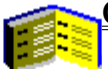
Contents

For additional assistance, contact Technical Support.

Basics of path animation

Step-by-step procedures

OpenScript reference

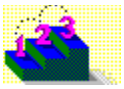


Glossary

Basics of path animation

The topics below provide basic information about path animation

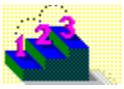
- About path animation
- How path animation works
- The Animation window



Step-by-step procedures

The topics below provide step-by-step instructions for working with path animation

- **Creating and modifying animations**
- **Playing an animation**

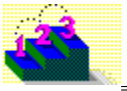


Step-by-step procedures

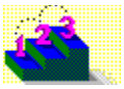
The topics below provide step-by-step instructions for working with path animation

Creating and modifying animations

- Changing an object's size during an animation
- Creating a cel animation
- Creating a path animation
- Creating and modifying curves in the animation path
- Creating multiple animation paths for one object
- Distributing a book with animation paths
- Extending or shortening an animation path
- Getting and setting animation properties
- Moving an animation path
- Selecting an animation to edit
- Setting animation options

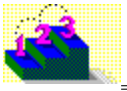


Playing an animation



Step-by-step procedures

The topics below provide step-by-step instructions for working with path animation



Creating and modifying animations

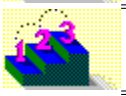
Playing an animation

Controlling animation with OpenScript messages
Requesting notification by animations
Starting an animation

OpenScript reference

The topics below provide information about specific OpenScript properties and messages

Properties



Messages

OpenScript reference

The topics below provide information about specific OpenScript properties and messages

Properties

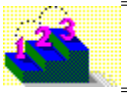
anim_CelAnimation
anim_CurrentStep
anim_CurrentTime
anim_Duration
anim_ElapsedTime
anim_EndSize
anim_NumRepetitions
anim_NumSteps
anim_Offset
anim_ShowAllSteps
anim_StartPosition
anim_StartSize
anim_StartTime
anim_StepsPerCel

Messages

OpenScript reference

The topics below provide information about specific OpenScript properties and messages

Properties



Messages

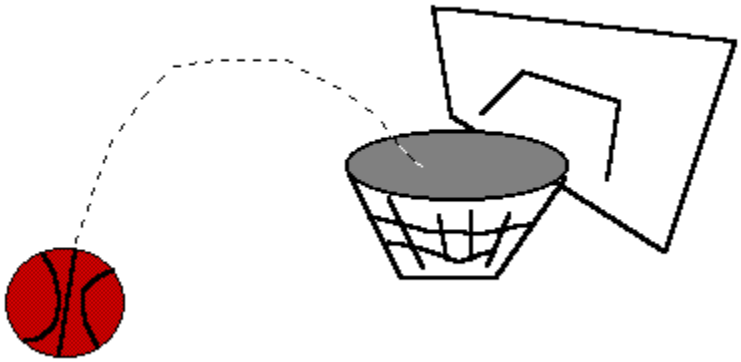
doneAnimatingNotify
jumpToPercent
jumpToStep
playAnimation
playStep
restoreAnimation
stopAnimation

About the Path Animation utility

Step-by-step

See also...

The Multimedia ToolBook Path Animation utility lets you animate objects without programming. You can draw a path using the mouse, and when you play the animation, the object moves along the path at a speed you determine. This makes it simple to create animations such as a ball going through a hoop.



You can also create cel animations that show and hide different views of an object in rapid succession, which is useful for animations such as the earth rotating.

In addition to creating animations with the mouse, you can use a dialog box to set the

- speed and duration of animation playback, including acceleration and deceleration.
- number of times to repeat the animation.

You can define multiple animation paths for each object, then play them back selectively. Information about an object's animation is stored with the object, so it is retained when you cut or copy the object to other pages or books.

