Authorware Help
Director Help

How to Order & Register
Licensing & Availability
Technical Support

Direct Media Xtra HELP

For more up-to-date information please visit our web site:
xtras.tabuleiro.com

Authorware Help

Direct Media Xtra
HELP

Getting Started

DirectMediaXtra is an AssetXtra. AssetXtras can be manipulated using the design and presentation windows, and their properties can be adjusted through scripting, just like Authorware's built-in media types.

DirectMediaXtra will take advantage of Microsoft’s DirectDraw and DirectSound if they are available on the playback system. The Xtra will always try to play the Media file using the fastest color depth and method available on the machine. The result is that Media files can play using thousands or millions of colors even if your Executable and Display are set to 256 colors. The DirectDraw engine will use overlay technology to write directly to the memory of the display adapter if this feature is available, resulting in faster and enhanced playback.

To create a DirectMediaXtra cast member, go to your INSERT menu, select TABULEIRO XTRAS->DirectMediaXtra. The standard DirectMediaXtra Properties Dialog Box will appear. Select browse. Another Dialog Box will appear, prompting you to choose the video file associated with this cast member. You can select MPEG, AVI, Quicktime, WAV, AIF, MP2 and MIDI files. If Windows Media Player is installed you can also select Windows Media files compressed in the ASF, WMV or WMA formats. Select a file with the correct type and extension. Click OK. You will return to the Properties Dialog Box. The file information should be displayed correctly, indicating the duration, height and width of the media file.

You can preview the file, mark cue points and set the playback options (toolbar, volume, rate, balance, among others).

Leave the Playback options as they are, for now. Finish by clicking OK.

A new icon will appear in your Design Window in Authorware. It has the DirectMediaXtra icon. Double-Click on it to adjust the position and size of the video in the Presentation window.

IMPORTANT: if the menu bar is enabled, the position of the video will be 20 pixels offset.

That's it!

Cue Points

DirectMediaXtra supports cue points as introduced in Director 6. Cue points can be marked in the Properties Dialog box that appears when you create or when you double-click a DirectMediaXtra cast member. Cue Point support works natively in Director 6, and through the Event interface in Authorware.

Each DirectMediaXtra can have up to 64 different cue points marked. Cue points times are measured in milliseconds, and the name of the cue point can have up to 32 characters. To create a cue point you should first locate the time using the preview movie in the Properties Dialog window. Then select "ADD". A window will appear for you to define the name of the cue point. If you need to modify a cue point later you can double-click its name on the list of cue points and edit its name and time directly. You can use the "GOTO" button to check the location of a cue point, and the "REMOVE" button can be used to delete a cue point.

DirectMediaXtra does not modify the video file in any way. The cue points are maintained by your DirectMediaXtra cast members, so you do not need to leave Authorware or mark them again if the original video is recompressed.

Creating Cue Points from Scripting

There are two functions that you can use to create and remove cue points from scripting.

```
CallIcon(@"IconName", #addcuepoint, cuepointtime,
"cuepointname")
```

```
CallIcon(@"IconName", #removecuepoint, cuepointindex)
```

Using Cue Points

DirectMediaXtra cue points can be used to control several aspects of playback. For advanced scripting users, the following properties are supported:

isPastCuePoint(sprite)	the mostRecentCuePoint of sprite
the cuepointtimes of icon	the cuepointnames of icon

You must setup an Event Interaction in order to be able to intercept cue points while the Media file is playing. The sample piece included with the Xtra shows how to do this. Basically, you should create an response of type "Event". Select the "Sender" Icon, and double-click the "CuePointPassed" event in the "Event Name" field. Your response branch will be called each time a new cue point is passed. To get information about the event you can use the EventLastMatched@"Response" format , as shown in the sample movie. Please consult Authorware’s documentation to learn more about the EventLastMatched variable.

Authorware Help

Icon Properties

All properties of the DirectMediaXtra icon can be easily set in the Properties Dialog box, which is displayed when the icon is created and also when you double-click the icon. These properties can also be set via scripting.

File - the complete pathname of the linked media file.
Example: result:=GetIconProperty(@"DirectMediaXtra", #file)
Trace(result)
-- "C:\myfile\file1.mpg"
SetIconProperty(@"DirectMediaXtra", #file,"C:\myfile2.mpg")

Title - the filename of the linked Media file. This property is used to resolve the pathname if the Media file is not found on the original location.
Example: result:=GetIconProperty(@"DirectMediaXtra", #title)
Trace(result)
-- "file.mpg"

Duration - the duration of the linked Media file in milliseconds.
Example: result:=GetIconProperty(@"DirectMediaXtra", #duration)
Trace(result)
-- 45200

Videowidth - the original encoded width of the linked Media file, in pixels. This is a read-only property.
Example: result:=GetIconProperty(@"DirectMediaXtra", #videowidth)
Trace(result)
-- 352

