Created with the Standard Edition of HelpNDoc: Leave the tedious WinHelp HLP to CHM conversion process behind with HelpNDoc

Reencoding_StartFrame

TVideoGrabber.Reencoding_StartFrame

[Prev](#)[Next](#)[TVideoGrabber](#) [Properties](#)

Specifies the start frame if the video clip must be cut.

Declaration

property Reencoding_StartFrame: LargeInteger **read** GetReencodingStartFrame **write** SetReencodingStartFrame **default** -1;

__property __int64 Reencoding_StartFrame=read=GetReencodingStartFrame, write=SetReencodingStartFrame, **default**=-1

Property Reencoding_StartFrame as Double

Description

Used to specify the start frame if the new video clip must contain only a part of the original video clip. Set the value to **-1** to start from the beginning of the video clip. Use [AVIDuration](#) to retrieve the number of frames of the video clip (before invoking [StartReencoding](#)).

See Also

[OnReencodingCompleted](#) [OnReencodingProgress](#) [OnReencodingStarted](#) [Reencoding_IncludeAudioStream](#) [Reencoding_IncludeVideoStream](#) [Reencoding_Method](#) [Reencoding_NewVideoClip](#) [Reencoding_SourceVideoClip](#) [Reencoding_StartFrame](#) [Reencoding_StopFrame](#) [Reencoding_StopTime](#) [Reencoding_UseAudioCompressor](#) [Reencoding_UseFrameGrabber](#) [Reencoding_UseVideoCompressor](#) [Reencoding_WMVOutput](#) [StartReencoding](#) [StopReencoding](#)

Created with the Standard Edition of HelpNDoc: [Create help files for the Qt Help Framework](#)

Reencoding_StartTime

TVideoGrabber.Reencoding_StartTime

[Prev](#)[Next](#)[TVideoGrabber](#) [Properties](#)

Specifies the start time if the video clip must be cut.

Declaration

property Reencoding_StartTime: LargeInteger **read** GetReencodingStartTime **write** SetReencodingStartTime **default** DEF_Reencoding_StartTime;

__property __int64 Reencoding_StartTime=read=GetReencodingStartTime, write=SetReencodingStartTime, **default**=- 1

Property Reencoding_StartTime As Double

Description

Used to specify the start time if the new video clip must contain only a part of the original video clip, expressed in 100ns units (e.g. 4 seconds = 400000000). Set the value to **-1** (or **0**) to start from the beginning of the video clip. Use [AVIDuration](#) to retrieve the duration of the video clip (before invoking [StartReencoding](#)).

See Also

[OnReencodingCompleted](#) [OnReencodingProgress](#) [OnReencodingStarted](#) [Reencoding_IncludeAudioStream](#) [Reencoding_IncludeVideoStream](#) [Reencoding_Method](#) [Reencoding_NewVideoClip](#) [Reencoding_SourceVideoClip](#) [Reencoding_StartFrame](#) [Reencoding_StopFrame](#) [Reencoding_StopTime](#) [Reencoding_UseAudioCompressor](#) [Reencoding_UseFrameGrabber](#) [Reencoding_UseVideoCompressor](#) [Reencoding_WMVOutput](#) [StartReencoding](#) [StopReencoding](#)

Reencoding_StopFrame

TVideoGrabber.Reencoding_StopFrame

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Properties](#)

Specifies the stop frame if the video clip must be cut.

Declaration

property Reencoding_StopFrame: LargeInteger **read** GetReencodingStopFrame **write** SetReencodingStopFrame **default** -1;

__property __int64 Reencoding_StopFrame=read=GetReencodingStopFrame, write=SetReencodingStopFrame, **default**=-1

Property Reencoding_StopFrame as Double

Description

Used to specify the stop frame if the new video clip must contain only a part of the original video clip. Set the value to **-1** to go on until the end of the video clip. Use [AVIDuration](#) to retrieve the frame count of the video clip (before invoking [StartReencoding](#)).

See Also

[OnReencodingCompleted](#) [OnReencodingProgress](#) [OnReencodingStarted](#) [Reencoding_IncludeAudioStream](#) [Reencoding_IncludeVideoStream](#) [Reencoding_Method](#) [Reencoding_NewVideoClip](#) [Reencoding_SourceVideoClip](#) [Reencoding_StartFrame](#) [Reencoding_StartTime](#) [Reencoding_StopTime](#) [Reencoding_UseAudioCompressor](#) [Reencoding_UseFrameGrabber](#) [Reencoding_UseVideoCompressor](#) [Reencoding_WMVOutput](#) [StartReencoding](#) [StopReencoding](#)

Reencoding_StopTime

TVideoGrabber.Reencoding_StopTime

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Properties](#)

Specifies the stop time if the video clip must be cut.

Declaration

property Reencoding_StopTime: LargeInteger **read** GetReencodingStopTime **write** SetReencodingStopTime **default** DEF_Reencoding_StopTime;

__property __int64 Reencoding_StopTime=read=GetReencodingStopTime, write=SetReencodingStopTime, **default**=- 1

Property Reencoding_StopTime As Double

Description

Used to specify the stop time if the new video clip must contain only a part of the original video clip, expressed in 100ns units (e.g. 4 seconds = 400000000). Set the value to **-1** to end at the normal duration of the video clip. Use [AVIDuration](#) to retrieve the duration of the video clip (before invoking [StartReencoding](#)).

See Also

[OnReencodingCompleted](#) [OnReencodingProgress](#) [OnReencodingStarted](#) [Reencoding_IncludeAudioStream](#)

[Reencoding_IncludeVideoStream](#) [Reencoding_Method](#) [Reencoding_NewVideoClip](#)
[Reencoding_SourceVideoClip](#) [Reencoding_StartFrame](#) [Reencoding_StartTime](#) [Reencoding_StopFrame](#)
[Reencoding_UseAudioCompressor](#) [Reencoding_UseFrameGrabber](#) [Reencoding_UseVideoCompressor](#)
[Reencoding_WMVOutput](#) [StartReencoding](#) [StopReencoding](#)

Created with the Standard Edition of HelpNDoc: [Easily Add Encryption and Password Protection to Your PDFs](#)

Reencoding_UseAudioCompressor

TVideoGrabber.Reencoding_UseAudioCompressor

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the current audio compressor to use.

Declaration

property Reencoding_UseAudioCompressor: Boolean **read** GetReencodingUseAudioCompressor **write** SetReencodingUseAudioCompressor **default** DEF_Reencoding_UseAudioCompressor;

__property **bool** Reencoding_UseAudioCompressor=read=GetReencodingUseAudioCompressor, write=SetReencodingUseAudioCompressor, **default**=0

Property Reencoding_UseAudioCompressor As Boolean

Description

Used to specify if the current [audio compressor](#) must to use to reencode the audio stream of the video clip.

See Also

[OnReencodingCompleted](#) [OnReencodingProgress](#) [OnReencodingStarted](#) [Reencoding_IncludeAudioStream](#)
[Reencoding_IncludeVideoStream](#) [Reencoding_Method](#) [Reencoding_NewVideoClip](#)
[Reencoding_SourceVideoClip](#) [Reencoding_StartFrame](#) [Reencoding_StartTime](#) [Reencoding_StopFrame](#)
[Reencoding_StopTime](#) [Reencoding_UseFrameGrabber](#) [Reencoding_UseVideoCompressor](#)
[Reencoding_WMVOutput](#) [StartReencoding](#) [StopReencoding](#)

Created with the Standard Edition of HelpNDoc: [Transform Your Help Documentation Process with a Help Authoring Tool](#)

Reencoding_UseFrameGrabber

TVideoGrabber.Reencoding_UseFrameGrabber

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies if the frame grabber must be used when reencoding.

Declaration

property Reencoding_UseFrameGrabber: Boolean **read** GetReencodingUseFrameGrabber **write** SetReencodingUseFrameGrabber **default** DEF_Reencoding_UseFrameGrabber;

__property **bool** Reencoding_UseFrameGrabber=read=GetReencodingUseFrameGrabber, write=SetReencodingUseFrameGrabber, **default**=1

Property Reencoding_UseFrameGrabber As Boolean

Description

Used to specify if the [frame grabber](#) must to use to reencode the video stream of the video clip. Using the frame grabber lets you apply text and/or graphics overlay over the video frames.

See Also

[OnReencodingCompleted](#) [OnReencodingProgress](#) [OnReencodingStarted](#) [Reencoding_IncludeAudioStream](#)

[Reencoding_IncludeVideoStream](#) [Reencoding_Method](#) [Reencoding_NewVideoClip](#)
[Reencoding_SourceVideoClip](#) [Reencoding_StartFrame](#) [Reencoding_StartTime](#) [Reencoding_StopFrame](#)
[Reencoding_StopTime](#) [Reencoding_UseAudioCompressor](#) [Reencoding_UseVideoCompressor](#)
[Reencoding_WMVOutput](#) [StartReencoding](#) [StopReencoding](#)

Created with the Standard Edition of HelpNDoc: Streamline Your Documentation Creation with a Help
 Authoring Tool

Reencoding_UseVideoCompressor

TVideoGrabber.Reencoding_UseVideoCompressor

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the current video compressor to use.

Declaration

property Reencoding_UseVideoCompressor: Boolean **read** GetReencodingUseVideoCompressor **write** SetReencodingUseVideoCompressor **default** DEF_Reencoding_UseVideoCompressor;

__property **bool** Reencoding_UseVideoCompressor=read=GetReencodingUseVideoCompressor,
 write=SetReencodingUseVideoCompressor, **default**=0

Property Reencoding_UseVideoCompressor As Boolean

Description

Used to specify if the current [video compressor](#) must to use to reencode the video stream of the video clip.

See Also

[OnReencodingCompleted](#) [OnReencodingProgress](#) [OnReencodingStarted](#) [Reencoding_IncludeAudioStream](#)
[Reencoding_IncludeVideoStream](#) [Reencoding_Method](#) [Reencoding_NewVideoClip](#)
[Reencoding_SourceVideoClip](#) [Reencoding_StartFrame](#) [Reencoding_StartTime](#) [Reencoding_StopFrame](#)
[Reencoding_StopTime](#) [Reencoding_UseAudioCompressor](#) [Reencoding_UseFrameGrabber](#)
[Reencoding_WMVOutput](#) [StartReencoding](#) [StopReencoding](#)

Created with the Standard Edition of HelpNDoc: Quickly and Easily Convert Your Word Document to an
 ePub or Kindle eBook

Reencoding_WMVOutput

TVideoGrabber.Reencoding_WMVOutput

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies if the reencoding output must be in WMV format

Declaration

property Reencoding_WMVOutput: Boolean **read** GetReencodingWMVOutput **write** SetReencodingWMVOutput **default** DEF_Reencoding_WMVOutput;

__property **bool** Reencoding_WMVOutput=read=GetReencodingWMVOutput,
 write=SetReencodingWMVOutput, **default**=1

Property Reencoding_WMVOutput As Boolean

Description

Used to specify if the reencoding output must be in WMV format.

1) [Reencoding_WMVOutput](#) enabled:

the extension of the video clip specified in [Reencoding_NewVideoClip](#) will be ".wmv".

1) [Reencoding WMVOutput](#) disabled:

you have to specify the proper value according to the reencoding settings:

- a) if [Reencoding Method](#) = rm_ASF, the extension must be **asf**,
- b) if [Reencoding Method](#) = rm_AVI, the extension must be:
 - **mpg** if the video clip will contain MPEG video,
 - **avi** if the video clip will contain only video, or both audio and video, compressed or not
 - **wav** if the video clip contains only uncompressed audio
 - **mp3** if the video clip contains only mp3 audio.

See Also

[OnReencodingCompleted](#) [OnReencodingProgress](#) [OnReencodingStarted](#) [Reencoding IncludeAudioStream](#)
[Reencoding IncludeVideoStream](#) [Reencoding Method](#) [Reencoding NewVideoClip](#)
[Reencoding SourceVideoClip](#) [Reencoding StartFrame](#) [Reencoding StartTime](#) [Reencoding StopFrame](#)
[Reencoding StopTime](#) [Reencoding UseAudioCompressor](#) [Reencoding UseFrameGrabber](#)
[Reencoding UseVideoCompressor](#) [StartReencoding](#) [StopReencoding](#)

Created with the Standard Edition of HelpNDoc: [Benefits of a Help Authoring Tool](#)

ScreenRecordingLayeredWindows**TVideoGrabber.ScreenRecordingLayeredWindows**[Prev](#)[Next](#)

[TVideoGrabber](#) [Properties](#)

Activates the screen recording of layered windows

Declaration

property ScreenRecordingLayeredWindows: Boolean **read** GetScreenRecordingLayeredWindows **write** SetScreenRecordingLayeredWindows **default** DEF_ScreenRecordingLayeredWindows;

__property bool ScreenRecordingLayeredWindows=read=GetScreenRecordingLayeredWindows, write=SetScreenRecordingLayeredWindows, **default**=0

Property ScreenRecordingLayeredWindows as Boolean

Description

Used to enable the screen recording of layered windows (e.g. floating or transparent windows).

By default the layered windows are not captured, enabling this property activates it.

Note that enabling this property may can affect the screen recording performances and/or cause a blinking cursor symptom.

See Also

[TOnEnumerateWindows](#) [EnumerateWindows](#) [ScreenRecordingMonitor](#)
[ScreenRecordingNonVisibleWindows](#) [ScreenRecordingUsingCoordinates](#) [ScreenRecordingWithCursor](#)
[SetWindowRecordingByHandle](#) [SetWindowRecordingByName](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Documentation Efficiency with a Help Authoring Tool](#)

ScreenRecordingMonitor**TVideoGrabber.ScreenRecordingMonitor**[Prev](#)[Next](#)

[TVideoGrabber](#) [Properties](#)

Number of the monitor to record when performing screen recording.

Declaration

property ScreenRecordingMonitor: LongInt **read** GetScreenRecordingMonitor **write** SetScreenRecordingMonitor **default** 0

__property int ScreenRecordingMonitor==GetScreenRecordingMonitor,
write=SetScreenRecordingMonitor, **default**=0;

Property ScreenRecordingMonitor as Long

Description

Specifies the monitor to record when using the screen recording and the desktop is extended to several monitors.

Just specify the monitor number (0 for the 1st monitor, 1 for the 2nd monitor, etc)...

See Also

[TOnEnumerateWindows](#) [EnumerateWindows](#) [ScreenRecordingLayeredWindows](#)
[ScreenRecordingNonVisibleWindows](#) [ScreenRecordingUsingCoordinates](#) [ScreenRecordingWithCursor](#)
[SetWindowRecordingByHandle](#) [SetWindowRecordingByName](#)

Created with the Standard Edition of HelpNDoc: [Bring your WinHelp HLP help files into the present with HelpNDoc's easy CHM conversion](#)

ScreenRecordingNonVisibleWindows

TVideoGrabber.ScreenRecordingNonVisibleWindows

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Activates the screen recording of non-visible windows

Declaration

property ScreenRecordingNonVisibleWindows: Boolean **read** GetScreenRecordingNonVisibleWindows **write** SetScreenRecordingNonVisibleWindows **default** DEF_ScreenRecordingNonVisibleWindows;

__property bool ScreenRecordingNonVisibleWindows=read=GetScreenRecordingNonVisibleWindows,
write=SetScreenRecordingNonVisibleWindows, **default**=0

Property ScreenRecordingNonVisibleWindows as Boolean

Description

Used to enable the screen recording of non-visible windows.

The window must have been selected first with [SetWindowRecordingByName](#) or [SetWindowRecordingByHandle](#)

Note: this mode may give variable results depending on the content of the window

See Also

[TOnEnumerateWindows](#) [EnumerateWindows](#) [ScreenRecordingLayeredWindows](#) [ScreenRecordingMonitor](#)
[ScreenRecordingUsingCoordinates](#) [ScreenRecordingWithCursor](#) [SetWindowRecordingByHandle](#)
[SetWindowRecordingByName](#)

Created with the Standard Edition of HelpNDoc: [Step-by-Step Guide: How to Turn Your Word Document into an eBook](#)

ScreenRecordingSizePercent

TVideoGrabber.ScreenRecordingSizePercent

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Downsizes the video size of the screen recording

Declaration

property ScreenRecordingSizePercent: LongInt **read** GetScreenRecordingSizePercent **write** SetScreenRecordingSizePercent **default** 100

__property int ScreenRecordingSizePercent==ScreenRecordingSizePercent,
write=SetScreenRecordingSizePercent, **default**=100;

Property ScreenRecordingSizePercent as Long

Description

By default, when ScreenRecordingSizePercent = 100, the video size of the screen recording is the pixel size of the monitor (100%)

To preview or record e.g. at half of the monitor size (50%), set ScreenRecordingSizePercent = 50

Created with the Standard Edition of HelpNDoc: [Benefits of a Help Authoring Tool](#)

ScreenRecordingThroughClipboard

TVideoGrabber.ScreenRecordingThroughClipboard

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Uses the clipboard to perform the screen recording.

Declaration

property ScreenRecordingThroughClipboard: Boolean **read** GetScreenRecordingThroughClipboard **write** SetScreenRecordingThroughClipboard **default** DEF_ScreenRecordingThroughClipboard;

__property bool ScreenRecordingThroughClipboard=read=GetScreenRecordingThroughClipboard,
write=SetScreenRecordingThroughClipboard, **default**=0

Property ScreenRecordingThroughClipboard As Boolean

Description

When enabled, the clipboard is used to perform the screen recording.

Enabling this property may let you capture more information than when it is disabled (by default).

If some parts of the screen are not properly captured or recorded, try to switch the value of this property.

Note: any existing clipboard data will be destroyed during preview or recording when this property is enabled.

Created with the Standard Edition of HelpNDoc: [Quickly and Easily Convert Your Word Document to an ePub or Kindle eBook](#)

ScreenRecordingWithCursor

TVideoGrabber.ScreenRecordingWithCursor

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Shows the cursor during the screen recording.

Declaration

property ScreenRecordingWithCursor: Boolean **read** GetScreenRecordingWithCursor **write** SetScreenRecordingWithCursor **default** DEF_ScreenRecordingWithCursor;

__property **bool** ScreenRecordingWithCursor=read=GetScreenRecordingWithCursor, write=SetScreenRecordingWithCursor, **default**=1

Property ScreenRecordingWithCursor As Boolean

Description

Enables the cursor recording during the screen recording.
When disabled, the cursor does not appear.

See Also

[TOnEnumerateWindows](#) [EnumerateWindows](#) [ScreenRecordingLayeredWindows](#) [ScreenRecordingMonitor](#) [ScreenRecordingNonVisibleWindows](#) [ScreenRecordingUsingCoordinates](#) [SetWindowRecordingByHandle](#) [SetWindowRecordingByName](#)

Created with the Standard Edition of HelpNDoc: [Experience the Power and Simplicity of HelpNDoc's User Interface](#)

SendToDV_DeviceIndex

TVideoGrabber.SendToDV_DeviceIndex

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Index of the DV device that will receive the audio/video streams.

Declaration

property SendToDV_DeviceIndex: LongInt **read** GetSendToDV_DeviceIndex **write** SetSendToDV_DeviceIndex;

__property **int** SendToDV_DeviceIndex=read=GetSendToDV_DeviceIndex, write=SetSendToDV_DeviceIndex, **nodefault**

Property SendToDV_DeviceIndex As Long

Description

This property is used to specify the index of the DV device (in the [VideoDevices](#) list) that will receive the audio/video streams when the [RecordingMethod](#) is set to **rm_SendToDV**.
See the "[Send to DV](#)" chapter for more information.

See Also

[Send to DV](#) [DVEncoder](#) [VideoFormat](#) [DVEncoder](#) [VideoResolution](#) [DVEncoder](#) [VideoStandard](#)

Created with the Standard Edition of HelpNDoc: [Full-featured EPub generator](#)

ShapeOverlay

TVideoGrabber.ShapeOverlay

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

TShape component that will be drawn over video frames.

Declaration

property ShapeOverlay: TShape **read** GetShapeOverlay **write** SetShapeOverlay **default** DEF_ShapeOverlay;

__property ExtCtrls::TShape *ShapeOverlay=read=GetShapeOverlay, write=SetShapeOverlay, **default**=0

n/a

Description

Used to assign to TVideoGrabber a TShape component that will be drawn over video frames. The TShape component is really drawn over video frames when [ShapeOverlayEnabled](#) is true.

To use this feature, simply put a TShape component on the form, then, using the object inspector, assign ShapeOverlay to it.

It is possible to draw **more than one shape** over video frames by using [ShapeOverlayList](#) instead of this property.

Notes:

- to avoid having TShape components placed on a form visible at runtime, simply set their Visible property to false.
- to use this property, the [frame grabber](#) is enabled.

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronon](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#) [SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#) [SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#) [SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#) [SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#) [TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#) [TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#) [TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#) [TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Effortlessly Convert Your Word Doc to an eBook: A Step-by-Step Guide

ShapeOverlayEnabled

TVideoGrabber.ShapeOverlayEnabled

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables drawing of the [ShapeOverlay](#) TShape component over video frames.

Declaration

property ShapeOverlayEnabled: Boolean **read** GetShapeOverlayEnabled **write** SetShapeOverlayEnabled **default** DEF_ShapeOverlayEnabled;

__property **bool** ShapeOverlayEnabled=read=GetShapeOverlayEnabled, write=SetShapeOverlayEnabled, **default**=0

n/a

Description

Used to enable/disable drawing of the [ShapeOverlay](#) TShape component over video frames. Available only if the [frame grabber](#) is enabled.

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#) [SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#) [SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#) [SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#) [SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#) [TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#) [TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#) [TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#) [TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Don't be left in the past: convert your WinHelp HLP help files to CHM with HelpNDoc

SourceStream

TVideoGrabber.SourceStream

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Used to play a video clip from a TStream

Declaration

property SourceStream: TStream **read** GetSourceStream **write** SetSourceStream;

__property **TStream** *SourceStream=read=GetSourceStream, write=SetSourceStream

Description

Used to play a video clip from a TMemoryStream, TFileStream or TStream descendent.

When this property is assigned before invoking [OpenPlayer](#), the stream data is used as source by the player.

To go back to the normal player behavior, set a NULL or nil value to this property after invoking [ClosePlayer](#) to let TVideoGrabber use the clip specified in the [PlayerFileName](#) property.

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#)

[PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#)
[PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#)
[PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#)
[PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchron](#)
[PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [StopPlayer](#)
[SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#)
[VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Achieve Professional Documentation Results with a Help Authoring Tool](#)

SpeakerBalance

TVideoGrabber.SpeakerBalance

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Sets the speaker balance.

Declaration

property SpeakerBalance: LongInt **read** GetSpeakerBalance **write** SetSpeakerBalance;

__property **int** SpeakerBalance=read=GetSpeakerBalance, write=SetSpeakerBalance, **nodefault**

Property SpeakerBalance As Long

Description

Used to set the speaker balance. This property is active when [SpeakerControl](#) is enabled.

Valid range: -32768 ... 32767.

0 is the center point.

If [AssociateAudioAndVideoDevices](#) is enabled, the current value is saved with the current [video capture device](#), and automatically restored then this video capture device is selected again (when [VideoDevice](#) is assigned).

See Also

[TVUMeter](#) [TVUMeterSetting](#) [AudioBalance](#) [AudioChannelRenderMode](#) [AudioVolume](#) [MuteAudioRendering](#)
[SpeakerControl](#) [SpeakerVolume](#) [VUMeter](#)

Created with the Standard Edition of HelpNDoc: [Easily create Qt Help files](#)

SpeakerControl

TVideoGrabber.SpeakerControl

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables the control of the speaker balance and volume.

Declaration

property SpeakerControl: Boolean **read** GetSpeakerControl **write** SetSpeakerControl **default** DEF_SpeakerControl;

__property **bool** SpeakerControl=read=GetSpeakerControl, write=SetSpeakerControl, **default**=0

Property SpeakerControl As Boolean

Description

Used to enable/disable the control of the speaker balance ([SpeakerBalance](#)) and volume ([SpeakerVolume](#)) from TVideoGrabber.

See Also

[TVUMeter](#) [TVUMeterSetting](#) [AudioBalance](#) [AudioChannelRenderMode](#) [AudioVolume](#) [MuteAudioRendering](#) [SpeakerBalance](#) [SpeakerVolume](#) [VUMeter](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Convert Your Markdown Content with HelpNDoc](#)

SpeakerVolume

TVideoGrabber.SpeakerVolume

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Sets the speaker volume.

Declaration

property SpeakerVolume: LongInt **read** GetSpeakerVolume **write** SetSpeakerVolume;

__property int SpeakerVolume=read=GetSpeakerVolume, write=SetSpeakerVolume, **nodefault**

Property SpeakerVolume As Long

Description

Used to set the speaker volume. This property is active when [SpeakerControl](#) is enabled.

Valid range: 0...65535.

If [AssociateAudioAndVideoDevices](#) is enabled, the current value is saved with the current [video capture device](#), and automatically restored then this video capture device is selected again (when [VideoDevice](#) is assigned).

See Also

[TVUMeter](#) [TVUMeterSetting](#) [AudioBalance](#) [AudioChannelRenderMode](#) [AudioVolume](#) [MuteAudioRendering](#) [SpeakerBalance](#) [SpeakerControl](#) [VUMeter](#)

Created with the Standard Edition of HelpNDoc: [Easily create Help documents](#)

StoragePath

TVideoGrabber.StoragePath

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Sets or retrieves the full qualified path for recordings and BMP or JPEG files. This property is NOT stored.

Declaration

property StoragePath: **string** **read** GetStoragePath **write** SetStoragePath **stored** False;

__property wchar_t *StoragePath=read=GetStoragePath, write=SetStoragePath, stored=**false**

Property StoragePath As String

Description

Used to set or retrieve the full qualified path where recordings and BMP or JPEG frame captures will be saved.

If left blank, the current directory is used.

As this property depends of the current platform and partition layout, it is not stored and must be set programmatically at runtime.

WARNING: be careful not to use a slow drive as storage path (networked drive, floppy disk, slow hard disk, etc...), otherwise it will slow down recording and cause an number of dropped frames.

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#)

[TFrameCaptureDest](#) [TOnFrameCaptureCompleted](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#)
[TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#)
[AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [BurstCount](#) [BurstInterval](#) [BurstMode](#)
[BurstType](#) [CaptureFrameSyncTo](#) [CaptureFrameTo](#) [Encoder](#) [SetInt](#) [FrameCaptureHeight](#)
[FrameCaptureWidth](#) [FrameCaptureWithoutOverlay](#) [FrameCaptureZoomSize](#) [GetFrameInfoString](#)
[GetLastFrameAsHBITMAP](#) [GetLastFrameAsTBitmap](#) [GetLastFrameBitmapBits](#) [GetLastFrameBitmapBits2](#)
[GetLastFrameWaitTimeoutMs](#) [HoldRecording](#) [IsRecordingPaused](#) [JPEGPerformance](#)
[JPEGProgressiveDisplay](#) [JPEGQuality](#) [Last](#) [BurstFrameCapture](#) [FileName](#)
[Last](#) [CaptureFrameTo](#) [FileName](#) [Last](#) [Recording](#) [FileName](#) [OnBacktimedFramesCountReached](#)
[OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#)
[OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnFrameBitmap](#)
[OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#)
[OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#)
[PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#)
[RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#)
[RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#)
[RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#)
[ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetFrameCaptureBounds](#)
[SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#)
[Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Easy to use tool to create HTML Help files and Help web sites](#)

StoreDeviceSettingsInRegistry

TVideoGrabber.StoreDeviceSettingsInRegistry

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

If true, device-dependent settings are saved in the registry.

Declaration

property StoreDeviceSettingsInRegistry: Boolean **read** GetStoreDeviceSettingsInRegistry **write** SetStoreDeviceSettingsInRegistry **default** DEF_StoreDeviceSettingsInRegistry;

__property **bool** StoreDeviceSettingsInRegistry=read=GetStoreDeviceSettingsInRegistry, write=SetStoreDeviceSettingsInRegistry, **default**=1

Property StoreDeviceSettingsInRegistry As Boolean

Description

Used to save device-dependent settings in the registry.

If enabled, TVideoGrabber automatically saves and retrieves the current value of all the [device-dependent](#) parameters whose value and range change from a video capture device to the other.

See Also

[FindIndexInListByName](#) [OnVideoDeviceSelected](#)

Created with the Standard Edition of HelpNDoc: [Streamline Your Documentation Process with HelpNDoc's Project Analyzer](#)

StreamingURL

TVideoGrabber.StreamingURL

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

URL that must be given to users to connect to the current live streaming.

Declaration

property StreamingURL: **string read** GetStreamingURL;

__property wchar_t *StreamingURL=read=GetStreamingURL

Property StreamingURL As String

Description

Returns the URL that must be given to users to connect to the current direct network streaming. This URL is also returned by the [OnDirectNetworkStreamingHostUrl](#) event when starting direct network streaming.

E.g.:

See Also

[TNetworkStreamingType](#) [TOnClientConnection](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#) [ASFAudioBitRate](#) [ASFAudioChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#) [ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#) [ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#) [ASFProfileFromCustomFile](#) [ASFProfiles](#) [ASFProfilesCount](#) [ASFProfileVersion](#) [ASFVideoBitRate](#) [ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#) [NetworkStreaming](#) [NetworkStreamingType](#) [OnAuthenticationNeeded](#) [OnClientConnection](#) [OnDirectNetworkStreamingHostUrl](#) [ShowDialog](#) [StartAudioRendering](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly upgrade your WinHelp HLP help files to CHM with HelpNDoc](#)

StreamInterface_Format

TVideoGrabber.StreamInterface_Format

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the format of the raw samples (e.g. H264) pushed through the [StreamInterface.PushData](#) function

Declaration

property StreamInterface_Format: **string read** GetStreamInterface_Format **write** SetStreamInterface_Format;

__property System::wchar_t *StreamInterface_Format==GetStreamInterface_Format, write=SetStreamInterface_Format;

Description

Specifies the format of the raw samples (e.g. H264) pushed through the [StreamInterface.PushData](#) function

See Also

[Stream Interface](#)

Created with the Standard Edition of HelpNDoc: [Create Professional CHM Help Files with HelpNDoc's Easy-to-Use Tool](#)

StreamInterface_FrameRate

TVideoGrabber.StreamInterface_FrameRate

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the frame rate of the raw samples (e.g. H264) pushed through the [StreamInterface.PushData](#) function

Declaration

property StreamInterface_FrameRate: Double **read** GetStreamInterface_FrameRate **write** SetStreamInterface_FrameRate;

__**property double** StreamInterface_FrameRate==GetStreamInterface_FrameRate,
write=SetStreamInterface_FrameRate;

Description

Specifies the frame rate of the raw samples (e.g. H264) pushed through the [StreamInterface_PushData](#) function

See Also

[Stream Interface](#)

Created with the Standard Edition of HelpNDoc: [Easily create Qt Help files](#)

StreamInterface_IsRealTime

TVideoGrabber.StreamInterface_IsRealTime

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies whether the raw samples (e.g. H264) pushed through the [StreamInterface_PushData](#) function are samples received in real-time, or samples coming e.g. from a file

Declaration

property StreamInterface_IsRealTime: Boolean **read** GetStreamInterface_IsRealTime **write** SetStreamInterface_IsRealTime;

__**property bool** StreamInterface_IsRealTime==GetStreamInterface_IsRealTime,
write=SetStreamInterface_IsRealTime, **nodefault**};

Description

Specifies whether the raw samples (e.g. H264) pushed through the [StreamInterface_PushData](#) function are samples received in real-time, or samples coming e.g. from a file

See Also

[stream interface](#)

Created with the Standard Edition of HelpNDoc: [Elevate Your Help Documentation with a Help Authoring Tool](#)

SyncCommands

TVideoGrabber.SyncCommands

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Synchronize the TVideoGrabber commands.

Declaration

property SyncCommands: Boolean **read** GetSyncCommands **write** SetSyncCommands **default** DEF_SyncCommands;

__**property bool** SyncCommands=read=GetSyncCommands, write=SetSyncCommands, **default**=0

property SyncCommands as Boolean

Description

By default this property is enabled so TVideoGrabber is blocking and synchronone (it waits for the commands to be completed)

If you want to invoke the component from a thread to avoid blocking the main thread, disable this property.

See also [EventNotificationSynchronone](#)

Created with the Standard Edition of HelpNDoc: [Bring your WinHelp HLP help files into the present with HelpNDoc's easy CHM conversion](#)

SynchronizationRole

TVideoGrabber.SynchronizationRole

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Specify if the synchronized player is a master or slave

Declaration

property SynchronizationRole: TSynchronizationRole **read** GetSynchronizationRole **write** SetSynchronizationRole **default** DEF_SynchronizationRole;

__property TSynchronizationRole SynchronizationRole=read=GetSynchronizationRole, write=SetSynchronizationRole, **default**=0

Property SynchronizationRole as TSynchronizationRole

Description

Used when the players are synchronized ([Synchronized](#) = true), to determine if the player is a master player or a slave player.

sr_Master: this player is the master component

sr_Slave: this player is a slave component

(only one master player allowed at the time)

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip](#) [Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchronone](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Elevate Your CHM Help Files with HelpNDoc's Advanced Customization Options](#)

Synchronized

TVideoGrabber.Synchronized

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Enable the synchronization between several TVideoGrabber components.

Declaration

property Synchronized: Boolean **read** GetSynchronized **write** SetSynchronized **default** DEF_Synchronized;

___property **bool** Synchronized=read=GetSynchronized, write=SetSynchronized, **default**=0

Property Synchronized As Boolean

Description

Used to enable the synchronization between several TVideoGrabber player components.

See the [Synchronization of several TVideoGrabber components](#) chapter.

See Also

[Player features](#) [Recording methods and properties](#) [TAVIMuxConfig](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnFrameProgress](#) [TOnPlayerBufferingData](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TOnResizeVideo](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioChannelRenderMode](#) [AudioRecording](#) [AudioStreamNumber](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AutoRefreshPreview](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [BackgroundColor](#) [ClosePlayer](#) [Display](#) [AutoSize](#) [Display](#) [FullScreen](#) [Display](#) [SetLocation](#) [Display](#) [VideoPortEnabled](#) [Encoder](#) [SetInt](#) [FastForwardPlayer](#) [HoldRecording](#) [ImageRatio](#) [InFrameProgressEvent](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [IsRecordingPaused](#) [IsVideoInterlaced](#) [IsVideoPortAvailable](#) [Last](#) [Clip](#) [Played](#) [Last](#) [Recording](#) [FileName](#) [MP4NeedsReindexing](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnFrameProgress](#) [OnKeyPress](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OnPreviewStarted](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [OnResizeVideo](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PausePreview](#) [PauseRecording](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchronise](#) [PlayerVideoCodec](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [PreviewZoomSize](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumePreview](#) [ResumeRecording](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [ShowDialog](#) [SourceStream](#) [StartAudioRecording](#) [StartPreview](#) [StartRecording](#) [StartSynchronized](#) [StopPlayer](#) [StopPreview](#) [StopRecording](#) [StoragePath](#) [SynchronizationRole](#) [SyncPreview](#) [UseNearestVideoSize](#) [VideoHeight](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoRenderer](#) [VideoRendererPriority](#) [VideoWidth](#) [VideoWidth](#) [PreferredAspectRatio](#) [Visible](#)

Created with the Standard Edition of HelpNDoc: Transform Your Help Documentation Process with a Help Authoring Tool

SyncPreview

TVideoGrabber.SyncPreview

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables/disables the audio/video sync of the preview.

Declaration

property SyncPreview: TSyncPreview **read** GetSyncPreview **write** SetSyncPreview **default** DEF_SyncPreview;

___property TSyncPreview SyncPreview=read=GetSyncPreview, write=SetSyncPreview, **default**=0

Property SyncPreview As TSyncPreview

Description

This [TSyncPreview](#) property is used to maintain the audio rendered and the video displayed in sync, during preview as well as during recording.

When this property is enabled during recording, this can cause an excessive amount of dropped frames with some video capture devices.

Disabling this property can cause a time shift between the audio rendered and the video displayed.

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last_Recording_FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Create Encrypted, Password-Protected PDFs](#)

SystemTempPath

TVideoGrabber.SystemTempPath

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Returns the current system temporary path.

Declaration

property SystemTempPath: **string** read GetSystemTempPath;

__property wchar_t *SystemTempPath=read=GetSystemTempPath

Property SystemTempPath As String

Description

Used to retrieve the current system temporary path.

Can be used e.g. to be assigned to the [StoragePath](#) property, in order to save the recorded clips and captured frames in the temporary directory.

Created with the Standard Edition of HelpNDoc: [Easy to use tool to create HTML Help files and Help web sites](#)

TextOverlay_Align

TVideoGrabber.TextOverlay_Align

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Sets or retrieves the alignment used to draw text over video frames for the current text overlay selected by [TextOverlay_Selector](#)

Declaration

property TextOverlay_Align: TTextOverlayAlign **read** GetTextOverlayAlign **write** SetTextOverlayAlign **default** DEF_TextOverlay_Align;

__property TTextOverlayAlign TextOverlay_Align=read=GetTextOverlayAlign, write=SetTextOverlayAlign, **default**=0

Property TextOverlay_Align As TTextOverlayAlign

Description

Used to set or retrieve the alignment used to draw text over video frames.
The text will be drawn between [TextOverlay_Left](#) and [TextOverlay_Right](#) positions.

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronon](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#) [SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#) [SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#) [SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#) [SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#) [SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay_Align](#) [SetTextOverlay_BkColor](#) [SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#) [SetTextOverlay_GradientColor](#) [SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#) [SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#) [SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#) [SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_BkColor](#) [TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#) [TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#) [TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#) [TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Free Kindle producer

TextOverlay_AlphaBlend

TVideoGrabber.TextOverlay_AlphaBlend

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables the alpha blending of the current text overlay

Declaration

property TextOverlay_AlphaBlend: Boolean **read** GetTextOverlay_AlphaBlend **write** SetTextOverlay_AlphaBlend **default** DEF_TextOverlay_AlphaBlend;

__property bool TextOverlay_AlphaBlend==GetTextOverlay_AlphaBlend, write=SetTextOverlay_AlphaBlend, **default**=0;

property TextOverlay_AlphaBlend as Boolean

Description

Enables the alpha blending of an overlayed text selected by [TextOverlay_Selector](#)
The alpha blending value in the 0..255 range must be specified with [TextOverlay_AlphaBlendValue](#)

Look at the [Text Overlays](#) chapter for more information.

Created with the Standard Edition of HelpNDoc: [5 Reasons Why a Help Authoring Tool is Better than Microsoft Word for Documentation](#)

TextOverlay_AlphaBlendValue

TVideoGrabber.TextOverlay_AlphaBlendValue

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the the alpha blending value of the current text overlay

Declaration

property TextOverlay_AlphaBlendValue: LongInt **read** GetTextOverlay_AlphaBlendValue **write** SetTextOverlay_AlphaBlendValue **default** DEF_TextOverlay_AlphaBlendValue;

__**property** int TextOverlay_AlphaBlendValue==GetTextOverlay_AlphaBlendValue, write=SetTextOverlay_AlphaBlendValue, **default**=180;

property TextOverlay_AlphaBlendValue as Long

Description

Specifies the alpha blending value of an overlaid text selected by [TextOverlay_Selector](#)
The alpha blending must be enabled by [TextOverlay_AlphaBlend](#)
Look at the [Text Overlays](#) chapter for more information.

Created with the Standard Edition of HelpNDoc: [Make CHM Help File Creation a Breeze with HelpNDoc](#)

TextOverlay_BkColor

TVideoGrabber.TextOverlay_BkColor

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Sets or retrieves the background color used to draw text over frames for the current text overlay selected by [TextOverlay_Selector](#)

Declaration

property TextOverlay_BkColor: TColor **read** GetTextOverlayBkColor **write** SetTextOverlayBkColor **default** DEF_TextOverlay_BkColor;

__property Graphics::TColor TextOverlay_BkColor=read=GetTextOverlayBkColor, write=SetTextOverlayBkColor, **default**=16777215

Property TextOverlay_BkColor As OLE_COLOR

Description

Used to set or retrieve the background color used to draw text over frames.
Useful only if [TextOverlay_Transparent](#) is disabled.

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#) [SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#) [SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#) [SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#)

[SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#)
[SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay_Align](#) [SetTextOverlay_BkColor](#) [SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#)
[SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#) [SetTextOverlay_GradientColor](#)
[SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#)
[SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#)
[SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#)
[SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#)
[ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_CreateCustomFont](#)
[TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#) [TextOverlay_FontColor](#)
[TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#) [TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#)
[TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#) [TextOverlay_ShadowDirection](#) [TextOverlay_String](#)
[TextOverlay_Top](#) [TextOverlay_Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: iPhone web sites made easy

TextOverlay_Enabled

TVideoGrabber.TextOverlay_Enabled

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables/disables drawing text over video frames for the current text overlay selected by [TextOverlay_Selector](#)

Declaration

property TextOverlay_Enabled: Boolean **read** GetTextOverlayEnabled **write** SetTextOverlayEnabled **default** DEF_TextOverlay_Enabled;

__property **bool** TextOverlay_Enabled=read=GetTextOverlayEnabled, write=SetTextOverlayEnabled, **default**=0

Property TextOverlay_Enabled As Boolean

Description

Used to enable /disable text overlay over video frames.

Note: the [frame grabber](#) must be enabled to use the set of TextOverlay properties.

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#)
[AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay_StretchToVideoSize](#)
[MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#)
[OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay_AlphaBlend](#)
[SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#)
[SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#)
[SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#)
[SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#)
[SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#)
[SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay_Align](#) [SetTextOverlay_BkColor](#) [SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#)
[SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#) [SetTextOverlay_GradientColor](#)
[SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#)
[SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#)
[SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#)
[SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#)

[ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#) [TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Font](#) [TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#) [TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#) [TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [iPhone web sites made easy](#)

TextOverlay_Font

TVideoGrabber.TextOverlay_Font

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Sets/retrieves the font used to draw text over video frames for the current text overlay selected by [TextOverlay_Selector](#)

Declaration

property TextOverlay_Font: TFont **read** GetTextOverlayFont **write** SetTextOverlayFont **default** DEF_TextOverlay_Font;

__property Graphics::TFont *TextOverlay_Font=read=GetTextOverlayFont, write=SetTextOverlayFont, default=0

Property TextOverlay_Font As Long

Description

Used to set / retrieve the font used to draw text over video frames.

In C#.NET, use the ToHfont().ToInt32() function of the Font object. E.g.:

```
Font NewFont = new Font("Courier New", 14);
axVideoGrabberNET1.TextOverlay_Font = NewFont.ToHfont().ToInt32();
axVideoGrabberNET1.TextOverlay_Enabled = true;
axVideoGrabberNET1.StartPreview();
```

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#) [SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#) [SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#) [SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#) [SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#) [TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#)

[TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#)
[TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#)
[TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#)
[TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Free EBook and documentation generator

TextOverlay_FontColor

TVideoGrabber.TextOverlay_FontColor

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Adjusts the font color in the OCX and DLL versions for the current text overlay selected by [TextOverlay_Selector](#)

Declaration

property TextOverlay_FontColor: TColor **read** GetTextOverlayFontColor **write** SetTextOverlayFontColor
default DEF_TextOverlay_FontColor

__property TColor TextOverlay_Selector==GetTextOverlaySelector, write=SetTextOverlaySelector, **default** 0xFFFF00;

property TextOverlay_FontColor as Long

Description

Used to adjust the font color in the OCX and DLL versions
 (for Delphi and C++Builder modify directly the [TextOverlay_Font](#) property)

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#)
[AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay_StretchToVideoSize](#)
[MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#)
[OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay_AlphaBlend](#)
[SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#)
[SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#)
[SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#)
[SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#)
[SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#)
[SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay_Align](#) [SetTextOverlay_BkColor](#) [SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#)
[SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#) [SetTextOverlay_GradientColor](#)
[SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#)
[SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#)
[SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#)
[SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#)
[ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#)
[TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#)
[TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#) [TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#)
[TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#) [TextOverlay_ShadowDirection](#) [TextOverlay_String](#)
[TextOverlay_Top](#) [TextOverlay_Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Free EPub producer

TextOverlay_Left

TVideoGrabber.TextOverlay_Left

[Prev](#)

[Next](#)

TVideoGrabber **Properties**

Left position (in pixels) where the text will be drawn over video frames for the current text overlay selected by [TextOverlay_Selector](#)

Declaration

property TextOverlay_Left: LongInt **read** GetTextOverlayLeft **write** SetTextOverlayLeft **default** DEF_TextOverlay_Left;

__property **int** TextOverlay_Left=read=GetTextOverlayLeft, write=SetTextOverlayLeft, **default**=0

Property TextOverlay_Left As Long

Description

Used to set or retrieve the left position (in pixels) where the text will be drawn over video frames. The text will be left-padded on this position if [TextOverlay_Align](#) = tf_Left.

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay_StretchToVideoSize](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay_AlphaBlend](#) [SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#) [SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#) [SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#) [SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#) [SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#) [SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay_Align](#) [SetTextOverlay_BkColor](#) [SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#) [SetTextOverlay_GradientColor](#) [SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#) [SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#) [SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#) [SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#) [TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#) [TextOverlay_FontColor](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#) [TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#) [TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Leave the tedious WinHelp HLP to CHM conversion process behind with HelpNDoc

TextOverlay_Right

TVideoGrabber.TextOverlay_Right

[Prev](#)
[Next](#)

TVideoGrabber **Properties**

Right position (in pixels) where the text will be drawn over video frames for the current text overlay selected by [TextOverlay_Selector](#)

Declaration

property TextOverlay_Right: LongInt **read** GetTextOverlayRight **write** SetTextOverlayRight **default** DEF_TextOverlay_Right;

__property **int** TextOverlay_Right=read=GetTextOverlayRight, write=SetTextOverlayRight, **default**=320

Property TextOverlay_Right As Long

Description

Used to set or retrieve the right position (in pixels) where the text will be drawn over video frames. The text will be left-padded on this position if [TextOverlay_Align](#) = `tf_Right`.

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#) [SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#) [SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#) [SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#) [SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#) [TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#) [TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Scrolling](#) [TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#) [TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Easily create CHM Help documents](#)

TextOverlay_Scrolling

TVideoGrabber.TextOverlay_Scrolling

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables the text scrolling for the current text overlay selected by [TextOverlay_Selector](#)

Declaration

property TextOverlay_Scrolling: Boolean **read** GetTextOverlayScrolling **write** SetTextOverlayScrolling
default DEF_TextOverlay_Scrolling;

__property **bool** TextOverlay_Scrolling=read=GetTextOverlayScrolling, write=SetTextOverlayScrolling, **default**=0

property TextOverlay_Scrolling as Boolean

Description

Enable the scrolling of the text overlay.

The scrolling speed is adjusted by the [TextOverlay_ScrollingSpeed](#) property.

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#)

[AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#)
[MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronon](#) [OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#)
[OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#)
[SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#)
[SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#)
[SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#)
[SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#)
[SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#)
[SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#)
[SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#)
[SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#)
[SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#)
[SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#)
[SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#)
[ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#)
[TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#)
[TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#)
[TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#) [TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#)
[TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [News and information about help authoring tools and software](#)

TextOverlay_ScrollingSpeed

TVideoGrabber.TextOverlay_ScrollingSpeed

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Sets the text scrolling speed for the current text overlay selected by [TextOverlay_Selector](#)

Declaration

property TextOverlay_ScrollingSpeed: LongInt **read** GetTextOverlayScrollingSpeed **write** SetTextOverlayScrollingSpeed **default** DEF_TextOverlay_ScrollingSpeed;

__property int TextOverlay_ScrollingSpeed=read=GetTextOverlayScrollingSpeed, write=SetTextOverlayScrollingSpeed, **default**=0

property TextOverlay_ScrollingSpeed as Long

Description

Used to adjust the scrolling speed of the text overlay.

The scrolling must be enabled first by enabling the [TextOverlay_Scrolling](#) property.

- TextOverlay_ScrollingSpeed = 0 -> no scrolling
- TextOverlay_ScrollingSpeed > 0 -> scrolling from right to left
- TextOverlay_ScrollingSpeed < 0 -> scrolling from left to right

E.g.:

TextOverlay_ScrollingSpeed = 1 -> slow scrolling
 TextOverlay_ScrollingSpeed = -3 -> backwards average scrolling
 TextOverlay_ScrollingSpeed = 6 -> fast scrolling

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#)

[AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronon](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#) [SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#) [SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#) [SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#) [SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#) [TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#) [TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#) [TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Easily convert your WinHelp HLP help files to CHM with HelpNDoc's step-by-step guide

TextOverlay_Selector

TVideoGrabber.TextOverlay_Selector

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Selects the current text overlay

Declaration

property TextOverlay_Selector: LongInt **read** GetTextOverlaySelector **write** SetTextOverlaySelector;

__property int TextOverlay_Selector==GetTextOverlaySelector, write=SetTextOverlaySelector, **nodefault**};

Property TextOverlay_Selector as Long

Description

Used to select a group of text overlay settings.

Up to 1000 text overlay groups of settings may be used, allowing to perform different text overlays (with different color, background, transparency, font size, etc...) at the same time.

The default text overlay settings correspond to the group 0 (TextOverlay_Selector = 0).

Note that TVideoGrabber does not allocate the resources for 1000 text overlays, by default it allocates only one font and one parameter set, corresponding to TextOverlay_Selector = 0 (the resource allocation is made when TextOverlay_Selector receives a new value never used before, see the [Text overlays](#) chapter for more explanations).

Set TextOverlay_Selector to the index of the text overlay properties to read or write before reading or writing their values.

It is applied to all the TextOverlay_... properties.

E.g.:

```
procedure TfrmMainForm.Button17Click(Sender: TObject);
begin
    VideoGrabber.TextOverlay_Selector := 0;
    VideoGrabber.TextOverlay_String := 'frame number: %frame_count%';
    VideoGrabber.TextOverlay_Left := 10;
    VideoGrabber.TextOverlay_Top := 10;
    VideoGrabber.TextOverlay_Transparent := True;
    VideoGrabber.TextOverlay_Font.Color := clRed;
    VideoGrabber.TextOverlay_Font.Size := 12;
    VideoGrabber.TextOverlay_Enabled := true;

    VideoGrabber.TextOverlay_Selector := 1;
    VideoGrabber.TextOverlay_String := 'frame time: %time_100ns%';
    VideoGrabber.TextOverlay_Left := VideoGrabber.VideoWidth - 120;
    VideoGrabber.TextOverlay_Right := VideoGrabber.VideoWidth;
    VideoGrabber.TextOverlay_Top := 60;
    VideoGrabber.TextOverlay_Transparent := False;
    VideoGrabber.TextOverlay_BkColor := clBlack;
    VideoGrabber.TextOverlay_Font.Color := clWhite;
    VideoGrabber.TextOverlay_Font.Size := 8;
    VideoGrabber.TextOverlay_Enabled := true;
end;
```

in C++:

```
void __fastcall TfrmMainForm::Button2Click(TObject *Sender)

    VideoGrabber->TextOverlay_Selector = 0;
    VideoGrabber->TextOverlay_String = "frame number: %frame_count%";
    VideoGrabber->TextOverlay_Left = 10;
    VideoGrabber->TextOverlay_Top = 10;
    VideoGrabber->TextOverlay_Transparent = True;
    VideoGrabber->TextOverlay_Font->Color = clRed;
    VideoGrabber->TextOverlay_Font->Size = 12;
    VideoGrabber->TextOverlay_Enabled = true;

    VideoGrabber->TextOverlay_Selector = 1;
    VideoGrabber->TextOverlay_String = "frame time: %time_100ns%";
    VideoGrabber->TextOverlay_Left = VideoGrabber->VideoWidth - 120;
    VideoGrabber->TextOverlay_Right = VideoGrabber->VideoWidth;
    VideoGrabber->TextOverlay_Top = 60;
    VideoGrabber->TextOverlay_Transparent = False;
    VideoGrabber->TextOverlay_BkColor = clBlack;
    VideoGrabber->TextOverlay_Font->Color = clWhite;
    VideoGrabber->TextOverlay_Font->Size = 8;
    VideoGrabber->TextOverlay_Enabled = true;
```

in VB:

```
Private Sub Command1_Click()
    VideoGrabberVB61.TextOverlay_Selector = 0
    VideoGrabberVB61.TextOverlay_String = "frame number: %frame_count%"
    VideoGrabberVB61.TextOverlay_Left = 10
    VideoGrabberVB61.TextOverlay_Top = 10
    VideoGrabberVB61.TextOverlay_Transparent = True
    VideoGrabberVB61.TextOverlay_FontColor = &HC0C0C0
```

```

VideoGrabberVB61.TextOverlay_Enabled = True

VideoGrabberVB61.TextOverlay_Selector = 1
VideoGrabberVB61.TextOverlay_String = "frame time: %time_100ns%"
VideoGrabberVB61.TextOverlay_Left = VideoGrabberVB61.VideoWidth - 120
VideoGrabberVB61.TextOverlay_Right = VideoGrabberVB61.VideoWidth
VideoGrabberVB61.TextOverlay_Top = 60
VideoGrabberVB61.TextOverlay_Transparent = False
VideoGrabberVB61.TextOverlay_BkColor = &H0
VideoGrabberVB61.TextOverlay_FontColor = &HF0B0C0
VideoGrabberVB61.TextOverlay_Enabled = True
End Sub

```

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#)
[AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#)
[MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#)
[OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#)
[SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#)
[SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#)
[SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#)
[SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#)
[SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#)
[SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#)
[SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#)
[SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#)
[SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#)
[SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#)
[SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#)
[ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#)
[TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#)
[TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#)
[TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#) [TextOverlay](#) [ShadowDirection](#)
[TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Revolutionize Your CHM Help File Output with HelpNDoc

TextOverlay_Shadow

TVideoGrabber.TextOverlay_Shadow

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables the shadow for the current text overlay selected by [TextOverlay_Selector](#)

Declaration

property TextOverlay_Shadow: Boolean **read** GetTextOverlayShadow **write** SetTextOverlaShadow **default** DEF_TextOverlay_Shadow;

__property **bool** TextOverlay_Shadow=read=GetTextOverlayShadow, write=SetTextOverlayShadow, **default=0**

Property TextOverlay_Shadow As Boolean

Description

Enables the shadow under the text overlay.

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#)
[AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#)
[MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronone](#) [OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#)
[OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#)
[SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#)
[SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#)
[SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#)
[SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#)
[SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#)
[SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#)
[SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#)
[SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#)
[SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#)
[SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#)
[SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#)
[ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#)
[TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#)
[TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#)
[TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [ShadowColor](#) [TextOverlay](#) [ShadowDirection](#)
[TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Maximize Your Documentation Output with HelpNDoc's
 Advanced Project Analyzer

TextOverlay_ShadowColor

TVideoGrabber.TextOverlay_ShadowColor

[Prev](#)[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the color of the shadow for the current text overlay selected by [TextOverlay_Selector](#)

Declaration

property TextOverlay_ShadowColor: TColor **read** GetTextOverlayShadowColor **write** SetTextOverlayShadowColor **default** DEF_TextOverlay_ShadowColor;

__property Graphics::TColor TextOverlay_ShadowColor=read=GetTextOverlayShadowColor,
 write=SetTextOverlayShadowColor, **default**=0

Property TextOverlay_ShadowColor As OLE_COLOR

Description

Specifies the color of the shadow under the text overlay when [TextOverlay_Shadow](#) is enabled.

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#)
[AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#)
[MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronone](#) [OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#)
[OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#)
[SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#)
[SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#)

[SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#)
[SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#)
[SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#)
[SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay_Align](#) [SetTextOverlay_BkColor](#) [SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#)
[SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#) [SetTextOverlay_GradientColor](#)
[SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#)
[SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#)
[SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#)
[SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#)
[ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#)
[TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#)
[TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#)
[TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowDirection](#)
[TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Experience the power of a responsive website for your documentation](#)

TextOverlay_ShadowDirection

TVideoGrabber.TextOverlay_ShadowDirection

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the direction of the shadow for the current text overlay selected by [TextOverlay_Selector](#)

Declaration

property TextOverlay_ShadowColor: TCardinalDirection **read** GetTextOverlayShadowColor **write** SetTextOverlayShadowColor **default** DEF_TextOverlay_ShadowColor;

__property TCardinalDirection TextOverlay_ShadowColor=read=GetTextOverlayShadowColor, write=SetTextOverlayShadowColor, **default**=0

Property TextOverlay_ShadowColor As TCardinalDirection

Description

Specifies the direction of the shadow under the text overlay when [TextOverlay_Shadow](#) is enabled.

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#)
[AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay_StretchToVideoSize](#)
[MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronon](#) [OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#)
[OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay_AlphaBlend](#)
[SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#)
[SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#)
[SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#)
[SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#)
[SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#)
[SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay_Align](#) [SetTextOverlay_BkColor](#) [SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#)
[SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#) [SetTextOverlay_GradientColor](#)
[SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#)
[SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#)
[SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#)

[SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay_BkColor](#) [TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#) [TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#) [TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Full-featured Help generator](#)

TextOverlay_String

TVideoGrabber.TextOverlay_String

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Sets or retrieves the text string that will be drawn over video frames for the current text overlay selected by [TextOverlay_Selector](#)

Declaration

property TextOverlay_String: **string read** GetTextOverlaystring **write** SetTextOverlaystring;

__property wchar_t *TextOverlay_String=read=GetTextOverlayString, write=SetTextOverlayString

Property TextOverlay_String As String

Description

Used to set or retrieve the text string that will be drawn over video frames.

The text string can be composed of several lines.

This property can be modified at any time, even if the text is currently drawn over a video frame.

Several TVideoGrabber variables can be used within the text string, delimited by a percent symbol **(2)**. When TVideoGrabber detects one of these variables the variable label is replaced by its current value:

"%sys_time[dd/mm/yy hh:nn:ss]%" : current system date/time **(1)**
 "%dv_time[dd/mm/yy hh:nn:ss]%" : current date/time stored on the DV VCR tape **(1)**
 "%time_code%" : current DV VCR time code, if available
 "%frame_count%" : number of the current frame
 "%time_full%" : time of the current frame in hh:mm:ss:cc format
 "%time_sec%" : time of the current frame, in seconds with 2 decimals
 "%time_100ns%" : time of the current frame, in 100 nano-seconds units
 "%custom0% to %custom9%" : up to 10 custom variables that can be set by using [SetTextOverlayCustomVar](#) .

(1) any valid [date/time format](#) is accepted between the brackets of sys_time and dv_time.

(2) the percent symbol is a reserved character. If you need to display the percent symbol itself, just duplicate it, e.g. TextOverlay_String = "the percent symbol is %%"

Note: the [frame grabber](#) must be enabled to use TextOverlay properties.

E.g., to initialize the 2 text overlay string, that will use custom variables:

```
...
VideoGrabber.TextOverlay_Selector = 0
VideoGrabber.TextOverlay_String = "my first text uses %custom0%"
VideoGrabber.TextOverlay_Left = 10
VideoGrabber.TextOverlay_Top = 10
VideoGrabber.TextOverlay_Enabled = true

VideoGrabber.TextOverlay_Selector = 1
VideoGrabber.TextOverlay_String = "my second text uses %custom3% and %custom4%"
VideoGrabber.TextOverlay_Left = 20
```

```
VideoGrabber.TextOverlay_Top = 100
VideoGrabber.TextOverlay_Enabled = true

VideoGrabber.StartPreview
...
```

then to update the custom variables of the 1st text overlay string:

```
...
VideoGrabber.SetTextOverlayCustomVar (0, 0, "value 0")
...
```

and of the 2nd text overlay string:

```
...
VideoGrabber.SetTextOverlayCustomVar (1, 3, "value 3")
VideoGrabber.SetTextOverlayCustomVar (1, 4, "value 4")
...
```

See Also

[TCardinalDirection](#)
[TOnFrameOverlayUsingDC](#)
[TOnFrameOverlayUsingDIB](#)
[TTextOverlayAlign](#)
[AdjustOverlayAspectRatio](#)
[DrawBitmapOverFrame](#)
[GetFrameInfoString](#)
[ImageOverlay](#)
[StretchToVideoSize](#)
[MouseWheelEventEnabled](#)
[OnFrameBitmap](#)
[OnFrameBitmapEventSynchronise](#)
[OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingDIB](#)
[OnFrameOverlayUsingVIDEOHDR](#)
[OnMouseDown](#)
[OnMouseMove](#)
[OnMouseUp](#)
[OnMouseWheel](#)
[OverlayAfterTransform](#)
[RefreshPlayerOverlays](#)
[SetImageOverlay](#)
[AlphaBlend](#)
[SetImageOverlay](#)
[AlphaBlendValue](#)
[SetImageOverlay](#)
[ChromaKey](#)
[SetImageOverlay](#)
[ChromaKeyLeewayPercent](#)
[SetImageOverlay](#)
[ChromaKeyRGBColor](#)
[SetImageOverlay](#)
[Enabled](#)
[SetImageOverlay](#)
[Height](#)
[SetImageOverlay](#)
[LeftLocation](#)
[SetImageOverlay](#)
[RotationAngle](#)
[SetImageOverlay](#)
[StretchToVideoSize](#)
[SetImageOverlay](#)
[TargetDisplay](#)
[SetImageOverlay](#)
[TopLocation](#)
[SetImageOverlay](#)
[Transparent](#)
[SetImageOverlay](#)
[TransparentColorValue](#)
[SetImageOverlay](#)
[UseTransparentColor](#)
[SetImageOverlay](#)
[Width](#)
[SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#)
[SetImageOverlayFromHBitmap2](#)
[SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#)
[SetImageOverlayFromJPEGFile](#)
[SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#)
[SetImageOverlayFromTImage](#)
[SetImageOverlayFromTImage2](#)
[SetTextOverlay](#)
[Align](#)
[SetTextOverlay](#)
[BkColor](#)
[SetTextOverlay](#)
[CustomVar](#)
[SetTextOverlay](#)
[Enabled](#)
[SetTextOverlay](#)
[Font](#)
[SetTextOverlay](#)
[FontColor](#)
[SetTextOverlay](#)
[GradientColor](#)
[SetTextOverlay](#)
[GradientMode](#)
[SetTextOverlay](#)
[HighResFont](#)
[SetTextOverlay](#)
[Left](#)
[SetTextOverlay](#)
[Right](#)
[SetTextOverlay](#)
[Scrolling](#)
[SetTextOverlay](#)
[ScrollingSpeed](#)
[SetTextOverlay](#)
[Shadow](#)
[SetTextOverlay](#)
[ShadowColor](#)
[SetTextOverlay](#)
[ShadowDirection](#)
[SetTextOverlay](#)
[String](#)
[SetTextOverlay](#)
[TargetDisplay](#)
[SetTextOverlay](#)
[Top](#)
[SetTextOverlay](#)
[Transparent](#)
[ShapeOverlay](#)
[ShapeOverlayEnabled](#)
[ShapeOverlayList](#)
[TextOverlay](#)
[Align](#)
[TextOverlay](#)
[BkColor](#)
[TextOverlay](#)
[CreateCustomFont](#)
[TextOverlay](#)
[CreateCustomFont2](#)
[TextOverlay](#)
[Enabled](#)
[TextOverlay](#)
[Font](#)
[TextOverlay](#)
[FontColor](#)
[TextOverlay](#)
[Left](#)
[TextOverlay](#)
[Right](#)
[TextOverlay](#)
[Scrolling](#)
[TextOverlay](#)
[ScrollingSpeed](#)
[TextOverlay](#)
[Selector](#)
[TextOverlay](#)
[Shadow](#)
[TextOverlay](#)
[ShadowColor](#)
[TextOverlay](#)
[ShadowDirection](#)
[TextOverlay](#)
[Top](#)
[TextOverlay](#)
[Transparent](#)
[TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Streamline Your Documentation Process with HelpNDoc's Project Analyzer

TextOverlay_Top

TVideoGrabber.TextOverlay_Top

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Top position (in pixels) where the text will be drawn over video frames for the current text overlay selected by [TextOverlay_Selector](#)

Declaration

property TextOverlay_Top: LongInt **read** GetTextOverlayTop **write** SetTextOverlayTop **default** DEF_TextOverlay_Top;

__property **int** TextOverlay_Top=read=GetTextOverlayTop, write=SetTextOverlayTop, **default**=0

Property TextOverlay_Top As Long

Description

Used to set or retrieve the top position (in pixels) where the text will be drawn over video frames.

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronon](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#) [SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#) [SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#) [SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#) [SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#) [TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#) [TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#) [TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#) [TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Make Help Documentation a Breeze with a Help Authoring Tool

TextOverlay_Transparent

TVideoGrabber.TextOverlay_Transparent

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables/disables transparency of text drawn over video frames for the current text overlay selected by [TextOverlay_Selector](#)

Declaration

property TextOverlay_Transparent: Boolean **read** GetTextOverlayTransparent **write** SetTextOverlayTransparent **default** DEF_TextOverlay_Transparent;

__property **bool** TextOverlay_Transparent=read=GetTextOverlayTransparent, write=SetTextOverlayTransparent, **default**=1

Property TextOverlay_Top As Long

Description

Used to enable or disable transparency of text drawn over video frames.
If enabled, [TextOverlay_BkColor](#) is ignored.

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchrone](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#) [SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#) [SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#) [SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#) [SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#) [TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#) [TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#) [TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#) [TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Generate EPub eBooks with ease](#)

TextOverlay_VideoAlignment

TVideoGrabber.TextOverlay_VideoAlignment

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Relative alignment of the text overlay

Declaration

property TextOverlay_VideoAlignment: TVideoAlignment **read** GetTextOverlayVideoAlignment **write** SetTextOverlayVideoAlignment;

__property TVideoAlignment TextOverlay_VideoAlignment==GetTextOverlayVideoAlignment, write=SetTextOverlayVideoAlignment, **nodefault**;

Description

[TVideoAlignment](#) relative alignment of the text overlay within the video frame.

Default: oa_LeftTop

See Also

[GetTextOverlay](#) [AlphaBlend](#) [GetTextOverlay](#) [AlphaBlendValue](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [AlphaBlend](#) [SetTextOverlay](#) [AlphaBlendValue](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#) [SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#) [SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#) [SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#) [SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#)

Created with the Standard Edition of HelpNDoc: [Streamline your documentation process with HelpNDoc's WinHelp HLP to CHM conversion feature](#)

ThirdPartyDeinterlacer

TVideoGrabber.ThirdPartyDeinterlacer

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Properties](#)

Specifies the deinterlacer filter to use

Declaration

property ThirdPartyDeinterlacer: **string** **read** GetThirdPartyDeinterlacer **write** SetThirdPartyDeinterlacer;

__property wchar_t *ThirdPartyDeinterlacer=read=GetThirdPartyDeinterlacer,
write=SetThirdPartyDeinterlacer

Property ThirdPartyDeinterlacer As String

Description

Specifies the deinterlacer filter to be used with interlaced sources.

You can find

Set [VideoProcessing_Deinterlacing](#) = di_ThirdPartyDeinterlacer to activate it.

Created with the Standard Edition of HelpNDoc: [Effortlessly Edit and Export Markdown Documents](#)

TranslateMouseCoordinates

TVideoGrabber.TranslateMouseCoordinates

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Properties](#)

Converts mouse coordinates of [OnMouseMove](#) , [OnMouseDown](#) and [OnMouseUp](#) into their corresponding native video coordinates.

Declaration

property TranslateMouseCoordinates: Boolean **read** GetTranslateMouseCoordinates **write** SetTranslateMouseCoordinates **default** DEF_TranslateMouseCoordinates;

__property **bool** TranslateMouseCoordinates=read=GetTranslateMouseCoordinates,
write=SetTranslateMouseCoordinates, **default**=1

Property TranslateMouseCoordinates As Boolean

Description

This property is used when drawing over video frames using mouse coordinates returned by mouse events and the [preview zoom size](#) is not 100%.

- when this property is disabled, the mouse events return the zoomed coordinates, that do not match the native coordinates necessary to draw over frames from the [OnFrameOverlayUsingDC](#) event.
- when this property is enabled, [OnMouseMove](#) , [OnMouseDown](#) and [OnMouseUp](#) always return the native video coordinates, whatever the preview zoom size.

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#)
[AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#)
[MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#)
[OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#)
[SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#)
[SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#)
[SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#)

[SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#) [SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#) [SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay_Align](#) [SetTextOverlay_BkColor](#) [SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#) [SetTextOverlay_GradientColor](#) [SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#) [SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#) [SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#) [SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#) [TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#) [TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#) [TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#) [TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#)

Created with the Standard Edition of HelpNDoc: Don't be left in the past: convert your WinHelp HLP help files to CHM with HelpNDoc

TunerFrequency

TVideoGrabber.TunerFrequency

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Sets/retrieve the TV frequency

Declaration

property TunerFrequency: LongInt **read** GetTunerFrequency **write** SetTunerFrequency **default** DEF_TunerFrequency;

__property int TunerFrequency=read=GetTunerFrequency, write=SetTunerFrequency, default=-1

Property TunerFrequency as Long

Description

Used to set / retrieve the TV frequency

See Also

[TTVChannelInfo](#) [TOnTVChannelScanStarted](#) [TOnTVChannelSelected](#) [TTunerInputType](#) [GetTVChannelInfo](#) [IsHorizontalSyncLocked](#) [IsTVAudioAvailable](#) [IsTVAutoTuneRunning](#) [IsTVTunerAvailable](#) [OnTVChannelScanCompleted](#) [OnTVChannelScanStarted](#) [OnTVChannelSelected](#) [TunerMode](#) [TVChannel](#) [TVClearFrequencyOverrides](#) [TVCountryCode](#) [TVGetMinMaxChannels](#) [TVSetChannelFrequencyOverride](#) [TVStartAutoScan](#) [TVStopAutoScan](#) [TVTunerInputType](#) [TVUseFrequencyOverrides](#) [VCRHorizontalLocking](#)

Created with the Standard Edition of HelpNDoc: Easy to use tool to create HTML Help files and Help web sites

TunerMode

TVideoGrabber.TunerMode

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Selects the mode of the tuner.

Declaration

property TunerMode: TTunerMode **read** GetTunerMode **write** SetTunerMode **default** DEF_TunerMode;

TTunerMode TunerMode==GetTunerMode, write=SetTunerMode, **default=0**;

property TunerMode as TxTunerMode

Description

Used to select the mode of the tuner.

The possible values are:

TV: **tm_TV**Tuner (default)
 FM radio: **tm_FMR**adioTuner
 AM radio: **tm_AMR**adioTuner
 Digital satellite: **tm_Digital**SatelliteTuner

See Also

[TTVChannelInfo](#) [TOnTVChannelScanStarted](#) [TOnTVChannelSelected](#) [TTunerInputType](#) [GetTVChannelInfo](#)
[IsHorizontalSyncLocked](#) [IsTVAudioAvailable](#) [IsTVAutoTuneRunning](#) [IsTVTunerAvailable](#)
[OnTVChannelScanCompleted](#) [OnTVChannelScanStarted](#) [OnTVChannelSelected](#) [TunerFrequency](#)
[TVChannel](#) [TVClearFrequencyOverrides](#) [TVCountryCode](#) [TVGetMinMaxChannels](#)
[TVSetChannelFrequencyOverride](#) [TVStartAutoScan](#) [TVStopAutoScan](#) [TVTunerInputType](#)
[TVUseFrequencyOverrides](#) [VCRHorizontalLocking](#)

Created with the Standard Edition of HelpNDoc: [Free CHM Help documentation generator](#)

TVChannel

TVideoGrabber.TVChannel

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Sets or retrieves the current TV channel.

Declaration

property TVChannel: LongInt **read** GetTVChannel **write** SetTVChannel;

___property **int** TVChannel=read=GetTVChannel, write=SetTVChannel, **nodefault**

Property TVChannel As Long

Description

Used to set/retrieve the current TV channel.

Be sure to select the proper country code ([TVCountryCode](#)) and tuner input type ([TVTunerInputType](#)) before.

See Also

[TTVChannelInfo](#) [TOnTVChannelScanStarted](#) [TOnTVChannelSelected](#) [TTunerInputType](#) [GetTVChannelInfo](#)
[IsHorizontalSyncLocked](#) [IsTVAudioAvailable](#) [IsTVAutoTuneRunning](#) [IsTVTunerAvailable](#)
[OnTVChannelScanCompleted](#) [OnTVChannelScanStarted](#) [OnTVChannelSelected](#) [TunerFrequency](#)
[TunerMode](#) [TVClearFrequencyOverrides](#) [TVCountryCode](#) [TVGetMinMaxChannels](#)
[TVSetChannelFrequencyOverride](#) [TVStartAutoScan](#) [TVStopAutoScan](#) [TVTunerInputType](#)
[TVUseFrequencyOverrides](#) [VCRHorizontalLocking](#)

Created with the Standard Edition of HelpNDoc: [Experience the Power and Ease of Use of a Help Authoring Tool](#)

TVCountryCode

TVideoGrabber.TVCountryCode

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Sets / retrieves the current TV country code.

Declaration

property TVCountryCode: LongInt **read** GetTVCountryCode **write** SetTVCountryCode;

__property int TVCountryCode=read=GetTVCountryCode, write=SetTVCountryCode, **nodefault**

Property TVCountryCode As Long

Description

Used to set / retrieve the current TV country code in which the [TV channels](#) retrieve their related TV frequencies, depending of the country.

Therefore be sure to select the proper country code before selecting a [TV channel](#).

See Also

[TTVChannelInfo](#) [TOnTVChannelScanStarted](#) [TOnTVChannelSelected](#) [TTunerInputType](#) [GetTVChannelInfo](#) [IsHorizontalSyncLocked](#) [IsTVAudioAvailable](#) [IsTVAutoTuneRunning](#) [IsTVTunerAvailable](#) [OnTVChannelScanCompleted](#) [OnTVChannelScanStarted](#) [OnTVChannelSelected](#) [TunerFrequency](#) [TunerMode](#) [TVChannel](#) [TVClearFrequencyOverrides](#) [TVGetMinMaxChannels](#) [TVSetChannelFrequencyOverride](#) [TVStartAutoScan](#) [TVStopAutoScan](#) [TVTunerInputType](#) [TVUseFrequencyOverrides](#) [VCRHorizontalLocking](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Reach: Convert Your Word Document to an ePub or Kindle eBook](#)

TVTunerInputType

TVideoGrabber.TVTunerInputType

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Sets / retrieves the current TV tuner input type.

Declaration

property TVTunerInputType: TTunerInputType **read** GetTVTunerInputType **write** SetTVTunerInputType;

__property TTunerInputType TVTunerInputType=read=GetTVTunerInputType, write=SetTVTunerInputType, **nodefault**

Property TVTunerInputType As TxTunerInputType

Description

Used to set / retrieve the current TV tuner input type (antenna or cable).

Be sure to set it before selecting a [TV channel](#).

See Also

[TTVChannelInfo](#) [TOnTVChannelScanStarted](#) [TOnTVChannelSelected](#) [TTunerInputType](#) [GetTVChannelInfo](#) [IsHorizontalSyncLocked](#) [IsTVAudioAvailable](#) [IsTVAutoTuneRunning](#) [IsTVTunerAvailable](#) [OnTVChannelScanCompleted](#) [OnTVChannelScanStarted](#) [OnTVChannelSelected](#) [TunerFrequency](#) [TunerMode](#) [TVChannel](#) [TVClearFrequencyOverrides](#) [TVCountryCode](#) [TVGetMinMaxChannels](#) [TVSetChannelFrequencyOverride](#) [TVStartAutoScan](#) [TVStopAutoScan](#) [TVUseFrequencyOverrides](#) [VCRHorizontalLocking](#)

Created with the Standard Edition of HelpNDoc: [Experience the Power and Simplicity of HelpNDoc's User Interface](#)

TVUseFrequencyOverrides

TVideoGrabber.TVUseFrequencyOverrides

[Prev](#)

[Next](#)

TVideoGrabber Properties

Enables / disables frequency overrides.

Declaration

property TVUseFrequencyOverrides: Boolean **read** GetTVUseFrequencyOverrides **write** SetTVUseFrequencyOverrides **default** DEF_TVUseFrequencyOverrides;

__property **bool** TVUseFrequencyOverrides=read=GetTVUseFrequencyOverrides, write=SetTVUseFrequencyOverrides, **default**=0

Property TVUseFrequencyOverrides As Boolean

Description

Used to enable / disable frequency overrides that could have been set on TV channels with [TVSetChannelFrequencyOverride](#) .

See Also

[TTVChannelInfo](#) [TOnTVChannelScanStarted](#) [TOnTVChannelSelected](#) [TTunerInputType](#) [GetTVChannelInfo](#) [IsHorizontalSyncLocked](#) [IsTVAudioAvailable](#) [IsTVAutoTuneRunning](#) [IsTVTunerAvailable](#) [OnTVChannelScanCompleted](#) [OnTVChannelScanStarted](#) [OnTVChannelSelected](#) [TunerFrequency](#) [TunerMode](#) [TVChannel](#) [TVClearFrequencyOverrides](#) [TVCountryCode](#) [TVGetMinMaxChannels](#) [TVSetChannelFrequencyOverride](#) [TVStartAutoScan](#) [TVStopAutoScan](#) [TVTunerInputType](#) [VCRHorizontalLocking](#)

Created with the Standard Edition of HelpNDoc: [Enhance Your Documentation with HelpNDoc's Advanced Project Analyzer](#)

UniqueID

TVideoGrabber.UniqueID

[Prev](#)
[Next](#)

TVideoGrabber Properties

Unique ID of the component.

Declaration

property UniqueID: LongInt **read** GetUniqueID **write** SetUniqueID;

__property **int** UniqueID=read=GetUniqueID, write=SetUniqueID, **nodefault**

Property UniqueID As Long

Description

Unique ID of a TVideoGrabber component in the app.

Used to associate the master component to the slaves components in the master/slave [multiplexed inputs mode](#).

See Also

[TMultiplexedRole](#) [AssociateMultiplexedSlave](#) [EnableMultiplexedInput](#) [MultiplexedInputEmulation](#) [MultiplexedRole](#) [MultiplexedStabilizationDelay](#) [MultiplexedSwitchDelay](#)

Created with the Standard Edition of HelpNDoc: [Write eBooks for the Kindle](#)

UseClock

TVideoGrabber.UseClock

[Prev](#)
[Next](#)

TVideoGrabber Properties

Let the player graph run at the maximal speed

Declaration

property UseClock: Boolean **read** GetUseClock **write** SetUseClock;

__property bool UseClock=read=GetUseClock, write=SetUseClock, **nodefault**

Property UseClock As Boolean

Description

The speed of the player graph is under control of the renderer (video renderer, audio renderer, streaming, etc).

Disabling UseClock lets the player run at maximal speed.

This is used mainly e.g. to play a clip at the processing speed, e.g. to analyse the video frames without dropping any of them because of the player speed.

Created with the Standard Edition of HelpNDoc: [Write EPub books for the iPad](#)

v360_AspectRatio

TVideoGrabber.v360_AspectRatio

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

360° video: aspect ratio of the decoded 360 video

Declaration

property v360_AspectRatio: Double **read** Getv360_AspectRatio **write** Setv360_AspectRatio **default** DEF_v360_AspectRatio;

__property double v360_AspectRatio==Getv360_AspectRatio, write=Setv360_AspectRatio, **default**=0;

Description

By default the aspect ratio is calculated automatically when v360_AspectRatio = -1.0

However it is possible to force an aspect ratio by setting a non-default value, e.g.:

- for an aspect ratio of 2:1 (2/1), set v360_AspectRatio = 2.0
- for an aspect ratio of 16:9 (16/9), set v360_AspectRatio = 1.7777778

See Also

[v360_AddYawPitchRoll](#) [v360_Enabled](#) [v360_GetAngle](#) [v360_GetYawPitchRoll](#) [v360_MasterAngle](#) [v360_MouseAction](#) [v360_MouseActionPercent](#) [v360_ResetAnglesToDefault](#) [v360_SetAngle](#) [v360_SetInterpolation](#) [v360_SetProjection](#) [v360_SetStereoFormat](#) [v360_SetTranspose](#) [v360_SetYawPitchRoll](#)

Created with the Standard Edition of HelpNDoc: [Converting Word Documents to eBooks: A Step-by-Step Guide with HelpNDoc](#)

v360_Enabled

TVideoGrabber.v360_Enabled

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

360° video: enables the video 360 decoding

Declaration

property v360_Enabled: Boolean **read** Getv360_Enabled **write** Setv360_Enabled **default** DEF_v360_Enabled;

__property bool v360_Enabled==Getv360_Enabled, write=Setv360_Enabled, **default=0**;

Description

Enables the video 360 decoding
Applies only to the player mode

See Also

[v360_AddYawPitchRoll](#) [v360_AspectRatio](#) [v360_GetAngle](#) [v360_GetYawPitchRoll](#) [v360_MasterAngle](#) [v360_MouseAction](#) [v360_MouseActionPercent](#) [v360_ResetAnglesToDefault](#) [v360_SetAngle](#) [v360_SetInterpolation](#) [v360_SetProjection](#) [v360_SetStereoFormat](#) [v360_SetTranspose](#) [v360_SetYawPitchRoll](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly optimize your documentation website for search engines](#)

v360_MasterAngle

TVideoGrabber.v360_MasterAngle

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

360° video: specifies the master angle

Declaration

property v360_MasterAngle: Tv360_Angle **read** Getv360_MasterAngle **write** Setv360_MasterAngle **default** DEF_v360_MasterAngle;

__property Tv360_Angle v360_MasterAngle==Getv360_MasterAngle, write=Setv360_MasterAngle, **default=1**;

Description

Specifies the [Tv360_Angle](#) angle (horizontal or vertical) from which is calculated automatically the opposite angle (vertical or horizontal) when the aspect ratio is not specified (when [AspectRatioToUse](#) = -1)
In other words, if v360_MasterAngle = v360_fov_Horizontal (which is the default), v360_fov_Vertical is calculated automatically according to the aspect ratio of the video clip.

Note: to control the aspect ratio yourself, set v360_AspectRatio with a value > 0.

E.g. for an aspect ratio of 2:1, set v360_AspectRatio = 0.5

See Also

[v360_AddYawPitchRoll](#) [v360_AspectRatio](#) [v360_Enabled](#) [v360_GetAngle](#) [v360_GetYawPitchRoll](#) [v360_MouseAction](#) [v360_MouseActionPercent](#) [v360_ResetAnglesToDefault](#) [v360_SetAngle](#) [v360_SetInterpolation](#) [v360_SetProjection](#) [v360_SetStereoFormat](#) [v360_SetTranspose](#) [v360_SetYawPitchRoll](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Documentation Capabilities with HelpNDoc's User-Friendly UI](#)

v360_MouseAction

TVideoGrabber.v360_MouseAction

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

360° video: specifies how the mouse can control the motion in the 360 video

Declaration

property v360_MouseAction: TV360_MouseAction **read** Getv360_MouseAction **write** Setv360_MouseAction **default** DEF_v360_MouseAction;

__property TV360_MouseAction v360_MouseAction==Getv360_MouseAction, write=Setv360_MouseAction, **default=2**;

Description

When enabled:

- the motion of the mouse controls the point of view (horizontal / vertical)
- the mouse wheel controls the output angles (zoom)

[TV360_MouseAction](#) possible values:

ma_Disabled: the mouse does not control the point of view.

ma_MouseUp: the point of view is updated when releasing the mouse button

ma_MouseMove: the point of view is updated while the mouse is moving

Note that the point of view can also be updated programmatically with [v360_SetYawPitchRoll](#) or [v360_AddYawPitchRoll](#).

See Also

[v360_AddYawPitchRoll](#) [v360_AspectRatio](#) [v360_Enabled](#) [v360_GetAngle](#) [v360_GetYawPitchRoll](#)
[v360_MasterAngle](#) [v360_MouseActionPercent](#) [v360_ResetAnglesToDefault](#) [v360_SetAngle](#)
[v360_SetInterpolation](#) [v360_SetProjection](#) [v360_SetStereoFormat](#) [v360_SetTranspose](#)
[v360_SetYawPitchRoll](#)

Created with the Standard Edition of HelpNDoc: [Free CHM Help documentation generator](#)

v360_MouseActionPercent

TVideoGrabber.v360_MouseActionPercent

[Prev](#)[Next](#)

[TVideoGrabber](#) [Properties](#)

360° video: percentage of displacement

Declaration

property v360_MouseActionPercent: LongInt **read** Getv360_MouseActionPercent **write** Setv360_MouseActionPercent **default** DEF_v360_MouseActionPercent;

__property int v360_MouseActionPercent==Getv360_MouseActionPercent,
 write=Setv360_MouseActionPercent, **default**=10;

Description

Specifies the percentage of displacement of the point view, depending on the mouse movment
 Default: 10%

See Also

[v360_AddYawPitchRoll](#) [v360_AspectRatio](#) [v360_Enabled](#) [v360_GetAngle](#) [v360_GetYawPitchRoll](#)
[v360_MasterAngle](#) [v360_MouseAction](#) [v360_ResetAnglesToDefault](#) [v360_SetAngle](#) [v360_SetInterpolation](#)
[v360_SetProjection](#) [v360_SetStereoFormat](#) [v360_SetTranspose](#) [v360_SetYawPitchRoll](#)

Created with the Standard Edition of HelpNDoc: [Ensure High-Quality Documentation with HelpNDoc's Hyperlink and Library Item Reports](#)

VCRHorizontalLocking

TVideoGrabber.VCRHorizontalLocking

[Prev](#)[Next](#)

[TVideoGrabber](#) [Properties](#)

Sets or retrieves whether the video is from a broadcast source or from a tape source.

Declaration

property VCRHorizontalLocking: Boolean **read** GetVCRHorizontalLocking **write** SetVCRHorizontalLocking;

__property bool VCRHorizontalLocking=read=GetVCRHorizontalLocking, write=SetVCRHorizontalLocking,
nodefault

Property VCRHorizontalLocking As Boolean

Description

Used to set or retrieve whether the video is from a broadcast source or from a tape source.

false: the video is from a broadcast source,

true: the video is from a tape source.

See Also

[TDVCommand](#) [IsDVCommandAvailable](#) [IsHorizontalSyncLocked](#) [IsTimeCodeReaderAvailable](#)
[SendDVCommand](#)

Created with the Standard Edition of HelpNDoc: [Produce online help for Qt applications](#)

Version

TVideoGrabber.Version

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Current version of TVideoGrabber.

Declaration

property Version: **string read** GetVersion **write** SetVersion **stored** false;

__property wchar_t *Version=read=GetVersion, write=SetVersion, stored=**false**

Property Version As String

Description

Used to retrieve the current version of TVideoGrabber.

This property is published and visible at design time from the Object Inspector.

Created with the Standard Edition of HelpNDoc: [Free EPub producer](#)

VideoCompression_DataRate

TVideoGrabber.VideoCompression_DataRate

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the data rate that should be used for video compression.

Declaration

property VideoCompression_DataRate: LongInt **read** GetVideoCompression_DataRate **write** SetVideoCompression_DataRate **default** DEF_VideoCompression_DataRate;

__property int VideoCompression_DataRate==GetVideoCompression_DataRate, write=SetVideoCompression_DataRate, **default**=-1;

Description

Used to Specify the data rate that should be used for video compression.

If the value specified is -1, the the default data rate of the codec will be used.

Use the [OnVideoCompressionSettings](#) event to know the default value for the current codec (see the [Software compression by using codecs](#) chapter for more explanations).

Created with the Standard Edition of HelpNDoc: [Make Documentation Review a Breeze with HelpNDoc's Advanced Project Analyzer](#)

VideoCompression_KeyFrameRate

TVideoGrabber.VideoCompression_KeyFrameRate

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the key-frame rate used for the [current video compressor](#).

Declaration

property VideoCompression_KeyFrameRate: LongInt **read** GetVideoCompression_KeyFrameRate **write** SetVideoCompression_KeyFrameRate **default** DEF_VideoCompression_KeyFrameRate;

__property **int** VideoCompression_KeyFrameRate=read=GetVideoCompression_KeyFrameRate, write=SetVideoCompression_KeyFrameRate, **default**=15

Property VideoCompression_KeyFrameRate As Long

Description

Specifies the desired key-frame rate for the [current video compressor](#).

If the value specified is -1, the filter will use the default key-frame rate. If the value is zero, only the first frame will be a key frame. If the value is negative (-1), the filter will use its default key-frame rate.

Use the [OnVideoCompressionSettings](#) event to know the default value for the current codec (see the [Software compression by using codecs](#) chapter for more explanations).

See Also

[Recording methods and properties](#) [TCompressionType](#) [TOnVideoCompressionSettings](#) [AudioCompressor](#) [AudioCompressorIndex](#) [AudioCompressorName](#) [AudioCompressors](#) [AudioCompressorsCount](#) [CompressionMode](#) [CompressionType](#) [GetVideoCompressionSettings](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [OnVideoCompressionSettings](#) [RefreshDevicesAndCompressorsLists](#) [SaveCompressorSettingsToDataString](#) [SetVideoCompressionDefaults](#) [SetVideoCompressionSettings](#) [VideoCompression](#) [PFramesPerKeyFrame](#) [VideoCompression](#) [Quality](#) [VideoCompression](#) [WindowSize](#) [VideoCompressor](#) [VideoCompressorIndex](#) [VideoCompressorName](#) [VideoCompressors](#) [VideoCompressorsCount](#)

Created with the Standard Edition of HelpNDoc: [Elevate Your Documentation Process with HelpNDoc's Advanced Features](#)

VideoCompression_PFramesPerKeyFrame

TVideoGrabber.VideoCompression_PFramesPerKeyFrame

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the rate of the predicted frames per key frame for the [current video compressor](#).

Declaration

property VideoCompression_PFramesPerKeyFrame: LongInt **read** GetVideoCompression_PFramesPerKeyFrame **write** SetVideoCompression_PFramesPerKeyFrame **default** DEF_VideoCompression_PFramesPerKeyFrame;

__property **int** VideoCompression_PFramesPerKeyFrame=read=GetVideoCompression_PFramesPerKeyFrame, write=SetVideoCompression_PFramesPerKeyFrame, **default**=0

Property VideoCompression_PFramesPerKeyFrame As Long

Description

Specifies the rate of predicted (P) frames per key frame for the [current video compressor](#). If the value is negative, the filter will use the default rate. If the value is negative (-1), the filter will use its default value.

Use the [OnVideoCompressionSettings](#) event to know the default value for the current codec (see the [Software compression by using codecs](#) chapter for more explanations).

See Also

[Recording methods and properties](#) [TCompressionType](#) [TOnVideoCompressionSettings](#) [AudioCompressor](#) [AudioCompressorIndex](#) [AudioCompressorName](#) [AudioCompressors](#) [AudioCompressorsCount](#) [CompressionMode](#) [CompressionType](#) [GetVideoCompressionSettings](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [OnVideoCompressionSettings](#) [RefreshDevicesAndCompressorsLists](#) [SaveCompressorSettingsToDataString](#) [SetVideoCompressionDefaults](#) [SetVideoCompressionSettings](#) [VideoCompression](#) [KeyFrameRate](#) [VideoCompression](#) [Quality](#) [VideoCompression](#) [WindowSize](#) [VideoCompressor](#) [VideoCompressorIndex](#) [VideoCompressorName](#) [VideoCompressors](#) [VideoCompressorsCount](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Documentation Capabilities with a Help Authoring Tool](#)

VideoCompression_Quality

TVideoGrabber.VideoCompression_Quality

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the compression quality for the [current video compressor](#).

Declaration

property VideoCompression_Quality: Double **read** GetVideoCompression_Quality **write** SetVideoCompression_Quality;

__property **double** VideoCompression_Quality=read=GetVideoCompression_Quality, write=SetVideoCompression_Quality

Property VideoCompression_Quality As Double

Description

Specifies compression quality for the [current video compressor](#).

This property is expressed as a value between 0.0 and 1.0, where 1.0 indicates the best quality and 0.0 indicates the worst quality. If the value is negative (-1), the filter will use the default quality.

Use the [OnVideoCompressionSettings](#) event to know the default value for the current codec (see the [Software compression by using codecs](#) chapter for more explanations).

See Also

[Recording methods and properties](#) [TCompressionType](#) [TOnVideoCompressionSettings](#) [AudioCompressor](#) [AudioCompressorIndex](#) [AudioCompressorName](#) [AudioCompressors](#) [AudioCompressorsCount](#) [CompressionMode](#) [CompressionType](#) [GetVideoCompressionSettings](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [OnVideoCompressionSettings](#) [RefreshDevicesAndCompressorsLists](#) [SaveCompressorSettingsToDataString](#) [SetVideoCompressionDefaults](#) [SetVideoCompressionSettings](#) [VideoCompression](#) [KeyFrameRate](#) [VideoCompression](#) [PFramesPerKeyFrame](#) [VideoCompression](#) [WindowSize](#) [VideoCompressor](#) [VideoCompressorIndex](#) [VideoCompressorName](#) [VideoCompressors](#) [VideoCompressorsCount](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly create a professional-quality documentation website with HelpNDoc](#)

VideoCompression_WindowSize

TVideoGrabber.VideoCompression_WindowSize

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the average data rate for the [current video compressor](#).

Declaration

property VideoCompression_WindowSize: LongInt **read** GetVideoCompression_WindowSize **write** SetVideoCompression_WindowSize **default** DEF_VideoCompression_WindowSize;

__property **int** VideoCompression_WindowSize=read=GetVideoCompression_WindowSize, write=SetVideoCompression_WindowSize, **default**=- 1

Property VideoCompression_WindowSize As Long

Description

Specifies the average data rate for the [current video compressor](#).

This property sets the number of frames over which the compressor must maintain an average data rate. For example, assuming a data rate of 100K/sec and a frame rate of 10 frames per second, if the window size is 1, then every frame will be 10K or less. If the window size is 5, then every five consecutive frames must average 10K per frame, but individual frames may exceed this size.

Use the [OnVideoCompressionSettings](#) event to know the default value for the current codec (see the [Software compression by using codecs](#) chapter for more explanations).

See Also

[Recording methods and properties](#) [TCompressionType](#) [TOnVideoCompressionSettings](#) [AudioCompressor](#) [AudioCompressorIndex](#) [AudioCompressorName](#) [AudioCompressors](#) [AudioCompressorsCount](#) [CompressionMode](#) [CompressionType](#) [GetVideoCompressionSettings](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [OnVideoCompressionSettings](#) [RefreshDevicesAndCompressorsLists](#) [SaveCompressorSettingsToDataString](#) [SetVideoCompressionDefaults](#) [SetVideoCompressionSettings](#) [VideoCompression](#) [KeyFrameRate](#) [VideoCompression](#) [PFramesPerKeyFrame](#) [VideoCompression](#) [Quality](#) [VideoCompressor](#) [VideoCompressorIndex](#) [VideoCompressorName](#) [VideoCompressors](#) [VideoCompressorsCount](#)

Created with the Standard Edition of HelpNDoc: [Free Web Help generator](#)

VideoCompressor

TVideoGrabber.VideoCompressor

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Selects the current video compressor.

Declaration

property VideoCompressor: LongInt **read** GetVideoCompressor **write** SetVideoCompressor;

__property **int** VideoCompressor=read=GetVideoCompressor, write=SetVideoCompressor, **nodefault**

Property VideoCompressor As Long

Description

Used to select the current video compressor in the [VideoCompressors](#) list.

See Also

[Recording methods and properties](#) [TCompressionType](#) [TOnVideoCompressionSettings](#) [AudioCompressor](#) [AudioCompressorIndex](#) [AudioCompressorName](#) [AudioCompressors](#) [AudioCompressorsCount](#) [CompressionMode](#) [CompressionType](#) [GetVideoCompressionSettings](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [OnVideoCompressionSettings](#) [RefreshDevicesAndCompressorsLists](#) [SaveCompressorSettingsToDataString](#) [SetVideoCompressionDefaults](#) [SetVideoCompressionSettings](#) [VideoCompression](#) [KeyFrameRate](#) [VideoCompression](#) [PFramesPerKeyFrame](#) [VideoCompression](#) [Quality](#)

[VideoCompression](#) [WindowSize](#) [VideoCompressorIndex](#) [VideoCompressorName](#) [VideoCompressors](#) [VideoCompressorsCount](#)

Created with the Standard Edition of HelpNDoc: Effortlessly create a professional-quality documentation website with HelpNDoc

VideoCompressorName

TVideoGrabber.VideoCompressorName

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Name of the current video compressor.

Declaration

property VideoCompressorName: **string read** GetVideoCompressorName;

__property wchar_t *VideoCompressorName=read=GetVideoCompressorName

Property VideoCompressorName As String

Description

Used to retrieve the name of the current video compressor selected by [VideoCompressor](#) in the [VideoCompressors](#) list.

See Also

[Recording methods and properties](#) [TCompressionType](#) [TOnVideoCompressionSettings](#) [AudioCompressor](#) [AudioCompressorIndex](#) [AudioCompressorName](#) [AudioCompressors](#) [AudioCompressorsCount](#) [CompressionMode](#) [CompressionType](#) [GetVideoCompressionSettings](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [OnVideoCompressionSettings](#) [RefreshDevicesAndCompressorsLists](#) [SaveCompressorSettingsToDataString](#) [SetVideoCompressionDefaults](#) [SetVideoCompressionSettings](#) [VideoCompression](#) [KeyFrameRate](#) [VideoCompression](#) [PFramesPerKeyFrame](#) [VideoCompression](#) [Quality](#) [VideoCompression](#) [WindowSize](#) [VideoCompressor](#) [VideoCompressorIndex](#) [VideoCompressors](#) [VideoCompressorsCount](#)

Created with the Standard Edition of HelpNDoc: Save time and frustration with HelpNDoc's WinHelp HLP to CHM conversion feature

VideoCompressors

TVideoGrabber.VideoCompressors

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

List of the video compressors available on the current platform.

Declaration

function VideoCompressors: **string**;

__property wchar_t * __fastcall VideoCompressors();

property VideoCompressors as string

Description

Used to retrieve a string that contains the list of the video compressors available on the current platform. This list can be assigned to list based controls. E.g.:

ComboBox1.Items.Text := VideoGrabber1.VideoCompressors;

It is possible to retrieve programmatically the index of a video compressor by using the [FindIndexInListByName](#) function as follows:

```
VideoGrabber.VideoCompressor := VideoGrabber.FindIndexInListByName (VideoGrabber.VideoCom
```

Note: when the application starts the [VideoCompressors](#) and [AudioCompressors](#) lists are populated with the codecs currently installed. To refresh the compressor lists after installing/uninstalling codecs without exiting/restarting your application, invoke [RefreshDevicesAndCompressorsLists](#).

See Also

[Recording methods and properties TCompressionType TOnVideoCompressionSettings AudioCompressor AudioCompressorIndex AudioCompressorName AudioCompressors AudioCompressorsCount CompressionMode CompressionType GetVideoCompressionSettings OnReencodingCompleted OnReencodingStarted OnVideoCompressionSettings RefreshDevicesAndCompressorsLists SaveCompressorSettingsToDataString SetVideoCompressionDefaults SetVideoCompressionSettings VideoCompression KeyFrameRate VideoCompression PFramesPerKeyFrame VideoCompression Quality VideoCompression WindowSize VideoCompressor VideoCompressorIndex VideoCompressorName VideoCompressorsCount](#)

Created with the Standard Edition of HelpNDoc: [Enhance Your Documentation with HelpNDoc's Advanced Project Analyzer](#)

VideoCompressorsCount

| | | |
|--|----------------------|----------------------|
| TVideoGrabber.VideoCompressorsCount | Prev | Next |
| TVideoGrabber Properties | | |

Number of video compressors (video codecs) available on the current platform.

Declaration

property VideoCompressorsCount: LongInt **read** GetGlobal_VideoCompressorsCount;

__property **int** VideoCompressorsCount=read=GetGlobal_VideoCompressorsCount, **nodefault**

Function VideoCompressorsCount as Long

Description

Number of video compressors (video codecs) in the [VideoCompressors](#) list.

Note: in Delphi and C++Builder versions of the component, this property is available as a [VideoCompressorsCount](#) global variable.

See Also

[Recording methods and properties TCompressionType TOnVideoCompressionSettings AudioCompressor AudioCompressorIndex AudioCompressorName AudioCompressors AudioCompressorsCount CompressionMode CompressionType GetVideoCompressionSettings OnReencodingCompleted OnReencodingStarted OnVideoCompressionSettings RefreshDevicesAndCompressorsLists SaveCompressorSettingsToDataString SetVideoCompressionDefaults SetVideoCompressionSettings VideoCompression KeyFrameRate VideoCompression PFramesPerKeyFrame VideoCompression Quality VideoCompression WindowSize VideoCompressor VideoCompressorIndex VideoCompressorName VideoCompressors](#)

Created with the Standard Edition of HelpNDoc: [Easily create iPhone documentation](#)

VideoControlSettings

| | | |
|--|----------------------|----------------------|
| TVideoGrabber.VideoControlSettings | Prev | Next |
| TVideoGrabber Properties | | |

Enables or disables the [video control](#) settings.

Declaration

property VideoControlSettings: Boolean **read** GetVideoControlSettings **write** SetVideoControlSettings
default DEF_VideoControlSettings;

__property **bool** VideoControlSettings=read=GetVideoControlSettings, write=SetVideoControlSettings,
default=1

Property VideoControlSettings As Boolean

Description

Enables or disables the automatic saving and restoring of the [video control](#) settings.

See Also

[TVideoControl](#) [GetVideoControlMode](#) [IsVideoControlAvailable](#) [IsVideoControlModeAvailable](#)
[SetVideoControlMode](#) [SetVideoControlMode2](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Documentation Capabilities with HelpNDoc's User-Friendly UI](#)

VideoCursor

TVideoGrabber.VideoCursor

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Cursor displayed in the video window

Declaration

property VideoCursor: TCursor **read** GetVideoCursor **write** SetVideoCursor **default** crDefault;

__property Controls::TCursor VideoCursor=read=GetVideoCursor, write=SetVideoCursor, **default**=0

Property VideoCursor As TCursors

Description

Used to specify the cursor that will be displayed in the video window.

Created with the Standard Edition of HelpNDoc: [Easy Qt Help documentation editor](#)

VideoDevice

TVideoGrabber.VideoDevice

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Selects the current video capture device.

Declaration

property VideoDevice: LongInt **read** GetVideoDevice **write** SetVideoDevice **default** DEF_VideoDevice;

__property **int** VideoDevice=read=GetVideoDevice, write=SetVideoDevice, **default**=0

Property VideoDevice As Long

Description

Used to select the current video capture device. This property is an index in the [VideoDevices](#) list. When a new value is assigned to this property, the related [device-dependent values](#) are loaded from the registry and the [OnVideoDeviceSelected](#) event occurs.

It is possible to select the video capture device programmatically by its name (as it appears in the [VideoDevices](#) list) by using the [FindIndexInListByName](#) function, e.g.:

```

procedure TfrmMainForm.Button1Click(Sender: TObject);
var
  i: LongInt;
begin
  i := VideoGrabber.FindIndexInListByName (VideoGrabber.VideoDevices, 'Microsoft DV Ca
  if i > -1 then begin // if this Device exists...
    VideoGrabber.VideoDevice := i;
  end;
end;
end;

```

See Also

[WDM drivers](#)
[AutoConnectRelatedPins](#)
[CurrentFrameRate](#)
[DeliveredFrames](#)
[DroppedFrames](#)
[FixFlickerOrBlackCapture](#)
[IsDVDevice](#)
[IsVideoControlAvailable](#)
[IsVideoCrossbarAvailable](#)
[IsVideoDeviceConnected](#)
[IsWDMVideoDriver](#)
[OnNoVideoDevices](#)
[OnVideoDeviceSelected](#)
[RefreshDevicesAndCompressorsLists](#)
[ResetVideoDeviceSettings](#)
[ShowDialog](#)
[VideoDeviceIndex](#)
[VideoDeviceIndexFromId](#)
[VideoDeviceName](#)
[VideoDevices](#)
[VideoDevicesCount](#)
[VideoDevicesId](#)
[WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Step-by-Step Guide: How to Turn Your Word Document into an eBook](#)

VideoDeviceName

TVideoGrabber.VideoDeviceName

[Prev](#)

[Next](#)

[TVideoGrabber](#)
[Properties](#)

Name of the current video capture device.

Declaration

property VideoDeviceName: **string** **read** GetVideoDeviceName;

__property wchar_t *VideoDeviceName=read=GetVideoDeviceName

Property VideoDeviceName As String

Description

Used to retrieve the name of the video capture device selected by [VideoDevice](#) in the [VideoDevices](#) list.

See Also

[WDM drivers](#)
[AutoConnectRelatedPins](#)
[CurrentFrameRate](#)
[DeliveredFrames](#)
[DroppedFrames](#)
[FixFlickerOrBlackCapture](#)
[IsDVDevice](#)
[IsVideoControlAvailable](#)
[IsVideoCrossbarAvailable](#)
[IsVideoDeviceConnected](#)
[IsWDMVideoDriver](#)
[OnNoVideoDevices](#)
[OnVideoDeviceSelected](#)
[RefreshDevicesAndCompressorsLists](#)
[ResetVideoDeviceSettings](#)
[ShowDialog](#)
[VideoDevice](#)
[VideoDeviceIndex](#)
[VideoDeviceIndexFromId](#)
[VideoDevices](#)
[VideoDevicesCount](#)
[VideoDevicesId](#)
[WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Transform Your Documentation Process with HelpNDoc's Project Analyzer](#)

VideoDevices

TVideoGrabber.VideoDevices

[Prev](#)

[Next](#)

[TVideoGrabber](#)
[Properties](#)

List of the video capture devices available on the current platform.

Declaration

function VideoDevices: **string**;

__property wchar_t * __fastcall VideoDevices();

property VideoDevices as string

Description

Used to retrieve a string that contains the list of video capture devices available on the current platform.

This list is updated if a video capture device is connected or removed (when the [OnDeviceArrivalOrRemoval](#) event occurs).

This list can be assigned to list based controls. E.g.:

`ComboBox1.Items.Text := VideoGrabber1.VideoDevices;`

`ComboBox1.ItemIndex := VideoGrabber1.VideoDevice; (* index in the VideoDevices list *)`

It is possible to retrieve programmatically the index of a Video capture device by using the

[FindIndexInListByName](#) function as follows:

`VideoGrabber.VideoDevice := VideoGrabber.FindIndexInListByName (VideoGrabber.VideoDevices`

Note: in Delphi and C++Builder versions of the component, this property is available as a [VideoDevices](#) global variable.

See Also

[WDM drivers](#) [AutoConnectRelatedPins](#) [CurrentFrameRate](#) [DeliveredFrames](#) [DroppedFrames](#) [FixFlickerOrBlackCapture](#) [IsDVDevice](#) [IsVideoControlAvailable](#) [IsVideoCrossbarAvailable](#) [IsVideoDeviceConnected](#) [IsWDMVideoDriver](#) [OnNoVideoDevices](#) [OnVideoDeviceSelected](#) [RefreshDevicesAndCompressorsLists](#) [ResetVideoDeviceSettings](#) [ShowDialog](#) [VideoDevice](#) [VideoDeviceIndex](#) [VideoDeviceIndexFromId](#) [VideoDeviceName](#) [VideoDevicesCount](#) [VideoDevicesId](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Create iPhone web-based documentation](#)

VideoDevicesCount**TVideoGrabber.VideoDevicesCount**

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Number of video capture devices available on the current platform.

Declaration

property VideoDevicesCount: LongInt **read** GetGlobal_VideoDevicesCount;

__property **int** VideoDevicesCount=read=GetGlobal_VideoDevicesCount, **nodefault**

Function VideoDevicesCount as Long

Description

Number of video capture devices in the [VideoDevices](#) and [VideoDevicesId](#) lists.

Note: in Delphi and C++Builder versions of the component, this property is available as a [VideoDevicesCount](#) global variable.

See Also

[WDM drivers](#) [AutoConnectRelatedPins](#) [CurrentFrameRate](#) [DeliveredFrames](#) [DroppedFrames](#) [FixFlickerOrBlackCapture](#) [IsDVDevice](#) [IsVideoControlAvailable](#) [IsVideoCrossbarAvailable](#) [IsVideoDeviceConnected](#) [IsWDMVideoDriver](#) [OnNoVideoDevices](#) [OnVideoDeviceSelected](#) [RefreshDevicesAndCompressorsLists](#) [ResetVideoDeviceSettings](#) [ShowDialog](#) [VideoDevice](#) [VideoDeviceIndex](#) [VideoDeviceIndexFromId](#) [VideoDeviceName](#) [VideoDevices](#) [VideoDevicesId](#) [WebcamStillCaptureButton](#)

VideoDevicesId

TVideoGrabber.VideoDevicesId

[Prev](#)[Next](#)[TVideoGrabber](#) [Properties](#)

List of the registry identifiers of the video capture devices available on the current platform.

Declaration

function VideoDevicesId: **string**;

__property wchar_t * __fastcall VideoDevicesId();

property VideoDevicesId as string

Description

Used to retrieve a string that contains the list of the registry identifiers of video capture devices available on the current platform.

Unlike [VideoDevices](#), **VideoDevicesId** returns the unique registry names of the video capture devices. This is useful mainly to distinguish between several video capture devices of the same brand and model on the same platform.

This list is updated if a video capture device is connected or removed (when the [OnDeviceArrivalOrRemoval](#) event occurs).

This list can be assigned to list based controls. E.g.:

ComboBox1.Items.Text := VideoGrabber1.VideoDevicesId;

ComboBox1.ItemIndex := VideoGrabber1.VideoDevice; (index in the VideoDevicesId list *)*

It is possible to retrieve programmatically the index of a Video capture device by using the [FindIndexInListByName](#) function as follows:

```
VideoGrabber.VideoDevice := VideoGrabber.FindIndexInListByName (VideoGrabber.VideoDevicesId, VideoDeviceName);
```

Note: for Delphi and C++Builder versions of the component, this property is available as a [VideoDevices](#) global variable.

See Also

[WDM drivers](#) [AutoConnectRelatedPins](#) [CurrentFrameRate](#) [DeliveredFrames](#) [DroppedFrames](#) [FixFlickerOrBlackCapture](#) [IsDVDevice](#) [IsVideoControlAvailable](#) [IsVideoCrossbarAvailable](#) [IsVideoDeviceConnected](#) [IsWDMVideoDriver](#) [OnNoVideoDevices](#) [OnVideoDeviceSelected](#) [RefreshDevicesAndCompressorsLists](#) [ResetVideoDeviceSettings](#) [ShowDialog](#) [VideoDevice](#) [VideoDeviceIndex](#) [VideoDeviceIndexFromId](#) [VideoDeviceName](#) [VideoDevices](#) [VideoDevicesCount](#) [WebcamStillCaptureButton](#)

VideoDoubleBuffered

TVideoGrabber.VideoDoubleBuffered

[Prev](#)[Next](#)[TVideoGrabber](#) [Properties](#)

Activates / deactivate the double buffering in the OCX / DLL versions

Declaration

property VideoDoubleBuffered: Boolean **read** GetVideoDoubleBuffered **write** SetVideoDoubleBuffered

default DEF_DoubleBuffered;

__property bool VideoDoubleBuffered=read=GetVideoDoubleBuffered, write=SetVideoDoubleBuffered, **default=0**

Property VideoDoubleBuffered as Boolean

Description

Used to activates / deactivate the double buffering in the OCX / DLL versions
(in the Delphi or C++Builder versions modify the DoubleBuffered property)

See Also

[TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display](#) [Active](#) [Display](#) [AlphaBlendEnabled](#) [Display](#) [AlphaBlendValue](#) [Display](#) [AutoSize](#) [Display](#) [Embedded](#) [Display](#) [FullScreen](#) [Display](#) [Height](#) [Display](#) [Left](#) [Display](#) [Monitor](#) [Display](#) [MouseMovesWindow](#) [Display](#) [PanScanRatio](#) [Display](#) [SetLocation](#) [Display](#) [StayOnTop](#) [Display](#) [Top](#) [Display](#) [TransparentColorEnabled](#) [Display](#) [TransparentColorValue](#) [Display](#) [VideoHeight](#) [Display](#) [VideoPortEnabled](#) [Display](#) [VideoWidth](#) [Display](#) [VideoWindowHandle](#) [Display](#) [Width](#) [DualDisplay](#) [Active](#) [DualDisplay](#) [AlphaBlendEnabled](#) [DualDisplay](#) [AlphaBlendValue](#) [DualDisplay](#) [AutoSize](#) [DualDisplay](#) [Embedded](#) [DualDisplay](#) [FullScreen](#) [DualDisplay](#) [Height](#) [DualDisplay](#) [Left](#) [DualDisplay](#) [Monitor](#) [DualDisplay](#) [MouseMovesWindow](#) [DualDisplay](#) [PanScanRatio](#) [DualDisplay](#) [StayOnTop](#) [DualDisplay](#) [Top](#) [DualDisplay](#) [TransparentColorEnabled](#) [DualDisplay](#) [TransparentColorValue](#) [DualDisplay](#) [VideoHeight](#) [DualDisplay](#) [VideoPortEnabled](#) [DualDisplay](#) [VideoWidth](#) [DualDisplay](#) [VideoWindowHandle](#) [DualDisplay](#) [Visible](#) [DualDisplay](#) [Width](#) [IsVideoPortAvailable](#) [Monitor](#) [Primary](#) [Index](#) [MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#) [SetParentWindow](#) [SetWindowTransparency](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: Upgrade Your Documentation Process with a Help Authoring Tool

VideoFormat

TVideoGrabber.VideoFormat

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Index of the video format in the VideoFormats list

Declaration

property VideoFormat: LongInt **read** GetVideoFormat **write** SetVideoFormat;

__property int VideoFormat==GetVideoFormat, write=SetVideoFormat, **nodefault**};

Property VideoFormat as Long

Description

Used to select a video format in the [VideoFormats](#) list

See [Video Sizes](#)

Important:

The current video size index is not the same from a video capture device to the other. It is reloaded when the video capture device is selected by assigning the [VideoDevice](#) property.

- any control displaying this value should re-read it from the [OnVideoDeviceSelected](#) event (that occurs when the video device is selected)
- be sure that the video capture device has been selected (by assigning the [VideoDevice](#) property) before setting this property

See [device-dependent properties](#)

See Also

[GetNearestVideoHeight](#) [GetNearestVideoSize](#) [GetNearestVideoWidth](#) [GetVideoHeightFromIndex](#) [GetVideoSizeFromIndex](#) [GetVideoWidthFromIndex](#) [UseNearestVideoSize](#) [VideoFormatIndex](#) [VideoSize](#) [VideoSizeIndex](#) [VideoSizes](#) [VideoSizesCount](#)

Created with the Standard Edition of HelpNDoc: [What is a Help Authoring tool?](#)

VideoFormats

TVideoGrabber.VideoFormats

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

List of the video formats available on the current video capture device

Declaration

property VideoFormats: **string read** GetVideoFormats;

__property wchar_t *VideoFormats==GetVideoFormats;

Property VideoFormats as String

Description

Used to retrieve a string that contains the list of the video formats available on the current video capture device.

This list is updated when the [OnVideoDeviceSelected](#) event occurs (when a video capture device is selected with [VideoDevice](#)).

This list can be assigned to list based controls. E.g.:

ComboBox1.Items.Text := VideoGrabber1.VideoFormats;

ComboBox1.ItemIndex := VideoGrabber1.VideoFormat;

Note: as the video formats list is not the same from a video capture device to the other, any code using this value should re-read it from the [OnVideoDeviceSelected](#) event (when a [video capture device](#) is selected).

See the [Video sizes](#) chapter for more information.

Created with the Standard Edition of HelpNDoc: [Produce electronic books easily](#)

VideoFormatsCount

TVideoGrabber.VideoFormatsCount

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Number of video formats

Declaration

property VideoFormatsCount: LongInt **read** GetVideoFormatsCount;

__property int VideoFormatsCount==GetVideoFormatsCount, **nodefault**};

property VideoFormatsCount as Long

Description

Number of video formats available in the [VideoFormats](#) list for the current video capture device.

Created with the Standard Edition of HelpNDoc: [Revolutionize Your CHM Help File Output with HelpNDoc](#)

VideoFromImages_BitmapsSortedBy

TVideoGrabber.VideoFromImages_BitmapsSortedBy

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Properties](#)

Specifies how the bitmap or JPEG files are sorted.

Declaration

property VideoFromImages_BitmapsSortedBy: TFileSort **read** GetVideoFromImages_BitmapsSortedBy
write SetVideoFromImages_BitmapsSortedBy **default** DEF_VideoFromImages_BitmapsSortedBy;

__property TFileSort VideoFromImages_BitmapsSortedBy=read=GetVideoFromImages_BitmapsSortedBy,
 write=SetVideoFromImages_BitmapsSortedBy, **default**=0

Property VideoFromImages_BitmapsSortedBy As TFileSort

Description

Used to specify in which order the BMP or JPEG files located in [VideoFromImages_SourceDirectory](#) will be recorded.

The value is a [TFileSort](#) type.

Note: to sort correctly numeric sequences of bitmaps:

- use fs_NameAsc or fs_NameDesc,
- be sure to make the name of the images"zero padded". E.g.:

```
001.bmp
002.bmp
003.bmp
...
009.bmp
010.bmp
011.bmp
...
098.bmp
099.bmp
100.bmp
101.bmp
...
```

See Also

[OnBitmapsLoadingProgress](#) [SendImageToVideoFromBitmaps](#) [SendImageToVideoFromBitmaps2](#)
[VideoFromImages_RepeatIndefinitely](#) [VideoFromImages_SourceDirectory](#)
[VideoFromImages_TemporaryFile](#)

Created with the Standard Edition of HelpNDoc: [Make CHM Help File Creation a Breeze with HelpNDoc](#)

VideoFromImages_RepeatIndefinitely

TVideoGrabber.VideoFromImages_RepeatIndefinitely

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Properties](#)

Specifies if TVideoGrabber must restart from the 1st image when the last image is reached.

Declaration

property VideoFromImages_RepeatIndefinitely: Boolean **read** GetVideoFromImages_RepeatIndefinitely
write SetVideoFromImages_RepeatIndefinitely **default** DEF_VideoFromImages_RepeatIndefinitely;

__property **bool** VideoFromImages_RepeatIndefinitely=read=GetVideoFromImages_RepeatIndefinitely, write=SetVideoFromImages_RepeatIndefinitely, **default**=0

Property VideoFromImages_RepeatIndefinitely As Boolean

Description

Used to specify if TVideoGrabber must restart from the 1st image when the last image is reached in the [VideoFromImages_SourceDirectory](#) directory

See Also

[OnBitmapsLoadingProgress](#) [SendImageToVideoFromBitmaps](#) [SendImageToVideoFromBitmaps2](#) [VideoFromImages_BitmapSortedBy](#) [VideoFromImages_SourceDirectory](#) [VideoFromImages_TemporaryFile](#)

Created with the Standard Edition of HelpNDoc: [Free CHM Help documentation generator](#)

VideoFromImages_SourceDirectory

TVideoGrabber.VideoFromImages_SourceDirectory

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the directory where the image used to create the video clip are located.

Declaration

property VideoFromImages_SourceDirectory: **String** read GetVideoFromImages_SourceDirectory **write** SetVideoFromImages_SourceDirectory **stored** False;

__property wchar_t *VideoFromImages_SourceDirectory=read=GetVideoFromImages_SourceDirectory, write=SetVideoFromImages_SourceDirectory, stored=false

Property VideoFromImages_SourceDirectory As String

Description

Used to specify the directory in which the image used to create the video clip are located.

See Also

[OnBitmapsLoadingProgress](#) [SendImageToVideoFromBitmaps](#) [SendImageToVideoFromBitmaps2](#) [VideoFromImages_BitmapSortedBy](#) [VideoFromImages_RepeatIndefinitely](#) [VideoFromImages_TemporaryFile](#)

Created with the Standard Edition of HelpNDoc: [Make the switch to CHM with HelpNDoc's hassle-free WinHelp HLP to CHM conversion tool](#)

VideoFromImages_TemporaryFile

TVideoGrabber.VideoFromImages_TemporaryFile

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the full qualified path and file name of the temporary file.

Declaration

property VideoFromImages_TemporaryFile: **String** read GetVideoFromImages_TemporaryFile **write** SetVideoFromImages_TemporaryFile;

__property wchar_t *VideoFromImages_TemporaryFile=read=GetVideoFromImages_TemporaryFile, write=SetVideoFromImages_TemporaryFile

Property VideoFromImages_TemporaryFile As String

Description

Specifies the full qualified path and file name of the temporary file in which the BMP or JPEG files will be encoded before creating the video clip.

See Also

[OnBitmapsLoadingProgress](#) [SendImageToVideoFromBitmaps](#) [SendImageToVideoFromBitmaps2](#) [VideoFromImages](#) [BitmapsSortedBy](#) [VideoFromImages](#) [RepeatIndefinitely](#) [VideoFromImages](#) [SourceDirectory](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Documentation Efficiency with a Help Authoring Tool](#)

VideoHeight

TVideoGrabber.VideoHeight

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Retrieves the height of the native video source, in pixels.

Declaration

property VideoHeight: LongInt **read** GetVideoHeight;

__property int VideoHeight=read=GetVideoHeight, **nodefault**

Property VideoHeight As Long

Description

Used to retrieve the height of the native video source, in pixels.

See Also

[TOnFrameProgress](#) [TOnResizeVideo](#) [AutoRefreshPreview](#) [BackgroundColor](#) [Display](#) [AutoSize](#) [Display](#) [FullScreen](#) [Display](#) [SetLocation](#) [Display](#) [VideoPortEnabled](#) [ImageRatio](#) [InFrameProgressEvent](#) [IsVideoInterlaced](#) [IsVideoPortAvailable](#) [OnFrameProgress](#) [OnKeyPress](#) [OnPreviewStarted](#) [OnResizeVideo](#) [PausePreview](#) [PreviewZoomSize](#) [ResumePreview](#) [StartPreview](#) [StartSynchronized](#) [StopPreview](#) [Synchronized](#) [UseNearestVideoSize](#) [VideoRenderer](#) [VideoRendererPriority](#) [VideoWidth](#) [Visible](#)

Created with the Standard Edition of HelpNDoc: [Free iPhone documentation generator](#)

VideoHeight_PreferedAspectRatio

TVideoGrabber.VideoHeight_PreferedAspectRatio

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Retrieves the preferred display height of the video clip, in pixels.

Declaration

property VideoHeight_PreferedAspectRatio: LongInt **read** GetVideoHeight_PreferedAspectRatio;

__property int VideoHeight_PreferedAspectRatio=read=GetVideoHeight_PreferedAspectRatio, **nodefault**

Property VideoHeight_PreferedAspectRatio As Long

Description

Used to retrieve the preferred display height of the video clip, in pixels.