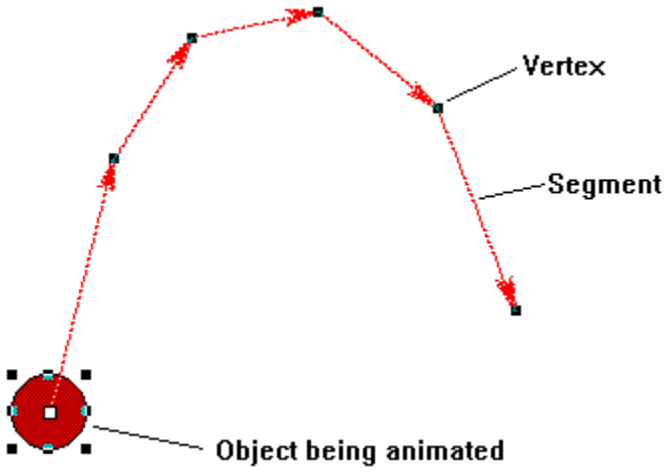
Step-by-step

Creating a cel animation
Creating a path animation
Distributing a book with animation paths
How path animation works
The Animation window

How path animation works

Step-by-step

To create a path-based animation, you draw the path in the Animation window. The path consists of segments defined by vertices.



After you draw the path, you can adjust it by moving segments, converting them into curves, adding new vertices, or removing existing vertices. When you are finished drawing the path, Multimedia ToolBook compiles information about the path and stores it as a user property of the object.

When you play the animation, the object moves from vertex to vertex. By default, the object moves at a constant rate over the path. If you set its rate to variable, however, the object spends the same amount of time between vertices, moving slower between closely-spaced vertices and faster between widely-spaced ones. This allows you to create animations that appear to accelerate and decelerate.

Animated objects do not move along a completely smooth path. Instead, they jump from point to point on the path; each point is called a step. The more steps in the path, the smoother the animation.

Step-by-step

About the Path Animation utility
Creating a cel animation
Creating a path animation
Creating multiple animation paths for one object
Distributing a book with animation paths
Setting animation options
Starting an animation
The Animation window

Creating a path animation

Step-by-step

To create a path animation, you display the Animation window and draw the path. When you are finished, Multimedia ToolBook compiles the resulting path into an animation that you can play by sending a message to the object.

To create a path animation:

1. In the Multimedia ToolBook Main window, select the object you want to animate. If the object is on the background, switch to the background first, then select the object.
2. Choose Path Animation from the Tools menu. The Path Animation utility displays the objects on the current page in the Animation window, and draws the first segment of the animation path for the selected object.
3. (Optional) To change the location of the first segment, move the cursor over the selected object until appears, then drag the segment to a new position.
4. To add a segment, click the vertex tool ( ) or press the spacebar, then click at the point where the new segment should end.
5. Move the vertex cursor and click to add segments until you have completed the path.
6. Click Done to save the animation and return to the Main window.

After you have created the animation path, you can set animation options to control the duration and smoothness of the animation.

Note To change the size of an object after you have created an animation path for it, edit the object in the Animation window. Otherwise Multimedia ToolBook displays the object during the animation at the size it was at when the animation was last compiled. For details, see Changing an object's size during an animation.

Step-by-step

About the Path Animation utility
Changing an object's size during an animation
Creating a cel animation
Creating and modifying curves in the animation path
Creating multiple animation paths for one object
Distributing a book with animation paths
Extending or shortening an animation path
Getting and setting animation properties
Moving an animation path
Setting animation options
Starting an animation
The Animation window

Setting animation options

Step-by-step

You can adjust the animation's duration or rate in the Animation Settings dialog box.

To display the Animation Settings dialog box:

1. Display the Animation window and select the animation to edit.
2. Click to display the Animation Settings dialog box.
3. Make the changes you want, then click OK.
4. Click Done in the Animation window to save your changes and recompile the animation.