Videoheight - the original encoded height of the linked Media file, in pixels. This is a read-only property.
Example: result:=GetIconProperty(@"DirectMediaXtra", #videoheight)
Trace(result)
-- 240

Volume - The default audio volume to be used during playback of new Xtra sprites. Values range from -100 to 0, expressed in dB. The sprite function SetVolume may be used when you want to modify the volume of a sprite that is already on the stage.
Example: result:=GetIconProperty(@"DirectMediaXtra", #volume)
Trace(result)
-- -10
SetIconProperty(@"DirectMediaXtra", #volume,0)

Balance - The default audio balance to be used during playback of new Xtra sprites. Values range from -100 to 100, expressed in dB. The sprite function SetBalance may be used when you want to modify the balance of audio of a sprite that is already on the stage.
Example: result:=GetIconProperty(@"DirectMediaXtra", #balance)
Trace(result)
-- -100
SetIconProperty(@"DirectMediaXtra", #balance,0)

Rate - The default playback rate to be used during playback. Values range from 0 to 200, expressed in percentage of the original playback rate. The sprite function SetRate may be used when you want to modify the rate of playback of a sprite that is already on the stage. IMPORTANT: Windos Media files do not support the rate property. Playback will fail if the rate is set to any value but the default of 100.
Example: result:=GetIconProperty(@"DirectMediaXtra", #rate)
Trace(result)
-- 100
SetIconProperty(@"DirectMediaXtra", #rate,80)

Cpuboost - DirectMediaXtra can slow down Director to allow more CPU time for the DirectShow driver. This can be useful if you are experiencing poor or stuttering playback. Legal values are in the range of 0 to 95 percent, although values over 80 may result in problems with the performance of your Director animation and/or Lingo event detection.
Example: SetIconProperty(@"DirectMediaXtra", #cpuboost,60)

AdjustDurationBeforePlayback - If this property is TRUE, the Xtra will scan the media file and adjust the duration of member property as soon as the sprite starts playing. This is necessary because different DirectShow filters may report different durations for the same clip. If this happens when you take your finished product to a machine with a different DirectShow filter, the Xtra may not reach certain cuepoints or it may consider that the end of the file was not reached. Depending on your scripting this can prevent your movie from operating correctly, so this setting should be enabled in the majority of cases.
Example:
SetIconProperty(@"DirectMediaXtra",#adjustdurationbeforeplayback,1)

ShowLocateFileDialog - the Xtra will try to locate the linked media files at runtime if you move them from the original location. The mechanism is described in the CREATING PROJECTORS section of the documentation. If the linked file can not be found, this property is used to specify if a dialog box is displayed asking the user to locate the file, or if the Xtra fails silently.
Example: SetIconProperty(@"DirectMediaXtra", #showlocatefiledialog,TRUE)

LinkedMediaLocated - This is a property that is set automatically by the Xtra, and indicates if the linked media file for this member was located when the movie loaded (either automatically by the Xtra, or using the Locate File dialog box.) The Xtra will not attempt to initialize playback of a linked file if this property is FALSE.
Example: result:=GetIconProperty(@"DirectMediaXtra", #linkedmedialocated)
Trace(result)
-- 1

HasToolbar - Indicates whether a toolbar is attached to the video window. Possible values are 0 (FALSE) or 1 (TRUE).
Example: SetIconProperty(@"DirectMediaXtra", #hastoolbar,1)

Loop - Indicates whether the linked Media file will loop automatically. Possible values are 0 (FALSE) or 1 (TRUE).
Example: SetIconProperty(@"DirectMediaXtra", #loop,1)

PlaySegment - Indicates whether only a segment of the Media file is to be played. Possible values are 0 (FALSE) and 1 (TRUE).
Example: SetIconProperty(@"DirectMediaXtra", #playsegment,1)

SegmentStart - Specifies the start time of the Media file to be played, in milliseconds. Only active if "the playsegment of the member" is TRUE.
Example: SetIconProperty(@"DirectMediaXtra", #segmentstart,10700)

SegmentEnd - Specifies the end time of the Media file to be played, in milliseconds. Only active if "the playsegment of the member" is TRUE.
Example: SetIconProperty(@"DirectMediaXtra", #segmentend,45800)

PausedAtStart - Indicates whether the Media file should be opened in "paused" state. Possible values are 0 (FALSE) and 1 (TRUE).
Example: SetIconProperty(@"DirectMediaXtra", #pausedatstart,1)

CuePointTimes - Contains a list of the cue point times associated with this member.
Example: result:=GetIconProperty(@"DirectMediaXtra", #cuepointtimes)
Trace(result)
-- [4500,12300,34650,67321]

CuePointNames - Contains a list of the cue point names associated with this member.
Example: result:=GetIconProperty(@"DirectMediaXtra", #cuepointnames)
Trace(result)
-- ["Cue 1","NewCuePoint","Start Point","Second part"]

Authorware Help

Sprite Properties

The Sprite properties are only accessible when a DirectMediaXtra sprite is being displayed on the presentation window.