It can be different of the [VideoHeight](#) property if the video clip has a display aspect ratio (DAR) different of the source aspect ratio (SAR)

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TVideoRenderer](#) [TOnPlayerBufferingData](#) [AdjustPixelAspectRatio](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [Display](#) [Active](#) [Display](#) [AlphaBlendEnabled](#) [Display](#) [AlphaBlendValue](#) [Display](#) [AutoSize](#) [Display](#) [Embedded](#) [Display](#) [FullScreen](#) [Display](#) [Height](#) [Display](#) [Left](#) [Display](#) [Monitor](#) [Display](#) [MouseMovesWindow](#) [Display](#) [PanScanRatio](#) [Display](#) [SetLocation](#) [Display](#) [StayOnTop](#) [Display](#) [Top](#) [Display](#) [TransparentColorEnabled](#) [Display](#) [TransparentColorValue](#) [Display](#) [VideoHeight](#) [Display](#) [VideoPortEnabled](#) [Display](#) [VideoWidth](#) [Display](#) [VideoWindowHandle](#) [Display](#) [Width](#) [DualDisplay](#) [Active](#) [DualDisplay](#) [AlphaBlendEnabled](#) [DualDisplay](#) [AlphaBlendValue](#) [DualDisplay](#) [AutoSize](#) [DualDisplay](#) [Embedded](#) [DualDisplay](#) [FullScreen](#) [DualDisplay](#) [Height](#) [DualDisplay](#) [Left](#) [DualDisplay](#) [Monitor](#) [DualDisplay](#) [MouseMovesWindow](#) [DualDisplay](#) [PanScanRatio](#) [DualDisplay](#) [StayOnTop](#) [DualDisplay](#) [Top](#) [DualDisplay](#) [TransparentColorEnabled](#) [DualDisplay](#) [TransparentColorValue](#) [DualDisplay](#) [VideoHeight](#) [DualDisplay](#) [VideoPortEnabled](#) [DualDisplay](#) [VideoWidth](#) [DualDisplay](#) [VideoWindowHandle](#) [DualDisplay](#) [Visible](#) [DualDisplay](#) [Width](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [IsVideoPortAvailable](#) [Last Clip](#) [Played](#) [Monitor](#) [Primary](#) [Index](#) [MonitorBounds](#) [MonitorsCount](#) [MP4NeedsReindexing](#) [OnLeavingFullScreen](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchron](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [SetParentWindow](#) [SetWindowTransparency](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoDoubleBuffered](#) [VideoPlayableWhileRecording](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Make your documentation accessible on any device with HelpNDoc](#)

VideoInput

TVideoGrabber.VideoInput

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Selects the video input.

Declaration

property VideoInput: LongInt **read** GetVideoInput **write** SetVideoInput;

___property **int** VideoInput=read=GetVideoInput, write=SetVideoInput, **nodefault**

Property VideoInput As Long

Description

Selects the video input.

This property is an index in the [VideoInputs](#) list, in the 0 ... [VideoInputsCount](#) - 1 range.

Important:

The current video size index is not the same from a video capture device to the other. It is reloaded when the video capture device is selected by assigning the [VideoDevice](#) property.

- any control displaying this value should re-read it from the [OnVideoDeviceSelected](#) event (that occurs when the video device is selected)
- be sure that the video capture device has been selected (by assigning the [VideoDevice](#) property) before setting this property

See [VideoInputs](#) and [device-dependent properties](#)

See Also

[Video inputs](#) [VideoInputIndex](#) [VideoInputs](#) [VideoInputsCount](#)

Created with the Standard Edition of HelpNDoc: [Easily create PDF Help documents](#)

VideoInputs**TVideoGrabber.VideoInputs**

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

List of the video inputs available on the current video capture device

Declaration

property VideoInputs: **string read** GetVideoInputs;

__property wchar_t *VideoInputs=read=GetVideoInputs

Property VideoInputs As String

Description

Used to retrieve a string that contains the list of the video inputs available on the current video capture device.

This list can be assigned to list based controls. E.g.:

ComboBox1.Items.Text := VideoGrabber1.VideoInputs;

ComboBox1.ItemIndex := VideoGrabber1.VideoInput;

Note: as this list is not the same from a video capture device to the other, any code using this value should re-read it from the [OnVideoDeviceSelected](#) event (when a [video capture device](#) is selected).

See Also

[Video inputs](#) [VideoInput](#) [VideoInputIndex](#) [VideoInputsCount](#)

Created with the Standard Edition of HelpNDoc: [Benefits of a Help Authoring Tool](#)

VideoInputsCount**TVideoGrabber.VideoInputsCount**

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Number of video inputs.

Declaration

property VideoInputsCount: LongInt **read** GetVideoInputsCount;

__property int VideoInputsCount=read=GetVideoInputsCount, **nodefault**

Property VideoInputsCount As Long

Description

Number of video inputs in the [VideoInputs](#) list for the current video capture device.

See Also

[Video inputs](#) [VideoInput](#) [VideoInputIndex](#) [VideoInputs](#)

Created with the Standard Edition of HelpNDoc: [Make your documentation accessible on any device with HelpNDoc](#)

VideoPlayableWhileRecording

TVideoGrabber.VideoPlayableWhileRecording

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Allow playing a video clip being recorded

Declaration

property VideoPlayableWhileRecording: Boolean read GetVideoPlayableWhileRecording write SetVideoPlayableWhileRecording **default** DEF_VideoPlayableWhileRecording;

__**property bool** VideoPlayableWhileRecording==GetVideoPlayableWhileRecording, write=SetVideoPlayableWhileRecording, **default**=0;

Description

When enabled, a .MP4 or .MKV video currently being recorded can be opened in the player of another TVideoGrabber instance or in a third-party player.

Note:

- for IP cameras and URL sources being recorded in native format, this requires the Datastead RTSP/RTMP/HTTP/ONVIF Source Filter to be installed.
- for other sources the Datastead Multipurpose Encoder must be installed

See Also

[Player features](#) [Recording methods and properties](#) [TAVIMuxConfig](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnPlayerBufferingData](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioChannelRenderMode](#) [AudioRecording](#) [AudioStreamNumber](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [Encoder](#) [SetInt](#) [FastForwardPlayer](#) [HoldRecording](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [IsRecordingPaused](#) [Last_Clip_Played](#) [Last_Recording_FileName](#) [MP4NeedsReindexing](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PauseRecording](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchronise](#) [PlayerVideoCodec](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [ShowDialog](#) [SourceStream](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopPlayer](#) [StopRecording](#) [StoragePath](#) [SynchronizationRole](#) [Synchronized](#) [SyncPreview](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: Streamline your documentation process with HelpNDoc's WinHelp HLP to CHM conversion feature

VideoProcessing_Brightness

TVideoGrabber.VideoProcessing_Brightness

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Modifies the brightness of the video stream on which the [frame grabber](#) is applied.

Declaration

property VideoProcessing_Pixellization: LongInt **read** GetVideoProcessingPixellization **write** SetVideoProcessingPixellization **default** DEF_VideoProcessing_Pixellization;

__**property int** VideoProcessing_Pixellization=read=GetVideoProcessingPixellization, write=SetVideoProcessingPixellization, **default**=0

Property VideoProcessing_Pixellization As Long

Description

Used to modify the brightness value of the video stream on which the [frame grabber](#) is applied. In the -128..+128 range.

See Also

[TVideoDeinterlacing](#) [MP4NeedsReindexing](#) [OnFrameBitmap](#) [RetrieveInitialXYAfterRotation](#) [VideoProcessing_Contrast](#) [VideoProcessing_Deinterlacing](#) [VideoProcessing_FlipHorizontal](#) [VideoProcessing_FlipVertical](#) [VideoProcessing_GrayScale](#) [VideoProcessing_Hue](#) [VideoProcessing_InvertColors](#) [VideoProcessing_Pixellization](#) [VideoProcessing_Rotation](#) [VideoProcessing_RotationCustomAngle](#) [VideoProcessing_Saturation](#) [TVideoRotation](#)

Created with the Standard Edition of HelpNDoc: [From Word to ePub or Kindle eBook: A Comprehensive Guide](#)

VideoProcessing_Contrast

TVideoGrabber.VideoProcessing_Contrast

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Modifies the contrast value of the video stream on which the [frame grabber](#) is applied.

Declaration

property VideoProcessing_Contrast: LongInt **read** GetVideoProcessingContrast **write** SetVideoProcessingContrast **default** DEF_VideoProcessing_Contrast;

__**property int** VideoProcessing_Contrast=read=GetVideoProcessingContrast, write=SetVideoProcessingContrast, **default**=0

Property VideoProcessing_Contrast As Long

Description

Used to modify the contrast value of the video stream on which the [frame grabber](#) is applied. In the -128..+128 range.

See Also

[TVideoDeinterlacing](#) [MP4NeedsReindexing](#) [OnFrameBitmap](#) [RetrieveInitialXYAfterRotation](#) [VideoProcessing_Brightness](#) [VideoProcessing_Deinterlacing](#) [VideoProcessing_FlipHorizontal](#) [VideoProcessing_FlipVertical](#) [VideoProcessing_GrayScale](#) [VideoProcessing_Hue](#) [VideoProcessing_InvertColors](#) [VideoProcessing_Pixellization](#) [VideoProcessing_Rotation](#) [VideoProcessing_RotationCustomAngle](#) [VideoProcessing_Saturation](#) [TVideoRotation](#)

Created with the Standard Edition of HelpNDoc: [Transform Your Word Doc into a Professional-Quality eBook with HelpNDoc](#)

VideoProcessing_Deinterlacing

TVideoGrabber.VideoProcessing_Deinterlacing

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enable the deinterlacing of the video stream.

Declaration

property VideoProcessing_Deinterlacing: TVideoDeinterlacing **read** GetVideoProcessingDeinterlacing **write** SetVideoProcessingDeinterlacing **default** DEF_VideoProcessing_Deinterlacing;

__property TVideoDeinterlacing VideoProcessing_Deinterlacing=read=GetVideoProcessingDeinterlacing, write=SetVideoProcessingDeinterlacing, **default**=0

Property VideoProcessing_Deinterlacing As TVideoDeinterlacing

Description

This [TVideoDeinterlacing](#) property is used to enable/disable the deinterlacing of the video stream.

See the [Deinterlacing](#) chapter.

See Also

[TVideoDeinterlacing](#) [MP4NeedsReindexing](#) [OnFrameBitmap](#) [RetrieveInitialXYAfterRotation](#) [VideoProcessing_Brightness](#) [VideoProcessing_Contrast](#) [VideoProcessing_FlipHorizontal](#) [VideoProcessing_FlipVertical](#) [VideoProcessing_GrayScale](#) [VideoProcessing_Hue](#) [VideoProcessing_InvertColors](#) [VideoProcessing_Pixellization](#) [VideoProcessing_Rotation](#) [VideoProcessing_RotationCustomAngle](#) [VideoProcessing_Saturation](#) [TVideoRotation](#)

Created with the Standard Edition of HelpNDoc: [Full-featured multi-format Help generator](#)

VideoProcessing_FlipHorizontal

TVideoGrabber.VideoProcessing_FlipHorizontal

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Flips horizontally the video stream on which the [frame grabber](#) is applied.

Declaration

property VideoProcessing_FlipHorizontal: Boolean **read** GetVideoProcessingLeftRight **write** SetVideoProcessingLeftRight **default** DEF_VideoProcessing_FlipHorizontal;

__property **bool** VideoProcessing_FlipHorizontal=read=GetVideoProcessingLeftRight, write=SetVideoProcessingLeftRight, **default**=0

Property VideoProcessing_FlipHorizontal As Boolean

Description

When enabled, video frames are flipped horizontally on the video stream on which the [frame grabber](#) is applied.

See Also

[TVideoDeinterlacing](#) [MP4NeedsReindexing](#) [OnFrameBitmap](#) [RetrieveInitialXYAfterRotation](#) [VideoProcessing_Brightness](#) [VideoProcessing_Contrast](#) [VideoProcessing_Deinterlacing](#) [VideoProcessing_FlipVertical](#) [VideoProcessing_GrayScale](#) [VideoProcessing_Hue](#) [VideoProcessing_InvertColors](#) [VideoProcessing_Pixellization](#) [VideoProcessing_Rotation](#) [VideoProcessing_RotationCustomAngle](#) [VideoProcessing_Saturation](#) [TVideoRotation](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Documentation Efficiency with a Help Authoring Tool](#)

VideoProcessing_FlipVertical

TVideoGrabber.VideoProcessing_FlipVertical

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Flips vertically the video stream on which the [frame grabber](#) is applied.

Declaration

property VideoProcessing_FlipVertical: Boolean **read** GetVideoProcessingTopDown **write** SetVideoProcessingTopDown **default** DEF_VideoProcessing_FlipVertical;

__property **bool** VideoProcessing_FlipVertical=read=GetVideoProcessingTopDown, write=SetVideoProcessingTopDown, **default**=0

Property VideoProcessing_FlipVertical As Boolean

Description

When enabled, video frames are flipped vertically on the video stream on which the [frame grabber](#) is applied.

See Also

[TVideoDeinterlacing](#) [MP4NeedsReindexing](#) [OnFrameBitmap](#) [RetrieveInitialXYAfterRotation](#) [VideoProcessing_Brightness](#) [VideoProcessing_Contrast](#) [VideoProcessing_Deinterlacing](#) [VideoProcessing_FlipHorizontal](#) [VideoProcessing_GrayScale](#) [VideoProcessing_Hue](#) [VideoProcessing_InvertColors](#) [VideoProcessing_Pixellization](#) [VideoProcessing_Rotation](#) [VideoProcessing_RotationCustomAngle](#) [VideoProcessing_Saturation](#) [TVideoRotation](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly optimize your documentation website for search engines](#)

VideoProcessing_GrayScale

TVideoGrabber.VideoProcessing_GrayScale

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Changes colors in greyscale on the video stream on which the [frame grabber](#) is applied.

Declaration

property VideoProcessing_GrayScale: Boolean **read** GetVideoProcessingGrayScale **write** SetVideoProcessingGrayScale **default** DEF_VideoProcessing_GrayScale;

__property **bool** VideoProcessing_GrayScale=read=GetVideoProcessingGrayScale, write=SetVideoProcessingGrayScale, **default**=0

Property VideoProcessing_GrayScale As Boolean

Description

When enabled, colors are changed in grayscale on the video stream on which the [frame grabber](#) is applied.

See Also

[TVideoDeinterlacing](#) [MP4NeedsReindexing](#) [OnFrameBitmap](#) [RetrieveInitialXYAfterRotation](#) [VideoProcessing_Brightness](#) [VideoProcessing_Contrast](#) [VideoProcessing_Deinterlacing](#) [VideoProcessing_FlipHorizontal](#) [VideoProcessing_FlipVertical](#) [VideoProcessing_Hue](#) [VideoProcessing_InvertColors](#) [VideoProcessing_Pixellization](#) [VideoProcessing_Rotation](#) [VideoProcessing_RotationCustomAngle](#) [VideoProcessing_Saturation](#) [TVideoRotation](#)

Created with the Standard Edition of HelpNDoc: [Streamline Your Documentation Creation with a Help Authoring Tool](#)

VideoProcessing_Hue

TVideoGrabber.VideoProcessing_Hue

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Modifies the hue of the video stream on which the [frame grabber](#) is applied.

Declaration

property VideoProcessing_Hue: LongInt **read** GetVideoProcessingHue **write** SetVideoProcessingHue
default DEF_VideoProcessing_Hue;

__property **int** VideoProcessing_Hue=read=GetVideoProcessingHue, write=SetVideoProcessingHue,
default=0

Property VideoProcessing_Hue As Long

Description

Used to modify the hue value of the video stream on which the [frame grabber](#) is applied.

See Also

[TVideoDeinterlacing](#) [MP4NeedsReindexing](#) [OnFrameBitmap](#) [RetrieveInitialXYAfterRotation](#)
[VideoProcessing_Brightness](#) [VideoProcessing_Contrast](#) [VideoProcessing_Deinterlacing](#)
[VideoProcessing_FlipHorizontal](#) [VideoProcessing_FlipVertical](#) [VideoProcessing_GrayScale](#)
[VideoProcessing_InvertColors](#) [VideoProcessing_Pixellization](#) [VideoProcessing_Rotation](#)
[VideoProcessing_RotationCustomAngle](#) [VideoProcessing_Saturation](#) [TVideoRotation](#)

Created with the Standard Edition of HelpNDoc: [Elevate Your Documentation with HelpNDoc's Project Analyzer Features](#)

VideoProcessing_InvertColors

TVideoGrabber.VideoProcessing_InvertColors

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Inverts the colors of the video stream on which the [frame grabber](#) is applied.

Declaration

property VideoProcessing_InvertColors: Boolean **read** GetVideoProcessingInvertColors **write** SetVideoProcessingInvertColors **default** DEF_VideoProcessing_InvertColors;

__property **bool** VideoProcessing_InvertColors=read=GetVideoProcessingInvertColors,
write=SetVideoProcessingInvertColors, **default**=0

Property VideoProcessing_InvertColors As Boolean

Description

When enabled, colors are inverted on the video stream on which the [frame grabber](#) is applied.

See Also

[TVideoDeinterlacing](#) [MP4NeedsReindexing](#) [OnFrameBitmap](#) [RetrieveInitialXYAfterRotation](#)
[VideoProcessing_Brightness](#) [VideoProcessing_Contrast](#) [VideoProcessing_Deinterlacing](#)
[VideoProcessing_FlipHorizontal](#) [VideoProcessing_FlipVertical](#) [VideoProcessing_GrayScale](#)
[VideoProcessing_Hue](#) [VideoProcessing_Pixellization](#) [VideoProcessing_Rotation](#)
[VideoProcessing_RotationCustomAngle](#) [VideoProcessing_Saturation](#) [TVideoRotation](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Publish Your Word Document as an eBook](#)

VideoProcessing_Pixellization

TVideoGrabber.VideoProcessing_Pixellization

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

ber

Used to pixellize the video frames

Declaration

property VideoProcessing_Pixellization: LongInt **read** GetVideoProcessingPixellization **write** SetVideoProcessingPixellization **default** DEF_VideoProcessing_Pixellization;

__**property int** VideoProcessing_Pixellization==GetVideoProcessingPixellization,
write=SetVideoProcessingPixellization, **default**=1

Property VideoProcessing_Pixellization as Long

Description

This property is used to create a pixellized effect.

Assign to this property specify a pixel size > 1.

The [FrameGrabber](#) must be enabled.

Default value: 1

See Also

[TVideoDeinterlacing](#) [MP4NeedsReindexing](#) [OnFrameBitmap](#) [RetrieveInitialXYAfterRotation](#)
[VideoProcessing_Brightness](#) [VideoProcessing_Contrast](#) [VideoProcessing_Deinterlacing](#)
[VideoProcessing_FlipHorizontal](#) [VideoProcessing_FlipVertical](#) [VideoProcessing_GrayScale](#)
[VideoProcessing_Hue](#) [VideoProcessing_InvertColors](#) [VideoProcessing_Rotation](#)
[VideoProcessing_RotationCustomAngle](#) [VideoProcessing_Saturation](#) [TVideoRotation](#)

Created with the Standard Edition of HelpNDoc: [Easily create HTML Help documents](#)

VideoProcessing_Rotation**TVideoGrabber.VideoProcessing_Rotation**
[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies if the video image must be rotated and/or mirrored, by 90, 180 or 270 degrees.

Declaration

property VideoProcessing_Rotation: TVideoRotation **read** GetVideoProcessingRotation **write** SetVideoProcessingRotation **default** DEF_VideoProcessing_Rotation;

__**property TVideoRotation** VideoProcessing_Rotation=read=GetVideoProcessingRotation,
write=SetVideoProcessingRotation, **default**=0

Property VideoProcessing_Rotation As TVideoRotation

Description

This [TVideoRotation](#) property is used to specify if the video image must be rotated and/or mirrored, by 90, 180 or 270 degrees.

rt_0_deg : no rotation, not mirrored

rt_90_deg : 90° rotation

rt_180_deg : 180° rotation

rt_270_deg : 270° rotation

rt_0_deg_mirror : no rotation, mirrored

rt_90_deg_mirror : 90° rotation, mirrored

rt_180_deg_mirror : 180° rotation, mirrored

rt_270_deg_mirror : 270° rotation, mirrored

rt_custom_angle : rotation of a [VideoProcessing_RotationCustomAngle](#) value

rt_custom_angle_mirror_vert : rotation of a [VideoProcessing_RotationCustomAngle](#) value, with vertical mirroring
rt_custom_angle_mirror_horz : rotation of a [VideoProcessing_RotationCustomAngle](#) value, with horizontal mirroring
rt_custom_angle_mirror_both : rotation of a [VideoProcessing_RotationCustomAngle](#) value, with vertical and horizontal mirroring

Remark: the rt_custom_angle... settings require a lot of CPU, if you get dropped frames use a lower video size.

See Also

[TVideoDeinterlacing](#) [MP4NeedsReindexing](#) [OnFrameBitmap](#) [RetrieveInitialXYAfterRotation](#)
[VideoProcessing_Brightness](#) [VideoProcessing_Contrast](#) [VideoProcessing_Deinterlacing](#)
[VideoProcessing_FlipHorizontal](#) [VideoProcessing_FlipVertical](#) [VideoProcessing_GrayScale](#)
[VideoProcessing_Hue](#) [VideoProcessing_InvertColors](#) [VideoProcessing_Pixellization](#)
[VideoProcessing_RotationCustomAngle](#) [VideoProcessing_Saturation](#) [TVideoRotation](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Productivity with HelpNDoc's Efficient User Interface](#)

VideoProcessing_RotationCustomAngle

TVideoGrabber.VideoProcessing_RotationCustomAngle

[Prev](#)
[Next](#)

e

[TVideoGrabber](#) [Properties](#)

Specifies the angle of rotation of the video image.

Declaration

property VideoProcessing_Rotation: TVideoRotation **read** GetVideoProcessingRotation **write** SetVideoProcessingRotation **default** DEF_VideoProcessing_Rotation;

__property double VideoProcessing_RotationCustomAngle==GetVideoProcessingRotationCustomAngle, write=SetVideoProcessingRotationCustomAngle;

property VideoProcessing_RotationCustomAngle as Double

Description

Used to specify the angle of rotation of the video image, in degrees, when using the **rt_custom_Angle...** settings of the [VideoProcessing_Rotation](#) property.

See Also

[TVideoDeinterlacing](#) [MP4NeedsReindexing](#) [OnFrameBitmap](#) [RetrieveInitialXYAfterRotation](#)
[VideoProcessing_Brightness](#) [VideoProcessing_Contrast](#) [VideoProcessing_Deinterlacing](#)
[VideoProcessing_FlipHorizontal](#) [VideoProcessing_FlipVertical](#) [VideoProcessing_GrayScale](#)
[VideoProcessing_Hue](#) [VideoProcessing_InvertColors](#) [VideoProcessing_Pixellization](#)
[VideoProcessing_Rotation](#) [VideoProcessing_Saturation](#) [TVideoRotation](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Publish Your Word Document as an eBook](#)

VideoProcessing_Saturation

TVideoGrabber.VideoProcessing_Saturation

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Modifies the saturation of the video stream on which the [frame grabber](#) is applied.

Declaration

property VideoProcessing_Saturation: LongInt **read** GetVideoProcessingSaturation **write**

SetVideoProcessingSaturation **default** DEF_VideoProcessing_Saturation;

__property **int** VideoProcessing_Saturation=read=GetVideoProcessingSaturation,
write=SetVideoProcessingSaturation, **default**=0

Property VideoProcessing_Saturation As Long

Description

Used to modify the saturation value of the video stream on which the [frame grabber](#) is applied.
In the -64..+64 range.

See Also

[TVideoDeinterlacing](#) [MP4NeedsReindexing](#) [OnFrameBitmap](#) [RetrieveInitialXYAfterRotation](#)
[VideoProcessing_Brightness](#) [VideoProcessing_Contrast](#) [VideoProcessing_Deinterlacing](#)
[VideoProcessing_FlipHorizontal](#) [VideoProcessing_FlipVertical](#) [VideoProcessing_GrayScale](#)
[VideoProcessing_Hue](#) [VideoProcessing_InvertColors](#) [VideoProcessing_Pixellization](#)
[VideoProcessing_Rotation](#) [VideoProcessing_RotationCustomAngle](#) [TVideoRotation](#)

Created with the Standard Edition of HelpNDoc: [Easy EPub and documentation editor](#)

VideoQualitySettings

TVideoGrabber.VideoQualitySettings

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Enables or disables the [video quality](#) settings.

Declaration

property VideoQualitySettings: Boolean **read** GetVideoQualitySettings **write** SetVideoQualitySettings
default DEF_VideoQualitySettings;

__property **bool** VideoQualitySettings=read=GetVideoQualitySettings, write=SetVideoQualitySettings,
default=1

Property VideoQualitySettings As Boolean

Description

Used to enable or disable the automatic saving / restoring of [video quality](#) settings.

See Also

[IsVideoQualityAvailable](#) [IsVideoQualitySettingAvailable](#) [SetVideoQuality](#) [VideoQualityAuto](#)
[VideoQualityDefault](#) [VideoQualityMax](#) [VideoQualityMin](#) [VideoQualityStep](#) [VideoRendererPriority](#)
[TVideoQuality](#)

Created with the Standard Edition of HelpNDoc: [Easily convert your WinHelp HLP help files to CHM with HelpNDoc's step-by-step guide](#)

VideoRenderer

TVideoGrabber.VideoRenderer

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the video renderer used for the video window.

Declaration

property VideoRenderer: TVideoRenderer **read** GetVideoRenderer **write** SetVideoRenderer **default**
DEF_VideoRenderer;

__property TVideoRenderer VideoRenderer=read=GetVideoRenderer, write=SetVideoRenderer, **default=0**

Property VideoRenderer As TxVideoRenderer

Description

Used to specify the [TVideoRenderer](#) renderer that performs rendering in the video window.

See Also

[TOnFrameProgress](#) [TOnResizeVideo](#) [AutoRefreshPreview](#) [BackgroundColor](#) [Display](#) [AutoSize](#) [Display_FullScreen](#) [Display_SetLocation](#) [Display_VideoPortEnabled](#) [ImageRatio](#) [InFrameProgressEvent](#) [IsVideoInterlaced](#) [IsVideoPortAvailable](#) [OnFrameProgress](#) [OnKeyPress](#) [OnPreviewStarted](#) [OnResizeVideo](#) [PausePreview](#) [PreviewZoomSize](#) [ResumePreview](#) [StartPreview](#) [StartSynchronized](#) [StopPreview](#) [Synchronized](#) [UseNearestVideoSize](#) [VideoHeight](#) [VideoRendererPriority](#) [VideoWidth](#) [Visible](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly bring your documentation online with HelpNDoc](#)

VideoRendererExternal

TVideoGrabber.VideoRendererExternal

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enable third-party video renderers (usually to a second monitor)

Declaration

property VideoRendererExternal: TVideoRendererExternal **read** GetVideoRendererExternal **write** SetVideoRendererExternal **default** DEF_VideoRendererExternal;

__property TVideoRendererExternal VideoRenderer=read=GetVideoRendererExternal, write=SetVideoRendererExternal, **default=0**

Property VideoRendererExternal As TxVideoRendererExternal

Description

Used to enable third-party video renderers (usually rendered to a second monitor).

The current allowed values are:

vre_None:

no external video renderer (default value)

vre_BlackMagic_Decklink:

external renderer on the [BlackMagic Decklink](#) card.

vre_Matrox_PRO:

external renderer on [Matrox](#) cards.

vre_Decklink_SD, vre_Decklink_Extreme:

deprecated.

vre_Pinnacle_MovieBoard:

external renderer on the [Pinnacle MovieBoard](#) card

Feel free to contact our support at support@datastead.com if your video card has an external renderer that is not supported by the current version

See also [VideoRendererExternalIndex](#)

See Also

[Dual display](#) [TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display_Active](#) [Display_AlphaBlendEnabled](#)

[Display_AlphaBlendValue](#) [Display_AutoSize](#) [Display_Embedded](#) [Display_FullScreen](#) [Display_Height](#) [Display_Left](#) [Display_Monitor](#) [Display_MouseMovesWindow](#) [Display_PanScanRatio](#) [Display_SetLocation](#) [Display_StayOnTop](#) [Display_Top](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [Display_VideoHeight](#) [Display_VideoPortEnabled](#) [Display_VideoWidth](#) [Display_VideoWindowHandle](#) [Display_Width](#) [DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#) [DualDisplay_AlphaBlendValue](#) [DualDisplay_AutoSize](#) [DualDisplay_Embedded](#) [DualDisplay_FullScreen](#) [DualDisplay_Height](#) [DualDisplay_Left](#) [DualDisplay_Monitor](#) [DualDisplay_MouseMovesWindow](#) [DualDisplay_PanScanRatio](#) [DualDisplay_SetLocation](#) [DualDisplay_StayOnTop](#) [DualDisplay_Top](#) [DualDisplay_TransparentColorEnabled](#) [DualDisplay_TransparentColorValue](#) [DualDisplay_VideoHeight](#) [DualDisplay_VideoPortEnabled](#) [DualDisplay_VideoWidth](#) [DualDisplay_VideoWindowHandle](#) [DualDisplay_Visible](#) [DualDisplay_Width](#) [IsVideoPortAvailable](#) [Monitor_Primary_Index](#) [MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#) [SetParentWindow](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: Modernize your help files with HelpNDoc's WinHelp HLP to CHM conversion tool

VideoRendererExternalIndex

TVideoGrabber.VideoRendererExternalIndex

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Index of the external video renderer to use

Declaration

property VideoRendererExternalIndex: LongInt **read** GetVideoRendererExternalIndex **write** SetVideoRendererExternalIndex;

__property int VideoRendererExternalIndex=read=GetVideoRendererExternalIndex, write=SetVideoRendererExternalIndex, **default**

Property VideoRendererExternalIndex As Long

Description

Index of the external video renderer to use.

Default value: -1 (the first card will be used)

This feature is useful when more than one external video renderer are installed, to let you specify what renderer should be used.

E.g. let's suppose a PC where 4 Blackmagic Decklink cards are installed:

e.g. to select the 1st Decklink card:

VideoGrabber.VideoRendererExternal = vre_BlackMagic_Decklink
VideoGrabber.VideoRendererExternalIndex = 0

e.g. to select the 2nd Decklink card:

VideoGrabber.VideoRendererExternal = vre_BlackMagic_Decklink
VideoGrabber.VideoRendererExternalIndex = 1

etc...

See Also

[Dual display](#) [TVideoRenderer_AdjustPixelAspectRatio](#) [Display_Active](#) [Display_AlphaBlendEnabled](#) [Display_AlphaBlendValue](#) [Display_AutoSize](#) [Display_Embedded](#) [Display_FullScreen](#) [Display_Height](#) [Display_Left](#) [Display_Monitor](#) [Display_MouseMovesWindow](#) [Display_PanScanRatio](#) [Display_SetLocation](#) [Display_StayOnTop](#) [Display_Top](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [Display_VideoHeight](#) [Display_VideoPortEnabled](#) [Display_VideoWidth](#) [Display_VideoWindowHandle](#) [Display_Width](#) [DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#) [DualDisplay_AlphaBlendValue](#)

[DualDisplay_AutoSize](#)
[DualDisplay_Embedded](#)
[DualDisplay_FullScreen](#)
[DualDisplay_Height](#)
[DualDisplay_Left](#)
[DualDisplay_Monitor](#)
[DualDisplay_MouseMovesWindow](#)
[DualDisplay_PanScanRatio](#)
[DualDisplay_SetLocation](#)
[DualDisplay_StayOnTop](#)
[DualDisplay_Top](#)
[DualDisplay_TransparentColorEnabled](#)
[DualDisplay_TransparentColorValue](#)
[DualDisplay_VideoHeight](#)
[DualDisplay_VideoPortEnabled](#)
[DualDisplay_VideoWidth](#)
[DualDisplay_VideoWindowHandle](#)
[DualDisplay_Visible](#)
[DualDisplay_Width](#)
[IsVideoPortAvailable](#)
[Monitor_Primary](#)
[Index](#)
[MonitorBounds](#)
[MonitorsCount](#)
[OnLeavingFullScreen](#)
[SetParentWindow](#)
[SetWindowTransparency](#)
[VideoDoubleBuffered](#)
[VideoHeight](#)
[PreferredAspectRatio](#)
[VideoRendererExternal](#)
[VideoRendererPriority](#)
[VideoVisibleWhenStopped](#)
[VideoWidth](#)
[PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [From Word to ePub or Kindle eBook: A Comprehensive Guide](#)

VideoRendererPriority

TVideoGrabber.VideoRendererPriority

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Specify whether the quality or speed must be favored.

Declaration

property VideoRendererPriority: TVideoRendererPriority **read** GetVideoRendererPriority **write** SetVideoRendererPriority **default** DEF_VideoRendererPriority;

__property TVideoRendererPriority VideoRendererPriority==GetVideoRendererPriority, write=SetVideoRendererPriority, **default**=2;

Description

Possible values here: [TVideoRendererPriority](#)

See Also

[TVideoRenderer](#)
[TOnFrameProgress](#)
[TOnResizeVideo](#)
[AdjustPixelAspectRatio](#)
[AutoRefreshPreview](#)
[BackgroundColor](#)
[Display_Active](#)
[Display_AlphaBlendEnabled](#)
[Display_AlphaBlendValue](#)
[Display_AutoSize](#)
[Display_Embedded](#)
[Display_FullScreen](#)
[Display_Height](#)
[Display_Left](#)
[Display_Monitor](#)
[Display_MouseMovesWindow](#)
[Display_PanScanRatio](#)
[Display_SetLocation](#)
[Display_StayOnTop](#)
[Display_Top](#)
[Display_TransparentColorEnabled](#)
[Display_TransparentColorValue](#)
[Display_VideoHeight](#)
[Display_VideoPortEnabled](#)
[Display_VideoWidth](#)
[Display_VideoWindowHandle](#)
[Display_Width](#)
[DualDisplay_Active](#)
[DualDisplay_AlphaBlendEnabled](#)
[DualDisplay_AlphaBlendValue](#)
[DualDisplay_AutoSize](#)
[DualDisplay_Embedded](#)
[DualDisplay_FullScreen](#)
[DualDisplay_Height](#)
[DualDisplay_Left](#)
[DualDisplay_Monitor](#)
[DualDisplay_MouseMovesWindow](#)
[DualDisplay_PanScanRatio](#)
[DualDisplay_StayOnTop](#)
[DualDisplay_Top](#)
[DualDisplay_TransparentColorEnabled](#)
[DualDisplay_TransparentColorValue](#)
[DualDisplay_VideoHeight](#)
[DualDisplay_VideoPortEnabled](#)
[DualDisplay_VideoWidth](#)
[DualDisplay_VideoWindowHandle](#)
[DualDisplay_Visible](#)
[DualDisplay_Width](#)
[ImageRatio](#)
[InFrameProgressEvent](#)
[IsVideoInterlaced](#)
[IsVideoPortAvailable](#)
[IsVideoQualityAvailable](#)
[IsVideoQualitySettingAvailable](#)
[Monitor_Primary](#)
[Index](#)
[MonitorBounds](#)
[MonitorsCount](#)
[OnFrameProgress](#)
[OnKeyPress](#)
[OnLeavingFullScreen](#)
[OnPreviewStarted](#)
[OnResizeVideo](#)
[PausePreview](#)
[PreviewZoomSize](#)
[ResumePreview](#)
[SetParentWindow](#)
[SetVideoQuality](#)
[SetWindowTransparency](#)
[StartPreview](#)
[StartSynchronized](#)
[StopPreview](#)
[Synchronized](#)
[UseNearestVideoSize](#)
[VideoDoubleBuffered](#)
[VideoHeight](#)
[VideoHeight](#)
[PreferredAspectRatio](#)
[VideoQualityAuto](#)
[VideoQualityDefault](#)
[VideoQualityMax](#)
[VideoQualityMin](#)
[VideoQualitySettings](#)
[VideoQualityStep](#)
[VideoRenderer](#)
[VideoRendererExternal](#)
[VideoRendererExternalIndex](#)
[VideoVisibleWhenStopped](#)
[VideoWidth](#)
[VideoWidth](#)
[PreferredAspectRatio](#)
[Visible](#)
[TVideoQuality](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Productivity with HelpNDoc's Efficient User Interface](#)

VideoSize

TVideoGrabber.VideoSize

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

ber

Selects a video size.

Declaration

property VideoSize: LongInt **read** GetVideoSize **write** SetVideoSize;

__property int VideoSize=read=GetVideoSize, write=SetVideoSize, **nodefault**

Property VideoSize As Long

Description

Used to select a video size in the [VideoSizes](#) list.

See [VideoSizes](#) .

Important:

The current video size index is not the same from a video capture device to the other. It is reloaded when the video capture device is selected by assigning the [VideoDevice](#) property.

- any control displaying this value should re-read it from the [OnVideoDeviceSelected](#) event (that occurs when the video device is selected)
- be sure that the video capture device has been selected (by assigning the [VideoDevice](#) property) before setting this property

See [device-dependent properties](#)

See Also

[GetNearestVideoHeight](#) [GetNearestVideoSize](#) [GetNearestVideoWidth](#) [GetVideoHeightFromIndex](#) [GetVideoSizeFromIndex](#) [GetVideoWidthFromIndex](#) [UseNearestVideoSize](#) [VideoFormat](#) [VideoFormatIndex](#) [VideoSizeIndex](#) [VideoSizes](#) [VideoSizesCount](#)

Created with the Standard Edition of HelpNDoc: [Full-featured EPub generator](#)

VideoSizes**TVideoGrabber.VideoSizes**
[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

List of the video sizes available on the current video capture device

Declaration

property VideoSizes: **string** **read** GetVideoSizes;

__property wchar_t *VideoSizes=read=GetVideoSizes

Property VideoSizes As String

Description

Used to retrieve a string that contains the list of the video sizes available on the current video capture device.

This list is updated when the [OnVideoDeviceSelected](#) event occurs (when a video capture device is selected with [VideoDevice](#)).

This list can be assigned to list based controls. E.g.:

ComboBox1.Items.Text := VideoGrabber1.VideoSizes;

ComboBox1.ItemIndex := VideoGrabber1.VideoSize;

Note: as the video sizes list is not the same from a video capture device to the other, any code using this value should re-read it from the [OnVideoDeviceSelected](#) event (when a [video capture device](#) is selected).

See the [Video sizes](#) chapter for more information.

See Also

[GetNearestVideoHeight](#) [GetNearestVideoSize](#) [GetNearestVideoWidth](#) [GetVideoHeightFromIndex](#) [GetVideoSizeFromIndex](#) [GetVideoWidthFromIndex](#) [UseNearestVideoSize](#) [VideoFormat](#) [VideoFormatIndex](#) [VideoSize](#) [VideoSizeIndex](#) [VideoSizesCount](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your PDF Protection with These Simple Steps](#)

VideoSizesCount**TVideoGrabber.VideoSizesCount**[Prev](#)[Next](#)

[TVideoGrabber](#) [Properties](#)

Number of video sizes.

Declaration

property VideoSizesCount: LongInt **read** GetVideoSizesCount;

__property **int** VideoSizesCount=read=GetVideoSizesCount, **nodefault**

Property VideoSizesCount As Long

Description

Number of video sizes in the [VideoSizes](#) list for the current video capture device.

See Also

[GetNearestVideoHeight](#) [GetNearestVideoSize](#) [GetNearestVideoWidth](#) [GetVideoHeightFromIndex](#) [GetVideoSizeFromIndex](#) [GetVideoWidthFromIndex](#) [UseNearestVideoSize](#) [VideoFormat](#) [VideoFormatIndex](#) [VideoSize](#) [VideoSizeIndex](#) [VideoSizes](#)

Created with the Standard Edition of HelpNDoc: [Create cross-platform Qt Help files](#)

VideoSource**TVideoGrabber.VideoSource**[Prev](#)[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the video source for [preview](#) and [recording](#).

Declaration

property VideoSource: TVideoSource **read** GetVideoSource **write** SetVideoSource **default** DEF_VideoSource;

__property TVideoSource VideoSource=read=GetVideoSource, write=SetVideoSource, **default**=0

Property VideoSource As TxVideoSource

Description

This [TVideoSource](#) property is used to specify the video source used for [preview](#) and [recording](#):

vs_VideoCaptureDevice:

the video source is the [current video capture device](#).

vs_ScreenRecording:

the video source is the screen. See the [Screen recording](#) chapter.

vs_VideoFileOrURL:

the video source is a video clip specified by the [VideoSource_FileOrURL](#) property.

vs_JPEGsOrBitmaps:

the video source are bitmap handles, or BMP files or JPEG files of the same format passed to the [OnVideoFromBitmaps_NextFrameNeeded](#) event.

See [Video clips built on the fly by passing bitmap handles, BMP or JPEG files](#).

vs_IPCamera:

the video source is an IP camera or a IP video server. See the [IP Cameras - IP Webcams - IP video servers](#) chapter.

vs_Mixer:

configures the TVideoGrabber component as a [video mixer](#), the video source(s) is(are) other(s) TVideoGrabber component(s)

vs_VideoFromImages:

the video source is a set of bitmaps (BMP or JPEG files). See [Video clip from bitmaps: Overview](#).

vs_ThirdPartyFilter:

a third-party DirectShow filter is used as video source

vs_StreamInterface:

the source is raw stream data (e.g. h264) pushed through the StreamInterface_PushData function

See Also

[Video sources supported for preview and recording](#) [VideoSources](#) [VideoSourcesCount](#) [TVideoSource](#)

Created with the Standard Edition of HelpNDoc: [Full-featured EBook editor](#)

VideoSource_FileOrURL

TVideoGrabber.VideoSource_FileOrURL

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the source video clip to use when [VideoSource](#) = vs_VideoFileOrURL

Declaration

property VideoSource_FileOrURL: **String** **read** GetVideoSource_FileOrURL **write** SetVideoSource_FileOrURL;

__property wchar_t *VideoSource_FileOrURL=read=GetVideoSource_FileOrURL, write=SetVideoSource_FileOrURL

Property VideoSource_FileOrURL As String

Description

Specifies the source video clip to use as live source (instead of a video capture device or a web cam) when [VideoSource](#) = vs_VideoFileOrURL

Can be:

- a file located on a local drive, a network drive
- or a streaming video clip specified by its URL
- a playlist.

E.g. to record a video file:

```
videograbber.VideoSource = vs_VideoFileOrURL
videograbber.VideoSource_FileOrURL = 'myclip.avi'
videograbber.StartRecording()
```

E.g. to record a playlist:

```
videograbber.Playlist (pl_Clear, '')
videograbber.Playlist (pl_Add, 'vg000004.avi')
videograbber.Playlist (pl_Add, 'vg000002.avi')
videograbber.Playlist (pl_Add, ...
videograbber.Playlist (pl_Add, ...
videograbber.VideoSource = vs_VideoFileOrURL
```

```
videograbber.VideoSource_FileOrURL = 'PLAYLIST'
videograbber.StartRecording()
```

- alternatively you can invoke [StartPreview](#) instead of [StartRecording](#) to just play the clip (without trackbar handling, if you need the trackbar handling use the [Player](#) features)

- look at the [Recording methods and properties](#) chapter for information about the recording settings

See Also

[VideoSource_FileOrURL_StartTime](#) [VideoSource_FileOrURL_StopTime](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your CHM Help File Capabilities with HelpNDoc](#)

VideoSource_FileOrURL_StartTime

TVideoGrabber.VideoSource_FileOrURL_StartTime

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the start time of the [VideoSource_FileOrUrl](#) video clip.

Declaration

property VideoSource_FileOrURL_StartTime: LargeInteger **read** GetVideoSource_FileOrURL_StartTime
write SetVideoSource_FileOrURL_StartTime **default** DEF_VideoSource_FileOrURL_StartTime;

__property __int64 VideoSource_FileOrURL_StartTime=read=GetVideoSource_FileOrURL_StartTime,
write=SetVideoSource_FileOrURL_StartTime, **default**=- 1

Property VideoSource_FileOrURL_StartTime As Double

Description

Specifies the start time of the [VideoSource_FileOrUrl](#) video clip when [VideoSource](#) = vs_VideoFileOrUrl.
The start time is expressed in 100ns units (e.g. 20000000 = 2 seconds).
When it is set to -1 the video clip starts from the beginning.

See Also

[VideoSource_FileOrURL_VideoSource_FileOrURL_StopTime](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Productivity with HelpNDoc's CHM Help File Creation Features](#)

VideoSource_FileOrURL_StopTime

TVideoGrabber.VideoSource_FileOrURL_StopTime

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Specifies the stop time of the [VideoSource_FileOrUrl](#) video clip.

Declaration

property VideoSource_FileOrURL_StopTime: LargeInteger **read** GetVideoSource_FileOrURL_StopTime
write SetVideoSource_FileOrURL_StopTime **default** DEF_VideoSource_FileOrURL_StopTime;

__property __int64 VideoSource_FileOrURL_StopTime=read=GetVideoSource_FileOrURL_StopTime,
write=SetVideoSource_FileOrURL_StopTime, **default**=- 1

Property VideoSource_FileOrURL_StopTime As Double

Description

Specifies the stop time of the [VideoSource_FileOrUrl](#) video clip when [VideoSource](#) = vs_VideoFileOrUrl.

The stop time is expressed in 100ns units (e.g. 20000000 = 2 seconds).
When it is set to **-1** the video clip will play until the end.

See Also

[VideoSource](#) [FileOrURL](#) [VideoSource](#) [FileOrURL](#) [StartTime](#)

Created with the Standard Edition of HelpNDoc: [Achieve Professional Documentation Results with a Help Authoring Tool](#)

VideoSources

TVideoGrabber.VideoSources

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

List of the video sources available.

Declaration

property VideoSources: **string read** GetVideoSources;

__property wchar_t *VideoSources=read=GetVideoSources

Property VideoSources As String

Description

List of the video sources available.

Used to select the [VideoSource](#), that is an index in this list.

See Also

[Video sources supported for preview and recording](#) [VideoSource](#) [VideoSourcesCount](#) [TVideoSource](#)

Created with the Standard Edition of HelpNDoc: [Don't be left in the past: convert your WinHelp HLP help files to CHM with HelpNDoc](#)

VideoSourcesCount

TVideoGrabber.VideoSourcesCount

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Number of items in the [VideoSources](#) list.

Declaration

property VideoSourcesCount: LongInt **read** GetVideoSourcesCount;

__property **int** VideoSourcesCount=read=GetVideoSourcesCount, **nodefault**

Property VideoSourcesCount As Long

Description

Returns the number of items in the [VideoSources](#) list.

See Also

[Video sources supported for preview and recording](#) [VideoSource](#) [VideoSources](#) [TVideoSource](#)

Created with the Standard Edition of HelpNDoc: [Say Goodbye to Documentation Headaches with a Help Authoring Tool](#)

VideoStreamNumber

TVideoGrabber.VideoStreamNumber

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Let choose the video stream

Declaration

property VideoStreamNumber: LongInt **read** GetVideoStreamNumber **write** GetVideoStreamNumber
default DEF_VideoStreamNumber;

__property int VideoStreamNumber=read=GetVideoStreamNumber, write=GetVideoStreamNumber,
default=DEF_VideoStreamNumber

Description

This property let choose the video stream when clips have multiple video streams.
The stream number must be in the 0..n-1 range (default = -1)

Created with the Standard Edition of HelpNDoc: [Transform Your Word Doc into a Professional-Quality eBook with HelpNDoc](#)

VideoSubtype

TVideoGrabber.VideoSubtype

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Selects a video subtype.

Declaration

property VideoSubtype: LongInt **read** GetVideoSubtype **write** SetVideoSubtype;

__property int VideoSubtype=read=GetVideoSubtype, write=SetVideoSubtype, **nodefault**

Property VideoSubtype As Long

Description

Used to select a video subtype in the [VideoSubtypes](#) list.
See [VideoSubtypes](#) .

Important:

The current video subtype index is not the same from a video capture device to the other. It is reloaded when the video capture device is selected by assigning the [VideoDevice](#) property.

- any control displaying this value should re-read it from the [OnVideoDeviceSelected](#) event (that occurs when the video device is selected)
- be sure that the video capture device has been selected (by assigning the [VideoDevice](#) property) before setting this property

See [device-dependent properties](#)

See Also

[VideoSubtypeIndex](#) [VideoSubtypes](#) [VideoSubtypesCount](#)

Created with the Standard Edition of HelpNDoc: [Easy EPub and documentation editor](#)

VideoSubtypes

TVideoGrabber.VideoSubtypes

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Properties](#)

List of the video subtypes available on the current video capture device

Declaration

property VideoSubtypes: **string read** GetVideoSubtypes;

__property wchar_t *VideoSubtypes=read=GetVideoSubtypes

Property VideoSubtypes As String

Description

Used to retrieve a string that contains the list of the video subtypes available on the current video capture device.

This list is updated when the [OnVideoDeviceSelected](#) event occurs (when a video capture device is selected with [VideoDevice](#)).

This list can be assigned to list based controls. E.g.:

ComboBox1.Items.Text := VideoGrabber1.VideoSubtypes;

ComboBox1.ItemIndex := VideoGrabber1.VideoSubtype;

Note: as this list is not the same from a video capture device to the other, any code using this value should re-read it from the [OnVideoDeviceSelected](#) event (when a [video capture device](#) is selected).

See Also

[VideoSubtype](#) [VideoSubtypeIndex](#) [VideoSubtypesCount](#)

Created with the Standard Edition of HelpNDoc: [Easily create HTML Help documents](#)

VideoSubtypesCount

TVideoGrabber.VideoSubtypesCount

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Properties](#)

Number of video subtypes.

Declaration

property VideoSubtypesCount: LongInt **read** GetVideoSubtypesCount;

__property **int** VideoSubtypesCount=read=GetVideoSubtypesCount, **nodefault**

Property VideoSubtypesCount As Long

Description

Number of video subtypes in the [VideoSubtypes](#) list for the current video capture device.

See Also

[VideoSubtype](#) [VideoSubtypeIndex](#) [VideoSubtypes](#)

Created with the Standard Edition of HelpNDoc: [Make Documentation a Breeze with HelpNDoc's Clean and Efficient User Interface](#)

VideoVisibleWhenStopped

TVideoGrabber.VideoVisibleWhenStopped

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Properties](#)

ber

Prevent the video window to be closed when the playback or preview ends.

Declaration

property VideoVisibleWhenStopped: Boolean **read** GetVideoVisibleWhenStopped **write** SetVideoVisibleWhenStopped **default** DEF_VideoVisibleWhenStopped;

__property bool VideoVisibleWhenStopped==GetVideoVisibleWhenStopped,
write=SetVideoVisibleWhenStopped, **default**=0;

property VideoVisibleWhenStopped as Boolean

Description

When enabled, keeps the video window active after the the playback or preview ends.
The last video frame remains displayed until [StartPreview](#), [StartRecording](#) or [OpenPlayer](#) is invoked.

See Also

[TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display_Active](#) [Display_AlphaBlendEnabled](#)
[Display_AlphaBlendValue](#) [Display_AutoSize](#) [Display_Embedded](#) [Display_FullScreen](#) [Display_Height](#)
[Display_Left](#) [Display_Monitor](#) [Display_MouseMovesWindow](#) [Display_PanScanRatio](#) [Display_SetLocation](#)
[Display_StayOnTop](#) [Display_Top](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#)
[Display_VideoHeight](#) [Display_VideoPortEnabled](#) [Display_VideoWidth](#) [Display_VideoWindowHandle](#)
[Display_Width](#) [DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#) [DualDisplay_AlphaBlendValue](#)
[DualDisplay_AutoSize](#) [DualDisplay_Embedded](#) [DualDisplay_FullScreen](#) [DualDisplay_Height](#)
[DualDisplay_Left](#) [DualDisplay_Monitor](#) [DualDisplay_MouseMovesWindow](#) [DualDisplay_PanScanRatio](#)
[DualDisplay_StayOnTop](#) [DualDisplay_Top](#) [DualDisplay_TransparentColorEnabled](#)
[DualDisplay_TransparentColorValue](#) [DualDisplay_VideoHeight](#) [DualDisplay_VideoPortEnabled](#)
[DualDisplay_VideoWidth](#) [DualDisplay_VideoWindowHandle](#) [DualDisplay_Visible](#) [DualDisplay_Width](#)
[IsVideoPortAvailable](#) [Monitor_Primary](#) [Index_MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#)
[SetParentWindow](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#)
[VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#)
[VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: Transform Your CHM Help File Creation Process with HelpNDoc

VideoWidth**TVideoGrabber.VideoWidth**
[Prev](#)
[Next](#)
TVideoGrabber **Properties**

Retrieves the width of the native video source, in pixels.

Declaration

property VideoWidth: LongInt **read** GetVideoWidth;

__property **int** VideoWidth=read=GetVideoWidth, **nodefault**

Property VideoWidth As Long

Description

Used to retrieve the width of the native video source, in pixels.

See Also

[TOnFrameProgress](#) [TOnResizeVideo](#) [AutoRefreshPreview](#) [BackgroundColor](#) [Display_AutoSize](#)
[Display_FullScreen](#) [Display_SetLocation](#) [Display_VideoPortEnabled](#) [ImageRatio](#) [InFrameProgressEvent](#)
[IsVideoInterlaced](#) [IsVideoPortAvailable](#) [OnFrameProgress](#) [OnKeyPress](#) [OnPreviewStarted](#) [OnResizeVideo](#)
[PausePreview](#) [PreviewZoomSize](#) [ResumePreview](#) [StartPreview](#) [StartSynchronized](#) [StopPreview](#)
[Synchronized](#) [UseNearestVideoSize](#) [VideoHeight](#) [VideoRenderer](#) [VideoRendererPriority](#) [Visible](#)

VideoWidth_PreferedAspectRatio

TVideoGrabber.VideoWidth_PreferedAspectRatio

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Retrieves the preferred display width of the video clip, in pixels.

Declaration

property VideoWidth_PreferedAspectRatio: LongInt **read** GetVideoWidth_PreferedAspectRatio;

__property int VideoWidth_PreferedAspectRatio=read=GetVideoWidth_PreferedAspectRatio, **nodefault**

Property VideoWidth_PreferedAspectRatio As Long

Description

Used to retrieve the preferred display width of the video clip, in pixels.

It can be different of the [VideoWidth](#) property if the video clip has a display aspect ratio (DAR) different of the source aspect ratio (SAR)

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TVideoRenderer](#) [TOnPlayerBufferingData](#) [AdjustPixelAspectRatio](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [Display_Active](#) [Display_AlphaBlendEnabled](#) [Display_AlphaBlendValue](#) [Display_AutoSize](#) [Display_Embedded](#) [Display_FullScreen](#) [Display_Height](#) [Display_Left](#) [Display_Monitor](#) [Display_MouseMovesWindow](#) [Display_PanScanRatio](#) [Display_SetLocation](#) [Display_StayOnTop](#) [Display_Top](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [Display_VideoHeight](#) [Display_VideoPortEnabled](#) [Display_VideoWidth](#) [Display_VideoWindowHandle](#) [DualDisplay_Width](#) [DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#) [DualDisplay_AlphaBlendValue](#) [DualDisplay_AutoSize](#) [DualDisplay_Embedded](#) [DualDisplay_FullScreen](#) [DualDisplay_Height](#) [DualDisplay_Left](#) [DualDisplay_Monitor](#) [DualDisplay_MouseMovesWindow](#) [DualDisplay_PanScanRatio](#) [DualDisplay_StayOnTop](#) [DualDisplay_Top](#) [DualDisplay_TransparentColorEnabled](#) [DualDisplay_TransparentColorValue](#) [DualDisplay_VideoHeight](#) [DualDisplay_VideoPortEnabled](#) [DualDisplay_VideoWidth](#) [DualDisplay_VideoWindowHandle](#) [DualDisplay_Visible](#) [DualDisplay_Width](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [IsVideoPortAvailable](#) [Last_Clip_Played](#) [Monitor_Primary](#) [Index_MonitorBounds](#) [MonitorsCount](#) [MP4NeedsReindexing](#) [OnLeavingFullScreen](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchron](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [SetParentWindow](#) [SetWindowTransparency](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized_VideoDoubleBuffered](#) [VideoHeight_PreferedAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#)

Visible

TVideoGrabber.Visible

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Properties](#)

Used to make the component visible / not visible.

Declaration

property Visible;

__property Visible

Description

Enable / disable the Visible property to make the component visible or not visible.

See Also

[TOnFrameProgress](#) [TOnResizeVideo](#) [AutoRefreshPreview](#) [BackgroundColor](#) [Display](#) [AutoSize](#) [Display_FullScreen](#) [Display_SetLocation](#) [Display_VideoPortEnabled](#) [ImageRatio](#) [InFrameProgressEvent](#) [IsVideoInterlaced](#) [IsVideoPortAvailable](#) [OnFrameProgress](#) [OnKeyPress](#) [OnPreviewStarted](#) [OnResizeVideo](#) [PausePreview](#) [PreviewZoomSize](#) [ResumePreview](#) [StartPreview](#) [StartSynchronized](#) [StopPreview](#) [Synchronized](#) [UseNearestVideoSize](#) [VideoHeight](#) [VideoRenderer](#) [VideoRendererPriority](#) [VideoWidth](#)

Created with the Standard Edition of HelpNDoc: [Free CHM Help documentation generator](#)

VUMeter

TVideoGrabber.VUMeter

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enable the audio VU-meters.

Declaration

property VUMeter: TVUMeter **read** GetVUMeter **write** SetVUMeter **default** DEF_VUMeter;

__property TVUMeter VUMeter=read=GetVUMeter, write=SetVUMeter, **default**=0

Property VUMeter As TxVUMeter

Description

Used to enable the audio VU-Meters.

Possible values:

vu_Disabled : VU-Meters disabled
vu_Analog : analog VU-Meters
vu_Bargraph : bargraph VU-Meters

See the "[Audio levels and VU-Meters](#)" chapter, that explains how to activate the VU-Meters.

See Also

[TVUMeter](#) [TVUMeterSetting](#) [AudioBalance](#) [AudioChannelRenderMode](#) [AudioVolume](#) [MuteAudioRendering](#) [SpeakerBalance](#) [SpeakerControl](#) [SpeakerVolume](#)

Created with the Standard Edition of HelpNDoc: [Leave the tedious WinHelp HLP to CHM conversion process behind with HelpNDoc](#)

WebcamStillCaptureButton

TVideoGrabber.WebcamStillCaptureButton

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Enable the frame capture by using the webcam's still capture button

Declaration

property WebcamStillCaptureButton: TWebcamStillCaptureButton **read** GetWebcamStillCaptureButton **write** SetWebcamStillCaptureButton **default** DEF_WebcamStillCaptureButton;

__**property** TWebcamStillCaptureButton WebcamStillCaptureButton=read=GetWebcamStillCaptureButton, write=SetWebcamStillCaptureButton, **default**=0

Property WebcamStillCaptureButton As TxWebcamStillCaptureButton

Description

Used to enable the frame capture when the webcam includes a still capture button.

The type of the property is [TWebcamStillCaptureButton](#) , possible values:

wb_Disabled:

the still capture button is disabled

wb_Enabled :

the still capture button is enabled, when pressed a frame will be captured.

The captured frame will be returned by the [OnFrameCaptureCompleted](#) event, according to a memory bitmap, a BMP file or a JPEG file according to the [BurstType](#) property (the filename structure depends on the [AutoFileName](#) property)

See Also

[WDM drivers](#) [TAutoFileName](#) [TFrameCaptureDest](#) [TFrameGrabberRGBFormat](#) [TOnFrameCaptureCompleted](#) [AutoConnectRelatedPins](#) [BurstCount](#) [BurstInterval](#) [BurstMode](#) [BurstType](#) [CaptureFrameSyncTo](#) [CaptureFrameTo](#) [CurrentFrameRate](#) [DeliveredFrames](#) [DroppedFrames](#) [FixFlickerOrBlackCapture](#) [FrameCaptureHeight](#) [FrameCaptureWidth](#) [FrameCaptureWithoutOverlay](#) [FrameCaptureZoomSize](#) [FrameGrabber](#) [FrameGrabberCurrentRGBFormat](#) [FrameGrabberRGBFormat](#) [FramerateDivider](#) [GetFrameInfo](#) [GetFrameInfoString](#) [GetLastFrameAsHBITMAP](#) [GetLastFrameAsTBitmap](#) [GetLastFrameBitmapBits](#) [GetLastFrameBitmapBits2](#) [GetLastFrameWaitTimeoutMs](#) [InFrameProgressEvent](#) [IsDVDevice](#) [IsVideoControlAvailable](#) [IsVideoCrossbarAvailable](#) [IsVideoDeviceConnected](#) [IsWDMVideoDriver](#) [JPEGPerformance](#) [JPEGProgressiveDisplay](#) [JPEGQuality](#) [Last_BurstFrameCapture](#) [FileName](#) [Last_CaptureFrameTo_FileName](#) [OnDiskFull](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronone](#) [OnNoVideoDevices](#) [OnVideoDeviceSelected](#) [RefreshDevicesAndCompressorsLists](#) [ResetVideoDeviceSettings](#) [SetFrameCaptureBounds](#) [ShowDialog](#) [StoragePath](#) [VideoDevice](#) [VideoDeviceIndex](#) [VideoDeviceIndexFromId](#) [VideoDeviceName](#) [VideoDevices](#) [VideoDevicesCount](#) [VideoDevicesId](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Create Encrypted, Password-Protected PDFs](#)

ZoomCoeff

TVideoGrabber.ZoomCoeff

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Zooming coefficient X 1000

Declaration

property ZoomCoeff: LongInt **read** GetZoomCoeff **write** SetZoomCoeff;

__**property** **int** ZoomCoeff=read=GetZoomCoeff, write=SetZoomCoeff, **nodefault**

Property ZoomCoeff As Long

Description

Zooming coefficient, expressed as per thousand.
e.g:

1000 = no zooming (default value)

1500 = 1.5x zoom

2000 = 2x zoom
etc...

The [frame grabber](#) must be enabled to use this feature.

Created with the Standard Edition of HelpNDoc: [What is a Help Authoring tool?](#)

ZoomXCenter

TVideoGrabber.ZoomXCenter

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

X center position of the zooming

Declaration

property ZoomXCenter: LongInt **read** GetZoomXCenter **write** SetZoomXCenter;

__property **int** ZoomXCenter=read=GetZoomXCenter, write=SetZoomXCenter, **nodefault**

Property ZoomXCenter As Long

Description

Secifies the vertical center position when the [zooming](#) is used, expressed in pixels.

0 means the vertical center of the video frame (default value)

- 50 means that the vertical center of the zooming is shifted of 50 pixels in the left direction

+ 50 means that the vertical center of the zooming is shifted of 50 pixels in the right direction

Created with the Standard Edition of HelpNDoc: [Generate EPub eBooks with ease](#)

ZoomYCenter

TVideoGrabber.ZoomYCenter

[Prev](#)

[TVideoGrabber](#) [Properties](#)

Y center position of the zooming

Declaration

property ZoomYCenter: LongInt **read** GetZoomYCenter **write** SetZoomYCenter;

__property **int** ZoomYCenter=read=GetZoomYCenter, write=SetZoomYCenter, **nodefault**

Property ZoomYCenter As Long

Description

Secifies the horizontal center position when the [zooming](#) is used, expressed in pixels.

0 means the horizontal center of the video frame (default value)

- 50 means that the vertical center of the zooming is shifted of 50 pixels in the top direction

+ 50 means that the vertical center of the zooming is shifted of 50 pixels in the bottom direction

Created with the Standard Edition of HelpNDoc: [Easy EBook and documentation generator](#)

Methods

TVideoGrabber Methods

[TVideoGrabber](#)

Public

[About](#)
[AnalogVideoStandardIndex](#)
[ASFStreaming_GetAuthorizationList](#)
[ASFStreaming_GetConnectedClients](#)
[ASFStreaming_GetConnectedClientsCount](#)
[ASFStreaming_ResetAuthorizations](#)
[ASFStreaming_SetAuthorization](#)
[AssociateMultiplexedSlave](#)
[AudioCompressorIndex](#)
[AudioDeviceIndex](#)
[AudioInputIndex](#)
[AudioRendererIndex](#)
[AVIDuration](#)
[AVIHeaderInfo](#)
[AVIInfo](#)
[AVIInfo2](#)
[CameraControlAuto](#)
[CameraControlDefault](#)
[CameraControlMax](#)
[CameraControlMin](#)
[CameraControlStep](#)
[CameraControlValue](#)
[Cancel](#)
[CanProcessMessages](#)
[CaptureFrameRenderedTo](#)
[CaptureFrameSyncTo](#)
[CaptureFrameTo](#)
[ClearHeaderAttributes](#)
[ClosePlayer](#)
[Create](#)
[CreatePreallocCapFile](#)
[Decrypt_File](#)
[Destroy](#)
[Display_SetLocation](#)
[DrawBitmapOverFrame](#)
[DualDisplay_SetLocation](#)
[DVDInfo](#)
[EnableMultiplexedInput](#)
[EnableThreadMode](#)
[Encoder_CloseOutputFile](#)
[Encoder_GetInt](#)
[Encoder_NewOutputFile](#)
[Encoder_Pause](#)
[Encoder_Resume](#)
[Encoder_SetInt](#)
[Encoder_SetStr](#)
[Encoders_CreateInstanceForRecording](#)
[Encoders_CreateInstanceForStreaming](#)
[Encoders_RemoveAllInstances](#)
[Encoders_RemoveInstance](#)
[Encrypt_File](#)
[EnumerateWindows](#)
[Facebook_GoLive_ReturnStreamURL](#)
[FastForwardPlayer](#)
[FindIndexInListByName](#)
[GetAudioCodec](#)
[GetCameraExposure](#)
[GetCameraExposureAsString](#)
[GetDisplayActive](#)
[GetDisplayAlphaBlendEnabled](#)
[GetDisplayAlphaBlendValue](#)

[GetDisplayAspectRatio](#)
[GetDisplayAutoSize](#)
[GetDisplayEmbedded](#)
[GetDisplayFullScreen](#)
[GetDisplayHeight](#)
[GetDisplayLeft](#)
[GetDisplayMonitor](#)
[GetDisplayMouseMovesWindow](#)
[GetDisplayPanScanRatio](#)
[GetDisplayStayOnTop](#)
[GetDisplayTop](#)
[GetDisplayTransparentColorEnabled](#)
[GetDisplayTransparentColorValue](#)
[GetDisplayVideoHeight](#)
[GetDisplayVideoPortEnabled](#)
[GetDisplayVideoWidth](#)
[GetDisplayVideoWindowHandle](#)
[GetDisplayVisible](#)
[GetDisplayWidth](#)
[GetFrameInfo](#)
[GetFrameInfoString](#)
[GetFWCam1394](#)
[GetFWCam1394List](#)
[GetImageOverlay_AlphaBlend](#)
[GetImageOverlay_AlphaBlendValue](#)
[GetImageOverlay_ChromaKey](#)
[GetImageOverlay_ChromaKeyLeewayPercent](#)
[GetImageOverlay_ChromaKeyRGBColor](#)
[GetImageOverlay_Enabled](#)
[GetImageOverlay_Height](#)
[GetImageOverlay_LeftLocation](#)
[GetImageOverlay_RotationAngle](#)
[GetImageOverlay_StretchToVideoSize](#)
[GetImageOverlay_TargetDisplay](#)
[GetImageOverlay_TopLocation](#)
[GetImageOverlay_Transparent](#)
[GetImageOverlay_TransparentColorValue](#)
[GetImageOverlay_UseTransparentColor](#)
[GetImageOverlay_Width](#)
[GetItemNameFromList](#)
[GetLastAverageStreamValue](#)
[GetLastErrorMessage](#)
[GetLastFrameAsHBITMAP](#)
[GetLastFrameAsTBitmap](#)
[GetLastFrameBitmapBits](#)
[GetLastFrameBitmapBits2](#)
[GetLogString](#)
[GetMiscDeviceControl](#)
[GetNearestVideoHeight](#)
[GetNearestVideoSize](#)
[GetNearestVideoWidth](#)
[GetPixelsDistance](#)
[GetPlaylist](#)
[GetRGBPixelAt](#)
[GetTextOverlay_Align](#)
[GetTextOverlay_AlphaBlend](#)
[GetTextOverlay_AlphaBlendValue](#)
[GetTextOverlay_BkColor](#)
[GetTextOverlay_Enabled](#)
[GetTextOverlay_Font](#)
[GetTextOverlay_GradientColor](#)
[GetTextOverlay_GradientMode](#)
[GetTextOverlay_HighResFont](#)

[GetTextOverlay_Left](#)
[GetTextOverlay_Right](#)
[GetTextOverlay_Scrolling](#)
[GetTextOverlay_ScrollingSpeed](#)
[GetTextOverlay_Shadow](#)
[GetTextOverlay_ShadowColor](#)
[GetTextOverlay_ShadowDirection](#)
[GetTextOverlay_String](#)
[GetTextOverlay_TargetDisplay](#)
[GetTextOverlay_Top](#)
[GetTextOverlay_Transparent](#)
[GetTranslatedCoordinates](#)
[GetTVChannelInfo](#)
[GetVideoCodec](#)
[GetVideoCompressionSettings](#)
[GetVideoControlMode](#)
[GetVideoHeightFromIndex](#)
[GetVideoSizeFromIndex](#)
[GetVideoWidthFromIndex](#)
[GetVMR9ImageAdjustmentBounds](#)
[GetVUMeterSetting](#)
[GraphState](#)
[IsAudioDeviceConnected](#)
[IsAudioRendererConnected](#)
[IsCameraControlSettingAvailable](#)
[IsDialogAvailable](#)
[IsDVDevice](#)
[IsPlaylistActive](#)
[IsURLResponding](#)
[IsURLVideoStreamAvailable](#)
[IsVideoControlModeAvailable](#)
[IsVideoDeviceConnected](#)
[IsVideoQualitySettingAvailable](#)
[IsVideoSignalDetected](#)
[IsVMR9ImageAdjustmentAvailable](#)
[LoadCompressorSettingsFromDataString](#)
[LoadCompressorSettingsFromTextFile](#)
[MixAudioSamples](#)
[Mixer_Activation](#)
[Mixer_AddToMixer](#)
[Mixer_RemoveFromMixer](#)
[Mixer_SetOverlayRoundedCorner](#)
[Mixer_SetupPIPFromSource](#)
[Monitor_Primary_Index](#)
[MonitorBounds](#)
[MonitorsCount](#)
[MotionDetector_CellColorIntensity](#)
[MotionDetector_CellMotionRatio](#)
[MotionDetector_EnumGridDialogControls](#)
[MotionDetector_Get2DTextGrid](#)
[MotionDetector_Get2DTextMotion](#)
[MotionDetector_GetCellLocation](#)
[MotionDetector_GetCellSensitivity](#)
[MotionDetector_GetCellSize](#)
[MotionDetector_GlobalColorIntensity](#)
[MotionDetector_GloballyIncOrDecSensitivity](#)
[MotionDetector_Reset](#)
[MotionDetector_ResetGlobalSensitivity](#)
[MotionDetector_SetCellSensitivity](#)
[MotionDetector_SetGridSize](#)
[MotionDetector_ShowGridDialog](#)
[MotionDetector_TriggerNow](#)
[MotionDetector_UseThisReferenceSample](#)

[MP4NeedsReindexing](#)
[MPEGProgramSetting](#)
[MultipurposeEncoder_QuickConfigure_UDPStreaming_H264](#)
[MultipurposeEncoder_ReindexClip](#)
[NotifyPlayerTrackbarAction](#)
[ONVIF_GetStr](#)
[ONVIF_SetStr](#)
[ONVIFCancelDiscovery](#)
[ONVIFDeviceInfo](#)
[ONVIFDiscoverCameras_IPRange](#)
[ONVIFDiscoverCameras_Multicast](#)
[ONVIFEnumCamerasDiscovered](#)
[ONVIFPTZGetLimits](#)
[ONVIFPTZGetPosition](#)
[ONVIFPTZPreset](#)
[ONVIFPTZSendAuxiliaryCommand](#)
[ONVIFPTZSetPosition](#)
[ONVIFPTZStartMove](#)
[ONVIFPTZStopMove](#)
[ONVIFSnapShot](#)
[OpenDVD](#)
[OpenPlayer](#)
[OpenPlayerAtFramePositions](#)
[OpenPlayerAtTimePositions](#)
[OpenURLAsyncStatus](#)
[PausePlayer](#)
[PausePreview](#)
[PauseRecording](#)
[PlayerFrameStep](#)
[Playlist](#)
[PointGreyConfig](#)
[PutMiscDeviceControl](#)
[RecordingKBytesWrittenToDisk](#)
[RecordToNewFileNow](#)
[RefreshDevicesAndCompressorsLists](#)
[RefreshPlayerOverlays](#)
[ResetVideoDeviceSettings](#)
[ResumePreview](#)
[ResumeRecording](#)
[RetrieveInitialXYAfterRotation](#)
[RewindPlayer](#)
[RunPlayer](#)
[RunPlayerBackwards](#)
[SaveCompressorSettingsToDataString](#)
[SaveCompressorSettingsToTextFile](#)
[ScreenRecordingUsingCoordinates](#)
[SendCameraCommand](#)
[SendDVCommand](#)
[SendImageToVideoFromBitmaps](#)
[SendImageToVideoFromBitmaps2](#)
[SendIPCameraCommand](#)
[SetAudioRendererAdditional](#)
[SetAuthentication](#)
[SetAVIMuxConfig](#)
[SetCameraControl](#)
[SetCameraExposure](#)
[SetDatasteadFilterDllName](#)
[SetDecryptionKey](#)
[SetDisplayActive](#)
[SetDisplayAlphaBlendEnabled](#)
[SetDisplayAlphaBlendValue](#)
[SetDisplayAspectRatio](#)
[SetDisplayAutoSize](#)

[SetDisplayEmbedded](#)
[SetDisplayFullScreen](#)
[SetDisplayHeight](#)
[SetDisplayLeft](#)
[SetDisplayLocation](#)
[SetDisplayMonitor](#)
[SetDisplayMouseMovesWindow](#)
[SetDisplayPanScanRatio](#)
[SetDisplayParent](#)
[SetDisplayStayOnTop](#)
[SetDisplayTop](#)
[SetDisplayTransparentColorEnabled](#)
[SetDisplayTransparentColorValue](#)
[SetDisplayVideoPortEnabled](#)
[SetDisplayVisible](#)
[SetDisplayWidth](#)
[SetEncryptionKey](#)
[SetFFmpegAudioFilter](#)
[SetFFmpegFilter](#)
[SetFrameCaptureBounds](#)
[SetFWCam1394](#)
[SetHeaderAttribute](#)
[SetImageOverlay_AlphaBlend](#)
[SetImageOverlay_AlphaBlendValue](#)
[SetImageOverlay_Attributes](#)
[SetImageOverlay_Attributes2](#)
[SetImageOverlay_ChromaKey](#)
[SetImageOverlay_ChromaKeyLeewayPercent](#)
[SetImageOverlay_ChromaKeyRGBColor](#)
[SetImageOverlay_Enabled](#)
[SetImageOverlay_Height](#)
[SetImageOverlay_LeftLocation](#)
[SetImageOverlay_RotationAngle](#)
[SetImageOverlay_StretchToVideoSize](#)
[SetImageOverlay_TargetDisplay](#)
[SetImageOverlay_TopLocation](#)
[SetImageOverlay_Transparent](#)
[SetImageOverlay_TransparentColorValue](#)
[SetImageOverlay_UseTransparentColor](#)
[SetImageOverlay_Width](#)
[SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromBMPFile2](#)
[SetImageOverlayFromHBitmap](#)
[SetImageOverlayFromHBitmap2](#)
[SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#)
[SetImageOverlayFromJPEGFile](#)
[SetImageOverlayFromJPEGFile2](#)
[SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#)
[SetImageOverlayFromTImage](#)
[SetImageOverlayFromTImage2](#)
[SetIPCameraSetting](#)
[SetLocation](#)
[SetLogoFromBMPFile](#)
[SetLogoFromHBitmap](#)
[SetLogoFromJPEGFile](#)
[SetLogoFromTBitmap](#)
[SetLogoFromTImage](#)
[SetMultiplexerFilterByName](#)
[SetParentWindow](#)
[SetTextOverlay_Align](#)
[SetTextOverlay_AlphaBlend](#)

[SetTextOverlay_AlphaBlendValue](#)
[SetTextOverlay_BkColor](#)
[SetTextOverlay_CustomVar](#)
[SetTextOverlay_Enabled](#)
[SetTextOverlay_Font](#)
[SetTextOverlay_FontColor](#)
[SetTextOverlay_GradientColor](#)
[SetTextOverlay_GradientMode](#)
[SetTextOverlay_HighResFont](#)
[SetTextOverlay_Left](#)
[SetTextOverlay_Right](#)
[SetTextOverlay_Scrolling](#)
[SetTextOverlay_ScrollingSpeed](#)
[SetTextOverlay_Shadow](#)
[SetTextOverlay_ShadowColor](#)
[SetTextOverlay_ShadowDirection](#)
[SetTextOverlay_String](#)
[SetTextOverlay_TargetDisplay](#)
[SetTextOverlay_Top](#)
[SetTextOverlay_Transparent](#)
[SetVideoCompressionDefaults](#)
[SetVideoCompressionSettings](#)
[SetVideoControlMode](#)
[SetVideoControlMode2](#)
[SetVideoQuality](#)
[SetVMR9ImageAdjustmentValue](#)
[SetVuMeter_Enabled](#)
[SetVUMeterSetting](#)
[SetWindowRecordingByHandle](#)
[SetWindowRecordingByName](#)
[SetWindowTransparency](#)
[ShapeOverlayList](#)
[ShowDebugWindow](#)
[ShowDialog](#)
[StartAudioRecording](#)
[StartAudioRendering](#)
[StartPreview](#)
[StartRecording](#)
[StartReencoding](#)
[StartSynchronized](#)
[Stop](#)
[StopPlayer](#)
[StopPreview](#)
[StopRecording](#)
[StopReencoding](#)
[StreamInterface_PushData](#)
[TextOverlay_CreateCustomFont](#)
[TextOverlay_CreateCustomFont2](#)
[ThirdPartyFilter_AddToList](#)
[ThirdPartyFilter_ClearList](#)
[ThirdPartyFilter_Enable](#)
[ThirdPartyFilter_RemoveFromList](#)
[ThirdPartyFilter_ShowDialog](#)
[TVClearFrequencyOverrides](#)
[TVGetMinMaxChannels](#)
[TVSetChannelFrequencyOverride](#)
[TVStartAutoScan](#)
[TVStopAutoScan](#)
[UseNearestVideoSize](#)
[v360_AddYawPitchRoll](#)
[v360_GetAngle](#)
[v360_GetYawPitchRoll](#)
[v360_ResetAnglesToDefault](#)

[v360_SetAngle](#)
[v360_SetInterpolation](#)
[v360_SetProjection](#)
[v360_SetStereoFormat](#)
[v360_SetTranspose](#)
[v360_SetYawPitchRoll](#)
[VideoCompressorIndex](#)
[VideoDeviceIndex](#)
[VideoDeviceIndexFromId](#)
[VideoFormatIndex](#)
[VideoFromImages_CreateSetOfBitmaps](#)
[VideoInputIndex](#)
[VideoQualityAuto](#)
[VideoQualityDefault](#)
[VideoQualityMax](#)
[VideoQualityMin](#)
[VideoQualityStep](#)
[VideoQualityValue](#)
[VideoSizeIndex](#)
[VideoSubtypeIndex](#)
[WriteScriptCommand](#)

Created with the Standard Edition of HelpNDoc: [Easily create iPhone documentation](#)

About

TVideoGrabber.About

[Next](#)

[TVideoGrabber](#)
[ber](#) [Methods](#)

Shows the current version of TVideoGrabber

Declaration

procedure About;

void __fastcall About(**void**)

Sub About()

Description

Invokes a dialog that displays the current version of TVideoGrabber.

Created with the Standard Edition of HelpNDoc: [Full-featured EPub generator](#)

AnalogVideoStandardIndex

TVideoGrabber.AnalogVideoStandardIndex

[Prev](#)

[Next](#)

[TVideoGrabber](#)
[ber](#) [Methods](#)

Returns the index of a given analog video standard in the [AnalogVideoStandards](#) list.

Declaration

function AnalogVideoStandardIndex(Value: **String**): LongInt;

int __fastcall AnalogVideoStandardIndex(wchar_t *Value)

Function AnalogVideoStandardIndex(param1 As String) As Long

Description

Used to retrieve the index of a given analog video standard in the [AnalogVideoStandards](#) list.

The function accepts wild chars. E.g.:

```
VideoGrabber1.AnalogVideoStandard := AnalogVideoStandardIndex ('PAL B'); // selects the
VideoGrabber1.AnalogVideoStandard := AnalogVideoStandardIndex ('PAL*'); // selects the
VideoGrabber1.AnalogVideoStandard := AnalogVideoStandardIndex ('*PAL*'); // selects the
```

See Also

[Analog Video standards](#) [AnalogVideoStandard](#) [AnalogVideoStandards](#) [AnalogVideoStandardsCount](#)
[IsAnalogVideoDecoderAvailable](#)

Created with the Standard Edition of HelpNDoc: [Benefits of a Help Authoring Tool](#)

ASFStreaming_GetAuthorizationList

TVideoGrabber.ASFStreaming_GetAuthorizationList

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns the list of the ASF streaming authorizations.

Declaration

function ASFStreaming_GetAuthorizationList: **string**;

wchar_t * __fastcall ASFStreaming_GetAuthorizationList()

Function ASFStreaming_GetAuthorizationList as String

Description

Returns the list of the current ASF streaming authorizations (allowed or blocked).

E.g.:

```
block: 192.168.0.53/255.255.255.255
block: 192.168.0.194/255.255.255.255
```

See Also

[ASFAudioBitRate](#) [ASFAudioChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#)
[ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#)
[ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#)
[ASFProfileFromCustomFile](#) [ASFProfiles](#) [ASFProfilesCount](#) [ASFProfileVersion](#)
[ASFStreaming_GetConnectedClients](#) [ASFStreaming_GetConnectedClientsCount](#)
[ASFStreaming_ResetAuthorizations](#) [ASFStreaming_SetAuthorization](#) [ASFVideoBitRate](#)
[ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#)

Created with the Standard Edition of HelpNDoc: [Transform Your CHM Help File Creation Process with HelpNDoc](#)

ASFStreaming_GetConnectedClients

TVideoGrabber.ASFStreaming_GetConnectedClients

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns the list of the clients currently connected to the ASF streaming.

Declaration

function ASFStreaming_GetConnectedClients: **string**;

wchar_t * __fastcall ASFStreaming_GetConnectedClients()

Function ASFStreaming_GetConnectedClients as String

Description

Returns the list of the clients currently connected to the ASF streaming (IP address and port).

E.g.:

```
192.168.0.121:4180
192.168.0.51:4685
```

See Also

[ASFVideoBitRate](#) [ASFVideoChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#)
[ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#)
[ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#)
[ASFProfileFromCustomFile](#) [ASFProfiles](#) [ASFProfilesCount](#) [ASFProfileVersion](#)
[ASFStreaming_GetAuthorizationList](#) [ASFStreaming_GetConnectedClientsCount](#)
[ASFStreaming_ResetAuthorizations](#) [ASFStreaming_SetAuthorization](#) [ASFVideoBitRate](#)
[ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#)

Created with the Standard Edition of HelpNDoc: Full-featured Documentation generator

ASFStreaming_GetConnectedClientsCount

TVideoGrabber.ASFStreaming_GetConnectedClientsCount

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns the number of clients currently connected to the ASF streaming.

Declaration

function ASFStreaming_GetConnectedClientsCount: LongInt;

int __fastcall ASFStreaming_GetConnectedClientsCount(**void**)

Function ASFStreaming_GetConnectedClientsCount as Long

Description

Returns the number of clients currently connected to the ASF streaming.

See Also

[ASFVideoBitRate](#) [ASFVideoChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#)
[ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#)
[ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#)
[ASFProfileFromCustomFile](#) [ASFProfiles](#) [ASFProfilesCount](#) [ASFProfileVersion](#)
[ASFStreaming_GetAuthorizationList](#) [ASFStreaming_GetConnectedClients](#)
[ASFStreaming_ResetAuthorizations](#) [ASFStreaming_SetAuthorization](#) [ASFVideoBitRate](#)
[ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#)

Created with the Standard Edition of HelpNDoc: Effortlessly Create High-Quality Help Documentation with a Help Authoring Tool

ASFStreaming_ResetAuthorizations

TVideoGrabber.ASFStreaming_ResetAuthorizations

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Erase all the current ASF streaming authorizations.

Declaration

function ASFStreaming_ResetAuthorizations: Boolean;

bool __fastcall ASFStreaming_ResetAuthorizations(**void**)

Public Overridable Sub ASFStreaming_ResetAuthorizations()

Description

Used to erase the current ASF streaming authorizations that have been set by invoking [ASFStreaming_SetAuthorization](#).

See Also

[ASFVideoBitRate](#) [ASFVideoChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#)
[ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#)
[ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#)
[ASFProfileFromCustomFile](#) [ASFProfiles](#) [ASFProfilesCount](#) [ASFProfileVersion](#)
[ASFStreaming_GetAuthorizationList](#) [ASFStreaming_GetConnectedClients](#)
[ASFStreaming_GetConnectedClientsCount](#) [ASFStreaming_SetAuthorization](#) [ASFVideoBitRate](#)
[ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#)

Created with the Standard Edition of HelpNDoc: Streamline Your Documentation Process with HelpNDoc's Project Analyzer

ASFStreaming_SetAuthorization

TVideoGrabber.ASFStreaming_SetAuthorization

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Prevents an IP address to connect to the ASF streaming.

Declaration

function ASFStreaming_SetAuthorization(Allowed: Boolean; IP: **String**; Mask: **String**): Boolean;

bool __fastcall ASFStreaming_SetAuthorization(**bool** Allowed, wchar_t *IP, wchar_t *Mask)

Function ASFStreaming_SetAuthorization (Allowed as Boolean, IP as String, Mask as String) as Boolean

Description

Used to block the ASF streaming for the specified IP address.

E.g:

- to block the 192.168.0.50 ip address:

```
ASFStreaming_SetAuthorization (false, '192.168.0.51', '255.255.255.255');
```

- to block all the IP addresses on 192.168.0.xxx:

```
ASFStreaming_SetAuthorization (false, '192.168.0.0', '255.255.255.0');
```

See Also

[ASFVideoBitRate](#) [ASFVideoChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#)
[ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#)
[ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#)

[ASFProfileFromCustomFile](#) [ASFProfiles](#) [ASFProfilesCount](#) [ASFProfileVersion](#)
[ASFStreaming_GetAuthorizationList](#) [ASFStreaming_GetConnectedClients](#)
[ASFStreaming_GetConnectedClientsCount](#) [ASFStreaming_ResetAuthorizations](#) [ASFVideoBitRate](#)
[ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#)

Created with the Standard Edition of HelpNDoc: [Easily create EPub books](#)

AssociateMultiplexedSlave

TVideoGrabber.AssociateMultiplexedSlave

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Associates a slave to a master component.

Declaration

function AssociateMultiplexedSlave(InputNumber: LongInt; SlaveUniqueID: LongInt): Boolean;

bool __fastcall AssociateMultiplexedSlave (int InputNumber, int SlaveUniqueID)

Function AssociateMultiplexedSlave(InputNumber As Long, SlaveUniqueID As Long) As Boolean

Description

Used in slave/master [multiplexed mode](#) to associate a slave TVideoGrabber component to a master TVideoGrabber component.

See Also

[TMultiplexedRole](#) [EnableMultiplexedInput](#) [MultiplexedInputEmulation](#) [MultiplexedRole](#)
[MultiplexedStabilizationDelay](#) [MultiplexedSwitchDelay](#) [UniqueID](#)

Created with the Standard Edition of HelpNDoc: [Produce electronic books easily](#)

AudioCompressorIndex

TVideoGrabber.AudioCompressorIndex

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns the index of a given compressor in the [AudioCompressors](#) list.

Declaration

function AudioCompressorIndex(Value: **String**): LongInt;

int __fastcall AudioCompressorIndex(wchar_t *Value)

Function AudioCompressorIndex(Value As String) As Long

Description

Used to retrieve the index of a given compressor in the [AudioCompressors](#) list.

The function accepts wild chars. E.g.:

```
VideoGrabber1.AudioCompressor := AudioCompressorIndex ('MPEG Layer-3'); // selects the 'MPEG Layer-3' compressor
VideoGrabber1.AudioCompressor := AudioCompressorIndex ('MPEG*');           // selects the first 'MPEG' compressor
VideoGrabber1.AudioCompressor := AudioCompressorIndex ('*MPEG*');         // selects the last 'MPEG' compressor
```

See Also

[Recording methods and properties](#) [TCompressionType](#) [TOnVideoCompressionSettings](#) [AudioCompressor](#)
[AudioCompressorName](#) [AudioCompressors](#) [AudioCompressorsCount](#) [CompressionMode](#)

[CompressionType](#) [GetVideoCompressionSettings](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [OnVideoCompressionSettings](#) [RefreshDevicesAndCompressorsLists](#) [SaveCompressorSettingsToDataString](#) [SetVideoCompressionDefaults](#) [SetVideoCompressionSettings](#) [VideoCompression](#) [KeyFrameRate](#) [VideoCompression](#) [PFramesPerKeyFrame](#) [VideoCompression](#) [Quality](#) [VideoCompression](#) [WindowSize](#) [VideoCompressor](#) [VideoCompressorIndex](#) [VideoCompressorName](#) [VideoCompressors](#) [VideoCompressorsCount](#)

Created with the Standard Edition of HelpNDoc: [Easily Add Encryption and Password Protection to Your PDFs](#)

AudioDeviceIndex

TVideoGrabber.AudioDeviceIndex

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns the index of a given audio device in the [AudioDevices](#) list.

Declaration

function AudioDeviceIndex(Value: **String**): LongInt;

int __fastcall AudioDeviceIndex(wchar_t *Value)

Function AudioDeviceIndex(Value As String) As Long

Description

Used to retrieve the index of a given audio device in the [AudioDevices](#) list.

The function accepts wild chars. E.g.:

```
VideoGrabber1.AudioDevice := AudioDeviceIndex ('Hauppauge WinTV USB Pro Audio (WDM)'); //
VideoGrabber1.AudioDevice := AudioDeviceIndex ('Hauppauge*'); //
VideoGrabber1.AudioDevice := AudioDeviceIndex ('*Hauppauge*'); //
```

See Also

[TAudioFormat](#) [AssociateAudioAndVideoDevices](#) [AudioBalance](#) [AudioDevice](#) [AudioDeviceName](#) [AudioDeviceRendering](#) [AudioDevices](#) [AudioDevicesCount](#) [AudioFormat](#) [AudioFormats](#) [AudioInput](#) [AudioInputIndex](#) [AudioInputLevel](#) [AudioInputMono](#) [AudioInputs](#) [AudioInputsCount](#) [AudioSource](#) [IsAudioCrossbarAvailable](#) [IsAudioDeviceASoundCard](#) [IsAudioDeviceConnected](#) [IsAudioInputBalanceAvailable](#) [OnAudioBufferNegotiation](#) [OnAudioDeviceSelected](#) [SpeakerBalance](#) [SpeakerControl](#) [SpeakerVolume](#)

Created with the Standard Edition of HelpNDoc: [Free EPub producer](#)

AudioInputIndex

TVideoGrabber.AudioInputIndex

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns the index of a given audio input in the [AudioInputs](#) list.

Declaration

function AudioInputIndex(Value: **String**): LongInt;

int __fastcall AudioInputIndex(wchar_t *Value)

Function AudioInputIndex(Value As String) As Long

Description

Used to retrieve the index of a given audio input in the [AudioInputs](#) list.

The function accepts wild chars. E.g.:

```

VideoGrabber1.AudioInput := AudioInputIndex ('Audio CD'); // selects the "Audio CD" audio
VideoGrabber1.AudioInput := AudioInputIndex ('CD*');      // selects the 1st audio input
VideoGrabber1.AudioInput := AudioInputIndex ('*CD*');     // selects the 1st audio input

```

See Also

[TAudioFormat](#) [AssociateAudioAndVideoDevices](#) [AudioBalance](#) [AudioDevice](#) [AudioDeviceIndex](#)
[AudioDeviceName](#) [AudioDeviceRendering](#) [AudioDevices](#) [AudioDevicesCount](#) [AudioFormat](#) [AudioFormats](#)
[AudioInput](#) [AudioInputLevel](#) [AudioInputMono](#) [AudioInputs](#) [AudioInputsCount](#) [AudioSource](#)
[IsAudioCrossbarAvailable](#) [IsAudioDeviceASoundCard](#) [IsAudioDeviceConnected](#)
[IsAudioInputBalanceAvailable](#) [OnAudioBufferNegotiation](#) [OnAudioDeviceSelected](#) [SpeakerBalance](#)
[SpeakerControl](#) [SpeakerVolume](#)

Created with the Standard Edition of HelpNDoc: [Elevate Your CHM Help Files with HelpNDoc's Advanced Customization Options](#)

AudioRendererIndex

TVideoGrabber.AudioRendererIndex

[Prev](#)[Next](#)

[TVideoGrabber](#)
[ber](#) [Methods](#)

Retrieve the index of a given audio renderer in the [AudioRenderers](#) list.

Declaration

function AudioRendererIndex (Value: **string**): LongInt;

int __fastcall AudioRendererIndex(wchar_t *Value);

Function AudioRendererIndex(Value As String) As Long

Description

Used to retrieve the index of a given audio renderer in the [AudioRenderers](#) list.

The function accepts wild chars. E.g.:

```

VideoGrabber1.AudioRenderer := AudioRendererIndex ('DirectSound: Realtek AC97 Audio');
VideoGrabber1.AudioRenderer := AudioRendererIndex ('*Realtek AC97 Audio');
VideoGrabber1.AudioRenderer := AudioRendererIndex ('*AC97*');

```

See Also

[IsAudioRendererConnected](#)

Created with the Standard Edition of HelpNDoc: [Import and export Markdown documents](#)

AVIDuration

TVideoGrabber.AVIDuration

[Prev](#)[Next](#)

[TVideoGrabber](#)
[ber](#) [Methods](#)

Returns the duration of the AVI file specified.

Declaration

function AVIDuration(AVIFile: **String**; **out** Duration: LargeInteger; **out** FrameCount: LargeInteger): Boolean;

bool __fastcall AVIDuration(wchar_t *AVIFile, __int64 &Duration, __int64 &FrameCount)

Function AVIDuration(AVIFile As String, AVIDuration, AVIFrameCount) As Boolean

Description

Used to retrieve the duration of the AVI file specified.

Duration: returns the duration of the AVI file, expressed in 100 nanoseconds units, returns 0 upon failure.

FrameCount: returns the total frame count if available, otherwise returns 0.

VideoWidth: width of the video

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchronise](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Elevate Your Help Documentation with a Help Authoring Tool](#)

AVIHeaderInfo**TVideoGrabber.AVIHeaderInfo**
[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns the header information about the video clip specified.

Declaration

function AVIHeaderInfo (AVIFile: **string**; HeaderAttribute: THeaderAttribute): **string**;

wchar_t * __fastcall AVIHeaderInfo(wchar_t *AVIFile, THeaderAttribute HeaderAttribute);

Function AVIHeaderInfo (AVIFile As String, HeaderAttribute as THeaderAttribute) as String

Description

Used to retrieve the [THeaderAttribute](#) information about the video clip specified.

When the clip is opened a first time, all the whole [THeaderAttribute](#) attributes are read, so the clip is not re-opened during the subsequent calls, unless a different file name is specified.

E.g.:

```
procedure TfrmMainForm.Button2Click(Sender: TObject);
begin
    Showmessage (VideoGrabber.AVIHeaderInfo('MyFirstVideoClip.avi', ha_Title));
    Showmessage (VideoGrabber.AVIHeaderInfo('MySecondVideoClip.asf', ha_Copyright));
end;
```

See Also

[Player features](#)
[Recording methods and properties](#)
[TAVIMuxConfig](#)
[TOnPlayerStateChanged](#)
[TPlayerState](#)
[TASFDeinterlaceMode](#)
[TAutoFileName](#)
[TOnPlayerBufferingData](#)
[TOnRecordingCompleted](#)
[TOnRecordingReadyToStart](#)
[TRecordingMethod](#)
[TSyncPreview](#)
[AudioChannelRenderMode](#)
[AudioRecording](#)
[AudioStreamNumber](#)
[AudioSyncAdjustment](#)
[AutoFileNameMinDigits](#)
[AutoStartPlayer](#)
[AVIDuration](#)
[AVIDurationUpdated](#)
[AVIFormatOpenDML](#)
[AVIInfo](#)
[AVIInfo2](#)
[ClosePlayer](#)
[Encoder](#)
[SetInt](#)
[FastForwardPlayer](#)
[HoldRecording](#)
[IsPlayerAudioStreamAvailable](#)
[IsPlayerVideoStreamAvailable](#)
[IsRecordingPaused](#)
[Last Clip Played](#)
[Last Recording](#)
[FileName](#)
[MP4NeedsReindexing](#)
[OnBacktimedFramesCountReached](#)
[OnCopyPreallocDataCompleted](#)
[OnCopyPreallocDataProgress](#)
[OnCopyPreallocDataStarted](#)
[OnCreatePreallocFileCompleted](#)
[OnCreatePreallocFileStarted](#)
[OnDiskFull](#)
[OnPlayerBufferingData](#)
[OnPlayerEndOfStream](#)
[OnPlayerStateChanged](#)
[OnPlayerUpdateTrackbarPosition](#)
[OnRecordingCompleted](#)
[OnRecordingPaused](#)
[OnRecordingReadyToStart](#)
[OnRecordingStarted](#)
[OnReencodingCompleted](#)
[OnReencodingStarted](#)
[OpenPlayer](#)
[OpenPlayerAtFramePositions](#)
[OpenPlayerAtTimePositions](#)
[PausePlayer](#)
[PauseRecording](#)
[PlayerAudioCodec](#)
[PlayerAudioRendering](#)
[PlayerDuration](#)
[PlayerDVSize](#)
[PlayerFastSeekSpeedRatio](#)
[PlayerFileName](#)
[PlayerForcedCodec](#)
[PlayerFrameCount](#)
[PlayerFramePosition](#)
[PlayerFrameRate](#)
[PlayerFrameStep](#)
[PlayerRefreshPausedDisplay](#)
[PlayerRefreshPausedDisplayFrameRate](#)
[PlayerSpeedRatio](#)
[PlayerTimePosition](#)
[PlayerTrackBar](#)
[PlayerTrackBarScale](#)
[PlayerTrackBarSynchron](#)
[PlayerVideoCodec](#)
[PreallocCapFileCopiedAfterRecording](#)
[PreallocCapFileEnabled](#)
[PreallocCapFileName](#)
[PreallocCapFileSizeInMB](#)
[RecordingBacktimedFramesCount](#)
[RecordingCanPause](#)
[RecordingDuration](#)
[RecordingFileName](#)
[RecordingFileSizeMaxInMB](#)
[RecordingFourCC](#)
[RecordingHeight](#)
[RecordingInNativeFormat](#)
[RecordingKBytesWrittenToDisk](#)
[RecordingMethod](#)
[RecordingSize](#)
[RecordingTimer](#)
[RecordingTimerInterval](#)
[RecordingWidth](#)
[ResumeRecording](#)
[RewindPlayer](#)
[RunPlayer](#)
[RunPlayerBackwards](#)
[SaveCompressorSettingsToDataString](#)
[SetMultiplexerFilterByName](#)
[ShowDialog](#)
[SourceStream](#)
[StartAudioRecording](#)
[StartRecording](#)
[StartSynchronized](#)
[StopPlayer](#)
[StopRecording](#)
[StoragePath](#)
[SynchronizationRole](#)
[Synchronized](#)
[SyncPreview](#)
[VideoHeight](#)
[PreferredAspectRatio](#)
[VideoPlayableWhileRecording](#)
[VideoWidth](#)
[PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Produce online help for Qt applications](#)

AVIInfo

TVideoGrabber.AVIInfo

[Prev](#)

[Next](#)

[TVideoGrabber](#)
[Methods](#)

Returns information about the specified video clip.

Declaration

function AVIInfo (AVIFile: **string**; **out** Duration: LargeInteger; **out** FrameCount: LargeInteger; **out** VideoWidth: LongInt; **out** VideoHeight: LongInt; **out** VideoFrameRateFps: Double; **out** AvgBitRate: LongInt; **out** AudioChannels: LongInt; **out** AudioSamplesPerSec: LongInt; **out** AudioBitsPerSample: LongInt; **out** VideoCodec: string; **out** AudioCodec: string): Boolean;

bool __fastcall AVIInfo (wchar_t *AVIFile, __int64 &Duration, __int64 &FrameCount, **int** &VideoWidth, **int** &VideoHeight, **double** &VideoFrameRateFps, **int** &AvgBitRate, **int** &AudioChannels, **int** &AudioSamplesPerSec, **int** &AudioBitsPerSample, wchar_t *&VideoCodec, wchar_t *&AudioCodec)

Function AVIInfo(AVIFile As String, Duration, FrameCount, VideoWidth, VideoHeight, VideoFrameRateFps, AvgBitRate, AudioChannels, AudioSamplesPerSec, AudioBitsPerSample, VideoCodec, AudioCodec) As Boolean

Description

Used to retrieve information about the video clip specified as parameter.

Duration: returns the duration of the AVI file, expressed in 100 nanoseconds units, returns 0 upon failure,
FrameCount: returns the total frame count if available, otherwise returns 0,
VideoWidth: returns the width of the video

VideoHeight: returns the height of the video

VideoFrameRateFps: returns the frame rate in frames per second

AvgBitRate: returns the average bit rate of the clip

AudioChannels: returns the number of audio channels

AudioSamplesPerSec: returns the audio sample frequency in Hz

AudioBitsPerSample: returns the number of bits per sample

VideoCodec:/cf0 returns the name of the video codec in which the video stream of the clip is encoded (if any)

AudioCodec:/cf0 returns the name of the audio codec in which the audio stream of the clip is encoded (if any)

E.g.:

```
procedure TForm1.Button1Click(Sender: TObject);
var
    Duration: int64;
    FrameCount: int64;
    VideoWidth: LongInt;
    VideoHeight: LongInt;
    VideoCodec: string;
    AudioCodec: string;
    VideoFrameRateFps: Double;
    AvgBitRate: LongInt;
    AudioChannels: LongInt;
    AudioSamplesPerSec: LongInt;
    AudioBitsPerSample: LongInt;
begin
    if VideoGrabber.AVIInfo (edtPlayerClip.Text, Duration, FrameCount, VideoWidth, VideoHeight, VideoCodec, AudioCodec, VideoFrameRateFps, AvgBitRate, AudioChannels, AudioSamplesPerSec, AudioBitsPerSample) then
        ShowMessage('');
        ShowMessage(edtPlayerClip.Text + ':');
        ShowMessage('duration (in sec): ' + FormatFloat ('0.00', Duration / 1000000));
        ShowMessage('frame count: ' + IntToStr (FrameCount));
        ShowMessage('video width: ' + IntToStr (VideoWidth));
        ShowMessage('video height: ' + IntToStr (VideoHeight));
        ShowMessage('video frame rate (fps): ' + FormatFloat ('0.00', VideoFrameRateFps));
        ShowMessage('average bit rate (kb/s): ' + IntToStr (AvgBitRate div 1024));
        ShowMessage('audio channels: ' + IntToStr (AudioChannels));
        ShowMessage('audio samples/sec: ' + IntToStr (AudioSamplesPerSec));
        ShowMessage('audio bits per sample: ' + IntToStr (AudioBitsPerSample));
        ShowMessage('video codec: ' + VideoCodec);
        ShowMessage('audio codec: ' + AudioCodec);
    end
    else begin
        ShowMessage(edtPlayerClip.Text + ' clip not found.');
```

See Also

[Player features](#) [Recording methods and properties](#) [TAVIMuxConfig](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnPlayerBufferingData](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioChannelRenderMode](#) [AudioRecording](#) [AudioStreamNumber](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo2](#) [ClosePlayer](#) [Encoder](#) [SetInt](#) [FastForwardPlayer](#) [HoldRecording](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [IsRecordingPaused](#) [Last Clip Played](#) [Last Recording](#) [FileName](#) [MP4NeedsReindexing](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#)

[OnReencodingCompleted](#) [OnReencodingStarted](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#)
[OpenPlayerAtTimePositions](#) [PausePlayer](#) [PauseRecording](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#)
[PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#)
[PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#)
[PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#)
[PlayerTrackBarScale](#) [PlayerTrackBarSynchronise](#) [PlayerVideoCodec](#) [PreallocCapFileCopiedAfterRecording](#)
[PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#)
[RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#)
[RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#)
[RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#)
[RecordingWidth](#) [ResumeRecording](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#)
[SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [ShowDialog](#) [SourceStream](#)
[StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopPlayer](#) [StopRecording](#) [StoragePath](#)
[SynchronizationRole](#) [Synchronized](#) [SyncPreview](#) [VideoHeight](#) [PreferredAspectRatio](#)
[VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Easily create EPub books](#)

AVIInfo2

TVideoGrabber.AVIInfo2

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Used to retrieve information about the specified video clip.

Declaration

function AVIInfo2 (AVIFile: **string**; AVIInfoType: TAVIInfoType): **string**;

wchar_t * __fastcall AVIInfo2(wchar_t *AVIFile, TAVIInfoType AVIInfoType);

Function AVIInfo2 (AVIFile as string; AVIInfoType as TAVIInfoType) as String

Description

This function returns the [TAVIInfoType](#) value as a string for the video clip specified as parameter.
E.g.:

```
Dim VideoCodec as string
VideoCodec = AVIInfo2 ("myvideoclip.avi", av_VideoCodec)
```

```
Dim ClipDuration as string
ClipDuration = AVIInfo2 ("myvideoclip.avi", av_Duration)
```

Note: if the clip file name does not change, the clip is opened only one time. All the values are read and cached.

Therefore all the subsequent calls to AVIInfo2 with the same file name do not reopen the video clip.

The possible values returned by AVIInfo2 depending on TAVIInfoType are (av_Duration, av_FrameCount, av_VideoWidth, av_VideoHeight, av_VideoFrameRateFps, av_VideoCodec, av_AudioCodec, av_AvgBitRate, av_AudioChannels, av_AudioSamplesPerSec, av_AudioBitsPerSample);

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#)
[AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [ClosePlayer](#) [FastForwardPlayer](#)
[IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last Clip Played](#) [MP4NeedsReindexing](#)
[OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#)
[OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#)
[PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#)
[PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#)
[PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#)
[PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchronise](#) [PlayerVideoCodec](#)
[RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#)

[SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Full-featured Documentation generator](#)

CameraControlAuto

TVideoGrabber.CameraControlAuto

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Reports if a [TCameraControl](#) setting is in "auto" mode.

Declaration

function CameraControlAuto(Setting: TCameraControl): Boolean;

bool __fastcall CameraControlAuto(TCameraControl Setting)

Function CameraControlAuto(Setting As TxCameraControl) As Boolean

Description

Used to know if a [TCameraControl](#) setting is in "auto" mode or "manual" mode. Returns "true" if the setting is in "auto" mode, "false" if it is in "manual" mode.

See Also

[TCameraControl](#) [CameraControlDefault](#) [CameraControlMax](#) [CameraControlMin](#) [CameraControlSettings](#) [CameraControlStep](#) [GetCameraExposure](#) [GetCameraExposureAsString](#) [IsCameraControlAvailable](#) [IsCameraControlSettingAvailable](#) [SendCameraCommand](#) [SetCameraControl](#) [SetCameraExposure](#)

Created with the Standard Edition of HelpNDoc: [Create HTML Help, DOC, PDF and print manuals from 1 single source](#)

CameraControlDefault

TVideoGrabber.CameraControlDefault

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets a camera control setting to its default value.

Declaration

function CameraControlDefault(Setting: TCameraControl): LongInt;

int __fastcall CameraControlDefault(TCameraControl Setting)

Function CameraControlDefault(Setting As TxCameraControl) As Long

Description

Used to set a camera control setting to its default value. Returns true upon success.

See Also

[TCameraControl](#) [CameraControlAuto](#) [CameraControlMax](#) [CameraControlMin](#) [CameraControlSettings](#) [CameraControlStep](#) [GetCameraExposure](#) [GetCameraExposureAsString](#) [IsCameraControlAvailable](#) [IsCameraControlSettingAvailable](#) [SendCameraCommand](#) [SetCameraControl](#) [SetCameraExposure](#)

Created with the Standard Edition of HelpNDoc: [Free Qt Help documentation generator](#)

CameraControlMax

TVideoGrabber.CameraControlMax

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Gets the maximum value of a given [TCameraControl](#) setting.

Declaration

function CameraControlMax(Setting: TCameraControl): LongInt;

int __fastcall CameraControlMax(TCameraControl Setting)

Function CameraControlMax(Setting As TxCameraControl) As Long

Description

Retrieves the maximum value for a given [TCameraControl](#) setting, if available for the current video capture device. Returns MAXINT upon failure.

See Also

[TCameraControl](#) [CameraControlAuto](#) [CameraControlDefault](#) [CameraControlMin](#) [CameraControlSettings](#) [CameraControlStep](#) [GetCameraExposure](#) [GetCameraExposureAsString](#) [IsCameraControlAvailable](#) [IsCameraControlSettingAvailable](#) [SendCameraCommand](#) [SetCameraControl](#) [SetCameraExposure](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your PDF Protection with These Simple Steps](#)

CameraControlMin

TVideoGrabber.CameraControlMin

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Gets the minimum value of a given [TCameraControl](#) setting.

Declaration

function CameraControlMin(Setting: TCameraControl): LongInt;

int __fastcall CameraControlMin(TCameraControl Setting)

Function CameraControlMin(Setting As TxCameraControl) As Long

Description

Retrieves the minimum value for a given [TCameraControl](#) setting, if available for the current video capture device. Returns MAXINT upon failure.

See Also

[TCameraControl](#) [CameraControlAuto](#) [CameraControlDefault](#) [CameraControlMax](#) [CameraControlSettings](#) [CameraControlStep](#) [GetCameraExposure](#) [GetCameraExposureAsString](#) [IsCameraControlAvailable](#) [IsCameraControlSettingAvailable](#) [SendCameraCommand](#) [SetCameraControl](#) [SetCameraExposure](#)

Created with the Standard Edition of HelpNDoc: [Revolutionize Your Documentation Review with HelpNDoc's Project Analyzer](#)

CameraControlStep

TVideoGrabber.CameraControlStep

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Gets the stepping value of a given [TCameraControl](#) setting.

Declaration

function CameraControlStep(Setting: TCameraControl): LongInt;

int __fastcall CameraControlStep(TCameraControl Setting)

Function CameraControlStep(Setting As TxCameraControl) As Long

Description

Retrieves the stepping value for a given [TCameraControl](#) setting, if available for the current video capture device. Returns MAXINT upon failure.

See Also

[TCameraControl](#) [CameraControlAuto](#) [CameraControlDefault](#) [CameraControlMax](#) [CameraControlMin](#) [CameraControlSettings](#) [GetCameraExposure](#) [GetCameraExposureAsString](#) [IsCameraControlAvailable](#) [IsCameraControlSettingAvailable](#) [SendCameraCommand](#) [SetCameraControl](#) [SetCameraExposure](#)

Created with the Standard Edition of HelpNDoc: [Free Web Help generator](#)

CameraControlValue

TVideoGrabber.CameraControlValue

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Retrieves a [TCameraControl](#) current value.

Declaration

function GetCameraControl(Setting: TCameraControl): LongInt;

int __fastcall GetCameraControl(TCameraControl Setting)

Function GetCameraControl(Setting As TxCameraControl) As Long

Description

Used to retrieve the current value of a [TCameraControl](#) setting. Returns MAXINT upon failure.

See Also

[TCameraControl](#) [CameraControlAuto](#) [CameraControlDefault](#) [CameraControlMax](#) [CameraControlMin](#) [CameraControlSettings](#) [CameraControlStep](#) [GetCameraExposure](#) [GetCameraExposureAsString](#) [IsCameraControlAvailable](#) [IsCameraControlSettingAvailable](#) [SendCameraCommand](#) [SetCameraControl](#) [SetCameraExposure](#)

Created with the Standard Edition of HelpNDoc: [Easy CHM and documentation editor](#)

Cancel

TVideoGrabber.Cancel

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Cancels any task currently running.

Declaration

function Cancel: Boolean;

void __fastcall Cancel(**void**)

Function Cancel() As Boolean

Description

Cancels any task currently running (preview, recording, playback as well as reencoding).

Recording

If the recording is running, it is immediately stopped:

- if the recording had to be recompressed after capture ([CompressionMode](#) = `cm_CompressAfterRecording`), this reencoding operation is cancelled,
- if the recording had to be recopied after capture ([PreallocCapFileEnabled](#) = `true`), this copy operation is cancelled.

Reecording

The current reencoding operation is stopped immediately.

Returns true upon success.

Created with the Standard Edition of HelpNDoc: [Achieve Professional Documentation Results with a Help Authoring Tool](#)

CanProcessMessages

TVideoGrabber.CanProcessMessages

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Used to know if message-based processings can be applied

Declaration

function CanProcessMessages: Boolean;

void __fastcall CanProcessMessages(**void**)

Function CanProcessMessages() As Boolean

Description

Used to know if message-based processings can be applied, mainly during the TVideoGrabber events like the [OnFrameCaptureCompleted](#) or **OnFrameOverlayUsing...** events.

This function returns false if the player is running and [UseClock](#) is disabled. In this case you must not invoke message-based code (e.g. like creating and displaying a form) during the [OnFrameCaptureCompleted](#) , [OnFrameProgress](#) or [OnFrameOverlayUsingDC](#) and other **OnFrameOverlayUsing...** events.

Created with the Standard Edition of HelpNDoc: [Maximize Your Reach: Convert Your Word Document to an ePub or Kindle eBook](#)

CaptureFrameRenderedTo

TVideoGrabber.CaptureFrameRenderedTo

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Captures the frame being rendered

Declaration

function CaptureFrameRenderedTo (DisplayIndex: LongInt; Dest: TFrameCaptureDest; FileName: **string**): Boolean;

bool __fastcall CaptureFrameRenderedTo(**int** DisplayIndex, TFrameCaptureDest Dest, System::wchar_t *FileName);

Description

Capture the current frame being displayed in the video renderer.

Works like [CaptureFrameSyncTo](#), but does not require the frame grabber to be enabled, allowing to

capture frames and saving CPU.

This function is supported only by the following [video renderers](#):
vr_EVR, vr_VMR9, vr_VMR7, vr_StandardRenderer, vr_madVR

Note :

Invoking this function with vr_StandardRenderer produces a (very short) pause/resume of the video stream.

It is instant with other video renderers.

See Also

[Frame capture features](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Create Encrypted, Password-Protected PDFs](#)

CaptureFrameSyncTo

TVideoGrabber.CaptureFrameSyncTo

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Invokes [CaptureFrameTo](#) and waits until completion.

Declaration

function CaptureFrameSyncTo(Dest: TFrameCaptureDest; FileName: **String**): Boolean;

bool __fastcall CaptureFrameSyncTo(TFrameCaptureDest Dest, wchar_t *FileName)

Function CaptureFrameSyncTo(Dest As TFrameCaptureDest, FileName As String) As Boolean

Description

Invokes [CaptureFrameTo](#) and waits until the frame to be captured, or the graph stopped.
Returns true upon success, false if the capture failed.

See Also

[TAutoFileName](#) [TFrameCaptureDest](#) [TOnFrameCaptureCompleted](#) [BurstCount](#) [BurstInterval](#) [BurstMode](#) [BurstType](#) [CaptureFrameTo](#) [FrameCaptureHeight](#) [FrameCaptureWidth](#) [FrameCaptureWithoutOverlay](#) [FrameCaptureZoomSize](#) [GetFrameInfoString](#) [GetLastFrameAsHBITMAP](#) [GetLastFrameAsTBitmap](#) [GetLastFrameBitmapBits](#) [GetLastFrameBitmapBits2](#) [GetLastFrameWaitTimeoutMs](#) [JPEGPerformance](#) [JPEGProgressiveDisplay](#) [JPEGQuality](#) [Last](#) [BurstFrameCapture](#) [FileName](#) [Last](#) [CaptureFrameTo](#) [FileName](#) [OnDiskFull](#) [OnFrameBitmap](#) [SetFrameCaptureBounds](#) [StoragePath](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Produce online help for Qt applications](#)

CaptureFrameTo

TVideoGrabber.CaptureFrameTo

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Captures asynchronously the next video frame to memory or to files automatically.

Declaration

function CaptureFrameTo(Dest: TFrameCaptureDest; FileName: **String**): Boolean;

bool __fastcall CaptureFrameTo(TFrameCaptureDest Dest, wchar_t *FileName)

Function CaptureFrameTo(Dest As TFrameCaptureDest, FileName As String) As Boolean

Description

Captures asynchronously the next video frame, depending of the Dest parameter ([TFrameCaptureDest](#)

type):

- **fc_TBitmap**: to a memory Bitmap,
- **fc_BmpFile**: to a BMP file
- **fc_JpegFile**: to a JPEG file,
- **fc_Clipboard**: to the clipboard (CF_BITMAP format)

The [frame grabber](#) must be enabled to use this function.

The video frame is returned asynchronously by the [OnFrameCaptureCompleted](#) event.

To let TVideoGrabber generate automatically the file name when capturing a BMP or JPEG image to a file (Dest = fc_BmpFile or Dest = fc_JpegFile) , **pass an empty string in the FileName property.**

In this case a file name will be generated automatically in the [StoragePath](#) directory according to [AutoFileName](#) and [AutoFilePrefix](#) .

E.g.:

```
CaptureFrameTo (fc_JpegFile, '');
```

Otherwise you can specify yourself in the FileName property the path/file name to save the BMP or JPEG image.

See Also

[TAutoFileName](#) [TFrameCaptureDest](#) [TOnFrameCaptureCompleted](#) [BurstCount](#) [BurstInterval](#) [BurstMode](#) [BurstType](#) [CaptureFrameSyncTo](#) [FrameCaptureHeight](#) [FrameCaptureWidth](#) [FrameCaptureWithoutOverlay](#) [FrameCaptureZoomSize](#) [GetFrameInfoString](#) [GetLastFrameAsHBITMAP](#) [GetLastFrameAsTBitmap](#) [GetLastFrameBitmapBits](#) [GetLastFrameBitmapBits2](#) [GetLastFrameWaitTimeoutMs](#) [JPEGPerformance](#) [JPEGProgressiveDisplay](#) [JPEGQuality](#) [Last](#) [BurstFrameCapture](#) [FileName](#) [Last](#) [CaptureFrameTo](#) [FileName](#) [OnDiskFull](#) [OnFrameBitmap](#) [SetFrameCaptureBounds](#) [StoragePath](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly upgrade your WinHelp HLP help files to CHM with HelpNDoc](#)

ClearHeaderAttributes

TVideoGrabber.ClearHeaderAttributes

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Resets the AVI and ASF header attributes.

Declaration

procedure ClearHeaderAttributes;

void __fastcall ClearHeaderAttributes(**void**)

Public Overridable Sub ClearHeaderAttributes()

Description

Used to Reset all the AVI and ASF header attributes that have been set by invoking [SetHeaderAttribute](#).

See Also

[THeaderAttribute](#) [SetHeaderAttribute](#)

Created with the Standard Edition of HelpNDoc: [Save time and frustration with HelpNDoc's WinHelp HLP to CHM conversion feature](#)

ClosePlayer

TVideoGrabber.ClosePlayer

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

ber

Closes the current video clip.

Declaration

procedure ClosePlayer;

void __fastcall ClosePlayer(**void**)

Public Overridable Sub ClosePlayer()

Description

Closes the [PlayerFileName](#) video clip, previously opened with [OpenPlayer](#) .

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last Clip Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchron](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Full-featured EBook editor](#)

Create

TVideoGrabber.Create

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Constructor.

Declaration

constructor Create(Aowner: TComponent); **override**;

Description

Constructor.

Created with the Standard Edition of HelpNDoc: [Effortlessly Support Your Windows Applications with HelpNDoc's CHM Generation](#)

CreatePreallocCapFile

TVideoGrabber.CreatePreallocCapFile

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Creates an huge preallocated file.

Declaration

function CreatePreallocCapFile: Boolean;

bool __fastcall CreatePreallocCapFile(**void**)

Function CreatePreallocCapFile() As Boolean

Description

Used to create an huge preallocated file, according to [PreallocCapFileName](#) and [PreallocCapFileSizeInMB](#).

See Also

[TOnCreatePreallocatedFileCompleted](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#)
[OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileProgress](#)
[OnCreatePreallocFileStarted](#)

Created with the Standard Edition of HelpNDoc: [Experience the Power and Ease of Use of HelpNDoc for CHM Help File Generation](#)

Decrypt_File

TVideoGrabber.Decrypt_File

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Decrypts a video clip

Declaration

function Decrypt_File (InputFile: **string**; OutputFile: **string**; OverwriteIfExists: Boolean): Boolean;

bool __fastcall Decrypt_File(System::UnicodeString InputFile, System::UnicodeString OutputFile, **bool** OverwriteIfExists);

Description

Note: if the input file exists but is not encrypted the function fails

See Also

[TEncryptionMethod](#) [Encrypt_File](#) [EncryptionMethod](#) [SetDecryptionKey](#) [SetEncryptionKey](#)

Created with the Standard Edition of HelpNDoc: [Full-featured EPub generator](#)

Destroy

TVideoGrabber.Destroy

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Destructor.

Declaration

destructor Destroy; **override**;

Description

Destructor.

Created with the Standard Edition of HelpNDoc: [Effortlessly Create High-Quality Help Documentation with a Help Authoring Tool](#)

Display_SetLocation

TVideoGrabber.Display_SetLocation

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Used to set the location of the video window, when it is not embedded.

Declaration

procedure Display_SetLocation(WindowLeft, WindowTop, WindowWidth, WindowHeight: LongInt);

void __fastcall Display_SetLocation(**int** WindowLeft, **int** WindowTop, **int** WindowWidth, **int** WindowHeight)

Public Overridable Sub Display_SetLocation(ByVal WindowLeft As Integer, ByVal WindowTop As Integer, ByVal WindowWidth As Integer, ByVal WindowHeight As Integer)

Description

Use this method to set the video window location and size at a time (left, top, width, height), when the video window is not embedded in the TVideoGrabber component ([Display_Embedded](#) = false).