Step-by-step

About the Path Animation utility
Creating a cel animation
Creating a path animation
Creating multiple animation paths for one object
Getting and setting animation properties
Selecting an animation to edit
Starting an animation
The Animation window

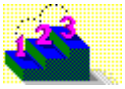
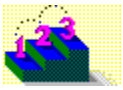
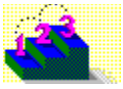
OpenScript reference

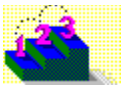
`anim_CelAnimation`
`anim_Duration`
`anim_NumRepetitions`
`anim_NumSteps`
`anim_ShowAllSteps`
`anim_StepsPerCel`

Animation Settings dialog box

Step-by-step

Adjusts the way an animation plays, including its duration, rate, and the number of times the animated object travels along its path.

Option	Description
Duration in Seconds	Specifies the time it will take the object to travel along its animation path from beginning to end. The default value is 5 seconds. The duration can contain a decimal value to allow synchronization with events that do not last an equal number of seconds. This option sets the animation's `anim_Duration` property.
Steps per Second	Determines the number of increments by which the animation moves in a second. The higher this number, the smoother the animation will appear. The maximum number of frames this control will allow is 30. The default is 15. If Multimedia ToolBook cannot show as many steps per second as you specified, the animation will skip steps to finish in the time that you specified under Duration. This guarantees that the animation lasts the specified amount of time.
Show All Steps	Check this to force Multimedia ToolBook to show all the specified steps per second, even if the animation lasts longer than the specified duration as a result. This feature is helpful if you have drawn an animation in which skipping a step would make the animation look wrong. This option sets the animation's `anim_ShowAllSteps` property.
Rate	Specifies how steps are distributed along the path. The default value of "Constant" causes Multimedia ToolBook to move the animation along the path at a fixed rate. A value of "Variable" causes Multimedia ToolBook to spend the same amount of time on each segment of the path; as a result, the object appears to move more slowly along shorter segments. Use a variable rate with short and long segments if you want to mimic motion that accelerates or decelerates, such as a ball shot out of a cannon.
Repetitions	Sets number of times that the object travels along its path every time the animation is played. The default is 1 time. If you click Forever, you can stop the animation by  pressing the Esc key.  sending the `stopAnimation` message to the object.  navigating to a page on which the object does not appear or activating a different window (such as the Command window or another viewer). This option sets the animation's `anim_NumRepetitions` property.
Update Path on Move	Specifies whether Multimedia ToolBook repositions the animation path if you move the object at Author level in the Main window. If this option is unchecked, the animation starts at a fixed point and Multimedia ToolBook moves the object to the animation's starting point when the animation begins.
Cel Animation On	Specifies that Multimedia ToolBook should treat the selected group as a cel animation, and hide and display objects in the group in layer order as the group moves along the animation path. This option is not available if the object being animated is not a group. This option sets the animation's `anim_CelAnimation` property.
Steps per Cel	In a cel animation, specifies the number of steps that Multimedia ToolBook moves before hiding one cel and displaying the next. This option sets the animation's `anim_StepsPerCel` property.


1
2
3

Selecting an animation to edit

Step-by-step

Before modifying an animation, you must select it. If the object for which you want to edit the path is difficult to select in the Main window, or if you want to edit an animation path other than the first one, follow these steps:

To select an animation:

1. In the Multimedia ToolBook Main window, choose Path Animation from the Tools menu to display the Animation window.
2. Click .
3. In the Object To Animate combo box choose the object or group to animate.
4. In the Animation combo box choose the number of the animation to edit, then click OK.

Tip If the object has only one animation, it is easier to select the object in the Multimedia ToolBook Main window, then choose Path Animation from the Tools menu. Multimedia ToolBook displays the Animation window with the object and its path already selected.

Step-by-step

About the Path Animation utility
Changing an object's size during an animation
Creating a cel animation
Creating a path animation
Creating and modifying curves in the animation path
Creating multiple animation paths for one object
Extending or shortening an animation path
Getting and setting animation properties
Moving an animation path
Setting animation options
The Animation window

Moving an animation path

Step-by-step

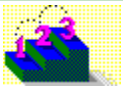
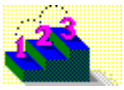
You can move the entire path at once, or you can move a single segment to reshape the animation path.