Currenttime - The current position of the Media file, in milliseconds.
Example: result:=GetSpriteProperty(@"DirectMediaXtra", #currenttime)
Trace(result)
-- *22345*

Mediabusy - Indicates whether the Media file is playing. Possible values are 0 (FALSE) and 1 (TRUE) .
Example: result:=GetSpriteProperty(@"DirectMediaXtra", #mediabusy)
Trace(result)
-- *1*

MostRecentCuePoint - The index number of the last cue point passed.
Example: result:=GetSpriteProperty(@"DirectMediaXtra", #mostrecentcuepoint)
Trace(result)
-- *2*

Icon Functions

DirectMediaXtra provides several functions that can be called from scripting to retrieve information about the Media linked to the icon.

CallIcon(@"IconName", #addcuepoint, cuepointtime, "cuepointname") - This function can be used to create a new cue point from scripting. Cue points are automatically sorted for you.
Example: CallIcon(@"DirectMediaXtra", #addcuepoint, 10000, "newcue")

CallIcon(@"IconName", #removecuepoint, cuepointindex) - This function can be used to remove a cue point. You should supply the index of the cue point to be removed.
Example: CallIcon(@"DirectMediaXtra", #removecuepoint, 1)

CallIcon(@"IconName", #scanfile) - This function can be used to extract the correct duration and dimensions of a video file when the linked media is changed via scripting. The file is queried and the castmember properties are updated automatically. With DirectMediaXtra 1.4 we introduced the adjustdurationbeforeplayback property, which also adjusts the duration, but does not impose an additional time penalty (the file is not opened twice), so it is generally better to leave it checked instead of using scanfile().
Example: CallIcon(@"DirectMediaXtra", #scanfile)

CallIcon(@"IconName", #register, registration number) -This function can be used in Projectors to register DirectMediaXtra at runtime, and allow developers to save the piece. It is not necessary for the normal operation of the Xtra.
Example: CallIcon(@"DirectMediaXtra", #register,"SERIALNUMBERHERE")

CallIcon(@"IconName", #isDirectShowInstalled) - Returns 1 (TRUE) or 0 (FALSE). Can be used to determine if DirectShow (formerly ActiveMovie) is installed on the user's machine. You should use a valid DirectMediaXtra icon to access this function.
Example: result:=CallIcon(@"DirectMediaXtra", #isDirectShowInstalled)
Trace(result)
-- *1*

Sprite Functions

The following spritel functions can be called from scripting to control playback of DirectMediaXtra sprites.

CallSprite(@"IconName", #videoplay) - Used to start or resume playback of a Media file. Playback is usually started automatically when the sprite is loaded, unless the "Paused at Start" option is checked.
Example: CallSprite(@"DirectMediaXtra ", #videoplay)

CallSprite(@"IconName", #videopause) - Used to pause playback of a Media file.
Example: CallSprite(@"DirectMediaXtra ", #videopause)

CallSprite(@"IconName", #videoseek, time) - Seeks the video to a different time in the Media file, and pauses the video. Time should be expressed in milliseconds.
Example: CallSprite(@"DirectMediaXtra ", #videoseek, 1500)

CallSprite(@"IconName", #videoplaysegment , starttime, endtime) - Plays a segment of the Media file. Please note that using this option actually limits access to the times of the stream that falls outside the playback segment boundaries. You should use another videoplaysegment() command to seek to a different time on the stream.
Example: CallSprite(@"DirectMediaXtra ", #videoplaysegment, 10000, 20000)

CallSprite(@"IconName", #isPastCuePoint , cuepointindex) - Returns TRUE if the cue point indicated has already passed, FALSE otherwise. DirectMediaXtra only supports this function used with cuepoint indexes, not names.
Example:result:=CallSprite(@"DirectMediaXtra ", #isPastCuePoint, 2)
Trace(result)
-- 1

CallSprite(@"IconName", #setvolume, volume) - Sets the volume of audio playback during the current sprite span. Valid volumes are in the range of -100 to 0 dB.
Example: CallSprite(@"DirectMediaXtra ", #setvolume , 0)

CallSprite(@"IconName", #getvolume) - Returns the current volume of audio playback, in dB.
Example: result:=CallSprite(@"DirectMediaXtra ", #getvolume)
Trace(result)
-- -20

CallSprite(@"IconName", #setbalance, balance) - Sets the balance of audio playback during the current sprite span. Valid values are in the range of -100 to 100 dB.
Example: CallSprite(@"DirectMediaXtra ", #setvolume , 0)