See Also

[TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display_Active](#) [Display_AlphaBlendEnabled](#) [Display_AlphaBlendValue](#) [Display_AutoSize](#) [Display_Embedded](#) [Display_FullScreen](#) [Display_Height](#) [Display_Left](#) [Display_Monitor](#) [Display_MouseMovesWindow](#) [Display_PanScanRatio](#) [Display_StayOnTop](#) [Display_Top](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [Display_VideoHeight](#) [Display_VideoPortEnabled](#) [Display_VideoWidth](#) [Display_VideoWindowHandle](#) [Display_Width](#) [DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#) [DualDisplay_AlphaBlendValue](#) [DualDisplay_AutoSize](#) [DualDisplay_Embedded](#) [DualDisplay_FullScreen](#) [DualDisplay_Height](#) [DualDisplay_Left](#) [DualDisplay_Monitor](#) [DualDisplay_MouseMovesWindow](#) [DualDisplay_PanScanRatio](#) [DualDisplay_StayOnTop](#) [DualDisplay_Top](#) [DualDisplay_TransparentColorEnabled](#) [DualDisplay_TransparentColorValue](#) [DualDisplay_VideoHeight](#) [DualDisplay_VideoPortEnabled](#) [DualDisplay_VideoWidth](#) [DualDisplay_VideoWindowHandle](#) [DualDisplay_Visible](#) [DualDisplay_Width](#) [IsVideoPortAvailable](#) [Monitor_Primary](#) [Index](#) [MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#) [SetParentWindow](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Generate Kindle eBooks with ease](#)

DrawBitmapOverFrame**TVideoGrabber.DrawBitmapOverFrame**
[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Draws the bitmap passed as parameter over video frames.

Declaration

function DrawBitmapOverFrame (Bitmap: Graphics.TBitmap; LeftLocation: LongInt; TopLocation: LongInt; bmWidth: LongInt; bmHeight: LongInt; Transparent: Boolean; UseTransparentColor: Boolean; TransparentColorValue: TColor; AlphaBlend: Boolean; AlphaBlendValue: LongInt): Boolean;

bool __fastcall DrawBitmapOverFrame(Graphics::TBitmap *Bitmap, **int** LeftLocation, **int** TopLocation, **int** bmWidth, **int** bmHeight, **bool** Transparent, **bool** UseTransparentColor, **TColor** TransparentColorValue, **bool** AlphaBlend, **int** AlphaBlendValue)

Function DrawBitmapOverFrame(BitmapHandle As Long, LeftLocation As Long, TopLocation As Long, bmWidth As Long, bmHeight As Long, Transparent As Boolean, UseTransparentColor As Boolean, TransparentColorValue As Long, AlphaBlend As Boolean, AlphaBlendValue As Long) As Boolean

Description

Used to draw bitmaps over video frames.

Location

The bitmap is drawn at the LeftLocation, TopLocation position over the video frame (expressed in pixels).

Size

If the **bmWidth** or **bmHeight** parameter is different from the bitmap's width or height, the bitmap is

stretched to **bmWidth** and **bmHeight**.

Transparency

- if the **Transparent** parameter is true, the image background is not drawn.
- if the **Transparent** parameter is true and the **UseTransparentColor** parameter is true, the **TransparentColorValue** value is used for transparency.

Alpha blending

If the **AlphaBlend** parameter is true, a progressive bitmap transparency is applied, depending of the **AlphaBlendValue**, in a 0..255 range.

It is not possible to perform alphablending when the **Transparent** parameter is true.

Important:

if the bitmap color format is not 32 bits, it will be converted into a 32 bit format. In other words, for better performance pass a bitmap already in 32 bits color format.

The *DrawBitmapOverFrame* function can be called ONLY from the [OnFrameOverlayUsingDC](#) event.
See the MainDemo project for sample code.

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#)
[AdjustOverlayAspectRatio](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#)
[MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#)
[OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#)
[SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#)
[SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#)
[SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#)
[SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#)
[SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#)
[SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#)
[SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#)
[SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#)
[SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#)
[SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#)
[SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#)
[ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#)
[TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#)
[TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#)
[TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#)
[TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#)
[TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Write eBooks for the Kindle](#)

DualDisplay_SetLocation

TVideoGrabber.DualDisplay_SetLocation

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Same as [Display_SetLocation](#) , but about the 2nd video window.

Declaration

procedure DualDisplay_SetLocation(WindowLeft, WindowTop, WindowWidth, WindowHeight: LongInt);

void __fastcall DualDisplay_SetLocation(int WindowLeft, int WindowTop, int WindowWidth, int WindowHeight)

Public Overridable Sub DualDisplay_SetLocation(ByVal WindowLeft As Integer, ByVal WindowTop As Integer, ByVal WindowWidth As Integer, ByVal WindowHeight As Integer)

Description

Same as [Display_SetLocation](#) , but about the 2nd video window.

See Also

[Dual display](#) [Display_AlphaBlendEnabled](#) [Display_AlphaBlendValue](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#) [DualDisplay_AlphaBlendValue](#) [DualDisplay_AutoSize](#) [DualDisplay_Embedded](#) [DualDisplay_FullScreen](#) [DualDisplay_Height](#) [DualDisplay_Left](#) [DualDisplay_Monitor](#) [DualDisplay_MouseMovesWindow](#) [DualDisplay_PanScanRatio](#) [DualDisplay_StayOnTop](#) [DualDisplay_Top](#) [DualDisplay_TransparentColorEnabled](#) [DualDisplay_TransparentColorValue](#) [DualDisplay_VideoHeight](#) [DualDisplay_VideoPortEnabled](#) [DualDisplay_VideoWidth](#) [DualDisplay_VideoWindowHandle](#) [DualDisplay_Visible](#) [DualDisplay_Width](#) [Monitor_Primary_Index](#) [MonitorBounds](#) [MonitorsCount](#) [SetWindowTransparency](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#)

Created with the Standard Edition of HelpNDoc: [Write eBooks for the Kindle](#)

DVDInfo**TVideoGrabber.DVDInfo**

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

DVD Info

Declaration

function DVDInfo(DVDRootDirectory: **string**; DVDInfoType: TDVDInfoType; TitleNumber: LongInt): Double;

double __fastcall DVDInfo(System::wchar_t *DVDRootDirectory, TDVDInfoType DVDInfoType, int TitleNumber)

Function DVDInfo (DVDRootDirectory as string, DVDInfoType as TDVDInfoType, TitleNumber as Long) as Double

Description

This function lets you retrieve information about the DVD.

DVDRootDirectory: path to the VIDEO_TS folder

DVDInfoType: selects the [TDVDInfoType](#) type of information requested. Possible values:

dvd_NumberOfVolumes
dvd_TotalDuration
dvd_NumberOfTitles
dvd_TitleDuration
dvd_TitleFrameRate
dvd_SourceResolutionX
dvd_SourceResolutionY
dvd_TitleFrameCount

TitleNumber: number of the title requested (in the 1..n range) or 0 for the global information (dvd_TotalDuration and dvd_NumberOfTitles)

(note that the DVD data is cached, therefore the DVD is read only one time, other calls to DVDInfo just return the cached values, unless you specify a different DVD folder)

(all the values are returned as double)

E.g. to get:

- *the total duration of the DVD:*

```
double TotalDuration = Videograbber.DvdInfo ("e:dvd_TotalDuration, 0)
```

- *the number of titles in the DVD:*

```
int NumberOfTitles = (int) Videograbber.DvdInfo ("e:dvd_NumberOfTitles, 0)
```

- *the duration of the 1st title:*

```
double TitleDuration = Videograbber.DvdInfo ("e:dvd_TitleDuration, 1)
```

- *the number of frames of the 1st title:*

```
int TitleFrameCount = (int) Videograbber.DvdInfo ("e:dvd_TitleFrameCount, 1)
```

- *the frame rate of the 1st title:*

```
int TitleFrameRate = (int) Videograbber.DvdInfo ("e:dvd_TitleFrameRate, 1)
```

- *the width of the video resolution of the 1st title:*

```
int TitleVideoWidth = (int) Videograbber.DvdInfo ("e:dvd_SourceResolutionX, 1)
```

- *the height of the video resolution of the 1st title:*

```
int TitleVideoHeight = (int) Videograbber.DvdInfo ("e:dvd_SourceResolutionY, 1)
```

See Also

[DVDTitle](#) [OpenDVD](#) [PlayerFileName](#)

Created with the Standard Edition of HelpNDoc: [Step-by-Step Guide: How to Turn Your Word Document into an eBook](#)

EnableMultiplexedInput

TVideoGrabber.EnableMultiplexedInput

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Enables / disables a multiplexed input.

Declaration

```
function EnableMultiplexedInput(InputNumber: LongInt; Enable: Boolean): Boolean;
```

```
bool __fastcall EnableMultiplexedInput(int InputNumber, bool Enable);
```

Function EnableMultiplexedInput(InputNumber as Long, Enable as Boolean) as Boolean

Description

Used to enable / disable a multiplexed input.

See Also

[TMultiplexedRole](#) [AssociateMultiplexedSlave](#) [MultiplexedInputEmulation](#) [MultiplexedRole](#)
[MultiplexedStabilizationDelay](#) [MultiplexedSwitchDelay](#) [UniqueID](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Convert Your Markdown Content with HelpNDoc](#)

EnableThreadMode

TVideoGrabber.EnableThreadMode

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Enables the thread mode

Declaration

procedure EnableThreadMode;

void EnableThreadMode()

Sub EnableThreadMode()

Description

To enable the thread mode, invoke EnableThreadMode just after creating the component.

See Also

[Opening a clip or an IP URL from a background thread without blocking the main thread](#) [Player features](#) [TOnThreadSync](#) [TThreadSyncPoint](#) [OnThreadSync](#)

Created with the Standard Edition of HelpNDoc: [What is a Help Authoring tool?](#)

Encoder_CloseOutputFile

TVideoGrabber.Encoder_CloseOutputFile

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Encoder_NewOutputFile

Declaration

function Encoder_CloseOutputFile (EncoderUniqueID: LongInt): Boolean;

int Encoder_CloseOutputFile(**int** EncoderUniqueID);

Description

Close the file curently being written

Created with the Standard Edition of HelpNDoc: [Say Goodbye to Documentation Headaches with a Help Authoring Tool](#)

Encoder_GetInt

TVideoGrabber.Encoder_GetInt

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Encoder_GetInt

Declaration

function Encoder_GetInt (EncoderID: LongInt; Setting: TEncoder_int; **var** Value: LongInt): Boolean;

bool Encoder_GetInt(**int** EncoderID, TEncoder_int Setting, **int** *Value);

Description

Retrives an integer value from the [Datastead Encoder](#)

Useful mainly to get the total Kb written by invoking Encoder_GetInt (ID, Enc_Bytes_Written_kb_readonly, Value)

See Also

[Datastead Encoder](#) [Pause/resume during recording](#) [Recording methods and properties](#) [Encoder_Pause](#) [Encoder_Resume](#) [Encoder_SetInt](#) [Encoder_SetStr](#) [Encoders_CreateInstanceForRecording](#) [Encoders_CreateInstanceForStreaming](#) [Encoders_RemoveAllInstances](#) [Encoders_RemoveInstance](#) [SetDatasteadFilterDllName](#)

Created with the Standard Edition of HelpNDoc: [Streamline your documentation process with HelpNDoc's HTML5 template](#)

Encoder_NewOutputFile**TVideoGrabber.Encoder_NewOutputFile**[Prev](#)[Next](#)

[TVideoGrabber](#) [Methods](#)

Encoder_NewOutputFile

Declaration

function Encoder_NewOutputFile (EncoderUniqueID: LongInt; OutputFile: **string**; OpenPaused: Boolean): Boolean;

int Encoder_NewOutputFile(**int** EncoderUniqueID, wchar_t *OutputFile, **bool** OpenPaused);

Description

Close the file curenly being written, and starts recording to the new file name specified. If OpenPaused is true, the file will not be written until [Encoder_Resume](#) is invoked.

See Also

[Datastead Encoder](#) [Pause/resume during recording](#) [Recording methods and properties](#) [Encoder_GetInt](#) [Encoder_Pause](#) [Encoder_Resume](#) [Encoder_SetInt](#) [Encoder_SetStr](#) [Encoders_CreateInstanceForRecording](#) [Encoders_CreateInstanceForStreaming](#) [Encoders_RemoveAllInstances](#) [Encoders_RemoveInstance](#) [SetDatasteadFilterDllName](#)

Created with the Standard Edition of HelpNDoc: [Streamline Your CHM Help File Creation with HelpNDoc](#)

Encoder_Pause**TVideoGrabber.Encoder_Pause**[Prev](#)[Next](#)

[TVideoGrabber](#) [Methods](#)

Encoder_Pause

Declaration

function Encoder_Pause (EncoderUniqueID: LongInt): Boolean;

bool Encoder_Pause(**int** EncoderUniqueID);

Description

Pauses the encoder:
 - if recording to a file, stops writing to the file
 - if streaming, stops streaming
 The video preview does not stop.

See Also

[Datastead Encoder](#) [Pause/resume during recording](#) [Recording methods and properties](#) [Encoder_GetInt](#) [Encoder_Resume](#) [Encoder_SetInt](#) [Encoder_SetStr](#) [Encoders_CreateInstanceForRecording](#) [Encoders_CreateInstanceForStreaming](#) [Encoders_RemoveAllInstances](#) [Encoders_RemoveInstance](#) [SetDatasteadFilterDllName](#)

Created with the Standard Edition of HelpNDoc: [Produce Kindle eBooks easily](#)

Encoder_Resume

TVideoGrabber.Encoder_Resume

[Prev](#)[Next](#)[TVideoGrabber](#) [Methods](#)

Encoder_Resume

Declaration

function Encoder_Resume (EncoderUniqueID: LongInt): Boolean;**bool** Encoder_Resume(int EncoderUniqueID);

Description

Resumes the encoder:

- if recording to a file, restart writing to the file
- if streaming, resume streaming

The video preview goes on without stop/restart

See Also

[Datastead Encoder Pause/resume during recording](#) [Recording methods and properties](#) [Encoder_GetInt](#) [Encoder_Pause](#) [Encoder_SetInt](#) [Encoder_SetStr](#) [Encoders_CreateInstanceForRecording](#) [Encoders_CreateInstanceForStreaming](#) [Encoders_RemoveAllInstances](#) [Encoders_RemoveInstance](#) [SetDatasteadFilterDllName](#)

Created with the Standard Edition of HelpNDoc: [Converting Word Docs to eBooks Made Easy with HelpNDoc](#)

Encoder_SetInt

TVideoGrabber.Encoder_SetInt

[Prev](#)[Next](#)[TVideoGrabber](#) [Methods](#)

Encoder_SetInt

Declaration

function Encoder_SetInt (EncoderID: LongInt; Setting: TEncoder_int; Value: LongInt): Boolean;**bool** Encoder_SetInt(int EncoderID, TEncoder_int Setting, int Value);

Description

Sets an integer parameter of the [Datastead Encoder](#)

See Also

[Datastead Encoder Pause/resume during recording](#) [Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder_GetInt](#) [Encoder_Pause](#) [Encoder_Resume](#) [Encoder_SetStr](#) [Encoders_CreateInstanceForRecording](#) [Encoders_CreateInstanceForStreaming](#) [Encoders_RemoveAllInstances](#) [Encoders_RemoveInstance](#) [HoldRecording](#) [IsRecordingPaused](#) [Last_Recording_FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetDatasteadFilterDllName](#) [SetMultiplexerFilterByName](#)

[StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#)
[SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [Upgrade Your Documentation Process with a Help Authoring Tool](#)

Encoder_SetStr

TVideoGrabber.Encoder_SetStr

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Encoder_SetStr

Declaration

function Encoder_SetStr (EncoderID: LongInt; Setting: TEncoder_str; Value: **string**): Boolean;

bool Encoder_SetStr(int EncoderID, TEncoder_str Setting, wchar_t *Value);

Description

Sets a string parameter of the [Datastead Encoder](#)

See Also

[Datastead Encoder](#) [Pause/resume during recording](#) [Recording methods and properties](#) [Encoder_GetInt](#)
[Encoder_Pause](#) [Encoder_Resume](#) [Encoder_SetInt](#) [Encoders_CreateInstanceForRecording](#)
[Encoders_CreateInstanceForStreaming](#) [Encoders_RemoveAllInstances](#) [Encoders_RemoveInstance](#)
[SetDatasteadFilterDllName](#)

Created with the Standard Edition of HelpNDoc: [Free help authoring environment](#)

Encoders_CreateInstanceForRecording

TVideoGrabber.Encoders_CreateInstanceForRecording

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Encoders_CreateInstanceForRecording

Declaration

function Encoders_CreateInstanceForRecording (OutputFile: **string**): LongInt;

int Encoders_CreateInstanceForRecording(wchar_t * OutputFile);

Description

Adds a [Datastead Encoder](#) instance to the graph, to write to a new file.

Returns an unique ID to this encoder.

This unique ID will be required to invoke [Encoder_SetInt](#), [Encoder_GetInt](#), [Encoder_SetStr](#),
[Encoder_Pause](#), [Encoder_Resume](#), [Encoder_RecordToNewFileNow](#), or [Encoders_RemoveInstance](#).

See Also

[Datastead Encoder](#) [Pause/resume during recording](#) [Recording methods and properties](#) [Encoder_GetInt](#)
[Encoder_Pause](#) [Encoder_Resume](#) [Encoder_SetInt](#) [Encoder_SetStr](#)
[Encoders_CreateInstanceForStreaming](#) [Encoders_RemoveAllInstances](#) [Encoders_RemoveInstance](#)
[SetDatasteadFilterDllName](#)

Created with the Standard Edition of HelpNDoc: [Full-featured EBook editor](#)

Encoders_CreateInstanceForStreaming

TVideoGrabber.Encoders_CreateInstanceForStreaming

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

[ber](#)

Encoders_CreateInstanceForStreaming

Declaration

function Encoders_CreateInstanceForStreaming (OutputURL: **string**): LongInt;

int Encoders_CreateInstanceForStreaming(wchar_t *OutputURL);

Description

Adds a [Datastead Encoder](#) instance to the graph, to stream to a new destination.

Returns an unique ID to this encoder.

This unique ID will be required to invoke [Encoder_SetInt](#), [Encoder_GetInt](#), [Encoder_SetStr](#), [Encoder_Pause](#), [Encoder_Resume](#), [Encoder_RecordToNewFileNow](#), or [Encoders_RemoveInstance](#).

See Also

[Datastead Encoder Pause/resume during recording](#) [Recording methods and properties](#) [Encoder_GetInt](#) [Encoder_Pause](#) [Encoder_Resume](#) [Encoder_SetInt](#) [Encoder_SetStr](#) [Encoders_CreateInstanceForRecording](#) [Encoders_RemoveAllInstances](#) [Encoders_RemoveInstance](#) [SetDatasteadFilterDllName](#)

Created with the Standard Edition of HelpNDoc: [Easily create Qt Help files](#)

Encoders_RemoveAllInstances

TVideoGrabber.Encoders_RemoveAllInstances

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Encoders_RemoveAllInstances

Declaration

procedure Encoders_RemoveAllInstances();

void Encoders_RemoveAllInstances(**void**);

Description

Remove all the [Datastead Encoder](#) instances previously added by [Encoders_CreateInstanceForRecording](#) or [Encoders_CreateInstanceForStreaming](#).

See Also

[Datastead Encoder Pause/resume during recording](#) [Recording methods and properties](#) [Encoder_GetInt](#) [Encoder_Pause](#) [Encoder_Resume](#) [Encoder_SetInt](#) [Encoder_SetStr](#) [Encoders_CreateInstanceForRecording](#) [Encoders_CreateInstanceForStreaming](#) [Encoders_RemoveInstance](#) [SetDatasteadFilterDllName](#)

Created with the Standard Edition of HelpNDoc: [Don't be left in the past: convert your WinHelp HLP help files to CHM with HelpNDoc](#)

Encoders_RemoveInstance

TVideoGrabber.Encoders_RemoveInstance

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Encoders_RemoveInstance

Declaration

function Encoders_RemoveInstance (EncoderUniqueID: LongInt): Boolean;

bool Encoders_RemoveInstance(**int** EncoderUniqueID);

Description

Remove all the [Datastead Encoder](#) instances previously added by [Encoders_CreateInstanceForRecording](#) or [Encoders_CreateInstanceForStreaming](#).

See Also

[Datastead Encoder Pause/resume during recording](#) [Recording methods and properties](#) [Encoder_GetInt](#) [Encoder_Pause](#) [Encoder_Resume](#) [Encoder_SetInt](#) [Encoder_SetStr](#) [Encoders_CreateInstanceForRecording](#) [Encoders_CreateInstanceForStreaming](#) [Encoders_RemoveAllInstances](#) [SetDatasteadFilterDllName](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your CHM Help File Capabilities with HelpNDoc](#)

Encrypt_File

TVideoGrabber.Encrypt_File

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Encrypts a video clip

Declaration

function Encrypt_File (InputFile: **string**; OutputFile: **string**; OverwriteIfExists: Boolean): Boolean;

bool __fastcall Encrypt_File(System::UnicodeString InputFile, System::UnicodeString OutputFile, **bool** OverwriteIfExists);

Description

Note: if the input file exists but is already encrypted the function fails

See Also

[TEncryptionMethod](#) [Decrypt_File](#) [EncryptionMethod](#) [SetDecryptionKey](#) [SetEncryptionKey](#)

Created with the Standard Edition of HelpNDoc: [Don't be left in the past: convert your WinHelp HLP help files to CHM with HelpNDoc](#)

EnumerateWindows

TVideoGrabber.EnumerateWindows

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Used in screen recording to retrieve information about visible windows.

Declaration

function EnumerateWindows: Boolean;

bool __fastcall EnumerateWindows(**void**);

Function EnumerateWindows as Boolean

Description

Helper function that can be used for the screen recording of a window.

Immediately after Invoking EnumerateWindows, information about all the visible windows will be returned by the [OnEnumerateWindows](#) event, that will occur for each visible window.

See the "[Recording a window](#)" chapter for more information.

See Also

[TOnEnumerateWindows](#) [ScreenRecordingLayeredWindows](#) [ScreenRecordingMonitor](#) [ScreenRecordingNonVisibleWindows](#) [ScreenRecordingUsingCoordinates](#) [ScreenRecordingWithCursor](#)

[SetWindowRecordingByHandle](#) [SetWindowRecordingByName](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Support Your Windows Applications with HelpNDoc's CHM Generation](#)

Facebook_GoLive_ReturnStreamURL

TVideoGrabber.Facebook_GoLive_ReturnStreamURL

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns the streaming URL of a Facebook live session

Declaration

function Facebook_GoLive_ReturnStreamURL (EndPointURL: **string**; Title: **string**; Description: **string**; Status: **string**; AccessToken: **string**; caBundlePath: **string**): **string**;

System::UnicodeString __fastcall Facebook_GoLive_ReturnStreamURL(System::UnicodeString EndPointURL, System::UnicodeString Title, System::UnicodeString Description, System::UnicodeString Status, System::UnicodeString AccessToken, System::UnicodeString caBundlePath);

Description

Used to obtain the RTMP URL of a Facebook endpoint, in order to pass it to the Multipurpose Encoder for Facebook live streaming.

C# sample code:

```
string strEndPointURL, strAccessToken, strRTMPURL;
strEndPointURL =
"https://graph.facebook.com/v23.0/64572161632085/live_videos";
strAccessToken =
"EAAALucaCaoyABPAYe9rl4agr5hZB6YjwVxUS9bVg2hCcR089kiSt0lA10smuVFxZAZAyspzW7Uh6M
lWZCocsv7pYy2AE4tZCnLK4iVcAcYkjOHHWNUD5omifalyMeL8ZBA0qm53TlsnDmGjVTtlpiZBMyx3
ggHZBKkuuZCDs9TNN9IlUp8z5nWAoNFsZCJq8pH8wvBGUDfFOV4y18ZBS4C5CZC7lUrhlaogWB8XS1
xAtZC3WDJ2t4efsuurGeZCRQo3tLYZD";
// Assuming you have a method similar to the helphi function:
strRTMPURL = VideoGrabber.Facebook_GoLive_ReturnStreamURL(
strEndPointURL,
"fb streaming test",
"my description",
"LIVE_NOW",
strAccessToken,
""
);
```

Delphi sample code:

```
procedure TestFacebookGoLive;
var strEndPointURL, strAccessToken, strRTMPURL: string;
begin
strEndPointURL :=
'https://graph.facebook.com/v23.0/64572161632085/live_videos';
strAccessToken :=
'EAAALucaCaoyABPAYe9rl4agr5hZB6YjwVxUS9bVg2hCcR089kiSt0lA10smuVFxZAZAyspzW7Uh6M
lWZCocsv7pYy2AE4tZCnLK4iVcAcYkjOHHWNUD5omifalyMeL8ZBA0qm53TlsnDmGjVTtlpiZBMyx3
ggHZBKkuuZCDs9TNN9IlUp8z5nWAoNFsZCJq8pH8wvBGUDfFOV4y18ZBS4C5CZC7lUrhlaogWB8XS1
xAtZC3WDJ2t4efsuurGeZCRQo3tLYZD';
strRTMPURL := VideoGrabber.Facebook_GoLive_ReturnStreamURL(strEndPointURL, 'fb
streaming test', 'my description', 'LIVE_NOW', strAccessToken, '');
if copy (strRTMPURL, 1, 5) <> 'ERROR' then begin
showmessage(strRTMPURL);
end;
end;
```

end ;

Created with the Standard Edition of HelpNDoc: [Full-featured EBook editor](#)

FastForwardPlayer

TVideoGrabber.FastForwardPlayer

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Plays a video clip forwards at a different speed.

Declaration

procedure FastForwardPlayer;

void __fastcall FastForwardPlayer(**void**)

Public Overridable Sub FastForwardPlayer()

Description

Used to play the clip forwards faster or slower than the normal speed.

The fast forward speed is the normal speed x [PlayerFastSeekRatio](#) .

Restriction: this feature is available only with seekable (indexed) AVI clips.

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchron](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Easily create EBooks](#)

FindIndexInListByName

TVideoGrabber.FindIndexInListByName

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Search the index of an item in a list of strings, by name or substring

Declaration

function FindIndexInListByName(List: **String**; SearchedString: **String**; IsSubString: boolean; IgnoreCase: Boolean): LongInt;

int __fastcall FindIndexInListByName(**UnicodeString** List, **UnicodeString** SearchedString, **bool** IsSubString, **bool** IgnoreCase)

Function FindIndexInListByName(List As String, SearchedString As String, IsSubString As Boolean, IgnoreCase As Boolean) As Long

Description

Used to find the index of an item in a list of strings, searching by the name of the item or a substring that

identifies the item.

E.g., for a given video capture device, the [VideoInputs](#) list returns:

Composite
SVideo
Tuner

Therefore the index of "Composite" is 0, the index of "SVideo" is 1, and the index of "Tuner" is 2.

E.g. to programmatically select the "Composite" input. Proceed as follows:

```
procedure TfrmMainForm.Button1Click(Sender: TObject);
var
    i: LongInt;
begin
    i := VideoGrabber.FindIndexInListByName (VideoGrabber.VideoInputs, 'Composite', false, true);
    if i > -1 then begin // if this input exists...
        VideoGrabber.VideoInput := i;
    end;
end;
```

Or to select the same input by using only "Compos" as substring of the input name:

```
procedure TfrmMainForm.Button1Click(Sender: TObject);
var
    i: LongInt;
begin
    i := VideoGrabber.FindIndexInListByName (VideoGrabber.VideoInputs, 'Compos', true, true);
    if i > -1 then begin // if this input exists...
        VideoGrabber.VideoInput := i;
    end;
end;
```

See Also

[OnVideoDeviceSelected](#) [StoreDeviceSettingsInRegistry](#)

Created with the Standard Edition of HelpNDoc: [Make Documentation Review a Breeze with HelpNDoc's Advanced Project Analyzer](#)

GetAudioCodec

TVideoGrabber.GetAudioCodec

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Current audio codec

Declaration

function GetAudioCodec: **string**;

System::wchar_t * __fastcall GetAudioCodec()

Function GetAudioCodec as String

Description

Returns the current audio codec beeing used.

GetCameraExposure

TVideoGrabber.GetCameraExposure

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Used to set the camera exposure as a double value, e.g. 1/8 s, 1/16 s, 1/32 s, etc...
Retrieves the camera exposure as double value

Declaration

function GetCameraExposure: Double;

double __fastcall GetCameraExposure(**void**);

Function GetCameraExposure() as Double

Description

Used to retrieve the camera exposure as a double value, e.g. 1/8 s, 1/16 s, 1/32 s, etc...

See Also

[TCameraControl](#) [CameraControlAuto](#) [CameraControlDefault](#) [CameraControlMax](#) [CameraControlMin](#) [CameraControlSettings](#) [CameraControlStep](#) [GetCameraExposureAsString](#) [IsCameraControlAvailable](#) [IsCameraControlSettingAvailable](#) [SendCameraCommand](#) [SetCameraControl](#) [SetCameraExposure](#)

GetCameraExposureAsString

TVideoGrabber.GetCameraExposureAsString

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Retrieves the camera exposure value as a string

Declaration

function GetCameraExposureAsString: **string**;

wchar_t * __fastcall GetCameraExposureAsString();

Function GetCameraExposureAsString as String

Description

Used to retrieve the camera exposure as a fractional string, e.g. "1/8" s, "1/16" s, "1/32" s, etc...

See Also

[TCameraControl](#) [CameraControlAuto](#) [CameraControlDefault](#) [CameraControlMax](#) [CameraControlMin](#) [CameraControlSettings](#) [CameraControlStep](#) [GetCameraExposure](#) [IsCameraControlAvailable](#) [IsCameraControlSettingAvailable](#) [SendCameraCommand](#) [SetCameraControl](#) [SetCameraExposure](#)

GetDisplayActive

TVideoGrabber.GetDisplayActive

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

see [SetDisplay_Active](#)

Declaration

function GetDisplayActive (DisplayIndex: LongInt): Boolean;

bool __fastcall GetDisplayActive(**int** DisplayIndex);

function GetDisplayActive (DisplayIndex as Long) as **bool**

Description

see [SetDisplay_Active](#)

Created with the Standard Edition of HelpNDoc: [Converting Word Docs to eBooks Made Easy with HelpNDoc](#)

GetDisplayAlphaBlendEnabled

TVideoGrabber.GetDisplayAlphaBlendEnabled

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

see [SetDisplayAlphaBlendEnabled](#)

Declaration

function GetDisplayAlphaBlendEnabled (DisplayIndex: LongInt): Boolean;

bool __fastcall GetDisplayAlphaBlendEnabled(**int** DisplayIndex);

function GetDisplayAlphaBlendEnabled (DisplayIndex as Long) as **bool**

Description

see [SetDisplayAlphaBlendEnabled](#)

Created with the Standard Edition of HelpNDoc: [Benefits of a Help Authoring Tool](#)

GetDisplayAlphaBlendValue

TVideoGrabber.GetDisplayAlphaBlendValue

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

see [SetDisplayAlphaBlendValue](#)

Declaration

function GetDisplayAlphaBlendValue (DisplayIndex: LongInt): LongInt;

int __fastcall GetDisplayAlphaBlendValue(**int** DisplayIndex);

function GetDisplayAlphaBlendValue (DisplayIndex as Long) as **Long**

Description

see [SetDisplayAlphaBlendValue](#)

Created with the Standard Edition of HelpNDoc: [Bring your WinHelp HLP help files into the present with HelpNDoc's easy CHM conversion](#)

GetDisplayAspectRatio

TVideoGrabber.GetDisplayAspectRatio

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

see [SetDisplayAspectRatio](#)

Declaration

function GetDisplayAspectRatio (DisplayIndex: LongInt): TAspectRatio;

TAspectRatio **__fastcall** GetDisplayAspectRatio(**int** DisplayIndex);

function GetDisplayAspectRatio (DisplayIndex as Long) as TAspectRatio

Description

see [SetDisplayAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Free Web Help generator](#)

GetDisplayAutoSize

TVideoGrabber.GetDisplayAutoSize

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

see [SetDisplayAutoSize](#)

Declaration

function GetDisplayAutoSize (DisplayIndex: LongInt): Boolean;

bool **__fastcall** GetDisplayAutoSize(**int** DisplayIndex);

function GetDisplayAutoSize (DisplayIndex as Long) as **bool**

Description

see [SetDisplayAutoSize](#)

Created with the Standard Edition of HelpNDoc: [Free Web Help generator](#)

GetDisplayEmbedded

TVideoGrabber.GetDisplayEmbedded

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

see [SetDisplayEmbedded](#)

Declaration

function GetDisplayEmbedded (DisplayIndex: LongInt): Boolean;

bool **__fastcall** GetDisplayEmbedded(**int** DisplayIndex);

function GetDisplayEmbedded (DisplayIndex as Long) as **bool**

Description

see [SetDisplayEmbedded](#)

Created with the Standard Edition of HelpNDoc: [Free EPub producer](#)

GetDisplayFullScreen

TVideoGrabber.GetDisplayFullScreen

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

see [SetDisplayFullScreen](#)

Declaration

function GetDisplayFullScreen (DisplayIndex: LongInt): Boolean;

bool __fastcall GetDisplayFullScreen(int DisplayIndex);

function GetDisplayFullScreen (DisplayIndex as Long) as **bool**

Description

see [SetDisplayFullScreen](#)

Created with the Standard Edition of HelpNDoc: [Free help authoring environment](#)

GetDisplayHeight

[TVideoGrabber](#).GetDisplayHeight

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

see [SetDisplayHeight](#)

Declaration

function GetDisplayHeight (DisplayIndex: LongInt): LongInt;

int __fastcall GetDisplayHeight(int DisplayIndex);

function GetDisplayHeight (DisplayIndex as Long) as Long

Description

see [SetDisplayHeight](#)

Created with the Standard Edition of HelpNDoc: [Converting Word Documents to eBooks: A Step-by-Step Guide with HelpNDoc](#)

GetDisplayLeft

[TVideoGrabber](#).GetDisplayLeft

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

see [SetDisplayLeft](#)

Declaration

function GetDisplayLeft (DisplayIndex: LongInt): LongInt;

int __fastcall GetDisplayLeft(int DisplayIndex);

function GetDisplayLeft (DisplayIndex as Long) as Long

Description

see [SetDisplayLeft](#)

Created with the Standard Edition of HelpNDoc: [Ensure High-Quality Documentation with HelpNDoc's Hyperlink and Library Item Reports](#)

GetDisplayMonitor

TVideoGrabber.GetDisplayMonitor

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

see [SetDisplayMonitor](#)

Declaration

function GetDisplayMonitor (DisplayIndex: LongInt): LongInt;

int __fastcall GetDisplayMonitor(**int** DisplayIndex);

function GetDisplayMonitor (DisplayIndex as Long) as Long

Description

see [SetDisplayMonitor](#)

Created with the Standard Edition of HelpNDoc: [Produce Kindle eBooks easily](#)

GetDisplayMouseMovesWindow

TVideoGrabber.GetDisplayMouseMovesWindow

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

see [SetDisplayMouseMovesWindow](#)

Declaration

function GetDisplayMouseMovesWindow (DisplayIndex: LongInt): Boolean;

bool __fastcall GetDisplayMouseMovesWindow(**int** DisplayIndex);

function GetDisplayMouseMovesWindow (DisplayIndex as Long) as **bool**

Description

see [SetDisplayMouseMovesWindow](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Productivity with a Help Authoring Tool](#)

GetDisplayPanScanRatio

TVideoGrabber.GetDisplayPanScanRatio

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

see [SetDisplayPanScanRatio](#)

Declaration

function GetDisplayPanScanRatio (DisplayIndex: LongInt): LongInt;

int __fastcall GetDisplayPanScanRatio(**int** DisplayIndex);

function GetDisplayPanScanRatio (DisplayIndex as Long) as Long

Description

see [SetDisplayPanScanRatio](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Support Your Windows Applications with](#)

GetDisplayStayOnTop

TVideoGrabber.GetDisplayStayOnTop

[Prev](#)[Next](#)[TVideoGrabber](#) [Methods](#)

see [SetDisplayStayOnTop](#)

Declaration

function GetDisplayStayOnTop (DisplayIndex: LongInt): Boolean;

bool __fastcall GetDisplayStayOnTop(**int** DisplayIndex);

function GetDisplayStayOnTop (DisplayIndex as Long) as **bool**

Description

see [SetDisplayStayOnTop](#)

Created with the Standard Edition of HelpNDoc: [Create HTML Help, DOC, PDF and print manuals from 1 single source](#)

GetDisplayTop

TVideoGrabber.GetDisplayTop

[Prev](#)[Next](#)[TVideoGrabber](#) [Methods](#)

see [SetDisplayTop](#)

Declaration

function GetDisplayTop (DisplayIndex: LongInt): LongInt;

int __fastcall GetDisplayTop(**int** DisplayIndex);

function GetDisplayTop (DisplayIndex as Long) as Long

Description

see [SetDisplayTop](#)

Created with the Standard Edition of HelpNDoc: [5 Reasons Why a Help Authoring Tool is Better than Microsoft Word for Documentation](#)

GetDisplayTransparentColorEnabled

TVideoGrabber.GetDisplayTransparentColorEnabled

[Prev](#)[Next](#)[TVideoGrabber](#) [Methods](#)

see [SetDisplayTransparentColorEnabled](#)

Declaration

function GetDisplayTransparentColorEnabled (DisplayIndex: LongInt): Boolean;

bool __fastcall GetDisplayTransparentColorEnabled(**int** DisplayIndex);

function GetDisplayTransparentColorEnabled (DisplayIndex as Long) as **bool**

Description

see [SetDisplayTransparentColorEnabled](#)

Created with the Standard Edition of HelpNDoc: [Transform Your Help Documentation Process with a Help Authoring Tool](#)

GetDisplayTransparentColorValue

TVideoGrabber.GetDisplayTransparentColorValue

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

see [SetDisplayTransparentColorValue](#)

Declaration

function GetDisplayTransparentColorValue (DisplayIndex: LongInt): LongInt;

int __fastcall GetDisplayTransparentColorValue(**int** DisplayIndex);

function GetDisplayTransparentColorValue (DisplayIndex as Long) as Long

Description

see [SetDisplayTransparentColorValue](#)

Created with the Standard Edition of HelpNDoc: [Create Professional CHM Help Files with HelpNDoc's Easy-to-Use Tool](#)

GetDisplayVideoHeight

TVideoGrabber.GetDisplayVideoHeight

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

current video height

Declaration

function GetDisplayVideoHeight (DisplayIndex: LongInt): LongInt;

int __fastcall GetDisplayVideoHeight(**int** DisplayIndex);

function GetDisplayVideoHeight (DisplayIndex as Long) as Long

Description

Returns the current video height of the video window

Created with the Standard Edition of HelpNDoc: [Save time and frustration with HelpNDoc's WinHelp HLP to CHM conversion feature](#)

GetDisplayVideoPortEnabled

TVideoGrabber.GetDisplayVideoPortEnabled

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

see [SetDisplayVideoPortEnabled](#)

Declaration

function GetDisplayVideoPortEnabled (DisplayIndex: LongInt): Boolean;

bool __fastcall GetDisplayVideoPortEnabled(**int** DisplayIndex);

function GetDisplayVideoPortEnabled ayActive (DisplayIndex as Long) as **bool**

Description

see [SetDisplayVideoPortEnabled](#)

Created with the Standard Edition of HelpNDoc: [Free EBook and documentation generator](#)

GetDisplayVideoWidth

TVideoGrabber.GetDisplayVideoWidth

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

current video width

Declaration

function GetDisplayVideoWidth (DisplayIndex: LongInt): LongInt;

int __fastcall GetDisplayVideoWidth(**int** DisplayIndex);

function GetDisplayVideoWidth (DisplayIndex as Long) as Long

Description

Returns the current video width of the video window

Created with the Standard Edition of HelpNDoc: [Easily create PDF Help documents](#)

GetDisplayVideoWindowHandle

TVideoGrabber.GetDisplayVideoWindowHandle

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

handle of the current video window

Declaration

function GetDisplayVideoWindowHandle (DisplayIndex: LongInt): HWND;

void * __fastcall GetDisplayVideoWindowHandle(**int** DisplayIndex);

function GetDisplayVideoWindowHandle (DisplayIndex as Long) as Long

Description

retuns the handle of the current video window

Created with the Standard Edition of HelpNDoc: [Easily create Web Help sites](#)

GetDisplayVisible

TVideoGrabber.GetDisplayVisible

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

see [SetDisplayVisible](#)

Declaration

function GetDisplayVisible (DisplayIndex: LongInt): Boolean;

bool __fastcall GetDisplayVisible(int DisplayIndex);

function GetDisplayVisible (DisplayIndex as Long) as **bool**

Description

see [SetDisplayVisible](#)

Created with the Standard Edition of HelpNDoc: [Full-featured Help generator](#)

GetDisplayWidth

TVideoGrabber.GetDisplayWidth

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

see [SetDisplayWidth](#)

Declaration

function GetDisplayWidth (DisplayIndex: LongInt): LongInt;

int __fastcall GetDisplayWidth(int DisplayIndex);

function GetDisplayWidth (DisplayIndex as Long) as Long

Description

see [SetDisplayWidth](#)

Created with the Standard Edition of HelpNDoc: [Elevate Your CHM Help Files with HelpNDoc's Advanced Customization Options](#)

GetFrameInfo

TVideoGrabber.GetFrameInfo

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Used to retrieve information about the current video frame.

Declaration

function GetFrameInfo(FrameId: LongInt; FrameInfoId: TFrameInfoId): LongWord;

unsigned __fastcall GetFrameInfo(int FrameId, TFrameInfoId FrameInfoId)

Function GetFrameInfo(FrameId As Long, FrameInfoId As TFrameInfoId) As Long

Description

Used to retrieve information about the current video frame.

Must be invoked from the TVideoGrabber events that occurs for video frames, like [OnFrameProgress](#), [OnFrameCaptureCompleted](#), [OnMotionDetected](#), etc...

- pass as first parameter the FrameId parameter returned by the event,
- pass as second parameter the [TFrameInfoId](#) value corresponding to the desired information.

E.g.:

```
procedure TfrmMainForm.VideoGrabberFrameProgress(Sender: TObject; FrameNumber: Cardinal);
begin
    edtFrameCount.Text := 'frame: ' + IntToStr (FrameNumber)
                        + ' dropped:' + IntToStr (VideoGrabber.GetFrameInfo (FrameId, fi_DroppedFrame));
```

```

+ ' time:' + Format ('%.2d:%.2d:%.2d %.2d', [VideoGrabber.GetFrameInfo (Fr
VideoGrabber.GetFrameInfo (Fr
VideoGrabber.GetFrameInfo (Fr
VideoGrabber.GetFrameInfo (Fr
end;

```

See Also

[TFrameGrabberRGBFormat](#) [FrameGrabber](#) [FrameGrabberCurrentRGBFormat](#) [FrameGrabberRGBFormat](#)
[FramerateDivider](#) [GetFrameInfoString](#) [InFrameProgressEvent](#) [OnFrameBitmap](#)
[OnFrameBitmapEventSynchron](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Productivity with HelpNDoc's CHM Help File Creation Features](#)

GetFrameInfoString**TVideoGrabber.GetFrameInfoString**[Prev](#)[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns information about the current frame as string

Declaration

function GetFrameInfoString (FrameInfoStringId: TFrameInfoStringId): **string**;

wchar_t * __fastcall GetFrameInfoString(TFrameInfoStringId FrameInfoStringId);

Function GetFrameInfoString (FrameInfoStringId as TFrameInfoStringId) as String

Description

Depending of the FrameInfoStringId parameter (type [TFrameInfoStringId](#)), this function returns the following string about the current video frame:

Examples:

fis_DVTimeCode

time code expressed in hh:mm:ss:ff, e.g. **00:15:42:08**

fis_DVDateTime

date/time stored on the tape, e.g. **"2009-03-02 10:15:26"**

(only for a DV camcorder playing a DV tape, this returns the date/time stored on the tape)

fis_TimeCode

frame time expressed in hh:mm:ss:hs, e.g. **00:00:00 53**

fis_FrameTime

time expressed in 100ns units, e.g. **5351096**

fis_FrameNumber

current frame number, e.g. **17**

fis_FullInfo

all the information above in a single string, separated by (A)...(B)...(C)...(D)...(E)...(F) delimiters to facilitate an eventual parsing, e.g.:

"(A) 17 (B) 5351096 (C) 00:00:00 53 (D) 2009-03-02 10:15:35 (E) 00:15:42:08 (F)"

See Also

[TCardinalDirection](#) [TAutoFileName](#) [TFrameCaptureDest](#) [TFrameGrabberRGBFormat](#)

[TOnFrameCaptureCompleted](#)
[TOnFrameOverlayUsingDC](#)
[TOnFrameOverlayUsingDIB](#)
[TTextOverlayAlign](#)
[AdjustOverlayAspectRatio](#)
[BurstCount](#)
[BurstInterval](#)
[BurstMode](#)
[BurstType](#)
[CaptureFrameSyncTo](#)
[CaptureFrameTo](#)
[DrawBitmapOverFrame](#)
[FrameCaptureHeight](#)
[FrameCaptureWidth](#)
[FrameCaptureWithoutOverlay](#)
[FrameCaptureZoomSize](#)
[FrameGrabber](#)
[FrameGrabberCurrentRGBFormat](#)
[FrameGrabberRGBFormat](#)
[FramerateDivider](#)
[GetFrameInfo](#)
[GetLastFrameAsHBITMAP](#)
[GetLastFrameAsTBitmap](#)
[GetLastFrameBitmapBits](#)
[GetLastFrameBitmapBits2](#)
[GetLastFrameWaitTimeoutMs](#)
[ImageOverlay](#)
[StretchToVideoSize](#)
[InFrameProgressEvent](#)
[JPEGPerformance](#)
[JPEGProgressiveDisplay](#)
[JPEGQuality](#)
[Last](#)
[BurstFrameCapture](#)
[FileName](#)
[Last](#)
[CaptureFrameTo](#)
[FileName](#)
[MouseWheelEventEnabled](#)
[OnDiskFull](#)
[OnFrameBitmap](#)
[OnFrameBitmapEventSynchronone](#)
[OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingDIB](#)
[OnFrameOverlayUsingVIDEOHDR](#)
[OnMouseDown](#)
[OnMouseMove](#)
[OnMouseUp](#)
[OnMouseWheel](#)
[OverlayAfterTransform](#)
[RefreshPlayerOverlays](#)
[SetFrameCaptureBounds](#)
[SetImageOverlay](#)
[AlphaBlend](#)
[SetImageOverlay](#)
[AlphaBlendValue](#)
[SetImageOverlay](#)
[ChromaKey](#)
[SetImageOverlay](#)
[ChromaKeyLeewayPercent](#)
[SetImageOverlay](#)
[ChromaKeyRGBColor](#)
[SetImageOverlay](#)
[Enabled](#)
[SetImageOverlay](#)
[Height](#)
[SetImageOverlay](#)
[LeftLocation](#)
[SetImageOverlay](#)
[RotationAngle](#)
[SetImageOverlay](#)
[StretchToVideoSize](#)
[SetImageOverlay](#)
[TargetDisplay](#)
[SetImageOverlay](#)
[TopLocation](#)
[SetImageOverlay](#)
[Transparent](#)
[SetImageOverlay](#)
[TransparentColorValue](#)
[SetImageOverlay](#)
[UseTransparentColor](#)
[SetImageOverlay](#)
[Width](#)
[SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#)
[SetImageOverlayFromHBitmap2](#)
[SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#)
[SetImageOverlayFromJPEGFile](#)
[SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#)
[SetImageOverlayFromTImage](#)
[SetImageOverlayFromTImage2](#)
[SetTextOverlay](#)
[Align](#)
[SetTextOverlay](#)
[BkColor](#)
[SetTextOverlay](#)
[CustomVar](#)
[SetTextOverlay](#)
[Enabled](#)
[SetTextOverlay](#)
[Font](#)
[SetTextOverlay](#)
[FontColor](#)
[SetTextOverlay](#)
[GradientColor](#)
[SetTextOverlay](#)
[GradientMode](#)
[SetTextOverlay](#)
[HighResFont](#)
[SetTextOverlay](#)
[Left](#)
[SetTextOverlay](#)
[Right](#)
[SetTextOverlay](#)
[Scrolling](#)
[SetTextOverlay](#)
[ScrollingSpeed](#)
[SetTextOverlay](#)
[Shadow](#)
[SetTextOverlay](#)
[ShadowColor](#)
[SetTextOverlay](#)
[ShadowDirection](#)
[SetTextOverlay](#)
[String](#)
[SetTextOverlay](#)
[TargetDisplay](#)
[SetTextOverlay](#)
[Top](#)
[SetTextOverlay](#)
[Transparent](#)
[ShapeOverlay](#)
[ShapeOverlayEnabled](#)
[ShapeOverlayList](#)
[StoragePath](#)
[TextOverlay](#)
[Align](#)
[TextOverlay](#)
[BkColor](#)
[TextOverlay](#)
[CreateCustomFont](#)
[TextOverlay](#)
[CreateCustomFont2](#)
[TextOverlay](#)
[Enabled](#)
[TextOverlay](#)
[Font](#)
[TextOverlay](#)
[FontColor](#)
[TextOverlay](#)
[Left](#)
[TextOverlay](#)
[Right](#)
[TextOverlay](#)
[Scrolling](#)
[TextOverlay](#)
[ScrollingSpeed](#)
[TextOverlay](#)
[Selector](#)
[TextOverlay](#)
[Shadow](#)
[TextOverlay](#)
[ShadowColor](#)
[TextOverlay](#)
[ShadowDirection](#)
[TextOverlay](#)
[String](#)
[TextOverlay](#)
[Top](#)
[TextOverlay](#)
[Transparent](#)
[TranslateMouseCoordinates](#)
[WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: Upgrade Your Documentation Process with a Help Authoring Tool

GetFWCam1394

TVideoGrabber.GetFWCam1394

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Retrieves a setting and min/max/default values of Firewire/GIE camera

Declaration

function GetFWCam1394 (FWCam1394ID: **string**; **var** Value: LongInt; **var** Flags: LongInt; **var** Capabilities: DWORD; **var** MinValue: LongInt; **var** MaxValue: LongInt; **var** Default: LongInt): Boolean;

bool **__fastcall** GetFWCam1394(System::wchar_t *FWCam1394ID, **int** &Value, **int** &Flags, **unsigned** &Capabilities, **int** &MinValue, **int** &MaxValue, **int** &Default);

Description

See [SetFWCam1394](#) for sample code.

Created with the Standard Edition of HelpNDoc: Maximize Your Documentation Capabilities with HelpNDoc's Project Analyzer

GetFWCam1394List

TVideoGrabber.GetFWCam1394List

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

[ber](#)

retrieves the list of possible custom settings for Firewire/GIE camera

Declaration

function GetFWCam1394List: **string**;

System::wchar_t * __fastcall GetFWCam1394List(void);

Description

Returns the list of the possible settings as a string made of text lines separated by characters

Created with the Standard Edition of HelpNDoc: [Transform Your Documentation Process with HelpNDoc's Project Analyzer](#)

GetImageOverlay_AlphaBlend

TVideoGrabber.GetImageOverlay_AlphaBlend

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetImageOverlay_AlphaBlend](#)

Declaration

function GetImageOverlay_AlphaBlend (**Index**: LongInt): Boolean;

bool __fastcall GetImageOverlay_AlphaBlend(int Index);

Description

retrieves the value set by [SetImageOverlay_AlphaBlend](#)

Created with the Standard Edition of HelpNDoc: [Write EPub books for the iPad](#)

GetImageOverlay_AlphaBlendValue

TVideoGrabber.GetImageOverlay_AlphaBlendValue

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetImageOverlay_AlphaBlendValue](#)

Declaration

function GetImageOverlay_AlphaBlendValue (**Index**: LongInt): LongInt;

int __fastcall GetImageOverlay_AlphaBlendValue(int Index);

Description

retrieves the value set by [SetImageOverlay_AlphaBlendValue](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Create Encrypted, Password-Protected PDFs](#)

GetImageOverlay_ChromaKey

TVideoGrabber.GetImageOverlay_ChromaKey

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetImageOverlay_ChromaKey](#)

Declaration

function GetImageOverlay_ChromaKey (**Index**: LongInt): Boolean;

bool __fastcall GetImageOverlay_ChromaKey(int Index);

Description

retrieves the value set by [SetImageOverlay_ChromaKey](#)

Created with the Standard Edition of HelpNDoc: [Produce Kindle eBooks easily](#)

GetImageOverlay_ChromaKeyLeewayPercent

TVideoGrabber.GetImageOverlay_ChromaKeyLeewayPercent

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetImageOverlay_ChromaKeyLeewayPercent](#)

Declaration

function GetImageOverlay_ChromaKeyLeewayPercent (**Index**: LongInt): LongInt;

int __fastcall GetImageOverlay_ChromaKeyLeewayPercent(int Index);

Description

retrieves the value set by [SetImageOverlay_ChromaKeyLeewayPercent](#)

Created with the Standard Edition of HelpNDoc: [Don't be left in the past: convert your WinHelp HLP help files to CHM with HelpNDoc](#)

GetImageOverlay_ChromaKeyRGBColor

TVideoGrabber.GetImageOverlay_ChromaKeyRGBColor

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetImageOverlay_ChromaKeyRGBColor](#)

Declaration

function GetImageOverlay_ChromaKeyRGBColor (**Index**: LongInt): LongInt;

int __fastcall GetImageOverlay_ChromaKeyRGBColor(int Index);

Description

retrieves the value set by [SetImageOverlay_ChromaKeyRGBColor](#)

Created with the Standard Edition of HelpNDoc: [Revolutionize your documentation process with HelpNDoc's online capabilities](#)

GetImageOverlay_Enabled

TVideoGrabber.GetImageOverlay_Enabled

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetImageOverlay_Enabled](#)

Declaration

function GetImageOverlay_Enabled (**Index**: LongInt): Boolean;

bool __fastcall GetImageOverlay_Enabled(int Index);

Description

retrieves the value set by [SetImageOverlay_Enabled](#)

Created with the Standard Edition of HelpNDoc: [Streamline Your Documentation Process with a Help Authoring Tool](#)

GetImageOverlay_Height

TVideoGrabber.GetImageOverlay_Height

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetImageOverlay_Height](#)

Declaration

function GetImageOverlay_Height (**Index**: LongInt): LongInt;

int __fastcall GetImageOverlay_Height(**int** Index);

Description

retrieves the value set by [SetImageOverlay_Height](#)

Created with the Standard Edition of HelpNDoc: [Powerful and User-Friendly Help Authoring Tool for Markdown Documents](#)

GetImageOverlay_LeftLocation

TVideoGrabber.GetImageOverlay_LeftLocation

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetImageOverlay_LeftLocation](#)

Declaration

function GetImageOverlay_LeftLocation (**Index**: LongInt): LongInt;

int __fastcall GetImageOverlay_LeftLocation(**int** Index);

Description

retrieves the value set by [SetImageOverlay_LeftLocation](#)

Created with the Standard Edition of HelpNDoc: [Streamline Your Documentation Creation with a Help Authoring Tool](#)

GetImageOverlay_RotationAngle

TVideoGrabber.GetImageOverlay_RotationAngle

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetImageOverlay_RotationAngle](#)

Declaration

function GetImageOverlay_RotationAngle (**Index**: LongInt): Double;

double __fastcall GetImageOverlay_RotationAngle(**int** Index);

Description

retrieves the value set by [SetImageOverlay_RotationAngle](#)

Created with the Standard Edition of HelpNDoc: [Free HTML Help documentation generator](#)

GetImageOverlay_StretchToVideoSize

TVideoGrabber.GetImageOverlay_StretchToVideoSize

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetImageOverlay_StretchToVideoSize](#)

Declaration

function GetImageOverlay_StretchToVideoSize (**Index**: LongInt): Boolean;

bool __fastcall GetImageOverlay_StretchToVideoSize(**int** Index);

Description

retrieves the value set by [SetImageOverlay_StretchToVideoSize](#)

Created with the Standard Edition of HelpNDoc: [Generate EPub eBooks with ease](#)

GetImageOverlay_TargetDisplay

TVideoGrabber.GetImageOverlay_TargetDisplay

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetImageOverlay_TargetDisplay](#)

Declaration

function GetImageOverlay_TargetDisplay(**Index**: LongInt): LongInt;

int __fastcall GetImageOverlay_TargetDisplay(**int** Index);

Description

retrieves the value set by [SetImageOverlay_TargetDisplay](#)

Created with the Standard Edition of HelpNDoc: [Free Qt Help documentation generator](#)

GetImageOverlay_TopLocation

TVideoGrabber.GetImageOverlay_TopLocation

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetImageOverlay_TopLocation](#)

Declaration

function GetImageOverlay_TopLocation (**Index**: LongInt): LongInt;

int __fastcall GetImageOverlay_TopLocation(**int** Index);

Description

retrieves the value set by [SetImageOverlay_TopLocation](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Documentation Efficiency with a Help Authoring Tool](#)

GetImageOverlay_Transparent

TVideoGrabber.GetImageOverlay_Transparent

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetImageOverlay_Transparent](#)

Declaration

function GetImageOverlay_Transparent (Index: LongInt): Boolean;

bool __fastcall GetImageOverlay_Transparent(int Index);

Description

retrieves the value set by [SetImageOverlay_Transparent](#)

Created with the Standard Edition of HelpNDoc: [Free Kindle producer](#)

GetImageOverlay_TransparentColorValue

TVideoGrabber.GetImageOverlay_TransparentColorValue

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetImageOverlay_TransparentColorValue](#)

Declaration

function GetImageOverlay_TransparentColorValue (Index: LongInt): LongInt;

int __fastcall GetImageOverlay_TransparentColorValue(int Index);

Description

retrieves the value set by [SetImageOverlay_TransparentColorValue](#)

Created with the Standard Edition of HelpNDoc: [Transform Your Word Doc into a Professional-Quality eBook with HelpNDoc](#)

GetImageOverlay_UseTransparentColor

TVideoGrabber.GetImageOverlay_UseTransparentColor

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetImageOverlay_UseTransparentColor](#)

Declaration

function GetImageOverlay_UseTransparentColor (Index: LongInt): Boolean;

bool __fastcall GetImageOverlay_UseTransparentColor(int Index);

Description

retrieves the value set by [SetImageOverlay_UseTransparentColor](#)

Created with the Standard Edition of HelpNDoc: [Transform Your Documentation Process with HelpNDoc's Project Analyzer](#)

GetImageOverlay_Width

TVideoGrabber.GetImageOverlay_Width

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetImageOverlay_Width](#)

Declaration

function GetImageOverlay_Width (**Index**: LongInt): LongInt;

int __fastcall GetImageOverlay_Width(**int** Index);

Description

retrieves the value set by [SetImageOverlay_Width](#)

Created with the Standard Edition of HelpNDoc: [Modernize your help files with HelpNDoc's WinHelp HLP to CHM conversion tool](#)

GetItemNameFromList

TVideoGrabber.GetItemNameFromList

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Name of a list item from index

Declaration

function GetItemNameFromList(List: **string**; ItemIndex: LongInt): **string**;

System::wchar_t * __fastcall GetItemNameFromList(System::wchar_t *List, **int** ItemIndex)

Function GetItemNameFromList (List as String, ItemIndex as Long) as String

Description

Used to retrieve the name of an item in a text list by its index

Created with the Standard Edition of HelpNDoc: [Free Kindle producer](#)

GetLastAverageStreamValue

TVideoGrabber.GetLastAverageStreamValue

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns the current average stream value

Declaration

function GetLastAverageStreamValue (StreamType: TStreamType): LongInt;

int __fastcall GetLastAverageStreamValue(TStreamType StreamType);

Function GetLastAverageStreamValue(StreamType as TStreamType)

Description

Used to get the current average stream value, depending on [StreamType](#) :

st_Video: current average value of the RGB pixels

st_Audio: current average audio level

Created with the Standard Edition of HelpNDoc: [Transform Your Help Documentation Process with a Help Authoring Tool](#)

GetLastErrorMassage

TVideoGrabber.GetLastErrorMassage

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

[ber](#)

returns the last error message during the URL connection, if any

Declaration

function GetLastErrorMessage: **string**;

System::UnicodeString __fastcall GetLastErrorMessage();

Description

returns the last error message that eventually occurred during the URL connection of the Datastead RTSP/RTMP/HTTP/ONVIF Source filter

See Also

[IR Cut Filter of Axis cameras](#) [TDiscoveryCallbackStatus](#) [TOnONVIFDiscoveryCompletedNotification](#) [TONVIFDeviceInfo](#) [Video stream of IP camera with audio of PC microphone or other audio capture device](#) [ExtraDLLPath](#) [IPCameraURL](#) [IsURLResponding](#) [IsURLVideoStreamAvailable](#) [OnONVIFDiscoveryCompletedNotification](#) [ONVIF_GetStr](#) [ONVIF_SetStr](#) [ONVIFCancelDiscovery](#) [ONVIFDeviceInfo](#) [ONVIFDiscoverCameras](#) [IPRange](#) [ONVIFDiscoverCameras](#) [Multicast](#) [ONVIFEnumCamerasDiscovered](#) [ONVIFPTZGetLimits](#) [ONVIFPTZGetPosition](#) [ONVIFPTZPreset](#) [ONVIFPTZSendAuxiliaryCommand](#) [ONVIFPTZSetPosition](#) [ONVIFPTZStartMove](#) [ONVIFPTZStopMove](#) [ONVIFSnapshot](#) [OpenURLAsync](#) [OpenURLAsyncStatus](#) [SendIPCameraCommand](#) [SetDatasteadFilterDllName](#) [SetIPCameraSetting](#)

Created with the Standard Edition of HelpNDoc: [Streamline Your Documentation Creation with a Help Authoring Tool](#)

GetLastFrameAsHBITMAP

TVideoGrabber.GetLastFrameAsHBITMAP

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns the last video frame as a HBITMAP memory bitmap handle

Declaration

function GetLastFrameAsHBITMAP (BufferIndex: LongInt; WithOverlays: Boolean; SrcLeftLocation: LongInt; SrcTopLocation: LongInt; SrcWidth: LongInt; SrcHeight: LongInt; DestWidth: LongInt; DestHeight: LongInt; BitmapColorBitCount: LongInt): HBITMAP;

HBITMAP __fastcall GetLastFrameAsHBITMAP(int BufferIndex, bool WithOverlays, int SrcLeftLocation, int SrcTopLocation, int SrcWidth, int SrcHeight, int DestWidth, int DestHeight, int BitmapColorBitCount);

function GetLastFrameAsHBITMAP (BufferIndex as Long, WithOverlays as Bool, SrcLeftLocation as Long, SrcTopLocation as Long, SrcWidth as Long, SrcHeight as Long, DestWidth as Long, DestHeight as Long, BitmapColorBitCount as Long) as Long

Description

Used to get the last video frame as a memory bitmap.

The function returns a **HBITMAP** bitmap handle.

*Don't forget to invoke **DeleteObject (bitmap handle)** to release the memory when you have done with the bitmap, otherwise you will quick run out of memory.*

BufferIndex: LongInt;

0 = the current video frame (usual value)

1 = the previous video frame (n-1)

2 = the video frame n-2

...

(the size of the buffer may vary, usually don't try to use frames older than n-4)

If the index is out of the buffer the function return 0

WithOverlays: Boolean;
if false, the native frame without overlays is returned
if true, the frame with overlays applied is returned

SrcLeftLocation: LongInt;
x location of the rectangle to capture (usually 0 for a full frame)

SrcTopLocation: LongInt;
y location of the rectangle to capture (usually 0 for a full frame)

SrcWidth: LongInt;
width of the rectangle to capture (usually 0 for a full width)

SrcHeight: LongInt;
height of the rectangle to capture (usually 0 for a full height)

DestWidth: LongInt;
width of the desired bitmap (0 will return a bitmap having the source width)

DestHeight: LongInt;
height of the desired bitmap (0 will return a bitmap having the source height)

BitmapColorBitCount: LongInt
color bit count of the desired bitmap. Valid values are 32, 24, 16, 15, 8.
0 returns a bitmap having the same color count than the video frame (usually 24 or 32, depending on the [FrameGrabberRGBFormat](#) setting)

E.g.:

BitmapHandle = VideoGrabber.GetLastFrameAsTBitmap (0, false, 0, 0, 0, 0, 0, 0, 0)
returns the last video frame without overlays in its current size

BitmapHandle = VideoGrabber.GetLastFrameAsTBitmap (0, false, 0, 0, 0, 0, 320, 240, 8)
returns the previous video frame with overlays as a 320x240 bitmap, 8 bits color

See Also

[TAutoFileName](#) [TFrameCaptureDest](#) [TOnFrameCaptureCompleted](#) [BurstCount](#) [BurstInterval](#) [BurstMode](#) [BurstType](#) [CaptureFrameSyncTo](#) [CaptureFrameTo](#) [FrameCaptureHeight](#) [FrameCaptureWidth](#) [FrameCaptureWithoutOverlay](#) [FrameCaptureZoomSize](#) [GetFrameInfoString](#) [GetLastFrameAsTBitmap](#) [GetLastFrameBitmapBits](#) [GetLastFrameBitmapBits2](#) [GetLastFrameWaitTimeoutMs](#) [JPEGPerformance](#) [JPEGProgressiveDisplay](#) [JPEGQuality](#) [Last](#) [BurstFrameCapture](#) [FileName](#) [Last](#) [CaptureFrameTo](#) [FileName](#) [OnDiskFull](#) [OnFrameBitmap](#) [SetFrameCaptureBounds](#) [StoragePath](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [From Word to ePub or Kindle eBook: A Comprehensive Guide](#)

GetLastFrameAsTBitmap

TVideoGrabber.GetLastFrameAsTBitmap

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns the last video frame as a TBitmap

Declaration

function GetLastFrameAsTBitmap (BufferIndex: LongInt; WithOverlays: Boolean; SrcLeftLocation: LongInt; SrcTopLocation: LongInt; SrcWidth: LongInt; SrcHeight: LongInt; DestWidth: LongInt; DestHeight: LongInt; BitmapColorBitCount: LongInt): TBitmap;

Graphics::TBitmap* __fastcall GetLastFrameAsTBitmap(int BufferIndex, bool WithOverlays, int SrcLeftLocation, int SrcTopLocation, int SrcWidth, int SrcHeight, int DestWidth, int DestHeight, int BitmapColorBitCount);

n/a

Description

Used to get a copy of the last video frame as a Delphi or C++Builder TBitmap (not available in the DLL and OCX versions)

See [GetLastFrameAsHBitmap](#) for the description of the parameters (it has exactly the same parameters).

See Also

[TAutoFileName](#) [TFrameCaptureDest](#) [TOnFrameCaptureCompleted](#) [BurstCount](#) [BurstInterval](#) [BurstMode](#) [BurstType](#) [CaptureFrameSyncTo](#) [CaptureFrameTo](#) [FrameCaptureHeight](#) [FrameCaptureWidth](#) [FrameCaptureWithoutOverlay](#) [FrameCaptureZoomSize](#) [GetFrameInfoString](#) [GetLastFrameAsHBITMAP](#) [GetLastFrameBitmapBits](#) [GetLastFrameBitmapBits2](#) [GetLastFrameWaitTimeoutMs](#) [JPEGPerformance](#) [JPEGProgressiveDisplay](#) [JPEGQuality](#) [Last](#) [BurstFrameCapture](#) [FileName](#) [Last](#) [CaptureFrameTo](#) [FileName](#) [OnDiskFull](#) [OnFrameBitmap](#) [SetFrameCaptureBounds](#) [StoragePath](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: Effortlessly upgrade your WinHelp HLP help files to CHM with HelpNDoc

GetLastFrameBitmapBits

TVideoGrabber.GetLastFrameBitmapBits

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns a pointer to the bitmap bits of the last video frame

Declaration

function GetLastFrameBitmapBits (BufferIndex: LongInt; WithOverlays: Boolean; ReleaseFrame: Boolean): Pointer;

void * __fastcall GetLastFrameBitmapBits(int BufferIndex, **bool** WithOverlays, **bool** ReleaseFrame);

Description

Returns a pointer to the bitmap bits of the last video frame.

Invoke this function to lock and get access to the bitmap bits.

Immediately after processing the bitmap bits, invoke it again with the "ReleaseFrame" set to false to unlock the sample.

E.g.:

...

```
BYTE *pBits = GetLastFrameBitmapBits (0, true, false)
```

```
// do what you need with the bitmap bits
```

```
GetLastFrameBitmapBits (0, true, true)
```

...

Note: To get the bitmap dimensions, invoke a first time [GetLastFrameAsBitmapBits2](#).

See Also

[TAutoFileName](#) [TFrameCaptureDest](#) [TOnFrameCaptureCompleted](#) [BurstCount](#) [BurstInterval](#) [BurstMode](#) [BurstType](#) [CaptureFrameSyncTo](#) [CaptureFrameTo](#) [FrameCaptureHeight](#) [FrameCaptureWidth](#) [FrameCaptureWithoutOverlay](#) [FrameCaptureZoomSize](#) [GetFrameInfoString](#) [GetLastFrameAsHBITMAP](#) [GetLastFrameAsTBitmap](#) [GetLastFrameBitmapBits](#) [GetLastFrameBitmapBits2](#) [GetLastFrameWaitTimeoutMs](#) [JPEGPerformance](#) [JPEGProgressiveDisplay](#) [JPEGQuality](#) [Last](#) [BurstFrameCapture](#) [FileName](#) [Last](#) [CaptureFrameTo](#) [FileName](#) [OnDiskFull](#) [OnFrameBitmap](#) [SetFrameCaptureBounds](#) [StoragePath](#) [WebcamStillCaptureButton](#)

GetLastFrameBitmapBits2

TVideoGrabber.GetLastFrameBitmapBits2

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns a pointer to the bitmap bits of the last video frame, with the bitmap format information

Declaration

function GetLastFrameBitmapBits2 (BufferIndex: LongInt; WithOverlays: Boolean; ReleaseFrame: Boolean; **out** BitmapWidth: LongInt; **out** BitmapHeight: LongInt; **out** BitmapLineSize: LongInt; **out** BitmapSize: LongInt; **out** BitmapBitsPerPixel: LongInt): Pointer;

void *__fastcall GetLastFrameBitmapBits2(int BufferIndex, **bool** WithOverlays, **bool** ReleaseFrame, /* outint &BitmapWidth, /* outint &BitmapHeight, /* outint &BitmapLineSize, /* outint &BitmapSize, /* outint &BitmapBitsPerPixel);

Description

Returns a pointer to the bitmap bits of the last video frame.

Invoke this function to lock and get access to the bitmap bits.

Immediately after processing the bitmap bits, invoke it again with the "ReleaseFrame" set to false to unlock the sample.

E.g.:

```
...
int BitmapWidth, BitmapHeight, BitmapLineSize, BitmapSize, BitmapBitsPerPixel;
BYTE *pBits = GetLastFrameBitmapBits2 (0, true, false, &BitmapWidth, &BitmapHeight, &BitmapLineSize,
&BitmapSize, &BitmapBitsPerPixel);
CopyMemory (pDest, pBits, BitMapSize);
GetLastFrameBitmapBits (0, true, true, null, null, null, null, null);
...
```

See Also

[TAutoFileName](#) [TFrameCaptureDest](#) [TOnFrameCaptureCompleted](#) [BurstCount](#) [BurstInterval](#) [BurstMode](#) [BurstType](#) [CaptureFrameSyncTo](#) [CaptureFrameTo](#) [FrameCaptureHeight](#) [FrameCaptureWidth](#) [FrameCaptureWithoutOverlay](#) [FrameCaptureZoomSize](#) [GetFrameInfoString](#) [GetLastFrameAsHBITMAP](#) [GetLastFrameAsTBitmap](#) [GetLastFrameBitmapBits](#) [GetLastFrameWaitTimeoutMs](#) [JPEGPerformance](#) [JPEGProgressiveDisplay](#) [JPEGQuality](#) [Last](#) [BurstFrameCapture](#) [FileName](#) [Last](#) [CaptureFrameTo](#) [FileName](#) [OnDiskFull](#) [OnFrameBitmap](#) [SetFrameCaptureBounds](#) [StoragePath](#) [WebcamStillCaptureButton](#)

GetLogString

TVideoGrabber.GetLogString

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns a TLogType value (returned by the OnLog event) as string

Declaration

function GetLogString (Value: TLogType): **string**;

wchar_t *GetLogString (TLogType Value);

Function GetLogString (Value as TLogType) as string

Description

Used to retrieve the string value of the [TLogType](#) parameter returned by the [OnLog](#) event.

E.g.

```
procedure TForm1.VideoGrabberLog (Sender: TObject; LogType: TLogType; Severity, InfoMsg:
var
    sLog: string;
begin
    sLog := VideoGrabber.GetLogString (LogType));
    Memo1.Lines.Add (sLog);
end;
```

Created with the Standard Edition of HelpNDoc: [Produce electronic books easily](#)

GetMiscDeviceControl

TVideoGrabber.GetMiscDeviceControl

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Used to retrieve specific values or states on some video capture devices.

Declaration

function GetMiscDeviceControl (DeviceDataType: TMiscDeviceControl; **Index**: LongInt): LongInt;

int __fastcall GetMiscDeviceControl(TMiscDeviceControl MiscDeviceControl, **int** Index);

Function GetMiscDeviceControl (MiscDeviceControl as TMiscDeviceControl, Index as Long) as Long

Description

Used to retrieve specific values or states on some video capture devices that support them, e.g. the GPIO, VPD, etc...

The use of this function is explained in the [Miscellaneous device control](#) chapter.

See Also

Created with the Standard Edition of HelpNDoc: [Write eBooks for the Kindle](#)

GetNearestVideoHeight

TVideoGrabber.GetNearestVideoHeight

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Retrieves the nearest video height to a given video width and height.

Declaration

function GetNearestVideoHeight(PreferredVideoWidth: LongInt; PreferredVideoHeight: LongInt): LongInt;

int __fastcall GetNearestVideoHeight(**int** PreferredVideoWidth, **int** PreferredVideoHeight)

Function GetNearestVideoHeight(PreferredVideoWidth As Long, PreferredVideoHeight As Long) As Long

Description

Retrieves the nearest video height to a given video width and height, available on the [current video capture device](#) .

Unlike [UseNearestVideoSize](#) , this function simply request the nearest height available for the specified size, without selecting it on the [current video capture device](#) .

See Also

[GetNearestVideoSize](#) [GetNearestVideoWidth](#) [GetVideoHeightFromIndex](#) [GetVideoSizeFromIndex](#) [GetVideoWidthFromIndex](#) [UseNearestVideoSize](#) [VideoFormat](#) [VideoFormatIndex](#) [VideoSize](#) [VideoSizeIndex](#) [VideoSizes](#) [VideoSizesCount](#)

Created with the Standard Edition of HelpNDoc: [What is a Help Authoring tool?](#)

GetNearestVideoSize

TVideoGrabber.GetNearestVideoSize

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Retrieves the nearest video size to a given video width and height.

Declaration

procedure GetNearestVideoSize(PreferredVideoWidth: LongInt; PreferredVideoHeight: LongInt; **out** NearestVideoWidth: LongInt; **out** NearestVideoHeight: LongInt);

void __fastcall GetNearestVideoSize(**int** PreferredVideoWidth, **int** PreferredVideoHeight, **int** &NearestVideoWidth, **int** &NearestVideoHeight)

Sub GetNearestVideoSize(PreferredVideoWidth As Long, PreferredVideoHeight As Long, Nearest_VideoWidth, Nearest_VideoHeight)

Description

Retrieves the nearest video size to a given video width and height, available on the [current video capture device](#) .

Unlike [UseNearestVideoSize](#) , this function simply request the nearest video size, without selecting it on the [current video capture device](#) .

See Also

[GetNearestVideoHeight](#) [GetNearestVideoWidth](#) [GetVideoHeightFromIndex](#) [GetVideoSizeFromIndex](#) [GetVideoWidthFromIndex](#) [UseNearestVideoSize](#) [VideoFormat](#) [VideoFormatIndex](#) [VideoSize](#) [VideoSizeIndex](#) [VideoSizes](#) [VideoSizesCount](#)

Created with the Standard Edition of HelpNDoc: [Experience the power of a responsive website for your documentation](#)

GetNearestVideoWidth

TVideoGrabber.GetNearestVideoWidth

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Retrieves the nearest video width to a given video width and height.

Declaration

function GetNearestVideoWidth(PreferredVideoWidth: LongInt; PreferredVideoHeight: LongInt): LongInt;

int __fastcall GetNearestVideoWidth(**int** PreferredVideoWidth, **int** PreferredVideoHeight)

Function GetNearestVideoWidth(PreferredVideoWidth As Long, PreferredVideoHeight As Long) As Long

Description

Retrieves the nearest video width to a given video width and height, available on the [current video capture device](#) .

Unlike [UseNearestVideoSize](#) , this function simply request the nearest width available for the specified size, without selecting it on the [current video capture device](#) .

See Also

[GetNearestVideoHeight](#) [GetNearestVideoSize](#) [GetVideoHeightFromIndex](#) [GetVideoSizeFromIndex](#) [GetVideoWidthFromIndex](#) [UseNearestVideoSize](#) [VideoFormat](#) [VideoFormatIndex](#) [VideoSize](#) [VideoSizeIndex](#) [VideoSizes](#) [VideoSizesCount](#)

Created with the Standard Edition of HelpNDoc: [Free EPub producer](#)

GetPixelsDistance

TVideoGrabber.GetPixelsDistance

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns the distance between 2 pixels

Declaration

function GetPixelsDistance (x1: LongInt; y1: LongInt; x2: LongInt; y2: LongInt): Double;

double __fastcall GetPixelsDistance(int x1, int y1, int x2, int y2);

Function GetPixelsDistance(x1 As Long, y1 As Long, x2 as Long, y2 as Long) As Double

Description

Used to retrieve the distance, in pixels, between a pixel located at the (x1,y1) coordinate and a pixel located at the (x2,y2) coordinate

See Also

[Image overlays](#) [GetRGBPixelAt](#) [ImageOverlay](#) [AlphaBlend](#) [ImageOverlay](#) [AlphaBlendValue](#) [ImageOverlay](#) [Height](#) [ImageOverlay](#) [LeftLocation](#) [ImageOverlay](#) [TopLocation](#) [ImageOverlay](#) [Transparent](#) [ImageOverlay](#) [TransparentColorValue](#) [ImageOverlay](#) [UseTransparentColor](#) [ImageOverlay](#) [VideoAlignment](#) [ImageOverlay](#) [Width](#) [ImageOverlay](#) [Enabled](#) [ImageOverlay](#) [Selector](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlay](#) [FromBMPFile](#) [SetImageOverlay](#) [FromHBitmap](#) [SetImageOverlay](#) [FromHBitmap2](#) [SetImageOverlay](#) [FromImageFile](#) [SetImageOverlay](#) [FromImageFile2](#) [SetImageOverlay](#) [FromJPEGFile](#) [SetImageOverlay](#) [FromTBitmap](#) [SetImageOverlay](#) [FromTBitmap2](#) [SetImageOverlay](#) [FromTImage](#) [SetImageOverlay](#) [FromTImage2](#)

Created with the Standard Edition of HelpNDoc: [Generate EPub eBooks with ease](#)

GetPlaylist

TVideoGrabber.GetPlaylist

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns the current playlist.

Declaration

function GetPlaylist: string;

wchar_t *GetPlaylist();

Function GetPlaylist as string

Description

Returns the current playlist as a list of strings separated by CR/LF characters (char (13) and char(10))

See the "[Using the playlist](#)" chapter for more information about the Playlist feature.

See Also

[TPlaylist](#) [Video formats](#) [IsPlaylistActive](#) [OnPlayerEndOfPlaylist](#) [Playlist](#) [PlaylistIndex](#)

Created with the Standard Edition of HelpNDoc: [Transform Your CHM Help File Creation Process with HelpNDoc](#)

GetRGBPixelAt

TVideoGrabber.GetRGBPixelAt

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns the RGB value of the specified pixel.

Declaration

function GetRGBPixelAt (x: LongInt; y: LongInt): Longint;

int GetRGBPixelAt(**int** x, **int** y);

Function GetRGBPixelAt (x as Long, y as Long) as Long

Description

Used to retrieve the RGB value of the specified pixel at the (x, y) location in the current video frame.

Note: this function must be invoked from one of the OnFrameOverlayUsing... events (e.g. from the [OnFrameOverlayUsingDC](#) event).

This function returns the RGB value at the x, y location on the video frame, with:
x between 0 and (VideoSourceWidth - 1)
and
y between 0 and <= (VideoSourceHeight - 1)

You can extract each pixel color from the value returned by using the following sample code:

DELPHI

```
procedure TfrmMainForm.VideoGrabberFrameOverlayUsingDC(Sender: TObject;
  Dc: HDC; FrameNumber: Cardinal; FrameTime: Int64; FrameId: Integer);
var
  RGBValue: LongInt;
  R, G, B: LongInt;
  x, y: LongInt;
begin
  x := 10; // e.g. row 10
  y := 15; // and column 15
  RGBValue := VideoGrabber.GetRGBPixelAt (x, y);
  R := RGBValue and $FF;
  G := (RGBValue shr 8) and $FF;
  B := (RGBValue shr 16) and $FF;
end;
```

C++

```
void __fastcall TfrmMainForm::VideoGrabberFrameOverlayUsingDC(
    TObject *Sender, HDC Dc, DWORD FrameNumber, Int64 FrameTime,
    int FrameId)

    int RGBValue, R, G, B, x, y;

    x = 10; // e.g. row 10
    y = 15; // and column 15

    RGBValue = VideoGrabber->GetRGBPixelAt (x, y);

    R = RGBValue && 0xFF;
    G = (RGBValue >> 8) && 0xFF;
    B = (RGBValue >> 16) && 0xFF;
```

C#

```
private void axVideoGrabberNET1_OnFrameOverlayUsingDC(object sender, Axvidgrab_NET.IVideo

...
Color RGBColor = System.Drawing.Color.FromArgb (axVideoGrabberNET1.GetRGBPixelAt(10, 1
textBoxR.Text = RGBColor.R.ToString();
textBoxG.Text = RGBColor.G.ToString();
textBoxB.Text = RGBColor.B.ToString();
...
```

See Also

[Image overlays](#) [GetPixelsDistance](#) [ImageOverlay](#) [AlphaBlend](#) [ImageOverlay](#) [AlphaBlendValue](#) [ImageOverlay](#) [Height](#) [ImageOverlay](#) [LeftLocation](#) [ImageOverlay](#) [TopLocation](#) [ImageOverlay](#) [Transparent](#) [ImageOverlay](#) [TransparentColorValue](#) [ImageOverlay](#) [UseTransparentColor](#) [ImageOverlay](#) [VideoAlignment](#) [ImageOverlay](#) [Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)

Created with the Standard Edition of HelpNDoc: Protect Your Confidential PDFs with These Simple Security Measures

GetTextOverlay_Align

TVideoGrabber.GetTextOverlay_Align

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetTextOverlay_Align](#)

Declaration

function GetTextOverlay_Align(Index: LongInt): TTextOverlayAlign;

TTextOverlayAlign __fastcall GetTextOverlay_Align(int Index);

Description

retrieves the value set by [SetTextOverlay_Align](#)

Created with the Standard Edition of HelpNDoc: [Save time and frustration with HelpNDoc's WinHelp HLP to CHM conversion feature](#)

GetTextOverlay_AlphaBlend

TVideoGrabber.GetTextOverlay_AlphaBlend

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetTextOverlay_AlphaBlend](#)

Declaration

function GetTextOverlay_AlphaBlend (**Index**: LongInt): Boolean;

bool __fastcall GetTextOverlay_AlphaBlend(**int** Index);

Description

retrieves the value set by [SetTextOverlay_AlphaBlend](#)

See Also

[GetTextOverlay_AlphaBlendValue](#) [SetTextOverlay_Align](#) [SetTextOverlay_AlphaBlend](#)
[SetTextOverlay_AlphaBlendValue](#) [SetTextOverlay_BkColor](#) [SetTextOverlay_CustomVar](#)
[SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#) [SetTextOverlay_GradientColor](#)
[SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#)
[SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#)
[SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#)
[SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#)
[TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_VideoAlignment](#)

Created with the Standard Edition of HelpNDoc: [Easily create Web Help sites](#)

GetTextOverlay_AlphaBlendValue

TVideoGrabber.GetTextOverlay_AlphaBlendValue

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetTextOverlay_AlphaBlendValue](#)

Declaration

function GetTextOverlay_AlphaBlendValue (**Index**: LongInt): LongInt;

int __fastcall GetTextOverlay_AlphaBlendValue(**int** Index);

Description

retrieves the value set by [SetTextOverlay_AlphaBlendValue](#)

See Also

[GetTextOverlay_AlphaBlend](#) [SetTextOverlay_Align](#) [SetTextOverlay_AlphaBlend](#)
[SetTextOverlay_AlphaBlendValue](#) [SetTextOverlay_BkColor](#) [SetTextOverlay_CustomVar](#)
[SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#) [SetTextOverlay_GradientColor](#)
[SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#)
[SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#)
[SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#)
[SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#)
[TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_VideoAlignment](#)

Created with the Standard Edition of HelpNDoc: [Easily Add Encryption and Password Protection to Your PDFs](#)

GetTextOverlay_BkColor

TVideoGrabber.GetTextOverlay_BkColor

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetTextOverlay_BkColor](#)

Declaration

function GetTextOverlay_BkColor(Index: LongInt): TColor;

Graphics::TColor __fastcall GetTextOverlay_BkColor(int Index);

Description

retrieves the value set by [SetTextOverlay_BkColor](#)

Created with the Standard Edition of HelpNDoc: [Free Kindle producer](#)

GetTextOverlay_Enabled

TVideoGrabber.GetTextOverlay_Enabled

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetTextOverlay_Enabled](#)

Declaration

function GetTextOverlay_Enabled(Index: LongInt): Boolean;

bool __fastcall GetTextOverlay_Enabled(int Index);

Description

retrieves the value set by [SetTextOverlay_Enabled](#)

Created with the Standard Edition of HelpNDoc: [Make Documentation Review a Breeze with HelpNDoc's Advanced Project Analyzer](#)

GetTextOverlay_Font

TVideoGrabber.GetTextOverlay_Font

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetTextOverlay_Font](#)

Declaration

function GetTextOverlay_Font(Index: LongInt): TFont;

Graphics::TFont* __fastcall GetTextOverlay_Font(int Index);

Description

retrieves the value set by [SetTextOverlay_Font](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Documentation Efficiency with a Help Authoring Tool](#)

GetTextOverlay_GradientColor

TVideoGrabber.GetTextOverlay_GradientColor

[Prev](#)[Next](#)[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetTextOverlay_GradientColor](#)

Declaration

function GetTextOverlay_GradientColor(Index: LongInt): TColor;

Graphics::TColor __fastcall GetTextOverlay_GradientColor(int Index);

Description

retrieves the value set by [SetTextOverlay_GradientColor](#)

Created with the Standard Edition of HelpNDoc: [Write EPub books for the iPad](#)

GetTextOverlay_GradientMode

TVideoGrabber.GetTextOverlay_GradientMode

[Prev](#)[Next](#)[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetTextOverlay_GradientMode](#)

Declaration

function GetTextOverlay_GradientMode(Index: LongInt): TTextOverlayGradientMode;

TTextOverlayGradientMode __fastcall GetTextOverlay_GradientMode(int Index);

Description

retrieves the value set by [SetTextOverlay_GradientMode](#)

Created with the Standard Edition of HelpNDoc: [Free iPhone documentation generator](#)

GetTextOverlay_HighResFont

TVideoGrabber.GetTextOverlay_HighResFont

[Prev](#)[Next](#)[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetTextOverlay_HighResFont](#)

Declaration

function GetTextOverlay_HighResFont(Index: LongInt): Boolean;

bool __fastcall GetTextOverlay_HighResFont(int Index);

Description

retrieves the value set by [SetTextOverlay_HighResFont](#)

Created with the Standard Edition of HelpNDoc: [Achieve Professional Documentation Results with a Help Authoring Tool](#)

GetTextOverlay_Left

TVideoGrabber.GetTextOverlay_Left

[Prev](#)[Next](#)[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetTextOverlay_Left](#)

Declaration

function GetTextOverlay_Left(Index: LongInt): LongInt;

int __fastcall GetTextOverlay_Left(int Index);

Description

retrieves the value set by [SetTextOverlay_Left](#)

Created with the Standard Edition of HelpNDoc: [Bring your WinHelp HLP help files into the present with HelpNDoc's easy CHM conversion](#)

GetTextOverlay_Right

TVideoGrabber.GetTextOverlay_Right

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetTextOverlay_Right](#)par

Declaration

function GetTextOverlay_Right(Index: LongInt): LongInt;

int __fastcall GetTextOverlay_Right(int Index);

Description

retrieves the value set by [SetTextOverlay_Right](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Documentation Efficiency with a Help Authoring Tool](#)

GetTextOverlay_Scrolling

TVideoGrabber.GetTextOverlay_Scrolling

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetTextOverlay_Scrolling](#)par

Declaration

function GetTextOverlay_Scrolling(Index: LongInt): Boolean;

bool __fastcall GetTextOverlay_Scrolling(int Index);

Description

retrieves the value set by [SetTextOverlay_Scrolling](#)

Created with the Standard Edition of HelpNDoc: [Make Your PDFs More Secure with Encryption and Password Protection](#)

GetTextOverlay_ScrollingSpeed

TVideoGrabber.GetTextOverlay_ScrollingSpeed

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetTextOverlay_ScrollingSpeed](#)

Declaration

```
function GetTextOverlay_ScrollingSpeed(Index: LongInt): LongInt;
```

```
int __fastcall GetTextOverlay_ScrollingSpeed(int Index);
```

Description

retrieves the value set by [SetTextOverlay_ScrollingSpeed](#)

Created with the Standard Edition of HelpNDoc: [Full-featured Documentation generator](#)

GetTextOverlay_Shadow

TVideoGrabber.GetTextOverlay_Shadow

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetTextOverlay_Shadow](#)

Declaration

```
function GetTextOverlay_Shadow(Index: LongInt): Boolean;
```

```
bool __fastcall GetTextOverlay_Shadow(int Index);
```

Description

retrieves the value set by [SetTextOverlay_Shadow](#)

Created with the Standard Edition of HelpNDoc: [Create cross-platform Qt Help files](#)

GetTextOverlay_ShadowColor

TVideoGrabber.GetTextOverlay_ShadowColor

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetTextOverlay_ShadowColor](#)

Declaration

```
function GetTextOverlay_ShadowColor(Index: LongInt): TColor;
```

```
Graphics::TColor __fastcall GetTextOverlay_ShadowColor(int Index);
```

Description

retrieves the value set by [SetTextOverlay_ShadowColor](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly upgrade your WinHelp HLP help files to CHM with HelpNDoc](#)

GetTextOverlay_ShadowDirection

TVideoGrabber.GetTextOverlay_ShadowDirection

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetTextOverlay_ShadowDirection](#)

Declaration

```
function GetTextOverlay_ShadowDirection(Index: LongInt): TCardinalDirection;
```

```
TCardinalDirection __fastcall GetTextOverlay_ShadowDirection(int Index);
```

Description

retrieves the value set by [SetTextOverlay_ShadowDirection](#)

GetTextOverlay_String

TVideoGrabber.GetTextOverlay_String

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetTextOverlay_String](#)

Declaration

function GetTextOverlay_String(Index: LongInt): string;

wchar_t *__fastcall GetTextOverlay_String(int Index);

Description

retrieves the value set by [SetTextOverlay_String](#)

GetTextOverlay_TargetDisplay

TVideoGrabber.GetTextOverlay_TargetDisplay

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetTextOverlay_TargetDisplay](#)

Declaration

function GetTextOverlay_TargetDisplay(Index: LongInt): LongInt;

int __fastcall GetTextOverlay_TargetDisplay(int Index);

Description

retrieves the value set by [SetTextOverlay_TargetDisplay](#)

GetTextOverlay_Top

TVideoGrabber.GetTextOverlay_Top

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetTextOverlay_Top](#)

Declaration

function GetTextOverlay_Top(Index: LongInt): LongInt;

int __fastcall GetTextOverlay_Top(int Index);

Description

retrieves the value set by [SetTextOverlay_Top](#)

GetTextOverlay_Transparent

TVideoGrabber.GetTextOverlay_Transparent

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

retrieves the value set by [SetTextOverlay_Transparent](#)

Declaration

function GetTextOverlay_Transparent(Index: LongInt): Boolean;

bool __fastcall GetTextOverlay_Transparent(int Index);

Description

retrieves the value set by [SetTextOverlay_Transparent](#)

Created with the Standard Edition of HelpNDoc: [Generate EPub eBooks with ease](#)

GetTranslatedCoordinates

TVideoGrabber.GetTranslatedCoordinates

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Used to retrieve the translated coordinates of the mouse

Declaration

function GetTranslatedCoordinates (DisplayIndex: LongInt; NativeX: LongInt; NativeY: LongInt; **var** TranslatedX: LongInt; **var** TranslatedY: LongInt): Boolean;

__fastcall GetTranslatedCoordinates(**int** DisplayIndex, **int** NativeX, **int** NativeY, **int** &TranslatedX, **int** &TranslatedY);

Function GetTranslatedCoordinates (DisplayIndex as Long, NativeX as Long, NativeY as Long, ByRef TranslatedX as Long, ByRef TranslatedY as Long) as Boolean

Description

Invoke GetTranslatedCoordinates to retrieve the translated coordinates (the coordinates into the video window, left/top corner = 0,0) of the mouse events when [TranslateMouseCoordinates](#) is disabled.

This function can be invoked from the following events:

[OnMouseDown](#)

[OnMouseUp](#)

[OnMouseMove](#)

[OnMouseWheel](#)

OnDragOver

OnDragDrop

Created with the Standard Edition of HelpNDoc: [Free iPhone documentation generator](#)

GetTVChannelInfo

TVideoGrabber.GetTVChannelInfo

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns information about the current TV channel selected.

Declaration

function GetTVChannelInfo(Value: TTVChannelInfo): LongInt;

int __fastcall GetTVChannelInfo(TTVChannelInfo Value)

Function GetTVChannelInfo(param1 As TxTVChannelInfo) As Long

Description

Used to get information about the current TV channel selected.

Conditions:

- the [current video capture device](#) must have a TV tuner ([IsTVTunerAvailable](#) must return "true")
- the [VideoInput](#) property must be set to the index of "tuner" in the [VideoInputs](#) list.
- the preview or recording must be running.

Pass to this function the [TTVChannelInfo](#) value TTVChannelInfo desired, e.g.:

```
CurrentDefaultFreq = VideoGrabber.GetTVChannelInfo (tci_DefaultVideoFrequency)
```

See Also

[TTVChannelInfo](#) [TOnTVChannelScanStarted](#) [TOnTVChannelSelected](#) [TTunerInputType](#) [IsHorizontalSyncLocked](#) [IsTVAudioAvailable](#) [IsTVAutoTuneRunning](#) [IsTVTunerAvailable](#) [OnTVChannelScanCompleted](#) [OnTVChannelScanStarted](#) [OnTVChannelSelected](#) [TunerFrequency](#) [TunerMode](#) [TVChannel](#) [TVClearFrequencyOverrides](#) [TVCountryCode](#) [TVGetMinMaxChannels](#) [TVSetChannelFrequencyOverride](#) [TVStartAutoScan](#) [TVStopAutoScan](#) [TVTunerInputType](#) [TVUseFrequencyOverrides](#) [VCRHorizontalLocking](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Create High-Quality Documentation with a Help Authoring Tool](#)

GetVideoCodec

TVideoGrabber.GetVideoCodec

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Current video codec

Declaration

function GetVideoCodec: **string**;

System::wchar_t * __fastcall GetVideoCodec()

Function GetVideoCodec as String

Description

Returns the current video codec beeing used.

Created with the Standard Edition of HelpNDoc: [Easy CHM and documentation editor](#)

GetVideoCompressionSettings

TVideoGrabber.GetVideoCompressionSettings

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Retrieves the video compression settings of certain codecs.

Declaration

function GetVideoCompressionSettings(**out** DataRate, KeyFrameRate, PFramesPerKeyFrame, WindowSize: LongInt; **out** Quality: Double; **out** CanQuality, CanCrunch, CanKeyFrame, CanBFrame, CanWindow: Boolean): Boolean;

bool __fastcall GetVideoCompressionSettings(**int** &DataRate, **int** &KeyFrameRate, **int** &PFramesPerKeyFrame, **int** &WindowSize, **double** &Quality, **bool** &CanQuality, **bool** &CanCrunch, **bool** &CanKeyFrame, **bool** &CanBFrame, **bool** &CanWindow)

Function GetVideoCompressionSettings(DataRate, KeyFrameRate, PFramesPerKeyFrame, WindowSize, Quality, CanQuality, CanCrunch, CanKeyFrame, CanBFrame, CanWindow) As Boolean

Description

Retrieves the video compression settings supported by some codecs. Returns true if the current video compressor supports this interface.

See [SetVideoCompressionSettings](#) .

DataRate: retrieves the output data rate.

KeyFrameRate: The key-frame rate is the number of frames per key frame. For example, if the rate is 15, then a key frame occurs every 15 frames.

PFramesPerKeyFrame: P frames are used only in MPEG compression. E.g. let's say a key frame occurs once every 10 frames, and there are three P frames per key frame. The P frames will be spaced evenly between the key frames. The remaining six frames are bi-directional (B) frames.

WindowSize: retrieves the number of frames over which the compressor will maintain the average data rate. E.g. if a data rate of 100K/sec and a frame rate of 10 frames per second, if the window size is 1, then every frame will be 10K or less. If the window size is 5, then every five consecutive frames will average 10K per frame, but individual frames may exceed this size.

Quality: The quality is expressed as a value between 0.0 and 1.0, where 1.0 indicates the best quality and 0.0 indicates the worst quality. If the value is negative, the filter will use the default quality.

CanCrunch: the compressor can compress video to a specified data rate (see DataRate above).

CanKeyFrame: the compressor supports the KeyFrame property above.

CanBFrame: the compressor supports the PFramesPerKeyFrame property above.

CanWindow: the compressor supports WindowSize property above.

CanQuality: the compressor supports Quality property above.

See Also

[Recording methods and properties](#) [TCompressionType](#) [TOnVideoCompressionSettings](#) [AudioCompressor](#) [AudioCompressorIndex](#) [AudioCompressorName](#) [AudioCompressors](#) [AudioCompressorsCount](#) [CompressionMode](#) [CompressionType](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [OnVideoCompressionSettings](#) [RefreshDevicesAndCompressorsLists](#) [SaveCompressorSettingsToDataString](#) [SetVideoCompressionDefaults](#) [SetVideoCompressionSettings](#) [VideoCompression](#) [KeyFrameRate](#) [VideoCompression](#) [PFramesPerKeyFrame](#) [VideoCompression](#) [Quality](#) [VideoCompression](#) [WindowSize](#) [VideoCompressor](#) [VideoCompressorIndex](#) [VideoCompressorName](#) [VideoCompressors](#) [VideoCompressorsCount](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your CHM Help File Capabilities with HelpNDoc](#)

GetVideoControlMode

TVideoGrabber.GetVideoControlMode

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Retrieve the state of one of the [TVideoControl](#) settings

Declaration

function GetVideoControlMode (Mode: TVideoControl): Boolean;

bool __fastcall GetVideoControlMode(TVideoControl Mode);

Function GetVideoControlMode (mode as TVideoControl) As Boolean

Description

Used to retrieve the state of one of the [TVideoControl](#) settings, if available for the current video capture device (see [IsVideoControlModeAvailable](#)).

See Also

[TVideoControl](#) [IsVideoControlAvailable](#) [IsVideoControlModeAvailable](#) [SetVideoControlMode](#)
[SetVideoControlMode2](#) [VideoControlSettings](#)

Created with the Standard Edition of HelpNDoc: [Easily create CHM Help documents](#)

GetVideoHeightFromIndex

TVideoGrabber.GetVideoHeightFromIndex

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Retrieves the height of a video size in the [VideoSizes](#) list.

Declaration

function GetVideoHeightFromIndex(VideoSizeIndex: LongInt): LongInt;

int __fastcall GetVideoHeightFromIndex(**int** VideoSizeIndex)

Function GetVideoHeightFromIndex(VideoSizeIndex As Long) As Long

Description

Used to retrieve the Height of a video size in the [VideoSizes](#) list of the [current video capture device](#) .
 The VideoSizeIndex parameter is the index of the size in the [VideoSizes](#) list.
 Returns 0 is VideoSizeIndex is out of the 0..[VideoSizesCount](#) - 1 range.

See Also

[GetNearestVideoHeight](#) [GetNearestVideoSize](#) [GetNearestVideoWidth](#) [GetVideoSizeFromIndex](#)
[GetVideoWidthFromIndex](#) [UseNearestVideoSize](#) [VideoFormat](#) [VideoFormatIndex](#) [VideoSize](#)
[VideoSizeIndex](#) [VideoSizes](#) [VideoSizesCount](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Documentation Efficiency with a Help Authoring Tool](#)

GetVideoSizeFromIndex

TVideoGrabber.GetVideoSizeFromIndex

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Retrieves the width and height of a video size in the [VideoSizes](#) list.

Declaration

function GetVideoSizeFromIndex(VideoSizeIndex: LongInt; **out** VideoWidth: LongInt; **out** VideoHeight: LongInt): Boolean;

bool __fastcall GetVideoSizeFromIndex(**int** VideoSizeIndex, **int** &VideoWidth, **int** &VideoHeight)

Function GetVideoSizeFromIndex(VideoSizeIndex As Long, Video_Width, Video_Height) As Boolean

Description

Used to retrieve the width and height of a video size in the [VideoSizes](#) list of the [current video capture device](#) .
 The VideoSizeIndex parameter is the index of the size in the [VideoSizes](#) list.
 The width and height are returned by the VideoWidth and VideoHeight parameters.
 Returns false is VideoSizeIndex is out of the 0..[VideoSizesCount](#) - 1 range.

See Also

[GetNearestVideoHeight](#) [GetNearestVideoSize](#) [GetNearestVideoWidth](#) [GetVideoHeightFromIndex](#)
[GetVideoWidthFromIndex](#) [UseNearestVideoSize](#) [VideoFormat](#) [VideoFormatIndex](#) [VideoSize](#)
[VideoSizeIndex](#) [VideoSizes](#) [VideoSizesCount](#)

GetVideoWidthFromIndex

TVideoGrabber.GetVideoWidthFromIndex

[Prev](#)[Next](#)[TVideoGrabber](#) [Methods](#)

Retrieves the width of a video size in the [VideoSizes](#) list.

Declaration

function GetVideoWidthFromIndex(VideoSizeIndex: LongInt): LongInt;

int __fastcall GetVideoWidthFromIndex(**int** VideoSizeIndex)

Function GetVideoWidthFromIndex(VideoSizeIndex As Long) As Long

Description

Used to retrieve the width of a video size in the [VideoSizes](#) list of the [current video capture device](#) . The VideoSizeIndex parameter is the index of the size in the [VideoSizes](#) list. Returns 0 is VideoSizeIndex is out of the 0..[VideoSizesCount](#) - 1 range.

See Also

[GetNearestVideoHeight](#) [GetNearestVideoSize](#) [GetNearestVideoWidth](#) [GetVideoHeightFromIndex](#) [GetVideoSizeFromIndex](#) [UseNearestVideoSize](#) [VideoFormat](#) [VideoFormatIndex](#) [VideoSize](#) [VideoSizeIndex](#) [VideoSizes](#) [VideoSizesCount](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Convert Your Markdown Content with HelpNDoc](#)

GetVMR9ImageAdjustmentBounds

TVideoGrabber.GetVMR9ImageAdjustmentBounds

[Prev](#)[Next](#)[TVideoGrabber](#) [Methods](#)

Retrieves the bounds of the specified VMR9 image adjustment property.

Declaration

function GetVMR9ImageAdjustmentBounds(MainDisplay: Boolean; VMR9ControlSetting: TVMR9ImageAdjustment; **out** MinValue: LongInt; **out** MaxValue: LongInt; **out** StepSize: LongInt; **out** DefaultValue: LongInt; **out** CurrentValue: LongInt): Boolean;

bool __fastcall GetVMR9ImageAdjustmentBounds(**bool** MainDisplay, TVMR9ImageAdjustment VMR9ControlSetting, **int** &MinValue, **int** &MaxValue, **int** &StepSize, **int** &DefaultValue, **int** &CurrentValue)

Function GetVMR9ImageAdjustmentBounds(MainDisplay As Boolean, VMR9ControlSetting As TxVMR9ImageAdjustment, MinValue, MaxValue, StepSize, DefaultValue, CurrentValue) As Boolean

Description

Used to retrieve the bounds of the specified VMR9 image adjustment property (brightness, contrast, hue, saturation).

The VMR9ControlSetting parameter is a [TVMR9ImageAdjustment](#) type.

Returns all the information useful to setup a trackbar component (min value, max value, step size, default value, current position).

E.g.:

```

procedure TForm1.VideoGrabberGraphBuilt(Sender: TObject);
var
    MinValue, MaxValue, StepSize, DefaultValue, CurrentValue: LongInt;
begin
    BrightnessTrackbar.Enabled := VideoGrabber.GetVMR9ImageAdjustmentBounds (True, vmr9_Brightness);
    if Brightness.Enabled then begin
        Brightness.Min := MinValue;
        Brightness.Max := MaxValue;
        Brightness.Frequency := StepSize;
        Brightness.Position := CurrentValue;
    end;
end;

procedure TForm1.BrightnessTrackbarChange(Sender: TObject);
begin
    VideoGrabber.SetVMR9ImageAdjustmentValue (True, vmr9_Brightness, tbrVMR9Brightness.Position);
end;

```

See the MainDemo project for more sample code.

See Also

[IsVMR9ImageAdjustmentAvailable](#) [SetVMR9ImageAdjustmentValue](#) [TVMR9ImageAdjustment](#)

Created with the Standard Edition of HelpNDoc: [Write EPub books for the iPad](#)

GetVUMeterSetting

TVideoGrabber.GetVUMeterSetting

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Retrieves a current VUMeter setting

Declaration

function GetVUMeterSetting(ChannelIndex: LongWord; VUMeterSetting: TVUMeterSetting): LongWord;

unsigned __fastcall GetVUMeterSetting(**unsigned** ChannelIndex, TVUMeterSetting VUMeterSetting)

Function GetVUMeterSetting (ChannelIndex as Long, VUMeterSetting as TVUMeterSetting) as Long

Description

Used to retrieve a current VUMeter setting.

See [SetVUMeterSetting](#)

Created with the Standard Edition of HelpNDoc: [Make Your PDFs More Secure with Encryption and Password Protection](#)

GraphState

TVideoGrabber.GraphState

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

returns the current state of the graph

Declaration

function GraphState: TGraphState;

TGraphState __fastcall GetGraphState(void)

Function GraphState as TxGraphState

Description

Used to retrieve the current state of the preview, recording or playback graph.

The function returns a [TGraphState](#) value.

Created with the Standard Edition of HelpNDoc: [Free EBook and documentation generator](#)

IsAudioDeviceASoundCard

TVideoGrabber.IsAudioDeviceASoundCard

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Properties](#)

Returns true if the audio device specified is a sound card.

Declaration

function IsAudioDeviceASoundCard (DeviceIndex: LongInt): Boolean;

bool __fastcall IsAudioDeviceASoundCard(int DeviceIndex)

Function IsAudioDeviceASoundCard (DeviceIndex as Long) As Boolean

Description

Used to know if the [audio device](#) specified is a sound card.

Specify as DeviceIndex the index of the audio capture device in the [AudioDevices](#) list.

See Also

[TAudioFormat](#) [AssociateAudioAndVideoDevices](#) [AudioBalance](#) [AudioDevice](#) [AudioDeviceIndex](#) [AudioDeviceName](#) [AudioDeviceRendering](#) [AudioDevices](#) [AudioDevicesCount](#) [AudioFormat](#) [AudioFormats](#) [AudioInput](#) [AudioInputIndex](#) [AudioInputLevel](#) [AudioInputMono](#) [AudioInputs](#) [AudioInputsCount](#) [AudioSource](#) [IsAudioCrossbarAvailable](#) [IsAudioDeviceConnected](#) [IsAudioInputBalanceAvailable](#) [OnAudioBufferNegotiation](#) [OnAudioDeviceSelected](#) [SpeakerBalance](#) [SpeakerControl](#) [SpeakerVolume](#)

Created with the Standard Edition of HelpNDoc: [Free CHM Help documentation generator](#)

IsAudioDeviceConnected

TVideoGrabber.IsAudioDeviceConnected

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Used to know if the current audio capture device is connected

Declaration

function IsAudioDeviceConnected(DeviceIndex: LongInt): Boolean;

bool __fastcall IsAudioDeviceConnected(int DeviceIndex);

Function IsAudioDeviceConnected (DeviceIndex As Long) As Boolean

Description

Returns true if the specified audio capture device (in the [AudioDevices](#) list) is connected.

Returns false if the device has been disconnected.

See the [Audio capture devices](#) chapter for more information.

See Also

[TAudioFormat](#) [AssociateAudioAndVideoDevices](#) [AudioBalance](#) [AudioDevice](#) [AudioDeviceIndex](#)

[AudioDeviceName](#) [AudioDeviceRendering](#) [AudioDevices](#) [AudioDevicesCount](#) [AudioFormat](#) [AudioFormats](#) [AudioInput](#) [AudioInputIndex](#) [AudioInputLevel](#) [AudioInputMono](#) [AudioInputs](#) [AudioInputsCount](#) [AudioSource](#) [IsAudioCrossbarAvailable](#) [IsAudioDeviceASoundCard](#) [IsAudioInputBalanceAvailable](#) [OnAudioBufferNegotiation](#) [OnAudioDeviceSelected](#) [SpeakerBalance](#) [SpeakerControl](#) [SpeakerVolume](#)

Created with the Standard Edition of HelpNDoc: [Easily create Qt Help files](#)

IsAudioRendererConnected

TVideoGrabber.IsAudioRendererConnected

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns true when an audio renderer is connected

Declaration

function IsAudioRendererConnected(RendererIndex: LongInt): Boolean;

bool __fastcall IsAudioRendererConnected(int RendererIndex)

Function IsAudioRendererConnected (RendererIndex as Long) as Boolean

Description

Used to know if the audio renderer is connected.

Specify as parameter the index of the audio renderer in the [AudioRenderers](#) list.

See Also

[AudioRendererIndex](#)

Created with the Standard Edition of HelpNDoc: [Transform Your Word Document into a Professional eBook with HelpNDoc](#)

IsCameraControlSettingAvailable

TVideoGrabber.IsCameraControlSettingAvailable

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Availability of a given [TCameraControl](#) setting.

Declaration

function IsCameraControlSettingAvailable(Setting: TCameraControl): Boolean;

bool __fastcall IsCameraControlSettingAvailable(TCameraControl Setting)

Function IsCameraControlSettingAvailable(Setting As TxCameraControl) As Boolean

Description

Retrieves the availability of a given [TCameraControl](#) setting for the current video capture device.

See Also

[TCameraControl](#) [CameraControlAuto](#) [CameraControlDefault](#) [CameraControlMax](#) [CameraControlMin](#) [CameraControlSettings](#) [CameraControlStep](#) [GetCameraExposure](#) [GetCameraExposureAsString](#) [IsCameraControlAvailable](#) [SendCameraCommand](#) [SetCameraControl](#) [SetCameraExposure](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Support Your Windows Applications with HelpNDoc's CHM Generation](#)

IsDialogAvailable

TVideoGrabber.IsDialogAvailable

[Prev](#)[Next](#)[TVideoGrabber](#) [Methods](#)

Availability of a given [TDialog](#) .

Declaration

function IsDialogAvailable(Dialog: TDialog): Boolean;

bool __fastcall IsDialogAvailable(TDialog Dialog)

Function IsDialogAvailable(Dialog As TxDIALOG) As Boolean

Description

Retrieves the availability of a given [TDialog](#) for the current video capture device.

Created with the Standard Edition of HelpNDoc: [Maximize Your CHM Help File Capabilities with HelpNDoc](#)

IsDVDevice

TVideoGrabber.IsDVDevice

[Prev](#)[Next](#)[TVideoGrabber](#) [Methods](#)

Used to know if the specified video capture device is a DV device.

Declaration

function IsDVDevice(Index: LongInt): Boolean;

bool __fastcall IsDVDevice(int Index);

Function IsDVDevice(Index As Long) As Boolean

Description

Used to know if the video capture device specified by its index [VideoDevices](#) list is a DV capture device.

See Also

[WDM drivers](#) [AutoConnectRelatedPins](#) [CurrentFrameRate](#) [DeliveredFrames](#) [DroppedFrames](#) [FixFlickerOrBlackCapture](#) [IsVideoControlAvailable](#) [IsVideoCrossbarAvailable](#) [IsVideoDeviceConnected](#) [IsWDMVideoDriver](#) [OnNoVideoDevices](#) [OnVideoDeviceSelected](#) [RefreshDevicesAndCompressorsLists](#) [ResetVideoDeviceSettings](#) [ShowDialog](#) [VideoDevice](#) [VideoDeviceIndex](#) [VideoDeviceIndexFromId](#) [VideoDeviceName](#) [VideoDevices](#) [VideoDevicesCount](#) [VideoDevicesId](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Streamline Your CHM Help File Creation with HelpNDoc](#)

IsPlaylistActive

TVideoGrabber.IsPlaylistActive

[Prev](#)[Next](#)[TVideoGrabber](#) [Methods](#)

Returns the state of the playlist.

Declaration

function IsPlaylistActive: Boolean;

bool IsPlaylistActive(void);

Function IsPlaylistActive as Boolean

Description

Returns true if the playlist is active (playing).

See the "[Using the playlist](#)" chapter for more information about the playlist feature.

See Also

[TPlaylist](#) [Video formats](#) [GetPlaylist](#) [OnPlayerEndOfPlaylist](#) [Playlist](#) [PlaylistIndex](#)

Created with the Standard Edition of HelpNDoc: [Free PDF documentation generator](#)

IsURLResponding

TVideoGrabber.IsURLResponding

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns true if the URL is alive

Declaration

function IsURLResponding: Boolean;

bool __fastcall IsURLResponding(**void**);

Description

Used to verify if the URL specified to [IPCameraURL](#) or [VideoSource_FileOrURL](#) without starting to decode the live stream.

See Also

[IR Cut Filter of Axis cameras](#) [TDiscoveryCallbackStatus](#) [TOnONVIFDiscoveryCompletedNotification](#) [Video stream of IP camera with audio of PC microphone or other audio capture device](#) [ExtraDLLPath](#) [GetLastError](#) [IPCameraURL](#) [IsURLVideoStreamAvailable](#) [OnONVIFDiscoveryCompletedNotification](#) [ONVIF_GetStr](#) [ONVIF_SetStr](#) [ONVIFCancelDiscovery](#) [ONVIFDiscoverCameras](#) [IPRange](#) [ONVIFDiscoverCameras](#) [Multicast](#) [ONVIFEnumCamerasDiscovered](#) [OpenURLAsync](#) [OpenURLAsyncStatus](#) [SendIPCameraCommand](#) [SetDatasteadFilterDllName](#) [SetIPCameraSetting](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Create Encrypted, Password-Protected PDFs](#)

IsURLVideoStreamAvailable

TVideoGrabber.IsURLVideoStreamAvailable

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

used to determine if the URL source really outputs video samples

Declaration

function IsURLVideoStreamAvailable (Timeout_Ms: LongInt): TTriState;

TTriState __fastcall IsURLVideoStreamAvailable (int Timeout_Ms);

Description

Used to verify if the URL specified to [IPCameraURL](#) or [VideoSource_FileOrURL](#) outputs video samples without starting the GUI.

This function can be invoked from a background thread.

The function waits for the first video sample to be received, until the Timeout_Ms duration (expressed in milliseconds) expires.

Return values:

ts_Undefined: the connection failed

ts_False: the connection succeeded but the source does not output video samples

ts_True: the connection succeeded and at least one video sample has been received

See Also

[IR Cut Filter of Axis cameras](#) [TDiscoveryCallbackStatus](#) [TOnONVIFDiscoveryCompletedNotification](#) [Video stream of IP camera with audio of PC microphone or other audio capture device](#) [ExtraDLLPath](#) [GetLastError](#) [IPCameraURL](#) [IsURLResponding](#) [OnONVIFDiscoveryCompletedNotification](#) [ONVIF_GetStr](#) [ONVIF_SetStr](#) [ONVIFCancelDiscovery](#) [ONVIFDiscoverCameras](#) [IPRange](#) [ONVIFDiscoverCameras_Multicast](#) [ONVIFEnumCamerasDiscovered](#) [OpenURLAsync](#) [OpenURLAsyncStatus](#) [SendIPCameraCommand](#) [SetDatasteadFilterDllName](#) [SetIPCameraSetting](#)

Created with the Standard Edition of HelpNDoc: [Revolutionize Your Documentation Output with HelpNDoc's Stunning User Interface](#)

IsVideoControlModeAvailable

TVideoGrabber.IsVideoControlModeAvailable

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Availability of a given [TVideoControl](#) mode.

Declaration

function IsVideoControlModeAvailable(Mode: TVideoControl): Boolean;

bool __fastcall IsVideoControlModeAvailable(TVideoControl Mode)

Function IsVideoControlModeAvailable(Mode As TxVideoControl) As Boolean

Description

Retrieves the availability of a given [TVideoControl](#) mode for the current video capture device.

See Also

[TVideoControl](#) [GetVideoControlMode](#) [IsVideoControlAvailable](#) [SetVideoControlMode](#) [SetVideoControlMode2](#) [VideoControlSettings](#)

Created with the Standard Edition of HelpNDoc: [Single source CHM, PDF, DOC and HTML Help creation](#)

IsVideoDeviceConnected

TVideoGrabber.IsVideoDeviceConnected

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns true if the specified video capture device is connected.

Declaration

function IsVideoDeviceConnected(DeviceIndex: LongInt): Boolean;

bool __fastcall IsVideoDeviceConnected(int DeviceIndex);

Function IsVideoDeviceConnected (DeviceIndex As Long) As Boolean

Description

Returns true if the video capture device specified by its index (in the [VideoDevices](#) list) is connected. Returns false if the device has been disconnected.

See Also

[WDM drivers](#) [AutoConnectRelatedPins](#) [CurrentFrameRate](#) [DeliveredFrames](#) [DroppedFrames](#) [FixFlickerOrBlackCapture](#) [IsDVDevice](#) [IsVideoControlAvailable](#) [IsVideoCrossbarAvailable](#) [IsWDMVideoDriver](#) [OnNoVideoDevices](#) [OnVideoDeviceSelected](#) [RefreshDevicesAndCompressorsLists](#) [ResetVideoDeviceSettings](#) [ShowDialog](#) [VideoDevice](#) [VideoDeviceIndex](#) [VideoDeviceIndexFromId](#)

[VideoDeviceName](#) [VideoDevices](#) [VideoDevicesCount](#) [VideoDevicesId](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Don't Let Unauthorized Users View Your PDFs: Learn How to Set Passwords](#)

IsVideoQualitySettingAvailable

TVideoGrabber.IsVideoQualitySettingAvailable

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Availability of a given [TVideoQuality](#) setting.

Declaration

function IsVideoQualitySettingAvailable(Setting: TVideoQuality): Boolean;

bool __fastcall IsVideoQualitySettingAvailable(TVideoQuality Setting)

Function IsVideoQualitySettingAvailable(Setting As TVideoQuality) As Boolean

Description

Retrieves the availability of a given [TVideoQuality](#) setting for the current video capture device.

See Also

[IsVideoQualityAvailable](#) [SetVideoQuality](#) [VideoQualityAuto](#) [VideoQualityDefault](#) [VideoQualityMax](#) [VideoQualityMin](#) [VideoQualitySettings](#) [VideoQualityStep](#) [VideoRendererPriority](#) [TVideoQuality](#)

Created with the Standard Edition of HelpNDoc: [Add an Extra Layer of Security to Your PDFs with Encryption](#)

IsVideoSignalDetected

TVideoGrabber.IsVideoSignalDetected

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Detects if a video signal is present.

Declaration

function IsVideoSignalDetected (DetectConnexantBlueScreen: **Boolean**; DetectCustomRGB: **Boolean**; CustomR, CustomG, CustomB: LongInt; UseAsMaxValues: Boolean): **Boolean**;

bool __fastcall IsVideoSignalDetected (**bool** DetectConnexantBlueScreen, **bool** DetectCustomRGB, **int** CustomR, **int** CustomG, **int** CustomB, **bool** UseAsMaxValues)

Function IsVideoSignalDetected (DetectConnexantBlueScreen as Boolean; DetectCustomRGB as Boolean, CustomR as Long, CustomG as Long, CustomB as Long, UseAsMaxValues as Boolean) as Boolean

Description

Used to detect if a video signal is present.

It returns:

- true if a video signal is detected,
- false if a blue screen or dark screen is detected (see below).

This function can act in 2 ways, depending of the DetectConnexantBlueScreen and DetectCustomRGB parameters (that can be combined):

DetectConnexantBlueScreen is enabled:

The "classical blue screen" of video capture boards including Connexant video chips, that corresponds to "no video signal" is detected (in this case the function will return false)

DetectCustomRGB is enabled:

The CustomR, CustomG and CustomB parameters specify the values of the colors used to detect if a video signal is present, according to the **UseAsMaxValues** parameter:

UseAsMaxValues is disabled:

The video signal will be reported as "not detected" if all the pixels tested on the video frames have the exact RGB values specified in CustomR, CustomG and CustomB.

UseAsMaxValues is enabled:

The video signal will be reported as "not detected" if all the pixels tested on the video frames have a lower value than the CustomR, CustomG and CustomB values specified.

E.g:

Connexant (BT8x8) cards:

IsVideoSignalDetected (true, false, 0, 0, 0, false) returns false when the blue screen of **Conexant BT8x8** based cards is detected.

Techwell cards:

IsVideoSignalDetected (false, true, 16, 16, 16, false) returns false when the black screen of **Techwell** cards is detected.

Conexant as well as Techwell cards

IsVideoSignalDetected (true, true, 16, 16, 16, false) returns false when either the Conexant's blue screen or the **Techwell's** black screen is detected.