To move an entire path:

1. Select the animation to edit.
2. Click
3. Move the cursor over any section of the path until appears.
4. Drag the path to a new location.

Note If Update Path On Move in the Animation Settings dialog box is checked, Multimedia ToolBook moves the path automatically if you move the object in the Main window. For details, see Setting animation options.

To move one segment of the path:

1. Select the animation to edit.
2. Click 
3. Move the cursor over one of the vertices that defines the path segment until  appears.
4. Drag the vertex to a new location.

You can also move the animation path by setting the object's `anim_StartPosition` property using OpenScript commands.

Step-by-step

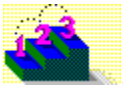
Changing an object's size during an animation
Creating a cel animation
Creating a path animation
Creating and modifying curves in the animation path
Creating multiple animation paths for one object
Extending or shortening an animation path
Getting and setting animation properties
Selecting an animation to edit
Setting animation options
The Animation window

Extending or shortening an animation path

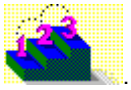
Step-by-step

You can extend a path by adding a segment to the end or in the middle of a path. To shorten a path, delete a segment.

To add a segment to the end of the path:

1. Select the animation to edit.
2. Click  . The vertex cursor appears ( ).
3. Move the vertex cursor to where you want the new segment to end, then click.
   Multimedia ToolBook draws a new segment from the end of the path to the vertex cursor.

To add a segment in the middle of the path:

1. Select the animation to edit.
2. Click  .
3. Move the vertex cursor over the path until appears, then click.
   Multimedia ToolBook adds a new vertex at that point. You can drag the new vertex to change the shape of the path.

To delete one segment of a path:

1. Select the animation to edit.
2. Click the vertex that defines the end of the path segment you want to delete.
3. Press Del.
   Multimedia ToolBook removes that vertex and defines a new path segment, attaching the vertices on either side of the deleted vertex.

Step-by-step

Changing an object's size during an animation
Creating a cel animation
Creating a path animation
Creating and modifying curves in the animation path
Creating multiple animation paths for one object
Getting and setting animation properties
Moving an animation path
Selecting an animation to edit
Setting animation options
The Animation window

Creating and modifying curves in the animation path

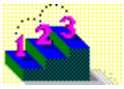
Step-by-step

To create a curve, you modify individual segments of the animation path.

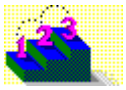
Note You cannot convert the path defined by the first or last vertex into a curve, nor can you create curves on two successive path segments.

To create a curve:

1. Select the animation to edit.
2. In the Animation window, double-click the vertex to be converted.

   The cursor changes to a move cursor  and the path defined by that vertex becomes curved.
3. Drag the vertex to shape the curve.

To reshape a curve:

1. Select the animation to edit.
2. In the Animation window, move the cursor over the vertex that defines the curve until  appears.
3. Drag the vertex to reshape the curve.

To delete a curve:

Double-click the vertex at the top of the curve. Multimedia ToolBook converts the curve back to a straight segment.

Step-by-step

Changing an object's size during an animation
Creating a cel animation
Creating a path animation
Creating and modifying curves in the animation path
Creating multiple animation paths for one object
Extending or shortening an animation path
Getting and setting animation properties
Moving an animation path
Selecting an animation to edit
The Animation window

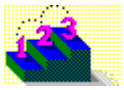
Creating a cel animation

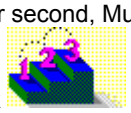
Step-by-step

Cel animation displays a series of images to give the illusion of motion. For example, you can use cel animation to create a spinning globe. The animation consists of different views of the earth, each called a cel. Showing one cel at a time in succession gives the illusion that the globe is turning.

Each cel in the animation is a separate object, which can include ToolBook objects, draw objects, and imported bitmaps or picture objects. After creating all the individual cels, you arrange them in layer order, and then group them. When you play a cel animation, Multimedia ToolBook hides the current cel and shows the next one from first to last over and over as the animation moves along the path.