CallSprite(@"IconName", #getbalance) - Returns the current balance of audio playback, in dB.
Example: result:=CallSprite(@"DirectMediaXtra ",

#getbalance)
Trace(result)
-- -100

CallSprite(@"IconName", #setrate , rate) - Sets the rate of playback during the current sprite span. Valid values are in the range of 0 to 200 %.
Example: CallSprite(@"DirectMediaXtra ", #setrate , 80)

CallSprite(@"IconName", #getrate) - Returns the current playback rate, in %.
Example: result:=CallSprite(@"DirectMediaXtra ", #getrate)
Trace(result)
-- 100

CallSprite(@"IconName", #setfullscreen) - This function will trigger the ActiveMovie/DirectShow full-screen video renderer. The DirectMediaXtra sprite will be promoted to full-screen playback and a new video surface will be created. The full-screen renderer works independently from the authoring application, and it will be full-screen even if the presentation window does not fill the whole monitor. Mouse event detection and keypresses are NOT passed to the authoring application, so you should use this function in conjunction with cuepoints to return control to the authoring application. Remember to call RemoveFullScreen() when you are done with playback. On most applications it is better to simply stretch the DirectMediaXtra sprite to cover the whole presentation window area: you will still benefit from any video driver acceleration and interactivity control will be retained.
Example: CallSprite(@"DirectMediaXtra ", #setfullscreen)

CallSprite(@"IconName", #removefullscreen) - This function will disable the ActiveMovie/DirectShow full-screen video renderer. The DirectMediaXtra sprite will return to its original window on the presentation window in Authorware.
Example: CallSprite(@"DirectMediaXtra ", #removefullscreen 1)

Packaged Files

DirectMediaXtra icons operate only under Windows 95, 98, ME, Windows NT 4 and Windows 2000. Please check our web site at http://xtras.tabuleiro.com for updated information about system requirements for different operational systems. You should also deliver the DirectMediaXtra.x32 file in a folder named XTRAS, located in the same directory of your executable.

The final structure of files should look like:

Projector - C:\myproj\proj.exe

.INI file - C:\myproj\proj.ini

Lingo.ini - C:\myproj\lingo.ini

The Xtra will automatically locate all linked Media files that are located in the same directory of the executable, or in any directory listed in Authorware's searchpath variable. Additional directories can also be specified using the SEARCHPATHS list attached to each DirectMediaXtra castmember. This list can be edited directly from the DirectMediaXtra castmember properties dialog box and is saved with the castmember in the Director movie. Both absolute and relative directory entries may be used. Relative directories are evaluated in relation to the packaged executable, while absolute directories must contain a drive letter and complete pathname information.

Examples:

Relative directory entry: media

Absolute directory entry: E:\media

Director Help

Getting Started

DirectMediaXtra is an Asset Xtra. Unlike XObjects or Lingo Xtras, Asset Xtras can be manipulated using the score and cast windows, and their properties can be adjusted through Lingo, just like Director's built-in media types.

Before starting, you should configure Director's authoring environment to properly display external palettes in 256 color mode. DirectMediaXtra needs to access the current palette when Director is running in 256 colors. This can be enabled by editing the DIRECTOR.INI file, which is located in the same directory of your Director application. You need to edit the palette section to look like:

```
[Palette]
Animation=0
; 1: allows director to take control of the palette for fast palette effects
; and transitions.
; 0: prevents director from taking control of the palette so other apps,
; wallpaper etc. look better but palette effects and transitions will
; be slower.
```

You will need to include a copy of this modified DIRECTOR.INI file with your Projectors (read the instructions on the "Creating Projectors" section).

It is also a good idea to enable the "Animate in Background" option. This can be done by selecting the FILE->PREFERENCES->GENERAL menu in Director, and checking the "Animate in Background" option.

DirectMediaXtra will take advantage of Microsoft’s DirectDraw and DirectSound if they are available on the playback system. The Xtra will always try to play the Media file using the fastest color depth and method available on the machine. The result is that Media files can play using thousands or millions of colors even if your Projector and Display are set to 256 colors. The DirectDraw engine will use overlay technology to write directly to the memory of the display adapter if this feature is available, resulting in faster and enhanced playback.

To create a DirectMediaXtra cast member, go to your INSERT menu, select TABULEIRO XTRAS->DirectMediaXtra. The standard DirectMediaXtra Properties Dialog Box will appear. Select browse. Another Dialog Box will appear, prompting you to choose the video file associated with this cast member. You can select MPEG, AVI, Quicktime, WAV, AIF, MP2 or MIDI files. If Windows Media Player is installed you can also select Windows Media files compressed in the ASF, WMV or WMA formats. Select a file with the correct type and extension. Click OK. You will return to the Properties Dialog Box. The file information should be displayed correctly, indicating the duration, height and width of the media file.