True black video

IsVideoSignalDetected (false, true, 0, 0, 0, false) returns false on a really black video (all RGB values = 0)

Detection of a dark ambient light

IsVideoSignalDetected (false, true, 30, 30, 30, true) returns false on a dark or black video (all RGB values below 30)

Created with the Standard Edition of HelpNDoc: [Elevate Your Help Documentation with a Help Authoring Tool](#)

IsVMR9ImageAdjustmentAvailable

TVideoGrabber.IsVMR9ImageAdjustmentAvailable

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns the availability of a VMR9 image adjustment setting.

Declaration

function IsVMR9ImageAdjustmentAvailable(MainDisplay: Boolean): Boolean;

bool __fastcall IsVMR9ImageAdjustmentAvailable(**bool** MainDisplay)

Function IsVMR9ImageAdjustmentAvailable(MainDisplay As Boolean) As Boolean

Description

Used to retrieve the availability of a VMR9 image adjustment property (brightness, contrast, hue, saturation).

See Also

[GetVMR9ImageAdjustmentBounds](#) [SetVMR9ImageAdjustmentValue](#) [TVMR9ImageAdjustment](#)

Created with the Standard Edition of HelpNDoc: [Easy to use tool to create HTML Help files and Help web sites](#)

LoadCompressorSettingsFromDataString

TVideoGrabber.LoadCompressorSettingsFromDataString

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Reloads the compressor settings from a data string

Declaration

function LoadCompressorSettingsFromDataString (IsVideoCompressor: boolean; CompressorIndex: LongInt; DataString: string): **Boolean**;

bool __fastcall LoadCompressorSettingsFromDataString (**bool** IsVideoCompressor, **int** CompressorIndex: LongInt, wchar_t *DataString);

function LoadCompressorSettingsFromDataString (IsVideoCompressor as Boolean, CompressorIndex as Long, FileName as String) As String

Description

Used to reload the compressor settings from a data string.

The compressor settings depend on the current [VideoCompressor](#) and [RecordingMethod](#), so be sure to first select the RecordingMethod() and VideoCompressor before saving its settings.

The **IsVideoCompressor** parameter must be set to **true** for a video compressor, or **false** for an audio compressor.

The Compressor parameter index must be the [VideoCompressor](#) or [AudioCompressor](#) index of the related compressor in the [VideoCompressors](#) or [AudioCompressors](#) lists.

See [Saving and restoring compressor settings programmatically](#) for sample code.

Created with the Standard Edition of HelpNDoc: [Make Documentation a Breeze with a Help Authoring Tool](#)

LoadCompressorSettingsFromTextFile

TVideoGrabber.LoadCompressorSettingsFromTextFile

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Reloads the compressor settings from a text file.

Declaration

function LoadCompressorSettingsFromTextFile (IsVideoCompressor: boolean; CompressorIndex: LongInt; FileName: **string**): Boolean;

bool __fastcall LoadCompressorSettingsFromTextFile (**bool** IsVideoCompressor, **int** CompressorIndex: LongInt, wchar_t *FileName);

function LoadCompressorSettingsFromTextFile (IsVideoCompressor as Boolean, CompressorIndex as Long, FileName as String) As String

Description

Used to reload the compressor settings from a text file.

The compressor settings depend on the current [VideoCompressor](#) and [RecordingMethod](#), so be sure to first select the RecordingMethod() and VideoCompressor before saving its settings.

The **IsVideoCompressor** parameter must be set to **true** for a video compressor, or **false** for an audio compressor.

The Compressor parameter index must be the [VideoCompressor](#) or [AudioCompressor](#) index of the related compressor in the [VideoCompressors](#) or [AudioCompressors](#) lists.

See [Saving and restoring compressor settings programmatically](#) for sample code.

Created with the Standard Edition of HelpNDoc: [Create HTML Help, DOC, PDF and print manuals from 1 single source](#)

MixAudioSamples

TVideoGrabber.MixAudioSamples

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Mixes the audio samples passed as parameter

Declaration

function MixAudioSamples (pSampleBuffer: Pointer; SampleBufferSize: LongInt; SampleDataLength: LongInt; SampleFormatType: TFormatType; pFormat: Pointer; SampleStartTime: LargeInteger; SampleStopTime: LargeInteger): Boolean;

bool __fastcall MixAudioSamples(void *pSampleBuffer, int SampleBufferSize, int SampleDataLength, TFormatType SampleFormatType, void *pFormat, __int64 SampleStartTime, __int64 SampleStopTime);

function __fastcall MixAudioSamples (pSampleBuffer as Long, SampleBufferSize as Long, SampleDataLength as Long, SampleFormatType as TxFormatType, pFormat as Long, SampleStartTime as Double, SampleStopTime as Double) as Bool

Description

Used to mix the audio samples coming out from another TVideoGrabber component.

See Also

[MixAudioSamples_CurrentSourceLevel](#) [MixAudioSamples_ExternalSourceLevel](#)
[Mixer_SetupPIPFromSource](#)

Created with the Standard Edition of HelpNDoc: [Generate Kindle eBooks with ease](#)

Mixer_Activation

TVideoGrabber.Mixer_Activation

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Enables / disables temporarily a mixing

Declaration

function Mixer_Activation(Id: LongInt; Activate: Boolean): Boolean;

bool __fastcall Mixer_Activation(int Id, bool Activate)

Function Mixer_Activation (Id as Long, Activate as Boolean) as Boolean

Description

Used to enable / disable the video mixing of a source component in the mixer component.
 Pass as parameter the mixer Id returned by [Mixer_AddToMixer](#) when associating the source component to the mixer.

E.g. VideoGrabber1 previews a video capture device, and VideoGrabber2 "duplicates" the video frames

from VideoGrabber1, like it was a 2nd component capturing the same device:

- to start the preview of the capture device:

```
VideoGrabber1.VideoSource = vs_VideoCaptureDevice
VideoGrabber1.StartPreview
```

- to start the 2nd component that will preview the frames of the 1st component, in mixer mode:

```
VideoGrabber2.VideoSource = vs_Mixer
MixerId = VideoGrabber2.Mixer_AddToMixer (VideoGrabber1.UniqueID, 0, 0, 0, 0, 0, true, true)
VideoGrabber2.StartPreview
```

- to disable temporarily the mixing:

```
VideoGrabber2.Mixer_Activation (MixerId, false)
```

- to resume temporarily the mixing:

```
VideoGrabber2.Mixer_Activation (MixerId, true)
```

See Also

[Mixer_AddToMixer](#) [Mixer_MosaicColumns](#) [Mixer_MosaicLines](#) [Mixer_RemoveFromMixer](#)
[Mixer_SetOverlayRoundedCorner](#) [Mixer_SetupPIPFromSource](#)

Created with the Standard Edition of HelpNDoc: [5 Reasons Why a Help Authoring Tool is Better than Microsoft Word for Documentation](#)

Mixer_AddToMixer

TVideoGrabber.Mixer_AddToMixer

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Used to associate 2 TVideoGrabber components in [Mixer](#) mode

Declaration

function Mixer_AddToMixer (SourceUniqueID: LongInt; SourceVideoInput: LongInt; MosaicLine: LongInt; MosaicColumn: LongInt; AlternatedGroup: LongInt; AlternatedTimeIntervalInMs: LongInt; ReplacePreviouslyAdded: Boolean; EraseBackground: Boolean): LongInt;

int __fastcall Mixer_AddToMixer(**int** SourceUniqueID, **int** SourceVideoInput, **int** MosaicLine, **int** MosaicColumn, **int** AlternatedGroup, **int** AlternatedTimeIntervalInMs, **bool** ReplacePreviouslyAdded, **bool** EraseBackground);

function Mixer_AddToMixer (SourceUniqueID as Long, SourceVideoInput as Long, MosaicLine as Long, MosaicColumn as Long, AlternatedGroup as Long, AlternatedTimeIntervalInMs as Long, ReplacePreviouslyAdded as Bool EraseBackground as Bool) as long

Description

This function is invoked **on the [mixer](#) component** to associate a source TVideoGrabber component in [Mixer](#) mode.

(the mixer component must have been set in this mode by setting [VideoSource](#) = vs_Mixer).

You will find sample code of this function in the "[How to mix several video sources into one a single one](#)" chapter.

Parameters:

SourceUniqueID:

UniqueID of the source component

SourceVideoInput:

not used yet, set it to 0

MosaicLine:

Line where the source video must be placed when displaying several sources on a mosaic layout (set to 0 if the mosaic layout is not used).

MosaicColumn:

Column where the source video must be placed when displaying several sources on a mosaic layout (set to 0 if the mosaic layout is not used).

AlternatedGroup:

Group in which the source video will be displayed alternatively.

You can specify any number.

The rule is: all sources associated to this mixer with this group number will be displayed at the same time, and never displayed at the same time than other sources associated to this mixer with other group numbers.

AlternatedTimeIntervalInMs:

Duration of display of the sources having this group number in the mixer.

The duration is expressed in milliseconds.

Remark: if several sources are associated to a mixer with the same group number but not the same duration, only one of these durations will be taken in account when this group will be displayed.

Therefore to prevent confusion, each sources of a given group should logically be associated to a mixer with the same duration.

ReplacePreviouslyAdded:

Specifies if invoking Mixer_AddToMixer on a given video source replaces any previous Mixer_AddToMixer setting already set. **Recommended value: TRUE**

EraseBackground:

specifies if the background must be erased on the mixer when Mixer_AddToMixer is invoked while the mixer is running. **Recommended value: TRUE**

RETURN VALUE:

The return value of this function is an identifier that may be used to remove the current Mixer_AddToMixer setting by invoking [Mixer_RemoveFromMixer](#).

See Also

[Mixer_Activation](#) [Mixer_MosaicColumns](#) [Mixer_MosaicLines](#) [Mixer_RemoveFromMixer](#)
[Mixer_SetOverlayRoundedCorner](#) [Mixer_SetupPIPFromSource](#)

Created with the Standard Edition of HelpNDoc: [Single source CHM, PDF, DOC and HTML Help creation](#)

Mixer_RemoveFromMixer

TVideoGrabber.Mixer_RemoveFromMixer

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Used to remove the association of 2 TVideoGrabber components in [Mixer](#) mode

Declaration

function Mixer_RemoveFromMixer (Id: LongInt): Boolean;

bool __fastcall Mixer_RemoveFromMixer(int Id);

function Mixer_RemoveFromMixer (Id as Long) as Bool

Description

Invoke `Mixer_RemoveFromMixer` to remove an between a source component and a Mixer component previously established by invoking [Mixer_AddToMixer](#).

Pass as parameter the identifier that was returned by the [Mixer_AddToMixer](#) function when it has been invoked.

See Also

[Mixer_Activation](#) [Mixer_AddToMixer](#) [Mixer_MosaicColumns](#) [Mixer_MosaicLines](#)
[Mixer_SetOverlayRoundedCorner](#) [Mixer_SetupPIPFromSource](#)

Created with the Standard Edition of HelpNDoc: [Free Qt Help documentation generator](#)

Mixer_SetOverlayRoundedCorner

TVideoGrabber.Mixer_SetOverlayRoundedCorner

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Rounds the corners of an image overlay

Declaration

function Mixer_SetOverlayRoundedCorner (Id: LongInt; CornerRadiusWidth: LongInt; CornerRadiusHeight: LongInt): Boolean;

bool __fastcall Mixer_SetOverlayRoundedCorner(**int** Id, **int** CornerRadiusWidth, **int** CornerRadiusHeight);

Description

Used to make an image overlay appear in the mixer with rounded corners or elipsoïdal, depending on the values specified

See Also

[Mixer_Activation](#) [Mixer_AddToMixer](#) [Mixer_MosaicColumns](#) [Mixer_MosaicLines](#) [Mixer_RemoveFromMixer](#)
[Mixer_SetupPIPFromSource](#)

Created with the Standard Edition of HelpNDoc: [Easily create PDF Help documents](#)

Mixer_SetupPIPFromSource

TVideoGrabber.Mixer_SetupPIPFromSource

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Used to make a PIP (Picture In Picture) between 2 TVideoGrabber components

Declaration

function Mixer_SetupPIPFromSource (SourceUniqueID: LongInt; Source_Left: LongInt; Source_Top: LongInt; Source_Width: LongInt; Source_Height: LongInt; ActivatePIP: Boolean; PIP_Left: LongInt; PIP_Top: LongInt; PIP_Width: LongInt; PIP_Height: LongInt; MoveToTop: Boolean): LongInt;

int __fastcall Mixer_SetupPIPFromSource(**int** SourceUniqueID, **int** Source_Left, **int** Source_Top, **int** Source_Width, **int** Source_Height, **bool** ActivatePIP, **int** PIP_Left, **int** PIP_Top, **int** PIP_Width, **int** PIP_Height, **bool** MoveToTop);

function SetPIPFromSource (SourceUniqueID as Long, Source_Left as Long, Source_Top as Long, Source_Width as Long, Source_Height as Long, ActivatePIP as Bool, PIP_Left as Long, PIP_Top as Long, PIP_Width as Long, PIP_Height as Long, MoveToTop as Bool) as Long

Description

This function setups a TVideoGrabber component to receive the whole video frames (or a cropped part of these video frames) from another TVideoGrabber component, and to display it in "PIP" mode (Picture In Picture), at any location or size in the destination video frame.

Parameters:

- the "Source_..." parameters specify a "rectangle part" of the source video frame that will be displayed in the destination video frame. Specifying 0, 0, 0, 0 means you send the whole source video frame.
- the "PIP_..." parameters specify a "rectangle part" of the destination video frame where the source will be displayed in "PIP" mode.

SourceUniqueID: LongInt

specifies the UniqueID property of the TVideoGrabber component that must send the video frames

Source_Left: LongInt

Left location in the source video frame

Source_Top: LongInt

Top location in the source video frame

Source_Width: LongInt

Width of the source rectangle (set 0 to let TVideoGrabber use automatically the full width of the source video frame)

Source_Height: LongInt

Height of the source rectangle (set 0 to let TVideoGrabber use automatically the full width of the source video frame)

ActivatePIP: Boolean

Specifies if the PIP display must be activated or not

PIP_Left: LongInt

Left location of the PIP on the destination video frame

PIP_Top: LongInt

Top location of the PIP on the destination video frame

PIP_Width: LongInt

Width of the PIP on the destination video frame

PIP_Height: LongInt

Height of the PIP on the destination video frame

MoveToTop: Boolean

Must be set to true if several PIP are partially overlapped and this one should appear above the others

RETURN VALUE: LongInt

Identifier of the mixer relation, this value is required e.g. by [Mixer_Activation](#) to activate/deactivate this PIP on the fly.

See Also

[MixAudioSamples](#) [MixAudioSamples_CurrentSourceLevel](#) [MixAudioSamples_ExternalSourceLevel](#)
[Mixer_Activation](#) [Mixer_AddToMixer](#) [Mixer_MosaicColumns](#) [Mixer_MosaicLines](#) [Mixer_RemoveFromMixer](#)
[Mixer_SetOverlayRoundedCorner](#)

Created with the Standard Edition of HelpNDoc: [Experience a User-Friendly Interface with HelpNDoc's Documentation Tool](#)

Monitor_Primary_Index

TVideoGrabber.Monitor_Primary_Index

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

returns the index of the primary monitor

Declaration

function Monitor_Primary_Index: LongInt;

int __fastcall Monitor_Primary_Index();

function Monitor_Primary_Index() As Long

Description

Used to retrieve the index of the primary monitor, in the 0..n-1 range

See Also

[Dual display](#) [TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display](#) [Active](#) [Display](#) [AlphaBlendEnabled](#) [Display](#) [AlphaBlendValue](#) [Display](#) [AutoSize](#) [Display](#) [Embedded](#) [Display](#) [FullScreen](#) [Display](#) [Height](#) [Display](#) [Left](#) [Display](#) [Monitor](#) [Display](#) [MouseMovesWindow](#) [Display](#) [PanScanRatio](#) [Display](#) [SetLocation](#) [Display](#) [StayOnTop](#) [Display](#) [Top](#) [Display](#) [TransparentColorEnabled](#) [Display](#) [TransparentColorValue](#) [Display](#) [VideoHeight](#) [Display](#) [VideoPortEnabled](#) [Display](#) [VideoWidth](#) [Display](#) [VideoWindowHandle](#) [Display](#) [Width](#) [DualDisplay](#) [Active](#) [DualDisplay](#) [AlphaBlendEnabled](#) [DualDisplay](#) [AlphaBlendValue](#) [DualDisplay](#) [AutoSize](#) [DualDisplay](#) [Embedded](#) [DualDisplay](#) [FullScreen](#) [DualDisplay](#) [Height](#) [DualDisplay](#) [Left](#) [DualDisplay](#) [Monitor](#) [DualDisplay](#) [MouseMovesWindow](#) [DualDisplay](#) [PanScanRatio](#) [DualDisplay](#) [SetLocation](#) [DualDisplay](#) [StayOnTop](#) [DualDisplay](#) [Top](#) [DualDisplay](#) [TransparentColorEnabled](#) [DualDisplay](#) [TransparentColorValue](#) [DualDisplay](#) [VideoHeight](#) [DualDisplay](#) [VideoPortEnabled](#) [DualDisplay](#) [VideoWidth](#) [DualDisplay](#) [VideoWindowHandle](#) [DualDisplay](#) [Visible](#) [DualDisplay](#) [Width](#) [IsVideoPortAvailable](#) [MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#) [SetParentWindow](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Transform Your CHM Help File Creation Process with HelpNDoc](#)

MonitorBounds**TVideoGrabber.MonitorBounds**
[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns the bounds of the specified video screen, in pixels.

Declaration

function MonitorBounds(MonitorNumber: LongInt; **var** Bounds: TRect): Boolean;

bool __fastcall MonitorBounds(**int** MonitorNumber, Windows::TRect &Bounds);

Function MonitorBounds(MonitorNumber As Long, LeftBound, TopBound, RightBound, BottomBound) As Boolean

Description

Returns the bounds of the specified video screen, in pixels.

Note: the MonitorNumber parameter is in the [0 .. [MonitorsCount](#) - 1] range.

Useful on platforms that have more than one video screen.

See the [Dual display](#) chapter for more information.

See Also

[Dual display](#) [TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display](#) [Active](#) [Display](#) [AlphaBlendEnabled](#) [Display](#) [AlphaBlendValue](#) [Display](#) [AutoSize](#) [Display](#) [Embedded](#) [Display](#) [FullScreen](#) [Display](#) [Height](#) [Display](#) [Left](#) [Display](#) [Monitor](#) [Display](#) [MouseMovesWindow](#) [Display](#) [PanScanRatio](#) [Display](#) [SetLocation](#) [Display](#) [StayOnTop](#) [Display](#) [Top](#) [Display](#) [TransparentColorEnabled](#) [Display](#) [TransparentColorValue](#) [Display](#) [VideoHeight](#) [Display](#) [VideoPortEnabled](#) [Display](#) [VideoWidth](#) [Display](#) [VideoWindowHandle](#) [Display](#) [Width](#) [DualDisplay](#) [Active](#) [DualDisplay](#) [AlphaBlendEnabled](#) [DualDisplay](#) [AlphaBlendValue](#)

[DualDisplay_AutoSize](#)
[DualDisplay_Embedded](#)
[DualDisplay_FullScreen](#)
[DualDisplay_Height](#)
[DualDisplay_Left](#)
[DualDisplay_Monitor](#)
[DualDisplay_MouseMovesWindow](#)
[DualDisplay_PanScanRatio](#)
[DualDisplay_SetLocation](#)
[DualDisplay_StayOnTop](#)
[DualDisplay_Top](#)
[DualDisplay_TransparentColorEnabled](#)
[DualDisplay_TransparentColorValue](#)
[DualDisplay_VideoHeight](#)
[DualDisplay_VideoPortEnabled](#)
[DualDisplay_VideoWidth](#)
[DualDisplay_VideoWindowHandle](#)
[DualDisplay_Visible](#)
[DualDisplay_Width](#)
[IsVideoPortAvailable](#)
[Monitor_Primary_Index](#)
[MonitorsCount](#)
[OnLeavingFullScreen](#)
[SetParentWindow](#)
[SetWindowTransparency](#)
[VideoDoubleBuffered](#)
[VideoHeight](#)
[PreferredAspectRatio](#)
[VideoRendererExternal](#)
[VideoRendererExternalIndex](#)
[VideoRendererPriority](#)
[VideoVisibleWhenStopped](#)
[VideoWidth_PreferedAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Easy to use tool to create HTML Help files and Help web sites](#)

MonitorsCount

TVideoGrabber.MonitorsCount

[Prev](#)
[Next](#)

[TVideoGrabber](#)
[Methods](#)

Number of monitors available on the current platform.

Declaration

function MonitorsCount: LongInt;

int __fastcall MonitorsCount(void);

Function MonitorsCount() As Long

Description

Returns the number of monitors available on the current platform.

Useful on platforms that have more than one video screen.

See the [Dual display](#) chapter for more information.

See Also

[Dual display](#)
[TVideoRenderer_AdjustPixelAspectRatio](#)
[Display_Active](#)
[Display_AlphaBlendEnabled](#)
[Display_AlphaBlendValue](#)
[Display_AutoSize](#)
[Display_Embedded](#)
[Display_FullScreen](#)
[Display_Height](#)
[Display_Left](#)
[Display_Monitor](#)
[Display_MouseMovesWindow](#)
[Display_PanScanRatio](#)
[Display_SetLocation](#)
[Display_StayOnTop](#)
[Display_Top](#)
[Display_TransparentColorEnabled](#)
[Display_TransparentColorValue](#)
[Display_VideoHeight](#)
[Display_VideoPortEnabled](#)
[Display_VideoWidth](#)
[Display_VideoWindowHandle](#)
[Display_Width](#)
[DualDisplay_Active](#)
[DualDisplay_AlphaBlendEnabled](#)
[DualDisplay_AlphaBlendValue](#)
[DualDisplay_AutoSize](#)
[DualDisplay_Embedded](#)
[DualDisplay_FullScreen](#)
[DualDisplay_Height](#)
[DualDisplay_Left](#)
[DualDisplay_Monitor](#)
[DualDisplay_MouseMovesWindow](#)
[DualDisplay_PanScanRatio](#)
[DualDisplay_SetLocation](#)
[DualDisplay_StayOnTop](#)
[DualDisplay_Top](#)
[DualDisplay_TransparentColorEnabled](#)
[DualDisplay_TransparentColorValue](#)
[DualDisplay_VideoHeight](#)
[DualDisplay_VideoPortEnabled](#)
[DualDisplay_VideoWidth](#)
[DualDisplay_VideoWindowHandle](#)
[DualDisplay_Visible](#)
[DualDisplay_Width](#)
[IsVideoPortAvailable](#)
[Monitor_Primary_Index](#)
[MonitorBounds](#)
[OnLeavingFullScreen](#)
[SetParentWindow](#)
[SetWindowTransparency](#)
[VideoDoubleBuffered](#)
[VideoHeight](#)
[PreferredAspectRatio](#)
[VideoRendererExternal](#)
[VideoRendererExternalIndex](#)
[VideoRendererPriority](#)
[VideoVisibleWhenStopped](#)
[VideoWidth_PreferedAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Benefits of a Help Authoring Tool](#)

MotionDetector_CellColorIntensity

TVideoGrabber.MotionDetector_CellColorIntensity

[Prev](#)
[Next](#)

[TVideoGrabber](#)
[Methods](#)

Retrieves the color intensity of a cell in the grid.

Declaration

function MotionDetector_CellColorIntensity (RGBSelector: TRGBSelector; x, y: LongInt): LongInt;

int __fastcall MotionDetector_CellColorIntensity(TRGBSelector RGBSelector, **int** x, **int** y);

function MotionDetector_CellColorIntensity (RGBSelector as TRGBSelector, x as Long, y as Long) as Long

Description

Retrieves the current average color intensity of a cell in the grid.
The [RGBSelector](#) parameter specifies the color to retrieve (R, G or B).

Created with the Standard Edition of HelpNDoc: [Maximize Your Productivity with HelpNDoc's CHM Help File Creation Features](#)

MotionDetector_CellMotionRatio

TVideoGrabber.MotionDetector_CellMotionRatio

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Retrieves the motion ratio of a cell on the grid.

Declaration

function MotionDetector_CellMotionRatio(x, y: LongInt): Double;

double __fastcall MotionDetector_CellMotionRatio(**int** x, **int** y)

Function MotionDetector_CellMotionRatio(x As Long, y As Long) As Double

Description

Used to retrieve the motion ratio (cell's moved pixels count / cell's total pixels count) of the cell specified by its x and y coordinates. The return value will be in the 0...1 range.
The x and y coordinates are zero-based index, from 0 to [MotionDetector_GridXCount](#) -1 and 0 to [MotionDetector_GridYCount](#) -1.

See Also

[Color / Greyscale Grid structure / grid sensitivity](#) [Motion ratio](#) [Recording only when motion is detected](#)
[Video noise](#) [TOnMotionDetected](#) [TOnMotionNotDetected](#) [MotionDetector_CompareBlue](#)
[MotionDetector_CompareGreen](#) [MotionDetector_CompareRed](#) [MotionDetector_Enabled](#)
[MotionDetector_EnumGridDialogControls](#) [MotionDetector_Get2DTextGrid](#) [MotionDetector_Get2DTextMotion](#)
[MotionDetector_GetCellLocation](#) [MotionDetector_GetCellSensitivity](#) [MotionDetector_GetCellSize](#)
[MotionDetector_GloballyIncOrDecSensitivity](#) [MotionDetector_GlobalMotionRatio](#) [MotionDetector_GreyScale](#)
[MotionDetector_Grid](#) [MotionDetector_GridXCount](#) [MotionDetector_GridYCount](#) [MotionDetector_IsGridValid](#)
[MotionDetector_MaxDetectionsPerSecond](#) [MotionDetector_ReduceCPULoad](#)
[MotionDetector_ReduceVideoNoise](#) [MotionDetector_Reset](#) [MotionDetector_ResetGlobalSensitivity](#)
[MotionDetector_SetCellSensitivity](#) [MotionDetector_SetGridSize](#) [MotionDetector_ShowGridDialog](#)
[MotionDetector_Triggered](#) [MotionDetector_UseThisReferenceSample](#) [OnBacktimedFramesCountReached](#)
[OnMotionDetected](#) [OnMotionNotDetected](#) [RecordingOnMotion_Enabled](#)
[RecordingOnMotion](#) [MotionThreshold](#) [RecordingOnMotion](#) [NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: [Create help files for the Qt Help Framework](#)

MotionDetector_EnumGridDialogControls

TVideoGrabber.MotionDetector_EnumGridDialogControls

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Gives access to the controls of the grid dialog.

Declaration

function MotionDetector_EnumGridDialogControls(FirstControl: Boolean): TComponent;

Classes::TComponent *__fastcall MotionDetector_EnumGridDialogControls(**bool** FirstControl)

n/a

Description

This function enumerates the controls of the grid dialog, including the dialog form.

This lets you e.g. customize the button names, the dialog size, etc...

- invoking MotionDetector_EnumGridDialogControls(True) returns the first control (in fact the dialog form), and resets the counter,
- invoking MotionDetector_EnumGridDialogControls(False) returns the next control, if any, or **nil** when the last control of the dialog has been reached.

E.g.:

```
procedure TForm1.Button1Click(Sender: TObject);
var
    Control: TComponent;
begin
    VideoGrabber1.MotionDetector_ShowGridDialog; // BEFORE...

    Control := VideoGrabber1.MotionDetector_EnumGridDialogControls (True); // reset the counter
    while assigned (Control) do begin // for each control...
        Mem1.Lines.Add (Control.Name + ': ' + Control.ClassName); // shows the control name
        if Control.Name = 'frmGridDialog' then begin
            TForm(Control).Caption := 'this is my custom dialog title';
        end
        else if Control.Name = 'btnOK' then begin
            TButton(Control).Caption := 'DONE';
        end
        else if Control.Name = 'btnCancel' then begin
            TButton(Control).Caption := 'Discard';
        end
        end;
        Control := VideoGrabber1.MotionDetector_EnumGridDialogControls (False); // returns the next control
    end;

    VideoGrabber1.MotionDetector_ShowGridDialog; // AFTER.
end;
```

See Also

[Color / Greyscale Grid structure / grid sensitivity Motion ratio Recording only when motion is detected](#)
[Video noise TOnMotionDetected TOnMotionNotDetected MotionDetector_CellMotionRatio](#)
[MotionDetector_CompareBlue MotionDetector_CompareGreen MotionDetector_CompareRed](#)
[MotionDetector_Enabled MotionDetector_Get2DTextGrid MotionDetector_Get2DTextMotion](#)
[MotionDetector_GetCellLocation MotionDetector_GetCellSensitivity MotionDetector_GetCellSize](#)
[MotionDetector_GloballyIncOrDecSensitivity MotionDetector_GlobalMotionRatio MotionDetector_GreyScale](#)
[MotionDetector_Grid MotionDetector_GridXCount MotionDetector_GridYCount MotionDetector_IsGridValid](#)
[MotionDetector_MaxDetectionsPerSecond MotionDetector_ReduceCPULoad](#)
[MotionDetector_ReduceVideoNoise MotionDetector_Reset MotionDetector_ResetGlobalSensitivity](#)
[MotionDetector_SetCellSensitivity MotionDetector_SetGridSize MotionDetector_ShowGridDialog](#)
[MotionDetector_Triggered MotionDetector_UseThisReferenceSample OnBacktimedFramesCountReached](#)
[OnMotionDetected OnMotionNotDetected RecordingOnMotion_Enabled](#)
[RecordingOnMotion_MotionThreshold RecordingOnMotion_NoMotionPauseDelayMs](#)

MotionDetector_Get2DTextGrid

TVideoGrabber.MotionDetector_Get2DTextGrid

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Retrieves a pseudo 2D text representation of the grid.

Declaration

function MotionDetector_Get2DTextGrid: **string**;

wchar_t * __fastcall MotionDetector_Get2DTextGrid()

Function MotionDetector_Get2DTextGrid() As String

Description

Used to retrieve a pseudo 2D text representation of the grid. Useful for debug / finalization purpose.

See Also

[Color / Greyscale Grid structure / grid sensitivity Motion ratio Recording only when motion is detected Video noise TOnMotionDetected TOnMotionNotDetected MotionDetector_CellMotionRatio MotionDetector_CompareBlue MotionDetector_CompareGreen MotionDetector_CompareRed MotionDetector_Enabled MotionDetector_EnumGridDialogControls MotionDetector_Get2DTextMotion MotionDetector_GetCellLocation MotionDetector_GetCellSensitivity MotionDetector_GetCellSize MotionDetector_GloballyIncOrDecSensitivity MotionDetector_GlobalMotionRatio MotionDetector_GreyScale MotionDetector_Grid MotionDetector_GridXCount MotionDetector_GridYCount MotionDetector_IsGridValid MotionDetector_MaxDetectionsPerSecond MotionDetector_ReduceCPULoad MotionDetector_ReduceVideoNoise MotionDetector_Reset MotionDetector_ResetGlobalSensitivity MotionDetector_SetCellSensitivity MotionDetector_SetGridSize MotionDetector_ShowGridDialog MotionDetector_Triggered MotionDetector_UseThisReferenceSample OnBacktimedFramesCountReached OnMotionDetected OnMotionNotDetected RecordingOnMotion_Enabled RecordingOnMotion_MotionThreshold RecordingOnMotion_NoMotionPauseDelayMs](#)

MotionDetector_Get2DTextMotion

TVideoGrabber.MotionDetector_Get2DTextMotion

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Retrieves a pseudo 2D text representation of the motion.

Declaration

function MotionDetector_Get2DTextMotion: **string**;

wchar_t * __fastcall MotionDetector_Get2DTextMotion()

Function MotionDetector_Get2DTextMotion() As String

Description

Used to retrieve a pseudo 2D text representation of the motion. Useful for debug / finalization purpose.

See Also

[Color / Greyscale Grid structure / grid sensitivity Motion ratio Recording only when motion is detected Video noise TOnMotionDetected TOnMotionNotDetected MotionDetector_CellMotionRatio](#)

[MotionDetector_CompareBlue](#)
[MotionDetector_CompareGreen](#)
[MotionDetector_CompareRed](#)
[MotionDetector_Enabled](#)
[MotionDetector_EnumGridDialogControls](#)
[MotionDetector_Get2DTextGrid](#)
[MotionDetector_GetCellLocation](#)
[MotionDetector_GetCellSensitivity](#)
[MotionDetector_GetCellSize](#)
[MotionDetector_GloballyIncOrDecSensitivity](#)
[MotionDetector_GlobalMotionRatio](#)
[MotionDetector_GreyScale](#)
[MotionDetector_Grid](#)
[MotionDetector_GridXCount](#)
[MotionDetector_GridYCount](#)
[MotionDetector_IsGridValid](#)
[MotionDetector_MaxDetectionsPerSecond](#)
[MotionDetector_ReduceCPULoad](#)
[MotionDetector_ReduceVideoNoise](#)
[MotionDetector_Reset](#)
[MotionDetector_ResetGlobalSensitivity](#)
[MotionDetector_SetCellSensitivity](#)
[MotionDetector_SetGridSize](#)
[MotionDetector_ShowGridDialog](#)
[MotionDetector_Triggered](#)
[MotionDetector_UseThisReferenceSample](#)
[OnBacktimedFramesCountReached](#)
[OnMotionDetected](#)
[OnMotionNotDetected](#)
[RecordingOnMotion_Enabled](#)
[RecordingOnMotion_MotionThreshold](#)
[RecordingOnMotion_NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: Transform Your Documentation Workflow with HelpNDoc's Intuitive UI

MotionDetector_GetCellLocation

TVideoGrabber.MotionDetector_GetCellLocation

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Retrieves the x an y coordinates of a cell in the grid.

Declaration

function MotionDetector_GetCellLocation(x, y: LongInt; **var** XLocation, YLocation: LongInt): Boolean;

bool __fastcall MotionDetector_GetCellLocation(int x, int y, int &XLocation, int &YLocation)

Function MotionDetector_GetCellLocation(x As Long, y As Long, X_Location, Y_Location) As Boolean

Description

Used to retrieve the x an y coordinates of a cell in the grid.

See Also

[Color / Greyscale Grid structure / grid sensitivity](#)
[Motion ratio](#)
[Recording only when motion is detected](#)
[Video noise](#)
[TOnMotionDetected](#)
[TOnMotionNotDetected](#)
[MotionDetector_CellMotionRatio](#)
[MotionDetector_CompareBlue](#)
[MotionDetector_CompareGreen](#)
[MotionDetector_CompareRed](#)
[MotionDetector_Enabled](#)
[MotionDetector_EnumGridDialogControls](#)
[MotionDetector_Get2DTextGrid](#)
[MotionDetector_Get2DTextMotion](#)
[MotionDetector_GetCellSensitivity](#)
[MotionDetector_GetCellSize](#)
[MotionDetector_GloballyIncOrDecSensitivity](#)
[MotionDetector_GlobalMotionRatio](#)
[MotionDetector_GreyScale](#)
[MotionDetector_Grid](#)
[MotionDetector_GridXCount](#)
[MotionDetector_GridYCount](#)
[MotionDetector_IsGridValid](#)
[MotionDetector_MaxDetectionsPerSecond](#)
[MotionDetector_ReduceCPULoad](#)
[MotionDetector_ReduceVideoNoise](#)
[MotionDetector_Reset](#)
[MotionDetector_ResetGlobalSensitivity](#)
[MotionDetector_SetCellSensitivity](#)
[MotionDetector_SetGridSize](#)
[MotionDetector_ShowGridDialog](#)
[MotionDetector_Triggered](#)
[MotionDetector_UseThisReferenceSample](#)
[OnBacktimedFramesCountReached](#)
[OnMotionDetected](#)
[OnMotionNotDetected](#)
[RecordingOnMotion_Enabled](#)
[RecordingOnMotion_MotionThreshold](#)
[RecordingOnMotion_NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: Simplify Your Help Documentation Process with a Help Authoring Tool

MotionDetector_GetCellSensitivity

TVideoGrabber.MotionDetector_GetCellSensitivity

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Retrieves the sensitivity of a cell in the grid.

Declaration

function MotionDetector_GetCellSensitivity(x, y: LongInt; **var** Value: LongInt): Boolean;

bool __fastcall MotionDetector_GetCellSensitivity(int x, int y, int &Value)

Function MotionDetector_GetCellSensitivity(x As Long, y As Long, Value as Long)As Boolean

Description

Used to retrieve the sensitivity of a cell in the grid.

See Also

[Color / Greyscale Grid structure / grid sensitivity](#) [Motion ratio](#) [Recording only when motion is detected](#) [Video noise](#) [TOnMotionDetected](#) [TOnMotionNotDetected](#) [MotionDetector_CellMotionRatio](#) [MotionDetector_CompareBlue](#) [MotionDetector_CompareGreen](#) [MotionDetector_CompareRed](#) [MotionDetector_Enabled](#) [MotionDetector_EnumGridDialogControls](#) [MotionDetector_Get2DTextGrid](#) [MotionDetector_Get2DTextMotion](#) [MotionDetector_GetCellLocation](#) [MotionDetector_GetCellSize](#) [MotionDetector_GloballyIncOrDecSensitivity](#) [MotionDetector_GlobalMotionRatio](#) [MotionDetector_GreyScale](#) [MotionDetector_Grid](#) [MotionDetector_GridXCount](#) [MotionDetector_GridYCount](#) [MotionDetector_IsGridValid](#) [MotionDetector_MaxDetectionsPerSecond](#) [MotionDetector_ReduceCPULoad](#) [MotionDetector_ReduceVideoNoise](#) [MotionDetector_Reset](#) [MotionDetector_ResetGlobalSensitivity](#) [MotionDetector_SetCellSensitivity](#) [MotionDetector_SetGridSize](#) [MotionDetector_ShowGridDialog](#) [MotionDetector_Triggered](#) [MotionDetector_UseThisReferenceSample](#) [OnBacktimedFramesCountReached](#) [OnMotionDetected](#) [OnMotionNotDetected](#) [RecordingOnMotion_Enabled](#) [RecordingOnMotion_MotionThreshold](#) [RecordingOnMotion_NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: Maximize Your Documentation Output with HelpNDoc's Advanced Project Analyzer

MotionDetector_GetCellSize

TVideoGrabber.MotionDetector_GetCellSize

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Retrieves the x and y sizes of a cell in the grid.

Declaration

function MotionDetector_GetCellSize(var XSize, YSize: LongInt): Boolean;

bool __fastcall MotionDetector_GetCellSize(int &XSize, int &YSize)

Function MotionDetector_GetCellSize(X_Size, Y_Size) As Boolean

Description

Used to retrieve the x and y sizes of a cell in the grid.

See Also

[Color / Greyscale Grid structure / grid sensitivity](#) [Motion ratio](#) [Recording only when motion is detected](#) [Video noise](#) [TOnMotionDetected](#) [TOnMotionNotDetected](#) [MotionDetector_CellMotionRatio](#) [MotionDetector_CompareBlue](#) [MotionDetector_CompareGreen](#) [MotionDetector_CompareRed](#) [MotionDetector_Enabled](#) [MotionDetector_EnumGridDialogControls](#) [MotionDetector_Get2DTextGrid](#) [MotionDetector_Get2DTextMotion](#) [MotionDetector_GetCellLocation](#) [MotionDetector_GetCellSensitivity](#) [MotionDetector_GloballyIncOrDecSensitivity](#) [MotionDetector_GlobalMotionRatio](#) [MotionDetector_GreyScale](#) [MotionDetector_Grid](#) [MotionDetector_GridXCount](#) [MotionDetector_GridYCount](#) [MotionDetector_IsGridValid](#) [MotionDetector_MaxDetectionsPerSecond](#) [MotionDetector_ReduceCPULoad](#) [MotionDetector_ReduceVideoNoise](#) [MotionDetector_Reset](#) [MotionDetector_ResetGlobalSensitivity](#) [MotionDetector_SetCellSensitivity](#) [MotionDetector_SetGridSize](#) [MotionDetector_ShowGridDialog](#) [MotionDetector_Triggered](#) [MotionDetector_UseThisReferenceSample](#) [OnBacktimedFramesCountReached](#) [OnMotionDetected](#) [OnMotionNotDetected](#) [RecordingOnMotion_Enabled](#) [RecordingOnMotion_MotionThreshold](#) [RecordingOnMotion_NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: Elevate Your CHM Help Files with HelpNDoc's Advanced Customization Options

MotionDetector_GlobalColorIntensity

TVideoGrabber.MotionDetector_GlobalColorIntensity

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Retrieves the global average color intensity of the last video frame.

Declaration

function MotionDetector_GlobalColorIntensity (RGBSelector: TRGBSelector): LongInt;

int __fastcall MotionDetector_GlobalColorIntensity(TRGBSelector RGBSelector);

function MotionDetector_GlobalColorIntensity (RGBSelector as TRGBSelector) as Long

Description

Used to retrieve the global average color intensity of the last video frame.
The [RGBSelector](#) parameter specifies the color to retrieve (R, G or B).

Created with the Standard Edition of HelpNDoc: [Revolutionize Your CHM Help File Output with HelpNDoc](#)

MotionDetector_GloballyIncOrDecSensitivity

TVideoGrabber.MotionDetector_GloballyIncOrDecSensitivity

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Increases or decreases the sensitivity of the grid cells.

Declaration

procedure MotionDetector_GloballyIncOrDecSensitivity(Value: integer);

void __fastcall MotionDetector_GloballyIncOrDecSensitivity(int Value)

Sub MotionDetector_GloballyIncOrDecSensitivity(Value As Long)

Description

Used to globally increase or decrease the sensitivity of the cells whose current sensitivity is greater than 0 (this increases or decreases only the sensitivity of the active cells).
The value passed as parameter is usually -1 (decrease) or 1 (increase).

See Also

[Color / Greyscale Grid structure / grid sensitivity Motion ratio Recording only when motion is detected Video noise TOnMotionDetected TOnMotionNotDetected MotionDetector_CellMotionRatio MotionDetector_CompareBlue MotionDetector_CompareGreen MotionDetector_CompareRed MotionDetector_Enabled MotionDetector_EnumGridDialogControls MotionDetector_Get2DTextGrid MotionDetector_Get2DTextMotion MotionDetector_GetCellLocation MotionDetector_GetCellSensitivity MotionDetector_GetCellSize MotionDetector_GlobalMotionRatio MotionDetector_GreyScale MotionDetector_Grid MotionDetector_GridXCount MotionDetector_GridYCount MotionDetector_IsGridValid MotionDetector_MaxDetectionsPerSecond MotionDetector_ReduceCPULoad MotionDetector_ReduceVideoNoise MotionDetector_Reset MotionDetector_ResetGlobalSensitivity MotionDetector_SetCellSensitivity MotionDetector_SetGridSize MotionDetector_ShowGridDialog MotionDetector_Triggered MotionDetector_UseThisReferenceSample OnBacktimedFramesCountReached OnMotionDetected OnMotionNotDetected RecordingOnMotion Enabled RecordingOnMotion MotionThreshold RecordingOnMotion NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Support Your Windows Applications with HelpNDoc's CHM Generation](#)

MotionDetector_Reset

TVideoGrabber.MotionDetector_Reset

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Disables temporarily the motion detection for the next frame.

Declaration

procedure MotionDetector_Reset;

void __fastcall MotionDetector_Reset(**void**)

Sub MotionDetector_Reset()

Description

Used to disable temporarily the motion detection for the next frame.

Call this method if the next frame is not frame that normally follows the previous one currently stored by the component (e.g. the motion detection has been stopped for a moment, now it restarts but you don't want the 1st frame to be interpreted as motion because it differs from the last one stored a few minutes ago).

See Also

[Color / Greyscale Grid structure / grid sensitivity](#) [Motion ratio](#) [Recording only when motion is detected](#) [Video noise](#) [TOnMotionDetected](#) [TOnMotionNotDetected](#) [MotionDetector_CellMotionRatio](#) [MotionDetector_CompareBlue](#) [MotionDetector_CompareGreen](#) [MotionDetector_CompareRed](#) [MotionDetector_Enabled](#) [MotionDetector_EnumGridDialogControls](#) [MotionDetector_Get2DTextGrid](#) [MotionDetector_Get2DTextMotion](#) [MotionDetector_GetCellLocation](#) [MotionDetector_GetCellSensitivity](#) [MotionDetector_GetCellSize](#) [MotionDetector_GloballyIncOrDecSensitivity](#) [MotionDetector_GlobalMotionRatio](#) [MotionDetector_GreyScale](#) [MotionDetector_Grid](#) [MotionDetector_GridXCount](#) [MotionDetector_GridYCount](#) [MotionDetector_IsGridValid](#) [MotionDetector_MaxDetectionsPerSecond](#) [MotionDetector_ReduceCPULoad](#) [MotionDetector_ReduceVideoNoise](#) [MotionDetector_ResetGlobalSensitivity](#) [MotionDetector_SetCellSensitivity](#) [MotionDetector_SetGridSize](#) [MotionDetector_ShowGridDialog](#) [MotionDetector_Triggered](#) [MotionDetector_UseThisReferenceSample](#) [OnBacktimedFramesCountReached](#) [OnMotionDetected](#) [OnMotionNotDetected](#) [RecordingOnMotion_Enabled](#) [RecordingOnMotion](#) [MotionThreshold](#) [RecordingOnMotion_NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: Effortlessly Create High-Quality Help Documentation with a Help Authoring Tool

MotionDetector_ResetGlobalSensitivity

TVideoGrabber.MotionDetector_ResetGlobalSensitivity

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Gobally sets/resets sensitivity of all cells.

Declaration

procedure MotionDetector_ResetGlobalSensitivity(Value: integer);

void __fastcall MotionDetector_ResetGlobalSensitivity(**int** Value)

Sub MotionDetector_ResetGlobalSensitivity(Value As Long)

Description

Used to globally set/reset the sensitivity of all cells, in the 0..9 range.

See Also

[Color / Greyscale Grid structure / grid sensitivity](#) [Motion ratio](#) [Recording only when motion is detected](#)

[Video noise](#)
[TOnMotionDetected](#)
[TOnMotionNotDetected](#)
[MotionDetector](#)
[CellMotionRatio](#)
[MotionDetector](#)
[CompareBlue](#)
[MotionDetector](#)
[CompareGreen](#)
[MotionDetector](#)
[CompareRed](#)
[MotionDetector](#)
[Enabled](#)
[MotionDetector](#)
[EnumGridDialogControls](#)
[MotionDetector](#)
[Get2DTextGrid](#)
[MotionDetector](#)
[Get2DTextMotion](#)
[MotionDetector](#)
[GetCellLocation](#)
[MotionDetector](#)
[GetCellSensitivity](#)
[MotionDetector](#)
[GetCellSize](#)
[MotionDetector](#)
[GloballyIncOrDecSensitivity](#)
[MotionDetector](#)
[GlobalMotionRatio](#)
[MotionDetector](#)
[GreyScale](#)
[MotionDetector](#)
[Grid](#)
[MotionDetector](#)
[GridXCount](#)
[MotionDetector](#)
[GridYCount](#)
[MotionDetector](#)
[IsGridValid](#)
[MotionDetector](#)
[MaxDetectionsPerSecond](#)
[MotionDetector](#)
[ReduceCPULoad](#)
[MotionDetector](#)
[ReduceVideoNoise](#)
[MotionDetector](#)
[Reset](#)
[MotionDetector](#)
[SetCellSensitivity](#)
[MotionDetector](#)
[SetGridSize](#)
[MotionDetector](#)
[ShowGridDialog](#)
[MotionDetector](#)
[Triggered](#)
[MotionDetector](#)
[UseThisReferenceSample](#)
[OnBacktimedFramesCountReached](#)
[OnMotionDetected](#)
[OnMotionNotDetected](#)
[RecordingOnMotion](#)
[Enabled](#)
[RecordingOnMotion](#)
[MotionThreshold](#)
[RecordingOnMotion](#)
[NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: [Free help authoring environment](#)

MotionDetector_SetCellSensitivity

TVideoGrabber.MotionDetector_SetCellSensitivity

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets the sensitivity of one cell in the grid.

Declaration

function MotionDetector_SetCellSensitivity(x, y, Value: LongInt): Boolean;

bool __fastcall MotionDetector_SetCellSensitivity(int x, int y, int Value)

Function MotionDetector_SetCellSensitivity(x As Long, y As Long, Value As Long) As Boolean

Description

Used to set the sensitivity of one cell in the grid.

Useful if the dialog cannot be used to define sensitivity areas because the grid is too dense (usually above 40 x 40).

See Also

[Color / Greyscale Grid structure / grid sensitivity](#)
[Motion ratio](#)
[Recording only when motion is detected](#)
[Video noise](#)
[TOnMotionDetected](#)
[TOnMotionNotDetected](#)
[MotionDetector](#)
[CellMotionRatio](#)
[MotionDetector](#)
[CompareBlue](#)
[MotionDetector](#)
[CompareGreen](#)
[MotionDetector](#)
[CompareRed](#)
[MotionDetector](#)
[Enabled](#)
[MotionDetector](#)
[EnumGridDialogControls](#)
[MotionDetector](#)
[Get2DTextGrid](#)
[MotionDetector](#)
[Get2DTextMotion](#)
[MotionDetector](#)
[GetCellLocation](#)
[MotionDetector](#)
[GetCellSensitivity](#)
[MotionDetector](#)
[GetCellSize](#)
[MotionDetector](#)
[GloballyIncOrDecSensitivity](#)
[MotionDetector](#)
[GlobalMotionRatio](#)
[MotionDetector](#)
[GreyScale](#)
[MotionDetector](#)
[Grid](#)
[MotionDetector](#)
[GridXCount](#)
[MotionDetector](#)
[GridYCount](#)
[MotionDetector](#)
[IsGridValid](#)
[MotionDetector](#)
[MaxDetectionsPerSecond](#)
[MotionDetector](#)
[ReduceCPULoad](#)
[MotionDetector](#)
[ReduceVideoNoise](#)
[MotionDetector](#)
[Reset](#)
[MotionDetector](#)
[ResetGlobalSensitivity](#)
[MotionDetector](#)
[SetGridSize](#)
[MotionDetector](#)
[ShowGridDialog](#)
[MotionDetector](#)
[Triggered](#)
[MotionDetector](#)
[UseThisReferenceSample](#)
[OnBacktimedFramesCountReached](#)
[OnMotionDetected](#)
[OnMotionNotDetected](#)
[RecordingOnMotion](#)
[Enabled](#)
[RecordingOnMotion](#)
[MotionThreshold](#)
[RecordingOnMotion](#)
[NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: [5 Reasons Why a Help Authoring Tool is Better than Microsoft Word for Documentation](#)

MotionDetector_SetGridSize

TVideoGrabber.MotionDetector_SetGridSize

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Modifies the size of the grid.

Declaration

procedure MotionDetector_SetGridSize(x, y: LongInt);

void __fastcall MotionDetector_SetGridSize(int x, int y)

Sub MotionDetector_SetGridSize(x As Long, y As Long)

Description

Used to modify the size of the grid.

The grid can be set from 1 x 1 to the *width x height* of the video frame.

If the grid oversize the frame, only the top/left part of the grid that covers the frame is active.

See Also

[Color / Greyscale Grid structure / grid sensitivity](#) [Motion ratio](#) [Recording only when motion is detected](#) [Video noise](#) [TOnMotionDetected](#) [TOnMotionNotDetected](#) [MotionDetector_CellMotionRatio](#) [MotionDetector_CompareBlue](#) [MotionDetector_CompareGreen](#) [MotionDetector_CompareRed](#) [MotionDetector_Enabled](#) [MotionDetector_EnumGridDialogControls](#) [MotionDetector_Get2DTextGrid](#) [MotionDetector_Get2DTextMotion](#) [MotionDetector_GetCellLocation](#) [MotionDetector_GetCellSensitivity](#) [MotionDetector_GetCellSize](#) [MotionDetector_GloballyIncOrDecSensitivity](#) [MotionDetector_GlobalMotionRatio](#) [MotionDetector_GreyScale](#) [MotionDetector_Grid](#) [MotionDetector_GridXCount](#) [MotionDetector_GridYCount](#) [MotionDetector_IsGridValid](#) [MotionDetector_MaxDetectionsPerSecond](#) [MotionDetector_ReduceCPULoad](#) [MotionDetector_ReduceVideoNoise](#) [MotionDetector_Reset](#) [MotionDetector_ResetGlobalSensitivity](#) [MotionDetector_SetCellSensitivity](#) [MotionDetector_ShowGridDialog](#) [MotionDetector_Triggered](#) [MotionDetector_UseThisReferenceSample](#) [OnBacktimedFramesCountReached](#) [OnMotionDetected](#) [OnMotionNotDetected](#) [RecordingOnMotion_Enabled](#) [RecordingOnMotion_MotionThreshold](#) [RecordingOnMotion_NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: Eliminate the Struggles of Documentation with a Help Authoring Tool

MotionDetector_ShowGridDialog

TVideoGrabber.MotionDetector_ShowGridDialog

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Displays the grid dialog.

Declaration

procedure MotionDetector_ShowGridDialog;

void __fastcall MotionDetector_ShowGridDialog(void)

Sub MotionDetector_ShowGridDialog()

Description

Used to display the grid dialog that allows to set the sensitivity of each cell interactively.

If at least one frame has been processed, the dialog shows the grid over the video frame, allowing to adjust accurately the sensitivity of the video areas.

See Also

[Color / Greyscale Grid structure / grid sensitivity](#) [Motion ratio](#) [Recording only when motion is detected](#) [Video noise](#) [TOnMotionDetected](#) [TOnMotionNotDetected](#) [MotionDetector_CellMotionRatio](#) [MotionDetector_CompareBlue](#) [MotionDetector_CompareGreen](#) [MotionDetector_CompareRed](#) [MotionDetector_Enabled](#) [MotionDetector_EnumGridDialogControls](#) [MotionDetector_Get2DTextGrid](#) [MotionDetector_Get2DTextMotion](#) [MotionDetector_GetCellLocation](#) [MotionDetector_GetCellSensitivity](#) [MotionDetector_GetCellSize](#) [MotionDetector_GloballyIncOrDecSensitivity](#) [MotionDetector_GlobalMotionRatio](#) [MotionDetector_GreyScale](#) [MotionDetector_Grid](#)

[MotionDetector_GridXCount](#)
[MotionDetector_GridYCount](#)
[MotionDetector_IsGridValid](#)
[MotionDetector_MaxDetectionsPerSecond](#)
[MotionDetector_ReduceCPULoad](#)
[MotionDetector_ReduceVideoNoise](#)
[MotionDetector_Reset](#)
[MotionDetector_ResetGlobalSensitivity](#)
[MotionDetector_SetCellSensitivity](#)
[MotionDetector_SetGridSize](#)
[MotionDetector_Triggered](#)
[MotionDetector_UseThisReferenceSample](#)
[OnBacktimedFramesCountReached](#)
[OnMotionDetected](#)
[OnMotionNotDetected](#)
[RecordingOnMotion_Enabled](#)
[RecordingOnMotion_MotionThreshold](#)
[RecordingOnMotion_NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: [Create Professional CHM Help Files with HelpNDoc's Easy-to-Use Tool](#)

MotionDetector_TriggerNow

TVideoGrabber.MotionDetector_TriggerNow

[Prev](#)

[Next](#)

[TVideoGrabber](#)
[Methods](#)

Triggers a motion detection.

Declaration

procedure MotionDetector_TriggerNow;

void __fastcall MotionDetector_TriggerNow(**void**)

Sub MotionDetector_TriggerNow()

Description

Used to trigger a motion detection, when the [MotionDetector_Triggered](#) property is enabled. After invoking this method, the motion detection will occur one time for the next video frame, until the next MotionDetector_TriggerNow call.

Created with the Standard Edition of HelpNDoc: [Free iPhone documentation generator](#)

MotionDetector_UseThisReferenceSample

TVideoGrabber.MotionDetector_UseThisReferenceSample

[Prev](#)

[Next](#)

[TVideoGrabber](#)
[Methods](#)

Let specify a reference sample used for motion detection.

Declaration

function MotionDetector_UseThisReferenceSample (Bitmap: TBitmap; BMPFile: **string**; JPEGFile: **string**): **Boolean**;

bool __fastcall MotionDetector_UseThisReferenceSample (Graphics::TBitmap *TBitmap, wchar_t *BMPFile, wchar_t *JPEGFile)

Function MotionDetector_UseThisReferenceSample (Bitmap as Long, BMPFile as string, JPEGFile as string) As Boolean

Description

Let specify a video frame that will be used as reference for motion detection, instead of the previous frame.

The frame can be passed in 3 ways, by using one of the 3 parameters:

1. **Bitmap**: as a pointer to a TBitmap (or a bitmap handle in the OCX versions), e.g.:
[VideoGrabber.MotionDetector_UseThisReferenceSample \(MyTBitmapReference, "", ""\)](#);

2. **BMPFile**: as a path to a BMP image, e.g.:

```
VideoGrabber.MotionDetector_UseThisReferenceSample (nil, "MyBitmapReference.bmp", "");
```

3. **JPEGFile**: as a path to a JPEG file, e.g.:

```
VideoGrabber.MotionDetector_UseThisReferenceSample (nil, "", "MyBitmapReference.jpg");
```

To deactivate the feature and reuse the last frame as reference, pass empty parameters, e.g.:

```
VideoGrabber.MotionDetector_UseThisReferenceSample (nil, "", "");
```

See Also

[Color / Greyscale Grid structure / grid sensitivity](#) [Motion ratio](#) [Recording only when motion is detected](#) [Video noise](#) [TOnMotionDetected](#) [TOnMotionNotDetected](#) [MotionDetector_CellMotionRatio](#) [MotionDetector_CompareBlue](#) [MotionDetector_CompareGreen](#) [MotionDetector_CompareRed](#) [MotionDetector_Enabled](#) [MotionDetector_EnumGridDialogControls](#) [MotionDetector_Get2DTextGrid](#) [MotionDetector_Get2DTextMotion](#) [MotionDetector_GetCellLocation](#) [MotionDetector_GetCellSensitivity](#) [MotionDetector_GetCellSize](#) [MotionDetector_GloballyIncOrDecSensitivity](#) [MotionDetector_GlobalMotionRatio](#) [MotionDetector_GreyScale](#) [MotionDetector_Grid](#) [MotionDetector_GridXCount](#) [MotionDetector_GridYCount](#) [MotionDetector_IsGridValid](#) [MotionDetector_MaxDetectionsPerSecond](#) [MotionDetector_ReduceCPULoad](#) [MotionDetector_ReduceVideoNoise](#) [MotionDetector_Reset](#) [MotionDetector_ResetGlobalSensitivity](#) [MotionDetector_SetCellSensitivity](#) [MotionDetector_SetGridSize](#) [MotionDetector_ShowGridDialog](#) [MotionDetector_Triggered](#) [OnBacktimedFramesCountReached](#) [OnMotionDetected](#) [OnMotionNotDetected](#) [RecordingOnMotion_Enabled](#) [RecordingOnMotion](#) [MotionThreshold](#) [RecordingOnMotion_NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: Free EBook and documentation generator

MP4NeedsReindexing

TVideoGrabber.MP4NeedsReindexing

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Determines if a MP4 video clip requires reindexing.

Declaration

```
function TVideoGrabber.MP4NeedsReindexing (FilePath: string): TTriState;
```

```
TTriState __fastcall MP4NeedsReindexing(System::UnicodeString FilePath);
```

Description

Used to determines if a MP4 video clip requires a reindexing.

Return values:

ts_Undefined: failed to open the clip

ts_False: the clip does not need a reindexing

ts_True: the clip needs to be reindexed

See [MultipurposeEncoder_ReindexClip](#)

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [TVideoDeinterlacing](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MultipurposeEncoder_ReindexClip](#) [OnFrameBitmap](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchron](#) [PlayerVideoCodec](#) [RetrievalInitialXYAfterRotation](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#)

[StopPlayer](#)
[SynchronizationRole](#)
[Synchronized](#)
[VideoHeight](#)
[PreferredAspectRatio](#)
[VideoPlayableWhileRecording](#)
[VideoProcessing](#)
[Brightness](#)
[VideoProcessing](#)
[Contrast](#)
[VideoProcessing](#)
[Deinterlacing](#)
[VideoProcessing](#)
[FlipHorizontal](#)
[VideoProcessing](#)
[FlipVertical](#)
[VideoProcessing](#)
[GrayScale](#)
[VideoProcessing](#)
[Hue](#)
[VideoProcessing](#)
[InvertColors](#)
[VideoProcessing](#)
[Pixellization](#)
[VideoProcessing](#)
[Rotation](#)
[VideoProcessing](#)
[RotationCustomAngle](#)
[VideoProcessing](#)
[Saturation](#)
[VideoWidth](#)
[PreferredAspectRatio](#)
[TVideoRotation](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly optimize your documentation website for search engines](#)

MPEGProgramSetting

TVideoGrabber.MPEGProgramSetting

[Prev](#)
[Next](#)

[TVideoGrabber](#)
[Methods](#)

Specifies the program setting to be used

Declaration

function MPEGProgramSetting (MPEGProgramSetting: TMPEGProgramSetting; Value: LongInt): LongInt;

int __fastcall MPEGProgramSetting(TMPEGProgramSetting MPEGProgramSetting, **int** Value);

function MPEGProgramSetting (MPEGProgramSetting as TxMPEGProgramSetting, Value as Long): as Long

Description

Used to specify the [TMPEGProgramSetting](#) MPEG program number to be used when an UDP MPEG Transport Stream network source is played

Created with the Standard Edition of HelpNDoc: [Maximize Your Documentation Capabilities with HelpNDoc's User-Friendly UI](#)

MultipurposeEncoder_QuickConfigure_UDPStreaming_H264

TVideoGrabber.MultipurposeEncoder_QuickConfigure_UDPStreaming_H264

[Prev](#)
[Next](#)

[TVideoGrabber](#)
[Methods](#)

Quick configure the Multipurpose Encoder for H264 streaming (unicast or multicast)

Declaration

function MultipurposeEncoder_QuickConfigure_UDPStreaming_H264 (LogToFile: Boolean; VideoEnabled: Boolean; AudioEnabled: Boolean; DestinationIP: **string**; DestinationPort: LongInt; VideoBitRateKb: LongInt; AudioBitRateKb: LongInt): Boolean;

System::wchar_t *__fastcall MultipurposeEncoder_QuickConfigure_UDPStreaming_H264(**bool** LogToFile, **bool** VideoEnabled, **bool** AudioEnabled, System::wchar_t *DestinationIP, **int** DestinationPort, **int** VideoBitRateKb, **int** AudioBitRateKb);

Description

Parameters:

LogToFile: Boolean: if true, a log file is generated in the current folder

VideoEnabled: Boolean: if true, the video is streamed

AudioEnabled: Boolean: if true, the audio is streamed

DestinationIP: string: destination IP (unicast or multicast)

DestinationPort: LongInt: destination port

VideoBitRateKb: LongInt: the video bitrate expressed in Kb/s, e.g. 2000Kbs

AudioBitRateKb: LongInt: the audio bitrate expressed in Kb/s, e.g. 128Kbs

See Also

[MultipurposeEncoder_ReindexClip](#) [SetFFmpegAudioFilter](#) [SetFFmpegFilter](#)

Created with the Standard Edition of HelpNDoc: [Streamline your documentation process with HelpNDoc's HTML5 template](#)

MultipurposeEncoder_ReindexClip

TVideoGrabber.MultipurposeEncoder_ReindexClip

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Used to reindex a video clip

Declaration

function MultipurposeEncoder_ReindexClip (CurrentFileName: **string**; NewFileName: **string**): Boolean;

bool __fastcall MultipurposeEncoder_ReindexClip(System::UnicodeString CurrentFileName, System::UnicodeString NewFileName);

Description

This function reindexes the video clip specified as **CurrentFileName** video clip and produces a new video clip specified by **NewFileName**.

The reindexing is performed without recompressing the video stream, so the process is quite fast.

The function is not specific to MP4, it can reindex MKV, AVI or other formats.

It is useful when seeking a video clip is slow.

You can determine if a MP4 video clip needs a reindexing by invoking [MP4NeedsReindexing](#).

E.g.:

SourceFileName = "c:

NewFileName := "c:

if (VideoGrabber->MP4NeedsReindexing (SourceFileName))

VideoGrabber->MultipurposeEncoder_ReindexClip (SourceFileName, NewFileName);

See Also

[MP4NeedsReindexing](#) [MultipurposeEncoder](#) [QuickConfigure](#) [UDPStreaming](#) [H264](#) [SetFFmpegAudioFilter](#) [SetFFmpegFilter](#)

Created with the Standard Edition of HelpNDoc: [Upgrade Your Documentation Process with a Help Authoring Tool](#)

NotifyPlayerTrackbarAction

TVideoGrabber.NotifyPlayerTrackbarAction

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Invoked by your trackbar events

Declaration

procedure NotifyPlayerTrackbarAction (TrackbarAction: TTrackbarAction);

void __fastcall NotifyPlayerTrackbarAction(TTrackbarAction TrackbarAction);

Sub NotifyPlayerTrackbarActio (TrackbarAction as TxTrackbarAction)

Description

If you are implementing a player trackbar, invoke this procedure from your trackbar's OnMouseDown, OnMouseUp, OnKeyDown and OnKeyUp events.

See the player's [Trackbar](#) chapter for more information.

ONVIF_GetStr

TVideoGrabber.ONVIF_GetStr

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Retrieves settings from the Datastead [RTSP/RTMP/HTTP/ONVIF DirectShow Source Filter](#)

Declaration

function ONVIF_GetStr (ParamIdentifier: **string**; var Value: **string**): Boolean;

bool __fastcall ONVIF_GetStr(System::wchar_t *ParamIdentifier, System::wchar_t *&Value);

Description

Used to retrieve settings from the Datastead [RTSP/RTMP/HTTP/ONVIF DirectShow Source Filter](#)

See Also

[IR Cut Filter of Axis cameras](#) [TDiscoveryCallbackStatus](#) [TOnONVIFDiscoveryCompletedNotification](#) [TONVIFDeviceInfo](#) [Video stream of IP camera with audio of PC microphone or other audio capture device](#) [ExtraDLLPath](#) [GetLastError](#) [IPCameraURL](#) [IsURLResponding](#) [IsURLVideoStreamAvailable](#) [OnONVIFDiscoveryCompletedNotification](#) [ONVIF_GetStr](#) [ONVIFCancelDiscovery](#) [ONVIFDeviceInfo](#) [ONVIFDiscoverCameras](#) [IPRange](#) [ONVIFDiscoverCameras_Multicast](#) [ONVIFEnumCamerasDiscovered](#) [ONVIFPTZGetLimits](#) [ONVIFPTZGetPosition](#) [ONVIFPTZPreset](#) [ONVIFPTZSendAuxiliaryCommand](#) [ONVIFPTZSetPosition](#) [ONVIFPTZStartMove](#) [ONVIFPTZStopMove](#) [ONVIFSnapshot](#) [OpenURLAsync](#) [OpenURLAsyncStatus](#) [SendIPCameraCommand](#) [SetDatasteadFilterDllName](#) [SetIPCameraSetting](#)

ONVIF_SetStr

TVideoGrabber.ONVIF_SetStr

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Pass settings to the Datastead [RTSP/RTMP/HTTP/ONVIF DirectShow Source Filter](#)

Declaration

function ONVIF_SetStr (ParamIdentifier: **string**; Value: **string**): Boolean;

bool __fastcall ONVIF_SetStr(System::wchar_t *ParamIdentifier, System::wchar_t *Value);

Description

Used to pass settings to the Datastead [RTSP/RTMP/HTTP/ONVIF DirectShow Source Filter](#)

See Also

[IR Cut Filter of Axis cameras](#) [TDiscoveryCallbackStatus](#) [TOnONVIFDiscoveryCompletedNotification](#) [TONVIFDeviceInfo](#) [Video stream of IP camera with audio of PC microphone or other audio capture device](#) [ExtraDLLPath](#) [GetLastError](#) [IPCameraURL](#) [IsURLResponding](#) [IsURLVideoStreamAvailable](#) [OnONVIFDiscoveryCompletedNotification](#) [ONVIF_GetStr](#) [ONVIFCancelDiscovery](#) [ONVIFDeviceInfo](#) [ONVIFDiscoverCameras](#) [IPRange](#) [ONVIFDiscoverCameras_Multicast](#) [ONVIFEnumCamerasDiscovered](#) [ONVIFPTZGetLimits](#) [ONVIFPTZGetPosition](#) [ONVIFPTZPreset](#) [ONVIFPTZSendAuxiliaryCommand](#) [ONVIFPTZSetPosition](#) [ONVIFPTZStartMove](#) [ONVIFPTZStopMove](#) [ONVIFSnapshot](#) [OpenURLAsync](#) [OpenURLAsyncStatus](#) [SendIPCameraCommand](#) [SetDatasteadFilterDllName](#) [SetIPCameraSetting](#)

ONVIFCancelDiscovery

TVideoGrabber.ONVIFCancelDiscovery

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Methods](#)

Cancels the ONVIF discovery

Declaration

function ONVIFCancelDiscovery: Boolean;

bool __fastcall ONVIFCancelDiscovery();

Description

Invoke it to cancel a [ONVIFDiscoverCameras_Multicast](#) or [ONVIFDiscoverCameras_IPRange](#) background process currently running

See Also

[IR Cut Filter of Axis cameras TDiscoveryCallbackStatus TOnONVIFDiscoveryCompletedNotification TONVIFDeviceInfo](#) [Video stream of IP camera with audio of PC microphone or other audio capture device](#) [ExtraDLLPath](#) [GetLastErrorMessage](#) [IPCameraURL](#) [IsURLResponding](#) [IsURLVideoStreamAvailable](#) [OnONVIFDiscoveryCompletedNotification](#) [ONVIF_GetStr](#) [ONVIF_SetStr](#) [ONVIFDeviceInfo](#) [ONVIFDiscoverCameras_IPRange](#) [ONVIFDiscoverCameras_Multicast](#) [ONVIFEnumCamerasDiscovered](#) [ONVIFPTZGetLimits](#) [ONVIFPTZGetPosition](#) [ONVIFPTZPreset](#) [ONVIFPTZSendAuxiliaryCommand](#) [ONVIFPTZSetPosition](#) [ONVIFPTZStartMove](#) [ONVIFPTZStopMove](#) [ONVIFSnapShot](#) [OpenURLAsync](#) [OpenURLAsyncStatus](#) [SendIPCameraCommand](#) [SetDatasteadFilterDllName](#) [SetIPCameraSetting](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Reach: Convert Your Word Document to an ePub or Kindle eBook](#)

ONVIFDeviceInfo

TVideoGrabber.ONVIFDeviceInfo

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Methods](#)

Retrieve ONVIF information

Declaration

function ONVIFDeviceInfo (ONVIFDeviceInfoType: TONVIFDeviceInfo): **string**;

const wchar_t * *_ONVIFDeviceInfo (void *TVGObject, TONVIFDeviceInfo ONVIFDeviceInfoType);

Description

Returns the specified ONVIF information as a string.
Set [ONVIFDeviceInfoType](#) with the [type of ONVIF information](#) desired.

See Also

[IR Cut Filter of Axis cameras TDiscoveryCallbackStatus TOnONVIFDiscoveryCompletedNotification TONVIFDeviceInfo](#) [ExtraDLLPath](#) [GetLastErrorMessage](#) [OnONVIFDiscoveryCompletedNotification](#) [ONVIF_GetStr](#) [ONVIF_SetStr](#) [ONVIFCancelDiscovery](#) [ONVIFDiscoverCameras_IPRange](#) [ONVIFDiscoverCameras_Multicast](#) [ONVIFEnumCamerasDiscovered](#) [ONVIFPTZGetLimits](#) [ONVIFPTZGetPosition](#) [ONVIFPTZPreset](#) [ONVIFPTZSendAuxiliaryCommand](#) [ONVIFPTZSetPosition](#) [ONVIFPTZStartMove](#) [ONVIFPTZStopMove](#) [ONVIFSnapShot](#) [OpenURLAsyncStatus](#) [SetDatasteadFilterDllName](#)

Created with the Standard Edition of HelpNDoc: [What is a Help Authoring tool?](#)

ONVIFDiscoverCameras_IPRange

TVideoGrabber.ONVIFDiscoverCameras_IPRange

[Prev](#)
[Next](#)

TVideoGrabber Methods

ONVIF discovery from a range of IP addresses

Declaration

function ONVIFDiscoverCameras_IPRange (First_IP: **string**; Last_IP: **string**; timeout_seconds_or_0_for_default: LongInt): Boolean;

bool __fastcall ONVIFDiscoverCameras_IPRange(System::UnicodeString First_IP, System::UnicodeString Last_IP, System::LongInt timeout_seconds_or_0_for_default);

Description

Scans the network for ONVIF cameras between the ip addresses specified, e.g. from 192.168.0.1 to 192.168.0.254

See [ONVIF Cameras Discovery](#)

See Also

[IR Cut Filter of Axis cameras](#) [TDiscoveryCallbackStatus](#) [TOnONVIFDiscoveryCompletedNotification](#) [TONVIFDeviceInfo](#) [Video stream of IP camera with audio of PC microphone or other audio capture device](#) [ExtraDLLPath](#) [GetLastErrorMessage](#) [IPCameraURL](#) [IsURLResponding](#) [IsURLVideoStreamAvailable](#) [OnONVIFDiscoveryCompletedNotification](#) [ONVIF_GetStr](#) [ONVIF_SetStr](#) [ONVIFCancelDiscovery](#) [ONVIFDeviceInfo](#) [ONVIFDiscoverCameras_IPRange](#) [Multicast](#) [ONVIFEnumCamerasDiscovered](#) [ONVIFPTZGetLimits](#) [ONVIFPTZGetPosition](#) [ONVIFPTZPreset](#) [ONVIFPTZSendAuxiliaryCommand](#) [ONVIFPTZSetPosition](#) [ONVIFPTZStartMove](#) [ONVIFPTZStopMove](#) [ONVIFSnapShot](#) [OpenURLAsync](#) [OpenURLAsyncStatus](#) [SendIPCameraCommand](#) [SetDatasteadFilterDllName](#) [SetIPCameraSetting](#)

Created with the Standard Edition of HelpNDoc: [Leave the tedious WinHelp HLP to CHM conversion process behind with HelpNDoc](#)

ONVIFDiscoverCameras_Multicast

TVideoGrabber.ONVIFDiscoverCameras_Multicast

[Prev](#)

[Next](#)

TVideoGrabber Methods

ONVIF discovery from a Multicast request

Declaration

function ONVIFDiscoverCameras_Multicast (timeout_seconds_or_0_for_default: LongInt): Boolean;

bool __fastcall ONVIFDiscoverCameras_Multicast(System::LongInt timeout_seconds_or_0_for_default);

Description

Scans the network for ONVIF cameras by broadcasting a multicast request.

Note that the network switche(s) must support IGMP snooping.

See [ONVIF Cameras Discovery](#)

See Also

[IR Cut Filter of Axis cameras](#) [TDiscoveryCallbackStatus](#) [TOnONVIFDiscoveryCompletedNotification](#) [TONVIFDeviceInfo](#) [Video stream of IP camera with audio of PC microphone or other audio capture device](#) [ExtraDLLPath](#) [GetLastErrorMessage](#) [IPCameraURL](#) [IsURLResponding](#) [IsURLVideoStreamAvailable](#) [OnONVIFDiscoveryCompletedNotification](#) [ONVIF_GetStr](#) [ONVIF_SetStr](#) [ONVIFCancelDiscovery](#) [ONVIFDeviceInfo](#) [ONVIFDiscoverCameras_IPRange](#) [Multicast](#) [ONVIFEnumCamerasDiscovered](#) [ONVIFPTZGetLimits](#) [ONVIFPTZGetPosition](#) [ONVIFPTZPreset](#) [ONVIFPTZSendAuxiliaryCommand](#) [ONVIFPTZSetPosition](#) [ONVIFPTZStartMove](#) [ONVIFPTZStopMove](#) [ONVIFSnapShot](#) [OpenURLAsync](#) [OpenURLAsyncStatus](#) [SendIPCameraCommand](#) [SetDatasteadFilterDllName](#) [SetIPCameraSetting](#)

Created with the Standard Edition of HelpNDoc: [Easily create iPhone documentation](#)

ONVIFEnumCamerasDiscovered

TVideoGrabber.ONVIFEnumCamerasDiscovered

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Methods](#)

Returns the ONVIF cameras discovered

Declaration

function ONVIFEnumCamerasDiscovered (CameraIndex: LongInt; **var** CameraType: **string**; **var** CameraONVIFUrl: **string**): Boolean;

bool __fastcall ONVIFDiscoverCameras_IPRange(System::UnicodeString First_IP, System::UnicodeString Last_IP, System::LongInt timeout_seconds_or_0_for_default);

Description

Retrieve the camera type and ONVIF service URL by specifying the index of the IP cameras in the 0..n-1 range.

The camera count is returned by each call of the [OnONVIFDiscoveryCompletedNotification](#) event.

Note that it is not required to know the camera count, just invoke ONVIFEnumCamerasDiscovered starting from CameraIndex = 0, by increasing it at each call.

The function call returns false when the CameraIndex = the number of cameras discovered.

See [ONVIF Cameras Discovery](#)

See Also

[IR Cut Filter of Axis cameras](#) [TDiscoveryCallbackStatus](#) [TOnONVIFDiscoveryCompletedNotification](#) [TONVIFDeviceInfo](#) [Video stream of IP camera with audio of PC microphone or other audio capture device](#) [ExtraDLLPath](#) [GetLastErrorMessage](#) [IPCameraURL](#) [IsURLResponding](#) [IsURLVideoStreamAvailable](#) [OnONVIFDiscoveryCompletedNotification](#) [ONVIF_GetStr](#) [ONVIF_SetStr](#) [ONVIFCancelDiscovery](#) [ONVIFDeviceInfo](#) [ONVIFDiscoverCameras_IPRange](#) [ONVIFDiscoverCameras_Multicast](#) [ONVIFPTZGetLimits](#) [ONVIFPTZGetPosition](#) [ONVIFPTZPreset](#) [ONVIFPTZSendAuxiliaryCommand](#) [ONVIFPTZSetPosition](#) [ONVIFPTZStartMove](#) [ONVIFPTZStopMove](#) [ONVIFSnapShot](#) [OpenURLAsync](#) [OpenURLAsyncStatus](#) [SendIPCameraCommand](#) [SetDatasteadFilterDllName](#) [SetIPCameraSetting](#)

Created with the Standard Edition of HelpNDoc: Effortlessly Edit and Export Markdown Documents

ONVIFPTZGetLimits

TVideoGrabber.ONVIFPTZGetLimits

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Methods](#)

Returns the limit PTZ positions

Declaration

function ONVIFPTZGetLimits (**out** Pan_Min: Double; **out** Pan_Max: Double; **out** Tilt_Min: Double; **out** Tilt_Max: Double; **out** Zoom_Min: Double; **out** Zoom_Max: Double): Boolean;

BOOL __ONVIFPTZGetLimits (void *TVGObject, **double***Pan_Min, **double***Pan_Max, **double***Tilt_Min, **double***Tilt_Max, **double***Zoom_Min, **double***Zoom_Max);

Description

Returns the limit PTZ positions (min and max) as double values

See Also

[IR Cut Filter of Axis cameras](#) [TDiscoveryCallbackStatus](#) [TOnONVIFDiscoveryCompletedNotification](#) [TONVIFDeviceInfo](#) [ExtraDLLPath](#) [GetLastErrorMessage](#) [OnONVIFDiscoveryCompletedNotification](#) [ONVIF_GetStr](#) [ONVIF_SetStr](#) [ONVIFCancelDiscovery](#) [ONVIFDeviceInfo](#) [ONVIFDiscoverCameras_IPRange](#) [ONVIFDiscoverCameras_Multicast](#) [ONVIFEnumCamerasDiscovered](#) [ONVIFPTZGetPosition](#) [ONVIFPTZPreset](#) [ONVIFPTZSendAuxiliaryCommand](#) [ONVIFPTZSetPosition](#) [ONVIFPTZStartMove](#) [ONVIFPTZStopMove](#) [ONVIFSnapShot](#) [OpenURLAsyncStatus](#) [SetDatasteadFilterDllName](#)

ONVIFPTZGetPosition

TVideoGrabber.ONVIFPTZGetPosition

[Prev](#)[Next](#)[TVideoGrabber](#) [Methods](#)

Returns the current PTZ position

Declaration

function ONVIFPTZGetPosition (**out** Pan: Double; **out** Tilt: Double; **out** Zoom: Double; **out** UTCTime: LargeInteger; **out** IsMoving: LongInt): Boolean;

BOOL _ONVIFPTZGetPosition (void *TVGObject, **double***Pan, **double***Tilt, **double***Zoom, __int64*UTCTime, **int***IsMoving);

Description

Returns the current PTZ position as double values

See Also

[IR Cut Filter of Axis cameras](#) [TDiscoveryCallbackStatus](#) [TOnONVIFDiscoveryCompletedNotification](#) [TONVIFDeviceInfo](#) [ExtraDLLPath](#) [GetLastErrorMessage](#) [OnONVIFDiscoveryCompletedNotification](#) [ONVIF_GetStr](#) [ONVIF_SetStr](#) [ONVIFCancelDiscovery](#) [ONVIFDeviceInfo](#) [ONVIFDiscoverCameras](#) [IPRange](#) [ONVIFDiscoverCameras_Multicast](#) [ONVIFEnumCamerasDiscovered](#) [ONVIFPTZGetLimits](#) [ONVIFPTZPreset](#) [ONVIFPTZSendAuxiliaryCommand](#) [ONVIFPTZSetPosition](#) [ONVIFPTZStartMove](#) [ONVIFPTZStopMove](#) [ONVIFSnapShot](#) [OpenURLAsyncStatus](#) [SetDatasteadFilterDllName](#)

ONVIFPTZPreset

TVideoGrabber.ONVIFPTZPreset

[Prev](#)[Next](#)[TVideoGrabber](#) [Methods](#)

Manage the presets

Declaration

function ONVIFPTZPreset (PresetAction: **string**; PresetName: **string**): Boolean;

BOOL _ONVIFPTZPreset (void *TVGObject, **const wchar_t** * *PresetAction, **const wchar_t** * *PresetName);

Description

Creates, remove or goto the specified preset, if the camera supports the presets

PresetAction can be "CREATE", "REMOVE" or "DELETE"

PresetName can be any choosen name, however some cameras support only their own predefined names of presets

To create a preset, move the camera to the desired position with ONVIFPTZStartContinuousMove or ONVIFPTZSetPosition, then invoke ONVIFPTZPreset ("CREATE", ThePresetName) to save the current position to the specified preset.

Then the camera can be positionned to this position when needed by invoking ONVIFPTZPreset ("GOTO", ThePresetName)

See Also

[IR Cut Filter of Axis cameras](#) [TDiscoveryCallbackStatus](#) [TOnONVIFDiscoveryCompletedNotification](#) [TONVIFDeviceInfo](#) [ExtraDLLPath](#) [GetLastErrorMessage](#) [OnONVIFDiscoveryCompletedNotification](#) [ONVIF_GetStr](#) [ONVIF_SetStr](#) [ONVIFCancelDiscovery](#) [ONVIFDeviceInfo](#) [ONVIFDiscoverCameras](#) [IPRange](#) [ONVIFDiscoverCameras_Multicast](#) [ONVIFEnumCamerasDiscovered](#) [ONVIFPTZGetLimits](#) [ONVIFPTZGetPosition](#) [ONVIFPTZSendAuxiliaryCommand](#) [ONVIFPTZSetPosition](#) [ONVIFPTZStartMove](#)

[ONVIFPTZStopMove](#) [ONVIFSnapShot](#) [OpenURLAsyncStatus](#) [SetDatasteadFilterDllName](#)

Created with the Standard Edition of HelpNDoc: [Easily create Qt Help files](#)

ONVIFPTZSendAuxiliaryCommand

TVideoGrabber.ONVIFPTZSendAuxiliaryCommand

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Send an auxiliary command

Declaration

function ONVIFPTZSendAuxiliaryCommand (AuxiliaryCommand: **string**): Boolean;

BOOL _ONVIFPTZSendAuxiliaryCommand (**void** *TVGObject, **const wchar_t** * *AuxiliaryCommand);

Description

Used to send a manufacturer-specific command to the camera.

Read the camera documentation to see what auxiliary commands the camera supports.

See Also

[IR Cut Filter of Axis cameras](#) [TDiscoveryCallbackStatus](#) [TOnONVIFDiscoveryCompletedNotification](#) [TONVIFDeviceInfo](#) [ExtraDLLPath](#) [GetLastErrorMessages](#) [OnONVIFDiscoveryCompletedNotification](#) [ONVIF_GetStr](#) [ONVIF_SetStr](#) [ONVIFCancelDiscovery](#) [ONVIFDeviceInfo](#) [ONVIFDiscoverCameras](#) [IPRange](#) [ONVIFDiscoverCameras](#) [Multicast](#) [ONVIFEnumCamerasDiscovered](#) [ONVIFPTZGetLimits](#) [ONVIFPTZGetPosition](#) [ONVIFPTZPreset](#) [ONVIFPTZSetPosition](#) [ONVIFPTZStartMove](#) [ONVIFPTZStopMove](#) [ONVIFSnapShot](#) [OpenURLAsyncStatus](#) [SetDatasteadFilterDllName](#)

Created with the Standard Edition of HelpNDoc: [Elevate Your Help Documentation with a Help Authoring Tool](#)

ONVIFPTZSetPosition

TVideoGrabber.ONVIFPTZSetPosition

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Set a new PTZ position

Declaration

function ONVIFPTZSetPosition (Pan: Double; Tilt: Double; Zoom: Double; SpeedRatio: Double; IsRelative: Boolean): Boolean;

BOOL _ONVIFPTZSetPosition (**void** *TVGObject, **double** Pan, **double** Tilt, **double** Zoom, **double** SpeedRatio, BOOL IsRelative);

Description

Set a new PTZ position.

The position values are usually in the -1.0 .. 1.0 range

The speed ratio is usually in the 0 .. 1.0 range

- if "IsRelative" is set to true the new position is relative to the current position
- if "IsRelative" is set to false the new position is an absolute position

See Also

[IR Cut Filter of Axis cameras](#) [TDiscoveryCallbackStatus](#) [TOnONVIFDiscoveryCompletedNotification](#) [TONVIFDeviceInfo](#) [ExtraDLLPath](#) [GetLastErrorMessages](#) [OnONVIFDiscoveryCompletedNotification](#) [ONVIF_GetStr](#) [ONVIF_SetStr](#) [ONVIFCancelDiscovery](#) [ONVIFDeviceInfo](#) [ONVIFDiscoverCameras](#) [IPRange](#) [ONVIFDiscoverCameras](#) [Multicast](#) [ONVIFEnumCamerasDiscovered](#) [ONVIFPTZGetLimits](#) [ONVIFPTZGetPosition](#) [ONVIFPTZPreset](#) [ONVIFPTZSendAuxiliaryCommand](#) [ONVIFPTZStartMove](#) [ONVIFPTZStopMove](#) [ONVIFSnapShot](#) [OpenURLAsyncStatus](#) [SetDatasteadFilterDllName](#)

ONVIFPTZStartMove

TVideoGrabber.ONVIFPTZStartMove

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Starts a continuous move

Declaration

function ONVIFPTZStartMove (PTZType: **string**; OppositeDirection: Boolean; SpeedRatio: Double; DurationMs: LongInt): Boolean;

BOOL _ONVIFPTZStartMove (**void** *TVGObject, **const wchar_t** * PTZType, BOOL OppositeDirection, **double** SpeedRatio, **int** DurationMs);

Description

Starts a continuous move of the specified PTZ

PTZType can be "Pan", "Tilt" or "Zoom"

SpeedRatio is usually in the 0 .. 1.0 range

- if the camera supports a limited move duration and a DurationMs value > 0, the camera will automatically stop the move when this duration is elapsed.
- if the camera does not support a limited move duration, the camera moves until ONVIFPTZStopMove is invoked.

See Also

[IR Cut Filter of Axis cameras](#) [TDiscoveryCallbackStatus](#) [TOnONVIFDiscoveryCompletedNotification](#) [TONVIFDeviceInfo](#) [ExtraDLLPath](#) [GetLastErrorMessage](#) [OnONVIFDiscoveryCompletedNotification](#) [ONVIF_GetStr](#) [ONVIF_SetStr](#) [ONVIFCancelDiscovery](#) [ONVIFDeviceInfo](#) [ONVIFDiscoverCameras](#) [IPRange](#) [ONVIFDiscoverCameras](#) [Multicast](#) [ONVIFEnumCamerasDiscovered](#) [ONVIFPTZGetLimits](#) [ONVIFPTZGetPosition](#) [ONVIFPTZPreset](#) [ONVIFPTZSendAuxiliaryCommand](#) [ONVIFPTZSetPosition](#) [ONVIFPTZStopMove](#) [ONVIFSnapShot](#) [OpenURLAsyncStatus](#) [SetDatasteadFilterDllName](#)

ONVIFPTZStopMove

TVideoGrabber.ONVIFPTZStopMove

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Stops moving

Declaration

function ONVIFPTZStopMove (PTZType: **string**): Boolean;

BOOL _ONVIFPTZStopMove (**void** *TVGObject, **const wchar_t** * PTZType);

Description

Stops moving the specified PTZ

PTZType can be "Pan", "Tilt" or "Zoom"

See Also

[IR Cut Filter of Axis cameras](#) [TDiscoveryCallbackStatus](#) [TOnONVIFDiscoveryCompletedNotification](#) [TONVIFDeviceInfo](#) [Video stream of IP camera with audio of PC microphone or other audio capture device](#) [ExtraDLLPath](#) [GetLastErrorMessage](#) [IPCameraURL](#) [IsURLResponding](#) [IsURLVideoStreamAvailable](#)

[OnONVIFDiscoveryCompletedNotification](#) [ONVIF_GetStr](#) [ONVIF_SetStr](#) [ONVIFCancelDiscovery](#) [ONVIFDeviceInfo](#) [ONVIFDiscoverCameras](#) [IPRange](#) [ONVIFDiscoverCameras](#) [Multicast](#) [ONVIFEnumCamerasDiscovered](#) [ONVIFPTZGetLimits](#) [ONVIFPTZGetPosition](#) [ONVIFPTZPreset](#) [ONVIFPTZSendAuxiliaryCommand](#) [ONVIFPTZSetPosition](#) [ONVIFPTZStartMove](#) [ONVIFSnapShot](#) [OpenURLAsync](#) [OpenURLAsyncStatus](#) [SendIPCameraCommand](#) [SetDatasteadFilterDllName](#) [SetIPCameraSetting](#)

Created with the Standard Edition of HelpNDoc: [Free EPub producer](#)

ONVIFSnapShot

TVideoGrabber.ONVIFSnapShot

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Capture a JPEG snapshot from an ONVIF IP camera

Declaration

function ONVIFSnapShot(OnRawVideoSampleCallbackEnabled: Boolean; SaveToFile: Boolean; FileName: string): Boolean;

BOOL ONVIFSnapShot(**bool** OnRawVideoSampleCallbackEnabled, **bool** SaveToFile, System::wchar_t *FileName);

Description

Used to capture a JPEG snapshot from an IP camera supporting the ONVIF protocol
bool OnRawVideoSampleCallbackEnabled: if true, the OnRawVideoSample event occurs, it is possible to get a direct access to the the buffer and buffer size parameters of the event, the buffer contains the JPEG image in memory.

bool SaveToFile: if true, the JPEG image is saved to the FileName specified

FileName: path/file name of the JPEG image to save

See Also

[IR Cut Filter of Axis cameras](#) [TDiscoveryCallbackStatus](#) [TOnONVIFDiscoveryCompletedNotification](#) [TONVIFDeviceInfo](#) [ExtraDLLPath](#) [GetLastErrorMessages](#) [OnONVIFDiscoveryCompletedNotification](#) [ONVIF_GetStr](#) [ONVIF_SetStr](#) [ONVIFCancelDiscovery](#) [ONVIFDeviceInfo](#) [ONVIFDiscoverCameras](#) [IPRange](#) [ONVIFDiscoverCameras](#) [Multicast](#) [ONVIFEnumCamerasDiscovered](#) [ONVIFPTZGetLimits](#) [ONVIFPTZGetPosition](#) [ONVIFPTZPreset](#) [ONVIFPTZSendAuxiliaryCommand](#) [ONVIFPTZSetPosition](#) [ONVIFPTZStartMove](#) [ONVIFPTZStopMove](#) [OpenURLAsyncStatus](#) [SetDatasteadFilterDllName](#)

Created with the Standard Edition of HelpNDoc: [Achieve Professional Documentation Results with a Help Authoring Tool](#)

OpenDVD

TVideoGrabber.OpenDVD

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Opens a DVD in playback mode

Declaration

function OpenDVD: LongBool;

BOOL __fastcall OpenDVD(void)

Function OpenDVD as Boolean

Description

Used to opens a DVD in playback mode

If [DVDTitle](#) = 0, the DVD menu appears when opening the DVD playback.

To start directly the playback of a DVD title, set DVDTitle with the number of the title.

E.g.:

```
VideoGrabber.PlayerFileName = "D:
VideoGrabber.DVDTitle = 0
VideoGrabber.OpenDVD ()
```

will open the DVD menu.

```
VideoGrabber.PlayerFileName = "E:
VideoGrabber.DVDTitle = 1
VideoGrabber.OpenDVD ()
```

will start playing directly the 1st title.

See Also

[DVDInfo](#) [DVDTitle](#) [PlayerFileName](#)

Created with the Standard Edition of HelpNDoc: [iPhone web sites made easy](#)

OpenPlayer

TVideoGrabber.OpenPlayer

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Opens a video clip.

Declaration

function OpenPlayer: Boolean;

bool __fastcall OpenPlayer(**void**)

Function OpenPlayer() As Boolean

Description

Opens the video clip specified by [PlayerFileName](#) .

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchronon](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Edit and Export Markdown Documents](#)

OpenPlayerAtFramePositions

TVideoGrabber.OpenPlayerAtFramePositions

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Opens a video clip within a specified start frame and stop frame.

Declaration

function OpenPlayerAtFramePositions (StartFrame: LargeInteger; StopFrame: LargeInteger; KeepBounds: Boolean; CloseAndReopenIfAlreadyOpened: Boolean): Boolean;

bool __fastcall OpenPlayerAtFramePositions(__int64 StartFrame, __int64 StopFrame, **bool** KeepBounds, **bool** CloseAndReopenIfAlreadyOpened)

Function OpenPlayerAtFramePositions(StartFrame As Double, StopFrame As Double, KeepBounds as Boolean, CloseAndReopenIfAlreadyOpened as Boolean)As Boolean

Description

Opens a video clip that will start and stop playing within boundaries between specified start and stop frames.

StartFrame:

opens the clip starting from this frame number (0 = from beginning)

StopFrame:

stops playing the clip at this frame number (0 = until the end)

KeepBounds:

- true:

- . when invoking StopPlayer the position returns to the StartFrame location
- . when the clip is paused, it is not possible to set a position before StartFrame or after StopFrame

- false:

- . when invoking StopPlayer the position returns at the frame number 1
- . when the clip is paused, it is possible to set a position before StartFrame or after StopFrame

CloseAndReopenIfAlreadyOpened:

- true:

if OpenPlayerAtFramePositions is invoked again the clip is reopened

- false:

if OpenPlayerAtFramePositions is invoked again the start and stop positions are updated to the specified positions

Notes:

- when the video clip has already been opened and the **CloseAndReopenIfAlreadyOpened** parameter is false, invoking this function lets you modify the playing boundaries, without closing/reopening the video clip
- to specify only a start frame and let the clip play until the end, set StopFrame = 0
- to specify only a stop frame and let the clip play from the beginning, set StartFrame = 0
- to reset the normal clip boundaries, set StopFrame = 0 and StartFrame = 0

E.g. OpenPlayerAtFramePositions (100, 900) plays the clip within the 100th and the 900th frames.

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#)

[PlayerForcedCodec](#)
[PlayerFrameCount](#)
[PlayerFramePosition](#)
[PlayerFrameRate](#)
[PlayerFrameStep](#)
[PlayerRefreshPausedDisplay](#)
[PlayerRefreshPausedDisplayFrameRate](#)
[PlayerSpeedRatio](#)
[PlayerTimePosition](#)
[PlayerTrackBar](#)
[PlayerTrackBarScale](#)
[PlayerTrackBarSynchronone](#)
[PlayerVideoCodec](#)
[RewindPlayer](#)
[RunPlayer](#)
[RunPlayerBackwards](#)
[ShowDialog](#)
[SourceStream](#)
[StopPlayer](#)
[SynchronizationRole](#)
[Synchronized](#)
[VideoHeight](#)
[PreferredAspectRatio](#)
[VideoPlayableWhileRecording](#)
[VideoWidth](#)
[PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Generate EPub eBooks with ease](#)

OpenPlayerAtTimePositions

TVideoGrabber.OpenPlayerAtTimePositions

[Prev](#)

[Next](#)

[TVideoGrabber](#)
[Methods](#)

Opens a video clip within a specified start time and stop time.

Declaration

function OpenPlayerAtTimePositions (StartTime: LargeInteger; StopTime: LargeInteger; KeepBounds: Boolean; CloseAndReopenIfAlreadyOpened: Boolean): Boolean;

bool __fastcall OpenPlayerAtTimePositions(__int64 StartTime, __int64 StopTime, **bool** KeepBounds, **bool** CloseAndReopenIfAlreadyOpened)

Function OpenPlayerAtTimePositions(StartTime As Double, StopTime As Double, CloseAndReopenIfAlreadyOpened as String) As Boolean

Description

Opens a video clip that will start and stop playing within boundaries between specified start and stop time. StartTime and StopTime are expressed in 100ns units (e.g. 2.5 seconds = 25000000).

StartTime:

opens the clip starting from this frame time (0 = from beginning)

StopTime:

stops playing the clip at this frame time (0 = until the end)

KeepBounds:

- true:

- . when invoking StopPlayer the position returns to the StartTime position
- . when the clip is paused, it is not possible to set a position before StartTime or after StopTime

- false:

- . when invoking StopPlayer the position returns at the frame time 0
- . when the clip is paused, it is possible to set a position before StartTime or after StopTime

CloseAndReopenIfAlreadyOpened:

- true:

if OpenPlayerAtTimePositions is invoked again the clip is reopened

- false:

if OpenPlayerAtTimePositions is invoked again the start and stop positions are updated to the specified positions

Notes:

- when the video clip has already been opened and the **CloseAndReopenIfAlreadyOpened** parameter is false, invoking this function lets you modify the playing boundaries, without closing/reopening the video clip
- to specify only a start time and let the clip play until the end, set StopTime = 0
- to specify only a stop time and let the clip play from the beginning, set StartTime = 0
- to reset the normal clip boundaries, set StopTime = 0 and StartTime = 0

E.g. `OpenPlayerAtFramePositions (30000000, 60000000)` plays the clip within 3 seconds and the 6 seconds (starting from the beginning).

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip](#) [Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchron](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Experience a User-Friendly Interface with HelpNDoc's Documentation Tool](#)

OpenURLAsyncStatus

TVideoGrabber.OpenURLAsyncStatus

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

returns the status of the URL currently being opened asynchronously

Declaration

function OpenURLAsyncStatus: TOpenURLAsyncStatus;

TOpenURLAsyncStatus __**fastcall** OpenURLAsyncStatus();

Description

meaning of the values here: [TOpenURLAsyncStatus](#)

See Also

[IR Cut Filter of Axis cameras](#) [TDiscoveryCallbackStatus](#) [TOnONVIFDiscoveryCompletedNotification](#) [TONVIFDeviceInfo](#) [Video stream of IP camera with audio of PC microphone or other audio capture device](#) [ExtraDLLPath](#) [GetLastErrorMessages](#) [IPCameraURL](#) [IsURLResponding](#) [IsURLVideoStreamAvailable](#) [OnONVIFDiscoveryCompletedNotification](#) [ONVIF](#) [GetStr](#) [ONVIF](#) [SetStr](#) [ONVIFCancelDiscovery](#) [ONVIFDeviceInfo](#) [ONVIFDiscoverCameras](#) [IPRange](#) [ONVIFDiscoverCameras](#) [Multicast](#) [ONVIFEnumCamerasDiscovered](#) [ONVIFPTZGetLimits](#) [ONVIFPTZGetPosition](#) [ONVIFPTZPreset](#) [ONVIFPTZSendAuxiliaryCommand](#) [ONVIFPTZSetPosition](#) [ONVIFPTZStartMove](#) [ONVIFPTZStopMove](#) [ONVIFSnapshot](#) [OpenURLAsync](#) [SendIPCameraCommand](#) [SetDatasteadFilterDllName](#) [SetIPCameraSetting](#)

Created with the Standard Edition of HelpNDoc: [HelpNDoc's Project Analyzer: Incredible documentation assistant](#)

PausePlayer

TVideoGrabber.PausePlayer

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Pauses a video clip currently playing back.

Declaration

procedure PausePlayer;

void __fastcall PausePlayer(**void**)

Sub PausePlayer()

Description

Used to pause a [PlayerFileName](#) video clip currently playing back.
Call [RunPlayer](#) to resume a paused clip.

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchronise](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRun](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Easily create EPub books](#)

PausePreview

TVideoGrabber.PausePreview

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Pauses running preview.

Declaration

function PausePreview: Boolean;

bool __fastcall PausePreview(**void**)

Function PausePreview() As Boolean

Description

Used to pause preview, previously started with [StartPreview](#).

Can be invoked also during recording (started with [StartRecording](#)) to pause the recording AND the preview. Then invoke [ResumePreview](#) to go on recording AND previewing (see also [Pause/resume during recording](#))

See Also

[TOnFrameProgress](#) [TOnResizeVideo](#) [AutoRefreshPreview](#) [BackgroundColor](#) [Display](#) [AutoSize](#) [Display_FullScreen](#) [Display_SetLocation](#) [Display_VideoPortEnabled](#) [ImageRatio](#) [InFrameProgressEvent](#) [IsVideoInterlaced](#) [IsVideoPortAvailable](#) [OnFrameProgress](#) [OnKeyPress](#) [OnPreviewStarted](#) [OnResizeVideo](#) [PreviewZoomSize](#) [ResumePreview](#) [StartPreview](#) [StartSynchronized](#) [StopPreview](#) [Synchronized](#) [UseNearestVideoSize](#) [VideoHeight](#) [VideoRenderer](#) [VideoRendererPriority](#) [VideoWidth](#) [Visible](#)

Created with the Standard Edition of HelpNDoc: [Easy EBook and documentation generator](#)

PauseRecording

TVideoGrabber.PauseRecording

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Pauses the current recording.

Declaration

function PauseRecording: Boolean;

bool __fastcall PauseRecording(**void**)

Function PauseRecording() As Boolean

Description

Used to pause the current recording, previously started by [StartRecording](#) .

[RecordingCanPause](#) must be enabled to use this feature.

Returns true if the recording has been successfully paused.

recording can be resumed with [ResumeRecording](#) .

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last](#) [Recording](#) [FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: Effortlessly optimize your documentation website for search engines

PlayerFrameStep

TVideoGrabber.PlayerFrameStep

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Steps the video clip n frames forward

Declaration

function PlayerFrameStep (FrameCount: LongInt): Boolean;

bool __fastcall PlayerFrameStep(**int** FrameCount);

Function PlayerFrameStep as Bool

Description

Used to step n frames forward when the video clip is paused.

The FrameCount parameter specifies the number of frames to skip.

- if FrameCount is 1, the graph steps forward one frame.

- if dwFrames is a number n greater than 1, the graph skips n - 1 frames and shows the nth frame.

See Also

[Player features](#)
[TOnPlayerStateChanged](#)
[TPlayerState](#)
[TOnPlayerBufferingData](#)
[AudioChannelRenderMode](#)
[AudioStreamNumber](#)
[AutoStartPlayer](#)
[AVIDuration](#)
[AVIHeaderInfo](#)
[AVIInfo](#)
[AVIInfo2](#)
[ClosePlayer](#)
[FastForwardPlayer](#)
[IsPlayerAudioStreamAvailable](#)
[IsPlayerVideoStreamAvailable](#)
[Last_Clip_Played](#)
[MP4NeedsReindexing](#)
[OnPlayerBufferingData](#)
[OnPlayerEndOfStream](#)
[OnPlayerStateChanged](#)
[OnPlayerUpdateTrackbarPosition](#)
[OpenPlayer](#)
[OpenPlayerAtFramePositions](#)
[OpenPlayerAtTimePositions](#)
[PausePlayer](#)
[PlayerAudioCodec](#)
[PlayerAudioRendering](#)
[PlayerDuration](#)
[PlayerDVSize](#)
[PlayerFastSeekSpeedRatio](#)
[PlayerFileName](#)
[PlayerForcedCodec](#)
[PlayerFrameCount](#)
[PlayerFramePosition](#)
[PlayerFrameRate](#)
[PlayerRefreshPausedDisplay](#)
[PlayerRefreshPausedDisplayFrameRate](#)
[PlayerSpeedRatio](#)
[PlayerTimePosition](#)
[PlayerTrackBar](#)
[PlayerTrackBarScale](#)
[PlayerTrackBarSynchronise](#)
[PlayerVideoCodec](#)
[RewindPlayer](#)
[RunPlayer](#)
[RunPlayerBackwards](#)
[ShowDialog](#)
[SourceStream](#)
[StopPlayer](#)
[SynchronizationRole](#)
[Synchronized](#)
[VideoHeight](#)
[PreferredAspectRatio](#)
[VideoPlayableWhileRecording](#)
[VideoWidth](#)
[PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly upgrade your WinHelp HLP help files to CHM with HelpNDoc](#)

Playlist

TVideoGrabber.Playlist

[Prev](#)

[Next](#)

[TVideoGrabber](#)
[Methods](#)

Used for building and playing playlists.

Declaration

function Playlist (PlaylistAction: TPlaylist; VideoClip: **String**): Boolean;

int __fastcall PointGreyConfig(TPointGreyConfig ConfigType, **unsigned** &PGRActionValue, **unsigned** ulRegister, **unsigned** ulMode, **unsigned** ulLeft, **unsigned** ulTop, **unsigned** ulWidth, **unsigned** ulHeight, **unsigned** ulPercentage, **unsigned** ulFormat);

function Playlist (PlaylistAction as TxPlaylist, VideoClip as string)

Description

This function is used for building and playing playlists in the player.

- the PlaylistAction parameter specifies the action
- the VideoClip parameter specifies the video clip on which the action is applied.

See the "[Using the playlist](#)" chapter in the **Player** section of the user guide.

See Also

[TPlaylist](#)
[Video formats](#)
[GetPlaylist](#)
[IsPlaylistActive](#)
[OnPlayerEndOfPlaylist](#)
[PlaylistIndex](#)

Created with the Standard Edition of HelpNDoc: [Revolutionize Your Documentation Output with a Help Authoring Tool](#)

PointGreyConfig

TVideoGrabber.PointGreyConfig

[Prev](#)

[Next](#)

[TVideoGrabber](#)
[Methods](#)

Used to configure a PointGrey camera

Declaration

function PointGreyConfig (ConfigType: TPointGreyConfig; **var** PGRActionValue: LongWord; ulRegister: LongWord; ulMode: LongWord; ulLeft: LongWord; ulTop: LongWord; ulWidth: LongWord; ulHeight: LongWord; ulPercentage: LongWord; ulFormat: LongWord): LongInt;

int __fastcall PointGreyConfig(TPointGreyConfig ConfigType, **unsigned** &PGRActionValue, **unsigned** ulRegister, **unsigned** ulMode, **unsigned** ulLeft, **unsigned** ulTop, **unsigned** ulWidth, **unsigned** ulHeight, **unsigned** ulPercentage, **unsigned** ulFormat);

function PointGreyConfig (ConfigType as TPointGreyConfig, ByRef PGRActionValue as Long, ulRegister as Long, ulMode as Long, ulLeft as Long, ulTop as Long, ulWidth as Long, ulHeight as Long, ulPercentage as Long, ulFormat as LongWord) as Long

Description

Used to configure PointGrey cameras using the PointGrey's "FlyStream" driver

The 1st PointGreyConfig parameter (type [TPointGreyConfig](#)) specify the type of action applied. The use of the other parameters depend on the PointGreyConfig parameter:

pgr_SetRegister

sets the register specified in **ulRegister** with the value specified in **PGRActionValue**
the other parameters are ignored

pgr_GetRegister

get the register specified in **ulRegister**, the value is returned by **PGRActionValue**
the other parameters are ignored

pgr_SetBufferSize

sets the buffer size with the value specified in **PGRActionValue**
the other parameters are ignored

pgr_GetBufferSize

gets the buffer size, the value is returned by **PGRActionValue**
the other parameters are ignored

pgr_SetFormat

sets the format with the values specified in **ulMode**, **ulLeft**, **ulTop**, **ulWidth**, **ulHeight**, **ulPercentage**, **ulFormat**
the other parameters are ignored

Created with the Standard Edition of HelpNDoc: [Eliminate the Struggles of Documentation with a Help Authoring Tool](#)

PutMiscDeviceControl

TVideoGrabber.PutMiscDeviceControl

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Used to set specific values or states on some video capture devices.

Declaration

function PutMiscDeviceControl (MiscDeviceControl: TMiscDeviceControl; **Index**: LongInt; **Value**: LongInt): Boolean;

bool __fastcall PutMiscDeviceControl(TMiscDeviceControl MiscDeviceControl, **int** Index, **int** Value);

Function PutMiscDeviceControl (MiscDeviceControl as TMiscDeviceControl, Index as Long, Value as Long) as Boolean

Description

Used to set specific values or states on some video capture devices that support them, e.g. the GPIO, VPD, etc...

The use of this function is explained in the [Miscellaneous device control](#) chapter.

See Also

Created with the Standard Edition of HelpNDoc: [Generate Kindle eBooks with ease](#)

RecordingKBytesWrittenToDisk**TVideoGrabber.RecordingKBytesWrittenToDisk**[Prev](#)[Next](#)
[TVideoGrabber](#) [Methods](#)

Returns the file size of the current recording.

Declaration

function RecordingKBytesWrittenToDisk: LongWord;

unsigned __fastcall RecordingKBytesWrittenToDisk(**void**);

function RecordingKBytesWrittenToDisk as Long

Description

While recording, this function returns the file size in KB of the file currently beeing recorded.

This function reports the current (growing) file size in KB while the recording is running.

Invoke this function periodically (e.g. every 10 seconds) to retrieve on the fly the current file size of the video clip being recorded.

When the recording ends, invoke [RecordingKBytesWrittenToDisk](#) from the [OnRecordingCompleted](#) event to get the final file size of the recorded clip.

(note: due to the recording file buffer, the real size at a given time is always a bit larger than the size reported by this function)

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last_Recording_FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Productivity with a Help Authoring Tool](#)

RecordToNewFileNow**TVideoGrabber.RecordToNewFileNow**[Prev](#)[Next](#)
[TVideoGrabber](#) [Methods](#)

Creates a new file during recording.

Declaration

function RecordToNewFileNow (NewRecordingFileName: **string**; ResetStreamTime: Boolean): Boolean;

bool __fastcall RecordToNewFileNow (wchar_t *NewRecordingFileName, **bool** ResetStreamTime);

Function RecordToNewFileNow((NewRecordingFileName as String, ResetStreamTime as Boolean) As Boolean

Description

During recording, invoking **RecordToNewFileNow** closes the current recording file, and creates a new recording file according to the [RecordingFileName](#) property.

- if [RecordingFileName](#) is empty, a new file name is generated automatically according to the [StoragePath](#) , [AutoFileName](#) and [AutoFilePrefix](#) .
- if [RecordingFileName](#) specifies a file name, this file name will be used to create the new file.

Created with the Standard Edition of HelpNDoc: [Make Documentation Review a Breeze with HelpNDoc's Advanced Project Analyzer](#)

RefreshDevicesAndCompressorsLists

TVideoGrabber.RefreshDevicesAndCompressorsLists

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Refreshes the devices and compressors list.

Declaration

procedure RefreshDevicesAndCompressorsLists;

void __fastcall RefreshDevicesAndCompressorsLists (**void**)

Sub RefreshDevicesAndCompressorsLists

Description

Used mainly to refresh the [VideoCompressors](#) and [AudioCompressors](#) list, e.g. after installing/uninstalling video compressors or audio compressors.

Calling this procedure refreshes also the [AudioDevices](#) and [VideoDevices](#) lists. However usually it is not useful to invoke this procedure for video capture devices or audio capture devices, because when connecting/disconnecting them, the corresponding lists are refreshed automatically and the [OnDeviceArrivalOrRemoval](#) event occurs to notify the list have been updated.

See Also

[Recording methods and properties](#) [WDM drivers](#) [TCompressionType](#) [TOnDeviceArrivalOrRemoval](#) [TOnVideoCompressionSettings](#) [AudioCompressor](#) [AudioCompressorIndex](#) [AudioCompressorName](#) [AudioCompressors](#) [AudioCompressorsCount](#) [AutoConnectRelatedPins](#) [CompressionMode](#) [CompressionType](#) [CurrentFrameRate](#) [DeliveredFrames](#) [DroppedFrames](#) [FixFlickerOrBlackCapture](#) [GetVideoCompressionSettings](#) [IsDVDevice](#) [IsVideoControlAvailable](#) [IsVideoCrossbarAvailable](#) [IsVideoDeviceConnected](#) [IsWDMVideoDriver](#) [OnDeviceArrivalOrRemoval](#) [OnDeviceLost](#) [OnNoVideoDevices](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [OnVideoCompressionSettings](#) [OnVideoDeviceSelected](#) [ResetVideoDeviceSettings](#) [SaveCompressorSettingsToDataString](#) [SetVideoCompressionDefaults](#) [SetVideoCompressionSettings](#) [ShowDialog](#) [VideoCompression](#) [KeyFrameRate](#) [VideoCompression_PFramesPerKeyFrame](#) [VideoCompression_Quality](#) [VideoCompression_WindowSize](#) [VideoCompressor](#) [VideoCompressorIndex](#) [VideoCompressorName](#) [VideoCompressors](#) [VideoCompressorsCount](#) [VideoDevice](#) [VideoDeviceIndex](#) [VideoDeviceIndexFromId](#) [VideoDeviceName](#) [VideoDevices](#) [VideoDevicesCount](#) [VideoDevicesId](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Quickly and Easily Convert Your Word Document to an ePub or Kindle eBook](#)

RefreshPlayerOverlays

TVideoGrabber.RefreshPlayerOverlays

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Methods](#)

must be invoked to refresh the graphic or text overlays when the video clip is paused

Declaration

procedure RefreshPlayerOverlays;

void __fastcall RefreshPlayerOverlays(**void**)

Sub RefreshPlayerOverlays()

Description

Invoking this method when the video clip is paused raises the OnFrameOverlayUsing... events and redraw the current video frame.

Invoke it if you are e.g. performing free hand drawing with the mouse over the video frames while the clip is paused, and you want the drawing to be applied.

Look at the "free hand drawing" sample code in the OnFrameOverlayUsingDC event of the MainDemo project.

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#)
[AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#)
[MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#)
[OnMouseWheel](#) [OverlayAfterTransform](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#)
[SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#)
[SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#)
[SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#)
[SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#)
[SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#)
[SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#)
[SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#)
[SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#)
[SetImageOverlayFromTImage2](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#)
[SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#)
[SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#)
[SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#)
[SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#)
[SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#)
[ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#)
[TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#)
[TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#)
[TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#)
[TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#)
[TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Generate Kindle eBooks with ease](#)

ResetVideoDeviceSettings

TVideoGrabber.ResetVideoDeviceSettings

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Methods](#)

Resets the current settings of the video capture devices

Declaration

function ResetVideoDeviceSettings: Boolean;

bool __fastcall ResetVideoDeviceSettings(**void**)

Function ResetVideoDeviceSettings() As Boolean

Description

Used to reset the settings of the current video capture device.

Just invoke this function one time while the component is inactive, then the default settings will be used the next time the preview or recording is started.

See Also

[WDM drivers](#) [AutoConnectRelatedPins](#) [CurrentFrameRate](#) [DeliveredFrames](#) [DroppedFrames](#) [FixFlickerOrBlackCapture](#) [IsDVDevice](#) [IsVideoControlAvailable](#) [IsVideoCrossbarAvailable](#) [IsVideoDeviceConnected](#) [IsWDMVideoDriver](#) [OnNoVideoDevices](#) [OnVideoDeviceSelected](#) [RefreshDevicesAndCompressorsLists](#) [ShowDialog](#) [VideoDevice](#) [VideoDeviceIndex](#) [VideoDeviceIndexFromId](#) [VideoDeviceName](#) [VideoDevices](#) [VideoDevicesCount](#) [VideoDevicesId](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: Powerful and User-Friendly Help Authoring Tool for Markdown Documents

ResumePreview**TVideoGrabber.ResumePreview**
[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Resumes paused preview.

Declaration

function ResumePreview: Boolean;

bool __fastcall ResumePreview(**void**)

Function ResumePreview() As Boolean

Description

Used to resume preview, previously paused with [PausePreview](#).

Can be invoked also during a recording (started with [StartRecording](#)) to resume a recording **AND** a preview if they have been paused by invoking [PausePreview](#). (see also [Pause/resume during recording](#))

See Also

[TOnFrameProgress](#) [TOnResizeVideo](#) [AutoRefreshPreview](#) [BackgroundColor](#) [Display](#) [AutoSize](#) [Display_FullScreen](#) [Display_SetLocation](#) [Display_VideoPortEnabled](#) [ImageRatio](#) [InFrameProgressEvent](#) [IsVideoInterlaced](#) [IsVideoPortAvailable](#) [OnFrameProgress](#) [OnKeyPress](#) [OnPreviewStarted](#) [OnResizeVideo](#) [PausePreview](#) [PreviewZoomSize](#) [StartPreview](#) [StartSynchronized](#) [StopPreview](#) [Synchronized](#) [UseNearestVideoSize](#) [VideoHeight](#) [VideoRenderer](#) [VideoRendererPriority](#) [VideoWidth](#) [Visible](#)

Created with the Standard Edition of HelpNDoc: Create Professional CHM Help Files with HelpNDoc's Easy-to-Use Tool

ResumeRecording**TVideoGrabber.ResumeRecording**
[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

ber

Resumes an recording, currently in a paused state.

Declaration

function ResumeRecording: Boolean;

bool __fastcall ResumeRecording(**void**)

Function ResumeRecording() As Boolean

Description

Used to resume an recording, currently paused by a [PauseRecording](#) call.

[RecordingCanPause](#) must be enabled to use this feature.

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last_Recording_FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: Upgrade your help files and your workflow with HelpNDoc's WinHelp HLP to CHM conversion

RetrieveInitialXYAfterRotation

TVideoGrabber.RetrieveInitialXYAfterRotation

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Retrieves the original frame coordinates after a rotation.

Declaration

procedure RetrieveInitialXYAfterRotation (X, Y: LongInt; var OriginalX: LongInt; var OriginalY: LongInt);

void __fastcall RetrieveInitialXYAfterRotation (**int** X, **int** Y, **int** &OriginalX, **int** &OriginalY)

Sub RetrieveInitialXYAfterRotation (X as Long, Y as Long, OriginalX as Variant, OriginalY as Variant)

Description

Used to retrieve the initial frame coordinates after a [VideoProcessing_Rotation](#) as been applied and [OverlayAfterTransform](#) is enabled.

This procedure is used mainly to make a drawing from the [OnFrameOverlayUsingDC](#) event that is not rotated when [OverlayAfterTransform](#) is enabled.

E.g.:

```
procedure TfrmMainForm.VideoGrabberFrameOverlayUsingDC(Sender: TObject; Dc: HDC; FrameNum
```



```

var
  Canvas: TCanvas;
  SrcX, SrcY: LongInt;
  DestX, DestY: LongInt;
begin
  Canvas := TCanvas.Create;
  Canvas.Handle := Dc;
  Canvas.Pen.Color := clWhite;
  VideoGrabber.RetrieveInitialXYAfterRotation (100, 100, SrcX, SrcY);
  VideoGrabber.RetrieveInitialXYAfterRotation (100, 200, DestX, DestY);
  Canvas.MoveTo (SrcX, SrcY);
  Canvas.Pixels[SrcX, SrcY] := clWhite;
  Canvas.LineTo (DestX, DestY);
  Canvas.Free;
end;

```

See Also

[TVideoDeinterlacing](#) [MP4NeedsReindexing](#) [OnFrameBitmap](#) [VideoProcessing_Brightness](#)
[VideoProcessing_Contrast](#) [VideoProcessing_Deinterlacing](#) [VideoProcessing_FlipHorizontal](#)
[VideoProcessing_FlipVertical](#) [VideoProcessing_GrayScale](#) [VideoProcessing_Hue](#)
[VideoProcessing_InvertColors](#) [VideoProcessing_Pixellization](#) [VideoProcessing_Rotation](#)
[VideoProcessing_RotationCustomAngle](#) [VideoProcessing_Saturation](#) [TVideoRotation](#)

Created with the Standard Edition of HelpNDoc: Import and export Markdown documents

RewindPlayer

TVideoGrabber.RewindPlayer

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)
[ber](#)

Plays a clip backwards.

Declaration

procedure RewindPlayer;

void __fastcall RewindPlayer(**void**)

Sub RewindPlayer()

Description

Used to play a video clip backwards.

The backwards speed is the normal speed x [PlayerFastSeekRatio](#) .

Restrictions:

- this feature is available only with seekable (indexed) AVI clips.
- according to the clip index and the disk access, playing backwards may be jerky.

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#)
[AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#)
[FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#)
[MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#)
[OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#)
[PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#)
[PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#)
[PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#)
[PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchron](#)
[PlayerVideoCodec](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#)

[SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: Make Your PDFs More Secure with Encryption and Password Protection

RunPlayer

TVideoGrabber.RunPlayer

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Starts playing a video clip.

Declaration

procedure RunPlayer;

void __fastcall RunPlayer(**void**)

Sub RunPlayer()

Description

Used to play a [PlayerFileName](#) video clip opened by [OpenPlayer](#) .

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchronon](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: Revolutionize Your CHM Help File Output with HelpNDoc

RunPlayerBackwards

TVideoGrabber.RunPlayerBackwards

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Starts playing a video clip backwards.

Declaration

procedure RunPlayerBackwards;

void __fastcall RunPlayerBackwards(**void**)

Sub RunPlayerBackwards()

Description

Used to play backwards a [PlayerFileName](#) video clip opened by [OpenPlayer](#) .

Restrictions:

- this feature is available only with seekable (indexed) AVI clips.
- according to the clip index and the disk access, playing backwards may be jerky.