To create a cel animation:

1. In the Multimedia ToolBook Main window, create individual objects to serve as cels in the animation.
2. Arrange the layer order of the objects so the first cel is on the furthest (lowest) layer and the last cel is on the closest (highest) layer.
3. Position the objects relative to one another. For example, in a spinning globe animation, the objects are placed on top of one another so only the top one is visible.
4. Group the objects.
5. Select the group, the choose Path Animation from the Tools menu.
6. Create a path as you would for other objects. For details, see Creating a path animation.
7. When you are done creating the path, click  to display the Animation Settings window. Under Cel Animation, click On, and then click OK.
8. Click Done in the Animation window to save the animation.

By default, Multimedia ToolBook shows the next cel of the animation at each step of the path. For example, if your animation is set to 15 steps per second, Multimedia ToolBook will show 15 cels every second. To slow the rate at which the cels are shown, click  to display the Animation Settings dialog box, and then enter a higher value under Steps Per Cel.

Step-by-step

About the Path Animation utility
Changing an object's size during an animation
Creating a path animation
Creating multiple animation paths for one object
Distributing a book with animation paths
Getting and setting animation properties
How path animation works
Selecting an animation to edit
Setting animation options
Starting an animation
The Animation window

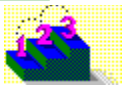
Creating multiple animation paths for one object

Step-by-step

You can create multiple animation paths for one object. For example, if you are creating an animation for a basketball, you could create two paths, one that goes through the hoop, and another that bounces off the rim. You could then play one or the other animation depending on other variables in your application.

To create multiple animation paths:

1. In the Multimedia ToolBook Main window, select the object you want to animate.
2. Choose Path Animation from the Tools menu. The Path Animation utility displays the objects on the current page in the Animation window.

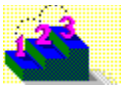
   If the object has no animation paths defined, Multimedia ToolBook draws one segment of the first animation path. If the object already has an animation path, Multimedia ToolBook displays the object and the path in the Animation window.
3. If the object does not already have an animation path, draw the first one. For details, see Creating a path animation.
4. Click  and under Animation select `<new>`.

   Multimedia ToolBook saves the animation you just finished and starts a new one.
5. Draw the next path and repeat Step 4 for each additional animation path you want to draw.
6. Click Done.

To play a specific animation, you specify the number of the animation when you send the `playAnimation` message.

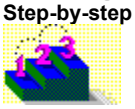
Step-by-step

About the Path Animation utility
Changing an object's size during an animation
Creating a cel animation
Creating a path animation
Creating multiple animation paths for one object
Getting and setting animation properties
How path animation works
Selecting an animation to edit
Setting animation options
The Animation window



Changing an object's size during an animation

Step-by-step



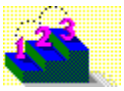
You can make an object change size as it moves along its animation path, so it appears to grow or shrink. To do so, you define the starting and ending sizes at each end of the animation path. When the animation plays, Multimedia ToolBook changes the object's size incrementally as it moves along the path until it reaches the end point.

Note If you want to resize an object in the Main window for which you have defined an animation, you must also edit the animation path and resize the object there. Otherwise when you play the animation the object will revert to the size it was when the animation was last edited.

To change an object's size during an animation:

1. Create an animation path for the object. For details, see Creating a path animation.
2. Click the first vertex in the animation to move the object to the beginning of the path. Or press Home.
3. Use the object's resize handles to size the object to its beginning size.
4. Click the last vertex in the animation to move the object to the end of the path. Or press End.
5. Resize the object to its ending size.
6. Click Done to save the animation.

Tip If you want the object to grow, then shrink, create two animations. Make the object grow during the first one and shrink during the second one. Add a notification request to the first animation that starts the second animation automatically. For details, see Requesting notification by animations.