You can preview the file, mark cue points and set the playback options (toolbar, volume, rate, balance, among others).

Leave the Playback options as they are, for now. Finish by clicking OK.

A new cast member will appear in your Cast Window in Director. It has the DirectMediaXtra icon, and the name of your linked video file. Click on it and drag the cast member to the score. A new sprite will be created on your score window, and you can adjust the position and the size of the video directly in the score.

You can use the "Wait for Cue Point" option in the tempo channel to wait for the end of the playback, or you can use a:

```
on exitframe
go the frame
end
```

script on the script channel to make Director loop on the frame while the video plays. That's it!

Note: the "Wait for cue point" option will not operate correctly if your DirectMediaXtra sprite is in the last frame of the score. The same behavior is true for built-in digital video and audio sprites.

Using Behaviors

Our technical support staff is constantly updating and creating new behaviors that can be used with Director 6 and later to control DirectMediaXtra sprites. Behaviors are available for free at the SUPPORT area on our web site at http://xtras.tabuleiro.com , even if you are using an evaluation copy of the Xtra. If you have a behavior that you want to share with the other developers using DirectMediaXtra we encourage you to login to our SUPPORT area and submit your creations.

Director Help

Cue Points

DirectMediaXtra supports cue points as introduced in Director 6. Cue points can be marked in the Properties Dialog box that appears when you create or when you double-click a DirectMediaXtra cast member. (Note: cue points can not be used with Director 5)

Each DirectMediaXtra can have up to 64 different cue points marked. Cue points times are measured in milliseconds, and the name of the cue point can have up to 32 characters. To create a cue point you should first locate the time using the preview movie in the Properties Dialog window. Then select "ADD". A window will appear for you to define the name of the cue point. If you need to modify a cue point later you can double-click its name on the list of cue points and edit its name and time directly. You can use the "GOTO" button to check the location of a cue point, and the "REMOVE" button can be used to delete a cue point.

DirectMediaXtra does not modify the video file in any way. The cue points are maintained by your DirectMediaXtra cast members, so you do not need to leave Director or mark them again if the original video is recompressed.

Creating Cue Points from Lingo

There are two functions that you can use to create and remove cue points from Lingo.

```
addcuepoint(member x, cuepointtime, cuepointname)

removecuepoint(member x, cuepointindex)
```

Using Cue Points

DirectMediaXtra cue points can be used just like Director built-in media cue points. You can use the "Wait for cue point" option in the Tempo channel of the score for easy synchronization of media, as shown in the "Testbed.dir" sample movie. For advanced Lingo users, the following properties are supported:

```
isPastCuePoint(sprite x)              the mostRecentCuePoint of sprite x

the cuepointtimes of member x         the cuepointnames of member x
```

Please consult Director's documentation for more information about these properties.

DirectMediaXtra sprites also receive the onCuePassed Lingo message. However, Director's documentation is inaccurate when it comes to the syntax of this particular Lingo message. The correct syntax is:

```
on cuePassed me, whichChannel, cuePointNumber, cuePointName
```

Director's help does not mention the me parameter, but we verified that this syntax also applies to SWA and Quicktime generated cue points. You can use this event to get information about cue points from inside your behaviors. Again, the "Testbed.dir" movie uses this command to display the name of the most recent cue point on the screen. Please examine the movie for more information.

IMPORTANT: The "Wait for cue point" setting in the Tempo channel does not work correctly if the DirectMediaXtra sprite is in the very last frame of the score. This is also true for SWA and Quicktime sprites. It is always a good idea to avoid using the very first and the very last frame of the score in Director, since they do not behave exactly like all the others.

Member Properties

All properties of the DirectMediaXtra cast member can be easily set in the Properties Dialog box, which is displayed when the cast member is created and also when you double-click the cast member icon in the cast window. These properties can also be set via Lingo.

File - the complete pathname of the linked media file.
Example: put the file of member 1
-- "C:\myfiles\myfile.mpg"

Title - the filename of the linked Media file. This property is used to resolve the pathname if the Media file is not found on the original location.
Example: put the title of member 1
-- "file1.mpg"

Duration - the duration of the linked Media file in milliseconds.
Example: put the duration of member 1
-- 45200

Videowidth - the original encoded width of the linked Media file, in pixels. This is a read-only property.
Example: put the videowidth of member 1
-- 320

Videoheight - the original encoded height of the linked Media file, in pixels. This is a read-only property.
Example: put the videoheight of member 1
-- 240

Volume - The default audio volume to be used during playback of new Xtra sprites. Values range from -100 to 0, expressed in dB. The sprite function SetVolume may be used when you want to modify the volume of a sprite that is already on the stage.
Example: put the volume of member 1
-- -10