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip](#) [Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchron](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Free iPhone documentation generator](#)

SaveCompressorSettingsToDataString**TVideoGrabber.SaveCompressorSettingsToDataString**[Prev](#)[Next](#)

[TVideoGrabber](#) [Methods](#)

Saves the compressor settings to a data string

Declaration

function SaveCompressorSettingsToDataString (IsVideoCompressor: boolean; CompressorIndex: LongInt): string;

wchar_t * __fastcall SaveCompressorSettingsToDataString (bool IsVideoCompressor, int CompressorIndex: LongInt);

function SaveCompressorSettingsToDataString (IsVideoCompressor as Boolean, CompressorIndex as Long) As String

Description

Used to save the current compressor settings to a data string.

The compressor settings depend on the current [VideoCompressor](#) and [RecordingMethod](#), so be sure to first select the RecordingMethod() and VideoCompressor before saving its settings.

The **IsVideoCompressor** parameter must be set to **true** for a video compressor, or **false** for an audio compressor.

The Compressor parameter index must be the [VideoCompressor](#) or [AudioCompressor](#) index of the related compressor in the [VideoCompressors](#) or [AudioCompressors](#) lists.

See [Saving and restoring compressor settings programmatically](#) for sample code.

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TCompressionType](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TOnVideoCompressionSettings](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioCompressor](#) [AudioCompressorIndex](#) [AudioCompressorName](#) [AudioCompressors](#) [AudioCompressorsCount](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [CompressionMode](#) [CompressionType](#) [Encoder](#) [SetInt](#) [GetVideoCompressionSettings](#) [HoldRecording](#) [IsRecordingPaused](#) [Last_Recording](#) [FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [OnVideoCompressionSettings](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#)

[PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#)
[RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#)
[RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#)
[RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [RefreshDevicesAndCompressorsLists](#)
[ResumeRecording](#) [SetMultiplexerFilterByName](#) [SetVideoCompressionDefaults](#)
[SetVideoCompressionSettings](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#)
[StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoCompression](#) [KeyFrameRate](#)
[VideoCompression](#) [PFramesPerKeyFrame](#) [VideoCompression](#) [Quality](#) [VideoCompression](#) [WindowSize](#)
[VideoCompressor](#) [VideoCompressorIndex](#) [VideoCompressorName](#) [VideoCompressors](#)
[VideoCompressorsCount](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [Easily create Qt Help files](#)

SaveCompressorSettingsToTextFile

TVideoGrabber.SaveCompressorSettingsToTextFile

[Prev](#)
[Next](#)

[TVideoGrabber](#)
[Methods](#)

Saves the current compressor settings to a text file.

Declaration

function SaveCompressorSettingsToTextFile (IsVideoCompressor: boolean; CompressorIndex: LongInt; FileName: **string**): Boolean;

bool __fastcall SaveCompressorSettingsToTextFile (**bool** IsVideoCompressor, **int** CompressorIndex: LongInt, wchar_t *FileName);

function SaveCompressorSettingsToTextFile (IsVideoCompressor as Boolean, CompressorIndex as Long, FileName as String) As Boolean

Description

Used to save the current compressor settings to a text file.

The compressor settings depend on the current [VideoCompressor](#) and [RecordingMethod](#), so be sure to first select the RecordingMethod() and VideoCompressor before saving its settings.

The **IsVideoCompressor** parameter must be set to **true** for a video compressor, or **false** for an audio compressor.

The Compressor parameter index must be the [VideoCompressor](#) or [AudioCompressor](#) index of the related compressor in the [VideoCompressors](#) or [AudioCompressors](#) lists.

See [Saving and restoring compressor settings programmatically](#) for sample code.

Created with the Standard Edition of HelpNDoc: [Make Help Documentation a Breeze with a Help Authoring Tool](#)

ScreenRecordingUsingCoordinates

TVideoGrabber.ScreenRecordingUsingCoordinates

[Prev](#)
[Next](#)

[TVideoGrabber](#)
[Methods](#)

Limits the screen recording to an area

Declaration

function ScreenRecordingUsingCoordinates (Enabled: Boolean; scLeft, scTop, scWidth, scHeight: LongInt): Boolean;

bool __fastcall ScreenRecordingUsingCoordinates(**bool** Enabled, **int** scLeft, **int** scTop, **int** scWidth, **int**

scHeight);

Function ScreenRecordingUsingCoordinates (Enabled as Boolean, scTop, scWidth, scHeight as Long) as Boolean

Description

Used to limit the screen recording to a part of the screen.

Enabled : activates the limitation (if disabled, the whole screen is recorded)

scLeft : left coordinate of the screen area to record

scTop : top coordinate of the screen area to record

scWidth : width of the screen area to record

scHeight : height of the screen area to record

Remarks:

- the scLeft and scTop location may be modified dynamically while previewing or recording
- this feature is inactive if the recording of a window has been activated with [SetWindowRecordingByName](#) or [SetWindowRecordingByHandle](#)

See Also

[TOnEnumerateWindows](#) [EnumerateWindows](#) [ScreenRecordingLayeredWindows](#) [ScreenRecordingMonitor](#) [ScreenRecordingNonVisibleWindows](#) [ScreenRecordingWithCursor](#) [SetWindowRecordingByHandle](#) [SetWindowRecordingByName](#)

Created with the Standard Edition of HelpNDoc: [Single source CHM, PDF, DOC and HTML Help creation](#)

SendCameraCommand

TVideoGrabber.SendCameraCommand

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Sends Pan and Tilt commands to the current camera

Declaration

function SendCameraCommand (Pan: LongInt; Tilt: LongInt; Relative: Boolean): Boolean;

bool __fastcall SendCameraCommand (int Pan, int Tilt, bool Relative);

function SendCameraCommand Pan as Long, Tilt as Long, Relative as Boolean)

Description

Used to send Pan and Tilt commands to the camera currently selected as [VideoDevice](#).

Pass as Pan and Tilt the number of steps to perform. Use:

- . negative values to pan left
- . positive values to pan right
- . negative values to tilt down
- . positive values to tilt up
- . if Relative is true, the steps are applied to the current position of the camera
- . if Relative is false, the steps are applied from the origin position of the camera.

E.g.:

- one step left, from the current position:

VideoGrabber.SendCameraCommand (-1, 0, true);

- two steps right, from the current position:

VideoGrabber.SendCameraCommand (2, 0, true);

- one step up, from the current position:

```
VideoGrabber.SendCameraCommand (0, 1, true);  
end;
```

- one step down, from the origin:

```
VideoGrabber.SendCameraCommand (0, -1, false);
```

- 2 steps right and 1 step up from the current position

```
VideoGrabber.SendCameraCommand (2, 1, true);
```

- back to the origin:

```
VideoGrabber.SendCameraCommand (0, 0, false);
```

See Also

[TCameraControl](#) [CameraControlAuto](#) [CameraControlDefault](#) [CameraControlMax](#) [CameraControlMin](#)
[CameraControlSettings](#) [CameraControlStep](#) [GetCameraExposure](#) [GetCameraExposureAsString](#)
[IsCameraControlAvailable](#) [IsCameraControlSettingAvailable](#) [SetCameraControl](#) [SetCameraExposure](#)

Created with the Standard Edition of HelpNDoc: [Elevate Your Documentation Process with HelpNDoc's Advanced Features](#)

SendDVCommand

TVideoGrabber.SendDVCommand

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Sends a command to the VCR.

Declaration

```
function SendDVCommand(DVCommand: TDVCommand): Boolean;
```

```
bool __fastcall SendDVCommand(TDVCommand DVCommand)
```

Function SendDVCommand(DVCommand As TxDVCommand) As Boolean

Description

Sends the specified [TDVCommand](#) transport command to the DV device.
Returns true upon success.

See Also

[TDVSize](#) [TDVVideoFormat](#) [TDVVideoStandard](#) [TOnDVCommandCompleted](#) [DVDateTimeEnabled](#)
[DVDiscontinuityMinimumInterval](#) [DVRecordingInNativeFormatSeparatesStreams](#) [DVReduceFrameRate](#)
[DVRgb219](#) [DVTimeCodeEnabled](#) [IsDigitalVideoIn](#) [IsDVCommandAvailable](#) [IsTimeCodeReaderAvailable](#)
[OnDVCommandCompleted](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your CHM Help File Capabilities with HelpNDoc](#)

SendImageToVideoFromBitmaps

TVideoGrabber.SendImageToVideoFromBitmaps

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Used to pass the image or bitmap handle to the VideoFromJPEGsOrBitmaps function

Declaration

```
function SendImageToVideoFromBitmaps (ImageFilePath: string; BitmapHandle: LongInt;  
CanFreeBitmapHandle: Boolean; EndOfData: Boolean): boolean;
```

```
bool __fastcall SendImageToVideoFromBitmaps(wchar_t *ImagePath, int BitmapHandle, bool CanFreeBitmapHandle, bool EndOfData);
```

function SendImageToVideoFromBitmaps (ImagePath as string, BitmapHandle as Long, CanFreeBitmapHandle as Boolean, EndOfData as Boolean) as Boolean

Description

When [VideoSource](#) = vs_JPEGsOrBitmaps, TVideoGrabber needs the video frames to be passed by SendImageToVideoFromBitmaps or [SendImageToVideoFromBitmaps2](#)

It is possible to:

- either invoke SendImageToVideoFromBitmaps from the [OnVideoFromBitmapsNextFrameNeeded](#) event, that occurs periodically to request the next frame,
- either invoke periodically SendImageToVideoFromBitmaps directly. In this 2nd case, SendImageToVideoFromBitmaps must be **called one time just before invoking** StartPreview() or StartRecording().

See [Video clips built on the fly by passing bitmap handles, BMP or JPEG files](#) for more information.

You will find sample code in the MainDemo project included in the package, or in the VideoFromBitmapHandles or VideoFromBMPorJPEGfiles demos.

See Also

[OnBitmapsLoadingProgress](#) [SendImageToVideoFromBitmaps2](#) [VideoFromImages](#) [BitmapsSortedByVideoFromImages](#) [RepeatIndefinitely](#) [VideoFromImages](#) [SourceDirectory](#) [VideoFromImages](#) [TemporaryFile](#)

Created with the Standard Edition of HelpNDoc: [Create help files for the Qt Help Framework](#)

SendImageToVideoFromBitmaps2

TVideoGrabber.SendImageToVideoFromBitmaps2

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Used to pass a pointer to the bitmap bits of the next image to pass to VideoFromJPGsOrBitmaps

Declaration

```
function SendImageToVideoFromBitmaps2 (pBtmapInfo: LongInt; pBitmapBits: LongInt; EndOfData: Boolean): Boolean;
```

```
bool __fastcall SendImageToVideoFromBitmaps2(Windows::PBitmapInfo pBtmapInfo, System::PByte pBitmapBits, bool EndOfData);
```

Function SendImageToVideoFromBitmaps2 (pBtmapInfo as Long, pBitmapBits as Long, EndOfData as Boolean) as Boolean

Description

Passes the next image to VideoFromJPGsOrBitmaps as a pointer to the bitmap bits of the next image, when building a clip from bitmaps.

Works the same way as [SendImageToVideoFromBitmaps](#).

See Also

[OnBitmapsLoadingProgress](#) [SendImageToVideoFromBitmaps](#) [VideoFromImages](#) [BitmapsSortedByVideoFromImages](#) [RepeatIndefinitely](#) [VideoFromImages](#) [SourceDirectory](#) [VideoFromImages](#) [TemporaryFile](#)

SendIPCameraCommand

TVideoGrabber.SendIPCameraCommand

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Sends an HTTP command to the IP camera.

Declaration

function SendIPCameraCommand (IPCameraCommand: **String**): Boolean;

bool __fastcall SendIPCameraCommand(wchar_t *IPCameraCommand);

function SendIPCameraCommand(IPCameraCommand as String) as Boolean

Description

Used to send an HTTP command to the IP camera.

The command string must be an HTTP URL. The syntax depends of the manufacturer.

E.g. to send PTZ commands to an Axis IP Camera with PTZ control:

SendIPCameraCommand ("http://x.x.x.x/axis-cgi/com/ptz.cgi?move=home")

SendIPCameraCommand ("http://x.x.x.x/axis-cgi/com/ptz.cgi?move=up")

etc...

(replace x.x.x.x by the IP address of the camera)

The list of the syntax for Axis IP cameras can be found here:

http://www.axis.com/techsup/cam_servers/dev/cam_http_api_index.php

See Also

[IR Cut Filter of Axis cameras](#) [TDiscoveryCallbackStatus](#) [TOnONVIFDiscoveryCompletedNotification](#) [Video stream of IP camera with audio of PC microphone or other audio capture device](#) [ExtraDLLPath](#) [GetLastErrorMessage](#) [IPCameraURL](#) [IsURLResponding](#) [IsURLVideoStreamAvailable](#) [OnONVIFDiscoveryCompletedNotification](#) [ONVIF](#) [GetStr](#) [ONVIF](#) [SetStr](#) [ONVIFCancelDiscovery](#) [ONVIFDiscoverCameras](#) [IPRange](#) [ONVIFDiscoverCameras](#) [Multicast](#) [ONVIFEnumCamerasDiscovered](#) [OpenURLAsync](#) [OpenURLAsyncStatus](#) [SetDatasteadFilterDllName](#) [SetIPCameraSetting](#)

SetAudioRendererAdditional

TVideoGrabber.SetAudioRendererAdditional

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Allows to use an additional audio renderer

Declaration

procedure SetAudioRendererAdditional (Value: LongInt);

void __fastcall SetAudioRendererAdditional(int Value);

Description

Allows to render the audio to a second audio renderer, additionally to the first one.

Set AudioRendererAdditional = [AudioRendererIndex](#) ("... name of the audio renderer to use...") among the audio renderers listed in the [AudioRenderers](#) list.

Note: be sure to specify another index that the index of the default audio renderer (the default audio renderer has the index 0).

Created with the Standard Edition of HelpNDoc: [Maximize Your Reach: Convert Your Word Document to an ePub or Kindle eBook](#)

SetAuthentication

TVideoGrabber.SetAuthentication

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets the username and password required to connect to a publishing point, a streaming URL or an IP camera

Declaration

procedure SetAuthentication (AuthenticationType: TAuthenticationType; UserName: **String**; Password: **String**);

void __fastcall SetAuthentication(TAuthenticationType TAuthenticationType; wchar_t *UserName; wchar_t *Password);

Description

Invoke SetAuthentication to set the username and password that will be required soon.

E.g.:

- to set the username and password for a publishing point on a windows media server:

```
VideoGrabber.NetworkStreaming = ns_ASFStreamingToPublishingPoint
VideoGrabber.ASFMediaServerPublishingPoint = "http://...."
VideoGrabber.SetAuthentication (at_PublishingPoint, "MyPubpointUser", "MyPubpointPassword");
VideoGrabber.StartPreview
```

- to set the username and password for an IP camera:

```
VideoGrabber.VideoSource = vs_IPCamera
VideoGrabber.IPCameraURL = "http://..."
VideoGrabber.SetAuthentication (at_IPCamera, "MyIPCamUser", "MyIPCamPassword");
VideoGrabber.StartPreview
```

- to set the username and password for a streaming URL:

```
VideoGrabber.VideoSource = vs_VideoFileOrUrl
VideoGrabber.VideoSource_FileOrUrl = "http://.....asf"
VideoGrabber.SetAuthentication (at_StreamingUrl, "MyURLUsername", "MyURLPassword");
VideoGrabber.StartPreview
```

Created with the Standard Edition of HelpNDoc: [News and information about help authoring tools and software](#)

SetAVIMuxConfig

TVideoGrabber.SetAVIMuxConfig

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

ber

Configures the AVI mux

Declaration

procedure SetAVIMuxConfig(AVIMuxSetting: TAVIMuxConfig; Value: LongInt);

void __fastcall SetAVIMuxConfig(TAVIMuxConfig AVIMuxSetting, **int** Value)

Sub SetAVIMuxConfig (AVIMuxSetting as TAVIMuxConfig, Value as Long)

Description

Used to adjust non-default values of the AVI multiplexer.

avmx_SetInterleavingMode

adjusts the interleaving mode:

0 = INTERLEAVE_NONE

1 = INTERLEAVE_CAPTURE

2 = INTERLEAVE_FULL

3 = INTERLEAVE_NONE_BUFFERED

avmx_SetInterleave

adjusts the interleave expressed in 100ns units

"The default value for prtInterleave is 1000 milliseconds; however, you can adjust this. The smaller the number, the larger the file."

avmx_SetPreroll:

adjusts the audio preroll

"An audio preroll of 750 milliseconds is recommended when authoring a file for distribution."

Created with the Standard Edition of HelpNDoc: [Produce electronic books easily](#)

SetCameraControl

TVideoGrabber.SetCameraControl

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets a [TCameraControl](#) setting.

Declaration

function SetCameraControl(Setting: TCameraControl; SetAuto: Boolean; SetDefault: Boolean; SetValue: LongInt): Boolean;

bool __fastcall SetCameraControl(TCameraControl Setting, bool SetAuto, bool SetDefault, int SetValue)

Function SetCameraControl(Setting As TxCameraControl, SetAuto as Boolean, SetDefault as Boolean, SetValue as Long) As Boolean

Description

Used to set a [TCameraControl](#) setting, if available for the current video capture device (test [IsCameraControlSetting](#) for availability).

- if **SetAuto** is true, SetDefault and SetValue are ignored and the setting is switched in "auto" mode
- if **SetAuto** is false and **SetDefault** is true, SetValue is ignored and the setting is set to its default value,
- if **SetAuto** is false and **SetDefault** is false, the **SetValue** value is applied to the setting.

See Also

[TCameraControl](#) [CameraControlAuto](#) [CameraControlDefault](#) [CameraControlMax](#) [CameraControlMin](#) [CameraControlSettings](#) [CameraControlStep](#) [GetCameraExposure](#) [GetCameraExposureAsString](#) [IsCameraControlAvailable](#) [IsCameraControlSettingAvailable](#) [SendCameraCommand](#) [SetCameraExposure](#)

SetCameraExposure

TVideoGrabber.SetCameraExposure

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets the camera exposure as a double value

Declaration

procedure SetCameraExposure (Value: Double);

void __fastcall SetCameraExposure(**double** Value);

Sub SetCameraExposure (Value as Double)

Description

Used to set the camera exposure as a double value, e.g. 1/8 s, 1/16 s, 1/32 s, etc...

See Also

[TCameraControl](#) [CameraControlAuto](#) [CameraControlDefault](#) [CameraControlMax](#) [CameraControlMin](#) [CameraControlSettings](#) [CameraControlStep](#) [GetCameraExposure](#) [GetCameraExposureAsString](#) [IsCameraControlAvailable](#) [IsCameraControlSettingAvailable](#) [SendCameraCommand](#) [SetCameraControl](#)

SetDatasteadFilterDllName

TVideoGrabber.SetDatasteadFilterDllName

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Specifies the Datastead .ax name

Declaration

function SetDatasteadFilterDllName (Value: **string**): Boolean;

bool __fastcall SetDatasteadFilterDllName(System::UnicodeString Value);

Description

This is related to the external TVideoGrabber components, e.g. the Datastead RTSP/RTMP/HTTP/ONVIF Source, the Datastead Multipurpose Encoder of the Datastead NDI Filters.

This function let specify the name of the Datastead ".ax" if it has been renamed.

Depending on the Datastead product, the name of the .ax file must be prefixed as follows:

- RTSP/RTMP/HTTP/ONVIF Source: **[rtsp]**
- Datastead Multipurpose Encoder: **[mpe]**
- Datastead NDI Filters: **[ndi]**

E.g. if DatasteadRTSPSource.ax has been renamed to DatasteadRTSP.ax , specify its name as follows:

`VideoGrabber.SetDatasteadFilterDLLName ("["rtsp"]DatasteadRTSP.ax");`

See Also

[Datastead Encoder](#) [IR Cut Filter of Axis cameras](#) [Pause/resume during recording](#) [Recording methods and properties](#) [TDiscoveryCallbackStatus](#) [TNDIFormatType](#) [TONVIFDiscoveryCompletedNotification](#) [TONVIFDeviceInfo](#) [Video stream of IP camera with audio of PC microphone or other audio capture device](#) [Encoder_GetInt](#) [Encoder_Pause](#) [Encoder_Resume](#) [Encoder_SetInt](#) [Encoder_SetStr](#)

[Encoders_CreateInstanceForRecording](#) [Encoders_CreateInstanceForStreaming](#)
[Encoders_RemoveAllInstances](#) [Encoders_RemoveInstance](#) [ExtraDLLPath](#)
[GetLastErrorMessages](#) [IPCameraURL](#) [IsURLResponding](#) [IsURLVideoStreamAvailable](#)
[NDIFormatType](#) [NDIName](#) [NetworkStreaming](#) [OnONVIFDiscoveryCompletedNotification](#)
[ONVIF_GetStr](#) [ONVIF_SetStr](#) [ONVIFCancelDiscovery](#) [ONVIFDeviceInfo](#)
[ONVIFDiscoverCameras_IPRange](#) [ONVIFDiscoverCameras_Multicast](#)
[ONVIFEnumCamerasDiscovered](#) [ONVIFPTZGetLimits](#) [ONVIFPTZGetPosition](#)
[ONVIFPTZPreset](#) [ONVIFPTZSendAuxiliaryCommand](#) [ONVIFPTZSetPosition](#)
[ONVIFPTZStartMove](#) [ONVIFPTZStopMove](#) [ONVIFSnapshot](#) [OpenURLAsync](#)
[OpenURLAsyncStatus](#) [SendIPCameraCommand](#) [SetIPCameraSetting](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly bring your documentation online with HelpNDoc](#)

SetDecryptionKey

TVideoGrabber.SetDecryptionKey

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Key used for decryption

Declaration

function SetDecryptionKey (Key: pBYTE; KeyLen: LongInt): Boolean;

bool __fastcall SetDecryptionKey(System::UnicodeString Value);

Description

Sets the key to decrypt the video/audio during recording

See Also

[TEncryptionMethod](#) [Decrypt_File](#) [Encrypt_File](#) [EncryptionMethod](#) [SetEncryptionKey](#)

Created with the Standard Edition of HelpNDoc: [Easily create PDF Help documents](#)

SetDisplayActive

TVideoGrabber.SetDisplayActive

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Enables/disables the window

Declaration

procedure SetDisplayActive (DisplayIndex: LongInt; Value: Boolean);

void __fastcall SetDisplayActive(int DisplayIndex, **bool** Value);

Description

Used to enable/disable the video window.

The window Index specified must be in the 2..8 range (0..1 reserved to Display... and Dual_Display)

Created with the Standard Edition of HelpNDoc: [Create Professional CHM Help Files with HelpNDoc's Easy-to-Use Tool](#)

SetDisplayAlphaBlendEnabled

TVideoGrabber.SetDisplayAlphaBlendEnabled

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

[ber](#)

Activates the alpha blending of the video window

Declaration

procedure SetDisplayAlphaBlendEnabled (DisplayIndex: LongInt; Value: Boolean);

void __fastcall SetDisplayAlphaBlendEnabled(**int** DisplayIndex, **bool** Value);

Description

Used to activate the alpha blending of the current video window with another video window when it is detached from the control by invoking [SetDisplayEmbedded](#) (index, false)

The window Index specified must be in the 2..8 range (0..1 reserved to Display... and Dual_Display)

Created with the Standard Edition of HelpNDoc: [Step-by-Step Guide: How to Turn Your Word Document into an eBook](#)

SetDisplayAlphaBlendValue**TVideoGrabber.SetDisplayAlphaBlendValue**[Prev](#)[Next](#)

[TVideoGrabber](#) [Methods](#)

Value of the alpha blending of the video window

Declaration

procedure SetDisplayAlphaBlendValue (DisplayIndex: LongInt; Value: LongInt);

void __fastcall SetDisplayAlphaBlendValue(**int** DisplayIndex, **int** Value);

Description

Value of the alpha blending of the video window, in the 0..255 range (e.g. 128 = semi-transparent)

This feature can be used after invoking [SetDisplayEmbedded](#) (index, false) and [SetDisplayAlphaBlendEnabled](#) (index, false)

The window Index specified must be in the 2..8 range (0..1 reserved to Display... and Dual_Display)

Created with the Standard Edition of HelpNDoc: [Save time and frustration with HelpNDoc's WinHelp HLP to CHM conversion feature](#)

SetDisplayAspectRatio**TVideoGrabber.SetDisplayAspectRatio**[Prev](#)[Next](#)

[TVideoGrabber](#) [Methods](#)

Specifies the aspect ratio to use within the video window.

Declaration

procedure SetDisplayAspectRatio (DisplayIndex: LongInt; Value: TAspectRatio);

void __fastcall SetDisplayAspectRatio(**int** DisplayIndex, TAspectRatio Value);

Description

Used after invoking [SetDisplayAutoSize](#) (index, false) to specify the aspect ratio method that must be used within the video window.

The window Index specified must be in the 2..8 range (0..1 reserved to Display... and Dual_Display)

SetDisplayAutoSize

TVideoGrabber.SetDisplayAutoSize

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

If enabled, the control is resized automatically according to the current video size.

Declaration

procedure SetDisplayAutoSize (DisplayIndex: LongInt; Value: Boolean);

void __fastcall SetDisplayAutoSize(**int** DisplayIndex, **bool** Value);

Description

Specifies whether the control must be resized automatically according to the current video size.

disabled: the control size depends of the Width and Height properties, and the video window is stretched inside.

enabled: the control size is automatically modified according to [VideoSize](#) or [UseNearestVideoSize](#) .

The window Index specified must be in the 2..8 range (0..1 reserved to Display... and Dual_Display)

SetDisplayEmbedded

TVideoGrabber.SetDisplayEmbedded

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Used to detach/attach the video window from the TVideoGrabber control.

Declaration

procedure SetDisplayEmbedded (DisplayIndex: LongInt; Value: Boolean);

void __fastcall SetDisplayEmbedded(**int** DisplayIndex, **bool** Value);

Description

Used to detach/attach the video window from the TVideoGrabber control.

- **enabled:** the video window is embedded into the TVideoGrabber control,
- **disabled:** the video window is located on the desktop at the positions specified by invoking [SetDisplayLeft](#) and [SetDisplayTop](#), and optionally [SetDisplayMonitor](#) if more than 1 monitor is used.

The window Index specified must be in the 2..8 range (0..1 reserved to Display... and Dual_Display)

SetDisplayFullScreen

TVideoGrabber.SetDisplayFullScreen

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Displays the preview window in full screen mode.

Declaration

procedure SetDisplayFullScreen (DisplayIndex: LongInt; Value: Boolean);

void __fastcall SetDisplayFullScreen(int DisplayIndex, **bool** Value);

Description

If enabled, the preview window is displayed in full screen mode when the preview, recording or play back starts.

To exit from the full screen mode disable this property or press the <ESC> key.

In full screen mode all keystrokes are returned by the [OnKeyPress](#) event.

The window Index specified must be in the 2..8 range (0..1 reserved to Display... and Dual_Display)

Created with the Standard Edition of HelpNDoc: [Easily create CHM Help documents](#)

SetDisplayHeight

TVideoGrabber.SetDisplayHeight

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Specifies the height of the video window, when it is not embedded in the TVideoGrabber control.

Declaration

procedure SetDisplayHeight (DisplayIndex: LongInt; Value: LongInt);

void __fastcall SetDisplayHeight(int DisplayIndex, int Value);

Description

Used to specify the height of the video window, when it is not embedded in the TVideoGrabber control.

Note: the left, top, width and height properties can be set at the same time by invoking [SetDisplayLocation](#).

The window Index specified must be in the 2..8 range (0..1 reserved to Display... and Dual_Display)

Created with the Standard Edition of HelpNDoc: [Full-featured EPub generator](#)

SetDisplayLeft

TVideoGrabber.SetDisplayLeft

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Specifies the left position of the video window, when it is not embedded in the TVideoGrabber control.

Declaration

procedure SetDisplayLeft (DisplayIndex: LongInt; Value: LongInt);

void __fastcall SetDisplayLeft(int DisplayIndex, int Value);

Description

Specifies the left position of the video window, when it is not embedded in the TVideoGrabber control.

Note: the left, top, width and height properties can be set at the same time by invoking [SetDisplayLocation](#).

The window Index specified must be in the 2..8 range (0..1 reserved to Display... and Dual_Display)

Created with the Standard Edition of HelpNDoc: [Maximize Your CHM Help File Capabilities with HelpNDoc](#)

SetDisplayLocation

TVideoGrabber.SetDisplayLocation

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets the location of the display window

Declaration

function SetDisplayLocation (DisplayIndex: LongInt; WindowLeft: LongInt; WindowTop: LongInt; WindowWidth: LongInt; WindowHeight: LongInt): Boolean;

bool __fastcall SetDisplayLocation(**int** DisplayIndex, **int** WindowLeft, **int** WindowTop, **int** WindowWidth, **int** WindowHeight);

Description

Used to set the location and size of the display window when it is not embedded in the control, after invoking SetDisplayEmbedded (index, false)

The window Index specified must be in the 2..8 range (0..1 reserved to Display... and Dual_Display)

Created with the Standard Edition of HelpNDoc: [Achieve Professional Documentation Results with a Help Authoring Tool](#)

SetDisplayMonitor

TVideoGrabber.SetDisplayMonitor

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Specifies the monitor used to display the video window.

Declaration

procedure SetDisplayMonitor (DisplayIndex: LongInt; Value: LongInt);

void __fastcall SetDisplayMonitor(**int** DisplayIndex, **int** Value);

Description

Used to specify the monitor used to display the video window.
The value is in the (0..[MonitorsCount](#) -1) range.

The window Index specified must be in the 2..8 range (0..1 reserved to Display... and Dual_Display)

Created with the Standard Edition of HelpNDoc: [Transform Your CHM Help File Creation Process with HelpNDoc](#)

SetDisplayMouseMovesWindow

TVideoGrabber.SetDisplayMouseMovesWindow

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Specifies whether the mouse moves the video window or returns mouse events for this window.

Declaration

procedure SetDisplayMouseMovesWindow (DisplayIndex: LongInt; Value: Boolean);

void __fastcall SetDisplayMouseMovesWindow(**int** DisplayIndex, **bool** Value);

Description

When the video window is not embedded in the TVideoGrabber control:

- **if enabled:** the mouse moves the video window,
- **if disabled:** the mouse returns mouse events, e.g. to draw graphic objects over the video window.

The window Index specified must be in the 2..8 range (0..1 reserved to Display... and Dual_Display)

Created with the Standard Edition of HelpNDoc: [iPhone web sites made easy](#)

SetDisplayPanScanRatio

TVideoGrabber.SetDisplayPanScanRatio

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Adjust the Pan/Scan ratio.

Declaration

procedure SetDisplayPanScanRatio (DisplayIndex: LongInt; Value: LongInt);

void __fastcall SetDisplayPanScanRatio(**int** DisplayIndex, **int** Value);

Description

After invoking [SetDisplayAspectRatio](#) (index, ar_PanScan), this function can be used to adjust the Pan/Scan ratio (in the 0..100 range, default value 50)

The window Index specified must be in the 2..8 range (0..1 reserved to Display... and Dual_Display)

Created with the Standard Edition of HelpNDoc: [Easy Qt Help documentation editor](#)

SetDisplayParent

TVideoGrabber.SetDisplayParent

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets the display parent of the video window

Declaration

function SetDisplayParent (DisplayIndex: LongInt; DisplayParent: HWND): Boolean;

bool __fastcall SetDisplayParent(**int** DisplayIndex, HWND DisplayParent);

Description

Invoke it to set the parent of the video window. The parent parameter must be a HWND window handle.

The window Index specified must be in the 2..8 range (0..1 reserved to Display... and Dual_Display)

Created with the Standard Edition of HelpNDoc: [Benefits of a Help Authoring Tool](#)

SetDisplayStayOnTop

TVideoGrabber.SetDisplayStayOnTop

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Specifies whether the video window must stay over other windows, when not embedded.

Declaration

procedure SetDisplayStayOnTop (DisplayIndex: LongInt; Value: Boolean);

void __fastcall SetDisplayStayOnTop(**int** DisplayIndex, **bool** Value);

Description

Specifies whether the video window must stay over other windows, when it is not embedded in the TVideoGrabber control.

The window Index specified must be in the 2..8 range (0..1 reserved to Display... and Dual_Display)

Created with the Standard Edition of HelpNDoc: [Streamline Your Documentation Process with a Help Authoring Tool](#)

SetDisplayTop

TVideoGrabber.SetDisplayTop

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Specifies the top position of the video window, when it is not embedded in the TVideoGrabber control.

Declaration

procedure SetDisplayTop (DisplayIndex: LongInt; Value: LongInt);

void __fastcall SetDisplayTop(int DisplayIndex, int Value);

Description

Specifies the top position of the video window, when it is not embedded in the TVideoGrabber control.

Note: the left, top, width and height properties can be set at the same time by invoking [SetDisplayLocation](#).

The window Index specified must be in the 2..8 range (0..1 reserved to Display... and Dual_Display)

Created with the Standard Edition of HelpNDoc: [Maximize Your Documentation Capabilities with a Help Authoring Tool](#)

SetDisplayTransparentColorEnabled

TVideoGrabber.SetDisplayTransparentColorEnabled

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Activate the color keying of the current video window

Declaration

procedure SetDisplayTransparentColorEnabled (DisplayIndex: LongInt; Value: Boolean);

void __fastcall SetDisplayTransparentColorEnabled(int DisplayIndex, bool Value);

Description

Used to activate the color keying of the current video window with another video window when it is detached from the control by invoking [SetDisplayEmbedded](#) (index, false)

The window Index specified must be in the 2..8 range (0..1 reserved to Display... and Dual_Display)

Created with the Standard Edition of HelpNDoc: [Effortlessly Create High-Quality Documentation with a Help Authoring Tool](#)

SetDisplayTransparentColorValue

TVideoGrabber.SetDisplayTransparentColorValue

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Value of the color keying of the video window

Declaration

procedure SetDisplayTransparentColorValue (DisplayIndex: LongInt; Value: LongInt);

void __fastcall SetDisplayTransparentColorValue(**int** DisplayIndex, **int** Value);

Description

Value of the color keying of the video window, expressed in RGB value.

This feature can be used after invoking [SetDisplayEmbedded](#) (index, false) and [SetDisplayTransparentColorEnabled](#) (index, false)

The window Index specified must be in the 2..8 range (0..1 reserved to Display... and Dual_Display)

Created with the Standard Edition of HelpNDoc: [Easily create Help documents](#)

SetDisplayVideoPortEnabled

TVideoGrabber.SetDisplayVideoPortEnabled

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Enables the video port renderer

Declaration

procedure SetDisplayVideoPortEnabled (DisplayIndex: LongInt; Value: Boolean);

void __fastcall SetDisplayVideoPortEnabled(**int** DisplayIndex, **bool** Value);

Description

Used to enable the video port renderer, if available

Created with the Standard Edition of HelpNDoc: [Easily create PDF Help documents](#)

SetDisplayVisible

TVideoGrabber.SetDisplayVisible

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Shows / hide the video window

Declaration

procedure SetDisplayVisible (DisplayIndex: LongInt; Value: Boolean);

void __fastcall SetDisplayVisible(**int** DisplayIndex, **bool** Value);

Description

Used to show / hide the video window.
Enabled by default.

The window Index specified must be in the 2..8 range (0..1 reserved to Display... and Dual_Display)

Created with the Standard Edition of HelpNDoc: [Maximize Your Productivity with a Help Authoring Tool](#)

SetDisplayWidth

TVideoGrabber.SetDisplayWidth

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Specifies the width of the video window, when it is not embedded in the TVideoGrabber control.

Declaration

procedure SetDisplayWidth (DisplayIndex: LongInt; Value: LongInt);

void __fastcall SetDisplayWidth(**int** DisplayIndex, **int** Value);

Description

Used to specify the width of the video window, when it is not embedded in the TVideoGrabber control.

Note: the left, top, width and height properties can be set at the same time by invoking [SetDisplayLocation](#).

The window Index specified must be in the 2..8 range (0..1 reserved to Display... and Dual_Display)

Created with the Standard Edition of HelpNDoc: [Create Professional CHM Help Files with HelpNDoc's Easy-to-Use Tool](#)

SetEncryptionKey

TVideoGrabber.SetEncryptionKey

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Key used for encryption

Declaration

function SetEncryptionKey (Key: pBYTE; KeyLen: LongInt): Boolean;

bool __fastcall SetEncryptionKey(System::UnicodeString Value);

Description

Sets the key to encrypt the video/audio during recording

See Also

[TEncryptionMethod Decrypt File Encrypt File EncryptionMethod SetDecryptionKey](#)

Created with the Standard Edition of HelpNDoc: [Free PDF documentation generator](#)

SetFFmpegAudioFilter

TVideoGrabber.SetFFmpegAudioFilter

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

enables a FFmpeg audio filter

Declaration

function SetFFmpegFilter (FilterIndex: LongInt; VideoFilterName: **string**; Parameters: **string**): Boolean;

bool __fastcall SetFFmpegFilter(System::LongInt FilterIndex, System::UnicodeString VideoFilterName, System::UnicodeString Parameters);

Description

Used to activate a FFmpeg audio filter.

Note: this feature requires the Datastead RTSP/RTMP/HTTP/ONVIF Source Filter to be installed.

Note: the filter must be an "in place" filter that outputs the same video format and video size than the input format.

Example to perform a voice masking:

```
VideoGrabber.SetFFmpegAudioFilter(0, "afftfilt",
"real='hypot(re,im)':imag='0':win_size=2048:overlap=0.75:win_func=hann");
VideoGrabber.SetFFmpegAudioFilter(1, "asetrate", "52000");
```

See Also

[MultipurposeEncoder_QuickConfigure_UDPStreaming_H264](#)
[MultipurposeEncoder_ReindexClip SetFFmpegFilter](#)

SetFFmpegFilter

TVideoGrabber.SetFFmpegFilter

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

enables a FFmpeg video filter

Declaration

function SetFFmpegFilter (FilterIndex: LongInt; VideoFilterName: **string**; Parameters: **string**): Boolean;

bool __fastcall SetFFmpegFilter(System::LongInt FilterIndex, System::UnicodeString VideoFilterName, System::UnicodeString Parameters);

Description

Used to activate a FFmpeg video filter.

Note: this feature requires the Datastead RTSP/RTMP/HTTP/ONVIF Source Filter to be installed.

Note: the filter must be an "in place" filter that outputs the same video format and video size than the input format.

Example:

```
TVideoGrabber.SetFFmpegFilter (0, 'vignette',
'angle=PI/2:mode=forward:x0=w/2:y0=h/2:dither=0');
TVideoGrabber.SetFFmpegFilter (1, 'vflip', '');
```

See Also

[MultipurposeEncoder_QuickConfigure_UDPStreaming_H264](#)
[MultipurposeEncoder_ReindexClip](#) [SetFFmpegAudioFilter](#)

SetFrameCaptureBounds

TVideoGrabber.SetFrameCaptureBounds

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Specifies a frame capture rectangle

Declaration

procedure SetFrameCaptureBounds (LeftPosition: LongInt; TopPosition: LongInt; RightPosition: LongInt; BottomPosition: LongInt);

void __fastcall SetFrameCaptureBounds(**int** LeftPosition, **int** TopPosition, **int** RightPosition, **int** BottomPosition)

Sub SetFrameCaptureBounds (LeftPosition as Long, TopPosition as Long, RightPosition as Long, BottomPosition as Long)

Description

Used to capture a sub-rectangle of the video frame

Used to specify a rectangle for the frame capture, allowing to capture only parts of the video frame.

E.g.:

Button1 action:

VideoGrabber.StartPreview()

Button2 action:

VideoGrabber.SetFrameCaptureBounds (50, 50, 150, 150)

VideoGrabber.CaptureFrameTo (fc_JPEGFile, "mysquarecapture.jpg")

When starting the preview with Button1, then capturing a frame with Button2, the captured frame will be a square 100x100 image captured at the "x=50 y=50" location on the video frame.

Note: the normal full frame capture can then be restored by invoking SetFrameCaptureBounds (0, 0, 0, 0)

Top-down or left-right capture

Additionally it is possible top-down or left-right the captured frame by just inverting the Left/Right or Top/Bottom values.

E.g.:

VideoGrabber.SetFrameCaptureBounds (50, **150**, 150, **50**) will capture a top-down frame.

Remark: to retrieve the current size of the video frame use [VideoWidth](#) and [VideoHeight](#)

See Also

[TAutoFileName](#) [TFrameCaptureDest](#) [TOnFrameCaptureCompleted](#) [BurstCount](#) [BurstInterval](#) [BurstMode](#) [BurstType](#) [CaptureFrameSyncTo](#) [CaptureFrameTo](#) [FrameCaptureHeight](#) [FrameCaptureWidth](#) [FrameCaptureWithoutOverlay](#) [FrameCaptureZoomSize](#) [GetFrameInfoString](#) [GetLastFrameAsHBITMAP](#) [GetLastFrameAsTBitmap](#) [GetLastFrameBitmapBits](#) [GetLastFrameBitmapBits2](#) [GetLastFrameWaitTimeoutMs](#) [JPEGPerformance](#) [JPEGProgressiveDisplay](#) [JPEGQuality](#) [Last](#) [BurstFrameCapture](#) [FileName](#) [Last](#) [CaptureFrameTo](#) [FileName](#) [OnDiskFull](#) [OnFrameBitmap](#) [StoragePath](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Elevate Your Help Documentation with a Help Authoring Tool](#)

SetFWCam1394

TVideoGrabber.SetFWCam1394

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets a custom setting of Firewire/GIE camera

Declaration

function SetFWCam1394 (FWCam1394ID: **string**; Value: LongInt): Boolean;

bool __fastcall SetFWCam1394(System::wchar_t *FWCam1394ID, **int** Value);

Description

Example in Delphi:

```
procedure TfrmMainForm.Button14Click(Sender: TObject);
```

```
var
```

```
Value: LongInt;
```

```
Flags: LongInt;
```

```
Capabilities: DWORD;
```

```
MinValue: LongInt;
```

```
MaxValue: LongInt;
```

```
Default: LongInt;
```

```
begin
```

```
if VideoGrabber.GetFWCam1394 ('AUTO_EXPOSURE_REFERENCE', Value, Flags,  
Capabilities, MinValue, MaxValue, Default) then begin
```

```
mmoLog.lines.add(inttostr (Value));
```

```
mmoLog.lines.add(inttostr (Flags));
```

```
mmoLog.lines.add(inttostr (Capabilities));
mmoLog.lines.add(inttostr (MinValue));
mmoLog.lines.add(inttostr (MaxValue));
mmoLog.lines.add(inttostr (Default));
if VideoGrabber.SetFWCam1394 ('AUTO_EXPOSURE_REFERENCE', 2) then begin
showmessage('new value applied');
end;
if VideoGrabber.SetFWCam1394 ('AUTO_EXPOSURE_REFERENCE', 128) then begin
showmessage('new value applied');
end;
end;
end;
end;
```

Created with the Standard Edition of HelpNDoc: [5 Reasons Why a Help Authoring Tool is Better than Microsoft Word for Documentation](#)

SetHeaderAttribute

TVideoGrabber.SetHeaderAttribute

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets the header attributes of AVI / ASF files.

Declaration

procedure SetHeaderAttribute(HeaderAttribute: THeaderAttribute; Value: String);

void __fastcall SetHeaderAttribute(THeaderAttribute HeaderAttributeGraphics, wchar_t *Value)

Sub SetHeaderAttribute(HeaderAttribute as TxHeaderAttribute, Value as String)

Description

Used to set the header attributes (strings like Author, Description, etc...) to the AVI of ASF files. See [AVI & ASF header attributes](#) for more information.

The allowed values for HeaderAttributes are:

- for both AVI and ASF files (RecordingMethod = rm_AVI or rm_ASF)

ha_Title
ha_Description
ha_Author
ha_Copyright

- for ASF files only (RecordingMethod = rm_ASF)

ha_AlbumArtist
ha_AlbumTitle
ha_Composer
ha_ContentDistributor
ha_Director
ha-EncodingTime
ha_Genre
ha_Language
ha_ParentalRating
ha_Producer
ha_Provider
ha_ToolName
ha_ToolVersion
ha_Writer

- for AVI files only (RecordingMethod = rm_AVI)

(from the Multimedia Programming Interface and Data Specifications 1.0)

ha_IARL: Archival Location. Indicates where the subject of the file is archived.

ha_ICMS: Commissioned. Lists the name of the person or organization that commissioned the subject of the file. For examplePope Julian II.

ha_ICMT: Comments. Provides general comments about the file or the subject of the file. If the comment is several sentences longend each sentence with a period. Do not include newline characters.

ha_ICRD: Creation date. Specifies the date the subject of the file was created. List dates in year-month-day formatpadding one-digit months and days with a zero on the left. For example1553-05-03 for May 31553.

ha_ICRP: Cropped. Describes whether an image has been cropped andif sohow it was cropped. For examplelower right corner.

ha_IDIM: Dimensions. Specifies the size of the original subject of the file.

ha_IDPI: Dots Per Inch. Stores dots per inch setting of the digitizer used to produce the filesuch as 300.

ha_IENG: Engineer. Stores the name of the engineer who worked on the file. If there are multiple engineersseparate the names by a semicolon and a blank. For exampleSmithJohn; AdamsJoe.

ha_IGNR: Genre. Describes the original worksuch aslandscapeportraitstill lifeetc.

ha_IKEY: Keywords. Provides a list of keywords that refer to the file or subject of the file. Separate multiple keywords with a semicolon and a blank. For exampleSeattle; aerial view; scenery.

ha_ILGT: Lightness. Describes the changes in lightness settings on the digitizer required to produce the file. Note that the format of this information depends on hardware used.

ha_IMED: Medium. Describes the original subject of the filesuch ascomputer imagedrawinglithographand so forth.

ha_IPLT: Palette Setting. Specifies the number of colors requested when digitizing an imagesuch as 256.

ha_IPRD: Product. Specifies the name of the title the file was originally intended forsuch as Encyclopedia of Pacific Northwest Geography.

ha_ISFT: Software. Identifies the name of the software package used to create the filesuch as Microsoft WaveEdit.

ha_ISHP: Sharpness. Identifies the changes in sharpness for the digitizer required to produce the file (the format depends on the hardware used).

ha_ISRC:/tab Source. Identifies the name of the person or organization who supplied the original subject of the file. For exampleTrey Research.

ha_ISRF: Source Form. Identifies the original form of the material that was digitizedsuch as slidepapermapand so forth. This is not necessarily the same as IMED.

ha_ITCH: Technician. Identifies the technician who digitized the subject file. For exampleSmithJohn.

See Also

[THeaderAttribute](#) [ClearHeaderAttributes](#)

Created with the Standard Edition of HelpNDoc: [Easily convert your WinHelp HLP help files to CHM with HelpNDoc's step-by-step guide](#)

SetImageOverlay_AlphaBlend**TVideoGrabber.SetImageOverlay_AlphaBlend**

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Enables the alpha blending for the specified image overlay

Declaration

procedure SetImageOverlay_AlphaBlend (**Index**: LongInt; **Value** :Boolean);

void __fastcall SetImageOverlay_AlphaBlend(**int** Index, **bool** Value);

Description

Enables the alpha blending for the image overlay of the specified index

The alpha blending value must be specified with [SetImageOverlay_AlphaBlend](#)

Look at the [Image Overlays](#) chapter for more information.

See Also

[Image overlays](#) [TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetPixelsDistance](#) [GetRGBPixelAt](#) [ImageOverlay](#) [AlphaBlend](#) [ImageOverlay](#) [AlphaBlendValue](#) [ImageOverlay](#) [Height](#) [ImageOverlay](#) [LeftLocation](#) [ImageOverlay](#) [StretchToVideoSize](#) [ImageOverlay](#) [TopLocation](#) [ImageOverlay](#) [Transparent](#) [ImageOverlay](#) [TransparentColorValue](#) [ImageOverlay](#) [UseTransparentColor](#) [ImageOverlay](#) [VideoAlignment](#) [ImageOverlay](#) [Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#) [SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#) [SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#) [SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#) [SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#) [TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#) [TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#) [TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#) [TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Revolutionize Your CHM Help File Output with HelpNDoc

SetImageOverlay_AlphaBlendValue

TVideoGrabber.SetImageOverlay_AlphaBlendValue

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets the the alpha blending value for the specified image overlay

Declaration

procedure SetImageOverlay_AlphaBlendValue (**Index**: LongInt; **Value** :LongInt);

void __fastcall SetImageOverlay_AlphaBlendValue(**int** Index, **int** Value);

Description

Sets the alpha blending value for the image overlay of the specified index
The alpha blending must be enabled by [SetImageOverlay_AlphaBlend](#)

Look at the [Image Overlays](#) chapter for more information.

See Also

[Image overlays](#) [TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetPixelsDistance](#) [GetRGBPixelAt](#) [ImageOverlay](#) [AlphaBlend](#) [ImageOverlay](#) [AlphaBlendValue](#) [ImageOverlay](#) [Height](#) [ImageOverlay](#) [LeftLocation](#) [ImageOverlay](#) [StretchToVideoSize](#) [ImageOverlay](#) [TopLocation](#) [ImageOverlay](#) [Transparent](#) [ImageOverlay](#) [TransparentColorValue](#) [ImageOverlay](#) [UseTransparentColor](#) [ImageOverlay](#) [VideoAlignment](#) [ImageOverlay](#) [Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#)

[OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#)
[OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay_AlphaBlend](#)
[SetImageOverlay_ChromaKey](#) [SetImageOverlay_ChromaKeyLeewayPercent](#)
[SetImageOverlay_ChromaKeyRGBColor](#) [SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#)
[SetImageOverlay_LeftLocation](#) [SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#)
[SetImageOverlay_TargetDisplay](#) [SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#)
[SetImageOverlay_TransparentColorValue](#) [SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#)
[SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#)
[SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#)
[SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#)
[SetImageOverlayFromTImage2](#) [SetTextOverlay_Align](#) [SetTextOverlay_BkColor](#) [SetTextOverlay_CustomVar](#)
[SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#) [SetTextOverlay_GradientColor](#)
[SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#)
[SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#)
[SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#)
[SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#)
[ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#)
[TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#)
[TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#)
[TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#)
[TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#)
[TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Enhance Your Documentation with HelpNDoc's Advanced Project Analyzer](#)

SetImageOverlay_Attributes

TVideoGrabber.SetImageOverlay_Attributes

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets the location, size and transparency for the current image overlay selected by [ImageOverlaySelector](#)

Declaration

procedure SetImageOverlayAttributes (LeftLocation: LongInt; TopLocation: LongInt; StretchedWidth: LongInt; StretchedHeight: LongInt; Transparent: Boolean; UseTransparentColor: Boolean; TransparentColorValue: LongInt; AlphaBlend: Boolean; AlphaBlendValue: LongInt);

void __fastcall SetImageOverlayAttributes(**int** LeftLocation, **int** TopLocation, **int** StretchedWidth, **int** StretchedHeight, **bool** Transparent, **bool** UseTransparentColor, **TColor** TransparentColorValue, **bool** AlphaBlend, **int** AlphaBlendValue)

Sub SetImageOverlayAttributes LeftLocation As Long, TopLocation As Long, bmWidth As Long, bmHeight As Long, Transparent As Boolean, UseTransparentColor As Boolean, TransparentColorValue As Long, AlphaBlend As Boolean, AlphaBlendValue As Long)

Description

Used to Specify the location, size and transparency of the image overlayed over the video frames

Note: see the [Image overlays](#) chapter for global information about this feature.

Location

The image will be drawn at the LeftLocation, TopLocation position over the video frames (expressed in pixels).

Size

- if StretchedWidth and StretchedHeight are > 0, the image is resized to these values
- if StretchedWidth = 0 and StretchedHeight = 0, the image size is used

Transparency

- if the **Transparent** parameter is true, the image background is not drawn.
- if the **Transparent** parameter is true and the **UseTransparentColor** parameter is true, the **TransparentColorValue** value is used for transparency.

Alpha blending

If the **AlphaBlend** parameter is true, a progressive bitmap transparency is applied, depending of the **AlphaBlendValue**, in a 0..255 range.

It is not possible to perform alphablending when the **Transparent** parameter is true.

Important:

if the bitmap color format is not 32 bits, it will be converted into a 32 bit format. In other words, for better performance pass a bitmap already in 32 bits color format.

See Also

[Image overlays](#) [TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetPixelsDistance](#) [GetRGBPixelAt](#) [ImageOverlay](#) [AlphaBlend](#) [ImageOverlay](#) [AlphaBlendValue](#) [ImageOverlay](#) [Height](#) [ImageOverlay](#) [LeftLocation](#) [ImageOverlay](#) [StretchToVideoSize](#) [ImageOverlay](#) [TopLocation](#) [ImageOverlay](#) [Transparent](#) [ImageOverlay](#) [TransparentColorValue](#) [ImageOverlay](#) [UseTransparentColor](#) [ImageOverlay](#) [VideoAlignment](#) [ImageOverlay](#) [Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronon](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#) [SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#) [SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#) [SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#) [SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#) [TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#) [TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#) [TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#) [TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Generate EPub eBooks with ease](#)

SetImageOverlay_Attributes2

TVideoGrabber.SetImageOverlay_Attributes2

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets the location, size and transparency for the image overlay specified by its index

Declaration

procedure SetImageOverlayAttributes2 (Index: LongInt; LeftLocation: LongInt; TopLocation: LongInt; StretchedWidth: LongInt; StretchedHeight: LongInt; Transparent: Boolean; UseTransparentColor: Boolean;

TransparentColorValue: LongInt; AlphaBlend: Boolean; AlphaBlendValue: LongInt);

void __fastcall SetImageOverlayAttributes2(**int** Index, **int** LeftLocation, **int** TopLocation, **int** StretchedWidth, **int** StretchedHeight, **bool** Transparent, **bool** UseTransparentColor, TColor TransparentColorValue, **bool** AlphaBlend, **int** AlphaBlendValue)

Sub SetImageOverlayAttributes2 (Index as Long, LeftLocation As Long, TopLocation As Long, bmWidth As Long, bmHeight As Long, Transparent As Boolean, UseTransparentColor As Boolean, TransparentColorValue As Long, AlphaBlend As Boolean, AlphaBlendValue As Long)

Description

Same as [SetImageOverlay_Attribute](#), but with the overlay Index as first parameter, so it is not necessary to pre-select it with ImageOverlaySelector.

Created with the Standard Edition of HelpNDoc: [Free help authoring environment](#)

SetImageOverlay_ChromaKey

TVideoGrabber.SetImageOverlay_ChromaKey

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Enables/disables the chroma key feature for the specified image overlay

Declaration

procedure SetImageOverlay_ChromaKey (**Index**: LongInt; Value :Boolean);

void __fastcall SetImageOverlay_ChromaKey(**int** Index, **bool** Value);

Description

Used to enable or disable the chroma key feature for the image overlay of the specified index

See the [Chroma Key](#) chapter.

See Also

[Image overlays](#) [TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetPixelsDistance](#) [GetRGBPixelAt](#) [ImageOverlay](#) [AlphaBlend](#) [ImageOverlay](#) [AlphaBlendValue](#) [ImageOverlay](#) [Height](#) [ImageOverlay](#) [LeftLocation](#) [ImageOverlay](#) [StretchToVideoSize](#) [ImageOverlay](#) [TopLocation](#) [ImageOverlay](#) [Transparent](#) [ImageOverlay](#) [TransparentColorValue](#) [ImageOverlay](#) [UseTransparentColor](#) [ImageOverlay](#) [VideoAlignment](#) [ImageOverlay](#) [Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#) [SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#) [SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#) [SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#) [SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#) [TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#) [TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#) [TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#)

[TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#)
[TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Easily convert your WinHelp HLP help files to CHM with HelpNDoc's step-by-step guide](#)

SetImageOverlay_ChromaKeyLeewayPercent

TVideoGrabber.SetImageOverlay_ChromaKeyLeewayPercent

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets the percentage of leeway of the chroma key for the specified image overlay

Declaration

procedure SetImageOverlay_ChromaKeyLeewayPercent (**Index**: LongInt; **Value** :LongInt);

void __fastcall SetImageOverlay_ChromaKeyLeewayPercent(**int** Index, **int** Value);

Description

Used to set the percentage of leeway of the chroma key for the image overlay of the specified index

See the [Chroma Key](#) chapter.

See Also

[Image overlays](#) [TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#)
[TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetPixelsDistance](#)
[GetRGBPixelAt](#) [ImageOverlay_AlphaBlend](#) [ImageOverlay_AlphaBlendValue](#) [ImageOverlay_Height](#)
[ImageOverlay_LeftLocation](#) [ImageOverlay_StretchToVideoSize](#) [ImageOverlay_TopLocation](#)
[ImageOverlay_Transparent](#) [ImageOverlay_TransparentColorValue](#) [ImageOverlay_UseTransparentColor](#)
[ImageOverlay_VideoAlignment](#) [ImageOverlay_Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#)
[MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#)
[OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay_AlphaBlend](#)
[SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#) [SetImageOverlay_ChromaKeyRGBColor](#)
[SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#)
[SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#)
[SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#)
[SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay_Align](#) [SetTextOverlay_BkColor](#) [SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#)
[SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#) [SetTextOverlay_GradientColor](#)
[SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#)
[SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#)
[SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#)
[SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#)
[ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#)
[TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#)
[TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#)
[TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#)
[TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#)
[TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Convert Your Markdown Content with HelpNDoc](#)

SetImageOverlay_ChromaKeyRGBColor

TVideoGrabber.SetImageOverlay_ChromaKeyRGBColor

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Methods](#)

Sets the RGB color used as chroma key for the specified image overlay

Declaration

procedure SetImageOverlay_ChromaKeyRGBColor (**Index**: LongInt; **Value**: LongInt);

void __fastcall SetImageOverlay_ChromaKeyRGBColor(**int** Index, **int** Value);

Description

Sets the RGB color used as chroma key for the image overlay of the specified index.

See the [Chroma Key](#) chapter.

See Also

[Image overlays](#) [TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#)
[TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetPixelsDistance](#)
[GetRGBPixelAt](#) [ImageOverlay_AlphaBlend](#) [ImageOverlay_AlphaBlendValue](#) [ImageOverlay_Height](#)
[ImageOverlay_LeftLocation](#) [ImageOverlay_StretchToVideoSize](#) [ImageOverlay_TopLocation](#)
[ImageOverlay_Transparent](#) [ImageOverlay_TransparentColorValue](#) [ImageOverlay_UseTransparentColor](#)
[ImageOverlay_VideoAlignment](#) [ImageOverlay_Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#)
[MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#)
[OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay_AlphaBlend](#)
[SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#)
[SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#)
[SetImageOverlay_LeftLocation](#) [SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#)
[SetImageOverlay_TargetDisplay](#) [SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#)
[SetImageOverlay_TransparentColorValue](#) [SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#)
[SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#)
[SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#)
[SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#)
[SetImageOverlayFromTImage2](#) [SetTextOverlay_Align](#) [SetTextOverlay_BkColor](#) [SetTextOverlay_CustomVar](#)
[SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#) [SetTextOverlay_GradientColor](#)
[SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#)
[SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#)
[SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#)
[SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#)
[ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#)
[TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#)
[TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#)
[TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#)
[TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#)
[TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Free help authoring tool](#)

SetImageOverlay_Enabled

TVideoGrabber.SetImageOverlay_Enabled

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Methods](#)

Enables / disables the specified image overlay

Declaration

procedure SetImageOverlay_Enabled (**Index**: LongInt; **Value**: Boolean);

void __fastcall SetImageOverlay_Enabled(**int** Index, **bool** Value);

Description

Used to enable/disable the image overlay with specified index.

Note: see the [Image overlays](#) chapter for global information about this feature.

See Also

[Image overlays](#) [TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetPixelsDistance](#) [GetRGBPixelAt](#) [ImageOverlay](#) [AlphaBlend](#) [ImageOverlay](#) [AlphaBlendValue](#) [ImageOverlay](#) [Height](#) [ImageOverlay](#) [LeftLocation](#) [ImageOverlay](#) [StretchToVideoSize](#) [ImageOverlay](#) [TopLocation](#) [ImageOverlay](#) [Transparent](#) [ImageOverlay](#) [TransparentColorValue](#) [ImageOverlay](#) [UseTransparentColor](#) [ImageOverlay](#) [VideoAlignment](#) [ImageOverlay](#) [Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronon](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#) [SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#) [SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#) [SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#) [SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#) [TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#) [TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#) [TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#) [TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Free PDF documentation generator](#)

SetImageOverlay_Height

TVideoGrabber.SetImageOverlay_Height

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets the height for the specified image overlay

Declaration

procedure SetImageOverlay_Height (**Index**: LongInt; **Value** :LongInt);

void __fastcall SetImageOverlay_Height(**int** Index, **int** Value);

Description

Sets the height for the image overlay of the specified index.

Set -1 (default) to use the original height of the loaded image.

Look at the [Image Overlays](#) chapter for more information.

See Also

[Image overlays](#) [TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#)
[TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetPixelsDistance](#)
[GetRGBPixelAt](#) [ImageOverlay](#) [AlphaBlend](#) [ImageOverlay](#) [AlphaBlendValue](#) [ImageOverlay](#) [Height](#)
[ImageOverlay](#) [LeftLocation](#) [ImageOverlay](#) [StretchToVideoSize](#) [ImageOverlay](#) [TopLocation](#)
[ImageOverlay](#) [Transparent](#) [ImageOverlay](#) [TransparentColorValue](#) [ImageOverlay](#) [UseTransparentColor](#)
[ImageOverlay](#) [VideoAlignment](#) [ImageOverlay](#) [Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#)
[MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#)
[OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#)
[SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#)
[SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#)
[SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#)
[SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#)
[SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#)
[SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#)
[SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#)
[SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#)
[SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#)
[SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#)
[SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#)
[ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#)
[TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#)
[TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#)
[TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#)
[TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#)
[TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [From Word to ePub or Kindle eBook: A Comprehensive Guide](#)

SetImageOverlay_LeftLocation**TVideoGrabber.SetImageOverlay_LeftLocation**[Prev](#)[Next](#)

TVideoGrabber [Methods](#)

Sets the left location for the specified image overlay

Declaration

procedure SetImageOverlay_LeftLocation (**Index**: LongInt; **Value** :LongInt);

void __fastcall SetImageOverlay_LeftLocation(int Index, int Value);

Description

Set the left location (on the video frames) for the image overlay of the specified index

The value must be between 0 and the current video width.

Look at the [Image Overlays](#) chapter for more information.

See Also

[Image overlays](#) [TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#)
[TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetPixelsDistance](#)
[GetRGBPixelAt](#) [ImageOverlay](#) [AlphaBlend](#) [ImageOverlay](#) [AlphaBlendValue](#) [ImageOverlay](#) [Height](#)
[ImageOverlay](#) [LeftLocation](#) [ImageOverlay](#) [StretchToVideoSize](#) [ImageOverlay](#) [TopLocation](#)
[ImageOverlay](#) [Transparent](#) [ImageOverlay](#) [TransparentColorValue](#) [ImageOverlay](#) [UseTransparentColor](#)
[ImageOverlay](#) [VideoAlignment](#) [ImageOverlay](#) [Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#)

[MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#)
[OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#)
[SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#)
[SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#)
[SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [RotationAngle](#)
[SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#)
[SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#)
[SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#)
[SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#)
[SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#)
[SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#)
[SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#)
[SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#)
[ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#)
[TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#)
[TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#)
[TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#)
[TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#)
[TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Easily create EPub books](#)

SetImageOverlay_RotationAngle

TVideoGrabber.SetImageOverlay_RotationAngle

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets a rotation angle for the specified image overlay

Declaration

procedure SetImageOverlay_RotationAngle (**Index**: LongInt; **Value**: Double);

void __fastcall SetImageOverlay_RotationAngle(**int** Index, **double** Value);

Description

Used to specify a rotation angle for the image overlay of the specified index

By default the rotation angle is 0.0, it can be any value between 0.0 and 360.0

See Also

[Image overlays](#) [TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#)
[TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetPixelsDistance](#)
[GetRGBPixelAt](#) [ImageOverlay](#) [AlphaBlend](#) [ImageOverlay](#) [AlphaBlendValue](#) [ImageOverlay](#) [Height](#)
[ImageOverlay](#) [LeftLocation](#) [ImageOverlay](#) [StretchToVideoSize](#) [ImageOverlay](#) [TopLocation](#)
[ImageOverlay](#) [Transparent](#) [ImageOverlay](#) [TransparentColorValue](#) [ImageOverlay](#) [UseTransparentColor](#)
[ImageOverlay](#) [VideoAlignment](#) [ImageOverlay](#) [Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#)
[MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#)
[OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#)
[SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#)
[SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#)
[SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#)
[SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#)
[SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#)
[SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)

[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#)
[SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#)
[SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#)
[SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#)
[SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#)
[SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#)
[ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#)
[TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#)
[TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#)
[TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#)
[TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#)
[TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Maximize Your Reach: Convert Your Word Document to an ePub or Kindle eBook

SetImageOverlay_StretchToVideoSize

TVideoGrabber.SetImageOverlay_StretchToVideoSize

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Stretches the specified image overlay to the size of the video frames

Declaration

procedure SetImageOverlay_StretchToVideoSize (**Index**: LongInt; **Value**: Boolean);

void __fastcall SetImageOverlay_StretchToVideoSize(**int** Index, **bool** Value);

Description

When enabled this property stretches the image overlay to the video size (in this case the image overlay location and size settings are ignored)

See [Graphic overlays](#)

See Also

[Image overlays](#) [TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#)
[TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetPixelsDistance](#)
[GetRGBPixelAt](#) [ImageOverlay](#) [AlphaBlend](#) [ImageOverlay](#) [AlphaBlendValue](#) [ImageOverlay](#) [Height](#)
[ImageOverlay](#) [LeftLocation](#) [ImageOverlay](#) [StretchToVideoSize](#) [ImageOverlay](#) [TopLocation](#)
[ImageOverlay](#) [Transparent](#) [ImageOverlay](#) [TransparentColorValue](#) [ImageOverlay](#) [UseTransparentColor](#)
[ImageOverlay](#) [VideoAlignment](#) [ImageOverlay](#) [Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#)
[MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#)
[OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#)
[SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#)
[SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#)
[SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#)
[SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#)
[SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#)
[SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#)
[SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#)
[SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#)
[SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#)
[SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#)
[SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#)

[ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#)
[TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#)
[TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#)
[TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#)
[TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#)
[TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Save time and frustration with HelpNDoc's WinHelp HLP to CHM conversion feature

SetImageOverlay_TargetDisplay

TVideoGrabber.SetImageOverlay_TargetDisplay

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Specifies the location of the overlay (frame or video window)

Declaration

procedure SetImageOverlay_TargetDisplay (**Index**: LongInt; **Value**: LongInt);

void __fastcall SetImageOverlay_TargetDisplay(**int** Index, **int** Value);

Description

Specifies if the overlay will be applied to the video frame or to the video window

See

See Also

[Image overlays](#) [TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#)
[TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetPixelsDistance](#)
[GetRGBPixelAt](#) [ImageOverlay](#) [AlphaBlend](#) [ImageOverlay](#) [AlphaBlendValue](#) [ImageOverlay](#) [Height](#)
[ImageOverlay](#) [LeftLocation](#) [ImageOverlay](#) [StretchToVideoSize](#) [ImageOverlay](#) [TopLocation](#)
[ImageOverlay](#) [Transparent](#) [ImageOverlay](#) [TransparentColorValue](#) [ImageOverlay](#) [UseTransparentColor](#)
[ImageOverlay](#) [VideoAlignment](#) [ImageOverlay](#) [Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#)
[MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#)
[OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#)
[SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#)
[SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#)
[SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#)
[SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TopLocation](#)
[SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#)
[SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#)
[SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#)
[SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#)
[SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#)
[SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#)
[SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#)
[ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#)
[TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#)
[TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#)
[TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#)
[TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#)
[TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Experience the Power and Ease of Use of HelpNDoc for CHM Help File Generation

SetImageOverlay_TopLocation

TVideoGrabber.SetImageOverlay_TopLocation

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets the top location for the specified image overlay

Declaration

procedure SetImageOverlay_TopLocation (**Index**: LongInt; **Value** :LongInt);

void __fastcall SetImageOverlay_TopLocation(**int** Index, **int** Value);

Description

Set the top location (on the video frames) for the image overlay of the specified index

The value must be between 0 and the current video height.

Look at the [Image Overlays](#) chapter for more information.

See Also

[Image overlays](#) [TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetPixelsDistance](#) [GetRGBPixelAt](#) [ImageOverlay](#) [AlphaBlend](#) [ImageOverlay](#) [AlphaBlendValue](#) [ImageOverlay](#) [Height](#) [ImageOverlay](#) [LeftLocation](#) [ImageOverlay](#) [StretchToVideoSize](#) [ImageOverlay](#) [TopLocation](#) [ImageOverlay](#) [Transparent](#) [ImageOverlay](#) [TransparentColorValue](#) [ImageOverlay](#) [UseTransparentColor](#) [ImageOverlay](#) [VideoAlignment](#) [ImageOverlay](#) [Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronon](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#) [SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#) [SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#) [SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#) [SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#) [TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#) [TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#) [TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#) [TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Effortlessly Edit and Export Markdown Documents

SetImageOverlay_Transparent

TVideoGrabber.SetImageOverlay_Transparent

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Enables the transparency for the specified image overlay

Declaration

procedure SetImageOverlay_Transparent (**Index**: LongInt; Value :Boolean);

void __fastcall SetImageOverlay_Transparent(**int** Index, **bool** Value);

Description

Used to enable the transparency for the image overlay of the specified index

Look at the [Image Overlays](#) chapter for more information.

See Also

[Image overlays](#) [TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetPixelsDistance](#) [GetRGBPixelAt](#) [ImageOverlay](#) [AlphaBlend](#) [ImageOverlay](#) [AlphaBlendValue](#) [ImageOverlay](#) [Height](#) [ImageOverlay](#) [LeftLocation](#) [ImageOverlay](#) [StretchToVideoSize](#) [ImageOverlay](#) [TopLocation](#) [ImageOverlay](#) [Transparent](#) [ImageOverlay](#) [TransparentColorValue](#) [ImageOverlay](#) [UseTransparentColor](#) [ImageOverlay](#) [VideoAlignment](#) [ImageOverlay](#) [Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronon](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#) [SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#) [SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#) [SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#) [SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#) [TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#) [TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#) [TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#) [TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Import and export Markdown documents](#)

SetImageOverlay_TransparentColorValue

TVideoGrabber.SetImageOverlay_TransparentColorValue

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets the transparency color for the specified image overlay

Declaration

procedure SetImageOverlay_TransparentColorValue (**Index**: LongInt; Value :LongInt);

void __fastcall SetImageOverlay_TransparentColorValue(**int** Index, **int** Value);

Description

Used to set the value of the color used for transparency when [SetImageOverlay_Transparent](#) and [SetImageOverlay_UseTransparentColor](#) have been enabled.

Look at the [Image Overlays](#) chapter for more information.

See Also

[Image overlays](#) [TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetPixelsDistance](#) [GetRGBPixelAt](#) [ImageOverlay](#) [AlphaBlend](#) [ImageOverlay](#) [AlphaBlendValue](#) [ImageOverlay](#) [Height](#) [ImageOverlay](#) [LeftLocation](#) [ImageOverlay](#) [StretchToVideoSize](#) [ImageOverlay](#) [TopLocation](#) [ImageOverlay](#) [Transparent](#) [ImageOverlay](#) [TransparentColorValue](#) [ImageOverlay](#) [UseTransparentColor](#) [ImageOverlay](#) [VideoAlignment](#) [ImageOverlay](#) [Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#) [SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#) [SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#) [SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#) [SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#) [TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#) [TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#) [TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#) [TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Maximize Your Productivity with HelpNDoc's Efficient User Interface

SetImageOverlay_UseTransparentColor

TVideoGrabber.SetImageOverlay_UseTransparentColor

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Enables the use of a transparency color for the specified image overlay

Declaration

procedure SetImageOverlay_UseTransparentColor (**Index**: LongInt; **Value** :Boolean);

void __fastcall SetImageOverlay_UseTransparentColor(**int** Index, **bool** Value);

Description

Used to activate the transparent color for the image overlay of the specified index.

The overlay transparency must be enabled with [SetImageOverlay Transparent](#)

If enabled, the color must be specified by [SetImageOverlay TransparentColorValue](#).

If disabled, the default color transparency (background color) of the image will be used.

Look at the [Image Overlays](#) chapter for more information.

See Also

[Image overlays](#) [TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetPixelsDistance](#) [GetRGBPixelAt](#) [ImageOverlay](#) [AlphaBlend](#) [ImageOverlay](#) [AlphaBlendValue](#) [ImageOverlay](#) [Height](#) [ImageOverlay](#) [LeftLocation](#) [ImageOverlay](#) [StretchToVideoSize](#) [ImageOverlay](#) [TopLocation](#) [ImageOverlay](#) [Transparent](#) [ImageOverlay](#) [TransparentColorValue](#) [ImageOverlay](#) [UseTransparentColor](#) [ImageOverlay](#) [VideoAlignment](#) [ImageOverlay](#) [Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#) [SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#) [SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#) [SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#) [SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#) [TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#) [TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#) [TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#) [TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Import and export Markdown documents

SetImageOverlay_Width

TVideoGrabber.SetImageOverlay_Width

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets the width for the specified image overlay

Declaration

procedure SetImageOverlay_Width (**Index**: LongInt; **Value** :LongInt);

void __fastcall SetImageOverlay_Width(int Index, int Value);

Description

Sets the width for the image overlay of the specified index.

Set -1 (default) to use the original width of the loaded image.

Look at the [Image Overlays](#) chapter for more information.

See Also

[Image overlays](#) [TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetPixelsDistance](#) [GetRGBPixelAt](#) [ImageOverlay](#) [AlphaBlend](#) [ImageOverlay](#) [AlphaBlendValue](#) [ImageOverlay](#) [Height](#) [ImageOverlay](#) [LeftLocation](#) [ImageOverlay](#) [StretchToVideoSize](#) [ImageOverlay](#) [TopLocation](#) [ImageOverlay](#) [Transparent](#) [ImageOverlay](#) [TransparentColorValue](#) [ImageOverlay](#) [UseTransparentColor](#) [ImageOverlay](#) [VideoAlignment](#) [ImageOverlay](#) [Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#)

[SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#)
[SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#)
[SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#)
[SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#)
[SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#)
[SetImageOverlay_UseTransparentColor](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#)
[SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#)
[SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#)
[SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay_Align](#) [SetTextOverlay_BkColor](#)
[SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#)
[SetTextOverlay_GradientColor](#) [SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#)
[SetTextOverlay_Left](#) [SetTextOverlay_Right](#) [SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#)
[SetTextOverlay_Shadow](#) [SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#)
[SetTextOverlay_String](#) [SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#)
[ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#)
[TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#)
[TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#)
[TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#)
[TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#)
[TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Documentation Capabilities with a Help Authoring Tool](#)

SetImageOverlayFromBMPFile

TVideoGrabber.SetImageOverlayFromBMPFile

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Deprecated, use [SetImageOverlayFromImageFile](#) or [SetImageOverlayFromImageFile2](#)

Declaration

function SetImageOverlayFromBMPFile(FileName: **string**): Boolean;

bool __fastcall SetImageOverlayFromBMPFile(wchar_t *FileName)

Function SetImageOverlayFromBMPFile(FileName As String) As Boolean

Description

Deprecated, kept for backward compatibility, see [SetImageOverlayFromImageFile](#) or [SetImageOverlayFromImageFile2](#)

See Also

[Image overlays](#) [TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#)
[TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetPixelsDistance](#)
[GetRGBPixelAt](#) [ImageOverlay_AlphaBlend](#) [ImageOverlay_AlphaBlendValue](#) [ImageOverlay_Height](#)
[ImageOverlay_LeftLocation](#) [ImageOverlay_StretchToVideoSize](#) [ImageOverlay_TopLocation](#)
[ImageOverlay_Transparent](#) [ImageOverlay_TransparentColorValue](#) [ImageOverlay_UseTransparentColor](#)
[ImageOverlay_VideoAlignment](#) [ImageOverlay_Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#)
[MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEODHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#)
[OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay_AlphaBlend](#)
[SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#)
[SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#)
[SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#)
[SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#)
[SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#)
[SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromHBitmap](#)
[SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#)
[SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#)
[SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay_Align](#) [SetTextOverlay_BkColor](#)

[SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#) [SetTextOverlay_GradientColor](#) [SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#) [SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#) [SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#) [SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#) [TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#) [TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#) [TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#) [TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Modernize your help files with HelpNDoc's WinHelp HLP to CHM conversion tool](#)

SetImageOverlayFromBMPFile2

TVideoGrabber.SetImageOverlayFromBMPFile2

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Deprecated, use [SetImageOverlayFromImageFile](#) or [SetImageOverlayFromImageFile2](#)

Declaration

function SetImageOverlayFromBMPFile2(**Index**: LongInt; **FileName**: **string**): Boolean;

bool __fastcall SetImageOverlayFromBMPFile2(int Index, wchar_t *FileName)

Function SetImageOverlayFromBMPFile2 (Index as Long, FileName As String) As Boolean

Description

Deprecated, kept for backward compatibility, see [SetImageOverlayFromImageFile](#) or [SetImageOverlayFromImageFile2](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Publish Your Word Document as an eBook](#)

SetImageOverlayFromHBitmap

TVideoGrabber.SetImageOverlayFromHBitmap

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Loads the current image overlay from a bitmap handle.

Declaration

function SetImageOverlayFromHBitmap(Bitmap: HBITMAP): Boolean;

bool __fastcall SetImageOverlayFromHBitmap(HBITMAP Bitmap)

Function SetImageOverlayFromHBitmap(FileName As String) As Boolean

Description

Used to load the image overlay currently selected by [ImageOverlaySelector](#) from a bitmap handle.

Note: see the [Image overlays](#) chapter for global information about this feature.

See Also

[Image overlays](#) [TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetPixelsDistance](#) [GetRGBPixelAt](#) [ImageOverlay](#) [AlphaBlend](#) [ImageOverlay](#) [AlphaBlendValue](#) [ImageOverlay](#) [Height](#) [ImageOverlay](#) [LeftLocation](#) [ImageOverlay](#) [StretchToVideoSize](#) [ImageOverlay](#) [TopLocation](#)

[ImageOverlay_Transparent](#) [ImageOverlay_TransparentColorValue](#) [ImageOverlay_UseTransparentColor](#) [ImageOverlay_VideoAlignment](#) [ImageOverlay_Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronon](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay_AlphaBlend](#) [SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#) [SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#) [SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#) [SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#) [SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#) [SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay_Align](#) [SetTextOverlay_BkColor](#) [SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#) [SetTextOverlay_GradientColor](#) [SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#) [SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#) [SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#) [SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#) [TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#) [TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#) [TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#) [TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Generate Kindle eBooks with ease](#)

SetImageOverlayFromHBitmap2

TVideoGrabber.SetImageOverlayFromHBitmap2

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Loads the specified image overlay from a bitmap handle.

Declaration

function SetImageOverlayFromHBitmap2(Index: LongInt; Bitmap: HBITMAP): Boolean;

bool __fastcall SetImageOverlayFromHBitmap2(int Index, HBITMAP Bitmap)

Function SetImageOverlayFromHBitmap2(Index as Long, FileName As String) As Boolean

Description

Used to load the image overlay specified by the index parameter from the specified bitmap handle.

Note: see the [Image overlays](#) chapter for global information about this feature.

See Also

[Image overlays](#) [TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetPixelsDistance](#) [GetRGBPixelAt](#) [ImageOverlay_AlphaBlend](#) [ImageOverlay_AlphaBlendValue](#) [ImageOverlay_Height](#) [ImageOverlay_LeftLocation](#) [ImageOverlay_StretchToVideoSize](#) [ImageOverlay_TopLocation](#) [ImageOverlay_Transparent](#) [ImageOverlay_TransparentColorValue](#) [ImageOverlay_UseTransparentColor](#) [ImageOverlay_VideoAlignment](#) [ImageOverlay_Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronon](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay_AlphaBlend](#) [SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#) [SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#) [SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#) [SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#)

[SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#)
[SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#)
[SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#)
[SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay_Align](#) [SetTextOverlay_BkColor](#)
[SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#)
[SetTextOverlay_GradientColor](#) [SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#)
[SetTextOverlay_Left](#) [SetTextOverlay_Right](#) [SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#)
[SetTextOverlay_Shadow](#) [SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#)
[SetTextOverlay_String](#) [SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#)
[ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#)
[TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#)
[TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#)
[TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#)
[TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#)
[TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Qt Help documentation made easy](#)

SetImageOverlayFromImageFile

TVideoGrabber.SetImageOverlayFromImageFile

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Loads the current image overlay from a file

Declaration

function SetImageOverlayFromImageFile (FileName: **string**): Boolean;

bool __fastcall SetImageOverlayFromImageFile(wchar_t *FileName);

Function SetImageOverlayFromImageFile(FileName as String) as Bool

Description

Used to load the image overlay currently selected by [ImageOverlaySelector](#) from the specified file

Various image formats are supported.

See the [Image overlays](#) chapter for more information about overlays.

See Also

[Image overlays](#) [TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#)
[TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetPixelsDistance](#)
[GetRGBPixelAt](#) [ImageOverlay_AlphaBlend](#) [ImageOverlay_AlphaBlendValue](#) [ImageOverlay_Height](#)
[ImageOverlay_LeftLocation](#) [ImageOverlay_StretchToVideoSize](#) [ImageOverlay_TopLocation](#)
[ImageOverlay_Transparent](#) [ImageOverlay_TransparentColorValue](#) [ImageOverlay_UseTransparentColor](#)
[ImageOverlay_VideoAlignment](#) [ImageOverlay_Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#)
[MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#)
[OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay_AlphaBlend](#)
[SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#)
[SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#)
[SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#)
[SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#)
[SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#)
[SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile2](#)
[SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#)
[SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay_Align](#) [SetTextOverlay_BkColor](#)
[SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#)
[SetTextOverlay_GradientColor](#) [SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#)

[SetTextOverlay_Left](#) [SetTextOverlay_Right](#) [SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#)
[SetTextOverlay_Shadow](#) [SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#)
[SetTextOverlay_String](#) [SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#)
[ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#)
[TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#)
[TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#)
[TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#)
[TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#)
[TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Make CHM Help File Creation a Breeze with HelpNDoc](#)

SetImageOverlayFromImageFile2

TVideoGrabber.SetImageOverlayFromImageFile2

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Loads the image overlay specified by the Index parameter from a file

Declaration

function SetImageOverlayFromImageFile2 (Index: LongInt; FileName: string): Boolean;

bool __fastcall SetImageOverlayFromImageFile2 (int Index, wchar_t *FileName);

Function SetImageOverlayFromImageFile2 (Index as Long, FileName as String) as Bool

Description

Used to load the image overlay currently selected by the Index parameter from the specified file.

Various image formats are supported.

See the [Image overlays](#) chapter for more information about overlays.

See Also

[Image overlays](#) [TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#)
[TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetPixelsDistance](#)
[GetRGBPixelAt](#) [ImageOverlay](#) [AlphaBlend](#) [ImageOverlay_AlphaBlendValue](#) [ImageOverlay_Height](#)
[ImageOverlay_LeftLocation](#) [ImageOverlay_StretchToVideoSize](#) [ImageOverlay_TopLocation](#)
[ImageOverlay_Transparent](#) [ImageOverlay_TransparentColorValue](#) [ImageOverlay_UseTransparentColor](#)
[ImageOverlay_VideoAlignment](#) [ImageOverlay_Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#)
[MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#)
[OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay_AlphaBlend](#)
[SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#)
[SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#)
[SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#)
[SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#)
[SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#)
[SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#)
[SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay_Align](#) [SetTextOverlay_BkColor](#)
[SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#)
[SetTextOverlay_GradientColor](#) [SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#)
[SetTextOverlay_Left](#) [SetTextOverlay_Right](#) [SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#)
[SetTextOverlay_Shadow](#) [SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#)
[SetTextOverlay_String](#) [SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#)
[ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#)
[TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#)
[TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#)
[TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#)

[TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#)
[TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Effortlessly Convert Your Word Doc to an eBook: A Step-by-Step Guide

SetImageOverlayFromJPEGFile

TVideoGrabber.SetImageOverlayFromJPEGFile

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Deprecated, use [SetImageOverlayFromImageFile](#) or [SetImageOverlayFromImageFile2](#)

Declaration

function SetImageOverlayFromJPEGFile(Bitmap: HBITMAP): Boolean;

bool __fastcall SetImageOverlayFromJPEGFile(wchar_t *FileName)

Function SetImageOverlayFromJPEGFile(FileName As String) As Boolean

Description

Deprecated, kept for backward compatibility, see [SetImageOverlayFromImageFile](#) or [SetImageOverlayFromImageFile2](#)

See Also

[Image overlays](#) [TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetPixelsDistance](#) [GetRGBPixelAt](#) [ImageOverlay_AlphaBlend](#) [ImageOverlay_AlphaBlendValue](#) [ImageOverlay_Height](#) [ImageOverlay_LeftLocation](#) [ImageOverlay_StretchToVideoSize](#) [ImageOverlay_TopLocation](#) [ImageOverlay_Transparent](#) [ImageOverlay_TransparentColorValue](#) [ImageOverlay_UseTransparentColor](#) [ImageOverlay_VideoAlignment](#) [ImageOverlay_Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay_AlphaBlend](#) [SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#) [SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#) [SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#) [SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#) [SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#) [SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay_Align](#) [SetTextOverlay_BkColor](#) [SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#) [SetTextOverlay_GradientColor](#) [SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#) [SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#) [SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#) [SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#) [TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#) [TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#) [TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#) [TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Effortlessly Create High-Quality Help Documentation with a Help Authoring Tool

SetImageOverlayFromJPEGFile2

TVideoGrabber.SetImageOverlayFromJPEGFile2

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Methods](#)

Deprecated, use [SetImageOverlayFromImageFile](#) or [SetImageOverlayFromImageFile2](#)

Declaration

function SetImageOverlayFromJPEGFile(**Index**: LongInt; **Bitmap**: HBITMAP): Boolean;

bool __fastcall SetImageOverlayFromJPEGFile(int Index, wchar_t *FileName)

Function SetImageOverlayFromJPEGFile(Index as Long, FileName As String) As Boolean

Description

Deprecated, kept for backward compatibility, see [SetImageOverlayFromImageFile](#) or [SetImageOverlayFromImageFile2](#)

Created with the Standard Edition of HelpNDoc: [Experience the Power and Ease of Use of HelpNDoc for CHM Help File Generation](#)

SetImageOverlayFromTBitmap

TVideoGrabber.SetImageOverlayFromTBitmap

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Methods](#)

Loads the current image overlay from a TBitmap (available only in the Delphi and C++Builder versions)

Declaration

function SetImageOverlayFromTBitmap(**Bitmap**: TBitmap): Boolean;

bool __fastcall SetImageOverlayFromTBitmap(TBitmap *Bitmap)

Description

Used to load the image overlay currently selected by [ImageOverlaySelector](#) from a TBitmap component (available only in the Delphi and C++Builder versions)

Note: see the [Image overlays](#) chapter for global information about this feature.

See Also

[Image overlays](#) [TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetPixelsDistance](#) [GetRGBPixelAt](#) [ImageOverlay](#) [AlphaBlend](#) [ImageOverlay](#) [AlphaBlendValue](#) [ImageOverlay](#) [Height](#) [ImageOverlay](#) [LeftLocation](#) [ImageOverlay](#) [StretchToVideoSize](#) [ImageOverlay](#) [TopLocation](#) [ImageOverlay](#) [Transparent](#) [ImageOverlay](#) [TransparentColorValue](#) [ImageOverlay](#) [UseTransparentColor](#) [ImageOverlay](#) [VideoAlignment](#) [ImageOverlay](#) [Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#)

[SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#)
[SetTextOverlay_GradientColor](#) [SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#)
[SetTextOverlay_Left](#) [SetTextOverlay_Right](#) [SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#)
[SetTextOverlay_Shadow](#) [SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#)
[SetTextOverlay_String](#) [SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#)
[ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#)
[TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#)
[TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#)
[TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#)
[TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#)
[TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Generate Kindle eBooks with ease](#)

SetImageOverlayFromTBitmap2

TVideoGrabber.SetImageOverlayFromTBitmap2

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Loads the specified image overlay from a TBitmap object (Dephi and C++Builder only)

Declaration

function SetImageOverlayFromTBitmap2(Index: LongInt; Bitmap: TBitmap): Boolean;

bool __fastcall SetImageOverlayFromTBitmap2(int Index, TBitmap *Bitmap)

Description

Used to load the image overlay specified by the index parameter from the specified TBitmap object (Dephi and C++Builder only)

See Also

[Image overlays](#) [TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#)
[TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetPixelsDistance](#)
[GetRGBPixelAt](#) [ImageOverlay_AlphaBlend](#) [ImageOverlay_AlphaBlendValue](#) [ImageOverlay_Height](#)
[ImageOverlay_LeftLocation](#) [ImageOverlay_StretchToVideoSize](#) [ImageOverlay_TopLocation](#)
[ImageOverlay_Transparent](#) [ImageOverlay_TransparentColorValue](#) [ImageOverlay_UseTransparentColor](#)
[ImageOverlay_VideoAlignment](#) [ImageOverlay_Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#)
[MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronon](#) [OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#)
[OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay_AlphaBlend](#)
[SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#)
[SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#)
[SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#)
[SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#)
[SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#)
[SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay_Align](#) [SetTextOverlay_BkColor](#)
[SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#)
[SetTextOverlay_GradientColor](#) [SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#)
[SetTextOverlay_Left](#) [SetTextOverlay_Right](#) [SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#)
[SetTextOverlay_Shadow](#) [SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#)
[SetTextOverlay_String](#) [SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#)
[ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#)
[TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#)
[TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#)
[TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#)
[TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#)
[TranslateMouseCoordinates](#)

SetImageOverlayFromTImage

TVideoGrabber.SetImageOverlayFromTImage

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Loads the current image overlay from a TImage component (available only in the Delphi and C++Builder versions)

Declaration

function SetImageOverlayFromTImage(Image: TImage): Boolean;

bool __fastcall SetImageOverlayFromTImage(TImage *Image)

Description

Used to load the image overlay currently selected by [ImageOverlaySelector](#) from a TImage component (available only in the Delphi and C++Builder versions)

Note: see the [Image overlays](#) chapter for global information about this feature.

See Also

[Image overlays](#) [TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetPixelsDistance](#) [GetRGBPixelAt](#) [ImageOverlay](#) [AlphaBlend](#) [ImageOverlay](#) [AlphaBlendValue](#) [ImageOverlay](#) [Height](#) [ImageOverlay](#) [LeftLocation](#) [ImageOverlay](#) [StretchToVideoSize](#) [ImageOverlay](#) [TopLocation](#) [ImageOverlay](#) [Transparent](#) [ImageOverlay](#) [TransparentColorValue](#) [ImageOverlay](#) [UseTransparentColor](#) [ImageOverlay](#) [VideoAlignment](#) [ImageOverlay](#) [Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#) [SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#) [SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#) [SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#) [SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#) [TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#) [TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#) [TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#) [TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#) [TranslateMouseCoordinates](#)

SetImageOverlayFromTImage2

TVideoGrabber.SetImageOverlayFromTImage2

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

ber

Loads the specified image overlay from a TImage object (Dephi and C++Builder only)

Declaration

function SetImageOverlayFromTImage2(Image: TImage): Boolean;

bool __fastcall SetImageOverlayFromTImage2(TImage *Image)

Description

Used to load the image overlay specified by the index parameter from the specified TImage object (Dephi and C++Builder only)

See Also

[Image overlays](#) [TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetPixelsDistance](#) [GetRGBPixelAt](#) [ImageOverlay](#) [AlphaBlend](#) [ImageOverlay](#) [AlphaBlendValue](#) [ImageOverlay](#) [Height](#) [ImageOverlay](#) [LeftLocation](#) [ImageOverlay](#) [StretchToVideoSize](#) [ImageOverlay](#) [TopLocation](#) [ImageOverlay](#) [Transparent](#) [ImageOverlay](#) [TransparentColorValue](#) [ImageOverlay](#) [UseTransparentColor](#) [ImageOverlay](#) [VideoAlignment](#) [ImageOverlay](#) [Width](#) [ImageOverlayEnabled](#) [ImageOverlaySelector](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#) [SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#) [SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#) [SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#) [SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#) [TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#) [TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#) [TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#) [TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Free PDF documentation generator](#)

SetIPCameraSetting**TVideoGrabber.SetIPCameraSetting**
[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets camera IP settings**Declaration**

function SetIPCameraSetting (Setting: TIPCameraSetting; Value: LongInt): Boolean;

bool __fastcall SetIPCameraSetting(TIPCameraSetting Setting, **int** Value);

function SetIPCameraSetting (Setting as TxIPCameraSetting, Value as Long) as Bool

Description

Used to adjust IP camera settings

The setting property accepts the following settings:

ips_ConnectionTimeout:

Specifies the connection timeout (when connecting to the IP camera) expressed in milliseconds.

Default value: 10000 (= 10 seconds)

Minimum value: 500 (0.5 second).

If the value is too low you may experience connection failure problems.

E.g. SetIPCameraSetting (ips_ConnectionTimeout, 5000) sets a connection timeout of 5 seconds

ips_ReceiveTimeout:

Specifies the receive timeout (after the camera preview or recording started) to the IP camera expressed in milliseconds.

Default value: 5000 (= 5 seconds)

Minimum value: 500 (0.5 second).

If the value is too low you may experience unexpected disconnection problems.

E.g. SetIPCameraSetting (ips_ReceiveTimeout, 1000) sets a receive timeout of 1 second

See Also

[IR Cut Filter of Axis cameras](#) [TDiscoveryCallbackStatus](#) [TOnONVIFDiscoveryCompletedNotification](#) [Video stream of IP camera with audio of PC microphone or other audio capture device](#) [ExtraDLLPath](#) [GetLastError](#) [IPCameraURL](#) [IsURLResponding](#) [IsURLVideoStreamAvailable](#) [OnONVIFDiscoveryCompletedNotification](#) [ONVIF](#) [GetStr](#) [ONVIF](#) [SetStr](#) [ONVIFCancelDiscovery](#) [ONVIFDiscoverCameras](#) [IPRange](#) [ONVIFDiscoverCameras](#) [Multicast](#) [ONVIFEnumCamerasDiscovered](#) [OpenURLAsync](#) [OpenURLAsyncStatus](#) [SendIPCameraCommand](#) [SetDatasteadFilterDllName](#)

Created with the Standard Edition of HelpNDoc: [Free help authoring environment](#)

SetLocation

TVideoGrabber.SetLocation

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Set the location of the video window within the parent control

Declaration

n/a

void SetLocation (int ILeft, int ITop, int IWidth, int IHeight);

Description

Used to set the location of the video window within the parent control

(not available in Delphi / CppBuilder, the component resizes automatically when the parent control is resized)

E.g.:

```
CMFCFormDlg::CMFCFormDlg(CWnd* pParent /*=nullptr*/)
: CDialogEx(IDD_MFCFORM_DIALOG, pParent)
, m_VideoGrabber(NULL)
m_hIcon = AfxGetApp()->LoadIcon(IDR_MAINFRAME);
m_VideoGrabber = new CVideoGrabber(this);
CMFCFormDlg::~CMFCFormDlg()
delete m_VideoGrabber;
BOOL CMFCFormDlg::OnInitDialog()
CDialogEx::OnInitDialog();
m_VideoGrabber->SetParentWindow(this->GetSafeHwnd());
...
return TRUE; // return TRUE unless you set the focus to a control
```

```
void CMFCFormDlg::OnSize(UINT nType, int w, int h)
m_VideoGrabber->SetLocation(0, 0, w, h);
void CMFCFormDlg::OnBnClickedOk()
m_VideoGrabber->SetVideoSource(TVideoSource::vs_ScreenRecording);
m_VideoGrabber->StartPreview();
```

Created with the Standard Edition of HelpNDoc: [Write eBooks for the Kindle](#)

SetLogoFromBMPFile

TVideoGrabber.SetLogoFromBMPFile

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Set the video logo from a BMP file.

Declaration

function SetLogoFromBMPFile(FileName: **string**): Boolean;

bool __fastcall SetLogoFromBMPFile(wchar_t *FileName)

Function SetLogoFromBMPFile(FileName As String) As Boolean

Description

Used to set the logo displayed in the inactive video window, by using the specified BMP file.

See Also

[Logo displayed in the video window](#) [LogoDisplayed](#) [LogoLayout](#) [SetLogoFromHBitmap](#)
[SetLogoFromJPEGFile](#) [SetLogoFromTBitmap](#)

Created with the Standard Edition of HelpNDoc: [Easy to use tool to create HTML Help files and Help web sites](#)

SetLogoFromHBitmap

TVideoGrabber.SetLogoFromHBitmap

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Set the video logo from a bitmap handle.

Declaration

function SetLogoFromHBitmap(Bitmap: HBITMAP): Boolean;

bool __fastcall SetLogoFromHBitmap(HBITMAP Bitmap)

Function SetLogoFromHBitmap(Bitmap As Long) As Boolean

Description

Used to set the logo displayed in the inactive video window, by using the specified bitmap handle.

See Also

[Logo displayed in the video window](#) [LogoDisplayed](#) [LogoLayout](#) [SetLogoFromBMPFile](#)
[SetLogoFromJPEGFile](#) [SetLogoFromTBitmap](#)

Created with the Standard Edition of HelpNDoc: [Make the switch to CHM with HelpNDoc's hassle-free WinHelp HLP to CHM conversion tool](#)

SetLogoFromJPEGFile

TVideoGrabber.SetLogoFromJPEGFile

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Set the video logo from a JPEG file.

Declaration

function SetLogoFromJPEGFile(FileName: **string**): Boolean;

bool __fastcall SetLogoFromJPEGFile(wchar_t *FileName)

Function SetLogoFromJPEGFile(FileName As String) As Boolean

Description

Used to set the logo displayed in the inactive video window, by using the specified JPEG file.

See Also

[Logo displayed in the video window](#) [LogoDisplayed](#) [LogoLayout](#) [SetLogoFromBMPFile](#) [SetLogoFromHBitmap](#) [SetLogoFromTBitmap](#)

Created with the Standard Edition of HelpNDoc: [Easily create EPub books](#)

SetLogoFromTBitmap

TVideoGrabber.SetLogoFromTBitmap

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Set the video logo from a TBitmap component.

Declaration

function SetLogoFromTBitmap(Bitmap: TBitmap): Boolean;

bool __fastcall SetLogoFromTBitmap(Graphics::TBitmap *Bitmap)

n/a

Description

Used to set the logo displayed in the inactive video window, by using the specified TBitmap component.

See Also

[Logo displayed in the video window](#) [LogoDisplayed](#) [LogoLayout](#) [SetLogoFromBMPFile](#) [SetLogoFromHBitmap](#) [SetLogoFromJPEGFile](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly optimize your documentation website for search engines](#)

SetLogoFromTImage

TVideoGrabber.SetLogoFromTImage

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Set the video logo from a TImage component (Delphi and C++Builder versions only).

Declaration

function SetLogoFromImage(Image: TImage): Boolean;

bool __fastcall SetLogoFromImage(ExtCtrls::TImage *Image)

n/a

Description

Used to set the logo displayed in the inactive video window, by using the specified TImage component (Delphi and C++Builder versions only).

See Also

[Logo displayed in the video window](#) [LogoDisplayed](#) [LogoLayout](#) [SetLogoFromBMPFile](#) [SetLogoFromHBitmap](#) [SetLogoFromJPEGFile](#) [SetLogoFromTBitmap](#)

Created with the Standard Edition of HelpNDoc: [iPhone web sites made easy](#)

SetMultiplexerFilterByName

TVideoGrabber.SetMultiplexerFilterByName

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Forces the use of a MPEG multiplexer

Declaration

procedure SetMultiplexerFilterByName(MultiplexerFilterName: **string**);

void __fastcall SetMultiplexerFilterByName(System::wchar_t *MultiplexerFilterName)

Sub SetMultiplexerFilterByName (MultiplexerFilterName as String)

Description

When a MPEG video compressor and a MPEG audio compressor are selected, TVideoGrabber tries to use the corresponding multiplexer (from the same manufacturer).

However you can tell TVideoGrabber what multiplexer to use with this function.

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last](#) [Recording](#) [FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [Streamline Your CHM Help File Creation with HelpNDoc](#)

SetParentWindow

TVideoGrabber.SetParentWindow

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Used to set a parent window

Declaration

procedure SetParentWindow(Value: HWnd);

void __fastcall SetParentWindow(HWND Value)

Description

This function is exposed by the Visual Studio C++ versions to let you attach the control to a parent window programmatically.

See Also

[TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display](#) [Active](#) [Display](#) [AlphaBlendEnabled](#) [Display](#) [AlphaBlendValue](#) [Display](#) [AutoSize](#) [Display](#) [Embedded](#) [Display](#) [FullScreen](#) [Display](#) [Height](#) [Display](#) [Left](#) [Display](#) [Monitor](#) [Display](#) [MouseMovesWindow](#) [Display](#) [PanScanRatio](#) [Display](#) [SetLocation](#) [Display](#) [StayOnTop](#) [Display](#) [Top](#) [Display](#) [TransparentColorEnabled](#) [Display](#) [TransparentColorValue](#) [Display](#) [VideoHeight](#) [Display](#) [VideoPortEnabled](#) [Display](#) [VideoWidth](#) [Display](#) [VideoWindowHandle](#) [Display](#) [Width](#) [DualDisplay](#) [Active](#) [DualDisplay](#) [AlphaBlendEnabled](#) [DualDisplay](#) [AlphaBlendValue](#) [DualDisplay](#) [AutoSize](#) [DualDisplay](#) [Embedded](#) [DualDisplay](#) [FullScreen](#) [DualDisplay](#) [Height](#) [DualDisplay](#) [Left](#) [DualDisplay](#) [Monitor](#) [DualDisplay](#) [MouseMovesWindow](#) [DualDisplay](#) [PanScanRatio](#) [DualDisplay](#) [StayOnTop](#) [DualDisplay](#) [Top](#) [DualDisplay](#) [TransparentColorEnabled](#) [DualDisplay](#) [TransparentColorValue](#) [DualDisplay](#) [VideoHeight](#) [DualDisplay](#) [VideoPortEnabled](#) [DualDisplay](#) [VideoWidth](#) [DualDisplay](#) [VideoWindowHandle](#) [DualDisplay](#) [Visible](#) [DualDisplay](#) [Width](#) [IsVideoPortAvailable](#) [Monitor](#) [Primary](#) [Index](#) [MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: Revolutionize Your Documentation Output with HelpNDoc's Stunning User Interface

SetTextOverlay_Align**TVideoGrabber.SetTextOverlay_Align**
[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets or retrieves the alignment used to draw text over video frames for the specified text overlay

Declaration

procedure SetTextOverlay_Align (Index: LongInt; Value: TTextOverlayAlign);

void __fastcall SetTextOverlay_Align(int Index, TTextOverlayAlign Value);

Description

Used to set or retrieve the alignment used to draw text over video frames.
The text will be drawn between [TextOverlay_Left](#) and [TextOverlay_Right](#) positions.

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetTextOverlay](#) [AlphaBlend](#) [GetTextOverlay](#) [AlphaBlendValue](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)

[SetTextOverlay_AlphaBlend](#)
[SetTextOverlay_AlphaBlendValue](#)
[SetTextOverlay_BkColor](#)
[SetTextOverlay_CustomVar](#)
[SetTextOverlay_Enabled](#)
[SetTextOverlay_Font](#)
[SetTextOverlay_FontColor](#)
[SetTextOverlay_GradientColor](#)
[SetTextOverlay_GradientMode](#)
[SetTextOverlay_HighResFont](#)
[SetTextOverlay_Left](#)
[SetTextOverlay_Right](#)
[SetTextOverlay_Scrolling](#)
[SetTextOverlay_ScrollingSpeed](#)
[SetTextOverlay_Shadow](#)
[SetTextOverlay_ShadowColor](#)
[SetTextOverlay_ShadowDirection](#)
[SetTextOverlay_String](#)
[SetTextOverlay_TargetDisplay](#)
[SetTextOverlay_Top](#)
[SetTextOverlay_Transparent](#)
[ShapeOverlay](#)
[ShapeOverlayEnabled](#)
[ShapeOverlayList](#)
[TextOverlay_Align](#)
[TextOverlay_BkColor](#)
[TextOverlay_CreateCustomFont](#)
[TextOverlay_CreateCustomFont2](#)
[TextOverlay_Enabled](#)
[TextOverlay_Font](#)
[TextOverlay_FontColor](#)
[TextOverlay_Left](#)
[TextOverlay_Right](#)
[TextOverlay_Scrolling](#)
[TextOverlay_ScrollingSpeed](#)
[TextOverlay_Selector](#)
[TextOverlay_Shadow](#)
[TextOverlay_ShadowColor](#)
[TextOverlay_ShadowDirection](#)
[TextOverlay_String](#)
[TextOverlay_Top](#)
[TextOverlay_Transparent](#)
[TextOverlay_VideoAlignment](#)
[TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Upgrade your help files and your workflow with HelpNDoc's WinHelp HLP to CHM conversion](#)

SetTextOverlay_AlphaBlend

TVideoGrabber.SetTextOverlay_AlphaBlend

[Prev](#)
[Next](#)

[TVideoGrabber](#)
[Methods](#)

Enables the alpha blending for the specified text overlay

Declaration

procedure SetTextOverlay_AlphaBlend (**Index**: LongInt; **Value** :Boolean);

void __fastcall SetTextOverlay_AlphaBlend(**int** Index, **bool** Value);

Description

Enables the alpha blending for the text overlay of the specified index

The alpha blending value must be specified with [SetTextOverlay_AlphaBlend](#)

Look at the [Text Overlays](#) chapter for more information.

See Also

[GetTextOverlay_AlphaBlend](#)
[GetTextOverlay_AlphaBlendValue](#)
[SetTextOverlay_Align](#)
[SetTextOverlay_AlphaBlendValue](#)
[SetTextOverlay_BkColor](#)
[SetTextOverlay_CustomVar](#)
[SetTextOverlay_Enabled](#)
[SetTextOverlay_Font](#)
[SetTextOverlay_FontColor](#)
[SetTextOverlay_GradientColor](#)
[SetTextOverlay_GradientMode](#)
[SetTextOverlay_HighResFont](#)
[SetTextOverlay_Left](#)
[SetTextOverlay_Right](#)
[SetTextOverlay_Scrolling](#)
[SetTextOverlay_ScrollingSpeed](#)
[SetTextOverlay_Shadow](#)
[SetTextOverlay_ShadowColor](#)
[SetTextOverlay_ShadowDirection](#)
[SetTextOverlay_String](#)
[SetTextOverlay_TargetDisplay](#)
[SetTextOverlay_Top](#)
[SetTextOverlay_Transparent](#)
[TextOverlay_CreateCustomFont](#)
[TextOverlay_CreateCustomFont2](#)
[TextOverlay_VideoAlignment](#)

Created with the Standard Edition of HelpNDoc: [Create iPhone web-based documentation](#)

SetTextOverlay_AlphaBlendValue

TVideoGrabber.SetTextOverlay_AlphaBlendValue

[Prev](#)
[Next](#)

[TVideoGrabber](#)
[Methods](#)

Sets the the alpha blending value for the specified text overlay

Declaration

procedure SetTextOverlay_AlphaBlendValue (**Index**: LongInt; **Value** :LongInt);

void __fastcall SetTextOverlay_AlphaBlendValue(**int** Index, **int** Value);

Description

Sets the alpha blending value for the text overlay of the specified index

The alpha blending must be enabled by [SetTextOverlay_AlphaBlend](#)

Look at the [Text Overlays](#) chapter for more information.

See Also

[GetTextOverlay_AlphaBlend](#) [GetTextOverlay_AlphaBlendValue](#) [SetTextOverlay_Align](#) [SetTextOverlay_AlphaBlend](#) [SetTextOverlay_BkColor](#) [SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#) [SetTextOverlay_GradientColor](#) [SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#) [SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#) [SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#) [SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_VideoAlignment](#)

Created with the Standard Edition of HelpNDoc: [Experience the Power and Simplicity of HelpNDoc's User Interface](#)

SetTextOverlay_BkColor

TVideoGrabber.SetTextOverlay_BkColor

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets or retrieves the background color used to draw text over frames for the specified text overlay

Declaration

procedure SetTextOverlay_BkColor (Index: LongInt; Value: TColor);

void __fastcall SetTextOverlay_BkColor(int Index, Graphics::TColor Value);

Description

Used to set or retrieve the background color used to draw text over frames.

Useful only if [TextOverlay_Transparent](#) is disabled.

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetTextOverlay_AlphaBlend](#) [GetTextOverlay_AlphaBlendValue](#) [ImageOverlay_StretchToVideoSize](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay_AlphaBlend](#) [SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#) [SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#) [SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#) [SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#) [SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#) [SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay_Align](#) [SetTextOverlay_AlphaBlend](#) [SetTextOverlay_AlphaBlendValue](#) [SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#) [SetTextOverlay_GradientColor](#) [SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#) [SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#) [SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#) [SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#) [TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#) [TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#) [TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#) [TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#) [TextOverlay_VideoAlignment](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Documentation Efficiency with a Help](#)

SetTextOverlay_CustomVar

TVideoGrabber.SetTextOverlay_CustomVar

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets a custom variable displayed in the TextOverlay string for the specified text overlay

Declaration

procedure SetTextOverlay_CustomVar(TextOverlaySelector: LongInt; VarIndex: LongInt; VarText: **String**);

void __fastcall SetTextOverlay_CustomVar(**int** TextOverlaySelector, **int** VarIndex, wchar_t *VarText)

Sub SetTextOverlay_CustomVar(TextOverlaySelector as Long, VarIndex As Long, VarText As String)

Description

Used to set the custom variable displayed in the TextOverlay string.

Setting custom variables is faster than rewriting the full text overlay string, because it saves TVideoGrabber having to parse it again.

Parameters:

Index: index of the overlay

VarIndex: index of the custom variable that has been set with [SetTextOverlay_String](#) (from 0 to 9)

VarText: text of the custom variable

E.g., to initialize the 2 text overlay string, that will use custom variables:

```
...
VideoGrabber.SetTextOverlay_String (0, "my first text uses %custom0%")
VideoGrabber.SetTextOverlay_Left (0, 10)
VideoGrabber.SetTextOverlay_Top (0, 10)
VideoGrabber.SetTextOverlay_Enabled (0, true)

VideoGrabber.SetTextOverlay_String (1, "my second text uses %custom3% and %custom4%")
VideoGrabber.SetTextOverlay_Left (1, 20)
VideoGrabber.SetTextOverlay_Top (1, 100)
VideoGrabber.SetTextOverlay_Enabled (1, true)
VideoGrabber.StartPreview
...
```

then to update the custom variables of the 1st text overlay string:

```
...
VideoGrabber.SetTextOverlayCustomVar (0, 0, "value 0")
...
```

and of the 2nd text overlay string:

```
...
VideoGrabber.SetTextOverlayCustomVar (1, 3, "value 3")
VideoGrabber.SetTextOverlayCustomVar (1, 4, "value 4")
...
```

See also [SetTextOverlay_String](#).

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#)

[AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetTextOverlay](#) [AlphaBlend](#)
[GetTextOverlay](#) [AlphaBlendValue](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#)
[OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#)
[OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#)
[OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#)
[SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#)
[SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#)
[SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#)
[SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#)
[SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#)
[SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [AlphaBlend](#) [SetTextOverlay](#) [AlphaBlendValue](#) [SetTextOverlay](#) [BkColor](#)
[SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#)
[SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#)
[SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#)
[SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#)
[SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#)
[ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#)
[TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#)
[TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#)
[TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#)
[TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#)
[TextOverlay](#) [VideoAlignment](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Free EBook and documentation generator

SetTextOverlay_Enabled

TVideoGrabber.SetTextOverlay_Enabled

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Enables/disables drawing text over video frames.

Declaration

procedure SetTextOverlay_Enabled (Index: LongInt; Value: Boolean);

void __fastcall SetTextOverlay_Enabled(int Index, bool Value);

Description

Used to enable /disable text overlay over video frames.

Look at the [Text overlays](#) chapter.

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#)
[AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetTextOverlay](#) [AlphaBlend](#)
[GetTextOverlay](#) [AlphaBlendValue](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#)
[OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#)
[OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#)
[OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#)
[SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#)
[SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#)
[SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#)
[SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#)
[SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#)
[SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)

[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay_Align](#) [SetTextOverlay_AlphaBlend](#) [SetTextOverlay_AlphaBlendValue](#) [SetTextOverlay_BkColor](#)
[SetTextOverlay_CustomVar](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#) [SetTextOverlay_GradientColor](#)
[SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#)
[SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#)
[SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#)
[SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#)
[ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#)
[TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#)
[TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#)
[TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#)
[TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#)
[TextOverlay_VideoAlignment](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Free HTML Help documentation generator](#)

SetTextOverlay_Font

TVideoGrabber.SetTextOverlay_Font

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets/retrieves the font used to draw text over video frames for the specified text overlay

Declaration

procedure SetTextOverlay_Font (Index: LongInt; Value: TFont);

void __fastcall SetTextOverlay_Font(int Index, Graphics::TFont* Value);

Description

Used to set / retrieve the font used to draw text over video frames.

In C#.NET, use the ToHfont().ToInt32() function of the Font object. E.g.:

```
Font NewFont = new Font("Courier New", 14);
VideoGrabber1.SetTextOverlay_Font (0, NewFont.ToHfont().ToInt32());
VideoGrabber1.SetTextOverlay_Enabled (0, true);
VideoGrabber1.StartPreview();
```

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#)
[AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetTextOverlay_AlphaBlend](#)
[GetTextOverlay_AlphaBlendValue](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#)
[OnFrameBitmap](#) [OnFrameBitmapEventSynchronize](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#)
[OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#)
[OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay_AlphaBlend](#)
[SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#)
[SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#)
[SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#)
[SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#)
[SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#)
[SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay_Align](#) [SetTextOverlay_AlphaBlend](#) [SetTextOverlay_AlphaBlendValue](#) [SetTextOverlay_BkColor](#)
[SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#) [SetTextOverlay_FontColor](#)
[SetTextOverlay_GradientColor](#) [SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#)
[SetTextOverlay_Left](#) [SetTextOverlay_Right](#) [SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#)

[SetTextOverlay_Shadow](#) [SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#)
[SetTextOverlay_String](#) [SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#)
[ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#)
[TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#)
[TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#)
[TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#)
[TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#)
[TextOverlay_VideoAlignment](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Elevate your documentation to new heights with HelpNDoc's built-in SEO](#)

SetTextOverlay_FontColor

TVideoGrabber.SetTextOverlay_FontColor

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Used to adjust the font color for the specified overlay in the OCX and DLL versions

Note: for Delphi and C++Builder set the font directly with [SetTextOverlay_Font](#)

Declaration

procedure SetTextOverlay_FontColor (**Index**: LongInt; Value: TColor);

void __fastcall SetTextOverlay_Font(int Index, DWORD Value);

Description

Used to adjust the font color of the specified overlay in the OCX and DLL versions

Note: for Delphi and C++Builder set the font directly with [SetTextOverlay_Font](#)

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#)
[AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetTextOverlay_AlphaBlend](#)
[GetTextOverlay_AlphaBlendValue](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#)
[OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#)
[OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#)
[OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay_AlphaBlend](#)
[SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#)
[SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#)
[SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#)
[SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#)
[SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#)
[SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay_Align](#) [SetTextOverlay_AlphaBlend](#) [SetTextOverlay_AlphaBlendValue](#) [SetTextOverlay_BkColor](#)
[SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_GradientColor](#)
[SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#)
[SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#)
[SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#)
[SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#)
[ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#)
[TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#)
[TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#)
[TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#)
[TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#)
[TextOverlay_VideoAlignment](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Make your documentation accessible on any device with](#)

SetTextOverlay_GradientColor

TVideoGrabber.SetTextOverlay_GradientColor

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets the gradient color of the specified overlay

Declaration

procedure SetTextOverlay_GradientColor (**Index**: LongInt; Value: TColor);

void __fastcall SetTextOverlay_GradientColor(**int** Index, Graphics::TColor Value);

Description

Used to set the gradient color of the specified overlay.

The gradient mode must first be enabled with [SetTextOverlay_GradientMode](#)

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetTextOverlay_AlphaBlend](#) [GetTextOverlay_AlphaBlendValue](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay_AlphaBlend](#) [SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#) [SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#) [SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#) [SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#) [SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#) [SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay_Align](#) [SetTextOverlay_AlphaBlend](#) [SetTextOverlay_AlphaBlendValue](#) [SetTextOverlay_BkColor](#) [SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#) [SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#) [SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#) [SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#) [SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#) [TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#) [TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#) [TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#) [TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#) [TextOverlay_VideoAlignment](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Easily create Help documents

SetTextOverlay_GradientMode

TVideoGrabber.SetTextOverlay_GradientMode

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Enables enable the gradient mode of the text overlays

Declaration

procedure SetTextOverlay_GradientMode (**Index**: LongInt; Value: TTextOverlayGradientMode);

void __fastcall SetTextOverlay_GradientMode(**int** Index, TTextOverlayGradientMode Value);

Description

Enables enable the gradient mode of the text overlays, and select the orientation

See [TTextOverlayGradientMode](#)

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetTextOverlay_AlphaBlend](#) [GetTextOverlay_AlphaBlendValue](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronone](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay_AlphaBlend](#) [SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#) [SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#) [SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#) [SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#) [SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#) [SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay_Align](#) [SetTextOverlay_AlphaBlend](#) [SetTextOverlay_AlphaBlendValue](#) [SetTextOverlay_BkColor](#) [SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#) [SetTextOverlay_GradientColor](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#) [SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#) [SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#) [SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#) [TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#) [TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#) [TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#) [TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#) [TextOverlay_VideoAlignment](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Eliminate the Struggles of Documentation with a Help Authoring Tool](#)

SetTextOverlay_HighResFont

TVideoGrabber.SetTextOverlay_HighResFont

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Enables the high-resolution font mode

Declaration

procedure SetTextOverlay_HighResFont (**Index**: LongInt; Value: Boolean);

void __fastcall SetTextOverlay_HighResFont(**int** Index, **bool** Value);

Description

Select the high-resolution fonts for the text overlays.

Note that the high resolution fonts are supported in C++, C# and VB.NET, ActiveX, Lazarus and Delphi/ C++Builder XE or higher.

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetTextOverlay_AlphaBlend](#)

[GetTextOverlay_AlphaBlendValue](#) [ImageOverlay_StretchToVideoSize](#) [MouseWheelEventEnabled](#)
[OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#)
[OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#)
[OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay_AlphaBlend](#)
[SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#)
[SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#)
[SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#)
[SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#)
[SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#)
[SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay_Align](#) [SetTextOverlay_AlphaBlend](#) [SetTextOverlay_AlphaBlendValue](#) [SetTextOverlay_BkColor](#)
[SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#)
[SetTextOverlay_GradientColor](#) [SetTextOverlay_GradientMode](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#)
[SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#)
[SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#)
[SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#)
[ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#)
[TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#)
[TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#)
[TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#)
[TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#)
[TextOverlay_VideoAlignment](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Full-featured EPub generator

SetTextOverlay_Left

TVideoGrabber.SetTextOverlay_Left

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Left position (in pixels) where the text will be drawn over video frames for the specified text overlay

Declaration

procedure SetTextOverlay_Left (Index: LongInt; Value: LongInt);

void __fastcall SetTextOverlay_Left(int Index, int Value);

Description

Used to set or retrieve the left position (in pixels) where the text will be drawn over video frames. The text will be left-padded on this position if [SetTextOverlay_Align](#) has been set to tf_Left.

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#)
[AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetTextOverlay_AlphaBlend](#)
[GetTextOverlay_AlphaBlendValue](#) [ImageOverlay_StretchToVideoSize](#) [MouseWheelEventEnabled](#)
[OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#)
[OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#)
[OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay_AlphaBlend](#)
[SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#)
[SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#)
[SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#)
[SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#)
[SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#)
[SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay_Align](#) [SetTextOverlay_AlphaBlend](#) [SetTextOverlay_AlphaBlendValue](#) [SetTextOverlay_BkColor](#)

[SetTextOverlay_CustomVar](#)
[SetTextOverlay_Enabled](#)
[SetTextOverlay_Font](#)
[SetTextOverlay_FontColor](#)
[SetTextOverlay_GradientColor](#)
[SetTextOverlay_GradientMode](#)
[SetTextOverlay_HighResFont](#)
[SetTextOverlay_Right](#)
[SetTextOverlay_Scrolling](#)
[SetTextOverlay_ScrollingSpeed](#)
[SetTextOverlay_Shadow](#)
[SetTextOverlay_ShadowColor](#)
[SetTextOverlay_ShadowDirection](#)
[SetTextOverlay_String](#)
[SetTextOverlay_TargetDisplay](#)
[SetTextOverlay_Top](#)
[SetTextOverlay_Transparent](#)
[ShapeOverlay](#)
[ShapeOverlayEnabled](#)
[ShapeOverlayList](#)
[TextOverlay_Align](#)
[TextOverlay_BkColor](#)
[TextOverlay_CreateCustomFont](#)
[TextOverlay_CreateCustomFont2](#)
[TextOverlay_Enabled](#)
[TextOverlay_Font](#)
[TextOverlay_FontColor](#)
[TextOverlay_Left](#)
[TextOverlay_Right](#)
[TextOverlay_Scrolling](#)
[TextOverlay_ScrollingSpeed](#)
[TextOverlay_Selector](#)
[TextOverlay_Shadow](#)
[TextOverlay_ShadowColor](#)
[TextOverlay_ShadowDirection](#)
[TextOverlay_String](#)
[TextOverlay_Top](#)
[TextOverlay_Transparent](#)
[TextOverlay_VideoAlignment](#)
[TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Make the switch to CHM with HelpNDoc's hassle-free WinHelp HLP to CHM conversion tool

SetTextOverlay_Right

TVideoGrabber.SetTextOverlay_Right

[Prev](#)
[Next](#)

[TVideoGrabber](#)
[Methods](#)

Right position (in pixels) where the text will be drawn over video frames for the specified text overlay

Declaration

procedure SetTextOverlay_Right (Index: LongInt; Value: LongInt);

void __fastcall SetTextOverlay_Right(int Index, int Value);

Description

Used to set or retrieve the right position (in pixels) where the text will be drawn over video frames. The text will be left-padded on this position if [SetTextOverlay_Align](#) has been set to tf_Right.

