

gives a quick summary of the property's purpose, but the full descriptions that follow later provide important details. All properties of the current timeline can be accessed without an explicit reference to it, as in `_alpha` versus `someClip._alpha`.

Table R-11. Movie clip property summary

Property name	Access	Type	Property description
<code>_alpha</code>	R/W	number	Opacity percentage: 0 is transparent; 100 is opaque
<code>_currentframe</code>	RO	number	Frame number at which playhead resides
<code>_droptarget</code>	RO	string	Target path of the clip over which a dragged clip hovers or has been dropped, in slash notation
<code>enabled</code>	R/W	Boolean	Allows or disallows button interaction
<code>focusEnabled</code>	R/W	Boolean	Allows the clip to receive keyboard focus programmatically
<code>_focusrect</code>	R/W	Boolean	Disables automatic yellow rectangle around focused movie clips
<code>_framesloaded</code>	RO	number	Number of frames that have been downloaded
<code>_height</code>	R/W	number	Height, in pixels, of the current contents
<code>hitArea*</code> †	R/W	string	Assigns the clickable area for a button-enabled movie clip
<code>_name*</code>	R/W	string	Identifier of an instance as a string (not a reference)
<code>_parent*</code>	RO	MovieClip reference	A reference to the instance or movie that contains the current instance
<code>_rotation</code>	R/W	number	Degrees of rotation
<code>tabChildren*</code> †	R/W	Boolean	Controls whether contained objects are included in the automatic tab order
<code>tabEnabled*</code> †	R/W	Boolean	Includes or excludes the movie clip from the current tab order
<code>tabIndex*</code> †	R/W	Boolean	Specifies the movie clip's index in the custom tab order
<code>_target</code>	RO	string	Target path in absolute terms, in slash notation
<code>_totalframes</code>	RO	number	Number of frames in the timeline
<code>trackAsMenu*</code> †	R/W		Modifies the <code>onRelease()</code> handler requirements, enabling menu-style behavior
<code>_url</code>	RO	string	Disk or network location of the source <code>.swf</code> file
<code>useHandCursor*</code> †	R/W	Boolean	Dictates whether a hand cursor is displayed when the mouse is over the clip (applies to clips with button handlers only)
<code>_visible</code>	R/W	Boolean	Visibility: <code>true</code> if shown; <code>false</code> if hidden
<code>_width</code>	R/W	number	Width, in pixels, of the current contents
<code>_x</code>	R/W	number	Horizontal position, in pixels
<code>_xmouse</code>	RO	number	Horizontal location of mouse pointer, in pixels
<code>_xscale</code>	R/W	number	Horizontal scaling percentage
<code>_y</code>	R/W	number	Vertical position, in pixels
<code>_ymouse</code>	RO	number	Vertical location of mouse pointer, in pixels
<code>_yscale</code>	R/W	number	Vertical scaling percentage

\* Applies to movie clip instances only; does not apply to the main timeline (`_root`).

† New in Flash 6. Not supported in prior versions.

Table R-12. Movie clip method summary

Method name	Method description
<code>attachMovie()</code>	Creates a new instance based on an exported symbol from the current document's Library
<code>createEmptyMovieClip()*†</code>	Creates a new, empty <i>MovieClip</i> instance
<code>createTextField()*†</code>	Creates a new <i>TextField</i> object
<code>duplicateMovieClip()*†</code>	Creates a copy of a movie clip instance
<code>getBounds()</code>	Returns an object whose properties give the coordinates of the bounding box that defines the visual region occupied by the clip
<code>getBytesLoaded()</code>	Returns the number of downloaded bytes of an instance or a movie (not applicable to internal clips)
<code>getBytesTotal()</code>	Returns the physical byte size of an instance or a main movie
<code>getDepth()*†</code>	Returns the movie clip's position in the visual content stack
<code>getURL()*†</code>	Loads an external document (usually a web page) into the browser
<code>globalToLocal()</code>	Converts the properties of a <i>coordinates</i> object from Stage coordinates to instance coordinates
<code>gotoAndPlay()*†</code>	Moves the playhead of an instance or movie to a specific frame and then plays the instance or movie
<code>gotoAndStop()*†</code>	Moves the playhead of an instance or movie to a specific frame and then stops the playhead
<code>hitTest()</code>	Returns a Boolean indicating whether a clip intersects with a given point or another clip
<code>loadMovie()*†</code>	Loads an external <i>.swf</i> file into the Player
<code>loadVariables()*†</code>	Retrieves external data composed of variable names and values, and converts that data into equivalent ActionScript variables
<code>localToGlobal()</code>	Converts the properties of a <i>coordinates</i> object from an instance's coordinates to Stage coordinates
<code>nextFrame()*†</code>	Moves the playhead of an instance or movie ahead one frame and stops it there
<code>play()*†</code>	Starts the playhead of an instance or movie in motion (i.e., plays the clip)
<code>prevFrame()*†</code>	Moves the playhead of an instance or movie back one frame and stops it there
<code>removeMovieClip()*†</code>	Deletes a duplicated or attached instance
<code>setMask()*†</code>	Assigns a movie clip as a mask for another clip
<code>startDrag()*†</code>	Causes an instance or movie to follow the mouse pointer physically

## MovieClip Class

Table R-14. Movie clip event handler summary (continued)

Clip event handler	Clip event occurs when...
<code>onMouseMove()</code>	Primary mouse button is depressed and then released while the clip is on stage
<code>onMouseUp()</code>	Mouse pointer moves (even a teensy bit) while the clip is on stage
<code>onPress()*</code>	The mouse is pressed over the clip
<code>onRelease()</code>	The mouse is pressed and then released over the clip
<code>onReleaseOutside()*</code>	The mouse is pressed over the clip and then released after moving off the clip
<code>onRollOut()*</code>	The mouse pointer moves off the clip while the mouse button is not depressed
<code>onRollOver()*</code>	The mouse pointer moves over the clip while the mouse button is not depressed
<code>onSetFocus()*</code>	The clip gains keyboard focus
<code>onUnload()*</code>	The clip is removed from the Stage

\* Applies to movie clip instances only; does not apply to the main timeline (`_root`).