Balance - The default audio balance to be used during playback of new Xtra sprites. Values range from -100 to 100, expressed in dB. The sprite function SetBalance may be used when you want to modify the balance of audio of a sprite that is already on the stage.
Example: put the balance of member 1
-- 0

Rate - The default playback rate to be used during playback. Values range from 0 to 200, expressed in percentage of the original playback rate. The sprite function SetRate may be used when you want to modify the rate of playback of a sprite that is already on the stage. IMPORTANT: Windos Media files do not support the rate property. Playback will fail if the rate is set to any value but the default of 100.
Example: set the rate of member 1 to 50

Cpuboost - DirectMediaXtra can slow down Director to allow more CPU time for the DirectShow driver. This can be useful if you are experiencing poor or stuttering playback. Legal

values are in the range of 0 to 95 percent, although values over 80 may result in problems with the performance of your Director animation and/or Lingo event detection.
Example: set the cpuboost of member 1 to 50

AdjustDurationBeforePlayback - If this property is TRUE, the Xtra will scan the media file and adjust the duration of member property as soon as the sprite starts playing. This is necessary because different DirectShow filters may report different durations for the same clip. If this happens when you take your finished product to a machine with a different DirectShow filter, the Xtra may not reach certain cuepoints or it may consider that the end of the file was not reached. Depending on your scripting this can prevent your movie from operating correctly, so this setting should be enabled in the majority of cases.
Example: set the adjustdurationbeforeplayback of member 1 to TRUE

ShowLocateFileDialog - the Xtra will try to locate the linked media files at runtime if you move them from the original location. The mechanism is described in the CREATING PROJECTORS section of the documentation. If the linked file can not be found, this property is used to specify if a dialog box is displayed asking the user to locate the file, or if the Xtra fails silently.
Example: set the showlocatefiledialog of member 1 to TRUE

LinkedMediaLocated - This is a property that is set automatically by the Xtra, and indicates if the linked media file for this member was located when the movie loaded (either automatically by the Xtra, or using the Locate File dialog box.) The Xtra will not attempt to initialize playback of a linked file if this property is FALSE.
Example: put the linkedmedialocated of member 1
-- 1

HasToolbar - Indicates whether a toolbar is attached to the video window. Possible values are 0 (FALSE) or 1 (TRUE).
Example: set the hastoolbar of member 1 to TRUE

Loop - Indicates whether the linked Media file will loop automatically. Possible values are 0 (FALSE) or 1 (TRUE).
Example: set the loop of member 1 to TRUE

PlaySegment - Indicates whether only a segment of the Media file is to be played. Possible values are 0 (FALSE) and 1 (TRUE).
Example: set the playsegment of member 1 to TRUE

SegmentStart - Specifies the start time of the Media file to be played, in milliseconds. Only active if "the playsegment of the member" is TRUE.
Example: set the segmentstart of member 1 to 10700

SegmentEnd - Specifies the end time of the Media file to be played, in milliseconds. Only active if "the playsegment of the member" is TRUE.
Example: set the segmentend of member 1 to 43200

PausedAtStart - Indicates whether the Media file should be

opened in "paused" state. Possible values are 0 (FALSE) and 1 (TRUE).
Example: set the pausedatstart of member 1 to TRUE

CuePointTimes - Contains a list of the cue point times associated with this member.
Example: put the cuepointtimes of member 1
-- [4500,12300,34650,67321]

CuePointNames - Contains a list of the cue point names associated with this member.
Example: put the cuepointnames of member 1
-- ["Cue 1","NewCuePoint","Start Point","Second part"]

Sprite Properties

The Sprite properties are only accessible when a DirectMediaXtra sprite is being displayed on the stage. You should use the sprite channel as the argument to pass commands to the sprite.

Movietime - The current position of the Media file, in milliseconds. This property is new in DirectMediaXtra 2, for Lingo compatibility with Quicktime and AVI members. This returns the same value as the currenttime of sprite property.
Example: put the movietime of sprite 1
-- 22345

Movierate - The playback rate Media file. This property is new in DirectMediaXtra 2, for Lingo compatibility with Quicktime and AVI. DirectMediaXtra supports rates between 0 and 2 (reverse playback is not supported.) The movierate can also be used to play or pause the media, setting it to 1 or 0. Windows Media files do not support this property: videoplay() and videopause() should be used for Windows Media files.
Example: set the movierate of sprite 1 to 0.5

Currenttime - The current position of the Media file, in milliseconds.
Example: put the currenttime of sprite 1
-- 22345

Mediabusy - Indicates whether the Media file is playing. Possible values are 0 (FALSE) and 1 (TRUE) .
Example: put the mediabusy of sprite 1
-- 1

MostRecentCuePoint - The index of the last cue point passed. Consult Director's documentation for additional info on this property.
Example: put the mostrecentcuepoint of sprite 1 to TRUE
-- 2

Director Help

Member Functions

DirectMediaXtra provides several functions that can be called from Lingo to retrieve information about the Media linked to the members.