See Also

[TCardinalDirection](#)
[TOnFrameOverlayUsingDC](#)
[TOnFrameOverlayUsingDIB](#)
[TTextOverlayAlign](#)
[AdjustOverlayAspectRatio](#)
[DrawBitmapOverFrame](#)
[GetFrameInfoString](#)
[GetTextOverlay_AlphaBlend](#)
[GetTextOverlay_AlphaBlendValue](#)
[ImageOverlay_StretchToVideoSize](#)
[MouseWheelEventEnabled](#)
[OnFrameBitmap](#)
[OnFrameBitmapEventSynchronise](#)
[OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingDIB](#)
[OnFrameOverlayUsingVIDEOHDR](#)
[OnMouseDown](#)
[OnMouseMove](#)
[OnMouseUp](#)
[OnMouseWheel](#)
[OverlayAfterTransform](#)
[RefreshPlayerOverlays](#)
[SetImageOverlay_AlphaBlend](#)
[SetImageOverlay_AlphaBlendValue](#)
[SetImageOverlay_ChromaKey](#)
[SetImageOverlay_ChromaKeyLeewayPercent](#)
[SetImageOverlay_ChromaKeyRGBColor](#)
[SetImageOverlay_Enabled](#)
[SetImageOverlay_Height](#)
[SetImageOverlay_LeftLocation](#)
[SetImageOverlay_RotationAngle](#)
[SetImageOverlay_StretchToVideoSize](#)
[SetImageOverlay_TargetDisplay](#)
[SetImageOverlay_TopLocation](#)
[SetImageOverlay_Transparent](#)
[SetImageOverlay_TransparentColorValue](#)
[SetImageOverlay_UseTransparentColor](#)
[SetImageOverlay_Width](#)
[SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#)
[SetImageOverlayFromHBitmap2](#)
[SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#)
[SetImageOverlayFromJPEGFile](#)
[SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#)
[SetImageOverlayFromTImage](#)
[SetImageOverlayFromTImage2](#)
[SetTextOverlay_Align](#)
[SetTextOverlay_AlphaBlend](#)
[SetTextOverlay_AlphaBlendValue](#)
[SetTextOverlay_BkColor](#)
[SetTextOverlay_CustomVar](#)
[SetTextOverlay_Enabled](#)
[SetTextOverlay_Font](#)
[SetTextOverlay_FontColor](#)
[SetTextOverlay_GradientColor](#)
[SetTextOverlay_GradientMode](#)
[SetTextOverlay_HighResFont](#)
[SetTextOverlay_Left](#)
[SetTextOverlay_Scrolling](#)
[SetTextOverlay_ScrollingSpeed](#)
[SetTextOverlay_Shadow](#)
[SetTextOverlay_ShadowColor](#)
[SetTextOverlay_ShadowDirection](#)
[SetTextOverlay_String](#)
[SetTextOverlay_TargetDisplay](#)
[SetTextOverlay_Top](#)
[SetTextOverlay_Transparent](#)
[ShapeOverlay](#)
[ShapeOverlayEnabled](#)
[ShapeOverlayList](#)
[TextOverlay_Align](#)
[TextOverlay_BkColor](#)
[TextOverlay_CreateCustomFont](#)
[TextOverlay_CreateCustomFont2](#)
[TextOverlay_Enabled](#)
[TextOverlay_Font](#)
[TextOverlay_FontColor](#)
[TextOverlay_Left](#)
[TextOverlay_Right](#)
[TextOverlay_Scrolling](#)
[TextOverlay_ScrollingSpeed](#)
[TextOverlay_Selector](#)
[TextOverlay_Shadow](#)
[TextOverlay_ShadowColor](#)
[TextOverlay_ShadowDirection](#)
[TextOverlay_String](#)
[TextOverlay_Top](#)
[TextOverlay_Transparent](#)
[TextOverlay_VideoAlignment](#)
[TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Produce electronic books easily

SetTextOverlay_Scrolling

TVideoGrabber.SetTextOverlay_Scrolling

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Enables the text scrolling for the specified text overlay

Declaration

procedure SetTextOverlay_Scrolling (Index: LongInt; Value: Boolean);

void __fastcall SetTextOverlay_Scrolling(int Index, bool Value);

Description

Enable the scrolling of the text overlay.

The scrolling speed is adjusted by [SetTextOverlay_ScrollingSpeed](#).

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetTextOverlay_AlphaBlend](#) [GetTextOverlay_AlphaBlendValue](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEODR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay_AlphaBlend](#) [SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#) [SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#) [SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#) [SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#) [SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#) [SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay_Align](#) [SetTextOverlay_AlphaBlend](#) [SetTextOverlay_AlphaBlendValue](#) [SetTextOverlay_BkColor](#) [SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#) [SetTextOverlay_GradientColor](#) [SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#) [SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#) [SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#) [TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#) [TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#) [TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#) [TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#) [TextOverlay_VideoAlignment](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Maximize Your Reach: Convert Your Word Document to an ePub or Kindle eBook

SetTextOverlay_ScrollingSpeed

TVideoGrabber.SetTextOverlay_ScrollingSpeed

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Activates the text scrolling for the specified text overlay

Declaration

procedure SetTextOverlay_ScrollingSpeed (Index: LongInt; Value: LongInt);

```
void __fastcall SetTextOverlay_ScrollingSpeed(int Index, int Value);
```

Description

Used to adjust the scrolling speed of the text overlay.

The scrolling must be enabled first with [SetTextOverlay_Scrolling](#).

Value = 0 -> no scrolling

Value > 0 -> scrolling from right to left

Value < 0 -> scrolling from left to right

E.g.:

SetTextOverlay_ScrollingSpeed (index, 1) -> slow scrolling

SetTextOverlay_ScrollingSpeed (index, -3) -> backwards average scrolling

SetTextOverlay_ScrollingSpeed (index, 6) -> fast scrolling

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetTextOverlay_AlphaBlend](#) [GetTextOverlay_AlphaBlendValue](#) [ImageOverlay_StretchToVideoSize](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay_AlphaBlend](#) [SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#) [SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#) [SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#) [SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#) [SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#) [SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay_Align](#) [SetTextOverlay_AlphaBlend](#) [SetTextOverlay_AlphaBlendValue](#) [SetTextOverlay_BkColor](#) [SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#) [SetTextOverlay_GradientColor](#) [SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#) [SetTextOverlay_Scrolling](#) [SetTextOverlay_Shadow](#) [SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#) [SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#) [TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#) [TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#) [TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#) [TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#) [TextOverlay_VideoAlignment](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Transform Your Documentation Process with HelpNDoc's Project Analyzer

SetTextOverlay_Shadow

TVideoGrabber.SetTextOverlay_Shadow

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Enables the shadow for the specified text overlay

Declaration

procedure SetTextOverlay_Shadow (Index: LongInt; Value: Boolean);

```
void __fastcall SetTextOverlay_Shadow(int Index, bool Value);
```

Description

Enables the shadow under the text overlay.

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetTextOverlay](#) [AlphaBlend](#) [GetTextOverlay](#) [AlphaBlendValue](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [AlphaBlend](#) [SetTextOverlay](#) [AlphaBlendValue](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#) [SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#) [SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#) [SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#) [TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#) [TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#) [TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#) [TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#) [TextOverlay](#) [VideoAlignment](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Create Professional CHM Help Files with HelpNDoc's Easy-to-Use Tool

SetTextOverlay_ShadowColor**TVideoGrabber.SetTextOverlay_ShadowColor**[Prev](#)[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets the color of the shadow for the specified text overlay

Declaration

procedure SetTextOverlay_ShadowColor (Index: LongInt; Value: TColor);

void __fastcall SetTextOverlay_ShadowColor(int Index, Graphics::TColor Value);

Description

Specifies the color of the shadow under the text overlay when the shadow has been enabled with [SetTextOverlay_Shadow](#).

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetTextOverlay](#) [AlphaBlend](#) [GetTextOverlay](#) [AlphaBlendValue](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#)

[SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#) [SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#) [SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay_Align](#) [SetTextOverlay_AlphaBlend](#) [SetTextOverlay_AlphaBlendValue](#) [SetTextOverlay_BkColor](#) [SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#) [SetTextOverlay_GradientColor](#) [SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#) [SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#) [SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#) [TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#) [TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#) [TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#) [TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#) [TextOverlay_VideoAlignment](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Revolutionize Your Documentation Output with HelpNDoc's Stunning User Interface

SetTextOverlay_ShadowDirection

TVideoGrabber.SetTextOverlay_ShadowDirection

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets the direction of the shadow for the specified text overlay

Declaration

```
procedure SetTextOverlay_ShadowDirection(Index: LongInt; Value: TCardinalDirection);
```

```
void __fastcall SetTextOverlay_ShadowDirection(int Index, TCardinalDirection Value);
```

Description

Specifies the direction of the shadow under the text overlay when the shadow has been enabled with [SetTextOverlay_Shadow](#).

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetTextOverlay_AlphaBlend](#) [GetTextOverlay_AlphaBlendValue](#) [ImageOverlay_StretchToVideoSize](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay_AlphaBlend](#) [SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#) [SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#) [SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#) [SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#) [SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#) [SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay_Align](#) [SetTextOverlay_AlphaBlend](#) [SetTextOverlay_AlphaBlendValue](#) [SetTextOverlay_BkColor](#) [SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#) [SetTextOverlay_GradientColor](#) [SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#) [SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#) [SetTextOverlay_ShadowColor](#) [SetTextOverlay_String](#) [SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#) [TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#)

[TextOverlay_Enabled](#) [TextOverlay_Font](#) [TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#) [TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#) [TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#) [TextOverlay_VideoAlignment](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Elevate Your Documentation Process with HelpNDoc's Advanced Features

SetTextOverlay_String

TVideoGrabber.SetTextOverlay_String

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets or retrieves the text string that will be drawn over video frames for the specified text overlay

Declaration

procedure SetTextOverlay_String (Index: LongInt; Value: string);

void __fastcall SetTextOverlay_String(int Index, wchar_t *Value);

Description

Used to set or retrieve the text string that will be drawn over video frames.

The text string can be composed of several lines.

This property can be modified at any time, even if the text is currently drawn over a video frame.

Several TVideoGrabber variables can be used within the text string, delimited by a percent symbol **(2)**. When TVideoGrabber detects one of these variables the variable label is replaced by its current value:

"%sys_time[dd/mm/yy hh:nn:ss]%" : current system date/time **(1)**

"%dv_time[dd/mm/yy hh:nn:ss]%" : current date/time stored on the DV VCR tape **(1)**

"%time_code%" : current DV VCR time code, if available

"%frame_count%" : number of the current frame

"%time_full%" : time of the current frame in hh:mm:ss:cc format

"%time_sec%" : time of the current frame, in seconds with 2 decimals

"%time_100ns%" : time of the current frame, in 100 nano-seconds units

"%custom0% to %custom9%" : up to 10 custom variables that can be set by using

[SetTextOverlay_CustomVar](#) .

(1) any valid [date/time format](#) is accepted between the brackets of sys_time and dv_time.

(2) the percent symbol is a reserved character. If you need to display the percent symbol itself, just duplicate it, e.g. TextOverlay_String = "the percent symbol is %%"

See [SetTextOverlay_CustomVar](#)

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetTextOverlay_AlphaBlend](#) [GetTextOverlay_AlphaBlendValue](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay_AlphaBlend](#) [SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#) [SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#) [SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#) [SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#) [SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#) [SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay_Align](#) [SetTextOverlay_AlphaBlend](#) [SetTextOverlay_AlphaBlendValue](#) [SetTextOverlay_BkColor](#)

[SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#)
[SetTextOverlay_GradientColor](#) [SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#)
[SetTextOverlay_Left](#) [SetTextOverlay_Right](#) [SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#)
[SetTextOverlay_Shadow](#) [SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#)
[SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#)
[ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#)
[TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#)
[TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#)
[TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#)
[TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#)
[TextOverlay_VideoAlignment](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Easily Add Encryption and Password Protection to Your PDFs](#)

SetTextOverlay_TargetDisplay

TVideoGrabber.SetTextOverlay_TargetDisplay

[Prev](#)
[Next](#)

TVideoGrabber **Methods**

Used to specify the location (video frame or window) of the text overlay

Declaration

procedure SetTextOverlay_TargetDisplay (Index: LongInt; Value: LongInt);

void __fastcall SetTextOverlay_TargetDisplay(int Index, int Value);

Description

Used to specify the location (video frame or window) of the text overlay specified by its index

See [Frame overlay vs window overlay](#)

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#)
[AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetTextOverlay_AlphaBlend](#)
[GetTextOverlay_AlphaBlendValue](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#)
[OnFrameBitmap](#) [OnFrameBitmapEventSynchrone](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#)
[OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#)
[OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay_AlphaBlend](#)
[SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#)
[SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#)
[SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#)
[SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#)
[SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#)
[SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay_Align](#) [SetTextOverlay_AlphaBlend](#) [SetTextOverlay_AlphaBlendValue](#) [SetTextOverlay_BkColor](#)
[SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#)
[SetTextOverlay_GradientColor](#) [SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#)
[SetTextOverlay_Left](#) [SetTextOverlay_Right](#) [SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#)
[SetTextOverlay_Shadow](#) [SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#)
[SetTextOverlay_String](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#)
[ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#)
[TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#)
[TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#)
[TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#)
[TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#)
[TextOverlay_VideoAlignment](#) [TranslateMouseCoordinates](#)

SetTextOverlay_Top

TVideoGrabber.SetTextOverlay_Top

[Prev](#)[Next](#)[TVideoGrabber](#) [Methods](#)

Sets the top position (in pixels) where the text will be drawn over video frames for the specified text overlay

Declaration

procedure SetTextOverlay_Top (Index: LongInt; Value: LongInt);

void __fastcall SetTextOverlay_Top(int Index, int Value);

Description

Used to set or retrieve the top position (in pixels) where the text will be drawn over video frames.

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetTextOverlay_AlphaBlend](#) [GetTextOverlay_AlphaBlendValue](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay_AlphaBlend](#) [SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#) [SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#) [SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#) [SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#) [SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#) [SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay_Align](#) [SetTextOverlay_AlphaBlend](#) [SetTextOverlay_AlphaBlendValue](#) [SetTextOverlay_BkColor](#) [SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#) [SetTextOverlay_GradientColor](#) [SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#) [SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#) [SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#) [SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#) [TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#) [TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#) [TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#) [TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#) [TextOverlay_VideoAlignment](#) [TranslateMouseCoordinates](#)

SetTextOverlay_Transparent

TVideoGrabber.SetTextOverlay_Transparent

[Prev](#)[Next](#)[TVideoGrabber](#) [Methods](#)

Enables/disables transparency of text drawn over video frames for the specified text overlay

Declaration

procedure SetTextOverlay_Transparent (Index: LongInt; Value: Boolean);

```
void __fastcall SetTextOverlay_Transparent(int Index, bool Value);
```

Description

Used to enable or disable transparency of text drawn over video frames.
If enabled, any value set with [SetTextOverlay_BkColor](#) is ignored.

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetTextOverlay_AlphaBlend](#) [GetTextOverlay_AlphaBlendValue](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay_AlphaBlend](#) [SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#) [SetImageOverlay_ChromaKeyLeewayPercent](#) [SetImageOverlay_ChromaKeyRGBColor](#) [SetImageOverlay_Enabled](#) [SetImageOverlay_Height](#) [SetImageOverlay_LeftLocation](#) [SetImageOverlay_RotationAngle](#) [SetImageOverlay_StretchToVideoSize](#) [SetImageOverlay_TargetDisplay](#) [SetImageOverlay_TopLocation](#) [SetImageOverlay_Transparent](#) [SetImageOverlay_TransparentColorValue](#) [SetImageOverlay_UseTransparentColor](#) [SetImageOverlay_Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay_Align](#) [SetTextOverlay_AlphaBlend](#) [SetTextOverlay_AlphaBlendValue](#) [SetTextOverlay_BkColor](#) [SetTextOverlay_CustomVar](#) [SetTextOverlay_Enabled](#) [SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#) [SetTextOverlay_GradientColor](#) [SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#) [SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#) [SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#) [SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#) [TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#) [TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#) [TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#) [TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#) [TextOverlay_VideoAlignment](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Transform your help documentation into a stunning website

SetVideoCompressionDefaults

TVideoGrabber.SetVideoCompressionDefaults

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Resets the current [video compressor](#) .

Declaration

function SetVideoCompressionDefaults: Boolean;

bool __fastcall SetVideoCompressionDefaults(**void**)

Function SetVideoCompressionDefaults() As Boolean

Description

Used to reset the compression settings of the current [video compressor](#) to their default values.

See Also

[Recording methods and properties](#) [TCompressionType](#) [TOnVideoCompressionSettings](#) [AudioCompressor](#) [AudioCompressorIndex](#) [AudioCompressorName](#) [AudioCompressors](#) [AudioCompressorsCount](#) [CompressionMode](#) [CompressionType](#) [GetVideoCompressionSettings](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [OnVideoCompressionSettings](#) [RefreshDevicesAndCompressorsLists](#)

[SaveCompressorSettingsToDataString](#) [SetVideoCompressionSettings](#) [VideoCompression](#) [KeyFrameRate](#) [VideoCompression](#) [PFramesPerKeyFrame](#) [VideoCompression](#) [Quality](#) [VideoCompression](#) [WindowSize](#) [VideoCompressor](#) [VideoCompressorIndex](#) [VideoCompressorName](#) [VideoCompressors](#) [VideoCompressorsCount](#)

Created with the Standard Edition of HelpNDoc: [Easily create HTML Help documents](#)

SetVideoCompressionSettings

TVideoGrabber.SetVideoCompressionSettings

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets video compression settings.

Declaration

function SetVideoCompressionSettings(DataRate, KeyFrameRate, PFramesPerKeyFrame, WindowSize: LongInt; Quality: Double): Boolean;

bool __fastcall SetVideoCompressionSettings(int DataRate, int KeyFrameRate, int PFramesPerKeyFrame, int WindowSize, double Quality)

Function SetVideoCompressionSettings(DataRate As Long, KeyFrameRate As Long, PFramesPerKeyFrame As Long, WindowSize As Long, Quality As Double) As Boolean

Description

Sets the video compression settings supported by some codecs. Returns true if the current video compressor supports this interface.

The current values can be retrieved by [GetVideoCompressionSettings](#) .

DataRate: retrieves the output data rate.

KeyFrameRate: The key-frame rate is the number of frames per key frame. For example, if the rate is 15, then a key frame occurs every 15 frames.

PFramesPerKeyFrame: P frames are used only in MPEG compression. E.g. let's say a key frame occurs once every 10 frames, and there are three P frames per key frame. The P frames will be spaced evenly between the key frames. The remaining six frames are bi-directional (B) frames.

WindowSize: retrieves the number of frames over which the compressor will maintain the average data rate. E.g. if a data rate of 100K/sec and a frame rate of 10 frames per second, if the window size is 1, then every frame will be 10K or less. If the window size is 5, then every five consecutive frames will average 10K per frame, but individual frames may exceed this size.

Quality: The quality is expressed as a value between 0.0 and 1.0, where 1.0 indicates the best quality and 0.0 indicates the worst quality. If the value is negative, the filter will use the default quality.

CanCrunch: the compressor can compress video to a specified data rate (see DataRate above).

CanKeyFrame: the compressor supports the KeyFrame property above.

CanBFrame: the compressor supports the PFramesPerKeyFrame property above.

CanWindow: the compressor supports WindowSize property above.

CanQuality: the compressor supports Quality property above.

See Also

[Recording methods and properties](#) [TCompressionType](#) [TOnVideoCompressionSettings](#) [AudioCompressor](#) [AudioCompressorIndex](#) [AudioCompressorName](#) [AudioCompressors](#) [AudioCompressorsCount](#) [CompressionMode](#) [CompressionType](#) [GetVideoCompressionSettings](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [OnVideoCompressionSettings](#) [RefreshDevicesAndCompressorsLists](#) [SaveCompressorSettingsToDataString](#) [SetVideoCompressionDefaults](#) [VideoCompression](#) [KeyFrameRate](#) [VideoCompression](#) [PFramesPerKeyFrame](#) [VideoCompression](#) [Quality](#) [VideoCompression](#) [WindowSize](#) [VideoCompressor](#) [VideoCompressorIndex](#) [VideoCompressorName](#) [VideoCompressors](#) [VideoCompressorsCount](#)

Created with the Standard Edition of HelpNDoc: [Achieve Professional Documentation Results with a Help Authoring Tool](#)

SetVideoControlMode

TVideoGrabber.SetVideoControlMode

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets the [TVideoControl](#) settings

Declaration

function SetVideoControlMode(FlipHorizontal, FlipVertical, ExternalTriggerEnable, Trigger: Boolean): Boolean;

bool __fastcall SetVideoControlMode(**bool** FlipHorizontal, **bool** FlipVertical, **bool** ExternalTriggerEnable, **bool** Trigger)

Function SetVideoControlMode (FlipHorizontal As Boolean, FlipVertical As Boolean, ExternalTriggerEnable As Boolean, Trigger As Boolean)As Boolean

Description

Used to set all the [TVideoControl](#) settings, if available for the current video capture device (see [IsVideoControlModeAvailable](#)).

See Also

[TVideoControl](#) [GetVideoControlMode](#) [IsVideoControlAvailable](#) [IsVideoControlModeAvailable](#) [SetVideoControlMode2](#) [VideoControlSettings](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Documentation Output with HelpNDoc's Advanced Project Analyzer](#)

SetVideoControlMode2

TVideoGrabber.SetVideoControlMode2

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets one of the [TVideoControl](#) settings

Declaration

function SetVideoControlMode(Mode: TVideoControl; Value: Boolean): Boolean;

bool __fastcall SetVideoControlMode2(TVideoControl Mode, **bool** Value);

Function SetVideoControlMode (mode as TVideoControl, Value as Boolean) As Boolean

Description

Used to set one of the [TVideoControl](#) settings, if available for the current video capture device (see [IsVideoControlModeAvailable](#)).

See Also

[TVideoControl](#) [GetVideoControlMode](#) [IsVideoControlAvailable](#) [IsVideoControlModeAvailable](#) [SetVideoControlMode](#) [VideoControlSettings](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Documentation Output with HelpNDoc's Advanced Project Analyzer](#)

SetVideoQuality

TVideoGrabber.SetVideoQuality

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

ber

Sets a [TVideoQuality](#) value.

Declaration

function SetVideoQuality(Setting: TVideoQuality; SetAuto: Boolean; SetDefault: Boolean; SetValue: LongInt): Boolean;

bool __fastcall SetVideoQuality(TVideoQuality Setting, **bool** SetAuto, **bool** SetDefault, **int** SetValue)

Function SetVideoQuality(Setting As TVideoQuality, SetAuto as Boolean, SetDefault as Boolean, SetValue As Long) As Boolean

Description

Used to set a [TVideoQuality](#) value, if available for the current video capture device (test [IsVideoQualitySettingAvailable](#) for availability).

- if **SetAuto** is true, SetDefault and SetValue are ignored and the setting is switched in "auto" mode
- if **SetAuto** is false and **SetDefault** is true, SetValue is ignored and the setting is set to its default value,
- if **SetAuto** is false and **SetDefault** is false, the **SetValue** value is applied to the setting.

See Also

[IsVideoQualityAvailable](#) [IsVideoQualitySettingAvailable](#) [VideoQualityAuto](#) [VideoQualityDefault](#) [VideoQualityMax](#) [VideoQualityMin](#) [VideoQualitySettings](#) [VideoQualityStep](#) [VideoRendererPriority](#) [TVideoQuality](#)

Created with the Standard Edition of HelpNDoc: [Don't Let Unauthorized Users View Your PDFs: Learn How to Set Passwords](#)

SetVMR9ImageAdjustmentValue

TVideoGrabber.SetVMR9ImageAdjustmentValue

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets a VMR9 image adjustment setting.

Declaration

function SetVMR9ImageAdjustmentValue(MainDisplay: Boolean; VMR9ControlSetting: TVMR9ImageAdjustment; Value: LongInt; FixRange: Boolean): Boolean;

bool __fastcall SetVMR9ImageAdjustmentValue(**bool** MainDisplay, TVMR9ImageAdjustment VMR9ControlSetting, **int** Value, **bool** FixRange)

Function SetVMR9ImageAdjustmentValue(MainDisplay As Boolean, VMR9ControlSetting As TVMR9ImageAdjustment, param3 As Long, FixRange As Boolean) As Boolean

Description

Used to set a VMR9 image adjustment property (brightness, contrast, hue, saturation). The VMR9ControlSetting parameter is a [TVMR9ImageAdjustment](#) type.

See also [GetVMR9ImageAdjustmentValue](#).

See Also

[GetVMR9ImageAdjustmentBounds](#) [IsVMR9ImageAdjustmentAvailable](#) [TVMR9ImageAdjustment](#)

Created with the Standard Edition of HelpNDoc: [Elevate Your CHM Help Files with HelpNDoc's Advanced Customization Options](#)

SetVuMeter_Enabled

TVideoGrabber.SetVuMeter_Enabled

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Methods](#)

Used to enable/disable a VU-meter

Declaration

procedure TVideoGrabber.SetVuMeter_Enabled (Index: LongInt; Value: TVuMeter);

void __fastcall SetVuMeter_Enabled(int Index, TVuMeter Value);

Description

Enables/disables the VU-meter specified by its index

See Also

[SetVUMeterSetting](#)

Created with the Standard Edition of HelpNDoc: [Generate EPub eBooks with ease](#)

SetVUMeterSetting

TVideoGrabber.SetVUMeterSetting

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Methods](#)

Used to adjust the VU-Meter settings.

Declaration

function SetVUMeterSetting (ChannellIndex: LongWord; VUMeterSetting: TVUMeterSetting; Value: LongWord): Boolean;

bool __fastcall SetVUMeterSetting (TVUMeterSetting VUMeterSetting, **unsigned int** Value)

Function SetVUMeterSetting (Setting As TxVUMeterSetting, Value as Long) As Boolean

Description

Used to adjust the VU-Meter settings, of the [TVUMeterSetting](#) type.

vu_Handle : sets the Handle of the panel or image control on which the VUMeter will be displayed (0 disables the previous handle).

vu_WarningPercent : percentage of the level above which the vu_WarningColor is used

vu_PeakPercent : percentage of the level above which the vu_PeakColor is used

vu_BkgndColor : color of the VU-meter background

vu_NormalColor : color of the normal level (green by default)

vu_WarningColor : color of the warning level (above the vu_WarningPercent)

vu_PeakColor : color of the peak level (above the vu_PeakPercent)

vu_TickSize : size of ticks in the bargraph VU-Meter

vu_TickInterval : interval between ticks in the bargraph VU-Meter

vu_NeedleThickness : thickness of the needle in the analog VU-Meter

See the "[Audio levels and VU-Meters](#)" chapter that explains how to use the VU-Meters.

See Also

[SetVuMeter_Enabled](#)

Created with the Standard Edition of HelpNDoc: [Easily create iPhone documentation](#)

SetWindowRecordingByHandle

TVideoGrabber.SetWindowRecordingByHandle

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Methods](#)

Used to specify a window for screen recording

Declaration

function SetWindowRecordingByHandle (WindowHandle: LongInt): **Boolean**;

bool __fastcall SetWindowRecordingByHandle(**int** WindowHandle);

Function SetWindowRecordingByHandle (WindowHandle as Long) as Boolean

Description

Used for the screen recording to specify a window to record by its handle.
The handle of this window can be retrieved by invoking [EnumerateWindows](#).

See the "[Recording a window](#)" chapter for more information.

See Also

[TOnEnumerateWindows](#) [EnumerateWindows](#) [ScreenRecordingLayeredWindows](#) [ScreenRecordingMonitor](#) [ScreenRecordingNonVisibleWindows](#) [ScreenRecordingUsingCoordinates](#) [ScreenRecordingWithCursor](#) [SetWindowRecordingByName](#)

Created with the Standard Edition of HelpNDoc: [5 Reasons Why a Help Authoring Tool is Better than Microsoft Word for Documentation](#)

SetWindowRecordingByName

TVideoGrabber.SetWindowRecordingByName

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Methods](#)

Used to specify a window for screen recording

Declaration

function SetWindowRecordingByName (WindowName: string; ExactMatch: Boolean): **Boolean**;

bool __fastcall SetWindowRecordingByName(wchar_t *WindowName, **bool** ExactMatch);

Function SetWindowRecordingByName (WindowName as String, ExactMatch as Boolean) as Boolean

Description

Used for the screen recording to specify a window to record by its window name or class name.

The window name or class name of this window can be picked up by looking at the title bar of the window, or by retrieved by invoking [EnumerateWindows](#).

- you can identify the exact window name by passing the full string to the WindowName parameter and setting ExactMatch = true
- you can identify the window by a substring of its name, passed to the WindowName parameter, and by setting ExactMatch = false.

Note: when ExactMatch = false, the WindowName parameter is not case-sensitive.

See the "[Recording a window](#)" chapter for more information.

See Also

[TOnEnumerateWindows](#) [EnumerateWindows](#) [ScreenRecordingLayeredWindows](#) [ScreenRecordingMonitor](#) [ScreenRecordingNonVisibleWindows](#) [ScreenRecordingUsingCoordinates](#) [ScreenRecordingWithCursor](#) [SetWindowRecordingByHandle](#)

Created with the Standard Edition of HelpNDoc: Effortlessly Create High-Quality Documentation with a Help Authoring Tool

SetWindowTransparency

TVideoGrabber.SetWindowTransparency

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets the transparency (color key or alpha value) on a window or form

Declaration

function SetWindowTransparency (WndHandle: HWND; UseColorKey: Boolean; ColorKeyValue: TColor; UseAlpha: Boolean; AlphaValue: LongInt): Boolean;

bool __fastcall SetWindowTransparency (HWND WndHandle, **bool** UseColorKey, COLORREF ColorKeyValue, **bool** UseAlpha, byte AlphaValue)

function SetWindowTransparency (WndHandle as Long, UseColorKey as Boolean, ColorKeyValue as Long, UseAlpha as Boolean, AlphaValue as Long) as Boolean

Description

Used to set the transparency (color key or alpha value) on a window or form.

Set [ColorKeyEnabled](#) = true, then invoke this function and pass the handle to the form you wish to be transparent.

You can find sample code in the transparency checkbox code of the "display" tab of the MainDemo project.

E.g. in Delphi:

```
procedure TfrmMainForm.chkTransparencyClick(Sender: TObject);
begin
    // in this sample the transparent window is created in the VideoGrabberColorKeyChange
    // event below when the color key is set
    VideoGrabber.ColorKeyEnabled := chkTransparency.Checked;
    if VideoGrabber.ColorKeyEnabled then begin
        if not assigned (FTransparentForm) then begin
            Application.CreateForm (TForm, FTransparentForm);
            FTransparentForm.Width := 300;
            FTransparentForm.Height := 100;
            FTransparentForm.Position := poScreenCenter;
            FTransparentForm.FormStyle := fsStayOnTop;
            FTransparentPanel := TPanel.Create (FTransparentForm);
            FTransparentPanel.Parent := FTransparentForm;
            FTransparentPanel.BevelInner := bvNone;
            FTransparentPanel.BevelOuter := bvNone;
            FTransparentPanel.Align := alClient;
            FTransparentPanel.Caption := 'transparent window';
            FTransparentPanel.Font.Color := clRed;
            FTransparentPanel.Font.Size := 16;
        end;

        FTransparentForm.Color := clBlue;
        FTransparentPanel.Color := VideoGrabber.ColorKey;
```

```

        VideoGrabber.SetWindowTransparency (FTransparentForm.Handle, true, VideoGrabber.Col
        FTransparentForm.Show;
    end;
end;

```

See Also

[DualDisplay](#) [TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display](#) [Active Display](#) [AlphaBlendEnabled](#) [Display](#) [AlphaBlendValue](#) [Display](#) [AutoSize](#) [Display](#) [Embedded Display](#) [FullScreen Display](#) [Height](#) [Display](#) [Left Display](#) [Monitor Display](#) [MouseMovesWindow Display](#) [PanScanRatio](#) [Display](#) [SetLocation](#) [Display](#) [StayOnTop Display](#) [Top Display](#) [TransparentColorEnabled](#) [Display](#) [TransparentColorValue](#) [Display](#) [VideoHeight](#) [Display](#) [VideoPortEnabled](#) [Display](#) [VideoWidth](#) [Display](#) [VideoWindowHandle](#) [Display](#) [Width](#) [DualDisplay](#) [Active](#) [DualDisplay](#) [AlphaBlendEnabled](#) [DualDisplay](#) [AlphaBlendValue](#) [DualDisplay](#) [AutoSize](#) [DualDisplay](#) [Embedded](#) [DualDisplay](#) [FullScreen](#) [DualDisplay](#) [Height](#) [DualDisplay](#) [Left](#) [DualDisplay](#) [Monitor](#) [DualDisplay](#) [MouseMovesWindow](#) [DualDisplay](#) [PanScanRatio](#) [DualDisplay](#) [SetLocation](#) [DualDisplay](#) [StayOnTop](#) [DualDisplay](#) [Top](#) [DualDisplay](#) [TransparentColorEnabled](#) [DualDisplay](#) [TransparentColorValue](#) [DualDisplay](#) [VideoHeight](#) [DualDisplay](#) [VideoPortEnabled](#) [DualDisplay](#) [VideoWidth](#) [DualDisplay](#) [VideoWindowHandle](#) [DualDisplay](#) [Visible](#) [DualDisplay](#) [Width](#) [IsVideoPortAvailable](#) [Monitor](#) [Primary](#) [Index](#) [MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#) [SetParentWindow](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Keep Your PDFs Safe from Unauthorized Access with These Security Measures](#)

ShapeOverlayList

TVideoGrabber.ShapeOverlayList

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

List of TShape components that have to be drawn over video frames.

Declaration

function ShapeOverlayList(Shape: TShape; Add: Boolean): Boolean;

bool __fastcall ShapeOverlayList(ExtCtrls::TShape *Shape, **bool** Add)

n/a

Description

Used to add/remove TShape components in the list of TShape components that will be drawn over video frames when [ShapeOverlayEnabled](#) is true.

- to add a TShape, put a TShape component (or create it dynamically at runtime), and then simply call [ShapeOverlayList](#) (YourShapeComponent, true).
- to remove a TShape component from the list, call [ShapeOverlayList](#) (YourShapeComponent, false).
- the function returns true upon success.

Each The TShape component in the list will be drawn at its Left, Top, Right, Height coordinates if its Enabled property is true.

It is possible to disable its Visible property whether it must not be visible on the form.

Note: to use only a single TShape component, it is possible to assign it to the [ShapeOverlay](#) property, instead of using this function.

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#)

[MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#)
[OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#)
[SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#)
[SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#)
[SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#)
[SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#)
[SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#)
[SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#)
[SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#)
[SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#)
[SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#)
[SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#)
[SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#)
[ShapeOverlayEnabled](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#) [TextOverlay](#) [CreateCustomFont](#)
[TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#) [TextOverlay](#) [FontColor](#)
[TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#) [TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#)
[TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#) [TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#)
[TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Qt Help documentation made easy](#)

ShowDebugWindow

TVideoGrabber.ShowDebugWindow

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Opens a window showing the TVideoGrabber log

Declaration

procedure ShowDebugWindow;

void __fastcall ShowDebugWindow(**void**)

Sub ShowDebugWindow()

Description

Invoke ShowDebugWindow() for debugging purpose.

This will popup a window that will report the TVideoGrabber log events.

This may be helpful for tracking more easily the errors reported by TVideoGrabber.

Created with the Standard Edition of HelpNDoc: [Write EPub books for the iPad](#)

ShowDialog

TVideoGrabber.ShowDialog

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Display a [TDialog](#) dialog.

Declaration

function ShowDialog(Dialog: TDialog): Boolean;

bool __fastcall ShowDialog(TDialog Dialog)

Function ShowDialog(Dialog As TxDIALOG) As Boolean

Description

Used to display a [TDialog](#) dialog, if available for the current video capture device (test [IsDialogAvailable](#) for availability).

See Also

[Player features](#) [TNetworkStreamingType](#) [TOnClientConnection](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [WDM drivers](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#) [TOnPlayerBufferingData](#) [ASFAudioBitRate](#) [ASFAudioChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#) [ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#) [ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#) [ASFProfileFromCustomFile](#) [ASFProfiles](#) [ASFProfilesCount](#) [ASFProfileVersion](#) [ASFVideoBitRate](#) [ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoConnectRelatedPins](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [CurrentFrameRate](#) [DeliveredFrames](#) [DroppedFrames](#) [FastForwardPlayer](#) [FixFlickerOrBlackCapture](#) [IsDVDevice](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [IsVideoControlAvailable](#) [IsVideoCrossbarAvailable](#) [IsVideoDeviceConnected](#) [IsWDMVideoDriver](#) [Last_Clip](#) [Played](#) [MP4NeedsReindexing](#) [NetworkStreaming](#) [NetworkStreamingType](#) [OnAuthenticationNeeded](#) [OnClientConnection](#) [OnDirectNetworkStreamingHostUrl](#) [OnNoVideoDevices](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackBarPosition](#) [OnVideoDeviceSelected](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchronise](#) [PlayerVideoCodec](#) [RefreshDevicesAndCompressorsLists](#) [ResetVideoDeviceSettings](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [SourceStream](#) [StartAudioRendering](#) [StopPlayer](#) [StreamingURL](#) [SynchronizationRole](#) [Synchronized](#) [VideoDevice](#) [VideoDeviceIndex](#) [VideoDeviceIndexFromId](#) [VideoDeviceName](#) [VideoDevices](#) [VideoDevicesCount](#) [VideoDevicesId](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Productivity with a Help Authoring Tool](#)

StartAudioRecording

TVideoGrabber.StartAudioRecording

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Starts the recording of the audio stream only.

Declaration

function StartAudioRecording: Boolean;

bool __fastcall StartAudioRecording(**void**)

Function StartAudioRecording() As Boolean

Description

Used to start the recording of the audio stream only by using the current [audio capture device](#).

However if the current [video capture device](#) has an audio output, it will be firstly used, so if you wish to be sure use the current [audio capture device](#) and not the possible audio stream coming out the current [video capture device](#), set [VideoDevice](#) = -1 to keep the video devices disabled.

When the **HoldRecording** parameter is set to "false", the recording starts as soon as the recording graph is built (after a variable number of seconds, depending of the video capture device, audio/video compressors, etc...).

When the **HoldRecording** parameter is set to "true", the recording graph is built and started in preview mode, and the recording is held until [ResumeRecording](#) has been invoked.

When the recording graph is built, the [OnRecordingReadyToStart](#) event occurs to let you know that you can invoke [ResumeRecording](#) when you want.

See the [recording](#) chapter for more information.

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last](#) [Recording](#) [FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: Upgrade your help files and your workflow with HelpNDoc's WinHelp HLP to CHM conversion

StartAudioRendering

TVideoGrabber.StartAudioRendering

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Starts the rendering of the audio stream only.

Declaration

function StartAudioRendering: Boolean;

bool __fastcall StartAudioRendering(**void**)

Function StartAudioRendering() As Boolean

Description

Used to start the rendering of the audio stream only by using the current [audio capture device](#).

However if the current [video capture device](#) has an audio output, it will be firstly used, so if you wish to be sure use the current [audio capture device](#) and not the possible audio stream coming out the current [video capture device](#), set [VideoDevice](#) = -1 to keep the video devices disabled.

This function designed to activate the audio broadcasting to the network. See the [Audio broadcasting](#) chapter.

See Also

[TNetworkStreamingType](#) [TOnClientConnection](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#) [ASFAudioBitRate](#) [ASFAudioChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#) [ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#) [ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#) [ASFProfileFromCustomFile](#) [ASFProfiles](#) [ASFProfilesCount](#) [ASFProfileVersion](#) [ASFVideoBitRate](#) [ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#) [NetworkStreaming](#) [NetworkStreamingType](#) [OnAuthenticationNeeded](#) [OnClientConnection](#) [OnDirectNetworkStreamingHostUrl](#) [ShowDialog](#) [StreamingURL](#)

StartPreview

TVideoGrabber.StartPreview

[Prev](#)[Next](#)[TVideoGrabber](#) [Methods](#)

Starts preview.

Declaration

function StartPreview: Boolean;

bool __fastcall StartPreview(**void**)

Function StartPreview() As Boolean

Description

Used to start the preview using the current [video capture device](#) (and [audio capture device](#) , whether [AudioRendering](#) is enabled).

See [Preview overview](#) .

See Also

[TOnFrameProgress](#) [TOnResizeVideo](#) [AutoRefreshPreview](#) [BackgroundColor](#) [Display](#) [AutoSize](#) [Display_FullScreen](#) [Display_SetLocation](#) [Display_VideoPortEnabled](#) [ImageRatio](#) [InFrameProgressEvent](#) [IsVideoInterlaced](#) [IsVideoPortAvailable](#) [OnFrameProgress](#) [OnKeyPress](#) [OnPreviewStarted](#) [OnResizeVideo](#) [PausePreview](#) [PreviewZoomSize](#) [ResumePreview](#) [StartSynchronized](#) [StopPreview](#) [Synchronized](#) [UseNearestVideoSize](#) [VideoHeight](#) [VideoRenderer](#) [VideoRendererPriority](#) [VideoWidth](#) [Visible](#)

StartRecording

TVideoGrabber.StartRecording

[Prev](#)[Next](#)[TVideoGrabber](#) [Methods](#)

Starts the recording.

Declaration

function StartRecording: Boolean;

bool __fastcall StartRecording(**void**)

Function StartRecording() As Boolean

Description

Used to start the recording by using the current [video capture device](#) (and [audio capture device](#) , if [AudioRecording](#) is enabled).

When the **HoldRecording** parameter is set to "false", the recording starts as soon as the recording graph is built (after a variable number of seconds, depending of the video capture device, audio/video compressors, etc...).

When the **HoldRecording** parameter is set to "true", the recording graph is built and started in preview mode, and the recording is held until [ResumeRecording](#) has been invoked.

When the recording graph is built, the [OnRecordingReadyToStart](#) event occurs to let you know that you can invoke [ResumeRecording](#) when you want.

See the [recording](#) chapter for more information.

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last](#) [Recording](#) [FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [Create help files for the Qt Help Framework](#)

StartReencoding

TVideoGrabber.StartReencoding

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Starts the reencoding of a video clip.

Declaration

function StartReencoding: Boolean;

bool __fastcall StartReencoding(**void**)

Function StartReencoding() As Boolean

Description

Used to starts the reencoding of a video clip, according to the Reencoding_... properties.
See the [Reencoding video clips](#) chapter for more information.

See Also

[OnReencodingCompleted](#) [OnReencodingProgress](#) [OnReencodingStarted](#) [Reencoding](#) [IncludeAudioStream](#) [Reencoding](#) [IncludeVideoStream](#) [Reencoding](#) [Method](#) [Reencoding](#) [NewVideoClip](#) [Reencoding](#) [SourceVideoClip](#) [Reencoding](#) [StartFrame](#) [Reencoding](#) [StartTime](#) [Reencoding](#) [StopFrame](#) [Reencoding](#) [StopTime](#) [Reencoding](#) [UseAudioCompressor](#) [Reencoding](#) [UseFrameGrabber](#) [Reencoding](#) [UseVideoCompressor](#) [Reencoding](#) [WMVOutput](#) [StopReencoding](#)

Created with the Standard Edition of HelpNDoc: [Streamline Your CHM Help File Creation with HelpNDoc](#)

StartSynchronized

TVideoGrabber.StartSynchronized

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Starts the preview or recording simultaneously on several components

Declaration

function StartSynchronized: Boolean;

bool __fastcall StartSynchronized(**void**)

Function StartSynchronized() As Boolean

Description

Used to start the preview or the recording at the same time on several TVideoGrabber components.

See the [Synchronization of several TVideoGrabber components](#) chapter.

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnFrameProgress](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TOnResizeVideo](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AutoRefreshPreview](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [BackgroundColor](#) [Display](#) [AutoSize](#) [Display](#) [FullScreen](#) [Display](#) [SetLocation](#) [Display](#) [VideoPortEnabled](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [ImageRatio](#) [InFrameProgressEvent](#) [IsRecordingPaused](#) [IsVideoInterlaced](#) [IsVideoPortAvailable](#) [Last](#) [Recording](#) [FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnFrameProgress](#) [OnKeyPress](#) [OnPreviewStarted](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [OnResizeVideo](#) [PausePreview](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [PreviewZoomSize](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumePreview](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartPreview](#) [StartRecording](#) [StopPreview](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [UseNearestVideoSize](#) [VideoHeight](#) [VideoPlayableWhileRecording](#) [VideoRenderer](#) [VideoRendererPriority](#) [VideoWidth](#) [Visible](#)

Created with the Standard Edition of HelpNDoc: [Easily create Web Help sites](#)

Stop

TVideoGrabber.Stop

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Stops any task currently running.

Declaration

function Stop: Boolean;

void __fastcall Stop(**void**)

Function Stop() As Boolean

Description

Stops any task currently running (preview, recording, playback as well as reencoding).

This function can be used instead of [StopPreview](#), [StopRecording](#), [StopReencoding](#) or [ClosePlayer](#).

Returns true upon success.

Created with the Standard Edition of HelpNDoc: [From Word to ePub or Kindle eBook: A Comprehensive Guide](#)

StopPlayer

TVideoGrabber.StopPlayer

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Stops playing a video clip.

Declaration

procedure StopPlayer;

void __fastcall StopPlayer(**void**)

Sub StopPlayer()

Description

Used to stop a video clip currently playing.
See the [Media Player](#) chapter.

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchron](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Full-featured Help generator](#)

StopPreview

TVideoGrabber.StopPreview

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Stops preview.

Declaration

procedure StopPreview;

void __fastcall StopPreview(**void**)

Sub StopPreview()

Description

Used to stop the current preview, previously started by [StartPreview](#) .

See Also

[TOnFrameProgress](#) [TOnResizeVideo](#) [AutoRefreshPreview](#) [BackgroundColor](#) [Display](#) [AutoSize](#) [Display_FullScreen](#) [Display_SetLocation](#) [Display_VideoPortEnabled](#) [ImageRatio](#) [InFrameProgressEvent](#) [IsVideoInterlaced](#) [IsVideoPortAvailable](#) [OnFrameProgress](#) [OnKeyPress](#) [OnPreviewStarted](#) [OnResizeVideo](#) [PausePreview](#) [PreviewZoomSize](#) [ResumePreview](#) [StartPreview](#) [StartSynchronized](#) [Synchronized](#) [UseNearestVideoSize](#) [VideoHeight](#) [VideoRenderer](#) [VideoRendererPriority](#) [VideoWidth](#) [Visible](#)

Created with the Standard Edition of HelpNDoc: [Free help authoring tool](#)

StopRecording

TVideoGrabber.StopRecording

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Stops the current recording.

Declaration

procedure StopRecording;

void __fastcall StopRecording(**void**)

Sub StopRecording()

Description

Used to stop the current recording, previously started by [StartRecording](#) .

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last](#) [Recording](#) [FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [Easily convert your WinHelp HLP help files to CHM with HelpNDoc's step-by-step guide](#)

StopReencoding

TVideoGrabber.StopReencoding

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Stops the reencoding of a video clip.

Declaration

function StopReencoding: Boolean;

bool __fastcall StopReencoding(**void**)

Function StopReencoding() As Boolean

Description

Used to stop the reencoding of a video clip before the process ends.
See the [Reencoding video clips](#) chapter for more information.

See Also

[OnReencodingCompleted](#) [OnReencodingProgress](#) [OnReencodingStarted](#) [Reencoding](#) [IncludeAudioStream](#) [Reencoding](#) [IncludeVideoStream](#) [Reencoding](#) [Method](#) [Reencoding](#) [NewVideoClip](#) [Reencoding](#) [SourceVideoClip](#) [Reencoding](#) [StartFrame](#) [Reencoding](#) [StartTime](#) [Reencoding](#) [StopFrame](#) [Reencoding](#) [StopTime](#) [Reencoding](#) [UseAudioCompressor](#) [Reencoding](#) [UseFrameGrabber](#) [Reencoding](#) [UseVideoCompressor](#) [Reencoding](#) [WMVOutput](#) [StartReencoding](#)

Created with the Standard Edition of HelpNDoc: [Make CHM Help File Creation a Breeze with HelpNDoc](#)

StreamInterface_PushData

TVideoGrabber.StreamInterface_PushData

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Methods](#)

Pushes the raw samples into the stream.

Declaration

function StreamInterface_PushData (Buffer: PByte; BufferSize: LongInt): LongInt;

int __fastcall StreamInterface_PushData(System::PByte Buffer, **int** BufferSize);

Description

Invoke this function to push the raw samples (e.g. H264) to (re)build the stream.
Look at the [Stream Interface](#) chapter for more information.

See Also

[Stream Interface](#)

Created with the Standard Edition of HelpNDoc: [Easily create iPhone documentation](#)

TextOverlay_CreateCustomFont

TVideoGrabber.TextOverlay_CreateCustomFont

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Methods](#)

Lets create a custom font for the current text overlay selected with [TextOverlay_Selector](#)

Declaration

function TextOverlay_CreateCustomFont(fHeight: LongInt; fWidth: LongInt; fEscapement: LongInt; fOrientation: LongInt; fWeight: LongInt; fItalic: Boolean; fUnderline: Boolean; fStrikeOut: Boolean; fCharSet: LongWord_; fOutputPrecision: LongWord_; fClipPrecision: LongWord_; fQuality: LongWord_; fPitchAndFamily: LongWord_; FontFacename: **string**): Boolean;

bool __fastcall TextOverlay_CreateCustomFont(**int** fHeight, **int** fWidth, **int** fEscapement, **int** fOrientation, **int** fWeight, **bool** fItalic, **bool** fUnderline, **bool** fStrikeOut, **unsigned** fCharSet, **unsigned** fOutputPrecision, **unsigned** fClipPrecision, **unsigned** fQuality, **unsigned** fPitchAndFamily, System::wchar_t *FontFacename)

Function TextOverlay_CreateCustomFont(fHeight as Long, fWidth as Long, fEscapement as Long, fOrientation as Long, fWeight as Long, fItalic as Boolean, fUnderline as Boolean, fStrikeOut as Boolean, fCharSet as Long, fOutputPrecision as Long, fClipPrecision as Long, fQuality as Long, fPitchAndFamily as Long, FontFacename as string) as Boolean

Description

Used to create a custom font used by the text overlay

E.g.:

```
VideoGrabber.TextOverlay_Selector = 4
VideoGrabber.TextOverlay_CreateCustomFont (16, 30, 0, 0, 200, true, false, false, 2, 0, 0, 0, 0,
"Wingdings")
VideoGrabber.TextOverlay_Enabled = true
```

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [GetTextOverlay_AlphaBlend](#) [GetTextOverlay_AlphaBlendValue](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#)

[OnFrameOverlayUsingVIDEOHDR](#)
[OnMouseDown](#)
[OnMouseMove](#)
[OnMouseUp](#)
[OnMouseWheel](#)
[OverlayAfterTransform](#)
[RefreshPlayerOverlays](#)
[SetImageOverlay](#)
[AlphaBlend](#)
[SetImageOverlay](#)
[AlphaBlendValue](#)
[SetImageOverlay](#)
[ChromaKey](#)
[SetImageOverlay](#)
[ChromaKeyLeewayPercent](#)
[SetImageOverlay](#)
[ChromaKeyRGBColor](#)
[SetImageOverlay](#)
[Enabled](#)
[SetImageOverlay](#)
[Height](#)
[SetImageOverlay](#)
[LeftLocation](#)
[SetImageOverlay](#)
[RotationAngle](#)
[SetImageOverlay](#)
[StretchToVideoSize](#)
[SetImageOverlay](#)
[TargetDisplay](#)
[SetImageOverlay](#)
[TopLocation](#)
[SetImageOverlay](#)
[Transparent](#)
[SetImageOverlay](#)
[TransparentColorValue](#)
[SetImageOverlay](#)
[UseTransparentColor](#)
[SetImageOverlay](#)
[Width](#)
[SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#)
[SetImageOverlayFromHBitmap2](#)
[SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#)
[SetImageOverlayFromJPEGFile](#)
[SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#)
[SetImageOverlayFromTImage](#)
[SetImageOverlayFromTImage2](#)
[SetTextOverlay](#)
[Align](#)
[SetTextOverlay](#)
[AlphaBlend](#)
[SetTextOverlay](#)
[AlphaBlendValue](#)
[SetTextOverlay](#)
[BkColor](#)
[SetTextOverlay](#)
[CustomVar](#)
[SetTextOverlay](#)
[Enabled](#)
[SetTextOverlay](#)
[Font](#)
[SetTextOverlay](#)
[FontColor](#)
[SetTextOverlay](#)
[GradientColor](#)
[SetTextOverlay](#)
[GradientMode](#)
[SetTextOverlay](#)
[HighResFont](#)
[SetTextOverlay](#)
[Left](#)
[SetTextOverlay](#)
[Right](#)
[SetTextOverlay](#)
[Scrolling](#)
[SetTextOverlay](#)
[ScrollingSpeed](#)
[SetTextOverlay](#)
[Shadow](#)
[SetTextOverlay](#)
[ShadowColor](#)
[SetTextOverlay](#)
[ShadowDirection](#)
[SetTextOverlay](#)
[String](#)
[SetTextOverlay](#)
[TargetDisplay](#)
[SetTextOverlay](#)
[Top](#)
[SetTextOverlay](#)
[Transparent](#)
[ShapeOverlay](#)
[ShapeOverlayEnabled](#)
[ShapeOverlayList](#)
[TextOverlay](#)
[Align](#)
[TextOverlay](#)
[BkColor](#)
[TextOverlay](#)
[CreateCustomFont2](#)
[TextOverlay](#)
[Enabled](#)
[TextOverlay](#)
[Font](#)
[TextOverlay](#)
[FontColor](#)
[TextOverlay](#)
[Left](#)
[TextOverlay](#)
[Right](#)
[TextOverlay](#)
[Scrolling](#)
[TextOverlay](#)
[ScrollingSpeed](#)
[TextOverlay](#)
[Selector](#)
[TextOverlay](#)
[Shadow](#)
[TextOverlay](#)
[ShadowColor](#)
[TextOverlay](#)
[ShadowDirection](#)
[TextOverlay](#)
[String](#)
[TextOverlay](#)
[Top](#)
[TextOverlay](#)
[Transparent](#)
[TextOverlay](#)
[VideoAlignment](#)
[TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Easy EBook and documentation generator

TextOverlay_CreateCustomFont2

TVideoGrabber.TextOverlay_CreateCustomFont2

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Lets create a custom font for the text overlay specified by index

Declaration

function TextOverlay_CreateCustomFont(Index: LongInt; fHeight: LongInt; fWidth: LongInt; fEscapement: LongInt; fOrientation: LongInt; fWeight: LongInt; fItalic: Boolean; fUnderline: Boolean; fStrikeOut: Boolean; fCharSet: LongWord_; fOutputPrecision: LongWord_; fClipPrecision: LongWord_; fQuality: LongWord_; fPitchAndFamily: LongWord_; FontFacename: **string**): Boolean;

bool __fastcall TextOverlay_CreateCustomFont(int Index, int fHeight, int fWidth, int fEscapement, int fOrientation, int fWeight, **bool** fItalic, **bool** fUnderline, **bool** fStrikeOut, **unsigned** fCharSet, **unsigned** fOutputPrecision, **unsigned** fClipPrecision, **unsigned** fQuality, **unsigned** fPitchAndFamily, System::wchar_t *FontFacename)

Function TextOverlay_CreateCustomFont(Index as Long, fHeight as Long, fWidth as Long, fEscapement as Long, fOrientation as Long, fWeight as Long, fItalic as Boolean, fUnderline as Boolean, fStrikeOut as Boolean, fCharSet as Long, fOutputPrecision as Long, fClipPrecision as Long, fQuality as Long, fPitchAndFamily as Long, FontFacename as string) as Boolean

Description

Used to create a custom font used by the text overlay having the specified index

E.g.: for the text overlay #5 (having the index **4**):

```
VideoGrabber.TextOverlay_CreateCustomFont2 (4, 16, 30, 0, 0, 200, true, false, false, 2, 0, 0, 0, 0, "Wingdings")
VideoGrabber.SetTextOverlay_Enabled (4, true)
```

See Also

[TCardinalDirection](#)
[TOnFrameOverlayUsingDC](#)
[TOnFrameOverlayUsingDIB](#)
[TTextOverlayAlign](#)
[AdjustOverlayAspectRatio](#)
[DrawBitmapOverFrame](#)
[GetFrameInfoString](#)
[GetTextOverlay](#)
[AlphaBlend](#)
[GetTextOverlay](#)
[AlphaBlendValue](#)
[ImageOverlay](#)
[StretchToVideoSize](#)
[MouseWheelEventEnabled](#)

[OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [AlphaBlend](#) [SetTextOverlay](#) [AlphaBlendValue](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#) [SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#) [SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#) [SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#) [SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#) [TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#) [TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#) [TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#) [TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#) [TextOverlay](#) [VideoAlignment](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Say Goodbye to Documentation Headaches with a Help Authoring Tool](#)

ThirdPartyFilter_AddToList

TVideoGrabber.ThirdPartyFilter_AddToList

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Adds a [third-party filter](#) to the list specified.

Declaration

function ThirdPartyFilter_AddToList (Location: TThirdPartyFilterList; GUIDString: **string**; OptionalDLLFilePath: **string**; FilterName: **string**; Enable: Boolean; CanSaveFilterState: Boolean): Boolean;

bool __fastcall ThirdPartyFilter_AddToList(TThirdPartyFilterList Location, wchar_t *GUIDString, wchar_t *OptionalDLLFilePath, wchar_t *FilterName, **bool** Enable, **bool** CanSaveFilterState);

Function ThirdPartyFilter_AddToList(Location As TxThirdPartyFilterList, GUIDString As String, OptionalDLLFilePath as String, FilterName As String, Enable As Boolean, SaveFilterState As Boolean) As Long

Description

Used to add a [third-party DirectShow filter](#) to a TThirdPartyFilterList list.

The function returns the index of the filter in the Location list. This index is necessary to access later the filter through other ThirdPartyFilter... functions.

Return -1 upon failure (e.g. if the filter is already in the list).

Location: [TThirdPartyFilterList](#) insertion point in the preview (or recording or play back) graph.

GUID: Guid of the filter to insert.

Name: name string associated to the filter (any value)

Enable: enables or disables the filter for the next preview (or recording or play back).

SaveFilterState: if true, the filter state will be saved in the registry and retrieved automatically for the next preview (or recording or play back).

See Also

[TThirdPartyFilterList](#) [TOnThirdPartyFilterConnected](#) [OnThirdPartyFilterConnected](#) [ThirdPartyFilter_ClearList](#) [ThirdPartyFilter_Enable](#) [ThirdPartyFilter_RemoveFromList](#) [ThirdPartyFilter_ShowDialog](#)

ThirdPartyFilter_ClearList

TVideoGrabber.ThirdPartyFilter_ClearList

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Clears a third-party filter list.

Declaration

function ThirdPartyFilter_ClearList(Location: TThirdPartyFilterList): Boolean;

bool __fastcall ThirdPartyFilter_ClearList(TThirdPartyFilterList Location)

Function ThirdPartyFilter_ClearList(Location As TThirdPartyFilterList) As Boolean

Description

Used to clear the third-party filter list specified.

Location: [TThirdPartyFilterList](#) location list to clear.

See Also

[TThirdPartyFilterList](#) [TOnThirdPartyFilterConnected](#) [OnThirdPartyFilterConnected](#)
[ThirdPartyFilter_AddToList](#) [ThirdPartyFilter_Enable](#) [ThirdPartyFilter_RemoveFromList](#)
[ThirdPartyFilter_ShowDialog](#)

ThirdPartyFilter_Enable

TVideoGrabber.ThirdPartyFilter_Enable

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Enables/disables a third-party filter.

Declaration

function ThirdPartyFilter_Enable(Location: TThirdPartyFilterList; **Index:** LongInt; Enable: Boolean): Boolean;

bool __fastcall ThirdPartyFilter_Enable(TThirdPartyFilterList Location, **int** Index, **bool** Enable)

Function ThirdPartyFilter_Enable(Location As TThirdPartyFilterList, Index As Long, Enable As Boolean) As Boolean

Description

Used to enable/disable a third-party filter already inserted in a list.

Location: [TThirdPartyFilterList](#) list in which the filter has been added with [ThirdPartyFilter_AddToList](#) .

index: Index of the filter in the **Location** list. This index is the one returned by [ThirdPartyFilter_AddToList](#) .

Enable: enables or disables the filter.

See Also

[TThirdPartyFilterList](#) [TOnThirdPartyFilterConnected](#) [OnThirdPartyFilterConnected](#)
[ThirdPartyFilter_AddToList](#) [ThirdPartyFilter_ClearList](#) [ThirdPartyFilter_RemoveFromList](#)
[ThirdPartyFilter_ShowDialog](#)

ThirdPartyFilter_RemoveFromList

TVideoGrabber.ThirdPartyFilter_RemoveFromList

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Removes a third-party filter from a list.

Declaration

function ThirdPartyFilter_RemoveFromList(Location: TThirdPartyFilterList; **Index**: LongInt): Boolean;

bool __fastcall ThirdPartyFilter_RemoveFromList(TThirdPartyFilterList Location, **int** Index)

Function ThirdPartyFilter_RemoveFromList(Location As TxThirdPartyFilterList, Index As Long) As Boolean

Description

Used to remove a third-party filter currently inserted in a list.

Location: [TThirdPartyFilterList](#) list in which the filter has been added with [ThirdPartyFilter_AddToList](#) .

index: Index of the filter in the **Location** list. This index is the one returned by [ThirdPartyFilter_AddToList](#) .

See Also

[TThirdPartyFilterList](#) [TOnThirdPartyFilterConnected](#) [OnThirdPartyFilterConnected](#)

[ThirdPartyFilter_AddToList](#) [ThirdPartyFilter_ClearList](#) [ThirdPartyFilter_Enable](#) [ThirdPartyFilter_ShowDialog](#)

Created with the Standard Edition of HelpNDoc: [From Word to ePub or Kindle eBook: A Comprehensive Guide](#)

ThirdPartyFilter_ShowDialog

TVideoGrabber.ThirdPartyFilter_ShowDialog

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Shows a third-party dialog.

Declaration

function ThirdPartyFilter_ShowDialog(Location: TThirdPartyFilterList; **Index**: LongInt): Boolean;

bool __fastcall ThirdPartyFilter_ShowDialog(TThirdPartyFilterList Location, **int** Index)

Function ThirdPartyFilter_ShowDialog(Location As TxThirdPartyFilterList, Index As Long) As Boolean

Description

Used to pop-up a third-party filter dialog, if available.

Location: [TThirdPartyFilterList](#) list in which the filter has been added with [ThirdPartyFilter_AddToList](#) .

index: Index of the filter in the **Location** list. This index is the one returned by [ThirdPartyFilter_AddToList](#) .

See Also

[TThirdPartyFilterList](#) [TOnThirdPartyFilterConnected](#) [OnThirdPartyFilterConnected](#)

[ThirdPartyFilter_AddToList](#) [ThirdPartyFilter_ClearList](#) [ThirdPartyFilter_Enable](#)

[ThirdPartyFilter_RemoveFromList](#)

Created with the Standard Edition of HelpNDoc: [Free help authoring environment](#)

TVClearFrequencyOverrides

TVideoGrabber.TVClearFrequencyOverrides

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Clears all the existing frequency overrides.

Declaration

function TVClearFrequencyOverrides: Boolean;

bool __fastcall TVClearFrequencyOverrides(**void**)

Function TVClearFrequencyOverrides() As Boolean

Description

Used to clear all the existing frequency overrides set for a given country code and tuner input type by [TVSetChannelFrequencyOverride](#) .

Important: setting or clearing a frequency override will be applied only after restarting preview. This will be done automatically if [AutoRefreshPreview](#) is enabled.

See Also

[TTVChannelInfo](#) [TOnTVChannelScanStarted](#) [TOnTVChannelSelected](#) [TTunerInputType](#) [GetTVChannelInfo](#) [IsHorizontalSyncLocked](#) [IsTVAudioAvailable](#) [IsTVAutoTuneRunning](#) [IsTVTunerAvailable](#) [OnTVChannelScanCompleted](#) [OnTVChannelScanStarted](#) [OnTVChannelSelected](#) [TunerFrequency](#) [TunerMode](#) [TVChannel](#) [TVCountryCode](#) [TVGetMinMaxChannels](#) [TVSetChannelFrequencyOverride](#) [TVStartAutoScan](#) [TVStopAutoScan](#) [TVTunerInputType](#) [TVUseFrequencyOverrides](#) [VCRHorizontalLocking](#)

Created with the Standard Edition of HelpNDoc: [Create cross-platform Qt Help files](#)

TVGetMinMaxChannels

TVideoGrabber.TVGetMinMaxChannels

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Retrieves the min and max TV channels for a given country code.

Declaration

function TVGetMinMaxChannels(**var** MinChannel: LongInt; **var** MaxChannel: LongInt): Boolean;

bool __fastcall TVGetMinMaxChannels(**int** &MinChannel, **int** &MaxChannel)

Function TVGetMinMaxChannels(Min_Channel, Max_Channel) As Boolean

Description

Used to retrieve the min and max TV channels for a given country code.
Be sure to select the proper [country code](#) before invoking this function.

See Also

[TTVChannelInfo](#) [TOnTVChannelScanStarted](#) [TOnTVChannelSelected](#) [TTunerInputType](#) [GetTVChannelInfo](#) [IsHorizontalSyncLocked](#) [IsTVAudioAvailable](#) [IsTVAutoTuneRunning](#) [IsTVTunerAvailable](#) [OnTVChannelScanCompleted](#) [OnTVChannelScanStarted](#) [OnTVChannelSelected](#) [TunerFrequency](#) [TunerMode](#) [TVChannel](#) [TVClearFrequencyOverrides](#) [TVCountryCode](#) [TVSetChannelFrequencyOverride](#) [TVStartAutoScan](#) [TVStopAutoScan](#) [TVTunerInputType](#) [TVUseFrequencyOverrides](#) [VCRHorizontalLocking](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Productivity with a Help Authoring Tool](#)

TVSetChannelFrequencyOverride

TVideoGrabber.TVSetChannelFrequencyOverride

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Sets a frequency overrides.

Declaration

function TVSetChannelFrequencyOverride(TVChannel: LongInt; FrequencyInHz: LongInt): LongInt;

int __fastcall TVSetChannelFrequencyOverride(**int** TVChannel, **int** FrequencyInHz)

Function TVSetChannelFrequencyOverride(TVChannel As Long, FrequencyInHz As Long) As Long

Description

Used to set a frequency override for a given TV channel, for the current [country code](#) and [tuner input type](#).

This function sets a frequency override on the specified TV channel.

This will affect the TV channel for the current [country code](#) and [tuner input type](#) only.

If you need to retrieve the default frequency for this TV channel, simply assign the channel number to the [TVChannel](#) property and the [OnTVChannelSelected](#) event will return the frequency (the event is synchrone).

To erase any frequency override on this TV channel simply pass " -1" as frequency value in the FrequencyInHz parameter.

Important: setting or clearing a frequency override will be applied only after restarting preview. This will be done automatically if [AutoRefreshPreview](#) is enabled.

See Also

[TTVChannelInfo](#) [TOnTVChannelScanStarted](#) [TOnTVChannelSelected](#) [TTunerInputType](#) [GetTVChannelInfo](#) [IsHorizontalSyncLocked](#) [IsTVAudioAvailable](#) [IsTVAutoTuneRunning](#) [IsTVTunerAvailable](#) [OnTVChannelScanCompleted](#) [OnTVChannelScanStarted](#) [OnTVChannelSelected](#) [TunerFrequency](#) [TunerMode](#) [TVChannel](#) [TVClearFrequencyOverrides](#) [TVCountryCode](#) [TVGetMinMaxChannels](#) [TVStartAutoScan](#) [TVStopAutoScan](#) [TVTunerInputType](#) [TVUseFrequencyOverrides](#) [VCRHorizontalLocking](#)

Created with the Standard Edition of HelpNDoc: [Experience the Power and Ease of Use of HelpNDoc for CHM Help File Generation](#)

TVStartAutoScan

TVideoGrabber.TVStartAutoScan

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Starts the automatic scanning of TV channels.

Declaration

function TVStartAutoScan: Boolean;

bool __fastcall TVStartAutoScan(**void**)

Function TVStartAutoScan() As Boolean

Description

Used to start the automatic scanning of TV channels for a given country code and tuner input type.

See Also

[TTVChannelInfo](#) [TOnTVChannelScanStarted](#) [TOnTVChannelSelected](#) [TTunerInputType](#) [GetTVChannelInfo](#) [IsHorizontalSyncLocked](#) [IsTVAudioAvailable](#) [IsTVAutoTuneRunning](#) [IsTVTunerAvailable](#) [OnTVChannelScanCompleted](#) [OnTVChannelScanStarted](#) [OnTVChannelSelected](#) [TunerFrequency](#) [TunerMode](#) [TVChannel](#) [TVClearFrequencyOverrides](#) [TVCountryCode](#) [TVGetMinMaxChannels](#) [TVSetChannelFrequencyOverride](#) [TVStopAutoScan](#) [TVTunerInputType](#) [TVUseFrequencyOverrides](#) [VCRHorizontalLocking](#)

TVStopAutoScan

TVideoGrabber.TVStopAutoScan

[Prev](#)[Next](#)[TVideoGrabber](#) [Methods](#)

Stops the automatic scanning of TV channels.

Declaration

function TVStopAutoScan: Boolean;

bool __fastcall TVStopAutoScan(**void**)

Function TVStopAutoScan() As Boolean

Description

Used to stop the automatic scanning of TV channels, previously started by invoking [TVStartAutoScan](#) .

See Also

[TTVChannelInfo](#) [TOnTVChannelScanStarted](#) [TOnTVChannelSelected](#) [TTunerInputType](#) [GetTVChannelInfo](#) [IsHorizontalSyncLocked](#) [IsTVAudioAvailable](#) [IsTVAutoTuneRunning](#) [IsTVTunerAvailable](#) [OnTVChannelScanCompleted](#) [OnTVChannelScanStarted](#) [OnTVChannelSelected](#) [TunerFrequency](#) [TunerMode](#) [TVChannel](#) [TVClearFrequencyOverrides](#) [TVCountryCode](#) [TVGetMinMaxChannels](#) [TVSetChannelFrequencyOverride](#) [TVStartAutoScan](#) [TVTunerInputType](#) [TVUseFrequencyOverrides](#) [VCRHorizontalLocking](#)

UseNearestVideoSize

TVideoGrabber.UseNearestVideoSize

[Prev](#)[Next](#)[TVideoGrabber](#) [Methods](#)

Automatically selects the nearest video source size to a preferred width and height.

Declaration

procedure UseNearestVideoSize(PreferredWidth, PreferredHeight: LongInt; Stretch: Boolean);

void __fastcall UseNearestVideoSize(**int** PreferredWidth, **int** PreferredHeight, **bool** Stretch)

Sub UseNearestVideoSize(PreferredWidth As Long, PreferredHeight As Long, Stretch As Boolean)

Description

Used to automatically use the video capture device size nearest to a preferred width and height.

- if **stretch** disabled, the control is resized to the nearest size found.
- if **stretch** enabled, the nearest size is selected, the control is resized to the preferred width and height, and the video window is stretched to fit the control.

E.g. we wish a 320x240 display size, and the video capture device offers only 176x144 and 352x288 sizes.

- UseNearestVideoSize (320, 240, false) selects the 352x288 video size and resize the display window to 352x288.
- UseNearestVideoSize (320, 240, true) selects the 352x288 video size and stretches the display window to 320x240.

Important note: when this function is called, the [VideoSize](#) property is ignored. To re-enable the [VideoSize](#) property, call UseNearestVideoSize (0, 0, false).

See Also

[GetNearestVideoHeight](#) [GetNearestVideoSize](#) [GetNearestVideoWidth](#) [GetVideoHeightFromIndex](#) [GetVideoSizeFromIndex](#) [GetVideoWidthFromIndex](#) [VideoFormat](#) [VideoFormatIndex](#) [VideoSize](#) [VideoSizeIndex](#) [VideoSizes](#) [VideoSizesCount](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Reach: Convert Your Word Document to an ePub or Kindle eBook](#)

v360_AddYawPitchRoll

TVideoGrabber.v360_AddYawPitchRoll

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

360° video: modifies the point of view

Declaration

function v360_AddYawPitchRoll (Yaw: Double; Pitch: Double; Roll: Double): Boolean;

bool v360_AddYawPitchRoll(**double** Yaw, **double** Pitch, **double** Roll);

Description

Used to increase or decrease the yaw, pitch or roll, in the -180.0 to 180.0 range.

E.g.:

v360_AddYawPitchRoll (-10.0, 5.5, 0.0);

See Also

[v360_AspectRatio](#) [v360_Enabled](#) [v360_GetAngle](#) [v360_GetYawPitchRoll](#) [v360_MasterAngle](#) [v360_MouseAction](#) [v360_MouseActionPercent](#) [v360_ResetAnglesToDefault](#) [v360_SetAngle](#) [v360_SetInterpolation](#) [v360_SetProjection](#) [v360_SetStereoFormat](#) [v360_SetTranspose](#) [v360_SetYawPitchRoll](#)

Created with the Standard Edition of HelpNDoc: [Easily create CHM Help documents](#)

v360_GetAngle

TVideoGrabber.v360_GetAngle

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

360° video: retrieve an angle

Declaration

function v360_GetAngle (Direction: Tv360_InOut; Angle: Tv360_Angle): Double;

double v360_GetAngle(Tv360_InOut Direction, Tv360_Angle Angle);

Description

Used to retrieve the current value of the input or output angles (vertical, horizontal or diagonal)

- **1st parameter:** [Tv360_InOut](#) Direction specifies the direction

- **2nd parameter:** [Tv360_Angle](#) Angle specify what angle must be set

The value returned is in the 0..180 range.

See Also

[v360_AddYawPitchRoll](#) [v360_AspectRatio](#) [v360_Enabled](#) [v360_GetYawPitchRoll](#) [v360_MasterAngle](#) [v360_MouseAction](#) [v360_MouseActionPercent](#) [v360_ResetAnglesToDefault](#) [v360_SetAngle](#) [v360_SetInterpolation](#) [v360_SetProjection](#) [v360_SetStereoFormat](#) [v360_SetTranspose](#) [v360_SetYawPitchRoll](#)

Created with the Standard Edition of HelpNDoc: [Experience the power of a responsive website for your](#)

v360_GetYawPitchRoll

TVideoGrabber.v360_GetYawPitchRoll

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

360° video: retrieves the point of view

Declaration

procedure v360_GetYawPitchRoll (**var** Yaw: Double; **var** Pitch: Double; **var** Roll: Double);

void v360_GetYawPitchRoll(**double** &Yaw, **double** &Pitch, **double** &Roll);

Description

Used to retrieve the current point of view (yaw, pitch and roll)

See Also

[v360_AddYawPitchRoll](#) [v360_AspectRatio](#) [v360_Enabled](#) [v360_GetAngle](#) [v360_MasterAngle](#) [v360_MouseAction](#) [v360_MouseActionPercent](#) [v360_ResetAnglesToDefault](#) [v360_SetAngle](#) [v360_SetInterpolation](#) [v360_SetProjection](#) [v360_SetStereoFormat](#) [v360_SetTranspose](#) [v360_SetYawPitchRoll](#)

Created with the Standard Edition of HelpNDoc: [Easily create HTML Help documents](#)

v360_ResetAnglesToDefault

TVideoGrabber.v360_ResetAnglesToDefault

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

360° video: reset the angles

Declaration

procedure v360_ResetAnglesToDefault();

void v360_ResetAnglesToDefault(**void**);

Description

Resets the angles to their default values

See Also

[v360_AddYawPitchRoll](#) [v360_AspectRatio](#) [v360_Enabled](#) [v360_GetAngle](#) [v360_GetYawPitchRoll](#) [v360_MasterAngle](#) [v360_MouseAction](#) [v360_MouseActionPercent](#) [v360_SetAngle](#) [v360_SetInterpolation](#) [v360_SetProjection](#) [v360_SetStereoFormat](#) [v360_SetTranspose](#) [v360_SetYawPitchRoll](#)

Created with the Standard Edition of HelpNDoc: [Easy to use tool to create HTML Help files and Help web sites](#)

v360_SetAngle

TVideoGrabber.v360_SetAngle

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

360° video: set an angle

Declaration

function v360_SetAngle (Direction: Tv360_InOut; Angle: Tv360_Angle; Value: Double): Boolean;

```
bool v360_SetAngle(Tv360_InOut Direction, Tv360_Angle Angle, double Value);
```

Description

Used to set the input or output angles (vertical, horizontal or diagonal)

- **1st parameter:** [Tv360_InOut](#) Direction specifies the direction
- **2nd parameter:** [Tv360_Angle](#) Angle specify what angle must be set
- **3rd parameter:** the value of the angle in the 0..180 range.

See Also

[v360_AddYawPitchRoll](#) [v360_AspectRatio](#) [v360_Enabled](#) [v360_GetAngle](#) [v360_GetYawPitchRoll](#)
[v360_MasterAngle](#) [v360_MouseAction](#) [v360_MouseActionPercent](#) [v360_ResetAnglesToDefault](#)
[v360_SetInterpolation](#) [v360_SetProjection](#) [v360_SetStereoFormat](#) [v360_SetTranspose](#)
[v360_SetYawPitchRoll](#)

Created with the Standard Edition of HelpNDoc: [Elevate Your CHM Help Files with HelpNDoc's Advanced Customization Options](#)

v360_SetInterpolation

TVideoGrabber.v360_SetInterpolation

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

360° video: sets the interpolation method

Declaration

```
function v360_SetInterpolation (Value: Tv360_Interpolation): Boolean;
```

```
bool v360_SetInterpolation(Tv360_Interpolation Value);
```

Description

Parameter of type [Tv360_Interpolation](#)

See Also

[v360_AddYawPitchRoll](#) [v360_AspectRatio](#) [v360_Enabled](#) [v360_GetAngle](#) [v360_GetYawPitchRoll](#)
[v360_MasterAngle](#) [v360_MouseAction](#) [v360_MouseActionPercent](#) [v360_ResetAnglesToDefault](#)
[v360_SetAngle](#) [v360_SetProjection](#) [v360_SetStereoFormat](#) [v360_SetTranspose](#) [v360_SetYawPitchRoll](#)

Created with the Standard Edition of HelpNDoc: [Converting Word Docs to eBooks Made Easy with HelpNDoc](#)

v360_SetProjection

TVideoGrabber.v360_SetProjection

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

360° video: sets the type of projection

Declaration

```
function v360_SetProjection (Direction: Tv360_InOut; Value: Tv360_Projection): Boolean;
```

```
bool v360_SetProjection(Tv360_InOut Direction, Tv360_Projection Value);
```

Description

Parameters:

Direction: [Tv360_InOut](#);

Value: [Tv360_Projection](#)

The default value are:

input: ipp_Equirectangular

output: ipp_Flat

See Also

[v360_AddYawPitchRoll](#) [v360_AspectRatio](#) [v360_Enabled](#) [v360_GetAngle](#) [v360_GetYawPitchRoll](#) [v360_MasterAngle](#) [v360_MouseAction](#) [v360_MouseActionPercent](#) [v360_ResetAnglesToDefault](#) [v360_SetAngle](#) [v360_SetInterpolation](#) [v360_SetStereoFormat](#) [v360_SetTranspose](#) [v360_SetYawPitchRoll](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly bring your documentation online with HelpNDoc](#)

v360_SetStereoFormat

TVideoGrabber.v360_SetStereoFormat

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

360° video: sets the stereo format

Declaration

function v360_SetStereoFormat (Direction: Tv360_InOut; Value: Tv360_StereoFormat): Boolean;

bool v360_SetStereoFormat(Tv360_InOut Direction, Tv360_StereoFormat Value);

Description

Used to set the input and/or output stereo format

Parameters:

Direction: [Tv360_InOut](#)

Value: [Tv360_StereoFormat](#)

See Also

[v360_AddYawPitchRoll](#) [v360_AspectRatio](#) [v360_Enabled](#) [v360_GetAngle](#) [v360_GetYawPitchRoll](#) [v360_MasterAngle](#) [v360_MouseAction](#) [v360_MouseActionPercent](#) [v360_ResetAnglesToDefault](#) [v360_SetAngle](#) [v360_SetInterpolation](#) [v360_SetProjection](#) [v360_SetTranspose](#) [v360_SetYawPitchRoll](#)

Created with the Standard Edition of HelpNDoc: [Converting Word Documents to eBooks: A Step-by-Step Guide with HelpNDoc](#)

v360_SetTranspose

TVideoGrabber.v360_SetTranspose

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

360° video: enables/disables the transposition

Declaration

function v360_SetTranspose (Direction: Tv360_InOut; Value: Boolean): Boolean;

bool v360_SetTranspose(Tv360_InOut Direction, **bool** Value);

Description

Parameters:

Direction: [Tv360_InOut](#)

Value: Boolean

See Also

[v360_AddYawPitchRoll](#) [v360_AspectRatio](#) [v360_Enabled](#) [v360_GetAngle](#) [v360_GetYawPitchRoll](#) [v360_MasterAngle](#) [v360_MouseAction](#) [v360_MouseActionPercent](#) [v360_ResetAnglesToDefault](#) [v360_SetAngle](#) [v360_SetInterpolation](#) [v360_SetProjection](#) [v360_SetStereoFormat](#) [v360_SetYawPitchRoll](#)

Created with the Standard Edition of HelpNDoc: [Achieve Professional Documentation Results with a Help](#)

v360_SetYawPitchRoll

TVideoGrabber.v360_SetYawPitchRoll

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

360° video: sets the point of view

Declaration

function v360_SetYawPitchRoll (Yaw: Double; Pitch: Double; Roll: Double): Boolean;

bool v360_SetYawPitchRoll(**double** Yaw, **double** Pitch, **double** Roll);

Description

Used to set the yaw, pitch or roll, in the -180.0 to 180.0 range.

E.g.:

v360_SetYawPitchRoll (12.0, 5.0, 0.0);

See Also

[v360_AddYawPitchRoll](#) [v360_AspectRatio](#) [v360_Enabled](#) [v360_GetAngle](#) [v360_GetYawPitchRoll](#) [v360_MasterAngle](#) [v360_MouseAction](#) [v360_MouseActionPercent](#) [v360_ResetAnglesToDefault](#) [v360_SetAngle](#) [v360_SetInterpolation](#) [v360_SetProjection](#) [v360_SetStereoFormat](#) [v360_SetTranspose](#)

Created with the Standard Edition of HelpNDoc: [Easily create Qt Help files](#)

VideoCompressorIndex

TVideoGrabber.VideoCompressorIndex

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns the index of a given compressor in the [VideoCompressors](#) list.

Declaration

function VideoCompressorIndex(Value: **String**): LongInt;

int __fastcall VideoCompressorIndex(wchar_t *Value)

Function VideoCompressorIndex(param1 As String) As Long

Description

Used to retrieve the index of a given compressor in the [VideoCompressors](#) list.

The function accepts wild chars. E.g.:

```
VideoGrabber1.VideoCompressor := VideoCompressorIndex ('Microsoft MPEG-4 Video Codec V2')
VideoGrabber1.VideoCompressor := VideoCompressorIndex ('Microsoft MPEG-4*');           // s
VideoGrabber1.VideoCompressor := VideoCompressorIndex ('*MPEG-4*');                   // selects the
```

See Also

[Recording methods and properties](#) [TCompressionType](#) [TOnVideoCompressionSettings](#) [AudioCompressor](#) [AudioCompressorIndex](#) [AudioCompressorName](#) [AudioCompressors](#) [AudioCompressorsCount](#) [CompressionMode](#) [CompressionType](#) [GetVideoCompressionSettings](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [OnVideoCompressionSettings](#) [RefreshDevicesAndCompressorsLists](#) [SaveCompressorSettingsToDataString](#) [SetVideoCompressionDefaults](#) [SetVideoCompressionSettings](#) [VideoCompression](#) [KeyFrameRate](#) [VideoCompression](#) [PFramesPerKeyFrame](#) [VideoCompression](#) [Quality](#)

[VideoCompression](#) [WindowSize](#) [VideoCompressor](#) [VideoCompressorName](#) [VideoCompressors](#)
[VideoCompressorsCount](#)

Created with the Standard Edition of HelpNDoc: [Transform Your CHM Help File Creation Process with HelpNDoc](#)

VideoDeviceIndex

TVideoGrabber.VideoDeviceIndex

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns the index of a given audio device in the [VideoDevices](#) list.

Declaration

function VideoDeviceIndex(Value: **String**): LongInt;

int __fastcall VideoDeviceIndex(wchar_t *Value)

Function VideoDeviceIndex(param1 As String) As Long

Description

Used to retrieve the index of a given video device in the [VideoDevices](#) list.

The function accepts wild chars. E.g.:

```
VideoGrabber1.VideoDevice := VideoDeviceIndex ('Microsoft DV Camera and VCR (WDM)'); // s
VideoGrabber1.VideoDevice := VideoDeviceIndex ('Microsoft DV Camera*');           // s
VideoGrabber1.VideoDevice := VideoDeviceIndex ('*DV Camera*');                     // s
```

See Also

[WDM drivers](#) [AutoConnectRelatedPins](#) [CurrentFrameRate](#) [DeliveredFrames](#) [DroppedFrames](#)
[FixFlickerOrBlackCapture](#) [IsDVDevice](#) [IsVideoControlAvailable](#) [IsVideoCrossbarAvailable](#)
[IsVideoDeviceConnected](#) [IsWDMVideoDriver](#) [OnNoVideoDevices](#) [OnVideoDeviceSelected](#)
[RefreshDevicesAndCompressorsLists](#) [ResetVideoDeviceSettings](#) [ShowDialog](#) [VideoDevice](#)
[VideoDeviceIndexFromId](#) [VideoDeviceName](#) [VideoDevices](#) [VideoDevicesCount](#) [VideoDevicesId](#)
[WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Revolutionize Your Documentation Output with HelpNDoc's Stunning User Interface](#)

VideoDeviceIndexFromId

TVideoGrabber.VideoDeviceIndexFromId

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns the index of a given video device in the [VideoDevicesId](#) list

Declaration

function VideoDeviceIndexFromId(Value: **String**): LongInt;

int __fastcall VideoDeviceIndexFromId(wchar_t *Value)

Function VideoDeviceIndexFromId(param1 As String) As Long

Description

Used to retrieve the index of a given video device in the [VideoDevicesId](#) list.

This lets select the device by its hardware identifier rather than its manufacturer name.

E.g.:

```
VideoGrabber1.VideoDevice := VideoDeviceIndexFromId
('devicepnpusbvid046dpid0826mi02739103c9a0000265e8773d-8f56-11d0-a3b9-
00a0c9223196globalHD Webcam C525');
```

See Also

[WDM drivers](#) [AutoConnectRelatedPins](#) [CurrentFrameRate](#) [DeliveredFrames](#) [DroppedFrames](#) [FixFlickerOrBlackCapture](#) [IsDVDevice](#) [IsVideoControlAvailable](#) [IsVideoCrossbarAvailable](#) [IsVideoDeviceConnected](#) [IsWDMVideoDriver](#) [OnNoVideoDevices](#) [OnVideoDeviceSelected](#) [RefreshDevicesAndCompressorsLists](#) [ResetVideoDeviceSettings](#) [ShowDialog](#) [VideoDevice](#) [VideoDeviceIndex](#) [VideoDeviceName](#) [VideoDevices](#) [VideoDevicesCount](#) [VideoDevicesId](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Simplify Your Help Documentation Process with a Help Authoring Tool](#)

VideoFormatIndex

TVideoGrabber.VideoFormatIndex

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

index of a video format

Declaration

function VideoFormatIndex(Value: **String**): LongInt;

int __fastcall VideoFormatIndex(wchar_t *Value)

Function VideoFormatIndex(Value As String) As Long

Description

Used to retrieve the index of a given video format in the [VideoFormats](#) list

See Also

[GetNearestVideoHeight](#) [GetNearestVideoSize](#) [GetNearestVideoWidth](#) [GetVideoHeightFromIndex](#) [GetVideoSizeFromIndex](#) [GetVideoWidthFromIndex](#) [UseNearestVideoSize](#) [VideoFormat](#) [VideoSize](#) [VideoSizeIndex](#) [VideoSizes](#) [VideoSizesCount](#)

Created with the Standard Edition of HelpNDoc: [Streamline Your CHM Help File Creation with HelpNDoc](#)

VideoFromImages_CreateSetOfBitmaps

TVideoGrabber.VideoFromImages_CreateSetOfBitmaps

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Creates a set of bitmaps that will be used to create a video clip.

Declaration

function VideoFromImages_CreateSetOfBitmaps: Boolean;

bool __fastcall VideoFromImages_CreateSetOfBitmaps(**void**)

Function VideoFromImages_CreateSetOfBitmaps() As Boolean

Description

Creates a set of bitmaps stored in a temporary file, that will be used to create a video clip from bitmaps.
See [Video clip from bitmaps overview](#).

Created with the Standard Edition of HelpNDoc: [Step-by-Step Guide: How to Turn Your Word Document into an eBook](#)

VideoInputIndex

TVideoGrabber.VideoInputIndex

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns the index of a given audio input in the [VideoInputs](#) list.

Declaration

function VideoInputIndex(Value: **String**): LongInt;

int __fastcall VideoInputIndex(wchar_t *Value)

Function VideoInputIndex(param1 As String) As Long

Description

Used to retrieve the index of a given audio input in the [VideoInputs](#) list.

The function accepts wild chars. E.g.:

```
VideoGrabber1.VideoInput := VideoInputIndex ('SVideo'); // selects the "SVideo" video input
VideoGrabber1.VideoInput := VideoInputIndex ('SVideo*'); // selects the 1st video input
VideoGrabber1.VideoInput := VideoInputIndex ('*SVideo*'); // selects the 1st video input
```

See Also

[Video inputs](#) [VideoInput](#) [VideoInputs](#) [VideoInputsCount](#)

Created with the Standard Edition of HelpNDoc: [Modernize your help files with HelpNDoc's WinHelp HLP to CHM conversion tool](#)

VideoQualityAuto

TVideoGrabber.VideoQualityAuto

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Reports if a [TVideoQuality](#) setting is in "auto" mode.

Declaration

function GetVideoQualityAuto(Setting: TVideoQuality): Boolean;

bool __fastcall GetVideoQualityAuto(TVideoQuality Setting)

Function GetVideoQualityAuto(Setting As TxVideoQuality) As Boolean

Description

Used to know if a [TVideoQuality](#) setting is in "auto" mode or "manual" mode.
Returns "true" if the setting is in "auto" mode, "false" if it is in "manual" mode.

See Also

[IsVideoQualityAvailable](#) [IsVideoQualitySettingAvailable](#) [SetVideoQuality](#) [VideoQualityDefault](#) [VideoQualityMax](#) [VideoQualityMin](#) [VideoQualitySettings](#) [VideoQualityStep](#) [VideoRendererPriority](#)

[TVideoQuality](#)

Created with the Standard Edition of HelpNDoc: [Revolutionize Your CHM Help File Output with HelpNDoc](#)

VideoQualityDefault

TVideoGrabber.VideoQualityDefault

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Resets a [TVideoQuality](#) setting.

Declaration

function VideoQualityDefault(Setting: TVideoQuality): LongInt;

int __fastcall VideoQualityDefault(TVideoQuality Setting)

Function VideoQualityDefault(Setting As TxVideoQuality) As Long

Description

Used to reset a [TVideoQuality](#) setting to its default value.
Returns MAXINT upon failure.

See Also

[IsVideoQualityAvailable](#) [IsVideoQualitySettingAvailable](#) [SetVideoQuality](#) [VideoQualityAuto](#) [VideoQualityMax](#) [VideoQualityMin](#) [VideoQualitySettings](#) [VideoQualityStep](#) [VideoRendererPriority](#) [TVideoQuality](#)

Created with the Standard Edition of HelpNDoc: [Full-featured multi-format Help generator](#)

VideoQualityMax

TVideoGrabber.VideoQualityMax

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Max value of a [TVideoQuality](#) setting.

Declaration

function VideoQualityMax(Setting: TVideoQuality): LongInt;

int __fastcall VideoQualityMax(TVideoQuality Setting)

Function VideoQualityMax(Setting As TxVideoQuality) As Long

Description

Used to retrieve the max value of a [TVideoQuality](#) setting.
Returns MAXINT upon failure.

See Also

[IsVideoQualityAvailable](#) [IsVideoQualitySettingAvailable](#) [SetVideoQuality](#) [VideoQualityAuto](#) [VideoQualityDefault](#) [VideoQualityMin](#) [VideoQualitySettings](#) [VideoQualityStep](#) [VideoRendererPriority](#) [TVideoQuality](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Create High-Quality Documentation with a Help Authoring Tool](#)

VideoQualityMin

TVideoGrabber.VideoQualityMin

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

ber

Min value of a [TVideoQuality](#) setting.

Declaration

function VideoQualityMin(Setting: TVideoQuality): LongInt;

int __fastcall VideoQualityMin(TVideoQuality Setting)

Function VideoQualityMin(Setting As TVideoQuality) As Long

Description

Used to retrieve the min value of a [TVideoQuality](#) setting.
Returns MAXINT upon failure.

See Also

[IsVideoQualityAvailable](#) [IsVideoQualitySettingAvailable](#) [SetVideoQuality](#) [VideoQualityAuto](#)
[VideoQualityDefault](#) [VideoQualityMax](#) [VideoQualitySettings](#) [VideoQualityStep](#) [VideoRendererPriority](#)
[TVideoQuality](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Create High-Quality Documentation with a Help Authoring Tool](#)

VideoQualityStep

TVideoGrabber.VideoQualityStep

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Stepping value of a [TVideoQuality](#) setting.

Declaration

function VideoQualityStep(Setting: TVideoQuality): LongInt;

int __fastcall VideoQualityStep(TVideoQuality Setting)

Function VideoQualityStep(Setting As TVideoQuality) As Long

Description

Used to retrieve the stepping value of a [TVideoQuality](#) setting.
Returns MAXINT upon failure.

See Also

[IsVideoQualityAvailable](#) [IsVideoQualitySettingAvailable](#) [SetVideoQuality](#) [VideoQualityAuto](#)
[VideoQualityDefault](#) [VideoQualityMax](#) [VideoQualityMin](#) [VideoQualitySettings](#) [VideoRendererPriority](#)
[TVideoQuality](#)

Created with the Standard Edition of HelpNDoc: [Streamline Your Documentation Process with a Help Authoring Tool](#)

VideoQualityValue

TVideoGrabber.VideoQualityValue

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Gets a [TVideoQuality](#) current value.

Declaration

function GetVideoQuality(Setting: TVideoQuality): LongInt;

int __fastcall GetVideoQuality(TVideoQuality Setting)

Function GetVideoQuality(Setting As TxVideoQuality) As Long

Description

Used to retrieves the current value of a [TVideoQuality](#) setting.
Returns MAXINT upon failure.

See Also

[IsVideoQualityAvailable](#) [IsVideoQualitySettingAvailable](#) [SetVideoQuality](#) [VideoQualityAuto](#)
[VideoQualityDefault](#) [VideoQualityMax](#) [VideoQualityMin](#) [VideoQualitySettings](#) [VideoQualityStep](#)
[VideoRendererPriority](#) [TVideoQuality](#)

Created with the Standard Edition of HelpNDoc: [Don't Let Unauthorized Users View Your PDFs: Learn How to Set Passwords](#)

VideoSizeIndex

TVideoGrabber.VideoSizeIndex

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns the index of a given audio input in the [VideoSizes](#) list.

Declaration

function VideoSizeIndex(Value: **String**): LongInt;

int __fastcall VideoSizeIndex(wchar_t *Value)

Function VideoSizeIndex(param1 As String) As Long

Description

Used to retrieve the index of a given audio input in the [VideoSizes](#) list.

The function accepts wild chars. E.g.:

```
VideoGrabber1.Videosize := VideosizeIndex ('720x576'); // selects the "720x576" video size
VideoGrabber1.Videosize := VideosizeIndex ('720x*');    // selects the 1st video size t
VideoGrabber1.Videosize := VideosizeIndex (*720x*');    // selects the 1st video size t
```

See Also

[GetNearestVideoHeight](#) [GetNearestVideoSize](#) [GetNearestVideoWidth](#) [GetVideoHeightFromIndex](#)
[GetVideoSizeFromIndex](#) [GetVideoWidthFromIndex](#) [UseNearestVideoSize](#) [VideoFormat](#) [VideoFormatIndex](#)
[VideoSize](#) [VideoSizes](#) [VideoSizesCount](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Edit and Export Markdown Documents](#)

VideoSubtypeIndex

TVideoGrabber.VideoSubtypeIndex

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Methods](#)

Returns the index of a given audio input in the [VideoSubtypes](#) list.

Declaration

function VideoSubtypeIndex(Value: **String**): LongInt;

int __fastcall VideoSubtypeIndex(wchar_t *Value)

Function VideoSubtypeIndex(param1 As String) As Long

Description

Used to retrieve the index of a given audio input in the [VideoSubtypes](#) list.

The function accepts wild chars. E.g.:

```
VideoGrabber1.VideoSubtype := VideoSubtypeIndex ('RGB24'); // selects the "RGB24" video subtype
VideoGrabber1.VideoSubtype := VideoSubtypeIndex ('RGB*'); // selects the 1st video subtype
VideoGrabber1.VideoSubtype := VideoSubtypeIndex ('*RGB*'); // selects the 1st video subtype
```

See Also

[VideoSubtype](#) [VideoSubtypes](#) [VideoSubtypesCount](#)

Created with the Standard Edition of HelpNDoc: [Benefits of a Help Authoring Tool](#)

WriteScriptCommand

TVideoGrabber.WriteScriptCommand

[Prev](#)

[TVideoGrabber](#) [Methods](#)

Not implemented

Declaration

function WriteScriptCommand (ScriptType: **string**; ScriptArgument: **string**; ScriptTime: int64): Boolean;

bool __fastcall WriteScriptCommand (wchar_t *ScriptType, wchar_t *ScriptArgument, __int64 ScriptTime) ;

function WriteScriptCommand (ScriptType as String, ScriptArgument as String, ScriptTime as Double) as Boolean

Description

Not implemented

Created with the Standard Edition of HelpNDoc: [Streamline Your Documentation Process with a Help Authoring Tool](#)

Events

TVideoGrabber Events

[TVideoGrabber](#)

Public

[OnAudioBufferNegotiation](#)
[OnAudioDeviceSelected](#)
[OnAudioPeak](#)
[OnAuthenticationNeeded](#)
[OnAVIDurationUpdated](#)
[OnBacktimedFramesCountReached](#)
[OnBitmapsLoadingProgress](#)
[OnClick](#)
[OnClientConnection](#)

[OnColorKeyChange](#)
[OnCopyPreallocDataCompleted](#)
[OnCopyPreallocDataProgress](#)
[OnCopyPreallocDataStarted](#)
[OnCreatePreallocFileCompleted](#)
[OnCreatePreallocFileProgress](#)
[OnCreatePreallocFileStarted](#)
[OnDbClick](#)
[OnDeviceArrivalOrRemoval](#)
[OnDeviceLost](#)
[OnDeviceReconnected](#)
[OnDeviceReconnecting](#)
[OnDirectNetworkStreamingHostUrl](#)
[OnDiskFull](#)
[OnDragDropFiles](#)
[OnDVCommandCompleted](#)
[OnDVDDiscontinuity](#)
[OnEnumerateWindows](#)
[OnFilterSelected](#)
[OnFrameBitmap](#)
[OnFrameCaptureCompleted](#)
[OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingDIB](#)
[OnFrameOverlayUsingVIDEOHDR](#)
[OnFrameProgress](#)
[OnFrameProgress2](#)
[OnGraphBuilt](#)
[OnInactive](#)
[OnKeyPress](#)
[OnLastCommandCompleted](#)
[OnLeavingFullScreen](#)
[OnLog](#)
[OnMotionDetected](#)
[OnMotionNotDetected](#)
[OnMouseDown](#)
[OnMouseDown_Video](#)
[OnMouseDown_Window](#)
[OnMouseMove](#)
[OnMouseMove_Video](#)
[OnMouseMove_Window](#)
[OnMouseUp](#)
[OnMouseUp_Video](#)
[OnMouseUp_Window](#)
[OnMouseWheel](#)
[OnNoVideoDevices](#)
[OnNTPTimeStamp](#)
[OnONVIFDiscoveryCompletedNotification](#)
[OnOpenURLAsyncStatusChanged](#)
[OnPlayerBufferingData](#)
[OnPlayerEndOfPlaylist](#)
[OnPlayerEndOfStream](#)
[OnPlayerOpened](#)
[OnPlayerStateChanged](#)
[OnPlayerUpdateTrackbarPosition](#)
[OnPreviewStarted](#)
[OnRawAudioSample](#)
[OnRawVideoSample](#)
[OnRecordingCompleted](#)
[OnRecordingPaused](#)
[OnRecordingReadyToStart](#)
[OnRecordingStarted](#)
[OnReencodingCompleted](#)
[OnReencodingProgress](#)

[OnReencodingStarted](#)
[OnReinitializing](#)
[OnResizeVideo](#)
[OnTextOverlayScrollingCompleted](#)
[OnThirdPartyFilterConnected](#)
[OnThreadSync](#)
[OnTVChannelScanCompleted](#)
[OnTVChannelScanStarted](#)
[OnTVChannelSelected](#)
[OnVideoCompressionSettings](#)
[OnVideoDeviceSelected](#)
[OnVideoFromBitmapsNextFrameNeeded](#)

Created with the Standard Edition of HelpNDoc: [Converting Word Documents to eBooks: A Step-by-Step Guide with HelpNDoc](#)

OnAudioBufferNegotiation

TVideoGrabber.OnAudioBufferNegotiation

[Next](#)

[TVideoGrabber](#) [Events](#)

Lets choose the audio buffer size when rendering the audio stream

Declaration

property OnAudioBufferNegotiation: TEventNotification **read** FOnAudioBufferNegotiation **write** FOnAudioBufferNegotiation;

__property TOnAudioBufferNegotiation OnAudioBufferNegotiation=read=FOnAudioBufferNegotiation, write=FOnAudioBufferNegotiation

Event OnAudioBufferNegotiation

Description

This event occurs when rendering the audio stream, allowing to change the default audio buffer size, by modifying the ProposedBufferSize parameter from this event

Type: [TOnAudioBufferNegotiation](#)

See Also

[TAudioFormat](#) [AssociateAudioAndVideoDevices](#) [AudioBalance](#) [AudioDevice](#) [AudioDeviceIndex](#)
[AudioDeviceName](#) [AudioDeviceRendering](#) [AudioDevices](#) [AudioDevicesCount](#) [AudioFormat](#) [AudioFormats](#)
[AudioInput](#) [AudioInputIndex](#) [AudioInputLevel](#) [AudioInputMono](#) [AudioInputs](#) [AudioInputsCount](#) [AudioSource](#)
[IsAudioCrossbarAvailable](#) [IsAudioDeviceASoundCard](#) [IsAudioDeviceConnected](#)
[IsAudioInputBalanceAvailable](#) [OnAudioDeviceSelected](#) [SpeakerBalance](#) [SpeakerControl](#) [SpeakerVolume](#)

Created with the Standard Edition of HelpNDoc: [Converting Word Docs to eBooks Made Easy with HelpNDoc](#)

OnAudioDeviceSelected

TVideoGrabber.OnAudioDeviceSelected

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when an audio capture device is selected.

Declaration

property OnAudioDeviceSelected: TEventNotification **read** FOnAudioDeviceSelected **write** FOnAudioDeviceSelected;

__property TEventNotification OnAudioDeviceSelected=read=FOnAudioDeviceSelected, write=FOnAudioDeviceSelected

Event OnAudioDeviceSelected()

Description

Occurs when an audio capture device is selected by the [AudioDevice](#) property. The device-dependent settings of this audio capture device are reloaded and this event occurs.

See Also

[TAudioFormat](#) [AssociateAudioAndVideoDevices](#) [AudioBalance](#) [AudioDevice](#) [AudioDeviceIndex](#) [AudioDeviceName](#) [AudioDeviceRendering](#) [AudioDevices](#) [AudioDevicesCount](#) [AudioFormat](#) [AudioFormats](#) [AudioInput](#) [AudioInputIndex](#) [AudioInputLevel](#) [AudioInputMono](#) [AudioInputs](#) [AudioInputsCount](#) [AudioSource](#) [IsAudioCrossbarAvailable](#) [IsAudioDeviceASoundCard](#) [IsAudioDeviceConnected](#) [IsAudioInputBalanceAvailable](#) [OnAudioBufferNegotiation](#) [SpeakerBalance](#) [SpeakerControl](#) [SpeakerVolume](#)

Created with the Standard Edition of HelpNDoc: [Make Documentation a Breeze with HelpNDoc's Clean and Efficient User Interface](#)

OnAudioPeak

TVideoGrabber.OnAudioPeak

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Returns the audio peak levels, in percentage or in DB.

Declaration

property OnAudioPeak: TOnAudioPeak **read** FOnAudioPeak **write** FOnAudioPeak;

__property TOnAudioPeak OnAudioPeak=read=FOnAudioPeak, write=FOnAudioPeak

Event OnAudioPeak()

Description

This [TOnAudioPeak](#) event returns the audio peak levels for the left and right audio channels, in percentage and in DB.

The [AudioPeakEvent](#) property must be enabled to activate this event.

Created with the Standard Edition of HelpNDoc: [Write eBooks for the Kindle](#)

OnAuthenticationNeeded

TVideoGrabber.OnAuthenticationNeeded

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs to pass a user name and password when an authentication is required

Declaration

property OnAuthenticationNeeded: TOnAuthenticationNeeded **read** FOnAuthenticationNeeded **write** FOnAuthenticationNeeded;

__property TOnAuthenticationNeeded OnAuthenticationNeeded=read=FOnAuthenticationNeeded, write=OnAuthenticationNeeded

Event OnAuthenticationNeeded(Id As Long, Realm as String, Server as String, UserName, Password)

Description

This [TOnAuthenticationNeeded](#) event occurs when an authentication is needed, e.g. to send streaming to a publishing point.

Parameters:

Id: reserved

Realm: reports the domain name, if any

Server: reports the server name, if any

UserName: set this property with the user name needed for the authentication

Password: set this parameter with the password needed for the authentication

See Also

[TNetworkStreamingType](#) [TOnClientConnection](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#) [ASFAudioBitRate](#) [ASFAudioChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#) [ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#) [ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#) [ASFProfileFromCustomFile](#) [ASFProfiles](#) [ASFProfilesCount](#) [ASFProfileVersion](#) [ASFVideoBitRate](#) [ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#) [NetworkStreaming](#) [NetworkStreamingType](#) [OnClientConnection](#) [OnDirectNetworkStreamingHostUrl](#) [ShowDialog](#) [StartAudioRendering](#) [StreamingURL](#)

Created with the Standard Edition of HelpNDoc: [Write eBooks for the Kindle](#)

OnAVIDurationUpdated

TVideoGrabber.OnAVIDurationUpdated

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when the AVI recording ends and the [AVIDurationUpdated](#) property is enabled.

Declaration

property OnAVIDurationUpdated: TOnAVIDurationUpdated **read** FOnAVIDurationUpdated **write** FOnAVIDurationUpdated;

__property TOnAVIDurationUpdated OnAVIDurationUpdated=read=FOnAVIDurationUpdated, write=FOnAVIDurationUpdated

Event OnAVIDurationUpdated(NewStateValue As Long, NewStateLabel As String)

Description

Occurs when closing an AVI file just recorded and the [AVIDurationUpdated](#) property is enabled.

This [TOnAVIDurationUpdated](#) event lets you update the frame rate and/or duration of the AVI clip, if needed.

This event is designed only to give access to the frame rate and duration of the AVI. **Be sure not to use it as notification for the end of recording.**

The end of the recording is notified by the [OnRecordingCompleted](#) event, whatever the kind of recording (AVI, ASF, MPEG).

Parameters:

FileName: returns name of the file recorded,

Frames: returns the number of frames recorded,

FrameRate: returns the real frame rate (can be adjusted),

Duration: returns the duration of the recording (can be adjusted).

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#)

[AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last](#) [Recording](#) [FileName](#)
[OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#)
[OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#)
[OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#)
[OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#)
[PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#)
[RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#)
[RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#)
[RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#)
[ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#)
[StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#)
[VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: Bring your WinHelp HLP help files into the present with HelpNDoc's easy CHM conversion

OnBacktimedFramesCountReached

TVideoGrabber.OnBacktimedFramesCountReached

[Prev](#)
[Next](#)

[TVideoGrabber](#)
[Events](#)

Occurs the when [AVIBacktimedVideoFramesCount](#) is enabled (> 0) and back-timed video frames start replacing normal video frames in the video stream.

Declaration

property OnBacktimedFramesCountReached: TEventNotification **read**
 FOnBacktimedFramesCountReached **write** FOnBacktimedFramesCountReached;

__property TEventNotification

OnBacktimedFramesCountReached=read=FOnBacktimedFramesCountReached,
 write=FOnBacktimedFramesCountReached

Event OnBacktimedFramesCountReached()

Description

Occurs the when [AVIBacktimedVideoFramesCount](#) is enabled (> 0) and back-timed video frames start replacing normal video frames in the video stream.

See [AVIBacktimedVideoFramesCount](#) for more information.

See Also

[Color / Greyscale](#) [Grid structure / grid sensitivity](#) [Motion ratio](#) [Recording only when motion is detected](#)
[Video noise](#) [TOnMotionDetected](#) [TOnMotionNotDetected](#) [MotionDetector](#) [CellMotionRatio](#)
[MotionDetector](#) [CompareBlue](#) [MotionDetector](#) [CompareGreen](#) [MotionDetector](#) [CompareRed](#)
[MotionDetector](#) [Enabled](#) [MotionDetector](#) [EnumGridDialogControls](#) [MotionDetector](#) [Get2DTextGrid](#)
[MotionDetector](#) [Get2DTextMotion](#) [MotionDetector](#) [GetCellLocation](#) [MotionDetector](#) [GetCellSensitivity](#)
[MotionDetector](#) [GetCellSize](#) [MotionDetector](#) [GloballyIncOrDecSensitivity](#)
[MotionDetector](#) [GlobalMotionRatio](#) [MotionDetector](#) [GreyScale](#) [MotionDetector](#) [Grid](#)
[MotionDetector](#) [GridXCount](#) [MotionDetector](#) [GridYCount](#) [MotionDetector](#) [IsGridValid](#)
[MotionDetector](#) [MaxDetectionsPerSecond](#) [MotionDetector](#) [ReduceCPULoad](#)
[MotionDetector](#) [ReduceVideoNoise](#) [MotionDetector](#) [Reset](#) [MotionDetector](#) [ResetGlobalSensitivity](#)
[MotionDetector](#) [SetCellSensitivity](#) [MotionDetector](#) [SetGridSize](#) [MotionDetector](#) [ShowGridDialog](#)
[MotionDetector](#) [Triggered](#) [MotionDetector](#) [UseThisReferenceSample](#) [OnMotionDetected](#)
[OnMotionNotDetected](#) [RecordingOnMotion](#) [Enabled](#) [RecordingOnMotion](#) [MotionThreshold](#)
[RecordingOnMotion](#) [NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: Transform Your Word Document into a Professional eBook with HelpNDoc

OnBitmapsLoadingProgress

TVideoGrabber.OnBitmapsLoadingProgress

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs periodically when building a set of bitmap that will be used to create a video clip.

Declaration

property OnBitmapsLoadingProgress: TOnProgressCommented **read** FOnBitmapsLoadingProgress **write** FOnBitmapsLoadingProgress;

__property TOnProgressCommented OnBitmapsLoadingProgress=read=FOnBitmapsLoadingProgress, write=FOnBitmapsLoadingProgress

Event OnBitmapsLoadingProgress(Percent As Long, Position As Double, Duration As Double, Comment As String)

Description

This [TOnProgressCommented](#) event occurs periodically when invoking [VideoFromImages_CreateSetOfBitmaps](#) to build a set of bitmap that will be used to create a video clip.

See Also

[SendImageToVideoFromBitmaps](#) [SendImageToVideoFromBitmaps2](#) [VideoFromImages](#) [BitmapsSortedByVideoFromImages](#) [RepeatIndefinitely](#) [VideoFromImages_SourceDirectory](#) [VideoFromImages_TemporaryFile](#)

Created with the Standard Edition of HelpNDoc: [Transform Your CHM Help File Creation Process with HelpNDoc](#)

OnClick

TVideoGrabber.OnClick

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

normal OnClick event

Declaration

property OnClick: TEventNotification **read** FOnClick **write** FOnClick;

__property TEventNotification OnClick=read=FOnClick, write=FOnClick

Event OnClick()

Description

normal OnClick event

Created with the Standard Edition of HelpNDoc: [Experience the power of a responsive website for your documentation](#)

OnClientConnection

TVideoGrabber.OnClientConnection

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when a network client connects or disconnects

Declaration

property OnClientConnection: TOnClientConnection **read** FOnClientConnection **write** FOnClientConnection;

__**property** TOnClientConnection OnClientConnection=read=FOnClientConnection, write=FOnClientConnection

Event OnClientConnection (Connected as Boolean, ClientInfo as String)

Description

This [TOnClientConnection](#) event occurs when the streaming is running (NetworkStreaming <> ns_Disabled), and a client connects or disconnects.

Parameters

Connected: connection = true, disconnection = false

ClientInfo: returns the client information (usually IP:port)

See Also

[TNetworkStreamingType](#) [TOnClientConnection](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#) [ASFVideoBitRate](#) [ASFVideoChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#) [ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#) [ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#) [ASFProfileFromCustomFile](#) [ASFProfiles](#) [ASFProfilesCount](#) [ASFProfileVersion](#) [ASFVideoBitRate](#) [ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#) [NetworkStreaming](#) [NetworkStreamingType](#) [OnAuthenticationNeeded](#) [OnDirectNetworkStreamingHostUrl](#) [ShowDialog](#) [StartAudioRendering](#) [StreamingURL](#)

Created with the Standard Edition of HelpNDoc: [Experience the power of a responsive website for your documentation](#)

OnColorKeyChange

TVideoGrabber.OnColorKeyChange

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Returns the color key for transparent windows.

Declaration

property OnColorKeyChange: TOnColorKeyChange **read** FOnColorKeyChange **write** FOnColorKeyChange;

__**property** TOnColorKeyChange OnColorKeyChange=read=FOnColorKeyChange, write=FOnColorKeyChange

Event OnColorKeyChange(ColorKey As OLE_COLOR)

Description

When [ColorKeyEnabled](#) is enabled, this [TOnColorKeyChange](#) event occurs when starting preview, recording or playback.

This event returns the color key to use for transparent windows.

See the VideoGrabber1ColorKeyChange event in the MainDemo project for sample code.

See Also

[TOnColorKeyChange](#) [ColorKeyEnabled](#)

Created with the Standard Edition of HelpNDoc: [Full-featured Documentation generator](#)

OnCopyPreallocDataCompleted

TVideoGrabber.OnCopyPreallocDataCompleted

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs after the preallocated recording file has been fully copied.

Declaration

property OnCopyPreallocDataCompleted: TOnSourceFileToDestFileCompleted **read** FOnCopyPreallocDataCompleted **write** FOnCopyPreallocDataCompleted;

```
__property TOnSourceFileToDestFileCompleted
OnCopyPreallocDataCompleted=read=FOnCopyPreallocDataCompleted,
write=FOnCopyPreallocDataCompleted
```

Event OnCopyPreallocDataCompleted(SourceFile As String, DestFile As String, Success As Boolean)

Description

This [TOnSourceFileToDestFileCompleted](#) event is used when [PreallocCapFileEnabled](#) is true (a preallocated huge capture file is used for recording).
Occurs when the streams data has been fully copied from the pre-allocated AVI file to the final AVI file.
As a result, the new file can be much smaller than the preallocated file.

See Also

[TOnCreatePreallocatedFileCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#)
[OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileProgress](#) [OnCreatePreallocFileStarted](#)

Created with the Standard Edition of HelpNDoc: [Create iPhone web-based documentation](#)

OnCopyPreallocDataProgress

TVideoGrabber.OnCopyPreallocDataProgress

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs periodically during the preallocated file copy.

Declaration

property OnCopyPreallocDataProgress: TOnProgress **read** FOnCopyPreallocDataProgress **write** FOnCopyPreallocDataProgress;

```
__property TOnProgress OnCopyPreallocDataProgress=read=FOnCopyPreallocDataProgress,
write=FOnCopyPreallocDataProgress
```

Event OnCopyPreallocDataProgress(Percent As Long, Position As Double, Duration As Double)

Description

This [TOnProgress](#) event occurs periodically during the preallocated file copy.
Used when [PreallocCapFileEnabled](#) is true (a preallocated huge capture file is used for recording).
Occurs when the streams data has been fully copied from the pre-allocated AVI file to the final AVI file.

See Also

[TOnCreatePreallocatedFileCompleted](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataStarted](#)
[OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileProgress](#) [OnCreatePreallocFileStarted](#)

Created with the Standard Edition of HelpNDoc: [Write EPub books for the iPad](#)

OnCopyPreallocDataStarted

TVideoGrabber.OnCopyPreallocDataStarted

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when the preallocated recording file begins to be copied.

Declaration

property OnCopyPreallocDataStarted: TOnSourceFileToDestFileStarted **read**
FOnCopyPreallocDataStarted **write** FOnCopyPreallocDataStarted;

__property TOnSourceFileToDestFileStarted
OnCopyPreallocDataStarted=read=FOnCopyPreallocDataStarted, write=FOnCopyPreallocDataStarted

Event OnCopyPreallocDataStarted(SourceFile As String, DestFile As String)

Description

This [TOnSourceFileToDestFileStarted](#) event is used when [PreallocCapFileEnabled](#) is true (a preallocated huge capture file is used for recording).

Occurs when the streams data begins to be copied from the pre-allocated AVI file to the final AVI file.

See Also

[TOnCreatePreallocatedFileCompleted](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#)
[OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileProgress](#) [OnCreatePreallocFileStarted](#)

Created with the Standard Edition of HelpNDoc: [Keep Your Sensitive PDFs Safe with These Easy Security Measures](#)

OnCreatePreallocFileCompleted

TVideoGrabber.OnCreatePreallocFileCompleted

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when the preallocated recording file has been fully created.

Declaration

property OnCreatePreallocFileCompleted: TOnCreatePreallocatedFileCompleted **read**
FOnCreatePreallocFileCompleted **write** FOnCreatePreallocFileCompleted;

__property TOnCreatePreallocatedFileCompleted
OnCreatePreallocFileCompleted=read=FOnCreatePreallocFileCompleted,
write=FOnCreatePreallocFileCompleted

Event OnCreatePreallocFileCompleted(FileName As String, Success As Boolean)

Description

This [TOnCreatePreallocatedFileCompleted](#) event is used when [PreallocCapFileEnabled](#) is true (a preallocated huge capture file is used for recording).

Occurs when the preallocated recording has been fully created.

See Also

[TOnCreatePreallocatedFileCompleted](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#)
[OnCopyPreallocDataStarted](#) [OnCreatePreallocFileProgress](#) [OnCreatePreallocFileStarted](#)

Created with the Standard Edition of HelpNDoc: [Achieve Professional Documentation Results with a Help Authoring Tool](#)

OnCreatePreallocFileProgress

TVideoGrabber.OnCreatePreallocFileProgress

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Events](#)

Occurs periodically during the creation of the preallocated file.

Declaration

property OnCreatePreallocFileProgress: TOnProgress **read** FOnCreatePreallocFileProgress **write** FOnCreatePreallocFileProgress;

__property TOnProgress OnCreatePreallocFileProgress=read=FOnCreatePreallocFileProgress, write=FOnCreatePreallocFileProgress

Event OnCreatePreallocFileProgress(Percent As Long, Position As Double, Duration As Double)

Description

This [TOnProgress](#) event occurs periodically during the creation of the preallocated file used for recording.

See Also

[TOnCreatePreallocatedFileCompleted](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#)

Created with the Standard Edition of HelpNDoc: [Upgrade Your Documentation Process with a Help Authoring Tool](#)

OnCreatePreallocFileStarted

TVideoGrabber.OnCreatePreallocFileStarted

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Events](#)

Occurs when the preallocated recording file begins to be created.

Declaration

property OnCreatePreallocFileStarted: TOnFileNotification **read** FOnCreatePreallocFileStarted **write** FOnCreatePreallocFileStarted;

__property TOnFileNotification OnCreatePreallocFileStarted=read=FOnCreatePreallocFileStarted, write=FOnCreatePreallocFileStarted

Event OnCreatePreallocFileStarted(FileName As String)

Description

This [TOnFileNotification](#) event is used when [PreallocCapFileEnabled](#) is true (a preallocated huge capture file is used for recording).

Occurs when the preallocated recording file begins to be created.

See Also

[TOnCreatePreallocatedFileCompleted](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileProgress](#)

Created with the Standard Edition of HelpNDoc: [Single source CHM, PDF, DOC and HTML Help creation](#)

OnDbIClick

TVideoGrabber.OnDbIClick

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Events](#)

[ber](#)

normal OnDbClick event

Declaration

property OnDbClick: TEventNotification **read** FOnDbClick **write** FOnDbClick;

__property TEventNotification OnDbClick=read=FOnDbClick, write=FOnDbClick

Event OnDbClick()

Description

normal OnDbClick event

Created with the Standard Edition of HelpNDoc: [5 Reasons Why a Help Authoring Tool is Better than Microsoft Word for Documentation](#)

OnDeviceArrivalOrRemoval

TVideoGrabber.OnDeviceArrivalOrRemoval

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs each time a video capture device is plugged-in or removed.

Declaration

property OnDeviceArrivalOrRemoval: TOnDeviceArrivalOrRemoval **read** FOnDeviceArrivalOrRemoval **write** FOnDeviceArrivalOrRemoval;

__property TOnDeviceArrivalOrRemoval OnDeviceArrivalOrRemoval=read=FOnDeviceArrivalOrRemoval, write=FOnDeviceArrivalOrRemoval

Event OnDeviceArrivalOrRemoval(IsDeviceArrival As Boolean, IsVideoDevice As Boolean, DeviceName As String, DeviceIndex As Long)

Description

This [TOnDeviceArrivalOrRemoval](#) event occurs each time a video capture device is plugged-in or removed and returns information about this device: audio or video device, connected or disconnected, device name.

The [VideoDevices](#) list or [AudioDevices](#) list is updated if a new capture device has been plugged-in.

If a video capture device has been unplugged, it remains in the list. To determine if a device is connected or not, test the IsVideoDeviceConnected (DeviceIndex: LongInt): Boolean function that will return false after the device removal.

When the video capture device is plugged-in again, IsVideoDeviceConnected (DeviceIndex: LongInt): Boolean returns true again.

(Use IsAudioDeviceConnected to test audio devices from the [AudioDevices](#) list)

Note: device arrival/removal detection works only with WDM drivers, "old" VFW capture device appear always in the device lists, even if not connected.

See Also

[TOnDeviceArrivalOrRemoval](#) [OnDeviceLost](#) [RefreshDevicesAndCompressorsLists](#)

Created with the Standard Edition of HelpNDoc: [Experience the Power and Ease of Use of a Help Authoring Tool](#)

OnDeviceLost

TVideoGrabber.OnDeviceLost

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Events](#)

Occurs when a capture device is disconnected during preview or recording.

Declaration

property OnDeviceLost: TEventNotification **read** FOnDeviceLost **write** FOnDeviceLost;

__property TEventNotification OnDeviceLost=read=FOnDeviceLost, write=FOnDeviceLost

Event OnDeviceLost()

Description

Occurs when a capture device is disconnected during preview or recording.

This event occurs for IP cameras when the connection is lost and the auto-reconnection is disabled (see [Auto-reconnection](#))

See Also

[TOnDeviceArrivalOrRemoval](#) [OnDeviceArrivalOrRemoval](#) [RefreshDevicesAndCompressorsLists](#)

Created with the Standard Edition of HelpNDoc: [Modernize your help files with HelpNDoc's WinHelp HLP to CHM conversion tool](#)

OnDeviceReconnected

TVideoGrabber.OnDeviceReconnected

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Events](#)

Occurs when an IP camera is reconnected after the connection has been lost

Declaration

property OnDeviceReconnected: TEventNotification **read** FOnDeviceReconnected **write** FOnDeviceReconnected;

__property TEventNotification OnDeviceReconnected=read=FOnDeviceReconnected, write=FOnDeviceReconnected

Event OnDeviceReconnected()

Description

Occurs when an IP camera is reconnected, after the connection has been lost

See [Auto-reconnection](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Documentation Output with HelpNDoc's Advanced Project Analyzer](#)

OnDeviceReconnecting

TVideoGrabber.OnDeviceReconnecting

[Prev](#)
[Next](#)
[TVideoGrabber](#) [Events](#)

Occurs when the connection to an IP camera has been lost and the RTSP filter tries to reconnect

Declaration

property OnDeviceReconnecting: TEventNotification **read** FOnDeviceReconnecting **write** FOnDeviceReconnecting;

__**property** TEventNotification OnDeviceReconnecting=read=FOnDeviceReconnecting, write=FOnDeviceReconnecting

Event OnDeviceReconnecting()

Description

Occurs when the connection to an IP camera has been lost and the RTSP filter tries to reconnect
See [Auto-reconnection](#)

Created with the Standard Edition of HelpNDoc: [Easily create Web Help sites](#)

OnDirectNetworkStreamingHostUrl

TVideoGrabber.OnDirectNetworkStreamingHostUrl

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Returns the host url to use for direct network streaming.

Declaration

property OnDirectNetworkStreamingHostUrl: TOnDirectNetworkStreamingHostUrl **read** FOnDirectNetworkStreamingHostUrl **write** FOnDirectNetworkStreamingHostUrl;

__**property** TOnDirectNetworkStreamingHostUrl
OnDirectNetworkStreamingHostUrl=read=FOnDirectNetworkStreamingHostUrl, write=FOnDirectNetworkStreamingHostUrl

Event OnDirectNetworkStreamingHostUrl(HostUrl As String, HostName As String, HostPort As Long)

Description

This [TOnDeviceArrivalOrRemoval](#) event is used to retrieve the host url when performing direct streaming.
Each user must connect to this URL on the platform where TVideoGrabber is streaming.
Returns the full URL, the host name and the port.

See Also

[TNetworkStreamingType](#) [TOnClientConnection](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#) [ASFVideoBitRate](#) [ASFVideoChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#) [ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#) [ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#) [ASFProfileFromCustomFile](#) [ASFProfiles](#) [ASFProfilesCount](#) [ASFProfileVersion](#) [ASFVideoBitRate](#) [ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#) [NetworkStreaming](#) [NetworkStreamingType](#) [OnAuthenticationNeeded](#) [OnClientConnection](#) [ShowDialog](#) [StartAudioRendering](#) [StreamingURL](#)

Created with the Standard Edition of HelpNDoc: [Easy to use tool to create HTML Help files and Help web sites](#)

OnDiskFull

TVideoGrabber.OnDiskFull

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when the disk becomes full during recording.

Declaration

property OnDiskFull: TEventNotification **read** FOnDiskFull **write** FOnDiskFull;

___property TEventNotification OnDiskFull=read=FOnDiskFull, write=FOnDiskFull

Event OnDiskFull()

Description

Occurs when the disk where is located the [StoragePath](#) becomes full during recording, which is stopped. This event occurs also when the space remaining on disk is lower than 25 Mb.

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last](#) [Recording](#) [FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [Streamline your documentation process with HelpNDoc's WinHelp HLP to CHM conversion feature](#)

OnDragDropFiles

TVideoGrabber.OnDragDropFiles

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when dropping file(s)

Declaration

property OnDragDropFiles: TOnDragDropFiles **read** FOnDragDropFiles **write** FOnDragDropFiles;

___**property** TOnDragDropFiles OnDragDropFiles=read=FOnDragDropFiles, write=FOnDragDropFiles

Event OnDragDropFiles

Description

Occurs when dropping file(s) over the TVideoGrabber control (for Delphi / C++Builder)

Type [TOnDragDropFiles](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly optimize your documentation website for search engines](#)

OnDVCommandCompleted

TVideoGrabber.OnDVCommandCompleted

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when a DV VCR command is completed.

Declaration

property OnDVCommandCompleted: TOnDVCommandCompleted **read** FOnDVCommandCompleted **write**

FOnDVCommandCompleted;

__property TOnDVCommandCompleted OnDVCommandCompleted=read=FOnDVCommandCompleted, write=FOnDVCommandCompleted

Event OnDVCommandCompleted(NewStateValue As Long, NewStateLabel As String)

Description

This [TOnDVCommandCompleted](#) event occurs:

- occurs also when invoking [StartPreview](#) or [StartRecording](#), to retrieve the current state of the DV device.
- occurs after invoking [SendDVCommand](#), when a DV VCR command is completed.
- occurs when the DV VCR state changes (e.g. when rewind ends).

See Also

[TDVSize](#) [TDVVideoFormat](#) [TDVVideoStandard](#) [TOnDVCommandCompleted](#) [DVDDateTimeEnabled](#) [DVDDiscontinuityMinimumInterval](#) [DVRecordingInNativeFormatSeparatesStreams](#) [DVReduceFrameRate](#) [DVRgb219](#) [DVTimeCodeEnabled](#) [IsDigitalVideoIn](#) [IsDVCommandAvailable](#) [IsTimeCodeReaderAvailable](#) [SendDVCommand](#)

Created with the Standard Edition of HelpNDoc: [Free Kindle producer](#)

OnDVDDiscontinuity

TVideoGrabber.OnDVDDiscontinuity

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when a DV a discontinuity occurs

Declaration

property OnDVDDiscontinuity: TOnDVDDiscontinuity **read** FOnDVDDiscontinuity **write** FOnDVDDiscontinuity;

__property TOnDVDDiscontinuity OnDVDDiscontinuity=read=FOnDVDDiscontinuity, write=FOnDVDDiscontinuity

Event OnDVDDiscontinuity(DeliverNewFrame as Boolean)

Description

Occurs when a time discontinuity is detected in the DV date/time, if available.

From this event you can invoke e.g. [StopRecording](#) to stop the current recording, or [RecordToNewFileNow](#) to start a new recording when a time discontinuity is detected in the DV date/time data.

Set the **DeliverNewFrame** parameter to "false" to prevent the current video frame (that corresponds to the 1st frame of the new time) to be saved in the file generated when closing the current recording.

Important: from this event you should not perform any actions with the potential to block, such as holding a critical section or waiting on another thread. Also, do not call any GDI or USER32.DLL APIs.

Created with the Standard Edition of HelpNDoc: [Easily create Web Help sites](#)

OnEnumerateWindows

TVideoGrabber.OnEnumerateWindows

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Return information about the visible windows.

Declaration

property OnEnumerateWindows: T OnEnumerateWindows **read** FOnEnumerateWindows **write** FOnEnumerateWindows;

__property TOnEnumerateWindows OnEnumerateWindows==FOnEnumerateWindows, write=FOnEnumerateWindows;

Event OnEnumerateWindow (WindowHandle as Long, WindowName as String, WindowClass as string)

Description

Can be used in the screen recording of a window, to retrieve the name, class and handle of the visible windows.

This lets you retrieve the identifier of a window to perform the screen recording on this window.

See the "[Recording a window](#)" chapter for more information.

Created with the Standard Edition of HelpNDoc: [Free PDF documentation generator](#)

OnFilterSelected

TVideoGrabber.OnFilterSelected

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when a filter is selected.

Declaration

property OnFilterSelected: TOnFilterSelected **read** FOnFilterSelected **write** FOnFilterSelected;

__property TOnFilterSelected OnFilterSelected==FOnFilterSelected, write=FOnFilterSelected;

Event OnFilterselected (FilterName as String, ByRef RejectFilter as Boolean)

Description

This [TOnFilterSelected](#) event occurs when filters are selected automatically when constructing the graph. The FilterName parameter lets you know what is the filter selected. The RejectFilter parameter lets you reject the filter.

Created with the Standard Edition of HelpNDoc: [Free help authoring environment](#)

OnFrameBitmap

TVideoGrabber.OnFrameBitmap

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Gives access to each frame bitmap

Declaration

property OnFrameBitmap: TOnFrameBitmap **read** FOnFrameBitmap **write** FOnFrameBitmap;

__property TOnFrameBitmap OnFrameBitmap=read=FOnFrameBitmap, write=FOnFrameBitmap

Event OnFrameBitmap(ByVal FrameInfo As Long, ByVal BitmapInfo As Long)

Description

This event occurs for each frame bitmap when the [FrameGrabber](#) is enabled.

To preserve the performance while offering a lot of useful values, this event returns 2 parameters that are

pointers to structures:

A) the **FrameInfo** parameter, that is a pointer to a [TFrameInfo](#) structure.
It returns time and frame number information about the current video frames

B) the **BitmapInfo** parameter, which is a pointer to a [TFrameBitmapInfo](#) structure.
It returns all the useful bitmap information, and also the current mouse information (current position, last clicked position, button states, mouse down, etc...)

From this event you can perform powerful bitmap tasks, like line drawing, mouse-based drawing, pixel modifications, etc...

You will find a lot of sample code in the OnFrameBitmap event of the MainDemo project (search "FrameBitmap" in the MainForm)

To activate this sample code, run the MainDemo project -> the checkboxes are located in the "Overlay" tab.

NOTE:

1. from this event you should not perform any actions with the potential to block:

1. do not hold a critical section or wait on another thread,
2. do not call any GDI or USER32.DLL APIs that might cause a window to move,
3. do not invoke component properties that send messages, e.g. do not read the ItemIndex property of a ListBox component.

If you need to read a such property, set an intermediary variable when the property changes, and then read the intermediary variable from the OnFrameOverlay... event.

2. if you need to perform blocking operations from this event, enable the [OnFrameBitmapEventSynchron](#) property.

Enabling the OnFrameBitmapEventSynchron property may slow down the application and/or introduce latency in the video stream.

See Also

[TCardinalDirection](#) [TAutoFileName](#) [TFrameCaptureDest](#) [TFrameGrabberRGBFormat](#)
[TOnFrameCaptureCompleted](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#)
[TVideoDeinterlacing](#) [AdjustOverlayAspectRatio](#) [BurstCount](#) [BurstInterval](#) [BurstMode](#) [BurstType](#)
[CaptureFrameSyncTo](#) [CaptureFrameTo](#) [DrawBitmapOverFrame](#) [FrameCaptureHeight](#) [FrameCaptureWidth](#)
[FrameCaptureWithoutOverlay](#) [FrameCaptureZoomSize](#) [FrameGrabber](#) [FrameGrabberCurrentRGBFormat](#)
[FrameGrabberRGBFormat](#) [FramerateDivider](#) [GetFrameInfo](#) [GetFrameInfoString](#) [GetLastFrameAsHBITMAP](#)
[GetLastFrameAsTBitmap](#) [GetLastFrameBitmapBits](#) [GetLastFrameBitmapBits2](#) [GetLastFrameWaitTimeoutMs](#)
[ImageOverlay](#) [StretchToVideoSize](#) [InFrameProgressEvent](#) [JPEGPerformance](#) [JPEGProgressiveDisplay](#)
[JPEGQuality](#) [LastBurstFrameCapture](#) [FileName](#) [LastCaptureFrameTo](#) [FileName](#)
[MouseWheelEventEnabled](#) [MP4NeedsReindexing](#) [OnDiskFull](#) [OnFrameBitmapEventSynchron](#)
[OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#)
[OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#)
[RetrievalInitialXYAfterRotation](#) [SetFrameCaptureBounds](#) [SetImageOverlay](#) [AlphaBlend](#)
[SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#)
[SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#)
[SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#)
[SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#)
[SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#)
[SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#)
[SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#)
[SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#)
[SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#)
[SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#)
[SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#)
[ShapeOverlayEnabled](#) [ShapeOverlayList](#) [StoragePath](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#)
[TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#)
[TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#)

[TextOverlay](#) [ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#) [TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#) [TranslateMouseCoordinates](#) [VideoProcessing_Brightness](#) [VideoProcessing_Contrast](#) [VideoProcessing_Deinterlacing](#) [VideoProcessing_FlipHorizontal](#) [VideoProcessing_FlipVertical](#) [VideoProcessing_GrayScale](#) [VideoProcessing_Hue](#) [VideoProcessing_InvertColors](#) [VideoProcessing_Pixellization](#) [VideoProcessing_Rotation](#) [VideoProcessing_RotationCustomAngle](#) [VideoProcessing_Saturation](#) [WebcamStillCaptureButton](#) [TVideoRotation](#)

Created with the Standard Edition of HelpNDoc: Revolutionize Your Documentation Output with a Help Authoring Tool

OnFrameCaptureCompleted

TVideoGrabber.OnFrameCaptureCompleted

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Returns each video frame captured.

Declaration

property OnFrameCaptureCompleted: TOnFrameCaptureCompleted **read** FOnFrameCaptureCompleted **write** FOnFrameCaptureCompleted;

__property TOnFrameCaptureCompleted

OnFrameCaptureCompleted=read=FOnFrameCaptureCompleted, write=FOnFrameCaptureCompleted

Event OnFrameCaptureCompleted(FrameBitmapHandle As Long, BitmapWidth As Long, BitmapHeight As Long, FrameNumber As Long, FrameTime As Double, DestType As TxFrameCaptureDest, FileName As String, Success As Boolean, FrameId As Long)

Description

This [TOnFrameCaptureCompleted](#) event returns each video frame captured by using [CaptureFrameTo](#) or the [burst mode](#) .

For Delphi and C++Buidler versions, the video frame is returned as TBitmap (from the "graphics" unit). For OCX versions the video frame is returned as a Bitmap handle.

For frame capture to BMP or JPEG files, the file name generated automatically is also returned by this event.

You can get various frame infos from this event by using [GetFrameInfo](#) (FrameId, ...).

Note: the [frame grabber](#) must be enabled to capture video frames.

Sample code is included in the MainDemo project.

To get information about the current video frame, invoke [GetFrameInfo](#) (FrameId, ...). E.g.

```
SampleTimeHour = VideoGrabber.GetFrameInfo (FrameId, fi_SampleTime_Hour)
```

See Also

[TAutoFileName](#) [TFrameCaptureDest](#) [TOnFrameCaptureCompleted](#) [BurstCount](#) [BurstInterval](#) [BurstMode](#) [BurstType](#) [CaptureFrameSyncTo](#) [CaptureFrameTo](#) [FrameCaptureHeight](#) [FrameCaptureWidth](#) [FrameCaptureWithoutOverlay](#) [FrameCaptureZoomSize](#) [GetFrameInfoString](#) [GetLastFrameAsHBITMAP](#) [GetLastFrameAsTBitmap](#) [GetLastFrameBitmapBits](#) [GetLastFrameBitmapBits2](#) [GetLastFrameWaitTimeoutMs](#) [JPEGPerformance](#) [JPEGProgressiveDisplay](#) [JPEGQuality](#) [Last_BurstFrameCapture](#) [FileName](#) [Last_CaptureFrameTo](#) [FileName](#) [OnDiskFull](#) [OnFrameBitmap](#) [SetFrameCaptureBounds](#) [StoragePath](#) [WebcamStillCaptureButton](#)

OnFrameOverlayUsingDC

TVideoGrabber.OnFrameOverlayUsingDC

[Prev](#)[Next](#)[TVideoGrabber](#) [Events](#)

Put your code in this event to draw graphic objects over video frames by using the frame DC.

Declaration

property OnFrameOverlayUsingDC: TOnFrameOverlayUsingDC **read** FOnFrameOverlayUsingDC **write** SetOnFrameOverlayUsingDC;

__property TOnFrameOverlayUsingDC OnFrameOverlayUsingDC=read=FOnFrameOverlayUsingDC, write=SetOnFrameOverlayUsingDC

Event OnFrameOverlayUsingDC(Dc As Long, FrameNumber As Long, FrameTime As Double, FrameId As Long)

Description

DEPRECATED, use [OnFrameBitmap](#) instead

This [TOnFrameOverlayUsingDC](#) event is used to draw graphic objects over video frames, by using the frame device context.

This event occurs for each video frame that goes through the video stream, during preview, recording or playback.

The [frame grabber](#) must be enabled for this event to occur.

From this event it is possible to:

- draw a bitmap over frames by calling DrawBitmapOverFrame,
- draw anything using the Dc parameter (device context) and GDI functions.

This event is TIME CRITICAL. The code used in this event must be as short and fast as possible.

Sample code is included in the MainDemo project.

To get information about the current video frame, invoke [GetFrameInfo](#) (FrameId, ...). E.g.

```
SampleTimeHour = VideoGrabber.GetFrameInfo (FrameId, fi_SampleTime_Hour)
```

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#)
[AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#)
[MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDIB](#)
[OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#)
[OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#)
[SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#)
[SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#)
[SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#)
[SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#)
[SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#)
[SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#)

[SetTextOverlay_Font](#) [SetTextOverlay_FontColor](#) [SetTextOverlay_GradientColor](#)
[SetTextOverlay_GradientMode](#) [SetTextOverlay_HighResFont](#) [SetTextOverlay_Left](#) [SetTextOverlay_Right](#)
[SetTextOverlay_Scrolling](#) [SetTextOverlay_ScrollingSpeed](#) [SetTextOverlay_Shadow](#)
[SetTextOverlay_ShadowColor](#) [SetTextOverlay_ShadowDirection](#) [SetTextOverlay_String](#)
[SetTextOverlay_TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#)
[ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay_Align](#) [TextOverlay_BkColor](#)
[TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#)
[TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#)
[TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#)
[TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#)
[TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Productivity with HelpNDoc's Efficient User Interface](#)

OnFrameOverlayUsingDIB

TVideoGrabber.OnFrameOverlayUsingDIB

[Prev](#)
[Next](#)

[TVideoGrabber](#)
[Events](#)

Put your code in this event to apply any processing to each video frame, through its DIB bitmap.

Declaration

property OnFrameOverlayUsingDIB: TOnFrameOverlayUsingDIB **read** FOnFrameOverlayUsingDIB **write** FOnFrameOverlayUsingDIB;

__property TOnFrameOverlayUsingDIB OnFrameOverlayUsingDIB=read=FOnFrameOverlayUsingDIB, write=FOnFrameOverlayUsingDIB

Event OnFrameOverlayUsingDIB(FrameBitmapHandle As Long, FrameNumber As Long, FrameTime As Double, FrameId As Long)

Description

DEPRECATED, use [OnFrameBitmap](#) instead

This TOnFrameOverlayUsingDIB event is used to process each video frame DIB bitmap.

This event occurs for each video frame that goes through the video stream, during preview, recording or playback.

The [frame grabber](#) must be enabled for this event to occur.

This event offers a full access to the DIB bitmap of each video frame.

This event is TIME CRITICAL. The code used in this event must be as short and fast as possible.

Sample code is included in the MainDemo project.

To get information about the current video frame, invoke [GetFrameInfo](#) (FrameId, ...). E.g.

```
SampleTimeHour = VideoGrabber.GetFrameInfo (FrameId, fi_SampleTime_Hour)
```

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#)
[AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#)
[MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronone](#) [OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#)
[OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#)
[SetImageOverlay_AlphaBlendValue](#) [SetImageOverlay_ChromaKey](#)

[SetImageOverlay_ChromaKeyLeewayPercent](#)
[SetImageOverlay_ChromaKeyRGBColor](#)
[SetImageOverlay_Enabled](#)
[SetImageOverlay_Height](#)
[SetImageOverlay_LeftLocation](#)
[SetImageOverlay_RotationAngle](#)
[SetImageOverlay_StretchToVideoSize](#)
[SetImageOverlay_TargetDisplay](#)
[SetImageOverlay_TopLocation](#)
[SetImageOverlay_Transparent](#)
[SetImageOverlay_TransparentColorValue](#)
[SetImageOverlay_UseTransparentColor](#)
[SetImageOverlay_Width](#)
[SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#)
[SetImageOverlayFromHBitmap2](#)
[SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#)
[SetImageOverlayFromJPEGFile](#)
[SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#)
[SetImageOverlayFromTImage](#)
[SetImageOverlayFromTImage2](#)
[SetTextOverlay_Align](#)
[SetTextOverlay_BkColor](#)
[SetTextOverlay_CustomVar](#)
[SetTextOverlay_Enabled](#)
[SetTextOverlay_Font](#)
[SetTextOverlay_FontColor](#)
[SetTextOverlay_GradientColor](#)
[SetTextOverlay_GradientMode](#)
[SetTextOverlay_HighResFont](#)
[SetTextOverlay_Left](#)
[SetTextOverlay_Right](#)
[SetTextOverlay_Scrolling](#)
[SetTextOverlay_ScrollingSpeed](#)
[SetTextOverlay_Shadow](#)
[SetTextOverlay_ShadowColor](#)
[SetTextOverlay_ShadowDirection](#)
[SetTextOverlay_String](#)
[SetTextOverlay_TargetDisplay](#)
[SetTextOverlay_Top](#)
[SetTextOverlay_Transparent](#)
[ShapeOverlay](#)
[ShapeOverlayEnabled](#)
[ShapeOverlayList](#)
[TextOverlay_Align](#)
[TextOverlay_BkColor](#)
[TextOverlay_CreateCustomFont](#)
[TextOverlay_CreateCustomFont2](#)
[TextOverlay_Enabled](#)
[TextOverlay_Font](#)
[TextOverlay_FontColor](#)
[TextOverlay_Left](#)
[TextOverlay_Right](#)
[TextOverlay_Scrolling](#)
[TextOverlay_ScrollingSpeed](#)
[TextOverlay_Selector](#)
[TextOverlay_Shadow](#)
[TextOverlay_ShadowColor](#)
[TextOverlay_ShadowDirection](#)
[TextOverlay_String](#)
[TextOverlay_Top](#)
[TextOverlay_Transparent](#)
[TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Effortlessly optimize your documentation website for search engines

OnFrameOverlayUsingVIDEOHDR

TVideoGrabber.OnFrameOverlayUsingVIDEOHDR

[Prev](#)
[Next](#)

[TVideoGrabber](#)
[Events](#)

Put your code in this event to apply any processing to each video frame, through a VFW VIDEOHDR format.

Declaration

property OnFrameOverlayUsingVIDEOHDR: TOnFrameOverlayUsingVIDEOHDR **read** FOnFrameOverlayUsingVIDEOHDR **write** FOnFrameOverlayUsingVIDEOHDR;

```
__property TOnFrameOverlayUsingVIDEOHDR
OnFrameOverlayUsingVIDEOHDR=read=FOnFrameOverlayUsingVIDEOHDR,
write=FOnFrameOverlayUsingVIDEOHDR
```

n/a

Description

DEPRECATED, use [OnFrameBitmap](#) instead

This TOnFrameOverlayUsingVIDEOHDR event is used to draw graphic objects over video frames, by using a VIDEOHDR video frame format.

This event occurs for each video frame that goes through the video stream, during preview, recording or playback.

The [frame grabber](#) must be enabled for this event to occur.

Sample code is included in the MainDemo project.

To get information about the current video frame, invoke [GetFrameInfo](#) (FrameId, ...). E.g.

```
SampleTimeHour = VideoGrabber.GetFrameInfo (FrameId, fi_SampleTime_Hour)
```

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#) [SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#) [SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#) [SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#) [SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#) [TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#) [TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#) [TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#) [TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: Save time and frustration with HelpNDoc's WinHelp HLP to CHM conversion feature

OnFrameProgress

TVideoGrabber.OnFrameProgress

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Returns information about each video frame.

Declaration

property OnFrameProgress: TOnFrameProgress **read** FOnFrameProgress **write** FOnFrameProgress;

__property TOnFrameProgress OnFrameProgress=read=FOnFrameProgress, write=FOnFrameProgress

Event OnFrameProgress(FrameNumber As Long, FrameTime As Double, FrameId As Long)

Description

DEPRECATED, use [OnFrameProgress2](#) instead

This [TOnFrameProgress](#) event occurs asynchronously for each video frame.

To get information about the current video frame within this event, invoke [GetFrameInfo](#) (FrameId, ...).
E.g.

```
SampleTimeHour = VideoGrabber.GetFrameInfo (FrameId, fi_SampleTime_Hour)
```

Look at the *VideoGrabberFrameProgress* event in the *MainDemo* project for sample code.

See Also

[TOnFrameProgress](#) [TOnResizeVideo](#) [AutoRefreshPreview](#) [BackgroundColor](#) [Display](#) [AutoSize](#)

[Display_FullScreen](#) [Display_SetLocation](#) [Display_VideoPortEnabled](#) [ImageRatio](#) [InFrameProgressEvent](#) [IsVideoInterlaced](#) [IsVideoPortAvailable](#) [OnKeyPress](#) [OnPreviewStarted](#) [OnResizeVideo](#) [PausePreview](#) [PreviewZoomSize](#) [ResumePreview](#) [StartPreview](#) [StartSynchronized](#) [StopPreview](#) [Synchronized](#) [UseNearestVideoSize](#) [VideoHeight](#) [VideoRenderer](#) [VideoRendererPriority](#) [VideoWidth](#) [Visible](#)

Created with the Standard Edition of HelpNDoc: [Free help authoring tool](#)

OnFrameProgress2

TVideoGrabber.OnFrameProgress2

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs for each video frame

Declaration

property OnFrameProgress2: TOnFrameProgress2 **read** FOnFrameProgress2 **write** FOnFrameProgress2;

__property TOnFrameProgress2 OnFrameProgress2 =read=FOnFrameProgress2, write=FOnFrameProgress2

Event OnFrameProgress2 (ByVal FrameInfo As Long)

Description

This event occurs asynchronously for each video frame.

It returns the current frame time, frame number, DV frame time, timecode, etc... through the FrameInfo pointer to the [TFrameInfo](#) structure.

You will find sample code in the MainForm of the MainDemo project (Search for "FrameProgress2")

Created with the Standard Edition of HelpNDoc: [Maximize Your CHM Help File Capabilities with HelpNDoc](#)

OnGraphBuilt

TVideoGrabber.OnGraphBuilt

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when the video graph is built.

Declaration

property OnGraphBuilt: TNotifyEvent **read** FOnGraphBuilt **write** FOnGraphBuilt;

__property Classes::TNotifyEvent OnGraphBuilt=read=FOnGraphBuilt, write=FOnGraphBuilt

Event OnGraphBuilt()

Description

Occurs when a preview, recording or playback video graph is built.

Used mainly to retrieve properties that are available only after the video graph is built, like [GetVMR9ImageAdjustmentBounds](#).

E.g.:

```
procedure TForm1.VideoGrabberGraphBuilt(Sender: TObject);
var
    MinValue, MaxValue, StepSize, DefaultValue, CurrentValue: LongInt;
```

```

begin
    BrightnessTrackbar.Enabled := VideoGrabber.GetVMR9ImageAdjustmentBounds (True, vmr9_Brightness);
    if Brightness.Enabled then begin
        Brightness.Min := MinValue;
        Brightness.Max := MaxValue;
        Brightness.Frequency := StepSize;
        Brightness.Position := CurrentValue;
    end;
end;

procedure TForm1.BrightnessTrackbarChange(Sender: TObject);
begin
    VideoGrabber.SetVMR9ImageAdjustmentValue (True, vmr9_Brightness, tbrVMR9Brightness.Position);
end;

```

See the MainDemo project for sample code.

Created with the Standard Edition of HelpNDoc: [Free Web Help generator](#)

OnInactive

TVideoGrabber.OnInactive

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when the current task ends.

Declaration

property OnInactive: TEventNotification **read** FOnInactive **write** FOnInactive;

__property TEventNotification OnInactive=read=FOnInactive, write=FOnInactive

Event OnInactive()

Description

Occurs when the current task ends, after invoking [StopPreview](#), [StopRecording](#), [ClosePlayer](#), etc...

Created with the Standard Edition of HelpNDoc: [Converting Word Docs to eBooks Made Easy with HelpNDoc](#)

OnKeyPress

TVideoGrabber.OnKeyPress

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when keys are pressed on a video window that has the control.

Declaration

property OnKeyPress: TOnVideoKeyPress **read** FOnKeyPress **write** FOnKeyPress;

__property TOnVideoKeyPress OnKeyPress=read=FOnKeyPress, write=FOnKeyPress

Event OnKeyPress(VideoWindow As Long, Key, PhysicalKey As Long)

Description

This [TOnVideoKeyPress](#) event occurs when keys are pressed on a video window that has the control. Useful mainly when the video window is not attached to the control ([Display embedded](#) disabled), to receive

key events from the window.

See Also

[TOnFrameProgress](#) [TOnResizeVideo](#) [AutoRefreshPreview](#) [BackgroundColor](#) [Display](#) [AutoSize](#) [Display_FullScreen](#) [Display_SetLocation](#) [Display_VideoPortEnabled](#) [ImageRatio](#) [InFrameProgressEvent](#) [IsVideoInterlaced](#) [IsVideoPortAvailable](#) [OnFrameProgress](#) [OnPreviewStarted](#) [OnResizeVideo](#) [PausePreview](#) [PreviewZoomSize](#) [ResumePreview](#) [StartPreview](#) [StartSynchronized](#) [StopPreview](#) [Synchronized](#) [UseNearestVideoSize](#) [VideoHeight](#) [VideoRenderer](#) [VideoRendererPriority](#) [VideoWidth](#) [Visible](#)

Created with the Standard Edition of HelpNDoc: [Create help files for the Qt Help Framework](#)

OnLastCommandCompleted

TVideoGrabber.OnLastCommandCompleted

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when a command has been completed

Declaration

property OnLastCommandCompleted: TOnEventNotification **read** FOnLastCommandCompleted **write** FOnLastCommandCompleted;

__property TOnEventNotification OnLastCommandCompleted=read=FOnLastCommandCompleted, write=FOnLastCommandCompleted

Event OnLastCommandCompleted

Description

Occurs when a command has been completed (e.g. StartPreview, OpenPlayer, StopRecording, etc...)

Created with the Standard Edition of HelpNDoc: [Write EPub books for the iPad](#)

OnLeavingFullScreen

TVideoGrabber.OnLeavingFullScreen

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when leaving the full screen mode

Declaration

property OnLeavingFullScreen: TOnEventNotification **read** FOnLeavingFullScreen **write** FOnLeavingFullScreen;

__property TOnEventNotification OnLeavingFullScreen==FOnLeavingFullScreen, write=FOnLeavingFullScreen;

Event OnLeavingFullScreen()

Description

This event occurs when leaving the full screen mode, e.g. when the user press the "ESC" key

See Also

[TVideoRenderer](#) [AdjustPixelAspectRatio](#) [Display_Active](#) [Display_AlphaBlendEnabled](#) [Display_AlphaBlendValue](#) [Display_AutoSize](#) [Display_Embedded](#) [Display_FullScreen](#) [Display_Height](#) [Display_Left](#) [Display_Monitor](#) [Display_MouseMovesWindow](#) [Display_PanScanRatio](#) [Display_SetLocation](#) [Display_StayOnTop](#) [Display_Top](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [Display_VideoHeight](#) [Display_VideoPortEnabled](#) [Display_VideoWidth](#) [Display_VideoWindowHandle](#) [Display_Width](#) [DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#) [DualDisplay_AlphaBlendValue](#) [DualDisplay_AutoSize](#) [DualDisplay_Embedded](#) [DualDisplay_FullScreen](#) [DualDisplay_Height](#)

[DualDisplay_Left](#) [DualDisplay_Monitor](#) [DualDisplay_MouseMovesWindow](#) [DualDisplay_PanScanRatio](#) [DualDisplay_StayOnTop](#) [DualDisplay_Top](#) [DualDisplay_TransparentColorEnabled](#) [DualDisplay_TransparentColorValue](#) [DualDisplay_VideoHeight](#) [DualDisplay_VideoPortEnabled](#) [DualDisplay_VideoWidth](#) [DualDisplay_VideoWindowHandle](#) [DualDisplay_Visible](#) [DualDisplay_Width](#) [IsVideoPortAvailable](#) [Monitor_Primary](#) [Index](#) [MonitorBounds](#) [MonitorsCount](#) [SetParentWindow](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your PDF Protection with These Simple Steps](#)

OnLog

TVideoGrabber.OnLog

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Returns information, warning and error messages

Declaration

property OnLog: TOnLog **read** FOnLog **write** FOnLog;

__property TOnLog OnLog=read=FOnLog, write=FOnLog

Event OnLog(LogType As TxLogType, Severity As String, InfoMsg As String)

Description

Warnings and error messages are returned by this event.

See also [TOnLog](#) and [GetLogString](#)

Created with the Standard Edition of HelpNDoc: [Eliminate the Struggles of Documentation with a Help Authoring Tool](#)

OnMotionDetected

TVideoGrabber.OnMotionDetected

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when a new frame is received and motion is detected.

Declaration

property OnMotionDetected: TOnMotionDetected **read** FOnMotionDetected **write** FOnMotionDetected;

__property TOnMotionDetected OnMotionDetected=read=FOnMotionDetected, write=FOnMotionDetected

Event OnMotionDetected(GlobalMotionRatio As Double, MaxMotionCellX As Long, MaxMotionCellY As Long, FrameBitmapHandle As Long, FrameNumber As Long, FrameTime As Double, FrameId As Long, CaptureFrame)

Description

This [TOnMotionDetected](#) event occurs when a new frame is received and motion is detected.

Note: be sure to activate the motion detector by enabling [MotionDetector_Enabled](#).

See Also

[Color / Greyscale Grid structure / grid sensitivity](#) [Motion ratio](#) [Recording only when motion is detected](#) [Video noise](#) [TOnMotionDetected](#) [TOnMotionNotDetected](#) [MotionDetector_CellMotionRatio](#) [MotionDetector_CompareBlue](#) [MotionDetector_CompareGreen](#) [MotionDetector_CompareRed](#) [MotionDetector_Enabled](#) [MotionDetector_EnumGridDialogControls](#) [MotionDetector_Get2DTextGrid](#) [MotionDetector_Get2DTextMotion](#) [MotionDetector_GetCellLocation](#) [MotionDetector_GetCellSensitivity](#)

[MotionDetector_GetCellSize](#)
[MotionDetector_GloballyIncOrDecSensitivity](#)
[MotionDetector_GlobalMotionRatio](#)
[MotionDetector_GreyScale](#)
[MotionDetector_Grid](#)
[MotionDetector_GridXCount](#)
[MotionDetector_GridYCount](#)
[MotionDetector_IsGridValid](#)
[MotionDetector_MaxDetectionsPerSecond](#)
[MotionDetector_ReduceCPULoad](#)
[MotionDetector_ReduceVideoNoise](#)
[MotionDetector_Reset](#)
[MotionDetector_ResetGlobalSensitivity](#)
[MotionDetector_SetCellSensitivity](#)
[MotionDetector_SetGridSize](#)
[MotionDetector_ShowGridDialog](#)
[MotionDetector_Triggered](#)
[MotionDetector_UseThisReferenceSample](#)
[OnBacktimedFramesCountReached](#)
[OnMotionNotDetected](#)
[RecordingOnMotion_Enabled](#)
[RecordingOnMotion_MotionThreshold](#)
[RecordingOnMotion_NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: [Upgrade Your Documentation Process with a Help Authoring Tool](#)

OnMotionNotDetected

TVideoGrabber.OnMotionNotDetected

[Prev](#)
[Next](#)

[TVideoGrabber](#)
[Events](#)

Occurs when a frame is received and no motion is detected.

Declaration

property OnMotionNotDetected: TOnMotionNotDetected **read** FOnMotionNotDetected **write** FOnMotionNotDetected;

__property TOnMotionNotDetected OnMotionNotDetected=read=FOnMotionNotDetected, write=FOnMotionNotDetected

Event OnMotionNotDetected(FrameBitmapHandle As Long, FrameNumber As Long, FrameTime As Double, FrameId As Long, CaptureFrame)

Description

This [TOnMotionNotDetected](#) event occurs when a frame is received and no motion is detected.

Note: be sure to activate the motion detector by enabling [MotionDetector_Enabled](#).

See Also

[Color / Greyscale Grid structure / grid sensitivity](#)
[Motion ratio](#)
[Recording only when motion is detected](#)
[Video noise](#)
[TOnMotionDetected](#)
[TOnMotionNotDetected](#)
[MotionDetector_CellMotionRatio](#)
[MotionDetector_CompareBlue](#)
[MotionDetector_CompareGreen](#)
[MotionDetector_CompareRed](#)
[MotionDetector_Enabled](#)
[MotionDetector_EnumGridDialogControls](#)
[MotionDetector_Get2DTextGrid](#)
[MotionDetector_Get2DTextMotion](#)
[MotionDetector_GetCellLocation](#)
[MotionDetector_GetCellSensitivity](#)
[MotionDetector_GetCellSize](#)
[MotionDetector_GloballyIncOrDecSensitivity](#)
[MotionDetector_GlobalMotionRatio](#)
[MotionDetector_GreyScale](#)
[MotionDetector_Grid](#)
[MotionDetector_GridXCount](#)
[MotionDetector_GridYCount](#)
[MotionDetector_IsGridValid](#)
[MotionDetector_MaxDetectionsPerSecond](#)
[MotionDetector_ReduceCPULoad](#)
[MotionDetector_ReduceVideoNoise](#)
[MotionDetector_Reset](#)
[MotionDetector_ResetGlobalSensitivity](#)
[MotionDetector_SetCellSensitivity](#)
[MotionDetector_SetGridSize](#)
[MotionDetector_ShowGridDialog](#)
[MotionDetector_Triggered](#)
[MotionDetector_UseThisReferenceSample](#)
[OnBacktimedFramesCountReached](#)
[OnMotionDetected](#)
[RecordingOnMotion_Enabled](#)
[RecordingOnMotion_MotionThreshold](#)
[RecordingOnMotion_NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: [Make Documentation a Breeze with HelpNDoc's Clean and Efficient User Interface](#)

OnMouseDown

TVideoGrabber.OnMouseDown

[Prev](#)
[Next](#)

[TVideoGrabber](#)
[Events](#)

Occurs when the user presses a mouse button with the mouse pointer over a control.

Declaration

property OnMouseDown: TOnVideoMouseUpDown **read** FOnMouseDown **write** FOnMouseDown;

__property TOnVideoMouseUpDown OnMouseDown=read=FOnMouseDown, write=FOnMouseDown

Event OnMouseDown(VideoWindow As Long, Button As TMouseButton, x As Long, y As Long)

Description

- if [TranslateMouseCoordinates](#) is disabled, the normal OnMouseUp event occurs and returns the real mouse coordinates across the control.
- if [TranslateMouseCoordinates](#) is enabled, this event returns the coordinates corresponding to the native video size.

The difference appears when [Display_AutoSize](#) is disabled or [PreviewZoomSize](#) <> 100. In this case, the event reports the corresponding native video source coordinate, necessary to draw objects over video frames from the [OnFrameBitmap](#) event.

Note that the [BitmapInfo](#) parameter of the [OnFrameBitmap](#) event returns directly the current mouse information.

Note: to detect if the video is displayed when the event occurs, test VideoWindow <> -1.

E.g.:

```
procedure TfrmMainForm.VideoGrabberMouseDown(Sender: TObject;
  VideoWindow: Integer; Button: TMouseButton; Shift: TShiftState; X,
  Y: Integer);
begin
  if VideoWindow <> -1 then begin // if the video is displayed
    if chkFreeHandEnabled.Checked then begin
      ...
      ...
    end;
  end;
end;
```

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#)
[AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#)
[MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#)
[OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#)
[SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#)
[SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#)
[SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#)
[SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#)
[SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#)
[SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#)
[SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#)
[SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#)
[SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#)
[SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#)
[SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#)
[ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#)
[TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#)
[TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#)

[TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#) [TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [What is a Help Authoring tool?](#)

OnMouseDown_Video

TVideoGrabber.OnMouseDown_Video

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when the user presses a mouse button with the mouse pointer over a control.

Declaration

property OnMouseDown_Video: TOnVideoMouseUpDown **read** FOnMouseDown **write** FOnMouseDown;

__property TOnVideoMouseUpDown OnMouseDown_Video=read=FOnMouseDown, write=FOnMouseDown

Event OnMouseDown_Video(VideoWindow As Long, Button As TxMouseButton, x As Long, y As Long)

Description

returns the coordinates of the original video frame at the cursor location

Created with the Standard Edition of HelpNDoc: [Streamline Your Documentation Process with a Help Authoring Tool](#)

OnMouseDown_Window

TVideoGrabber.OnMouseDown_Window

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when the user presses a mouse button with the mouse pointer over a control.

Declaration

property OnMouseDown_Window: TOnVideoMouseUpDown **read** FOnMouseDown **write** FOnMouseDown;

__property TOnVideoMouseUpDown OnMouseDown_Window=read=FOnMouseDown, write=FOnMouseDown

Event OnMouseDown_Window(VideoWindow As Long, Button As TxMouseButton, x As Long, y As Long)

Description

returns the coordinates of the video window at the cursor location

Created with the Standard Edition of HelpNDoc: [Streamline Your CHM Help File Creation with HelpNDoc](#)

OnMouseMove

TVideoGrabber.OnMouseMove

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when the user moves the mouse pointer while the mouse pointer is over a control.

Declaration

property OnMouseMove: TOnVideoMouseMove **read** FOnMouseMove **write** FOnMouseMove;

___property TOnVideoMouseMove OnMouseMove=read=FOnMouseMove, write=FOnMouseMove

Event OnMouseMove(VideoWindow As Long, x As Long, y As Long)

Description

- if [TranslateMouseCoordinates](#) is enabled:

this event returns the video frame coordinates corresponding to the native video size. **Therefore the event does not occur if the mouse is moved out of the video window.**

- if [TranslateMouseCoordinates](#) is disabled:

the normal OnMouseUp event occurs and returns the usual mouse coordinates **over the video window, as well as out of the video window** if the mouse has been clicked within the video window and then moved out of it.

The difference appears when [Display_AutoSize](#) is disabled or [PreviewZoomSize](#) <> 100. In this case, the event reports the corresponding native video source coordinate, necessary to draw objects over video frames from the [OnFrameBitmap](#) event.

Note that the BitmapInfo parameter of the [OnFrameBitmap](#) event returns directly the current mouse information.

Note: to detect if the video is displayed when the event occurs, test VideoWindow <> -1.

E.g.:

```
procedure TfrmMainForm.VideoGrabberMouseMove(Sender: TObject;
  VideoWindow: Integer; Shift: TShiftState; X, Y: Integer);
begin
  if VideoWindow <> -1 then begin // if the video is displayed
    if chkFreeHandEnabled.Checked then begin
      ...
      ...
    end;
  end;
end;
```

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#)
[AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#)
[MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronise](#) [OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseUp](#) [OnMouseWheel](#)
[OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#)
[SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#)
[SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#)
[SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#)
[SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#)
[SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#)
[SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#)
[SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#)
[SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#)
[SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#)
[SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#)
[SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#)
[ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#)
[TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#)
[TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#)

[TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#) [TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Easily create iPhone documentation](#)

OnMouseMove_Video

TVideoGrabber.OnMouseMove_Video

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when the user moves the mouse pointer while the mouse pointer is over a control.

Declaration

property OnMouseMove_Video: TOnVideoMouseMove **read** FOnMouseMove **write** FOnMouseMove;

__property TOnVideoMouseMove OnMouseMove_Video=read=FOnMouseMove, write=FOnMouseMove

Event OnMouseMove_Video(VideoWindow As Long, x As Long, y As Long)

Description

returns the coordinates of the original video frame at the cursor location

Created with the Standard Edition of HelpNDoc: [Converting Word Docs to eBooks Made Easy with HelpNDoc](#)

OnMouseMove_Window

TVideoGrabber.OnMouseMove_Window

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when the user moves the mouse pointer while the mouse pointer is over a control.

Declaration

property OnMouseMove_Window: TOnVideoMouseMove **read** FOnMouseMove **write** FOnMouseMove;

__property TOnVideoMouseMove OnMouseMove_Window=read=FOnMouseMove, write=FOnMouseMove

Event OnMouseMove_Window(VideoWindow As Long, x As Long, y As Long)

Description

returns the coordinates of the video window at the cursor location

Created with the Standard Edition of HelpNDoc: [Maximize Your Reach: Convert Your Word Document to an ePub or Kindle eBook](#)

OnMouseUp

TVideoGrabber.OnMouseUp

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when the user releases a mouse button that was pressed with the mouse pointer over a component.

Declaration

property OnMouseUp: TOnVideoMouseDown **read** FOnMouseUp **write** FOnMouseUp;

__property TOnVideoMouseUpDown OnMouseUp=read=FOnMouseUp, write=FOnMouseUp

Event OnMouseUp(VideoWindow As Long, Button As TMouseButton, x As Long, y As Long)

Description

- if [TranslateMouseCoordinates](#) is disabled, the normal OnMouseUp event occurs and returns the real mouse coordinates across the control.
- if [TranslateMouseCoordinates](#) is enabled, this event returns the coordinates corresponding to the native video size.

The difference appears when [Display_AutoSize](#) is disabled or [PreviewZoomSize](#) <> 100. In this case, the event reports the corresponding native video source coordinate, necessary to draw objects over video frames from the [OnFrameBitmap](#) event.

Note that the BitmapInfo parameter of the [OnFrameBitmap](#) event returns directly the current mouse information.

Note: to detect if the video is displayed when the event occurs, test VideoWindow <> -1.
E.g.:

```
procedure TfrmMainForm.VideoGrabberMouseUp(Sender: TObject;
  VideoWindow: Integer; Button: TMouseButton; Shift: TShiftState; X,
  Y: Integer);
begin
  if VideoWindow <> -1 then begin // if the video is displayed
    if chkFreeHandEnabled.Checked then begin
      ...
      ...
    end;
  end;
end;
```

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#)
[AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#)
[MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronon](#) [OnFrameOverlayUsingDC](#)
[OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#)
[OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#)
[SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#)
[SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#)
[SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#)
[SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#)
[SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#)
[SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#)
[SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#)
[SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#)
[SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#)
[SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#)
[SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#)
[ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#)
[TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#)
[TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#)
[TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#)
[TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#)
[TranslateMouseCoordinates](#)

OnMouseUp_Video

TVideoGrabber.OnMouseUp_Video

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when the user releases a mouse button that was pressed with the mouse pointer over a component.

Declaration

property OnMouseUp_Video: TOnVideoMouseUpDown **read** FOnMouseUp **write** FOnMouseUp;

__property TOnVideoMouseUpDown OnMouseUp_Video=read=FOnMouseUp, write=FOnMouseUp

Event OnMouseUp_Video(VideoWindow As Long, Button As TxMouseButton, x As Long, y As Long)

Description

returns the coordinates of the original video frame at the cursor location

OnMouseUp_Window

TVideoGrabber.OnMouseUp_Window

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when the user releases a mouse button that was pressed with the mouse pointer over a component.

Declaration

property OnMouseUp_Window: TOnVideoMouseUpDown **read** FOnMouseUp **write** FOnMouseUp;

__property TOnVideoMouseUpDown OnMouseUp_Window=read=FOnMouseUp, write=FOnMouseUp

Event OnMouseUp_Window(VideoWindow As Long, Button As TxMouseButton, x As Long, y As Long)

Description

returns the coordinates of the video window at the cursor location

OnMouseWheel

TVideoGrabber.OnMouseWheel

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when the user turns the mouse wheel up or down

Declaration

property OnMouseWheel: TOnMouseWheel **read** FOnMouseWheel **write** FOnMouseWheel;

__property TOnMouseWheel OnMouseWheel==FOnMouseWheel, write=FOnMouseWheel;

Event OnMouseWheel (VideoWindow As Long, DirectionUp as Bool, x As Long, y As Long)

Description

Occurs when the user turns the mouse wheel up or down and the pointer is over the video window.

To activate this event enable the [MouseWheelEventEnabled](#) property (disabled by default)

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchronon](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEONHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#) [SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#) [SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#) [SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#) [SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#) [TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#) [TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#) [TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#) [TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Easily create Qt Help files](#)

OnNoVideoDevices

TVideoGrabber.OnNoVideoDevices

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs if no video device is detected.

Declaration

property OnNoVideoDevices: TEventNotification **read** FOnNoVideoDevices **write** FOnNoVideoDevices;

___property TEventNotification OnNoVideoDevices=read=FOnNoVideoDevices, write=FOnNoVideoDevices

Event OnNoVideoDevices()

Description

Occurs if no video device is currently installed or plugged-in on the current platform.

See Also

[WDM drivers](#) [AutoConnectRelatedPins](#) [CurrentFrameRate](#) [DeliveredFrames](#) [DroppedFrames](#) [FixFlickerOrBlackCapture](#) [IsDVDevice](#) [IsVideoControlAvailable](#) [IsVideoCrossbarAvailable](#) [IsVideoDeviceConnected](#) [IsWDMVideoDriver](#) [OnVideoDeviceSelected](#) [RefreshDevicesAndCompressorsLists](#) [ResetVideoDeviceSettings](#) [ShowDialog](#) [VideoDevice](#) [VideoDeviceIndex](#) [VideoDeviceIndexFromId](#) [VideoDeviceName](#) [VideoDevices](#) [VideoDevicesCount](#) [VideoDevicesId](#) [WebcamStillCaptureButton](#)

OnNTPTimeStamp

TVideoGrabber.OnNTPTimeStamp

[Prev](#)[Next](#)[TVideoGrabber](#) [Events](#)

Reports the NTP timestamps

Declaration

TOnNTPTimeStamp=**procedure** (Sender: TObject; last_rtcp_ntp_time: int64; last_rtcp_timestamp: DWORD; delta_rtcp_ntp_time: int64; delta_rtcp_timestamp: DWORD; UTCDateTime: TDateTime; LocalDateTime: TDateTime) **of object**;

```
typedef void __fastcall (__closure *TOnNTPTimeStamp)(System::TObject* Sender, __int64 last_rtcp_ntp_time, unsigned last_rtcp_timestamp, __int64 delta_rtcp_ntp_time, unsigned delta_rtcp_timestamp);
```

Description

Reports the NTP timestamps if the source is a RTSP URL (requires the Dastead RTSP/RTMP/HTTP/ONVIF Source Filter)

OnONVIFDiscoveryCompletedNotification

TVideoGrabber.OnONVIFDiscoveryCompletedNotification

[Prev](#)[Next](#)[TVideoGrabber](#) [Events](#)

Occurs for each ONVIF camera discovered and at the end of the discovery processes

Declaration

property OnONVIFDiscoveryCompleted: TOnONVIFDiscoveryCompletedNotification **read** FOnONVIFDiscoveryCompleted **write** FOnONVIFDiscoveryCompleted;

__property TOnONVIFDiscoveryCompletedNotification OnONVIFDiscoveryCompleted==FOnONVIFDiscoveryCompleted, write=FOnONVIFDiscoveryCompleted;

Description

DiscoveryCallbackStatus: TDiscoveryCallbackStatus:

whether the event occurs for a new camera discovered, or for the completion of the Multicast or IP range discovery

CameraCount: LongInt:

number of cameras currently discovered

See Also

[IR Cut Filter of Axis cameras](#) [TDiscoveryCallbackStatus](#) [TOnONVIFDiscoveryCompletedNotification](#) [TONVIFDeviceInfo](#) [Video stream of IP camera with audio of PC microphone or other audio capture device](#) [ExtraDLLPath](#) [GetLastErrorMessage](#) [IPCameraURL](#) [IsURLResponding](#) [IsURLVideoStreamAvailable](#) [ONVIF_GetStr](#) [ONVIF_SetStr](#) [ONVIFCancelDiscovery](#) [ONVIFDeviceInfo](#) [ONVIFDiscoverCameras](#) [IPRange](#) [ONVIFDiscoverCameras](#) [Multicast](#) [ONVIFEnumCamerasDiscovered](#) [ONVIFPTZGetLimits](#) [ONVIFPTZGetPosition](#) [ONVIFPTZPreset](#) [ONVIFPTZSendAuxiliaryCommand](#) [ONVIFPTZSetPosition](#) [ONVIFPTZStartMove](#) [ONVIFPTZStopMove](#) [ONVIFSnapShot](#) [OpenURLAsync](#) [OpenURLAsyncStatus](#) [SendIPCameraCommand](#) [SetDasteadFilterDllName](#) [SetIPCameraSetting](#)

OnOpenURLAsyncStatusChanged

TVideoGrabber.OnOpenURLAsyncStatusChanged

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

URL opening status notification

Declaration

property OnOpenURLAsyncStatusChanged: TOnOpenURLAsyncStatusChanged read
FOnOpenURLAsyncStatusChanged write FOnOpenURLAsyncStatusChanged;

```
__property TOnOpenURLAsyncStatusChanged  
OnOpenURLAsyncStatusChanged==FOnOpenURLAsyncStatusChanged,  
write=FOnOpenURLAsyncStatusChanged;
```

Description

This [TOnOpenURLAsyncStatusChanged](#) event occurs when status of the URL being opened is changing

Created with the Standard Edition of HelpNDoc: [Revolutionize Your Documentation Review with HelpNDoc's Project Analyzer](#)

OnPlayerBufferingData

TVideoGrabber.OnPlayerBufferingData

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when opening an URL in the player and the player is buffering data.

Declaration

property OnPlayerBufferingData: TOnPlayerBufferingData **read** FOnPlayerBufferingData **write**
FOnPlayerBufferingData;

```
__property TOnPlayerBufferingData OnPlayerBufferingData=read=FOnPlayerBufferingData,  
write=FOnPlayerBufferingData
```

Event OnPlayerBufferingData(StartingToBuffer As Boolean)

Description

Occurs when opening an URL in the player.

This [TOnPlayerBufferingData](#) event occurs just before the clip begins to play:

- a first time when the player starts buffering data (**StartingToBuffer** is true),
- a second time when the player ends buffering data (**StartingToBuffer** is true),

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip](#) [Played](#) [MP4NeedsReindexing](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchron](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

OnPlayerEndOfPlaylist

TVideoGrabber.OnPlayerEndOfPlaylist

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs on the end of the playlist.

Declaration

property OnPlayerEndOfPlaylist: TEventNotification **read** FOnPlayerEndOfPlaylist **write** FOnPlayerEndOfPlaylist;

__property TEventNotification OnPlayerEndOfPlaylist=read=FOnPlayerEndOfPlaylist, write=FOnPlayerEndOfPlaylist

Event OnPlayerEndOfPlaylist()

Description

Occurs when the playlist is active and the end of the playlist is reached.
See the "[Using the playlist](#)" chapter for more information about the playlist feature.

See Also

[TPlaylist](#) [Video formats](#) [GetPlaylist](#) [IsPlaylistActive](#) [Playlist](#) [PlaylistIndex](#)

OnPlayerEndOfStream

TVideoGrabber.OnPlayerEndOfStream

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when AVI playback ends.

Declaration

property OnPlayerEndOfStream: TEventNotification **read** FOnPlayerEndOfStream **write** FOnPlayerEndOfStream;

__property TEventNotification OnPlayerEndOfStream=read=FOnPlayerEndOfStream, write=FOnPlayerEndOfStream

Event OnPlayerEndOfStream()

Description

Occurs during AVI playback, when the end of the video stream is reached.

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchronon](#) [PlayerVideoCodec](#)

[RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#)
[SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#)
[VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Productivity with HelpNDoc's Efficient User Interface](#)

OnPlayerOpened

TVideoGrabber.OnPlayerOpened

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when AVI playback starts.

Declaration

property OnPlayerOpened: TEventNotification **read** FOnPlayerOpened **write** FOnPlayerOpened;

__property TEventNotification OnPlayerOpened=read=FOnPlayerOpened, write=FOnPlayerOpened

Event OnPlayerOpened()

Description

Occurs when AVI playback starts, just before the first video frame is rendered.

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last](#) [Clip](#) [Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchron](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Easily create CHM Help documents](#)

OnPlayerStateChanged

TVideoGrabber.OnPlayerStateChanged

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when the player state changes.

Declaration

property OnPlayerStateChanged: TOnPlayerStateChanged **read** FOnPlayerStateChanged **write** FOnPlayerStateChanged;

__property TOnPlayerStateChanged OnPlayerStateChanged==FOnPlayerStateChanged, write=FOnPlayerStateChanged;

Event OnPlayerStateChanged()

Description

This event occurs when the player state changes. It returns the old and the new player [states](#).

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchron](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Easy EBook and documentation generator](#)

OnPlayerUpdateTrackbarPosition

TVideoGrabber.OnPlayerUpdateTrackbarPosition

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when the player trackbar position must be updated

Declaration

property OnPlayerUpdateTrackbarPosition: TOnFrameProgress **read** FOnPlayerUpdateTrackbarPosition **write** FOnPlayerUpdateTrackbarPosition;

__property TOnFrameProgress
OnPlayerUpdateTrackbarPosition=read=FOnPlayerUpdateTrackbarPosition,
write=FOnPlayerUpdateTrackbarPosition

Event OnPlayerUpdateTrackbarPosition()

Description

This event occurs while playback when you must update your trackbar position with the new frame (or time) position reported by this event.

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchron](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Transform Your CHM Help File Creation Process with HelpNDoc](#)

OnPreviewStarted

TVideoGrabber.OnPreviewStarted

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when preview starts.

Declaration

property OnPreviewStarted: TEventNotification **read** FOnPreviewStarted **write** FOnPreviewStarted;

__property TEventNotification OnPreviewStarted=read=FOnPreviewStarted, write=FOnPreviewStarted

Event OnPreviewStarted()

Description

Occurs after calling [StartPreview](#) , just before the first video frame is rendered.

See Also

[TOnFrameProgress](#) [TOnResizeVideo](#) [AutoRefreshPreview](#) [BackgroundColor](#) [Display](#) [AutoSize](#) [Display_FullScreen](#) [Display_SetLocation](#) [Display_VideoPortEnabled](#) [ImageRatio](#) [InFrameProgressEvent](#) [IsVideoInterlaced](#) [IsVideoPortAvailable](#) [OnFrameProgress](#) [OnKeyPress](#) [OnResizeVideo](#) [PausePreview](#) [PreviewZoomSize](#) [ResumePreview](#) [StartPreview](#) [StartSynchronized](#) [StopPreview](#) [Synchronized](#) [UseNearestVideoSize](#) [VideoHeight](#) [VideoRenderer](#) [VideoRendererPriority](#) [VideoWidth](#) [Visible](#)

Created with the Standard Edition of HelpNDoc: [HelpNDoc's Project Analyzer: Incredible documentation assistant](#)

OnRawAudioSample

TVideoGrabber.OnRawAudioSample

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Gives access to the current audio sample.

Declaration

property OnRawAudioSample: TOnRawSample **read** FOnRawAudioSample **write** FOnRawAudioSample;

__property TOnRawSample OnRawAudioSample=read=FOnRawAudioSample, write=FOnRawAudioSample

Event OnRawAudioSample(pSampleBuffer As Long, SampleBufferSize As Long, SampleDataLength As Long, FormatType As TxFormatType, pFormat As Long, pBitmapInfoHeader As Long, SampleStartTime As Double, SampleStopTime As Double)

Description

This [TOnRawSample](#) event gives access to the current audio sample.

See Also

[TOnRawSample](#) [OnRawVideoSample](#) [RawAudioSampleCapture](#) [RawCaptureAsyncEvent](#) [RawVideoSampleCapture](#)

Created with the Standard Edition of HelpNDoc: [Easily create EPub books](#)

OnRawVideoSample

TVideoGrabber.OnRawVideoSample

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Gives access to the current video sample

Declaration

property OnRawVideoSample: TOnRawSample **read** FOnRawVideoSample **write** FOnRawVideoSample;

__property TOnRawSample OnRawVideoSample=read=FOnRawVideoSample, write=FOnRawVideoSample

Event OnRawVideoSample(pSampleBuffer As Long, SampleBufferSize As Long, SampleDataLength As Long, FormatType As TxFormatType, pFormat As Long, pBitmapInfoHeader As Long, SampleStartTime As Double, SampleStopTime As Double)

Description

This [TOnRawSample](#) gives access to the current video sample

See Also

[TOnRawSample](#) [OnRawAudioSample](#) [RawAudioSampleCapture](#) [RawCaptureAsyncEvent](#) [RawVideoSampleCapture](#)

Created with the Standard Edition of HelpNDoc: [Streamline your documentation process with HelpNDoc's HTML5 template](#)

OnRecordingCompleted

TVideoGrabber.OnRecordingCompleted

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when recording ends.

Declaration

property OnRecordingCompleted: TOnRecordingCompleted **read** FOnRecordingCompleted **write** FOnRecordingCompleted;

__property TOnRecordingCompleted OnRecordingCompleted=read=FOnRecordingCompleted, write=FOnRecordingCompleted

Event OnRecordingCompleted(FileName As String, Success As Boolean)

Description

This [TOnRecordingCompleted](#) event occurs after a [StopRecording](#) call, when the recording process is completed (successfully or not).

This event returns the name of the AVI file created, if any.

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last_Recording_FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [Ensure High-Quality Documentation with HelpNDoc's Hyperlink and Library Item Reports](#)

OnRecordingPaused

TVideoGrabber.OnRecordingPaused

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Events](#)

Notifies a recording paused

Declaration

property OnRecordingPaused: TOnEventNotification **read** FOnRecordingPaused **write** FOnRecordingPaused;

__property TOnEventNotification OnRecordingPaused=read=FOnRecordingPaused, write=FOnRecordingPaused

Event OnRecordingPaused

Description

Occurs when the recording went in a paused state, e.g. after calling [PauseRecording](#).

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last](#) [Recording](#) [FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [Easily create HTML Help documents](#)

OnRecordingReadyToStart

TVideoGrabber.OnRecordingReadyToStart

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when the recording is ready to start.

Declaration

property OnRecordingReadyToStart: TEventNotification **read** FOnRecordingReadyToStart **write** FOnRecordingReadyToStart;

__property TEventNotification OnRecordingReadyToStart=read=FOnRecordingReadyToStart, write=FOnRecordingReadyToStart

Event OnRecordingReadyToStart()

Description

This [TOnRecordingReadyToStart](#) event allows you to start the recording at a given accurate time.

When calling [StartRecording](#) , a delay is necessary to create/open the recording file and build the recording graph.

When these operations are completed, and before the recording starts, this event occurs and holds the recording until exiting.

At this point:

- you can show a dialog asking to start or cancel recording (*see the MainDemo project from sample code*).
- invoke [ResumeRecording\(\)](#) to start the recording, or [StopRecording\(\)](#) to cancel.

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last_Recording_FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: Free CHM Help documentation generator

OnRecordingStarted**TVideoGrabber.OnRecordingStarted**
[Prev](#)
[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when the recording starts writing audio/video streams to disk.

Declaration

property OnRecordingStarted: TOnFileNotification **read** FOnRecordingStarted **write** FOnRecordingStarted;

__property TOnFileNotification OnRecordingStarted=read=FOnRecordingStarted, write=FOnRecordingStarted

Event OnRecordingStarted(FileName As String)

Description

This [TOnFileNotification](#) event occurs when the recording starts writing audio/video streams to disk.

When calling [StartRecording](#) , a delay is necessary to create/open the recording file and build the recording graph. After all these operations are completed, this event occurs when the recording starts saving to disk.

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last_Recording_FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: Elevate Your CHM Help Files with HelpNDoc's Advanced Customization Options

OnReencodingCompleted

TVideoGrabber.OnReencodingCompleted

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when AVI reencodingion begins.

Declaration

property OnReencodingCompleted: TOnSourceFileToDestFileCompleted **read** FOnReencodingCompleted **write** FOnReencodingCompleted;

__property TOnSourceFileToDestFileCompleted

OnReencodingCompleted=read=FOnReencodingCompleted, write=FOnReencodingCompleted

Event OnReencodingCompleted(SourceFile As String, DestFile As String, Success As Boolean)

Description

This [TOnSourceFileToDestFileCompleted](#) event occurs after stopping recording, when AVI reencodingion begins.

Note: [CompressionMode](#) must be set to cm_CompressAfterRecording.

See Also

[OnReencodingProgress](#) [OnReencodingStarted](#) [Reencoding](#) [IncludeAudioStream](#) [Reencoding](#) [IncludeVideoStream](#) [Reencoding](#) [Method](#) [Reencoding](#) [NewVideoClip](#) [Reencoding](#) [SourceVideoClip](#) [Reencoding](#) [StartFrame](#) [Reencoding](#) [StartTime](#) [Reencoding](#) [StopFrame](#) [Reencoding](#) [StopTime](#) [Reencoding](#) [UseAudioCompressor](#) [Reencoding](#) [UseFrameGrabber](#) [Reencoding](#) [UseVideoCompressor](#) [Reencoding](#) [WMVOutput](#) [StartReencoding](#) [StopReencoding](#)

Created with the Standard Edition of HelpNDoc: [Free help authoring environment](#)

OnReencodingProgress

TVideoGrabber.OnReencodingProgress

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs periodically during AVI reencodingion.

Declaration

property OnReencodingProgress: TOnProgress **read** FOnReencodingProgress **write** FOnReencodingProgress;

__property TOnProgress OnReencodingProgress=read=FOnReencodingProgress, write=FOnReencodingProgress

Event OnReencodingProgress(Percent As Long, Position As Double, Duration As Double)

Description

This [TOnProgress](#) event occurs periodically during AVI reencoding.

Note: [CompressionMode](#) must be set to cm_CompressAfterRecording.

See Also

[OnReencodingCompleted](#) [OnReencodingStarted](#) [Reencoding](#) [IncludeAudioStream](#) [Reencoding](#) [IncludeVideoStream](#) [Reencoding](#) [Method](#) [Reencoding](#) [NewVideoClip](#) [Reencoding](#) [SourceVideoClip](#) [Reencoding](#) [StartFrame](#) [Reencoding](#) [StartTime](#) [Reencoding](#) [StopFrame](#) [Reencoding](#) [StopTime](#) [Reencoding](#) [UseAudioCompressor](#) [Reencoding](#) [UseFrameGrabber](#) [Reencoding](#) [UseVideoCompressor](#) [Reencoding](#) [WMVOutput](#) [StartReencoding](#) [StopReencoding](#)

OnReencodingStarted

TVideoGrabber.OnReencodingStarted

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when AVI reencodingion ends.

Declaration

property OnReencodingStarted: TOnSourceFileToDestFileStarted **read** FOnReencodingStarted **write** FOnReencodingStarted;

__property TOnSourceFileToDestFileStarted OnReencodingStarted=read=FOnReencodingStarted, write=FOnReencodingStarted

Event OnReencodingStarted(SourceFile As String, DestFile As String)

Description

This [TOnSourceFileToDestFileStarted](#) event occurs after stopping recording, when AVI reencodingion ends. Note: [CompressionMode](#) must be set to cm_CompressAfterRecording.

See Also

[OnReencodingCompleted](#) [OnReencodingProgress](#) [Reencoding_IncludeAudioStream](#) [Reencoding_IncludeVideoStream](#) [Reencoding_Method](#) [Reencoding_NewVideoClip](#) [Reencoding_SourceVideoClip](#) [Reencoding_StartFrame](#) [Reencoding_StartTime](#) [Reencoding_StopFrame](#) [Reencoding_StopTime](#) [Reencoding_UseAudioCompressor](#) [Reencoding_UseFrameGrabber](#) [Reencoding_UseVideoCompressor](#) [Reencoding_WMVOutput](#) [StartReencoding](#) [StopReencoding](#)

OnReinitializing

TVideoGrabber.OnReinitializing

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when preview, recording or playback starts.

Declaration

property OnReinitializing: TEventNotification **read** FOnReinitializing **write** FOnReinitializing;

__property TEventNotification OnReinitializing=read=FOnReinitializing, write=FOnReinitializing

Event OnReinitializing()

Description

Occurs when preview, recording or playback starts.

OnResizeVideo

TVideoGrabber.OnResizeVideo

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when the video source size changes.

Declaration

property OnResizeVideo: TOnResizeVideo **read** FOnResizeVideo **write** FOnResizeVideo;

__property TOnResizeVideo OnResizeVideo=read=FOnResizeVideo, write=FOnResizeVideo

Event OnResizeVideo(SourceWidth As Long, SourceHeight As Long)

Description

Unlike OnResize, that occurs when the video window is resized, this [TOnResizeVideo](#) event occurs when the video source size changes, even if [Display_AutoSize](#) is disabled.

See Also

[TOnFrameProgress](#) [TOnResizeVideo](#) [AutoRefreshPreview](#) [BackgroundColor](#) [Display_AutoSize](#) [Display_FullScreen](#) [Display_SetLocation](#) [Display_VideoPortEnabled](#) [ImageRatio](#) [InFrameProgressEvent](#) [IsVideoInterlaced](#) [IsVideoPortAvailable](#) [OnFrameProgress](#) [OnKeyPress](#) [OnPreviewStarted](#) [PausePreview](#) [PreviewZoomSize](#) [ResumePreview](#) [StartPreview](#) [StartSynchronized](#) [StopPreview](#) [Synchronized](#) [UseNearestVideoSize](#) [VideoHeight](#) [VideoRenderer](#) [VideoRendererPriority](#) [VideoWidth](#) [Visible](#)

Created with the Standard Edition of HelpNDoc: [Free EPub and documentation generator](#)

OnTextOverlayScrollingCompleted

TVideoGrabber.OnTextOverlayScrollingCompleted

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when a text overlay scrolling is completed

Declaration

property OnTextOverlayScrollingCompleted: TOnTextOverlayScrollingCompleted **read** FOnTextOverlayScrollingCompleted **write** FOnTextOverlayScrollingCompleted

__property TOnTextOverlayScrollingCompleted
OnTextOverlayScrollingCompleted=read=FOnTextOverlayScrollingCompleted,
write=FOnTextOverlayScrollingCompleted

Description

This [TOnTextOverlayScrollingCompleted](#) event occurs when a text overlay scrolling is completed (when the last character has disappeared from the video frame).

The **TextOverlayIndex** parameter returns the index of the text overlay concerned (initially selected by the [TextOverlay_Selector](#) index)

Created with the Standard Edition of HelpNDoc: [From Word to ePub or Kindle eBook: A Comprehensive Guide](#)

OnThirdPartyFilterConnected

TVideoGrabber.OnThirdPartyFilterConnected

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Unknown interface of the [third-party filter](#) when it connects

Declaration

property OnThirdPartyFilterConnected: TOnThirdPartyFilterConnected **read** FOnThirdPartyFilterConnected **write** FOnThirdPartyFilterConnected;

__property TOnThirdPartyFilterConnected
OnThirdPartyFilterConnected=read=FOnThirdPartyFilterConnected, write=FOnThirdPartyFilterConnected

n/a

Description

This [TOnThirdPartyFilterConnected](#) event returns the IUnknown interface of a [third-party filter](#) when it connects, allowing to set/retrieve the filter properties programmatically.

See Also

[TThirdPartyFilterList](#) [TOnThirdPartyFilterConnected](#) [ThirdPartyFilter](#) [AddToList](#) [ThirdPartyFilter](#) [ClearList](#) [ThirdPartyFilter](#) [Enable](#) [ThirdPartyFilter](#) [RemoveFromList](#) [ThirdPartyFilter](#) [ShowDialog](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly create a professional-quality documentation website with HelpNDoc](#)

OnThreadSync

TVideoGrabber.OnThreadSync

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when the component needs to set or unset the video window parent.

Declaration

TOnThreadSync=**procedure** (Sender: TObject; ThreadSyncPoint: TThreadSyncPoint) **of object**;

typedef void __fastcall (__closure *TOnThreadSync)(System::TObject* Sender, TThreadSyncPoint ThreadSyncPoint);

Event OnThreadSync

Description

Occurs in [threaded mode](#) when the component needs to set or unset the video window parent.

See Also

[Opening a clip or an IP URL from a background thread without blocking the main thread](#) [Player features](#) [OnThreadSync](#) [TThreadSyncPoint](#) [EnableThreadMode](#)

Created with the Standard Edition of HelpNDoc: [Revolutionize Your Documentation Review with HelpNDoc's Project Analyzer](#)

OnTVChannelScanCompleted

TVideoGrabber.OnTVChannelScanCompleted

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Notifies the end of the TV channel scanning.

Declaration

property OnTVChannelScanCompleted: TEventNotification **read** FOnTVChannelScanCompleted **write** FOnTVChannelScanCompleted;

__property TEventNotification OnTVChannelScanCompleted=read=FOnTVChannelScanCompleted, write=FOnTVChannelScanCompleted

Event OnTVChannelScanCompleted()

Description

Occurs when the TV channel scanning ends, after invoking [TVStopAutoScan](#).

This event reports the minimal and maximal channel that will be scanned for the current country code and tuner input type.

See Also

[TTVChannelInfo](#) [TOnTVChannelScanStarted](#) [TOnTVChannelSelected](#) [TTunerInputType](#) [GetTVChannelInfo](#) [IsHorizontalSyncLocked](#) [IsTVAudioAvailable](#) [IsTVAutoTuneRunning](#) [IsTVTunerAvailable](#) [OnTVChannelScanStarted](#) [OnTVChannelSelected](#) [TunerFrequency](#) [TunerMode](#) [TVChannel](#) [TVClearFrequencyOverrides](#) [TVCountryCode](#) [TVGetMinMaxChannels](#) [TVSetChannelFrequencyOverride](#) [TVStartAutoScan](#) [TVStopAutoScan](#) [TVTunerInputType](#) [TVUseFrequencyOverrides](#) [VCRHorizontalLocking](#)

Created with the Standard Edition of HelpNDoc: [Free EPub and documentation generator](#)

OnTVChannelScanStarted

TVideoGrabber.OnTVChannelScanStarted

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Events](#)

Notifies the beginning of the TV channel scanning.

Declaration

property OnTVChannelScanStarted: TOnTVChannelScanStarted **read** FOnTVChannelScanStarted **write** FOnTVChannelScanStarted;

__property TOnTVChannelScanStarted OnTVChannelScanStarted=read=FOnTVChannelScanStarted, write=FOnTVChannelScanStarted

Event OnTVChannelScanStarted(MinChannel As Long, MaxChannel As Long)

Description

This [TOnTVChannelScanStarted](#) event occurs when the TV channel scanning begins, after invoking [TVStartAutoScan](#).

This event reports the minimal and maximal channel that will be scanned for the current country code and tuner input type.

See Also

[TTVChannelInfo](#) [TOnTVChannelScanStarted](#) [TOnTVChannelSelected](#) [TTunerInputType](#) [GetTVChannelInfo](#) [IsHorizontalSyncLocked](#) [IsTVAudioAvailable](#) [IsTVAutoTuneRunning](#) [IsTVTunerAvailable](#) [OnTVChannelScanCompleted](#) [OnTVChannelSelected](#) [TunerFrequency](#) [TunerMode](#) [TVChannel](#) [TVClearFrequencyOverrides](#) [TVCountryCode](#) [TVGetMinMaxChannels](#) [TVSetChannelFrequencyOverride](#) [TVStartAutoScan](#) [TVStopAutoScan](#) [TVTunerInputType](#) [TVUseFrequencyOverrides](#) [VCRHorizontalLocking](#)

Created with the Standard Edition of HelpNDoc: [Write eBooks for the Kindle](#)

OnTVChannelSelected

TVideoGrabber.OnTVChannelSelected

[Prev](#)
[Next](#)

[TVideoGrabber](#) [Events](#)

Occurs when a TV channel is selected.

Declaration

property OnTVChannelSelected: TOnTVChannelSelected **read** FOnTVChannelSelected **write** FOnTVChannelSelected;

__property TOnTVChannelSelected OnTVChannelSelected=read=FOnTVChannelSelected, write=FOnTVChannelSelected

Event OnTVChannelSelected(Channel As Long, Locked As Boolean, DefaultVideoFrequency As Long, OverriddenVideoFrequency As Long, TunerVideoFrequency As Long, TunerAudioFrequency As Long)

Description

This [TOnTVChannelSelected](#) event occurs:

- when a TV channel is selected by assigning a value to [TVChannel](#),
- for each TV channel scanned, after invoking [TVStartAutoScan](#).

See the type of this event for more information: [TOnTVChannelSelected](#).

See Also

[TTVChannelInfo](#) [TOnTVChannelScanStarted](#) [TOnTVChannelSelected](#) [TTunerInputType](#) [GetTVChannelInfo](#) [IsHorizontalSyncLocked](#) [IsTVAudioAvailable](#) [IsTVAutoTuneRunning](#) [IsTVTunerAvailable](#) [OnTVChannelScanCompleted](#) [OnTVChannelScanStarted](#) [TunerFrequency](#) [TunerMode](#) [TVChannel](#) [TVClearFrequencyOverrides](#) [TVCountryCode](#) [TVGetMinMaxChannels](#) [TVSetChannelFrequencyOverride](#) [TVStartAutoScan](#) [TVStopAutoScan](#) [TVTunerInputType](#) [TVUseFrequencyOverrides](#) [VCRHorizontalLocking](#)

Created with the Standard Edition of HelpNDoc: [Easy EPub and documentation editor](#)

OnVideoCompressionSettings

TVideoGrabber.OnVideoCompressionSettings

[Prev](#)

[Next](#)

[TVideoGrabber](#) [Events](#)

Returns capabilities about the [current video compressor](#).

Declaration

property OnVideoCompressionSettings: TOnVideoCompressionSettings **read**
FOnVideoCompressionSettings **write** FOnVideoCompressionSettings;

__property TOnVideoCompressionSettings

OnVideoCompressionSettings=read=FOnVideoCompressionSettings, write=FOnVideoCompressionSettings

Event OnVideoCompressionSettings(CanKeyFrameRate As Boolean, CanPFFramesPerKeyFrame As Boolean, CanQuality As Boolean, CanWindowSize As Boolean, DefaultFrameRate As Long, DefaultPFFramesPerKey As Long, DefaultQuality As Double)

Description

This [TOnTVChannelSelected](#) event returns the general compression capabilities of the [current video compressor](#), and its default values.

See "General video compression properties" in the [Software compression using codecs](#) chapter for more information about this event.

See Also

[Recording methods and properties](#) [TCompressionType](#) [TOnVideoCompressionSettings](#) [AudioCompressor](#) [AudioCompressorIndex](#) [AudioCompressorName](#) [AudioCompressors](#) [AudioCompressorsCount](#) [CompressionMode](#) [CompressionType](#) [GetVideoCompressionSettings](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [RefreshDevicesAndCompressorsLists](#) [SaveCompressorSettingsToDataString](#) [SetVideoCompressionDefaults](#) [SetVideoCompressionSettings](#) [VideoCompression](#) [KeyFrameRate](#) [VideoCompression](#) [PFFramesPerKeyFrame](#) [VideoCompression](#) [Quality](#) [VideoCompression](#) [WindowSize](#) [VideoCompressor](#) [VideoCompressorIndex](#) [VideoCompressorName](#) [VideoCompressors](#) [VideoCompressorsCount](#)

Created with the Standard Edition of HelpNDoc: [Make the switch to CHM with HelpNDoc's hassle-free WinHelp HLP to CHM conversion tool](#)

OnVideoDeviceSelected

TVideoGrabber.OnVideoDeviceSelected

[Prev](#)

[Next](#)

TVideoGrabber Events

Occurs when a video capture device is selected.

Declaration

property OnVideoDeviceSelected: TEventNotification **read** FOnVideoDeviceSelected **write** FOnVideoDeviceSelected;

__property TEventNotification OnVideoDeviceSelected=read=FOnVideoDeviceSelected, write=FOnVideoDeviceSelected

Event OnVideoDeviceSelected()

Description

Occurs when an video capture device is selected by the [VideoDevice](#) property. The device-dependent settings of this video capture device are reloaded and then this event occurs.

See Also

[WDM drivers](#) [AutoConnectRelatedPins](#) [CurrentFrameRate](#) [DeliveredFrames](#) [DroppedFrames](#) [FixFlickerOrBlackCapture](#) [IsDVDevice](#) [IsVideoControlAvailable](#) [IsVideoCrossbarAvailable](#) [IsVideoDeviceConnected](#) [IsWDMVideoDriver](#) [OnNoVideoDevices](#) [RefreshDevicesAndCompressorsLists](#) [ResetVideoDeviceSettings](#) [ShowDialog](#) [VideoDevice](#) [VideoDeviceIndex](#) [VideoDeviceIndexFromId](#) [VideoDeviceName](#) [VideoDevices](#) [VideoDevicesCount](#) [VideoDevicesId](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: Make Your PDFs More Secure with Encryption and Password Protection

OnVideoFromBitmapsNextFrameNeeded

TVideoGrabber.OnVideoFromBitmapsNextFrameNeeded

[Prev](#)

TVideoGrabber Events

Requests a bitmap handle to build the next video frame.

Declaration

property OnVideoFromBitmapsNextFrameNeeded: TOnVideoFromBitmapsNextFrameNeeded **read** FOnVideoFromBitmapsNextFrameNeeded **write** FOnVideoFromBitmapsNextFrameNeeded;

__property TOnVideoFromBitmapsNextFrameNeeded OnVideoFromBitmapsNextFrameNeeded=read=FOnVideoFromBitmapsNextFrameNeeded, write=FOnVideoFromBitmapsNextFrameNeeded

Event OnVideoFromBitmapsNextFrameNeeded(FirstSample As Boolean, BitmapHandle, BMPorJPEGFile, EndOfData)

Description

This [TOnVideoFromBitmapsNextFrameNeeded](#) event occurs for each video frame that will be build from a bitmap handle when a preview or recording graph is running and [VideoSource](#) = vs_JPEGsOrBitmaps.

From this event you must invoke [SendImageToVideoFromBitmaps](#), and pass as parameter:

- either a file path to a BMP or JPEG file through the **BMPorJPEGFile** parameter (a)
- either a bitmap handle through the **BitmapHandle** parameter (b)

If no more images are available, set the **EndOfData** parameter to true to notify the end of the recording or preview.

Remarks:

- (a) if a BMPorJPEGFile string is specified (the string is not empty), set the BitmapHandle parameter to 0
- (b) if a bitmap handle is used, enable **CanFreeBitmapHandle** if TVideoGrabber must free the bitmap handle, or disable **CanFreeBitmapHandle** if you need to reuse the bitmap later.

You can find sample code in the MainDemo project included in the package.

See Also

[TOnVideoFromBitmapsNextFrameNeeded](#)

Created with the Standard Edition of HelpNDoc: [Easily create HTML Help documents](#)

Types

Created with the Standard Edition of HelpNDoc: [Produce online help for Qt applications](#)

TAero

TAero

[Prev](#)

[Next](#)

Aero type

Declaration

TAero=(ae_Default, ae_AutoBestRenderingQuality, ae_ForceOnWhenStartingVideo, ae_ForceOffWhenStartingVideo, ae_ForceOnImmediately, ae_ForceOffImmediately);

enum TAero ae_Default, ae_AutoBestRenderingQuality, ae_ForceOnWhenStartingVideo, ae_ForceOffWhenStartingVideo, ae_ForceOnImmediately, ae_ForceOffImmediately ;

Description

Type of the Aero property

ae_Default: does nothing

ae_AutoBestRenderingQuality: automatically disable Aero if required for the best video quality

ae_ForceOnWhenStartingVideo: re-enable aero (if disabled) when starting the video

ae_ForceOffWhenStartingVideo: disable aero (if enabled) when starting the video

ae_ForceOnImmediately: re-enable aero immediately

ae_ForceOffImmediately: disable aero immediately

Created with the Standard Edition of HelpNDoc: [Full-featured EPub generator](#)

TApplicationPriority

TApplicationPriority

[Prev](#)

[Next](#)

Type of the [ApplicationPriority](#) property.

Description

Type of the [ApplicationPriority](#) property.

Values

ap_default:

current priority

(usually ap_normal
value)

ap_idle:

idle priority

ap_normal:

normal priority

ap_high:

high priority

Created with the Standard Edition of HelpNDoc: [Bring your WinHelp HLP help files into the present with](#)

TASFDeinterlaceMode

TASFDeinterlaceMode

[Prev](#)

[Next](#)

Type of the [ASFDeinterlaceMode](#) property.

Unit

[VidGrab](#)

Declaration

```
TASFDeinterlaceMode=(adm_NotInterlaced, adm_DeinterlaceNormal, adm_DeinterlaceHalfSize,
adm_DeinterlaceHalfSizeDoubleRate, adm_DeinterlaceInverseTelecine,
adm_DeinterlaceVerticalHalfSizeDoubleRate);
```

```
enum TASFDeinterlaceMode adm_NotInterlaced, adm_DeinterlaceNormal, adm_DeinterlaceHalfSize,
adm_DeinterlaceHalfSizeDoubleRate, adm_DeinterlaceInverseTelecine,
adm_DeinterlaceVerticalHalfSizeDoubleRate ;
```

Description

Type of the [ASFDeinterlaceMode](#) property.

Possible values:

```
adm_NotInterlaced
adm_DeinterlaceNormal
adm_DeinterlaceHalfSize
adm_DeinterlaceHalfSizeDoubleRate
adm_DeinterlaceInverseTelecine
adm_DeinterlaceVerticalHalfSizeDoubleRate
```

See Also

Created with the Standard Edition of HelpNDoc: [Experience the power of a responsive website for your documentation](#)

TASFProfileVersion

TASFProfileVersion

[Prev](#)

[Next](#)

Type of the [ASFProfileVersion](#) property.

Description

Allowed values:

```
apv_ProfileVersion_8: use the WMV profiles version 8
apv_ProfileVersion_9: use the WMV profiles version 9
```

Created with the Standard Edition of HelpNDoc: [Converting Word Docs to eBooks Made Easy with HelpNDoc](#)

TAspectRatio

TAspectRatio

[Prev](#)

[Next](#)

Type of the [Display_AspectRatio](#) and [DualDisplay_AspectRatio](#) properties.

Description

ar_Box: the video appears in full within the control in a letterbox (bars on the top and the bottom) or pillarbox (bars on the left and the right)

ar_NoResize: the video appears in its "native" size. If the native size is larger than then control, it appears centered within the control. If the native size is larger than the control, the video appears truncated.

ar_Stretch: the video is stretched as needed to fill the control

ar_PanScan: the video may be panned or scanned, according to the [Display_PanScanRatio](#) value (in the 0..100 range, default value 50).

Created with the Standard Edition of HelpNDoc: [Easily create iPhone documentation](#)

TAudioSource

TAudioSource

[Prev](#)
[Next](#)

Type of the [AudioSource](#) property.

Declaration

```
TAudioSource=(as_Default, as_UseExternalAudio);
```

```
enum TAudioSource as_Default, as_UseExternalAudio ;
```

Description

Type of the [AudioSource](#) property.

Possible values:

as_Default:

uses the audio output of the [video capture device](#) if any, otherwise uses the current [audio capture device](#).

as_UseExternalAudio:

always uses the current [audio capture device](#) for the audio recording, even if the current source (e.g. an URL) exposes an audio output.

as_SpeakerOutput:

uses the soundboard's audio output (captures whatever is rendered)

as_DefaultWithSpeakerRecording: (during recording only)

if AudioDeviceRendering is enabled, the audio of the current source is rendered, but the soundboard's audio output is recorded

as_silent

generates an "empty" audio stream

as_IPCamera

allows to use an IP camera as audio source when previewing or recording an USB source, the screen or the video mixer (when VideoSource = vs_VideoCaptureDevice, vs_ScreenRecording or vs_Mixer).

VideoGrabber.IPCameraURL and eventually the VideoGrabber.SetAuthentication (at_IPCamera, ...) must be set accordingly.

Created with the Standard Edition of HelpNDoc: [Easily convert your WinHelp HLP help files to CHM with HelpNDoc's step-by-step guide](#)

TAuthenticationType

TAuthenticationType

[Prev](#)
[Next](#)

Authentication type.

Declaration

TAuthenticationType=(at_PublishingPoint, at_IPCamera, at_StreamingUrl);

enum TAuthenticationType at_PublishingPoint, at_IPCamera, at_StreamingUrl ;

Description

Type used by the [SetAuthentication](#) method and the [OnAuthenticationNeeded](#) event.

Created with the Standard Edition of HelpNDoc: [Easily create EPub books](#)

TAutoFileName**TAutoFileName**
[Prev](#)
[Next](#)

Type used by the [AutoFileName](#) property.

Unit

[VidGrab](#)

Declaration

TAutoFileName=(fn_Sequential, fn_DateTime);

enum TAutoFileName fn_Sequential, fn_DateTime ;

Description**fn_Sequential:**

The file name is based on a sequential number prefixed by [AutoFilePrefix](#) , starting from 000001.

e.g.: vg000010.avi, vg000011.avi

fn_DateTime:

The file name is based on the yymmdd_hhmmss_zzz date/time format (zzz = milliseconds) prefixed by

[AutoFilePrefix](#) .

e.g. vg030616_090603_904.jpg

fn_Counter:

equivalent to **fn_Sequential**, but existing files are overwritten.

fn_Modulo:

used to capture the last N frames.

If you set BurstMode = true and BurstCount = N, the last N frames are generated, and when N is reached the file number restarts from 1.

E.g. if BurstCount = 5, the files are written in the following order:

vg000001.jpg vg000002.jpg vg000003.jpg vg000004.jpg vg000005.jpg **vg000001.jpg vg000002.jpg vg000003.jpg ...**

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TFrameCaptureDest](#) [TOnFrameCaptureCompleted](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [BurstCount](#) [BurstInterval](#) [BurstMode](#) [BurstType](#) [CaptureFrameSyncTo](#) [CaptureFrameTo](#) [Encoder](#) [SetInt](#) [FrameCaptureHeight](#) [FrameCaptureWidth](#) [FrameCaptureWithoutOverlay](#) [FrameCaptureZoomSize](#) [GetFrameInfoString](#) [GetLastFrameAsHBITMAP](#) [GetLastFrameAsTBitmap](#) [GetLastFrameBitmapBits](#) [GetLastFrameBitmapBits2](#) [GetLastFrameWaitTimeoutMs](#) [HoldRecording](#) [IsRecordingPaused](#) [JPEGPerformance](#) [JPEGProgressiveDisplay](#) [JPEGQuality](#) [Last_BurstFrameCapture](#) [FileName](#) [Last_CaptureFrameTo](#) [FileName](#) [Last_Recording](#) [FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnFrameBitmap](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#)

[RecordingInNativeFormat](#)
[RecordingKBytesWrittenToDisk](#)
[RecordingMethod](#)
[RecordingSize](#)
[RecordingTimer](#)
[RecordingTimerInterval](#)
[RecordingWidth](#)
[ResumeRecording](#)
[SaveCompressorSettingsToDataString](#)
[SetFrameCaptureBounds](#)
[SetMultiplexerFilterByName](#)
[StartAudioRecording](#)
[StartRecording](#)
[StartSynchronized](#)
[StopRecording](#)
[StoragePath](#)
[Synchronized](#)
[SyncPreview](#)
[VideoPlayableWhileRecording](#)
[WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: Streamline your documentation process with HelpNDoc's HTML5 template

TAVIInfoType

TAVIInfoType

[Prev](#)
[Next](#)

Type used by the AVIInfo2 function

Description

Type used to select the string returned by the AVIInfo2 function

Possible values:

av_Duration
 av_FrameCount
 av_VideoWidth
 av_VideoHeight
 av_VideoFrameRateFps
 av_VideoCodec
 av_AudioCodec
 av_AvgBitRate
 av_AudioChannels
 av_AudioSamplesPerSec
 av_AudioBitsPerSample
 av_FileSizeInKB
 av_AudioStreams

Created with the Standard Edition of HelpNDoc: Make the switch to CHM with HelpNDoc's hassle-free WinHelp HLP to CHM conversion tool

TAVIMuxConfig

TAVIMuxConfig

[Prev](#)
[Next](#)

TAVIMuxConfig type

Declaration

TAVIMuxConfig=(avmx_SetInterleavingMode, avmx_SetInterleave, avmx_SetPreroll);

enum TAVIMuxConfig avmx_SetInterleavingMode, avmx_SetInterleave, avmx_SetPreroll ;

Description

Type used by [SetAVIMuxConfig](#)

See Also

[Recording methods and properties](#)
[TASFDeinterlaceMode](#)
[TAutoFileName](#)
[TOnRecordingCompleted](#)
[TOnRecordingReadyToStart](#)
[TRecordingMethod](#)
[TSyncPreview](#)
[AudioRecording](#)
[AudioSyncAdjustment](#)
[AutoFileNameMinDigits](#)
[AVIDurationUpdated](#)
[AVIFormatOpenDML](#)
[AVIHeaderInfo](#)
[AVIInfo](#)
[Encoder](#)
[SetInt](#)
[HoldRecording](#)
[IsRecordingPaused](#)
[Last_Recording_FileName](#)
[OnBacktimedFramesCountReached](#)
[OnCopyPreallocDataCompleted](#)
[OnCopyPreallocDataProgress](#)
[OnCopyPreallocDataStarted](#)
[OnCreatePreallocFileCompleted](#)
[OnCreatePreallocFileStarted](#)
[OnDiskFull](#)
[OnRecordingCompleted](#)

[OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#)
[OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#)
[PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#)
[RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#)
[RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#)
[RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#)
[SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#)
[StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: Maximize Your Documentation Output with HelpNDoc's
Advanced Project Analyzer

TCardinalDirection

TCardinalDirection

[Prev](#)
[Next](#)

Type used by the [TextOverlay_ShadowDirection](#) property.

Declaration

```
TCardinalDirection=(cd_North, cd_NorthEast, cd_East, cd_SouthEast, cd_South, cd_SouthWest,  
cd_West, cd_NorthWest, cd_Center);
```

```
enum TCardinalDirection _North, cd_NorthEast, cd_East, cd_SouthEast, cd_South, cd_SouthWest,  
cd_West, cd_NorthWest, cd_Center;
```

Description

Type used by the [TextOverlay_ShadowDirection](#) property.

The value can be one of the following:

```
cd_North  
cd_NorthEast  
cd_East  
cd_SouthEast  
cd_South  
cd_SouthWest  
cd_West  
cd_NorthWest  
cd_Center
```

See Also

[TOnFrameOverlayUsingDC](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#)
[DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#)
[OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#)
[OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#)
[OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#)
[SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#)
[SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#)
[SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#)
[SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#)
[SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#)
[SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#)
[SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#)
[SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#)
[SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#)
[SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#)

[SetTextOverlay_TargetDisplay](#)
[SetTextOverlay_Top](#)
[SetTextOverlay_Transparent](#)
[ShapeOverlay](#)
[ShapeOverlayEnabled](#)
[ShapeOverlayList](#)
[TextOverlay_Align](#)
[TextOverlay_BkColor](#)
[TextOverlay_CreateCustomFont](#)
[TextOverlay_CreateCustomFont2](#)
[TextOverlay_Enabled](#)
[TextOverlay_Font](#)
[TextOverlay_FontColor](#)
[TextOverlay_Left](#)
[TextOverlay_Right](#)
[TextOverlay_Scrolling](#)
[TextOverlay_ScrollingSpeed](#)
[TextOverlay_Selector](#)
[TextOverlay_Shadow](#)
[TextOverlay_ShadowColor](#)
[TextOverlay_ShadowDirection](#)
[TextOverlay_String](#)
[TextOverlay_Top](#)
[TextOverlay_Transparent](#)
[TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Transform your help documentation into a stunning website](#)

TDialog

TDialog

[Prev](#)
[Next](#)

Type used by the [ShowDialog](#) function.

Unit

[VidGrab](#)

Declaration

```
TDialog=(dlg_VideoDevice, dlg_VideoCompressor, dlg_AudioCompressor, dlg_StreamConfig,
dlg_VfwFormat, dlg_VfwSource, dlg_vfwDisplay, dlg_VideoCrossbar, dlg_AudioCrossbar, dlg_TVTuner,
dlg_TVAudio, dlg_AudioDevice, dlg_NetShowConfig, dlg_DScaler, dlg_FFDSHOWVideo,
dlg_FFDSHOWAudio);
```

```
enum TDialog _VideoDevice, dlg_VideoCompressor, dlg_AudioCompressor, dlg_StreamConfig,
dlg_VfwFormat, dlg_VfwSource, dlg_vfwDisplay, dlg_VideoCrossbar, dlg_AudioCrossbar, dlg_TVTuner,
dlg_TVAudio, dlg_AudioDevice, dlg_NetShowConfig, dlg_DScaler, dlg_FFDSHOWVideo,
dlg_FFDSHOWAudio;
```

Description

Type used by the [ShowDialog](#) function.

Value Meaning

dlg_VideoDevice: dialog of the [video capture device](#) ,
dlg_VideoCompressor: dialog of the [video compressor](#) , if any,
dlg_AudioCompressor: dialog of the [audio compressor](#) , if any,
dlg_StreamConfig: dialog of the video capture stream,
dlg_VfwFormat: format dialog of the video capture device ([vfw](#) drivers only),
dlg_VfwSource: source dialog of the video capture device ([vfw](#) drivers only),
dlg_vfwDisplay: source dialog of the video capture device ([vfw](#) drivers only),
dlg_VideoCrossbar: dialog of the video crossbar, if any,
dlg_AudioCrossbar: dialog of the audio crossbar, if any,
dlg_TVTuner: dialog of the [TV tuner](#) , if any,
dlg_TVAudio: dialog of the TV audio interface, if any,
dlg_AudioDevice: dialog of the [audio capture device](#)
dlg_NetShowConfig: dialog of the streaming source (when playing from an URL)
dlg_DScaler: dialog of the DScaler deinterlacer (see [DScaler deinterlacer](#))
dlg_FFDSHOWVideo: dialog of the [FFDSHOW Video Decoder](#)
dlg_FFDSHOWAudio: dialog of the [FFDSHOW Audio Decoder](#)

Created with the Standard Edition of HelpNDoc: [Transform Your Help Documentation Process with a Help Authoring Tool](#)

TDiscoveryCallbackStatus

TDiscoveryCallbackStatus

[Prev](#)
[Next](#)

TDiscoveryCallbackStatus type

Description

TDiscoveryCallbackStatus = (dcs_CameraFound, dcs_MulticastCompleted, dcs_IPRangeCompleted);
This type is returned the the [OnONVIFDiscoveryCompletedNotification](#) event.

See Also

[IR Cut Filter of Axis cameras](#) [TOnONVIFDiscoveryCompletedNotification](#) [TONVIFDeviceInfo](#) [Video stream of IP camera with audio of PC microphone or other audio capture device](#) [ExtraDLLPath](#) [GetLastErrorMessage](#) [IPCameraURL](#) [IsURLResponding](#) [IsURLVideoStreamAvailable](#) [OnONVIFDiscoveryCompletedNotification](#) [ONVIF_GetStr](#) [ONVIF_SetStr](#) [ONVIFCancelDiscovery](#) [ONVIFDeviceInfo](#) [ONVIFDiscoverCameras_IPRange](#) [ONVIFDiscoverCameras_Multicast](#) [ONVIFEnumCamerasDiscovered](#) [ONVIFPTZGetLimits](#) [ONVIFPTZGetPosition](#) [ONVIFPTZPreset](#) [ONVIFPTZSendAuxiliaryCommand](#) [ONVIFPTZSetPosition](#) [ONVIFPTZStartMove](#) [ONVIFPTZStopMove](#) [ONVIFSnapShot](#) [OpenURLAsync](#) [OpenURLAsyncStatus](#) [SendIPCameraCommand](#) [SetDatasteadFilterDllName](#) [SetIPCameraSetting](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Support Your Windows Applications with HelpNDoc's CHM Generation](#)

TDVCommand

TDVCommand

[Prev](#)

[Next](#)

Type used by the DV camcoders functions.

Unit

[VidGrab](#)

Declaration

TDVCommand=(dv_Play, dv_Stop, dv_Freeze, dv_Thaw, dv_Ff, dv_Rew, dv_Record, dv_RecordFreeze, dv_RecordStrobe, dv_StepFwd, dv_StepRev, dv_ModeShuttle, dv_PlayFastestFwd, dv_PlaySlowestFwd, dv_PlayFastestRev, dv_PlaySlowestRev);

enum TDVCommand dv_Play, dv_Stop, dv_Freeze, dv_Thaw, dv_Ff, dv_Rew, dv_Record, dv_RecordFreeze, dv_RecordStrobe, dv_StepFwd, dv_StepRev, dv_ModeShuttle, dv_PlayFastestFwd, dv_PlaySlowestFwd, dv_PlayFastestRev, dv_PlaySlowestRev ;

Description

Type used by the DV camcoders functions.

Values:

dv_Play
dv_Stop
dv_Freeze
dv_Thaw
dv_Ff
dv_Rew
dv_Record
dv_RecordFreeze
dv_RecordStrobe
dv_StepFwd
dv_StepRev
dv_ModeShuttle
dv_PlayFastestFwd
dv_PlaySlowestFwd
dv_PlayFastestRev
dv_PlaySlowestRev

See Also

[TDVSize](#) [TDVVideoFormat](#) [TDVVideoStandard](#) [TOnDVCommandCompleted](#) [DVDDateTimeEnabled](#)
[DVDDiscontinuityMinimumInterval](#) [DVRecordingInNativeFormatSeparatesStreams](#) [DVReduceFrameRate](#)
[DVRgb219](#) [DVTimeCodeEnabled](#) [IsDigitalVideoIn](#) [IsDVCommandAvailable](#) [IsTimeCodeReaderAvailable](#)
[OnDVCommandCompleted](#) [SendDVCommand](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Create High-Quality Help Documentation with a Help Authoring Tool](#)

TDVDInfoType

TDVDInfoType

[Prev](#)[Next](#)

TDVDInfo type

Declaration

TDVDInfoType=(dvd_NumberOfVolumes, dvd_TotalDuration, dvd_NumberOfTitles, dvd_TitleDuration, dvd_TitleFrameRate, dvd_SourceResolutionX, dvd_SourceResolutionY, dvd_TitleFrameCount)

enum TDVDInfoType dvd_NumberOfVolumes, dvd_TotalDuration, dvd_NumberOfTitles, dvd_TitleDuration, dvd_TitleFrameRate, dvd_SourceResolutionX, dvd_SourceResolutionY, dvd_TitleFrameCount ;

Description

Type used by the [DVDInfo](#) function.

Created with the Standard Edition of HelpNDoc: [Effortlessly Spot and Fix Problems in Your Documentation with HelpNDoc's Project Analyzer](#)

TDVSize

TDVSize

[Prev](#)[Next](#)

Type of the [PlayerDVSize](#) property.

Unit

[VidGrab](#)

Declaration

TDVSize=(dv_Full, dv_Half, dv_Quarter, dv_DC);

enum TDVSize dv_Full, dv_Half, dv_Quarter, dv_DC ;

Description

Type of the [PlayerDVSize](#) property.

Value Meaning

dvFull full size
 dvHalf half size
 dvQuarter quarter size
 dvDC 1/8 size

See Also

[TDVVideoFormat](#) [TDVVideoStandard](#) [TOnDVCommandCompleted](#) [DVDDateTimeEnabled](#)

[DVDDiscontinuityMinimumInterval](#) [DVRecordingInNativeFormatSeparatesStreams](#) [DVReduceFrameRate](#) [DVRgb219](#) [DVTimeCodeEnabled](#) [IsDigitalVideoIn](#) [IsDVCommandAvailable](#) [IsTimeCodeReaderAvailable](#) [OnDVCommandCompleted](#) [SendDVCommand](#)

Created with the Standard Edition of HelpNDoc: [Why Microsoft Word Isn't Cut Out for Documentation: The Benefits of a Help Authoring Tool](#)

TDVVideoFormat

TDVVideoFormat

[Prev](#)

[Next](#)

Type of DV Format.

Unit

[VidGrab](#)

Declaration

TDVVideoFormat=(dvf_Default, dvf_DVSD, dvf_DVHD, dvf_DVSL);

enum TDVVideoFormat dvf_Default, dvf_DVSD, dvf_DVHD, dvf_DVSL ;

Description

Value

dvDVSD

dvDVHD

dvDVSL

See Also

[TDVSize](#) [TDVVideoStandard](#) [TOnDVCommandCompleted](#) [DVDateTimeEnabled](#) [DVDDiscontinuityMinimumInterval](#) [DVRecordingInNativeFormatSeparatesStreams](#) [DVReduceFrameRate](#) [DVRgb219](#) [DVTimeCodeEnabled](#) [IsDigitalVideoIn](#) [IsDVCommandAvailable](#) [IsTimeCodeReaderAvailable](#) [OnDVCommandCompleted](#) [SendDVCommand](#)

Created with the Standard Edition of HelpNDoc: [Converting Word Documents to eBooks: A Step-by-Step Guide with HelpNDoc](#)

TDVVideoStandard

TDVVideoStandard

[Prev](#)

[Next](#)

Type of DV standard.

Unit

[VidGrab](#)

Declaration

TDVVideoStandard=(dvs_Default, dvs_PAL, dvs_NTSC);

enum TDVVideoStandard dvs_Default, dvs_PAL, dvs_NTSC ;

Description

Value

dvPAL

dvNTSC

See Also

[TDVSize](#)
[TDVVideoFormat](#)
[TOnDVCommandCompleted](#)
[DVDateTimeEnabled](#)
[DVDDiscontinuityMinimumInterval](#)
[DVRecordingInNativeFormatSeparatesStreams](#)
[DVReduceFrameRate](#)
[DVRgb219](#)
[DVTimeCodeEnabled](#)
[IsDigitalVideoIn](#)
[IsDVCommandAvailable](#)
[IsTimeCodeReaderAvailable](#)
[OnDVCommandCompleted](#)
[SendDVCommand](#)

Created with the Standard Edition of HelpNDoc: [Full-featured EBook editor](#)

TEncoder_int

TEncoder_int

[Prev](#)
[Next](#)

TEncoder_int

Declaration

```

TEncoder_int=(
Enc_IsActive_bool,
Enc_Bytes_Written_kb_readonly,
Enc_Audio_Enabled_bool,
Enc_Audio_Channels,
Enc_Audio_SamplesPerSec,
Enc_Audio_BitsPerSample,
Enc_Audio_BitRate_kb,
Enc_Audio_PCM_Format,
Enc_Video_Enabled_bool,
Enc_Video_Width,
Enc_Video_Height,
Enc_Video_BitCount,
Enc_Video_AvgTimePerFrame,
Enc_Video_BitRate_kb,
Enc_Video_rc_MinBitRate_kb,
Enc_Video_rc_MaxBitRate_kb,
Enc_Video_rc_BufferSize_kb,
Enc_Video_IDR_Interval,
Enc_Video_Max_BFrames,
Enc_Video_FrameRate_x100,
Enc_IsRealTime_bool,
Enc_IsScreenRecording_bool,
Enc_Video_Thread_Count,
Enc_Video_GPU_Encoder,
Enc_Video_GPU_EncoderDevice

```

```

enum TEncoder_int Enc_IsActive, Enc_Bytes_Written_kb_readonly, Enc_Audio_Enabled,
Enc_Audio_Channels, Enc_Audio_SamplesPerSec, Enc_Audio_BitsPerSample, Enc_Audio_BitRate_kb,
Enc_Video_Enabled, Enc_Video_Width, Enc_Video_Height, Enc_Video_BitRate_kb,
Enc_Video_rc_MinBitRate_kb, Enc_Video_rc_MaxBitRate_kb, Enc_Video_rc_BufferSize_kb,
Enc_Video_IDR_Interval, Enc_Video_Max_BFrames, Enc_Video_FrameRate_x100,
Enc_Video_Thread_Count, Enc_Video_GPU_Encoder, Enc_Video_GPU_Encoder_DeviceNumber ;

```


Enc_Video_GPU_Encoder possible values (pass the index as integer, in the 0..n-1 range):

```
TGPUEncoder=(
Enc_GPU_None,
Enc_GPU_Auto,
Enc_GPU_Intel_QSV,
Enc_GPU_NVidia_NVENC,
Enc_GPU_AMD_AMF
```

Enc_Audio_PCM_Format possible values (pass the index as integer, in the 0..n-1 range):

```
TAudioPCMFormat=(
PCM_U8,
PCM_S16,
PCM_S32,
PCM_FLT,
PCM_DBL
```

Description

The TEncoder_int type is used by the [Datastead Encoder](#)

Created with the Standard Edition of HelpNDoc: [Experience the Power and Ease of Use of a Help Authoring Tool](#)

TEncoder_str

TEncoder_str

[Prev](#)
[Next](#)

TEncoder_str

Declaration

```
TEncoder_str=(
Enc_OutputURL,
Enc_Video_Codec,
Enc_Audio_Codec,
Enc_Extra_Parameters
```

```
enum TEncoder_str Enc_OutputURL, Enc_Video_Codec, Enc_Audio_Codec, Enc_Extra_Parameters ;
```

Description

The TEncoder_str type is used by the [Datastead Encoder](#)

Created with the Standard Edition of HelpNDoc: [Streamline Your Documentation Creation with a Help Authoring Tool](#)

TEncryptionMethod

TEncryptionMethod

[Prev](#)
[Next](#)

Type of encryption method

Declaration

```
TEncryptionMethod =(em_Fast_Encryption, em_AES_Encryption);
```

```
enum TEncryptionMethod: unsigned int em_Fast_Encryption, em_AES_Encryption ;
```

Description

Type of encryption method:

em_Fast_Encryption: fast encryption, suitable for real-time encryption of large video sizes and/or several concurrent recordings

em_AES_Encryption: AES encryption. Safer, but slower.

See Also

[Decrypt_File](#) [Encrypt_File](#) [EncryptionMethod](#) [SetDecryptionKey](#) [SetEncryptionKey](#)

Created with the Standard Edition of HelpNDoc: [Make CHM Help File Creation a Breeze with HelpNDoc](#)

TEventNotification

TEventNotification

[Prev](#)

[Next](#)

TNotifyEvent

Unit

[VidGrab](#)

Declaration

TEventNotification=**procedure**(Sender: TObject) **of object**;

typedef void __fastcall(__closure *TEventNotification)(System::TObject *Sender);

Description

Equivalent to a TNotifyEvent type.

Created with the Standard Edition of HelpNDoc: [Easily create Help documents](#)

TFileSort

TFileSort

[Prev](#)

[Next](#)

Type of the [VideoFromImages_BitmapSortedBy](#) property.

Unit

[VidGrab](#)

Declaration

TFileSort=(fs_TimeAsc, fs_TimeDesc, fs_NameAsc, fs_NameDesc);

enum TFileSort fs_TimeAsc, fs_TimeDesc, fs_NameAsc, fs_NameDesc ;

Description

Value Meaning

fs_TimeAsc the image files are sorted by time, ascending

fs_TimeDesc the image files are sorted by time, descending

fs_NameAsc the image files are sorted by name, ascending

fs_NameDesc the image files are sorted by name, descending

Created with the Standard Edition of HelpNDoc: [Effortlessly upgrade your WinHelp HLP help files to CHM with HelpNDoc](#)

TFormatType

TFormatType

[Prev](#)

[Next](#)

Type of the format returned by the raw sample events.

Description

Type of the format returned by the [OnRawVideoSample](#) and [OnRawAudioSample](#) events.

Possible values:

```
ft_VideoInfo,
ft_VideoInfo2
ft_DvInfo
ft_Mpeg1Video
ft_Mpeg2Video
ft_MpegStreamType
ft_MpegCustom
ft_WaveFormatEx
ft_Unknown
```

Created with the Standard Edition of HelpNDoc: [Free PDF documentation generator](#)

TFrameBitmapInfo

TFrameBitmapInfo

[Prev](#)
[Next](#)

Record type returning the bitmap info of the current video frame

Description

Record type used by the [OnFrameBitmap](#) event.

This record contains information about the current frame bitmap, and information about current/recent mouse positions and states

```
BitmapWidth: LongInt
BitmapHeight: LongInt
BitmapBitsPerPixel: LongInt
BitmapLineSize: LongInt
BitmapSize: LongInt
BitmapPlanes: LongInt
BitmapHandle: THandle           // handle of the frame bitmap DIB
BitmapDataPtr: pByte           // pointer to the bitmap data array
BitmapDC: HDC                  // device context of the frame bitmap
CurrentXMouseLocation: LongInt  // current X position of the mouse
CurrentYMouseLocation: LongInt  // current y position of the mouse
LastXMouseDownLocation: LongInt // last X position of the mouse when the button was clicked
LastYMouseDownLocation: LongInt // last Y position of the mouse when the button was clicked
IsMouseDown: Boolean           // true if the mouse button is currently down
LastMouseButtonClicked: TMouseButton // last mouse button clicked
Reserved0: LongInt
```

Created with the Standard Edition of HelpNDoc: [Easy CHM and documentation editor](#)

TFrameCaptureDest

TFrameCaptureDest

[Prev](#)
[Next](#)

Type used by the [BurstType](#) property and [CaptureFrameTo](#) function.

Unit

[VidGrab](#)

Declaration

TFrameCaptureDest=(fc_TBitmap, fc_BmpFile, fc_JpegFile);

enum TFrameCaptureDest fc_TBitmap, fc_BmpFile, fc_JpegFile ;

Description

Type used by the [BurstType](#) property and [CaptureFrameTo](#) function.

Value Meaning

fc_TBitmap: the frame will be captured to to a memory bitmap (TBitmap in Delphi / C++Builder, HBitmap handle in the OCX versions) and returned by the [OnFrameCaptureCompleted](#) event.

fc_BmpFile: the frame will be saved to a Bmp file, (*)

fc_JpegFile: the frame will be saved to a Jpeg file, (*)

fc_Clipboard: the frame will be saved in the clipboard in CF_BITMAP format, it can be pasted from another application.

(*) in both cases the [OnFrameCaptureCompleted](#) event occurs and returns also the file name of the file created. The file name generated depends of the [AutoFilePrefix](#) and [AutoFileName](#) properties.

See Also

[TAutoFileName](#) [TOnFrameCaptureCompleted](#) [BurstCount](#) [BurstInterval](#) [BurstMode](#) [BurstType](#) [CaptureFrameSyncTo](#) [CaptureFrameTo](#) [FrameCaptureHeight](#) [FrameCaptureWidth](#) [FrameCaptureWithoutOverlay](#) [FrameCaptureZoomSize](#) [GetFrameInfoString](#) [GetLastFrameAsHBITMAP](#) [GetLastFrameAsTBitmap](#) [GetLastFrameBitmapBits](#) [GetLastFrameBitmapBits2](#) [GetLastFrameWaitTimeoutMs](#) [JPEGPerformance](#) [JPEGProgressiveDisplay](#) [JPEGQuality](#) [Last](#) [BurstFrameCapture](#) [FileName](#) [Last](#) [CaptureFrameTo](#) [FileName](#) [OnDiskFull](#) [OnFrameBitmap](#) [SetFrameCaptureBounds](#) [StoragePath](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Create iPhone web-based documentation](#)

TFrameInfo

TFrameInfo

[Prev](#)

[Next](#)

Record type returning the current frame info

Description

Record type used by the [OnFrameBitmap](#) and [OnFrameProgress2](#) events.

This record contains the following information about the current frame:

FrameTime: int64;
 FrameTime_TotalMin: int64;
 FrameTime_TotalSec: int64;
 FrameTime_TotalHs: int64;
 FrameNumber: int64;
 DroppedFrameCount: LongInt;
 FrameTime_Hour: LongInt;
 FrameTime_Min: LongInt;
 FrameTime_Sec: LongInt;
 FrameTime_Hs: LongInt;
 DVTimeCode_IsAvailable: LongInt;
 DVTimeCode_Hour: LongInt;
 DVTimeCode_Min: LongInt;
 DVTimeCode_Sec: LongInt;

```

DVTimeCode_Ff: LongInt;
DVTimeCode_TrackNumber: LongInt;
DVDDateTime_IsAvailable: LongInt;
DVDDateTime_Year: LongInt;
DVDDateTime_Month: LongInt;
DVDDateTime_Day: LongInt;
DVDDateTime_Hour: LongInt;
DVDDateTime_Min: LongInt;
DVDDateTime_Sec: LongInt;
CurrentState: TCurrentState;
GraphState: TGraphState;
PlayerState: TPlayerState;
PlaylistIndex: LongInt;
NTPFrameTime: int64;

```

Created with the Standard Edition of HelpNDoc: [Experience the Power and Simplicity of HelpNDoc's User Interface](#)

TFrameInfoId

TFrameInfoId

[Prev](#)
[Next](#)

Type of the frame info requested. Used with [GetFrameInfo](#).

Unit

[VidGrab](#)

Declaration

```

TFrameInfoId=(fi_FrameNumber, fi_DroppedFrameCount, fi_SampleTime_Hour, fi_SampleTime_Min,
fi_SampleTime_Sec, fi_SampleTime_Hs, fi_SampleTime_TotalMin, fi_DVTimeCode_IsAvailable,
fi_DVTimeCode_Hour, fi_DVTimeCode_Min, fi_DVTimeCode_Sec, fi_DVTimeCode_Ff,
fi_DVTimeCode_TrackNumber, fi_DVDDateTime_IsAvailable, fi_DVDDateTime_Year, fi_DVDDateTime_Month,
fi_DVDDateTime_Day, fi_DVDDateTime_Hour, fi_DVDDateTime_Min, fi_DVDDateTime_Sec);

```

```

enum TFrameInfoId fi_FrameNumber, fi_DroppedFrameCount, fi_SampleTime_Hour, fi_SampleTime_Min,
fi_SampleTime_Sec, fi_SampleTime_Hs, fi_SampleTime_TotalMin, fi_DVTimeCode_IsAvailable,
fi_DVTimeCode_Hour, fi_DVTimeCode_Min, fi_DVTimeCode_Sec, fi_DVTimeCode_Ff,
fi_DVTimeCode_TrackNumber, fi_DVDDateTime_IsAvailable, fi_DVDDateTime_Year, fi_DVDDateTime_Month,
fi_DVDDateTime_Day, fi_DVDDateTime_Hour, fi_DVDDateTime_Min, fi_DVDDateTime_Sec ;

```

Description

Value

```

fi_FrameNumber
fi_DroppedFrameCount
fi_SampleTime_Hour
fi_SampleTime_Min
fi_SampleTime_Sec
fi_SampleTime_Hs
fi_SampleTime_TotalMin
fi_DVTimeCode_IsAvailable
fi_DVTimeCode_Hour
fi_DVTimeCode_Min
fi_DVTimeCode_Sec
fi_DVTimeCode_Ff
fi_DVTimeCode_TrackNumber
fi_DVDDateTime_IsAvailable
fi_DVDDateTime_Year
fi_DVDDateTime_Month
fi_DVDDateTime_Day
fi_DVDDateTime_Hour

```

fi_DVDateTime_Min
 fi_DVDateTime_Sec
 fi_NTPFrameTime

Created with the Standard Edition of HelpNDoc: [Modernize your help files with HelpNDoc's WinHelp HLP to CHM conversion tool](#)

TFrameInfoStringId

TFrameInfoStringId

[Prev](#)
[Next](#)

Type used by the [GetFrameInfoString](#) function

Description

Type used by the [GetFrameInfoString](#) function

Possible values:

```
TFrameInfoStringId = (
  fis_DVTimeCode,
  fis_DVDateTime,
  fis_TimeCode,
  fis_FrameTime,
  fis_FrameNumber,
  fis_FullInfo
);
```

Created with the Standard Edition of HelpNDoc: [Experience the Power and Ease of Use of a Help Authoring Tool](#)

TGPUEncoder

TGPUEncoder

[Prev](#)
[Next](#)

TGPUEncoder

Declaration

```
TGPUEncoder=(
  Enc_GPU_None,
  Enc_GPU_Auto,
  Enc_GPU_Intel_QSV,
  Enc_GPU_NVidia_NVENC,
  Enc_GPU_AMD_AMF
```

```
enum TGPUEncoder Enc_GPU_None, Enc_GPU_Auto, Enc_GPU_Intel_QSV, Enc_GPU_NVidia_NVENC,
Enc_GPU_AMD_AMF ;
```

Description

The TGPUEncoder type is used by the [Datastead Encoder](#)

Created with the Standard Edition of HelpNDoc: [Free Qt Help documentation generator](#)

TGraphState

TGraphState

[Prev](#)
[Next](#)

Type returned by the [GraphState](#) function.

Declaration

TGraphState=(gs_Stopped, gs_Paused, gs_Running);

enum TGraphState gs_Stopped, gs_Paused, gs_Running ;

Description

Type returned by the [GraphState](#) function.

Possible values:

gs_Stopped: the graph is in a stopped state

gs_Paused: the graph is in a paused state

gs_Running: the graph is in a running state

Created with the Standard Edition of HelpNDoc: [Converting Word Documents to eBooks: A Step-by-Step Guide with HelpNDoc](#)

THeaderAttribute

THeaderAttribute

[Prev](#)
[Next](#)

Type of the AVI or ASF attribute that can be set by invoking [SetHeaderAttribute](#).

Description

See [SetHeaderAttribute](#) for the list of possible values.