AddCuePoint(member x, cuepointtime, cuepointname) - This function can be used to create a new cue point from Lingo. Cue points are automatically sorted for you.
Example: addcuepoint(member 2, 10500, "NewCue")

RemoveCuePoint(member x, cuepointindex) - This function can be used to remove a cue point. You should supply the index of the cue point to be removed.
Example: removecuepoint(member 2, 2)

ScanFile(member x) - This function can be used to extract the correct duration and dimensions of a video file when the linked media is changed via Lingo. The file is queried and the castmember properties are updated automatically. With DirectMediaXtra 1.4 we introduced the adjustdurationbeforeplayback property, which also adjusts the duration, but does not impose an additional time penalty (the file is not opened twice), so it is generally better to leave it checked instead of using scanfile().
Example: scanfile(member 2)

Register(member x, registrationnumber) -This function can be used in Projectors to register DirectMediaXtra at runtime, and allow developers to save linked casts. It is not necessary for the normal operation of the Xtra.
Example: register(member 2, "SERIALNUMBERHERE")

IsDirectShowInstalled(member x) - Returns 1 (TRUE) or 0 (FALSE). Can be used to determine if DirectShow (formerly ActiveMovie) is installed on the user's machine. You should use a valid DirectMediaXtra cast member index to access this function.
Example:isDirectShowInstalled(member 1)
if the result=0 then
alert("Please install DShow")
else
--nothing
end if

Sprite Functions

The following spritel functions can be called from Lingo to control playback of DirectMediaXtra sprites. The sprite must be present in the score, and the function should be targeted at the sprite channel number.

Videoplay(sprite x) - Used to start or resume playback of a Media file. Playback is usually started automatically when the sprite is loaded, unless the "Paused at Start" option is checked.
Example: videoplay(sprite 1)

Videopause(sprite x) - Used to pause playback of a Media file.
Example: videopause(sprite 1)

Videoseek(sprite x, time) - Seeks the video to a different time in the Media file, and pauses the video. Time should be expressed in milliseconds.
Example: videoseek(sprite 1, 15000)

Videoplaysegment(sprite x, starttime, endtime) -Plays a segment of the Media file. Please note that using this option actually limits access to the times of the stream that falls outside the playback segment boundaries. You should use another videoplaysegment() command to seek to a different time on the stream.
Example: videoplaysegment(sprite 2, 10000, 25000)

IsPastCuePoint(sprite x, cuepointindex) - Returns TRUE if the cue point indicated has already passed, FALSE otherwise. DirectMediaXtra only supports this function used with cuepoint indexes, not names.
Example:isPastCuePoint(sprite 1, 2)
put the result
-- 1

SetVolume(sprite x, volume) - Sets the volume of audio playback during the current sprite span. Valid volumes are in the range of -100 to 0 dB.
Example: setvolume(sprite 1, 0)

GetVolume(sprite x) - Returns the current volume of audio playback, in dB.
Example: put getvolume(sprite 1).
-- -20

SetBalance(sprite x, balance) - Sets the balance of audio playback during the current sprite span. Valid values are in the range of -100 to 100 dB.
Example: setbalance(sprite 1, 0)

GetBalance(sprite x) - Returns the current balance of audio playback, in dB.
Example: put getbalance(sprite 1).
-- -100

SetRate(sprite x, rate) - Sets the rate of playback during the current sprite span. Valid values are in the range of 0 to 200 %.

Example: setrate(sprite 1, 80)

GetRate(sprite x) - Returns the current playback rate, in %.
Example: put getrate(sprite 1)
-- 150

SetFullScreen(sprite x) - This function will trigger the ActiveMovie/DirectShow full-screen video renderer. The DirectMediaXtra sprite will be promoted to full-screen playback and a new video surface will be created. The full-screen renderer works independently from the authoring application, and it will be full-screen even if the stage in Director does not fill the whole monitor. Mouse event detection and keypresses are NOT passed to the authoring application, so you should use this function in conjunction with cuepoints to return control to the authoring application. Remember to call RemoveFullScreen() when you are done with playback. On most applications it is better to simply stretch the DirectMediaXtra sprite to cover the whole stage area: you will still benefit from any video driver acceleration and interactivity control will be retained.
Example: setfullscreen(sprite 1)

RemoveFullScreen(sprite x) - This function will disable the ActiveMovie/DirectShow full-screen video renderer. The DirectMediaXtra sprite will return to its original window on the stage in Director.
Example: removefullscreen(sprite 1)