See Also

[ClearHeaderAttributes](#) [SetHeaderAttribute](#)

Created with the Standard Edition of HelpNDoc: [Easy to use tool to create HTML Help files and Help web sites](#)

THwAccel

THwAccel

[Prev](#)
[Next](#)

Type of the [PlayerHwAccel](#) property

Description

Possible values:

hw_None: no hardware acceleration

hw_Cuda: NVidia CUDA

hw_QuickSync:/b0 Intel QuickSync

hw_Dxva2: DirectX dxva2

hw_d3d11: DirectX d3d11

Created with the Standard Edition of HelpNDoc: [Write EPub books for the iPad](#)

TIPCameraSetting

TIPCameraSetting

[Prev](#)
[Next](#)

TIPCameraSetting

Declaration

TIPCameraSetting=(ips_ConnectionTimeOut, ips_ReceiveTimeOut);

enum TIPCameraSetting ips_ConnectionTimeOut, ips_ReceiveTimeOut);

Description

Type used by the [SetIPCameraSetting](#) method.

Created with the Standard Edition of HelpNDoc: [Import and export Markdown documents](#)

TLogoLayout

TLogoLayout

[Prev](#)

[Next](#)

Type of the [LogoLayout](#) property.

Unit

[VidGrab](#)

Declaration

TLogoLayout=(lg_Centered, lg_Stretched, lg_Repeated, lg_TopLeft, lg_TopRight, lg_BottomLeft, lg_BottomRight);

enum TLogoLayout lg_Centered, lg_Stretched, lg_Repeated, lg_TopLeft, lg_TopRight, lg_BottomLeft, lg_BottomRight ;

Description

Value Meaning

lg_Centered the logo is centered "as is" in the middle of the video window

lg_Stretched the logo is stretched to fit the video window

lg_Repeated the logo is repeated over the video window, until filled out

lg_TopLeft the logo is aligned on the top left of the control

lg_TopRight the logo is aligned on the top right of the control

lg_BottomLeft the logo is aligned on the bottom left of the control

lg_BottomRight the logo is aligned on the bottom right of the control

lg_Boxed the logo is boxed to preserve its aspect ratio. Depending of its aspect ratio, borders are added on the top and bottom, or on the left and right (the border color is [BackgroundColor](#))

Created with the Standard Edition of HelpNDoc: [Revolutionize Your Documentation Output with a Help Authoring Tool](#)

TLogType

TLogType

[Prev](#)

[Next](#)

Constants returned by the [OnLog](#) event.

Unit

[VidGrab](#)

Declaration

TLogType=(e_add_filter, e_add_source_filter, e_audio_compressor_not_suitable, e_bind_moniker_to_filter, e_compressor_possibly_not_suitable, e_create_instance, e_ddraw_caps_not_suitable, e_device_in_use_in_another_graph, e_disk_full, e_failed, e_failed_to_allocate_recording_file, e_failed_to_bind_codec, e_failed_to_connect_crossbar_pin, e_failed_to_connect_to_server, e_failed_to_create_directory, e_failed_to_create_file, e_failed_to_create_temp, e_failed_to_bind_frame_grabber, e_failed_to_load_ASF_profile, e_failed_to_load_ASF_profile_custom_file, e_failed_to_load_set_of_bitmaps, e_failed_to_set_image_overlay, e_failed_to_set_logo, e_failed_to_play_backwards, e_failed_to_render_file, e_failed_to_renew_recording_file,

e_failed_to_set_player_speed_ratio_with_audio, e_failed_to_setup_network_streaming,
e_failed_to_start_preview, e_failed_to_start_recording, e_file_in_use, e_file_name_not_specified,
e_file_not_found, e_get_audio_format, e_get_interface, e_get_video_format, e_graph_error,
e_graph_cant_run, e_graph_must_be_restarted, e_hw_deinterlace_not_supported, e_incompatible_options,
e_index_out_of_range, e_invalid_directory, e_library_not_found, e_load_filter, e_no_audio_input_device,
e_no_device_available, e_no_dialog_for_this_compressor, e_no_stream_control, e_no_tv_tuner,
e_no_video_device_selected, e_no_video_input_device, e_not_allowed_during_network_streaming,
e_not_assigned, e_not_multiplexed_master, e_not_previewing, e_not_recording, e_not_reencoding,
e_not_streaming, e_out_of_memory, e_pause_resume_disabled, e_pin_not_found,
e_interface_not_assigned, e_query_config_avi_mux, e_reencoding, e_recording_cannot_pause,
e_render_audio_stream, e_render_video_stream, e_must_restart_master, e_recording_on_motion_failed,
e_sendtdv_device_index_out_of_bound, e_sendtdv_deviceindex_and_videodevice_have_same_value,
e_sendtdv_failed_to_bind_dv_device, e_set_filter_graph, e_set_interleaving_mode, e_set_master_stream,
e_set_output_compatibility_index, e_set_output_file_name, e_set_format, e_start_preview_first,
e_stop_player_first, e_stop_preview_first, e_stop_recording_first, e_stop_reencoding_first,
e_storage_path_read_only, e_streaming_type_not_specified, e_third_party_filter_already_inserted,
e_third_party_filter_error, e_trace_log, e_tv_command_not_allowed_during_tv_tuning,
e_tuner_input_not_selected, e_TVideoGrabber_Filter_obsolete, e_value_out_of_range,
e_video_compressor_not_suitable, e_window_transparency_failed,
i_audio_device_associated_to_video_device, i_begin_discovering_device,
i_binding_device_or_compressor, i_discovering_device, i_end_discovering_device,
i_preallocated_file_size_large_enough, i_preallocated_file_size_changed, i_preallocated_file_not_suitable,
i_streaming_to_publishing_point, i_third_party_filter_inserted, i_using_ASF_Profile,
i_recording_videosubtype, i_ismpegstream, i_new_recording_filename, i_using_property_group,
w_cannot_connect_thirdparty_filter, w_cannot_connect_thirdparty_renderer,
w_cannot_instantiate_thirdparty_filter, w_cannot_route_audio_crossbar, w_cannot_use_color_key,
w_command_delayed, w_does_not_apply_to_dv, w_find_audio_device, w_filter_does_not_save_properties,
w_frame_grabber_requires_CPU, w_hold_recording, w_information, w_not_playing,
w_player_audio_should_be_disabled, w_recording_cancelled_by_user,
w_can_pause_and_ASF_incompatible, w_set_audio_format, w_storage_path_on_network, w_tv_tuner,
w_using_nearest_video_size, w_divx_codec_not_installed,
w_intervideo_codec_does_not_support_debugger, w_should_install_divx_codec,
w_device_partially_supported);

enum TLogType e_add_filter, e_add_source_filter, e_audio_compressor_not_suitable,
e_bind_moniker_to_filter, e_compressor_possibly_not_suitable, e_create_instance,
e_ddraw_caps_not_suitable, e_device_in_use_in_another_graph, e_disk_full, e_failed,
e_failed_to_allocate_recording_file, e_failed_to_bind_codec, e_failed_to_connect_crossbar_pin,
e_failed_to_connect_to_server, e_failed_to_create_directory, e_failed_to_create_file,
e_failed_to_create_temp, e_failed_to_bind_frame_grabber, e_failed_to_load_ASF_profile,
e_failed_to_load_ASF_profile_custom_file, e_failed_to_load_set_of_bitmaps,
e_failed_to_set_image_overlay, e_failed_to_set_logo, e_failed_to_play_backwards, e_failed_to_render_file,
e_failed_to_renew_recording_file, e_failed_to_set_player_speed_ratio_with_audio,
e_failed_to_setup_network_streaming, e_failed_to_start_preview, e_failed_to_start_recording,
e_file_in_use, e_file_name_not_specified, e_file_not_found, e_get_audio_format, e_get_interface,
e_get_video_format, e_graph_error, e_graph_cant_run, e_graph_must_be_restarted,
e_hw_deinterlace_not_supported, e_incompatible_options, e_index_out_of_range, e_invalid_directory,
e_library_not_found, e_load_filter, e_no_audio_input_device, e_no_device_available,
e_no_dialog_for_this_compressor, e_no_stream_control, e_no_tv_tuner, e_no_video_device_selected,
e_no_video_input_device, e_not_allowed_during_network_streaming, e_not_assigned,
e_not_multiplexed_master, e_not_previewing, e_not_recording, e_not_reencoding, e_not_streaming,
e_out_of_memory, e_pause_resume_disabled, e_pin_not_found, e_interface_not_assigned,
e_query_config_avi_mux, e_reencoding, e_recording_cannot_pause, e_render_audio_stream,
e_render_video_stream, e_must_restart_master, e_recording_on_motion_failed,
e_sendtdv_device_index_out_of_bound, e_sendtdv_deviceindex_and_videodevice_have_same_value,
e_sendtdv_failed_to_bind_dv_device, e_set_filter_graph, e_set_interleaving_mode, e_set_master_stream,
e_set_output_compatibility_index, e_set_output_file_name, e_set_format, e_start_preview_first,
e_stop_player_first, e_stop_preview_first, e_stop_recording_first, e_stop_reencoding_first,
e_storage_path_read_only, e_streaming_type_not_specified, e_third_party_filter_already_inserted,
e_third_party_filter_error, e_trace_log, e_tv_command_not_allowed_during_tv_tuning,
e_tuner_input_not_selected, e_TVideoGrabber_Filter_obsolete, e_value_out_of_range,
e_video_compressor_not_suitable, e_window_transparency_failed,

i_audio_device_associated_to_video_device, i_begin_discovering_device,
i_binding_device_or_compressor, i_discovering_device, i_end_discovering_device,
i_preallocated_file_size_large_enough, i_preallocated_file_size_changed, i_preallocated_file_not_suitable,
i_streaming_to_publishing_point, i_third_party_filter_inserted, i_using_ASF_Profile,
i_recording_videosubtype, i_ismpegstream, i_new_recording_filename, i_using_property_group,
w_cannot_connect_thirdparty_filter, w_cannot_connect_thirdparty_renderer,
w_cannot_instantiate_thirdparty_filter, w_cannot_route_audio_crossbar, w_cannot_use_color_key,
w_command_delayed, w_does_not_apply_to_dv, w_find_audio_device, w_filter_does_not_save_properties,
w_frame_grabber_requires_CPU, w_hold_recording, w_information, w_not_playing,
w_player_audio_should_be_disabled, w_recording_cancelled_by_user,
w_can_pause_and_ASF_incompatible, w_set_audio_format, w_storage_path_on_network, w_tv_tuner,
w_using_nearest_video_size, w_divx_codec_not_installed,
w_intervideo_codec_does_not_support_debugger, w_should_install_divx_codec,
w_device_partially_supported ;

Description

Possible constants returned by the [OnLog](#) event:

e_add_filter
e_add_source_filter
e_audio_compressor_not_suitable
e_bind_moniker_to_filter
e_compressor_possibly_not_suitable
e_create_instance
e_ddraw_caps_not_suitable
e_device_in_use_in_another_graph
e_disk_full
e_failed
e_failed_to_allocate_recording_file
e_failed_to_bind_codec
e_failed_to_connect_crossbar_pin
e_failed_to_connect_to_server
e_failed_to_create_directory
e_failed_to_create_file
e_failed_to_create_temp
e_failed_to_bind_frame_grabber
e_failed_to_load_ASF_profile
e_failed_to_load_ASF_profile_custom_file
e_failed_to_load_set_of_bitmaps
e_failed_to_set_image_overlay
e_failed_to_set_logo
e_failed_to_play_backwards
e_failed_to_render_file
e_failed_to_renew_recording_file
e_failed_to_set_player_speed_ratio_with_audio
e_failed_to_setup_network_streaming
e_failed_to_start_preview
e_failed_to_start_recording
e_file_in_use
e_file_name_not_specified
e_file_not_found
e_get_audio_format
e_get_interface
e_get_video_format
e_graph_error
e_graph_cant_run
e_graph_must_be_restarted
e_hw_deinterlace_not_supported
e_incompatible_options
e_index_out_of_range
e_invalid_directory
e_library_not_found
e_load_filter
e_no_audio_in_device

e_no_device_available
 e_no_dialog
 e_no_stream_control
 e_no_tv_tuner
 e_no_device_selected
 e_no_video_input_device
 e_not_allowed_during_network_streaming
 e_not_allowed_with_streaming_URL
 e_not_assigned
 e_not_multiplexed_master
 e_not_previewing
 e_not_recording
 e_not_reencoding
 e_not_streaming
 e_out_of_memory
 e_pause_resume_disabled
 e_pin_not_found
 e_interface_not_assigned
 e_query_config_avi_mux
 e_reencoding
 e_recording_cannot_pause
 e_render_audio_stream
 e_render_video_stream
 e_must_restart_master
 e_recording_on_motion_failed
 e_sendtdv_device_index_out_of_bound
 e_sendtdv_deviceindex_and_videodevice_have_same_value
 e_sendtdv_failed_to_bind_dv_device
 e_set_filter_graph
 e_set_interleaving_mode
 e_set_master_stream
 e_set_output_compatibility_index
 e_set_output_file_name
 e_set_format
 e_start_preview_first
 e_stop_player_first
 e_stop_preview_first
 e_stop_recording_first
 e_stop_reencoding_first
 e_storage_path_read_only
 e_streaming_type_not_specified
 e_third_party_filter_already_inserted
 e_third_party_filter_error
 e_trace_log
 e_tv_command_not_allowed_during_tv_tuning
 e_tuner_input_not_selected
 e_TVideoGrabber_Filter_obsolete
 e_value_out_of_range
 e_video_compressor_not_suitable
 e_window_transparency_failed
 e_invalid_size
 e_invalid_window_handle
 e_tuner_mode_not_supported
 e_publishing_point_connection_failed
 e_speaker_control_disabled
 i_audio_device_associated_to_video_device
 i_begin_discovering_device
 i_binding_device_or_compressor
 i_discovering_device
 i_end_discovering_device
 i_preallocated_file_size_large_enough
 i_preallocated_file_size_changed
 i_preallocated_file_not_suitable

i_streaming_to_publishing_point
i_third_party_filter_inserted
i_using_ASF_Profile
i_recording_videosubtype
i_ismpegstream
i_new_recording_filename
i_using_property_group
i_streaming_client_connected
i_streaming_client_disconnected
i_refreshing_preview
i_recording_on_motion
i_window_found
i_limiting_preview
i_codec_recommended
i_tuner_mode
i_DV_date_time_discontinuity
w_cannot_connect_thirdparty_filter
w_cannot_connect_thirdparty_renderer
w_cannot_instantiate_thirdparty_filter
w_cannot_route_crossbar
w_cannot_use_color_key
w_command_delayed
w_does_not_apply_to_dv
w_find_audio_device
w_filter_does_not_save_properties
w_frame_grabber_requires_CPU
w_hold_recording
w_information
w_not_playing
w_player_audio_should_be_disabled
w_recording_cancelled_by_user
w_can_pause_and_ASF_incompatible
w_set_audio_format
w_storage_path_on_network
w_tv_tuner
w_using_nearest_video_size
w_divx_codec_not_installed
w_codec_does_not_support_debugger
w_divx_codec_profile
w_device_partially_supported
w_excessive_grid_size
w_grid_too_large_for_dialog
w_operation_may_lock
w_audio_streaming_needs_audiorecording_property_enabled
w_network_streaming_disabled
w_server_lost_next_retry
w_overlay_mixer_not_available
w_network_streaming_change_requires_application_to_be_restarted
w_standard_renderer_recommended
w_window_transparency_and_recording_not_recommended
w_clip_not_seekable
w_only_WMV_recording_during_network_streaming
w_check_analog_video_standard
w_recording_timer_set
w_stream_time_beyond_script_time
w_generate_new_file
w_hires_timer_not_available
w_applies_to_the_current_recording_method
i_leaving_full_screen_mode
i_stream_info
i_async_url_connection_in_progress
i_async_url_connection_cancelled
e_obsolete

i_codec_info
 i_preview_started
 i_recording_started
 i_reencoding_started
 i_recording_completed
 i_reencoding_completed
 i_player_opened
 i_inactive
 i_using_stream_index
 e_failed_to_start_reencoding
 e_recording_failed
 e_failed_to_open_player

Created with the Standard Edition of HelpNDoc: [iPhone web sites made easy](#)

TMiscDeviceControl

TMiscDeviceControl

[Prev](#)
[Next](#)

Type used by the [PutMiscDeviceData](#) and [GetMiscDeviceData](#) functions.

Description

Type used by the [PutMiscDeviceData](#) and [GetMiscDeviceData](#) functions.
Possible values:

mdc_GPIO: used to set/retrieve the GPIO state

mdc_VPD: used to set/retrieve the VPD (Virtual Private Data) state

mdc_VPD_Data: used to set/retrieve the VPD values.

Created with the Standard Edition of HelpNDoc: [Experience the Power and Ease of Use of HelpNDoc for CHM Help File Generation](#)

TMouseButton

TMouseButton

[Prev](#)
[Next](#)

TMouseButton type

Description

TMouseButton type

Possible values:

mbLeft

mbRight

mbMiddle

Created with the Standard Edition of HelpNDoc: [Transform Your Documentation Process with HelpNDoc's Project Analyzer](#)

TMPEGProgramSetting

TMPEGProgramSetting

[Prev](#)
[Next](#)

TMPEGProgramSetting type

Declaration

TMPEGProgramSetting=(mps_Program_Number, mps_Program_PCR_PID, mps_VideoStream_PID, mps_AudioStream_PID);

enum TMPEGProgramSetting mps_Program_Number, mps_Program_PCR_PID, mps_VideoStream_PID, mps_AudioStream_PID

Description

TMPEGProgramSetting type

mps_Program_Number
mps_Program_PCR_PID
mps_VideoStream_PID
mps_AudioStream_PID

Created with the Standard Edition of HelpNDoc: [Maximize Your PDF Protection with These Simple Steps](#)

TMpegStreamType

TMpegStreamType

[Prev](#)
[Next](#)

Type of the Mpeg stream.

Unit

[VidGrab](#)

Declaration

TMpegStreamType=(mpst_Default, mpst_Program, mpst_Program_DVD, mpst_Program_DVD_MC, mpst_Program_SVCD, mpst_MPEG1, mpst_MPEG1_VCD);

enum TMpegStreamType mpst_Default, mpst_Program, mpst_Program_DVD, mpst_Program_DVD_MC, mpst_Program_SVCD, mpst_MPEG1, mpst_MPEG1_VCD ;

Description

Type of the Mpeg stream.

Possible values:

mpst_Unknown
mpst_Program
mpst_Program_DVD
mpst_Program_DVD_MC
mpst_Program_SVCD
mpst_MPEG1
mpst_MPEG1_VCD

Created with the Standard Edition of HelpNDoc: [Easily create Web Help sites](#)

TMultiplexedRole

TMultiplexedRole

[Prev](#)
[Next](#)

Type of the [MultiplexedRole](#) property.

Unit

[VidGrab](#)

Declaration

TMultiplexedRole=(mr_NotMultiplexed, mr_MultiplexedMosaic4, mr_MultiplexedMosaic16, mr_MultiplexedMaster, mr_MultiplexedSlave);

enum TMultiplexedRole mr_NotMultiplexed, mr_MultiplexedMosaic4, mr_MultiplexedMosaic16,

mr_MultiplexedMaster, mr_MultiplexedSlave ;

Description

Value Meaning

mr_NotMultiplexed
mr_MultiplexedMosaic4
mr_MultiplexedMosaic16
mr_MultiplexedMaster
mr_MultiplexedSlave

See Also

[AssociateMultiplexedSlave](#) [EnableMultiplexedInput](#) [MultiplexedInputEmulation](#) [MultiplexedRole](#)
[MultiplexedStabilizationDelay](#) [MultiplexedSwitchDelay](#) [UniqueID](#)

Created with the Standard Edition of HelpNDoc: [Write EPub books for the iPad](#)

TNDIFormatType

TNDIFormatType

[Prev](#)
[Next](#)

Type of NDI streaming format

Declaration

TNDIFormatType=(nft_interleaved, nft_progressive, nft_field_0, nft_field_1);

enum TNDIFormatType: **unsigned int** nft_interleaved, nft_progressive, nft_field_0, nft_field_1 ;

Description

Possible values:

nft_interleaved
nft_progressive
nft_field_0
nft_field_1

Default:

nft_progressive

See Also

[NDIFormatType](#) [NDIName](#) [NetworkStreaming](#) [SetDatasteadFilterDllName](#)

Created with the Standard Edition of HelpNDoc: [Transform Your Documentation Process with HelpNDoc's Project Analyzer](#)

TNetworkStreaming

TNetworkStreaming

[Prev](#)
[Next](#)

Type of the [NetworkStreaming](#) property.

Unit

[VidGrab](#)

Declaration

TNetworkStreaming=(ns_Disabled, ns_ASFDirectNetworkStreaming, ns_ASFStreamingToPublishingPoint);

enum TNetworkStreaming ns_Disabled, ns_ASFDirectNetworkStreaming,
ns_ASFStreamingToPublishingPoint ;

Description

Value Meaning

ns_Disabled network streaming disabled

ns_ASFDirectNetworkStreaming direct network streaming from the platform running TVideoGrabber

ns_ASFStreamingToPublishingPoint TVideoGrabber sends the streaming media to the Windows Media Server specified by [ASFMediaServerPublishingPoint](#).

See Also

[TNetworkStreamingType](#) [TOnClientConnection](#) [TOnDirectNetworkStreamingHostUrl](#) [ASFAudioBitRate](#) [ASFAudioChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#) [ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#) [ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#) [ASFProfileFromCustomFile](#) [ASFProfiles](#) [ASFProfilesCount](#) [ASFProfileVersion](#) [ASFVideoBitRate](#) [ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#) [NetworkStreaming](#) [NetworkStreamingType](#) [OnAuthenticationNeeded](#) [OnClientConnection](#) [OnDirectNetworkStreamingHostUrl](#) [ShowDialog](#) [StartAudioRendering](#) [StreamingURL](#)

Created with the Standard Edition of HelpNDoc: [Easily create Web Help sites](#)

TNetworkStreamingType

TNetworkStreamingType

[Prev](#)

[Next](#)

Type of the [NetworkStreamingType](#) property.

Declaration

```
TNetworkStreamingType=(nst_AudioVideoStreaming, nst_VideoStreaming, nst_AudioStreaming);
```

```
enum TNetworkStreamingType _AudioVideoStreaming, nst_VideoStreaming, nst_AudioStreaming
```

Description

Values:

nst_AudioVideoStreaming : audio + video streaming

nst_VideoStreaming : video streaming only

nst_AudioStreaming : audio streaming only

See Also

[TOnClientConnection](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#) [ASFAudioBitRate](#) [ASFAudioChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#) [ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#) [ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#) [ASFProfileFromCustomFile](#) [ASFProfiles](#) [ASFProfilesCount](#) [ASFProfileVersion](#) [ASFVideoBitRate](#) [ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#) [NetworkStreaming](#) [NetworkStreamingType](#) [OnAuthenticationNeeded](#) [OnClientConnection](#) [OnDirectNetworkStreamingHostUrl](#) [ShowDialog](#) [StartAudioRendering](#) [StreamingURL](#)

Created with the Standard Edition of HelpNDoc: [Quickly and Easily Convert Your Word Document to an ePub or Kindle eBook](#)

TNotificationMethod

TNotificationMethod

[Prev](#)

[Next](#)

Type of [NotificationMethod](#) property.

Unit

[VidGrab](#)**Declaration**

TNotificationMethod=(nm_Timer, nm_Thread);

enum TNotificationMethod nm_Timer, nm_Thread ;

Description**Value Meaning**

nm_Timer notification by timer

nm_Thread notification by thread

Created with the Standard Edition of HelpNDoc: [Step-by-Step Guide: How to Turn Your Word Document into an eBook](#)

TOnAudioBufferNegotiation**TOnAudioBufferNegotiation**[Prev](#)[Next](#)

TOnAudioBufferNegotiation

Declaration

TOnAudioBufferNegotiation=procedure (Sender: TObject; nChannels: LongInt; nSamplesPerSec: LongInt; BitsPerSample: LongInt; var ProposedBufferSize: LongInt);

Description

Type of the [OnAudioBufferNegotiation](#) event

Created with the Standard Edition of HelpNDoc: [iPhone web sites made easy](#)

TOnAudioPeak**TOnAudioPeak**[Prev](#)[Next](#)

Type of the [OnAudioPeak](#) event.

Declaration

TOnAudioPeak=**procedure** (Sender: TObject; Left_Percent: Double; Left_DB: Double; Right_Percent: Double; Right_DB: Double) **of object**;

typedef void __fastcall(__closure *TOnAudioPeak)(System::TObject *Sender, double Left_Percent, Left_DB, Right_Percent, Right_DB);

Description

Type of the [OnAudioPeak](#) event.

Created with the Standard Edition of HelpNDoc: [Effortlessly Edit and Export Markdown Documents](#)

TOnAuthenticationNeeded**TOnAuthenticationNeeded**[Prev](#)[Next](#)

Type of the [OnAuthenticationNeeded](#) event.

Declaration

TOnAuthenticationNeeded=**procedure** (Sender: TObject; Id: LongInt; Realm: **String**; Server: **string**; var

Username: **string**; var Password: **string**);

```
typedef void __fastcall (__closure *TOnAuthenticationNeeded)(System::TObject* Sender, int Id, wchar_t *Realm, wchar_t *Server, wchar_t *&Username, wchar_t *&Password);
```

Description

Type of the [OnAuthenticationNeeded](#) event.

Created with the Standard Edition of HelpNDoc: [Easily create EPub books](#)

TOnAVIDurationUpdated

TOnAVIDurationUpdated

[Prev](#)
[Next](#)

Type of the [OnEndOfAVIRecording](#) event.

Declaration

TOnAVIDurationUpdated=**procedure**(Sender: TObject; FileName: **string**; FrameCount: LargeInteger; var FrameRate: Double; var Duration: LargeInteger) **of object**;

```
typedef void __fastcall(__closure *TOnAVIDurationUpdated)(System::TObject *Sender, wchar_t *FileName, __int64 FrameCount, double &FrameRate, __int64 &Duration);
```

Description

Type of the [OnEndOfAVIRecording](#) event.

Created with the Standard Edition of HelpNDoc: [Easily create PDF Help documents](#)

TOnClientConnection

TOnClientConnection

[Prev](#)
[Next](#)

Type of the [OnClientConnection](#) event.

Declaration

TOnClientConnection=**procedure** (Sender: TObject; Connected: Boolean; ClientInfo: **String**)**of object**;

```
typedef void __fastcall (__closure *TOnClientConnection)(System::TObject* Sender, bool Connected, wchar_t *ClientInfo);
```

Description

Type of the [OnClientConnection](#) event.

See Also

[TNetworkStreamingType](#) [TNetworkStreaming](#) [TOnDirectNetworkStreamingHostUrl](#) [ASFVideoBitRate](#) [ASFVideoChannels](#) [ASFDeinterlaceMode](#) [ASFFixedFrameRate](#) [ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#) [ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#) [ASFProfileFromCustomFile](#) [ASFProfiles](#) [ASFProfilesCount](#) [ASFProfileVersion](#) [ASFVideoBitRate](#) [ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#) [NetworkStreaming](#) [NetworkStreamingType](#) [OnAuthenticationNeeded](#) [OnClientConnection](#) [OnDirectNetworkStreamingHostUrl](#) [ShowDialog](#) [StartAudioRendering](#) [StreamingURL](#)

Created with the Standard Edition of HelpNDoc: [Make Your PDFs More Secure with Encryption and Password Protection](#)

TOnColorKeyChange

TOnColorKeyChange

[Prev](#)
[Next](#)

The type of the [OnColorKeyChange](#) event.

Unit

[VidGrab](#)

Declaration

TOnColorKeyChange=**procedure**(Sender: TObject; ColorKey: TColor) **of object**;

```
typedef void __fastcall(__closure *TOnColorKeyChange)(System::TObject *Sender, Graphics::TColor ColorKey);
```

Description

The type of the [OnColorKeyChange](#) event.

ColorKey: new color key used for window transparency.

See Also

[ColorKeyEnabled](#) [OnColorKeyChange](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly Support Your Windows Applications with HelpNDoc's CHM Generation](#)

TOnCreatePreallocatedFileCompleted

TOnCreatePreallocatedFileCompleted

[Prev](#)
[Next](#)

Type of the [OnCreatePreallocatedFileCompleted](#) event.

Unit

[VidGrab](#)

Declaration

TOnCreatePreallocatedFileCompleted=**procedure**(Sender: TObject; FileName: **String**; Success: Boolean) **of object**;

```
typedef void __fastcall(__closure *TOnCreatePreallocatedFileCompleted)(System::TObject *Sender, wchar_t *FileName, bool Success);
```

Description

Type of the [OnCreatePreallocatedFileCompleted](#) event.

See Also

[OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#)
[OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileProgress](#) [OnCreatePreallocFileStarted](#)

Created with the Standard Edition of HelpNDoc: [Say Goodbye to Documentation Headaches with a Help Authoring Tool](#)

TOnDeviceArrivalOrRemoval

TOnDeviceArrivalOrRemoval

[Prev](#)
[Next](#)

The type of the [OnDeviceArrivalOrRemoval](#) event.

Unit

[VidGrab](#)

Declaration

TOnDeviceArrivalOrRemoval=**procedure**(Sender: TObject; IsDeviceArrival: Boolean; IsVideoDevice: Boolean; DeviceName: **string**; DeviceIndex: LongInt) **of object**;

```
typedef void __fastcall(__closure *TOnDeviceArrivalOrRemoval)(System::TObject *Sender, bool IsDeviceArrival, bool IsVideoDevice, wchar_t *DeviceName, int DeviceIndex);
```

Description

The type of the [OnDeviceArrivalOrRemoval](#) event.

IsDeviceArrival: true if the device has been connected, false if the device has been disconnected

IsVideoDevice: true if the device is a video capture device, false if the device is an audio capture device.

DeviceName: name of the related capture device.

DeviceIndex: index of the capture device in the [VideoDevices](#) list or [AudioDevices](#) list, according to IsVideoDevice.

See Also

[OnDeviceArrivalOrRemoval](#) [OnDeviceLost](#) [RefreshDevicesAndCompressorsLists](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly bring your documentation online with HelpNDoc](#)

TOnDirectNetworkStreamingHostUrl

TOnDirectNetworkStreamingHostUrl

[Prev](#)

[Next](#)

Type of the [OnDirectNetworkStreamingHostUrl](#) event.

Unit

[VidGrab](#)

Declaration

TOnDirectNetworkStreamingHostUrl=**procedure**(Sender: TObject; HostUrl: **String**; HostName: **String**; HostPort: LongInt) **of object**;

```
typedef void __fastcall(__closure *TOnDirectNetworkStreamingHostUrl)(System::TObject *Sender, wchar_t *HostUrl, wchar_t *HostName, int HostPort);
```

Description

Type of the [OnDirectNetworkStreamingHostUrl](#) event.

HostUrl: the full qualified URL on which users must connect to get live streaming on the current platform

HostName: host name on the current platform

HostPort: streaming port on the current platform

See Also

[TNetworkStreamingType](#) [TOnClientConnection](#) [TNetworkStreaming](#) [ASFVideoBitRate](#) [ASFVideoChannels](#) [ASFDeinterlaceMode](#) [ASFVideoFrameRate](#) [ASFMediaServerPublishingPoint](#) [ASFMediaServerRemovePublishingPointAfterDisconnect](#) [ASFMediaServerTemplatePublishingPoint](#) [ASFNetworkMaxUsers](#) [ASFNetworkPort](#) [ASFProfile](#) [ASFProfileFromCustomFile](#) [ASFProfiles](#) [ASFProfilesCount](#) [ASFProfileVersion](#) [ASFVideoBitRate](#) [ASFVideoFrameRate](#) [ASFVideoHeight](#) [ASFVideoMaxKeyFrameSpacing](#) [ASFVideoQuality](#) [ASFVideoWidth](#) [NetworkStreaming](#)

[NetworkStreamingType](#) [OnAuthenticationNeeded](#) [OnClientConnection](#) [OnDirectNetworkStreamingHostUrl](#) [ShowDialog](#) [StartAudioRendering](#) [StreamingURL](#)

Created with the Standard Edition of HelpNDoc: [5 Reasons Why a Help Authoring Tool is Better than Microsoft Word for Documentation](#)

TOnDragDropFiles

TOnDragDropFiles

[Prev](#)

[Next](#)

TOnDragDropFiles

Declaration

TOnDragDropFiles=**procedure** (Sender: Pointer; FileCount: LongInt; FirstFile: **string**; FilesList: **string**);

Description

Type of the [OnDragDropFiles](#) event

Created with the Standard Edition of HelpNDoc: [Step-by-Step Guide: How to Turn Your Word Document into an eBook](#)

TOnDVCommandCompleted

TOnDVCommandCompleted

[Prev](#)

[Next](#)

Type of the [OnDVCommandCompleted](#) event.

Unit

[VidGrab](#)

Declaration

TOnDVCommandCompleted=**procedure**(Sender: TObject; NewStateValue: LongInt; NewStateLabel: **String**)of **object**;

typedef void __fastcall(__closure *TOnDVCommandCompleted)(System::TObject *Sender, **int** NewStateValue, wchar_t *NewStateLabel);

Description

Type of the [OnDVCommandCompleted](#) event.

See Also

[TDVSize](#) [TDVVideoFormat](#) [TDVVideoStandard](#) [DVDateTimeEnabled](#) [DVDDiscontinuityMinimumInterval](#) [DVRecordingInNativeFormatSeparatesStreams](#) [DVReduceFrameRate](#) [DVRgb219](#) [DVTimeCodeEnabled](#) [IsDigitalVideoIn](#) [IsDVCommandAvailable](#) [IsTimeCodeReaderAvailable](#) [OnDVCommandCompleted](#) [SendDVCommand](#)

Created with the Standard Edition of HelpNDoc: [Streamline Your Documentation Process with HelpNDoc's Intuitive Interface](#)

TOnDVDDiscontinuity

TOnDVDDiscontinuity

[Prev](#)

[Next](#)

Type of the [OnDVDDiscontinuity](#) event.

Declaration

TOnDVDDiscontinuity=**procedure**(Sender: TObject; var DeliverNewFrame: Boolean)**of object**;

```
typedef void __fastcall(__closure *TOnDVDDiscontinuity)(System::TObject *Sender, bool
*DeliverNewFrame);
```

Description

Type of the [OnDVDDiscontinuity](#) event.

Created with the Standard Edition of HelpNDoc: [Maximize Your Documentation Capabilities with HelpNDoc's Project Analyzer](#)

TOnEnumerateWindows

TOnEnumerateWindows

[Prev](#)

[Next](#)

Type of the [OnEnumerateWindows](#) event.

Declaration

TOnEnumerateWindows=**procedure** (Sender: TObject; WindowHandle: LongInt; WindowName: String; WindowClass: String) **of object**;

```
typedef void __fastcall (__closure *TOnEnumerateWindows)(System::TObject* Sender, int
WindowHandle, wchar_t *WindowName, wchar_t *WindowClass);
```

Description

Type of the [OnEnumerateWindows](#) event.

See Also

[EnumerateWindows](#) [ScreenRecordingLayeredWindows](#) [ScreenRecordingMonitor](#)
[ScreenRecordingNonVisibleWindows](#) [ScreenRecordingUsingCoordinates](#) [ScreenRecordingWithCursor](#)
[SetWindowRecordingByHandle](#) [SetWindowRecordingByName](#)

Created with the Standard Edition of HelpNDoc: [Create HTML Help, DOC, PDF and print manuals from 1 single source](#)

TOnFileNotification

TOnFileNotification

[Prev](#)

[Next](#)

Type of the [OnRecordingStarted](#) and the [OnCreatePreallocFileStarted](#) events.

Unit

[VidGrab](#)

Declaration

TOnFileNotification=**procedure**(Sender: TObject; FileName: **String**)**of object**;

```
typedef void __fastcall(__closure *TOnFileNotification)(System::TObject *Sender, wchar_t *FileName);
```

Description

Type of the [OnRecordingStarted](#) and the [OnCreatePreallocFileStarted](#) events.

Created with the Standard Edition of HelpNDoc: [Make the switch to CHM with HelpNDoc's hassle-free WinHelp HLP to CHM conversion tool](#)

TONFilterSelected

TONFilterSelected

[Prev](#)
[Next](#)

Type of the OnFilterSelected event

Declaration

TONFilterSelected=**procedure** (Sender: TObject; FilterName: **string**; **var** RejectFilter: Boolean)

typedef void __fastcall(__closure *TONFilterSelected)(System::TObject *Sender, wchar_t *FilterName, **bool** &RejectFilter)

Description

Type of the OnFilterSelected event

Created with the Standard Edition of HelpNDoc: [Keep Your PDFs Safe from Unauthorized Access with These Security Measures](#)

TONFrameCaptureCompleted

TONFrameCaptureCompleted

[Prev](#)
[Next](#)

Type of the [OnFrameCaptureCompleted](#) event.

Unit

[VidGrab](#)

Declaration

TONFrameCaptureCompleted=**procedure**(Sender: TObject; FrameBitmap: Graphics.TBitmap; FrameBitmap: Graphics.TBitmap; BitmapWidth: LongInt; BitmapHeight: LongInt; FrameNumber: LongWord; FrameTime: LargeInteger; DestType: TFrameCaptureDest; FileName: **string**; Success: Boolean; FrameId: LongInt) **of object**;

typedef void __fastcall(__closure *TONFrameCaptureCompleted)(System::TObject *Sender, Graphics.TBitmap *FrameBitmap, **int** BitmapWidth, **int** BitmapHeight, **unsigned** FrameNumber, __int64 FrameTime, TFrameCaptureDest DestType, wchar_t *FileName, **bool** Success, **int** FrameId);

Description

Type of the [OnFrameCaptureCompleted](#) event.

The values returned are described in the [OnFrameCaptureCompleted](#) event.

See Also

[TAutoFileName](#) [TFrameCaptureDest](#) [BurstCount](#) [BurstInterval](#) [BurstMode](#) [BurstType](#) [CaptureFrameSyncTo](#) [CaptureFrameTo](#) [FrameCaptureHeight](#) [FrameCaptureWidth](#) [FrameCaptureWithoutOverlay](#) [FrameCaptureZoomSize](#) [GetFrameInfoString](#) [GetLastFrameAsHBITMAP](#) [GetLastFrameAsTBitmap](#) [GetLastFrameBitmapBits](#) [GetLastFrameBitmapBits2](#) [GetLastFrameWaitTimeoutMs](#) [JPEGPerformance](#) [JPEGProgressiveDisplay](#) [JPEGQuality](#) [Last](#) [BurstFrameCapture](#) [FileName](#) [Last](#) [CaptureFrameTo](#) [FileName](#) [OnDiskFull](#) [OnFrameBitmap](#) [SetFrameCaptureBounds](#) [StoragePath](#) [WebcamStillCaptureButton](#)

Created with the Standard Edition of HelpNDoc: [Elevate Your CHM Help Files with HelpNDoc's Advanced Customization Options](#)

TONFrameOverlayUsingDC

TONFrameOverlayUsingDC

[Prev](#)
[Next](#)

Type of the [OnFrameOverlayUsingDC](#) event.

Unit

[VidGrab](#)

Declaration

TOnFrameOverlayUsingDC=**procedure**(Sender: TObject; Dc: HDC; FrameNumber: LongWord; FrameTime: LargeInteger; Frameld: LongInt) **of object**;

typedef void __fastcall(__closure *TOnFrameOverlayUsingDC)(System::TObject *Sender, HDC Dc, **unsigned** FrameNumber, __int64 FrameTime, **int** Frameld);

Description

Type of the [OnFrameOverlayUsingDC](#) event.

Dc is the current device context of the current video frame.

Frameld must be passed to [GetFrameInfo](#) to retrieve information about the current frame.

Use the **Dc** device context to draw objects over video frames by using GDI functions.

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDIB](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#) [DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#) [OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#) [OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#) [OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#) [SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#) [SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#) [SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#) [SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#) [SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#) [SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#) [SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#) [SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#) [SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#) [SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#) [SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#) [SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#) [SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#) [SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#) [SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay](#) [Top](#) [SetTextOverlay](#) [Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay](#) [BkColor](#) [TextOverlay](#) [CreateCustomFont](#) [TextOverlay](#) [CreateCustomFont2](#) [TextOverlay](#) [Enabled](#) [TextOverlay](#) [Font](#) [TextOverlay](#) [FontColor](#) [TextOverlay](#) [Left](#) [TextOverlay](#) [Right](#) [TextOverlay](#) [Scrolling](#) [TextOverlay](#) [ScrollingSpeed](#) [TextOverlay](#) [Selector](#) [TextOverlay](#) [Shadow](#) [TextOverlay](#) [ShadowColor](#) [TextOverlay](#) [ShadowDirection](#) [TextOverlay](#) [String](#) [TextOverlay](#) [Top](#) [TextOverlay](#) [Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Enhance Your Documentation with HelpNDoc's Advanced Project Analyzer](#)

TOnFrameOverlayUsingDIB

TOnFrameOverlayUsingDIB

[Prev](#)

[Next](#)

Type of the OnFrameOverlayUsingDIB event.

Unit**VidGrab****Declaration**

TOnFrameOverlayUsingDIB=**procedure**(Sender: TObject; FrameBitmapHandle: HBITMAP; FrameNumber: LongWord; FrameTime: LargeInteger; Frameld: LongInt) **of object**;

typedef void __fastcall(__closure *TOnFrameOverlayUsingDIB)(System::TObject *Sender, HBITMAP FrameBitmapHandle, **unsigned** FrameNumber, __int64 FrameTime, **int** Frameld);

Description

Type of the [OnFrameOverlayUsingDIB](#) event.

FrameBitmapHandle: handle of the bitmap of the current video frame

Frameld: must be passed to [GetFrameInfo](#) to retrieve information about the current frame.

Use GetObject to retrieve the DIB section information. E.g.:

```
procedure TfrmMainForm.VideoGrabberFrameOverlayUsingDIB(Sender: TObject;
  FrameBitmapHandle: HBITMAP; FrameNumber: Cardinal; FrameTime: Int64;
  FrameId: Integer);
var
  DibSection: TDibSection;
  i: integer;
  SkipLine: boolean;
begin
  if GetObject (FrameBitmapHandle, sizeof (TDibSection), @DibSection) = 0 then Exit;
  SkipLine := false;
  for i := 0 to DibSection.dsBm.bmHeight - 1 do begin
    if SkipLine then begin
      ZeroMemory (Pointer (Integer(DibSection.dsBm.bmBits) + (i * DibSection.dsBm.bmPitch)),
        sizeof (TDibSection));
      SkipLine := not SkipLine;
    end;
  end;
end;
```

See Also

[TCardinalDirection](#) [TOnFrameOverlayUsingDC](#) [TTextOverlayAlign](#) [AdjustOverlayAspectRatio](#)
[DrawBitmapOverFrame](#) [GetFrameInfoString](#) [ImageOverlay](#) [StretchToVideoSize](#) [MouseWheelEventEnabled](#)
[OnFrameBitmap](#) [OnFrameBitmapEventSynchron](#) [OnFrameOverlayUsingDC](#) [OnFrameOverlayUsingDIB](#)
[OnFrameOverlayUsingVIDEOHDR](#) [OnMouseDown](#) [OnMouseMove](#) [OnMouseUp](#) [OnMouseWheel](#)
[OverlayAfterTransform](#) [RefreshPlayerOverlays](#) [SetImageOverlay](#) [AlphaBlend](#)
[SetImageOverlay](#) [AlphaBlendValue](#) [SetImageOverlay](#) [ChromaKey](#)
[SetImageOverlay](#) [ChromaKeyLeewayPercent](#) [SetImageOverlay](#) [ChromaKeyRGBColor](#)
[SetImageOverlay](#) [Enabled](#) [SetImageOverlay](#) [Height](#) [SetImageOverlay](#) [LeftLocation](#)
[SetImageOverlay](#) [RotationAngle](#) [SetImageOverlay](#) [StretchToVideoSize](#) [SetImageOverlay](#) [TargetDisplay](#)
[SetImageOverlay](#) [TopLocation](#) [SetImageOverlay](#) [Transparent](#) [SetImageOverlay](#) [TransparentColorValue](#)
[SetImageOverlay](#) [UseTransparentColor](#) [SetImageOverlay](#) [Width](#) [SetImageOverlayFromBMPFile](#)
[SetImageOverlayFromHBitmap](#) [SetImageOverlayFromHBitmap2](#) [SetImageOverlayFromImageFile](#)
[SetImageOverlayFromImageFile2](#) [SetImageOverlayFromJPEGFile](#) [SetImageOverlayFromTBitmap](#)
[SetImageOverlayFromTBitmap2](#) [SetImageOverlayFromTImage](#) [SetImageOverlayFromTImage2](#)
[SetTextOverlay](#) [Align](#) [SetTextOverlay](#) [BkColor](#) [SetTextOverlay](#) [CustomVar](#) [SetTextOverlay](#) [Enabled](#)
[SetTextOverlay](#) [Font](#) [SetTextOverlay](#) [FontColor](#) [SetTextOverlay](#) [GradientColor](#)
[SetTextOverlay](#) [GradientMode](#) [SetTextOverlay](#) [HighResFont](#) [SetTextOverlay](#) [Left](#) [SetTextOverlay](#) [Right](#)
[SetTextOverlay](#) [Scrolling](#) [SetTextOverlay](#) [ScrollingSpeed](#) [SetTextOverlay](#) [Shadow](#)
[SetTextOverlay](#) [ShadowColor](#) [SetTextOverlay](#) [ShadowDirection](#) [SetTextOverlay](#) [String](#)

[SetTextOverlay](#) [TargetDisplay](#) [SetTextOverlay_Top](#) [SetTextOverlay_Transparent](#) [ShapeOverlay](#) [ShapeOverlayEnabled](#) [ShapeOverlayList](#) [TextOverlay](#) [Align](#) [TextOverlay_BkColor](#) [TextOverlay_CreateCustomFont](#) [TextOverlay_CreateCustomFont2](#) [TextOverlay_Enabled](#) [TextOverlay_Font](#) [TextOverlay_FontColor](#) [TextOverlay_Left](#) [TextOverlay_Right](#) [TextOverlay_Scrolling](#) [TextOverlay_ScrollingSpeed](#) [TextOverlay_Selector](#) [TextOverlay_Shadow](#) [TextOverlay_ShadowColor](#) [TextOverlay_ShadowDirection](#) [TextOverlay_String](#) [TextOverlay_Top](#) [TextOverlay_Transparent](#) [TranslateMouseCoordinates](#)

Created with the Standard Edition of HelpNDoc: [Modernize your help files with HelpNDoc's WinHelp HLP to CHM conversion tool](#)

TOnFrameProgress

TOnFrameProgress

[Prev](#)

[Next](#)

The type of the [OnFrameProgress](#) event.

Unit

[VidGrab](#)

Declaration

TOnFrameProgress=**procedure**(Sender: TObject; FrameNumber: LongWord; FrameTime: LargeInteger; FrameId: LongInt) **of object**;

typedef void __fastcall(__closure *TOnFrameProgress)(System::TObject *Sender, **unsigned** FrameNumber, __int64 FrameTime, **int** FrameId);

Description

The type of the [OnFrameProgress](#) event.

FrameId must be passed to [GetFrameInfo](#) to retrieve information about the current frame.

Look at the MainDemo project for sample code.

See Also

[TOnResizeVideo](#) [AutoRefreshPreview](#) [BackgroundColor](#) [Display_AutoSize](#) [Display_FullScreen](#) [Display_SetLocation](#) [Display_VideoPortEnabled](#) [ImageRatio](#) [InFrameProgressEvent](#) [IsVideoInterlaced](#) [IsVideoPortAvailable](#) [OnFrameProgress](#) [OnKeyPress](#) [OnPreviewStarted](#) [OnResizeVideo](#) [PausePreview](#) [PreviewZoomSize](#) [ResumePreview](#) [StartPreview](#) [StartSynchronized](#) [StopPreview](#) [Synchronized](#) [UseNearestVideoSize](#) [VideoHeight](#) [VideoRenderer](#) [VideoRendererPriority](#) [VideoWidth](#) [Visible](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your PDF Protection with These Simple Steps](#)

TOnLog

TOnLog

[Prev](#)

[Next](#)

The type of the [OnLog](#) event.

Unit

[VidGrab](#)

Declaration

TOnLog=**procedure**(Sender: TObject; LogType: TLogType; Severity: **string**; InfoMsg: **string**)**of object**;

typedef void __fastcall(__closure *TOnLog)(System::TObject *Sender, TLogType LogType, wchar_t *Severity, wchar_t *InfoMsg);

Description

The type of the [OnLog](#) event.

Severity returns "[INFO]", "[WARNING]" or "[ERROR]".

LogType returns a [TLogType](#) constant.

InfoMsg returns a comment about this log.

Note: it is possible to retrieve the string value of the LogType parameter by invoking [GetLogString](#) (LogType).

Created with the Standard Edition of HelpNDoc: [Save time and frustration with HelpNDoc's WinHelp HLP to CHM conversion feature](#)

TOnMotionDetected

TOnMotionDetected

[Prev](#)
[Next](#)

Type of the [OnMotionDetected](#) event.

Unit

[VidGrab](#)

Declaration

TOnMotionDetected=**procedure**(Sender: TObject; GlobalMotionRatio: Double; MaxMotionCellX: LongInt; MaxMotionCellY: LongInt; FrameBitmap: Graphics.TBitmap; FrameBitmap: Graphics.TBitmap; FrameNumber: LongWord; FrameTime: LargeInteger; FrameId: LongInt; **var** CaptureFrame: Boolean) **of object**;

typedef void __fastcall(__closure *TOnMotionDetected)(System::TObject *Sender, **double** GlobalMotionRatio, **int** MaxMotionCellX, **int** MaxMotionCellY, Graphics.TBitmap *FrameBitmap, **unsigned** FrameNumber, __int64 FrameTime, **int** FrameId, **bool** &CaptureFrame);

Description

Type of the [OnMotionDetected](#) event.

GlobalMotionRatio: motion ratio of the whole video frame, compared to the previous one

FrameData: returns the bitmap of the video frame, the frame number and the frame time

CaptureFrame: set this parameter to TRUE if you want to capture this frame. In this case it will be returned by the OnFrameCaptureCompleted event.

See Also

[Color / Greyscale Grid structure / grid sensitivity Motion ratio Recording only when motion is detected Video noise TOnMotionNotDetected MotionDetector_CellMotionRatio MotionDetector_CompareBlue MotionDetector_CompareGreen MotionDetector_CompareRed MotionDetector_Enabled MotionDetector_EnumGridDialogControls MotionDetector_Get2DTextGrid MotionDetector_Get2DTextMotion MotionDetector_GetCellLocation MotionDetector_GetCellSensitivity MotionDetector_GetCellSize MotionDetector_GloballyIncOrDecSensitivity MotionDetector_GlobalMotionRatio MotionDetector_GreyScale MotionDetector_Grid MotionDetector_GridXCount MotionDetector_GridYCount MotionDetector_IsGridValid MotionDetector_MaxDetectionsPerSecond MotionDetector_ReduceCPULoad MotionDetector_ReduceVideoNoise MotionDetector_Reset MotionDetector_ResetGlobalSensitivity MotionDetector_SetCellSensitivity MotionDetector_SetGridSize MotionDetector_ShowGridDialog MotionDetector_Triggered MotionDetector_UseThisReferenceSample OnBacktimedFramesCountReached OnMotionDetected OnMotionNotDetected RecordingOnMotion_Enabled RecordingOnMotion_MotionThreshold RecordingOnMotion_NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: [Elevate your documentation to new heights with HelpNDoc's built-in SEO](#)

TOnMotionNotDetected

TOnMotionNotDetected

[Prev](#)
[Next](#)

Type of the [OnMotionNotDetected](#) event.

Unit

[VidGrab](#)

Declaration

TOnMotionNotDetected=**procedure**(Sender: TObject; FrameBitmap: Graphics.TBitmap; FrameBitmap: Graphics.TBitmap; FrameNumber: LongWord; FrameTime: LargeInteger; Frameld: LongInt; **var** CaptureFrame: Boolean) **of object**;

typedef void __fastcall(__closure *TOnMotionNotDetected)(System::TObject *Sender, Graphics.TBitmap *FrameBitmap, **unsigned** FrameNumber, __int64 FrameTime, **int** Frameld, **bool** &CaptureFrame);

Description

Type of the [OnMotionNotDetected](#) event.

FrameData: returns the bitmap of the video frame, the frame number and the frame time

CaptureFrame: set this parameter to TRUE if you want to capture this frame. In this case it will be returned by the OnFrameCaptureCompleted event.

See Also

[Color / Greyscale Grid structure / grid sensitivity Motion ratio Recording only when motion is detected Video noise TOnMotionDetected MotionDetector CellMotionRatio MotionDetector CompareBlue MotionDetector CompareGreen MotionDetector CompareRed MotionDetector Enabled MotionDetector EnumGridDialogControls MotionDetector Get2DTextGrid MotionDetector Get2DTextMotion MotionDetector GetCellLocation MotionDetector GetCellSensitivity MotionDetector GetCellSize MotionDetector GloballyIncOrDecSensitivity MotionDetector GlobalMotionRatio MotionDetector GreyScale MotionDetector Grid MotionDetector GridXCount MotionDetector GridYCount MotionDetector IsGridValid MotionDetector MaxDetectionsPerSecond MotionDetector ReduceCPULoad MotionDetector ReduceVideoNoise MotionDetector Reset MotionDetector ResetGlobalSensitivity MotionDetector SetCellSensitivity MotionDetector SetGridSize MotionDetector ShowGridDialog MotionDetector Triggered MotionDetector UseThisReferenceSample OnBacktimedFramesCountReached OnMotionDetected OnMotionNotDetected RecordingOnMotion Enabled RecordingOnMotion MotionThreshold RecordingOnMotion NoMotionPauseDelayMs](#)

Created with the Standard Edition of HelpNDoc: Maximize Your Documentation Efficiency with a Help Authoring Tool

TOnMouseWheel

TOnMouseWheel

[Prev](#)
[Next](#)

Type of the [OnMouseWheel](#) event

Declaration

TOnMouseWheel=**procedure** (Sender: TObject; VideoWindow: LongInt; DirectionUp: Boolean; X: LongInt; Y: LongInt) **of object**;

typedef void __fastcall (__closure *TOnMouseWheel)(System::TObject* Sender, **int** VideoWindow, **bool** DirectionUp, **int** X, **int** Y);

Description

Type of the [OnMouseWheel](#) event.

The DirectionUp parameter returns:

- true when the mouse wheel goes up,
- false when the mouse wheel goes down.

Created with the Standard Edition of HelpNDoc: [Achieve Professional Documentation Results with a Help Authoring Tool](#)

TOnONVIFDiscoveryCompletedNotification

TOnONVIFDiscoveryCompletedNotification

[Prev](#)

[Next](#)

Type of the [OnONVIFDiscoveryCompletedNotification](#) event

Declaration

TOnONVIFDiscoveryCompletedNotification=**procedure** (Sender: TObject; DiscoveryCallbackStatus: TDiscoveryCallbackStatus; CameraCount: LongInt) **of object**;

Description

DiscoveryCallbackStatus: TDiscoveryCallbackStatus:

specifying whether the event occurs for a new camera discovered, or for the completion of the Multicast or IP range discovery

CameraCount: LongInt:

number of cameras currently discovered

See Also

[IR Cut Filter of Axis cameras](#) [TDiscoveryCallbackStatus](#) [TONVIFDeviceInfo](#) [Video stream of IP camera with audio of PC microphone or other audio capture device](#) [ExtraDLLPath](#) [GetLastErrorMessage](#) [IPCameraURL](#) [IsURLResponding](#) [IsURLVideoStreamAvailable](#) [OnONVIFDiscoveryCompletedNotification](#) [ONVIF_GetStr](#) [ONVIF_SetStr](#) [ONVIFCancelDiscovery](#) [ONVIFDeviceInfo](#) [ONVIFDiscoverCameras](#) [IPRange](#) [ONVIFDiscoverCameras](#) [Multicast](#) [ONVIFEnumCamerasDiscovered](#) [ONVIFPTZGetLimits](#) [ONVIFPTZGetPosition](#) [ONVIFPTZPreset](#) [ONVIFPTZSendAuxiliaryCommand](#) [ONVIFPTZSetPosition](#) [ONVIFPTZStartMove](#) [ONVIFPTZStopMove](#) [ONVIFSnapShot](#) [OpenURLAsync](#) [OpenURLAsyncStatus](#) [SendIPCameraCommand](#) [SetDatasteadFilterDllName](#) [SetIPCameraSetting](#)

Created with the Standard Edition of HelpNDoc: [Revolutionize Your CHM Help File Output with HelpNDoc](#)

TOnOpenURLAsyncStatusChanged

TOnOpenURLAsyncStatusChanged

[Prev](#)

[Next](#)

Type of the [OnOpenURLAsyncStatusChanged](#) event

Declaration

TOnOpenURLAsyncStatusChanged=**procedure** (Sender: TObject; OldStatus: TOpenURLAsyncStatus; NewStatus: TOpenURLAsyncStatus; TargetState: TCurrentState) **of object**;

typedef void __fastcall (__closure *TOnOpenURLAsyncStatusChanged)(System::TObject* Sender, TOpenURLAsyncStatus OldStatus, TOpenURLAsyncStatus NewStatus, TCurrentState TargetState);

Created with the Standard Edition of HelpNDoc: [Effortlessly Support Your Windows Applications with HelpNDoc's CHM Generation](#)

TOnPlayerBufferingData

TOnPlayerBufferingData

[Prev](#)
[Next](#)

Type of the [OnPlayerBufferingData](#) event.

Unit

[VidGrab](#)

Declaration

TOnPlayerBufferingData=**procedure**(Sender: TObject; StartingToBuffer: Boolean) **of object**;

```
typedef void __fastcall(__closure *TOnPlayerBufferingData)(System::TObject *Sender, bool StartingToBuffer);
```

Description

Type of the [OnPlayerBufferingData](#) event.

StartingToBuffer returns true when the player starts buffering data.

StartingToBuffer returns false when the player ends buffering data.

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TPlayerState](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchronise](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Write EPub books for the iPad](#)

TOnPlayerStateChanged

TOnPlayerStateChanged

[Prev](#)
[Next](#)

Type of the [OnPlayerStateChanged](#) event.

Declaration

TOnPlayerStateChanged=**procedure** (Sender: TObject; OldPlayerState: TPlayerState; NewPlayerState: TPlayerState) **of object**

```
typedef void __fastcall (__closure *TOnPlayerStateChanged)(System::TObject* Sender, TPlayerState OldPlayerState, TPlayerState NewPlayerState);
```

Description

Type of the [OnPlayerStateChanged](#) event.

See Also

[Player features](#) [TPlayerState](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#) [AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#) [FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#) [MP4NeedsReindexing](#)

[OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#) [OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#) [PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#) [PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#) [PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#) [PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchrone](#) [PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#) [SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Easily create Qt Help files](#)

TOnProgress

TOnProgress

[Prev](#)

[Next](#)

Type of the [OnCopyPreallocDataProgress](#) and [OnReencodingProgress](#) events.

Unit

[VidGrab](#)

Declaration

TOnProgress=**procedure**(Sender: TObject; Percent: LongInt; Position: LargeInteger; Duration: LargeInteger) **of object**;

```
typedef void __fastcall(__closure *TOnProgress)(System::TObject *Sender, int Percent, __int64 Position, __int64 Duration);
```

Description

Type of the [OnCopyPreallocDataProgress](#) and [OnReencodingProgress](#) events.

Created with the Standard Edition of HelpNDoc: [Transform Your Documentation Process with HelpNDoc's Project Analyzer](#)

TOnProgressCommented

TOnProgressCommented

[Prev](#)

[Next](#)

Type of the [OnBitmapsLoadingProgress](#) event.

Unit

[VidGrab](#)

Declaration

TOnProgressCommented=**procedure**(Sender: TObject; Percent: LongInt; Position: LargeInteger; Duration: LargeInteger; Comment: **String**)**of object**;

```
typedef void __fastcall(__closure *TOnProgressCommented)(System::TObject *Sender, int Percent, __int64 Position, __int64 Duration, wchar_t *Comment);
```

Description

Type of the [OnBitmapsLoadingProgress](#) event.

Created with the Standard Edition of HelpNDoc: [Full-featured Help generator](#)

TOnRawSample

TOnRawSample

[Prev](#)

[Next](#)

Type of the [OnRawVideoSample](#) and [OnRawAudioSample](#) events.

Unit

[VidGrab](#)

Declaration

TOnRawSample=**procedure**(Sender: TObject; pSampleBuffer: pByte; SampleBufferSize: LongInt; SampleDataLength: LongInt; FormatType: TFormatType; pFormat: Pointer; pBitmapInfoHeader: Pointer; SampleStartTime: LargeInteger; SampleStopTime: LargeInteger) **of object**;

```
typedef void __fastcall(__closure *TOnRawSample)(System::TObject *Sender, Windows::PByte
pSampleBuffer, int SampleBufferSize, int SampleDataLength, TFormatType FormatType, void *pFormat,
void *pBitmapInfoHeader, __int64 SampleStartTime, __int64 SampleStopTime);
```

Description

Type of the [OnRawVideoSample](#) and [OnRawAudioSample](#) events.

SampleBuffer: pByte

Pointer to the buffer containing the sample data.

SampleBufferSize: Integer

Size of the buffer containing the sample data.

SampleDataLength: Integer

Length of the valid data in the buffer.

FormatType: type of the format, see [TFormatType](#)

pFormat: pointer

Pointer to the format whose type depends of the FormatType:

ft_VideoInfo: pFormat is a pVideoInfoHeader pointer

ft_VideoInfo2: pFormat is a pVideoInfoHeader2 pointer

ft_MpegVideo: pFormat is a pMpegVideoInfo pointer

ft_Mpeg2Video: pFormat is a pMpeg2VideoInfo pointer

ft_WaveFormatEx: pFormat is a pWaveFormatEx pointer.

SampleStartTime: int64 (or Double for the OCX version)

Start time of the sample, expressed in 100 nano-seconds units.

SampleEndTime: int64 (or Double for the OCX version)

End time of the sample, expressed in 100 nano-seconds units.

See Also

[OnRawAudioSample](#) [OnRawVideoSample](#) [RawAudioSampleCapture](#) [RawCaptureAsyncEvent](#)
[RawVideoSampleCapture](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Productivity with HelpNDoc's CHM Help File Creation Features](#)

TOnRecordingCompleted

TOnRecordingCompleted

[Prev](#)

[Next](#)

The type of the [OnRecordingCompleted](#) event.

Unit

[VidGrab](#)

Declaration

TOnRecordingCompleted=**procedure**(Sender: TObject; FileName: **string**; Success: Boolean) **of object**;

```
typedef void __fastcall(__closure *TOnRecordingCompleted)(System::TObject *Sender, wchar_t
*FileName, bool Success);
```

Description

The type of the [OnRecordingCompleted](#) event.

FileName: full qualified path of the AVI file created.

Success: true upon success, false upon failure (e.g. insufficient disk space when reencoding).

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last_Recording_FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [Effortlessly bring your documentation online with HelpNDoc](#)

TOnRecordingReadyToStart

TOnRecordingReadyToStart

[Prev](#)

[Next](#)

The type of the [OnRecordingReadyToStart](#) event.

Unit

[VidGrab](#)

Declaration

TOnRecordingReadyToStart=**procedure**(Sender: TObject) **of object**;

```
typedef void __fastcall(__closure *TOnRecordingReadyToStart)(System::TObject *Sender);
```

Description

The type of the [OnRecordingReadyToStart](#) event.

OkToStart: assign true to OkToStart to allow the recording to start, false to cancel the recording.

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TRecordingMethod](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last_Recording_FileName](#) [OnBacktimedFramesCountReached](#)

[OnCopyPreallocDataCompleted](#)
[OnCopyPreallocDataProgress](#)
[OnCopyPreallocDataStarted](#)
[OnCreatePreallocFileCompleted](#)
[OnCreatePreallocFileStarted](#)
[OnDiskFull](#)
[OnRecordingCompleted](#)
[OnRecordingPaused](#)
[OnRecordingReadyToStart](#)
[OnRecordingStarted](#)
[OnReencodingCompleted](#)
[OnReencodingStarted](#)
[PauseRecording](#)
[PreallocCapFileCopiedAfterRecording](#)
[PreallocCapFileEnabled](#)
[PreallocCapFileName](#)
[PreallocCapFileSizeInMB](#)
[RecordingBacktimedFramesCount](#)
[RecordingCanPause](#)
[RecordingDuration](#)
[RecordingFileName](#)
[RecordingFileSizeMaxInMB](#)
[RecordingFourCC](#)
[RecordingHeight](#)
[RecordingInNativeFormat](#)
[RecordingKBytesWrittenToDisk](#)
[RecordingMethod](#)
[RecordingSize](#)
[RecordingTimer](#)
[RecordingTimerInterval](#)
[RecordingWidth](#)
[ResumeRecording](#)
[SaveCompressorSettingsToDataString](#)
[SetMultiplexerFilterByName](#)
[StartAudioRecording](#)
[StartRecording](#)
[StartSynchronized](#)
[StopRecording](#)
[StoragePath](#)
[Synchronized](#)
[SyncPreview](#)
[VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [Converting Word Docs to eBooks Made Easy with HelpNDoc](#)

TOnResizeVideo

TOnResizeVideo

[Prev](#)

[Next](#)

The type of the [OnResizeVideo](#) event.

Unit

[VidGrab](#)

Declaration

TOnResizeVideo=**procedure**(Sender: TObject; SourceWidth, SourceHeight: LongInt) **of object**;

```
typedef void __fastcall(__closure *TOnResizeVideo)(System::TObject *Sender, int SourceWidth, int SourceHeight);
```

Description

The type of the [OnResizeVideo](#) event.

SourceWidth: width of the video source

SourceHeight: height of the video source

See Also

[TOnFrameProgress](#)
[AutoRefreshPreview](#)
[BackgroundColor](#)
[Display](#)
[AutoSize](#)
[Display_FullScreen](#)
[Display_SetLocation](#)
[Display_VideoPortEnabled](#)
[ImageRatio](#)
[InFrameProgressEvent](#)
[IsVideoInterlaced](#)
[IsVideoPortAvailable](#)
[OnFrameProgress](#)
[OnKeyPress](#)
[OnPreviewStarted](#)
[OnResizeVideo](#)
[PausePreview](#)
[PreviewZoomSize](#)
[ResumePreview](#)
[StartPreview](#)
[StartSynchronized](#)
[StopPreview](#)
[Synchronized](#)
[UseNearestVideoSize](#)
[VideoHeight](#)
[VideoRenderer](#)
[VideoRendererPriority](#)
[VideoWidth](#)
[Visible](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Documentation Output with HelpNDoc's Advanced Project Analyzer](#)

TOnSourceFileToDestFileCompleted

TOnSourceFileToDestFileCompleted

[Prev](#)

[Next](#)

Type of the [OnCopyPreallocDataCompleted](#) and [OnReencodingCompleted](#) events.

Unit

[VidGrab](#)

Declaration

TOnSourceFileToDestFileCompleted=**procedure**(Sender: TObject; SourceFile, DestFile: **String**; Success: Boolean) **of object**;

```
typedef void __fastcall(__closure *TOnSourceFileToDestFileCompleted)(System::TObject *Sender,
wchar_t *SourceFile, wchar_t *DestFile, bool Success);
```

Description

Type of the [OnCopyPreallocDataCompleted](#) and [OnReencodingCompleted](#) events.

Created with the Standard Edition of HelpNDoc: [Effortlessly Create High-Quality Help Documentation with a Help Authoring Tool](#)

TOnSourceFileToDestFileStarted

TOnSourceFileToDestFileStarted

[Prev](#)
[Next](#)

Type of the [OnCopyPreallocDataStarted](#) and [OnReencodingStarted](#) events.

Unit

[VidGrab](#)

Declaration

TOnSourceFileToDestFileStarted=**procedure**(Sender: TObject; SourceFile, DestFile: **String**)**of object**;

```
typedef void __fastcall(__closure *TOnSourceFileToDestFileStarted)(System::TObject *Sender, wchar_t
*SourceFile, wchar_t *DestFile);
```

Description

Type of the [OnCopyPreallocDataStarted](#) and [OnReencodingStarted](#) events.

Created with the Standard Edition of HelpNDoc: [Elevate your documentation to new heights with HelpNDoc's built-in SEO](#)

TOnTextOverlayScrollingCompleted

TOnTextOverlayScrollingCompleted

[Prev](#)
[Next](#)

Type of the [OnTextOverlayScrollingCompleted](#) event

Declaration

TOnTextOverlayScrollingCompleted=**procedure** (Sender: TObject; TextOverlayIndex: LongInt) **of object**;

```
typedef void __fastcall(__closure *TOnTextOverlayScrollingCompleted)(System::TObject *Sender, int
TextOverlayIndex)
```

Description

Type of the [OnTextOverlayScrollingCompleted](#) event

Created with the Standard Edition of HelpNDoc: [Effortlessly Create Encrypted, Password-Protected PDFs](#)

TOnThirdPartyFilterConnected

TOnThirdPartyFilterConnected

[Prev](#)
[Next](#)

The type of the [OnThirdPartyFilterConnected](#) event.

Unit

[VidGrab](#)

Declaration

TOnThirdPartyFilterConnected=**procedure**(Sender: TObject; Location: TThirdPartyFilterList; **Index**: LongInt; Intf: IUnknown) **of object**;

```
typedef void __fastcall(__closure *TOnThirdPartyFilterConnected)(System::TObject *Sender,
TThirdPartyFilterList Location, int Index, _di_IUnknown Intf);
```

Description

The type of the [OnThirdPartyFilterConnected](#) event.

Returns the IUnknown interface of the connected third-party filter.

Location: list of the insertion point of the filter

Index: index of the filter in the Location list

Intf: returns the IUnknown interface of the third-party filter immediately after it has been connected, allowing to set/retrieve its properties programmatically.

See Also

[TThirdPartyFilterList](#) [OnThirdPartyFilterConnected](#) [ThirdPartyFilter](#) [AddToList](#) [ThirdPartyFilter](#) [ClearList](#) [ThirdPartyFilter](#) [Enable](#) [ThirdPartyFilter](#) [RemoveFromList](#) [ThirdPartyFilter](#) [ShowDialog](#)

Created with the Standard Edition of HelpNDoc: [Easily create Web Help sites](#)

TOnThreadSync

TOnThreadSync

[Prev](#)

[Next](#)

Type of the [OnThreadSync](#) event.

Description

Type of the [OnThreadSync](#) event.

Parameter:

ThreadSyncPoint: [TThreadSyncPoint](#)

See Also

[Opening a clip or an IP URL from a background thread without blocking the main thread](#) [Player features](#) [TThreadSyncPoint](#) [EnableThreadMode](#) [OnThreadSync](#)

Created with the Standard Edition of HelpNDoc: [Revolutionize your documentation process with HelpNDoc's online capabilities](#)

TOnTVChannelScanStarted

TOnTVChannelScanStarted

[Prev](#)

[Next](#)

Type of the [OnTVChannelScanStarted](#) event.

Unit

[VidGrab](#)

Declaration

TOnTVChannelScanStarted=**procedure**(Sender: TObject; MinChannel: LongInt; MaxChannel: LongInt) **of object**;

```
typedef void __fastcall(__closure *TOnTVChannelScanStarted)(System::TObject *Sender, int MinChannel,
```

```
int MaxChannel);
```

Description

Type of the [OnTVChannelScanStarted](#) event.

Reports the minimal and maximal channel that will be scanned for the current [country code](#) and [tuner input type](#).

See Also

[TTVChannelInfo](#) [TOnTVChannelSelected](#) [TTunerInputType](#) [GetTVChannelInfo](#) [IsHorizontalSyncLocked](#) [IsTVAudioAvailable](#) [IsTVAutoTuneRunning](#) [IsTVTunerAvailable](#) [OnTVChannelScanCompleted](#) [OnTVChannelScanStarted](#) [OnTVChannelSelected](#) [TunerFrequency](#) [TunerMode](#) [TVChannel](#) [TVClearFrequencyOverrides](#) [TVCountryCode](#) [TVGetMinMaxChannels](#) [TVSetChannelFrequencyOverride](#) [TVStartAutoScan](#) [TVStopAutoScan](#) [TVTunerInputType](#) [TVUseFrequencyOverrides](#) [VCRHorizontalLocking](#)

Created with the Standard Edition of HelpNDoc: [Create iPhone web-based documentation](#)

TOnTVChannelSelected

TOnTVChannelSelected

[Prev](#)
[Next](#)

Type of the [OnTVChannelSelected](#) event.

Unit

[VidGrab](#)

Declaration

TOnTVChannelSelected=**procedure**(Sender: TObject; Channel: LongInt; Locked: Boolean; DefaultVideoFrequency: LongInt; OverriddenVideoFrequency: LongInt; TunerVideoFrequency: LongInt; TunerAudioFrequency: LongInt) **of object**;

```
typedef void __fastcall(__closure *TOnTVChannelSelected)(System::TObject *Sender, int Channel, bool Locked, int DefaultVideoFrequency, int OverriddenVideoFrequency, int TunerVideoFrequency, int TunerAudioFrequency);
```

Description

Type of the [OnTVChannelSelected](#) event.

Channel:

Reports the current TV channel in use

Locked:

Returns true if the horizontal sync is locked for this channel.

DefaultVideoFrequency:

Reports the default video frequency for the current TV country code / tuner input type.

Can report -1 if a frequency override has been set and the default TV frequency is not saved in the registry.

OverriddenVideoFrequency:

Reports the overridden frequency for this channel, if it has been previously set by invoking TVSetChannelFrequencyOverrideVidGrab.TVideoGrabber.TVSetChannelFrequencyOverride.

TunerVideoFrequency:

Reports the current audio frequency, as reported by the TV tuner when selecting the TV channel.

TunerAudioFrequency:

Reports the current video frequency, as reported by the TV tuner when selecting the TV channel.

See Also

[TTVChannelInfo](#) [TOnTVChannelScanStarted](#) [TTunerInputType](#) [GetTVChannelInfo](#) [IsHorizontalSyncLocked](#) [IsTVAudioAvailable](#) [IsTVAutoTuneRunning](#) [IsTVTunerAvailable](#) [OnTVChannelScanCompleted](#) [OnTVChannelScanStarted](#) [OnTVChannelSelected](#) [TunerFrequency](#) [TunerMode](#) [TVChannel](#) [TVClearFrequencyOverrides](#) [TVCountryCode](#) [TVGetMinMaxChannels](#) [TVSetChannelFrequencyOverride](#) [TVStartAutoScan](#) [TVStopAutoScan](#) [TVTunerInputType](#) [TVUseFrequencyOverrides](#) [VCRHorizontalLocking](#)

Created with the Standard Edition of HelpNDoc: [Free EBook and documentation generator](#)

TOnVideoCompressionSettings

TOnVideoCompressionSettings

[Prev](#)

[Next](#)

Type of the [OnVideoCompressionSettings](#) event.

Unit

[VidGrab](#)

Declaration

TOnVideoCompressionSettings=**procedure**(Sender: TObject; CanKeyFrameRate: Boolean; CanPFFramesPerKeyFrame: Boolean; CanQuality: Boolean; CanWindowSize: Boolean; CanSpecifyDataRate: Boolean; DefaultFrameRate: LongInt; DefaultPFFramesPerKey: LongInt; DefaultQuality: Double; DefaultWindowSize: LongWord; DefaultDataRate: LongWord) **of object**;

typedef void __fastcall(**__closure** *TOnVideoCompressionSettings)(System::TObject *Sender, **bool** CanKeyFrameRate, **bool** CanPFFramesPerKeyFrame, **bool** CanQuality, **bool** CanWindowSize, **bool** CanSpecifyDataRate, **int** DefaultFrameRate, **int** DefaultPFFramesPerKey, **double** DefaultQuality, **unsigned** DefaultWindowSize, **unsigned** DefaultDataRate);

Description

Type of the [OnVideoCompressionSettings](#) event.

See "**General video compression properties**" in the [Software compression using codecs](#) chapter for more information about this event.

See Also

[Recording methods and properties](#) [TCompressionType](#) [AudioCompressor](#) [AudioCompressorIndex](#) [AudioCompressorName](#) [AudioCompressors](#) [AudioCompressorsCount](#) [CompressionMode](#) [CompressionType](#) [GetVideoCompressionSettings](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [OnVideoCompressionSettings](#) [RefreshDevicesAndCompressorsLists](#) [SaveCompressorSettingsToDataString](#) [SetVideoCompressionDefaults](#) [SetVideoCompressionSettings](#) [VideoCompression](#) [KeyFrameRate](#) [VideoCompression](#) [PFFramesPerKeyFrame](#) [VideoCompression](#) [Quality](#) [VideoCompression](#) [WindowSize](#) [VideoCompressor](#) [VideoCompressorIndex](#) [VideoCompressorName](#) [VideoCompressors](#) [VideoCompressorsCount](#)

Created with the Standard Edition of HelpNDoc: [Easily Add Encryption and Password Protection to Your PDFs](#)

TOnVideoFromBitmapsNextFrameNeeded

TOnVideoFromBitmapsNextFrameNeeded

[Prev](#)

[Next](#)

Type of the [OnVideoFromBitmapsNextFrameNeeded](#) event.

Unit

[VidGrab](#)

Declaration

TOnVideoFromBitmapsNextFrameNeeded=**procedure**(Sender: TObject; FirstSample: Boolean; **var** BitmapHandle: HBITMAP; **var** CanFreeBitmapHandle: Boolean; **var** BMPorJPEGFile: **String**; **var** EndOfData: Boolean) **of object**;

typedef void __fastcall(__closure *TOnVideoFromBitmapsNextFrameNeeded)(System::TObject *Sender, **bool** FirstSample, HBITMAP &BitmapHandle, **bool** &CanFreeBitmapHandle, wchar_t *&BMPorJPEGFile, **bool** &EndOfData);

Description

Type of the [OnVideoFromBitmapsNextFrameNeeded](#) event.

See Also

Created with the Standard Edition of HelpNDoc: [Easy CHM and documentation editor](#)

TOnVideoKeyPress

TOnVideoKeyPress

[Prev](#)

[Next](#)

Type of the [TOnKeyPress](#) event.

Unit

[VidGrab](#)

Declaration

TOnVideoKeyPress=**procedure**(Sender: TObject; VideoWindow: LongInt; **var** Key: Char; PhysicalKey: LongInt; ShiftState: TShiftState) **of object**;

typedef void __fastcall(__closure *TOnVideoKeyPress)(System::TObject *Sender, **int** VideoWindow, **char** &Key, **int** PhysicalKey, Classes::TShiftState ShiftState);

Description

Type of the [TOnKeyPress](#) event.

Note:

VideoWindow returns the number of the video window where the event occurred:

- 0 (by default) for single display,
- 0 or 1 when using dual display, according to the video window on which the event occurred.

Created with the Standard Edition of HelpNDoc: [Make the switch to CHM with HelpNDoc's hassle-free WinHelp HLP to CHM conversion tool](#)

TOnVideoMouseMove

TOnVideoMouseMove

[Prev](#)

[Next](#)

Type of the [OnMouseMove](#) event.

Unit

[VidGrab](#)

Declaration

TOnVideoMouseMove=**procedure**(Sender: TObject; VideoWindow: LongInt; Shift: TShiftState; X, Y: LongInt) **of object**;

typedef void __fastcall(__closure *TOnVideoMouseMove)(System::TObject *Sender, **int** VideoWindow,

Classes::TShiftState Shift, **int** X, **int** Y);

Description

Type of the [OnMouseMove](#) event.

Note:

VideoWindow returns the number of the video window where the event occurred:

- 0 (by default) for single display,
- 0 or 1 when using dual display, according to the video window on which the event occurred.

Created with the Standard Edition of HelpNDoc: [Free help authoring tool](#)

TOnVideoMouseUpDown

TOnVideoMouseUpDown

[Prev](#)

[Next](#)

Type of the [OnMouseDown](#) and [OnMouseUp](#) events.

Unit

[VidGrab](#)

Declaration

TOnVideoMouseUpDown=**procedure**(Sender: TObject; VideoWindow: LongInt; Button: TMouseButton; Shift: TShiftState; X, Y: LongInt) **of object**;

typedef void __fastcall(__closure *TOnVideoMouseUpDown)(System::TObject *Sender, **int** VideoWindow, Controls::TMouseButton Button, Classes::TShiftState Shift, **int** X, **int** Y);

Description

Type of the [OnMouseDown](#) and [OnMouseUp](#) events.

Note:

VideoWindow returns the number of the video window where the event occurred:

- 0 (by default) for single display,
- 0 or 1 when using dual display, according to the video window on which the event occurred.

Created with the Standard Edition of HelpNDoc: [Maximize Your CHM Help File Capabilities with HelpNDoc](#)

TONVIFDeviceInfo

TONVIFDeviceInfo

[Prev](#)

[Next](#)

Type of parameter returned by the [ONVIFDeviceInfo](#) function

Declaration

TONVIFDeviceInfo=(onv_Manufacturer, onv_Model, onv_HardwareId, onv_SerialNumber, onv_FirmwareVersion, onv_PTZInfo, onv_PTZPresets);

typedef enum TONVIFDeviceInfo

onv_Manufacturer=0,

onv_Model=1,

onv_HardwareId=2,

onv_SerialNumber=3,

onv_FirmwareVersion=4,

onv_PTZInfo=5,

```
onv_PTZPresets=6
TONVIFDeviceInfo;
```

Description

Type of parameter returned by the [ONVIFDeviceInfo](#) function

```
TONVIFDeviceInfo = (onv_Manufacturer, onv_Model, onv_HardwareId, onv_SerialNumber,  
onv_FirmwareVersion, onv_PTZInfo, onv_PTZPresets, onv_MacAddress, onv_AuxiliaryCommands,  
onv_XMLReplay, onv_XMLInfo);
```

See Also

[IR Cut Filter of Axis cameras](#) [TDiscoveryCallbackStatus](#) [TOnONVIFDiscoveryCompletedNotification](#) [ExtraDLLPath](#) [GetLastErrorMessages](#) [OnONVIFDiscoveryCompletedNotification](#) [ONVIF_GetStr](#) [ONVIF_SetStr](#) [ONVIFCancelDiscovery](#) [ONVIFDeviceInfo](#) [ONVIFDiscoverCameras](#) [IPRange](#) [ONVIFDiscoverCameras](#) [Multicast](#) [ONVIFEnumCamerasDiscovered](#) [ONVIFPTZGetLimits](#) [ONVIFPTZGetPosition](#) [ONVIFPTZPreset](#) [ONVIFPTZSendAuxiliaryCommand](#) [ONVIFPTZSetPosition](#) [ONVIFPTZStartMove](#) [ONVIFPTZStopMove](#) [ONVIFSnapShot](#) [OpenURLAsyncStatus](#) [SetDataSteeringFilterDllName](#)

Created with the Standard Edition of HelpNDoc: [Simplify Your Help Documentation Process with a Help Authoring Tool](#)

TOpenURLAsyncStatus

TOpenURLAsyncStatus

Prev

Next

Type of the [OpenURLAsyncStatus](#) function

Description

```
TOpenURLAsyncStatus = (oas_InProgress_Connecting, oas_InProgress_Connected,
oas_Completed_Success, oas_Undefined, oas_Completed_Error);
```

oas_Undefined: no asynchronous URL opening initiated yet

oas_InProgress_Connecting: asynchronous URL opening just started

oas_InProgress_Connected: URL connection successful, analyzis in progress

oas_Completed_Success: URL sucessfully connected

oas_Completed_Error: URL connection failed

Created with the Standard Edition of HelpNDoc: [Effortlessly Convert Your Word Doc to an eBook: A Step-by-Step Guide](#)

TPlayerState

TPlayerState

Prev

Next

Type used by the [PlayerState](#) property and [OnPlayerStateChanged](#) events.

Declaration

```
TPlayerState=(ps_Closed, ps_Stopped, ps_Paused, ps_Playing, ps_PlayingBackward, ps_FastForwarding, ps_FastRewinding);
```

```
enum TPlayerState ps_Closed, ps_Stopped, ps_Paused, ps_Playing, ps_PlayingBackward,  
ps_FastForwarding, ps_FastRewinding ;
```

Description

Type used by the [PlayerState](#) property and [OnPlayerStateChanged](#) events.

The possible values are:

ps_Closed

ps_Stopped
 ps_Paused
 ps_Playing
 ps_PlayingBackward
 ps_FastForwarding
 ps_FastRewinding
 ps_Downloading
 ps_DownloadCompleted
 ps_DownloadCancelled

Each value is greater than the previous one, so it is possible to test e.g. if the clip is playing in any direction by testing `PlayerState >= ps_Playing`.

See Also

[Player features](#) [TOnPlayerStateChanged](#) [TOnPlayerBufferingData](#) [AudioChannelRenderMode](#)
[AudioStreamNumber](#) [AutoStartPlayer](#) [AVIDuration](#) [AVIHeaderInfo](#) [AVIInfo](#) [AVIInfo2](#) [ClosePlayer](#)
[FastForwardPlayer](#) [IsPlayerAudioStreamAvailable](#) [IsPlayerVideoStreamAvailable](#) [Last_Clip_Played](#)
[MP4NeedsReindexing](#) [OnPlayerBufferingData](#) [OnPlayerEndOfStream](#) [OnPlayerStateChanged](#)
[OnPlayerUpdateTrackbarPosition](#) [OpenPlayer](#) [OpenPlayerAtFramePositions](#) [OpenPlayerAtTimePositions](#)
[PausePlayer](#) [PlayerAudioCodec](#) [PlayerAudioRendering](#) [PlayerDuration](#) [PlayerDVSize](#)
[PlayerFastSeekSpeedRatio](#) [PlayerFileName](#) [PlayerForcedCodec](#) [PlayerFrameCount](#) [PlayerFramePosition](#)
[PlayerFrameRate](#) [PlayerFrameStep](#) [PlayerRefreshPausedDisplay](#) [PlayerRefreshPausedDisplayFrameRate](#)
[PlayerSpeedRatio](#) [PlayerTimePosition](#) [PlayerTrackBar](#) [PlayerTrackBarScale](#) [PlayerTrackBarSynchron](#)
[PlayerVideoCodec](#) [RewindPlayer](#) [RunPlayer](#) [RunPlayerBackwards](#) [ShowDialog](#) [SourceStream](#) [StopPlayer](#)
[SynchronizationRole](#) [Synchronized](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoPlayableWhileRecording](#)
[VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Maximize Your Productivity with HelpNDoc's CHM Help File Creation Features](#)

TPlaylist

TPlaylist

[Prev](#)
[Next](#)

Type used by the [Playlist](#) function.

Declaration

TPlaylist=(pl_Add, pl_Remove, pl_Clear, pl_Loop, pl_NoLoop, pl_Play, pl_Stop, pl_Next, pl_Previous, pl_SortAlpha, pl_SortRevAlpha, pl_Random, pl_Sequential);

enum TPlaylist pl_Add, pl_Remove, pl_Clear, pl_Loop, pl_NoLoop, pl_Play, pl_Stop, pl_Next, pl_Previous, pl_SortAlpha, pl_SortRevAlpha, pl_Random, pl_Sequential ;

Description

See the [Playlist](#) function.

See Also

[Video formats](#) [GetPlaylist](#) [IsPlaylistActive](#) [OnPlayerEndOfPlaylist](#) [Playlist](#) [PlaylistIndex](#)

Created with the Standard Edition of HelpNDoc: [Experience the Power and Ease of Use of a Help Authoring Tool](#)

TPointGreyConfig

TPointGreyConfig

[Prev](#)
[Next](#)

Type used by the [PointGreyConfig](#) function.

Declaration

TPointGreyConfig=(pgr_SetRegister, pgr_GetRegister, pgr_SetBufferSize, pgr_GetBufferSize, pgr_SetFormat, pgr_GetFormat)

enum TPointGreyConfig pgr_SetRegister, pgr_GetRegister, pgr_SetBufferSize, pgr_GetBufferSize, pgr_SetFormat, pgr_GetFormat ;

Description

Possible values:

pgr_SetRegister
pgr_GetRegister
pgr_SetBufferSize
pgr_GetBufferSize
pgr_SetFormat
pgr_GetFormat

See the [PointGreyConfig](#) function.

Created with the Standard Edition of HelpNDoc: [Easily create EPub books](#)

TRawSampleCaptureLocation

TRawSampleCaptureLocation

[Prev](#)

[Next](#)

Type of the [RawSampleCaptureLocation](#) property.

Declaration

TRawSampleCaptureLocation=(rl_SourceFormat, rl_AfterCompression);

enum TRawSampleCaptureLocationrl_SourceFormat, rl_AfterCompression ;

Description

Value Meaning

nm_rl_SourceFormat The sample capture is inserted on the capture device output.

nm_rl_AfterCompression The sample capture is inserted after the audio or video compressor, when recording and compressing on the fly.

Created with the Standard Edition of HelpNDoc: [Create help files for the Qt Help Framework](#)

TRecordingMethod

TRecordingMethod

[Prev](#)

[Next](#)

Type of the [RecordingMethod](#) property.

Unit

[VidGrab](#)

Declaration

TRecordingMethod=(rm_AVI, rm_ASF, rm_SendToDV, rm_MKV, rm_FLV, rm_MP4, rm_WebM, rm_MPG, rm_Multiplexer, rm_MOV, rm_TS);

enum TRecordingMethod rm_AVI, rm_ASF, rm_SendToDV, rm_MKV, rm_FLV, rm_MP4, rm_WebM, rm_MPG, rm_Multiplexer, rm_MOV, rm_TS ;

Description

Value Meaning

rm_AVI the recording will create an AVI or MPEG file,
 rm_ASF the recording will create an ASF file,
 rm_SendToDV the audio/video streams will be sent to the DV device.
 rm_MKV similar to rm_AVI, but the recording will create a MKV file, see [AVI and MKV recording](#) (the Matroska muxer must be installed, it can be downloaded at <http://www.matroska.org/>)
 rm_FLV the recording will create a Flash video in FLV format, see [FLV recording](#)
 rm_MP4 the recording will create a .MP4 video clip, see [MP4 recording](#)
 rm_WebM the recording will create a .WebM video clip, see [WebM recording](#)
 rm_MPG the recording will create a .mpg file (MPEG)
 rm_Multiplexer used to record by using another third-party multiplexer specified with [SetMultiplexerFilterByName](#)
 rm_MOV the recording will create a .MOV file, use the same settings than for [MP4 recording](#)
 rm_TS the recording will create a .TS file (Transport Stream). Useful if the capture device outputs a transport stream and [RecordingInNativeFormat](#)=true
 rm_H264 the recording will create a .H264 video file (no audio)
 rm_MP3 the recording will create a .MP3 audio clip
 rm_WMA the recording will create a .WMA audio clip
 rm_WAV the recording will create a .WAV audio clip

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TSyncPreview](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last_Recording_FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: Make Documentation Review a Breeze with HelpNDoc's Advanced Project Analyzer

TRecordingTimer

TRecordingTimer

[Prev](#)
[Next](#)

Type of the [RecordingTimer](#) property.

Declaration

```
TRecordingTimer=(rt_Disabled, rt_RecordToNewFile, rt_StopRecording, rt_StartRecording);
```

```
enum TRecordingTimer rt_Disabled, rt_RecordToNewFile, rt_StopRecording, rt_StartRecording ;
```

Description

Type of the [RecordingTimer](#) property.

Possible values:

```
rt_Disabled  
rt_RecordToNewFile
```

rt_StopRecording
rt_StartRecording
rt_FrameCapture

Created with the Standard Edition of HelpNDoc: [Maximize Your CHM Help File Capabilities with HelpNDoc](#)

TRegistryRoot

TRegistryRoot

[Prev](#)

[Next](#)

Type of the Rootkey global property.

Unit

[VidGrab](#)

Declaration

TRegistryRoot=(RR_HKEY_CURRENT_USER, RR_HKEY_LOCAL_MACHINE);

enum TRegistryRoot RR_HKEY_CURRENT_USER, RR_HKEY_LOCAL_MACHINE ;

Description

Value Meaning

RR_HKEY_CURRENT_USER

RR_HKEY_LOCAL_MACHINE

Created with the Standard Edition of HelpNDoc: [Easily Add Encryption and Password Protection to Your PDFs](#)

TRGBSelector

TRGBSelector

[Prev](#)

[Next](#)

Used to select the R, G or B color.

Declaration

TRGBSelector=(rs_Red, rs_Green, rs_Blue);

enum RGBSelector rs_Red, rs_Green, rs_Blue;

Description

Type of the RGBSelector parameter in the [MotionDetector_CellColorIntensity](#) and [MotionDetector_GlobalColorIntensity](#) functions.

Used to select one of the R, G or B colors.

Created with the Standard Edition of HelpNDoc: [Streamline Your CHM Help File Creation with HelpNDoc](#)

TStreamType

TStreamType

[Prev](#)

[Next](#)

TStreamType

Description

TStreamType = (st_Video, st_Audio);

Created with the Standard Edition of HelpNDoc: [Easily create Qt Help files](#)

TSynchronizationRole

TSynchronizationRole

[Prev](#)
[Next](#)

Synchronization role

Declaration

TSynchronizationRole =(sr_Master, sr_Slave);

enum TSynchronizationRole sr_Master, sr_Slave ;

Description

Type of the [SynchronizationRole](#) function

Created with the Standard Edition of HelpNDoc: [Free iPhone documentation generator](#)

TSyncPreview

TSyncPreview

[Prev](#)
[Next](#)

Type of the [SyncPreview](#) property.

Unit

[VidGrab](#)

Declaration

TSyncPreview=(sp_Auto, sp_Disabled, sp_Enabled);

enum TSyncPreview sp_Auto, sp_Disabled, sp_Enabled ;

Description

Value Meaning

sp_Auto Automatically selected

sp_Disabled the sync of the audio rendering and video preview is disabled (this can cause time shift between the audio rendering and the video displayed)

sp_Enabled the sync of the audio rendering and video display is enabled (this can cause dropped frames and/or audio/video sync problems in the recorded AVI)

See Also

[Recording methods and properties](#) [TAVIMuxConfig](#) [TASFDeinterlaceMode](#) [TAutoFileName](#) [TOnRecordingCompleted](#) [TOnRecordingReadyToStart](#) [TRecordingMethod](#) [AudioRecording](#) [AudioSyncAdjustment](#) [AutoFileNameMinDigits](#) [AVIDurationUpdated](#) [AVIFormatOpenDML](#) [AVIHeaderInfo](#) [AVIInfo](#) [Encoder](#) [SetInt](#) [HoldRecording](#) [IsRecordingPaused](#) [Last_Recording_FileName](#) [OnBacktimedFramesCountReached](#) [OnCopyPreallocDataCompleted](#) [OnCopyPreallocDataProgress](#) [OnCopyPreallocDataStarted](#) [OnCreatePreallocFileCompleted](#) [OnCreatePreallocFileStarted](#) [OnDiskFull](#) [OnRecordingCompleted](#) [OnRecordingPaused](#) [OnRecordingReadyToStart](#) [OnRecordingStarted](#) [OnReencodingCompleted](#) [OnReencodingStarted](#) [PauseRecording](#) [PreallocCapFileCopiedAfterRecording](#) [PreallocCapFileEnabled](#) [PreallocCapFileName](#) [PreallocCapFileSizeInMB](#) [RecordingBacktimedFramesCount](#) [RecordingCanPause](#) [RecordingDuration](#) [RecordingFileName](#) [RecordingFileSizeMaxInMB](#) [RecordingFourCC](#) [RecordingHeight](#) [RecordingInNativeFormat](#) [RecordingKBytesWrittenToDisk](#) [RecordingMethod](#) [RecordingSize](#) [RecordingTimer](#) [RecordingTimerInterval](#) [RecordingWidth](#) [ResumeRecording](#) [SaveCompressorSettingsToDataString](#) [SetMultiplexerFilterByName](#) [StartAudioRecording](#) [StartRecording](#) [StartSynchronized](#) [StopRecording](#) [StoragePath](#) [Synchronized](#) [SyncPreview](#) [VideoPlayableWhileRecording](#)

Created with the Standard Edition of HelpNDoc: [Produce Kindle eBooks easily](#)

TTextOverlayGradientMode

TTextOverlayGradientMode

[Prev](#)

[Next](#)

Used to enable the gradient mode of the text overlays

Declaration

```
TTextOverlayGradientMode=(gm_Disabled, gm_Horizontal, gm_Vertical, gm_ForwardDiagonal, gm_BackwardDiagonal);
```

```
enum TTextOverlayGradientMode gm_Disabled, gm_Horizontal, gm_Vertical, gm_ForwardDiagonal, gm_BackwardDiagonal ;
```

Description

Enables enable the gradient mode of the text overlays, and select the orientation

gm_Disabled:

gradient mode disabled

gm_Horizontal:

gradient applied from left to right

gm_Vertical:

gradient applied for top to bottom

gm_ForwardDiagonal:

gradient applied from top-left to right-bottom

gm_BackwardDiagonal:

gradient applied from rop-right to left-bottom

Created with the Standard Edition of HelpNDoc: [Transform Your Documentation Workflow with HelpNDoc's Intuitive UI](#)

TThirdPartyFilterList

TThirdPartyFilterList

[Prev](#)

[Next](#)

Type used by the [third-party](#) functions.

Declaration

```
TThirdPartyFilterList=(tpf_VideoSource, tpf_VideoPreview, tpf_VideoRecording, tpf_AudioSource, tpf_AudioRendering, tpf_AudioRecording, tpf_VideoRenderer, tpf_AudioRenderer);
```

```
enum TThirdPartyFilterList tpf_VideoSource, tpf_VideoPreview, tpf_VideoRecording, tpf_AudioSource, tpf_AudioRendering, tpf_AudioRecording, tpf_VideoRenderer, tpf_AudioRenderer ;
```

Description

Type used by the [third-party](#) functions to specify the insertion point of the third-party filter:

THIRD-PARTY FILTERS USED AS VIDEO AND/OR AUDIO SOURCE

tpf_ThirdPartyVideoSource: the filter will be used as a video source if [VideoSource](#) = vs_ThirdPartyFilter (and as an audio source if it exposes also an audio pin and [AudioSource](#) = as_Default)

tpf_ThirdPartyAudioSource: the filter will be used as an audio source if [VideoSource](#) = vs_ThirdPartyFilter (if you are using also another third-party filter as video source and it exposes an audio

pin, set [AudioSource](#) = as_UseExternalAudio to ignore the audio pin of the video source filter and use the tpf_ThirdPartyAudioSource filter as audio source)

THIRD-PARTY FILTERS INSERTED IN THE NORMAL DIRECTSHOW GRAPH

Note: excepted the tpf_VideoRenderer and the tpf_AudioRenderer (that are must be renderers without output pins), all the filters must be "inline" filters with input pins and output pins.

tpf_VideoSource: the filter is inserted before the video recording/preview switching, so the effect is applied to both video preview and video recording.

tpf_VideoPreview: the filter is inserted on the video preview stream, after the recording/preview splitter. The effect is applied only to the video preview.

tpf_VideoRecording: the filter is inserted on the video recording stream, after the recording/preview splitter. The effect is applied only to the video recording.

tpf_AudioSource: the filter is inserted before the audio recording/rendering switching, so the effect is applied to both audio rendering and audio recording.

tpf_AudioRendering: the filter is inserted on the audio rendering stream, after the recording/rendering splitter. The effect is applied only to audio rendering.

tpf_AudioRecording: the filter is inserted on the audio recording stream, recording/rendering splitter. The effect is applied only to audio rendering.

tpf_VideoRenderer: the filter IS A VIDEO RENDERER FILTER. It is inserted at the end of the preview stream.

tpf_AudioRenderer: the filter IS AN AUDIO RENDERER FILTER. It is inserted at the end of the audio rendering stream.

tpf_ThirdPartyVideoSource: the filter IS A VIDEO SOURCE

tpf_ThirdPartyAudioSource: the filter IS AN AUDIO SOURCE

tpf_AddToGraph: the filter is added to the graph in order to be used if compatible

See Also

[TOnThirdPartyFilterConnected](#) [OnThirdPartyFilterConnected](#) [ThirdPartyFilter_AddToList](#) [ThirdPartyFilter_ClearList](#) [ThirdPartyFilter_Enable](#) [ThirdPartyFilter_RemoveFromList](#) [ThirdPartyFilter_ShowDialog](#)

Created with the Standard Edition of HelpNDoc: [Revolutionize Your Documentation Review with HelpNDoc's Project Analyzer](#)

TThreadSyncPoint

TThreadSyncPoint

[Prev](#)
[Next](#)

TThreadSyncPoint

Description

TThreadSyncPoint = (tsp_Starting_CanSetParent, tsp_Started_CanSetParent, tsp_Stop_UnSetParent);

tsp_Starting_CanSetParent, tsp_Started_CanSetParent:

the video window parent must be set from one of these events

tsp_Stop_UnSetParent:

the video window parent must be unset from this event

See Also

[Opening a clip or an IP URL from a background thread without blocking the main thread](#) [Player features](#) [TOnThreadSync](#) [EnableThreadMode](#) [OnThreadSync](#)

Created with the Standard Edition of HelpNDoc: [Create iPhone web-based documentation](#)

TTrackbarAction

TTrackbarAction

[Prev](#)
[Next](#)

TTrackbarAction type

Declaration

```
TTrackbarAction=(tba_MouseDown, tba_MouseUp, tba_KeyDown, tba_KeyUp);
```

```
enum TTrackbarAction tba_MouseDown, tba_MouseUp, tba_KeyDown, tba_KeyUp ;
```

Description

Type used by the [NotifyPlayerTrackbarAction](#) procedure.

See the player's [Trackbar](#) chapter for more information.

Created with the Standard Edition of HelpNDoc: [What is a Help Authoring tool?](#)

TTriState

TTriState

[Prev](#)
[Next](#)

Like a boolean, but with an third "undefined" state.

Unit

[VidGrab](#)

Declaration

```
TTriState=(ts_Undefined, ts_False, ts_True);
```

```
enum TTriState ts_Undefined, ts_False, ts_True ;
```

Description

Value

ts_Undefined

ts_False

ts_True

Created with the Standard Edition of HelpNDoc: [Experience the Power and Ease of Use of a Help Authoring Tool](#)

TTunerMode

TTunerMode

[Prev](#)
[Next](#)

Type of the TunerMode property

Description

Type of the TunerMode property.

The possible values are:

tm_TVTuner

tm_FMRadioTuner

tm_AMRadioTuner

tm_DigitalSatelliteTuner

TTVChannelInfo

TTVChannelInfo

[Prev](#)[Next](#)

Type passed to the [GetTVChannelInfo](#) function.

Declaration

```
TTVChannelInfo=(tci_Channel, tci_DefaultVideoFrequency, tci_OverriddenVideoFrequency,
tci_TunerVideoFrequency, tci_TunerAudioFrequency, tci_Locked);
```

```
enum TTVChannelInfo tci_Channel, tci_DefaultVideoFrequency, tci_OverriddenVideoFrequency,
tci_TunerVideoFrequency, tci_TunerAudioFrequency, tci_Locked ;
```

Description

Type of the desired channel info that will be returned by [GetTVChannelInfo](#).

tci_Channel:

Reports the current TV channel in use (previewing or capturing to AVI).

tci_DefaultVideoFrequency:

Reports the default video frequency for the current TV country code / tuner input type.
Can report -1 if a frequency override has been set and the default TV frequency is not saved in the registry.

tci_OverriddenVideoFrequency:

Reports the overridden frequency for this channel, if it has been previously set by invoking TVSetChannelFrequencyOverrideVidGrab.TVideoGrabber.TVSetChannelFrequencyOverride.

tci_TunerVideoFrequency:

Reports the current audio frequency, as reported by the TV tuner when selecting the TV channel.

tci_TunerAudioFrequency, :

Reports the current video frequency, as reported by the TV tuner when selecting the TV channel.

tci_Locked:

Returns 1 if the horizontal sync is locked for this channel, otherwise returns 0.

See Also

[TOnTVChannelScanStarted](#) [TOnTVChannelSelected](#) [TTunerInputType](#) [GetTVChannelInfo](#) [IsHorizontalSyncLocked](#) [IsTVAudioAvailable](#) [IsTVAutoTuneRunning](#) [IsTVTunerAvailable](#) [OnTVChannelScanCompleted](#) [OnTVChannelScanStarted](#) [OnTVChannelSelected](#) [TunerFrequency](#) [TunerMode](#) [TVChannel](#) [TVClearFrequencyOverrides](#) [TVCountryCode](#) [TVGetMinMaxChannels](#) [TVSetChannelFrequencyOverride](#) [TVStartAutoScan](#) [TVStopAutoScan](#) [TVTunerInputType](#) [TVUseFrequencyOverrides](#) [VCRHorizontalLocking](#)

Tv360_Angle

Tv360_Angle

[Prev](#)[Next](#)

Tv360_Angle

Declaration

Tv360_Angle=(v360_fov_Diagonal, v360_fov_Horizontal, v360_fov_Vertical);

enum Tv360_Angle v360_fov_Diagonal, v360_fov_Horizontal, v360_fov_Vertical ;

Description

Tv360_Angle enumeration

Created with the Standard Edition of HelpNDoc: [From Word to ePub or Kindle eBook: A Comprehensive Guide](#)

Tv360_InOut

Tv360_InOut

[Prev](#)

[Next](#)

Tv360_InOut

Declaration

Tv360_InOut=(v360_in, v360_out);

enum Tv360_InOut v360_in, v360_out ;

Description

Tv360_InOut enumeration

Created with the Standard Edition of HelpNDoc: [Free EPub producer](#)

Tv360_Interpolation

Tv360_Interpolation

[Prev](#)

[Next](#)

Tv360_Interpolation

Declaration

Tv360_Interpolation=(ipl_Default, ipl_Bilinear, ipl_Nearest, ipl_Lagrange9, ipl_Bicubic, ipl_Lanczos, ipl_Spline16, ipl_Gaussian, ipl_Mitchell);

enum Tv360_Interpolation ipl_Default, ipl_Bilinear, ipl_Nearest, ipl_Lagrange9, ipl_Bicubic, ipl_Lanczos, ipl_Spline16, ipl_Gaussian, ipl_Mitchell ;

Description

Tv360_Interpolation enumeration

Created with the Standard Edition of HelpNDoc: [Maximize Your Documentation Efficiency with a Help Authoring Tool](#)

TV360_MouseAction

TV360_MouseAction

[Prev](#)

[Next](#)

TV360_MouseAction

Declaration

```
TV360_MouseAction=(ma_Disabled, ma_MouseUp, ma_MouseMove);
```

```
enum TV360_MouseAction ma_Disabled, ma_MouseUp, ma_MouseMove ;
```

Description

```
TV360_MouseAction enumeration
```

Created with the Standard Edition of HelpNDoc: [Easy EBook and documentation generator](#)

Tv360_Projection**Tv360_Projection**[Prev](#)[Next](#)

Tv360_Projection

Declaration

```
Tv360_Projection=(
ipp_Equirectangular,
ipp_Cubemap_3_2,
ipp_Cubemap_6_1,
ipp_Equiangular,
ipp_Flat,
ipp_Dual_fisheye,
ipp_Barrel,
ipp_Cubemap_1_6,
ipp_Stereographic,
ipp_Mercator,
ipp_Ball,
ipp_Hammer,
ipp_Sinusoidal,
ipp_Fisheye,
ipp_Pannini,
ipp_Cylindrical,
ipp_Perspective,
ipp_Tetrahedron,
ipp_Barrel_split,
ipp_Tspyramid,
ipp_Hequirectangular,
ipp_Equisolid,
ipp_Orthographic,
ipp_Octahedron
);
```

enum Tv360_Projection

```
ipp_Equirectangular,
ipp_Cubemap_3_2,
ipp_Cubemap_6_1,
ipp_Equiangular,
ipp_Flat,
ipp_Dual_fisheye,
ipp_Barrel,
ipp_Cubemap_1_6,
```

```

ipp_Stereographic,
ipp_Mercator,
ipp_Ball,
ipp_Hammer,
ipp_Sinusoidal,
ipp_Fisheye,
ipp_Pannini,
ipp_Cylindrical,
ipp_Perspective,
ipp_Tetrahedron,
ipp_Barrel_split,
ipp_Tsipyramid,
ipp_Hequirectangular,
ipp_Equisolid,
ipp_Orthographic,
ipp_Octahedron
};

```

Description

Tv360_Projection enumeration

Created with the Standard Edition of HelpNDoc: [Converting Word Documents to eBooks: A Step-by-Step Guide with HelpNDoc](#)

Tv360_StereoFormat

Tv360_StereoFormat

[Prev](#)
[Next](#)

Tv360_StereoFormat

Declaration

Tv360_StereoFormat=(sf_2DMono, sf_SideBySide, sf_TopBottom);

enum Tv360_StereoFormat sf_2DMono, sf_SideBySide, sf_TopBottom ;

Description

Tv360_StereoFormat enumeration

Created with the Standard Edition of HelpNDoc: [Produce Kindle eBooks easily](#)

TVideoAlignment

TVideoAlignment

[Prev](#)
[Next](#)

Type of the [TextOverlay_VideoAlignment](#) and [ImageOverlay_VideoAlignment](#) properties

Description

TVideoAlignment = (oa_LeftTop, oa_CenterTop, oa_RightTop, oa_LeftCenter, oa_Center, oa_RightCenter, oa_LeftBottom, oa_CenterBottom, oa_RightBottom);

Created with the Standard Edition of HelpNDoc: [Revolutionize Your Documentation Output with a Help Authoring Tool](#)

TVideoDeinterlacing

TVideoDeinterlacing

[Prev](#)
[Next](#)

Type of the [VideoProcessing_Deinterlacing](#) property.

Unit

[VidGrab](#)

Declaration

TVideoDeinterlacing=(di_Disabled, di_HalfSize, di_FullSize);

enum TVideoDeinterlacing di_Disabled, di_HalfSize, di_FullSize ;

Description

di_Disabled : deinterlacing is disabled

di_HalfSize : half-size deinterlacing

di_FullSize : full size deinterlacing.

di_DScaler : uses the DScaler deinterlacer

di_AVISynth : uses the AVISynth deinterlacer (AVISynth must be installed)

di_FFDSHOW : uses the FFDSHOW deinterlacer (FFDSHOW must be installed)

di_ThirdPartyDeinterlacer: uses the deinterlacer specified to the ThirdPartyDeinterlacer property

See the [Deinterlacing](#) chapter for more information.

See Also

[MP4NeedsReindexing](#) [OnFrameBitmap](#) [RetrievalInitialXYAfterRotation](#) [VideoProcessing_Brightness](#) [VideoProcessing_Contrast](#) [VideoProcessing_Deinterlacing](#) [VideoProcessing_FlipHorizontal](#) [VideoProcessing_FlipVertical](#) [VideoProcessing_GrayScale](#) [VideoProcessing_Hue](#) [VideoProcessing_InvertColors](#) [VideoProcessing_Pixellization](#) [VideoProcessing_Rotation](#) [VideoProcessing_RotationCustomAngle](#) [VideoProcessing_Saturation](#) [TVideoRotation](#)

Created with the Standard Edition of HelpNDoc: [Qt Help documentation made easy](#)

TVideoRenderer

TVideoRenderer

[Prev](#)

[Next](#)

Type of the [VideoRenderer](#) property.

Declaration

TVideoRenderer=(vr_AutoSelect, vr_VMR9, vr_VMR7, vr_StandardRenderer, vr_OverlayRenderer, vr_None);

enum TVideoRenderer vr_AutoSelect, vr_VMR9, vr_VMR7, vr_StandardRenderer, vr_OverlayRenderer, vr_None ;

Description

TVideoRenderer = (vr_AutoSelect, vr_EVR, vr_VMR9, vr_VMR7, vr_StandardRenderer, vr_OverlayRenderer, vr_RecordingPriority, vr_None, vr_madVR, vr_DatasteadRealtimeRenderer);

vr_AutoSelect: selected automatically among the values below, depending on the context

vr_None: no video renderer

vr_RecordingPriority: suitable for general use

vr_DatasteadRealtimeRenderer: low latency, real-time renderer. Requires the [Datatstead RTSP/RTMP/HTTP/ONVIF Source filter](#) to be intalled.

vr_StandardRenderer: standard DirectShow renderer

vr_VMR7: VMR7 DirectShow renderer

vr_VMR9: VMR9 DirectShow renderer, requires DirectX to be enabled

vr_EVR: Enhanced DirectShow renderer, good quality but higher CPU consumption

vr_OverlayRenderer: deprecated

vr_madVR: deprecated

See Also

[AdjustPixelAspectRatio](#) [Display_Active](#) [Display_AlphaBlendEnabled](#) [Display_AlphaBlendValue](#) [Display_AutoSize](#) [Display_Embedded](#) [Display_FullScreen](#) [Display_Height](#) [Display_Left](#) [Display_Monitor](#) [Display_MouseMovesWindow](#) [Display_PanScanRatio](#) [Display_SetLocation](#) [Display_StayOnTop](#) [Display_Top](#) [Display_TransparentColorEnabled](#) [Display_TransparentColorValue](#) [Display_VideoHeight](#) [Display_VideoPortEnabled](#) [Display_VideoWidth](#) [Display_VideoWindowHandle](#) [Display_Width](#) [DualDisplay_Active](#) [DualDisplay_AlphaBlendEnabled](#) [DualDisplay_AlphaBlendValue](#) [DualDisplay_AutoSize](#) [DualDisplay_Embedded](#) [DualDisplay_FullScreen](#) [DualDisplay_Height](#) [DualDisplay_Left](#) [DualDisplay_Monitor](#) [DualDisplay_MouseMovesWindow](#) [DualDisplay_PanScanRatio](#) [DualDisplay_StayOnTop](#) [DualDisplay_Top](#) [DualDisplay_TransparentColorEnabled](#) [DualDisplay_TransparentColorValue](#) [DualDisplay_VideoHeight](#) [DualDisplay_VideoPortEnabled](#) [DualDisplay_VideoWidth](#) [DualDisplay_VideoWindowHandle](#) [DualDisplay_Visible](#) [DualDisplay_Width](#) [IsVideoPortAvailable](#) [Monitor_Primary](#) [Index](#) [MonitorBounds](#) [MonitorsCount](#) [OnLeavingFullScreen](#) [SetParentWindow](#) [SetWindowTransparency](#) [VideoDoubleBuffered](#) [VideoHeight](#) [PreferredAspectRatio](#) [VideoRendererExternal](#) [VideoRendererExternalIndex](#) [VideoRendererPriority](#) [VideoVisibleWhenStopped](#) [VideoWidth](#) [PreferredAspectRatio](#)

Created with the Standard Edition of HelpNDoc: [Transform Your Help Documentation Process with a Help Authoring Tool](#)

TVideoRendererPriority

TVideoRendererPriority

[Prev](#)[Next](#)

Type of the [VideoRendererPriority](#) property

Description

TVideoRendererPriority = (vrp_Speed, vrp_Quality, vrp_Auto);

vrp_Speed: favors performance and smoothness. The video quality may be lower especially when the video window is downsized.

vrp_Quality: favors video quality. Requires more CPU.

vrp_Auto: vrp_Speed or vrp_Quality is selected automatically depending on the context.

Created with the Standard Edition of HelpNDoc: [Free help authoring tool](#)

TVideoRotation

TVideoRotation

[Prev](#)[Next](#)

Type of the [VideoProcessing_Rotation](#) property.

Unit

[VidGrab](#)

Declaration

TVideoRotation=(rt_0_deg, rt_90_deg, rt_180_deg, rt_270_deg, rt_0_deg_mirror, rt_90_deg_mirror, rt_180_deg_mirror, rt_270_deg_mirror);

enum TVideoRotation rt_0_deg, rt_90_deg, rt_180_deg, rt_270_deg, rt_0_deg_mirror, rt_90_deg_mirror, rt_180_deg_mirror, rt_270_deg_mirror ;

Description

Type of the [VideoProcessing_Rotation](#) property.

See [VideoProcessing_Rotation](#).

See Also

[TVideoDeinterlacing](#) [MP4NeedsReindexing](#) [OnFrameBitmap](#) [RetrieveInitialXYAfterRotation](#)

[VideoProcessing_Brightness](#)
[VideoProcessing_Contrast](#)
[VideoProcessing_Deinterlacing](#)
[VideoProcessing_FlipHorizontal](#)
[VideoProcessing_FlipVertical](#)
[VideoProcessing_GrayScale](#)
[VideoProcessing_Hue](#)
[VideoProcessing_InvertColors](#)
[VideoProcessing_Pixellization](#)
[VideoProcessing_Rotation](#)
[VideoProcessing_RotationCustomAngle](#)
[VideoProcessing_Saturation](#)

Created with the Standard Edition of HelpNDoc: Streamline Your Documentation Process with HelpNDoc's Project Analyzer

TVideoSource

TVideoSource

[Prev](#)
[Next](#)

Type of the [VideoSource](#) property.

Unit

[VidGrab](#)

Declaration

TVideoSource=(vs_VideoCaptureDevice, vs_ScreenRecording, vs_VideoFileOrURL, vs_JPEGsOrBitmaps, vs_VideoFromImages);

enum TVideoSource vs_VideoCaptureDevice, vs_ScreenRecording, vs_VideoFileOrURL, vs_JPEGsOrBitmaps, vs_VideoFromImages ;

Description

Value Meaning

vs_VideoCaptureDevice,
 vs_ScreenRecording,
 vs_VideoFileOrURL,
 vs_JPEGsOrBitmaps,
 vs_IPCamera,
 vs_Mixer,
 vs_VideoFromImages,
 vs_ThirdPartyFilter,
 vs_StreamInterface

These values are explained in the [VideoSource](#) property.

See Also

[Video sources supported for preview and recording](#)
[VideoSource](#)
[VideoSources](#)
[VideoSourcesCount](#)

Created with the Standard Edition of HelpNDoc: Effortlessly Spot and Fix Problems in Your Documentation with HelpNDoc's Project Analyzer

TVMR9ImageAdjustment

TVMR9ImageAdjustment

[Prev](#)
[Next](#)

Type of the VMR9 image adjustment settings.

Unit

[VidGrab](#)

Declaration

TVMR9ImageAdjustment=(vmr9_Brightness, vmr9_Contrast, vmr9_Hue, vmr9_Saturation, vmr9_Alpha);


```
enum TVMR9ImageAdjustment vmr9_Brightness, vmr9_Contrast, vmr9_Hue, vmr9_Saturation, vmr9_Alpha
;
```

Description

Value Meaning

vmr9_Brightness
vmr9_Contrast
vmr9_Hue
vmr9_Saturation
vmr9_Alpha

See Also

[GetVMR9ImageAdjustmentBounds](#) [IsVMR9ImageAdjustmentAvailable](#) [SetVMR9ImageAdjustmentValue](#)

Created with the Standard Edition of HelpNDoc: [Easily create EBooks](#)

TVUMeter

TVUMeter

[Prev](#)

[Next](#)

Type of the [VUMeter](#) property.

Declaration

```
TVuMeter=(vu_Disabled, vu_Analog, vu_Bargraph);
```

```
enum TVuMeter vu_Disabled, vu_Analog, vu_Bargraph ;
```

Description

Value Meaning

vu_Disabled : VU-Meter disabled
vu_Analog : analog VU-Meter
vu_Bargraph : bargraph VU-Meter
vu_AnalogOverlay: analog VU-meter overlayed over the video frames
vu_BargraphOverlay: bargraph VU-meter overlayed over the video frames

See Also

[TVUMeterSetting](#) [AudioBalance](#) [AudioChannelRenderMode](#) [AudioVolume](#) [MuteAudioRendering](#)
[SpeakerBalance](#) [SpeakerControl](#) [SpeakerVolume](#) [VUMeter](#)

Created with the Standard Edition of HelpNDoc: [Elevate your documentation to new heights with HelpNDoc's built-in SEO](#)

TVUMeterSetting

TVUMeterSetting

[Prev](#)

[Next](#)

Type used in the [SetVUMeterSetting](#) method.

Declaration

```
TVuMeter=(vu_Handle, vu_WarningPercent, vu_PeakPercent, vu_BkgndColor, vu_NormalColor,
vu_WarningColor, vu_PeakColor, vu_TickSize, vu_TickInterval, vu_NeedleThickness);
```

```
enum TVuMeter vu_Handle, vu_WarningPercent, vu_PeakPercent, vu_BkgndColor, vu_NormalColor,
vu_WarningColor, vu_PeakColor, vu_TickSize, vu_TickInterval, vu_NeedleThickness ;
```

Description

Type used in the [SetVUMeterSetting](#) method.

Value Meaning

vu_Handle : sets the Handle of the panel or image control on which the VUMeter w
vu_WarningPercent : percentage of the level above which the vu_WarningColor is use
vu_PeakPercent : percentage of the level above which the vu_PeakColor is used
vu_BkgndColor : color of the VU-meter background
vu_NormalColor : color of the normal level (green by default)
vu_WarningColor : color of the warning level (above the vu_WarningPercent)
vu_PeakColor : color of the peak level (above the vu_PeakPercent)
vu_TickSize : size of ticks in the bargraph VU-Meter
vu_TickInterval : interval between ticks in the bargraph VU-Meter
vu_NeedleThickness : thickness of the needle in the analog VU-Meter
vu_OverlayLeft : left location of the VU-Meter when overlayed over the video fra
vu_OverlayTop : top location of the VU-Meter when overlayed over the video fram
vu_OverlayWidth : width of the VU-Meter when overlayed over the video frames
vu_OverlayHeight : height of the VU-Meter when overlayed over the video frames
vu_Transparent(*) : enables/disables the transparency of the background of the VU-M
vu_FlipVert(*) : flips the VUMeter vertically when overlayed over the video fram
vu_FlipHorz(*) : flips the VUMeter horizontally when overlayed over the video fr
vu_CustomPercentValue : the VUMeter as a custom bargraph. Just pass (or update in real
vu_LogarithmicScale(*) : the VU-meter scale is logarithmic (linear by default)

(*) 0 = disabled, 1 = enabled

See Also

[TVUMeter](#) [AudioBalance](#) [AudioChannelRenderMode](#) [AudioVolume](#) [MuteAudioRendering](#) [SpeakerBalance](#)
[SpeakerControl](#) [SpeakerVolume](#) [VUMeter](#)

Created with the Standard Edition of HelpNDoc: [Produce electronic books easily](#)

TWebcamStillCaptureButton

TWebcamStillCaptureButton

[Prev](#)[Next](#)

Type of the WebcamStillCaptureButton property.

Declaration

TWebcamStillCaptureButton=(wb_Disabled, wb_Enabled);

enum WebcamStillCaptureButton wb_Disabled, wb_Enabled ;

Description

Type of the [WebcamStillCaptureButton](#) property.

Created with the Standard Edition of HelpNDoc: [Generate Kindle eBooks with ease](#)