Creating Projectors

DirectMediaXtra icons operate only under Windows 95, 98, ME, Windows NT 4 and Windows 2000. Please check our web site at http://xtras.tabuleiro.com for updated information about system requirements for different operational systems. You should create your Projectors for Windows 95 and Windows NT, and with the "Animate in Background" option checked. To prevent problems with palettes you should include with each projector a copy of the modified DIRECTOR.INI file you created following the instructions in the Getting Started section. The .ini file should be renamed to match the name of the Projector, i.e.:

Projector: myproj.exe

.INI file: myproj.ini

The Xtra is automatically included in the Projector if your first movie contains a DirectMediaXtra cast member. If you are creating a "dummy" projector that will call your .dxr movie, you can include the Xtra in the Projector using the MODIFY->MOVIE->XTRAS menu, and adding the DirectMediaXtra.x32 file.

TIP: You can also deliver the DirectMediaXtra.x32 file in a folder named XTRAS, located in the same directory of your Projector, if you do not want to embed the file into your projector. This is recommended for faster startup of your Projector, and for delivery on the NT platform.

We also recommend including a LINGO.INI file with your Projector. This file should contain the following code to prevent problems playing sounds when Intel RSX and DirectSound are installed:

```
on startup
     set the soundkeepdevice to 0
end startup
```

The final structure of files should look like:

Projector - C:\myproj\proj.exe

.INI file - C:\myproj\proj.ini

Lingo.ini - C:\myproj\lingo.ini

Optional (if Xtra was not embedded into the Projector):

Xtra file - C:\myproj\Xtras\DirectMediaXtra.x32

The Xtra will automatically locate all linked Media files that are located in the same directory of the Projector. Additional directories can also be specified using the SEARCHPATHS list attached to each DirectMediaXtra castmember. This list can be edited directly from the DirectMediaXtra castmember properties dialog box and is saved with the castmember in the

Director movie. Both absolute and relative directory entries may be used. Relative directories are evaluated in relation to the projector executable, while absolute directories must contain a drive letter and complete pathname information.

Examples:

Relative directory entry: media

Absolute directory entry: E:\media

When used with Director 5 and 6, the Xtra will also search all the directories listed in "the searchpaths" Lingo variable for missing Media files. The searchpaths is also evaluated before any Lingo in the current movie executes so it should be set preferably in a LINGO.INI file. The release version of Director 7 had a bug that was corrected in version 7.02, and "the searchpath" and "the searchpaths" variables were not being scanned by Director. We included special code on DirectMediaXtra to work around the searchpath bug, using a global variable named tbsearchpath. You can include the following code on your LINGO.INI file to define and add directories to the searchpath and tbsearchpath variables, specifying relative directories to hold your linked MPEG files:

DIRECTOR 5 and 6 LINGO.INI:

```
on startup
        set the soundkeepdevice to 0
        -- Added to look for files in the "media" subdirectory
        append(the searchpath, the pathname & "media")
end startup
```

DIRECTOR 7 and 8 LINGO.INI:

```
on startup
        set the soundkeepdevice to 0
        -- Added to look for files in the "media" subdirectory
        global tbsearchpath
        set tbsearchpath=[]
        append(tbsearchpath, the applicationPath & "media\")
        append(the searchpaths, the applicationPath & "media")
end startup
```

You can safely put your Media files in the "media" subdirectory (C:\myproj\media in the example above) and the Xtra will automatically find them when the Projector starts.

How to Order & Register

The unregistered version of DirectMediaXtra is fully-functional and may be used for evaluation, nonprofit and educational purposes only: commercial distribution is strictly prohibited. A registered version of DirectMediaXtra can be used in commercial products, and may be purchased online at xtras.tabuleiro.com, using a secure server. At our web site you can also consult our purchase policy, purchase instructions, payment, delivery and security methods.

If you decide to buy the Xtra you don't need to download a new copy of the software. After your order is processed you will receive an e-mail with a serial number to register the software you've already installed on your machine.

To register you should double-click a DirectMediaXtra castmember / icon and select "Register" in the Properties dialog box. Please keep your serial number archived. You will need it to register again if you reinstall Director, Authorware or the DirectMediaXtra.

Licensing & Availability

DirectMediaXtra is a commercial product. Current price and updated information can be found at xtras.tabuleiro.com. The Xtra can be distributed with your Projectors. You have to include the following copyright information in the product documentation:

DirectMediaXtra(tm) (c) Tabuleiro Prod. Ltda EPP 1997-2001

All Rights Reserved

Xtra is a trademark of Macromedia, Inc.

No royalty-fees are required for a distribution of the playback version of the Xtra with your product.

Technical Support

Please use the Support Account available at our web site (xtras.tabuleiro.com) to submit your questions. The site also contains Technotes and other resources that can help you identify and solve the most common problems quickly.