Step-by-step

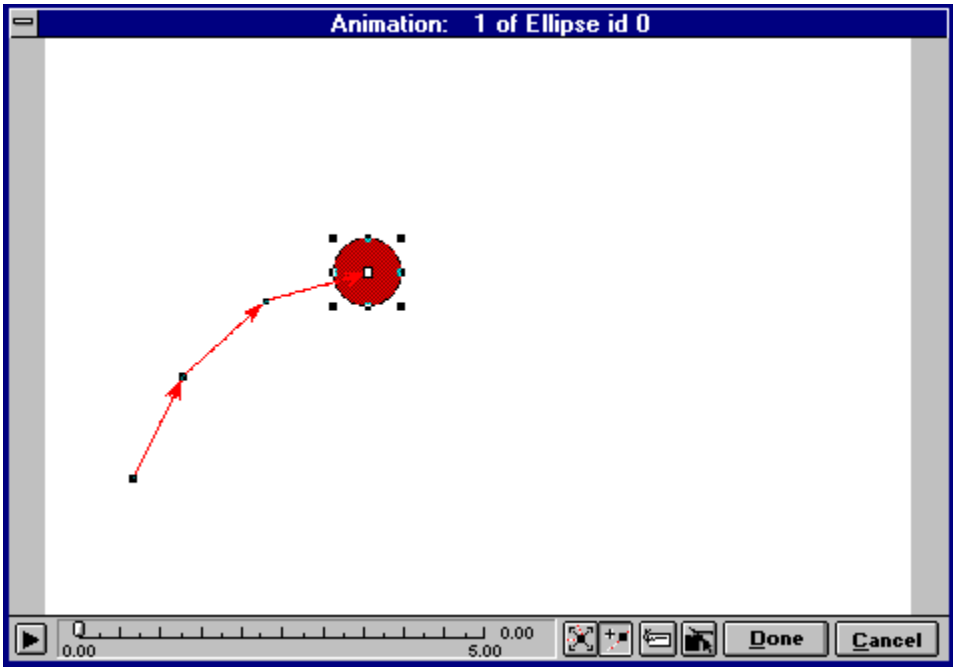
Creating a cel animation
Creating a path animation
Extending or shortening an animation path
Getting and setting animation properties
How path animation works
Requesting notification by animations
Selecting an animation to edit
Setting animation options
The Animation window

The Animation window

Step-by-step

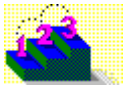
You draw animation paths in the Animation window, which appears when you choose Path Animation from the Tools menu. Multimedia ToolBook copies all the objects from the current page or background into the Animation window, and allows you to edit animation paths for objects or groups.

Note If you display the Animation window while on a page, you can edit the animations for objects on that page. To edit animations for objects on the background, switch to the background before displaying the Animation window.

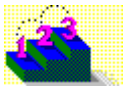


Click to run the current animation.

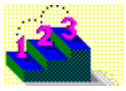
Drag the slider bar to move to a specific point on the animation path.



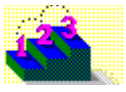
Click to turn the cursor into a move cursor and move segments, vertices, or the entire path.



Click to turn the cursor into a vertex cursor and add new segments to the animation.



Click to display the Animation Settings dialog box and set animation options.



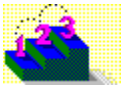
Click to display the Select Object dialog box and select an animation to edit.

Done

Click to save the current animation and return to the Multimedia ToolBook Main window.

Cancel

Click to cancel changes to the current animation and return to the Multimedia ToolBook Main window.



Step-by-step

About the Path Animation utility
Creating a cel animation
Creating a path animation
Distributing a book with animation paths
Getting and setting animation properties
How path animation works
Selecting an animation to edit
Setting animation options
Starting an animation

Starting an animation

Step-by-step

To run an animation, use OpenScript commands in a script or in the Command window to send the `playAnimation` message to the object whose animation you want to play. For example, this button handler runs the animation for an ellipse when the user clicks the button:

```
to handle buttonClick
  send playAnimation 1 to ellipse "basketball"
end
```

If the object has more than one animation, specify the animation number to play:

```
to handle buttonClick
  system svMakePoint  --System variable set elsewhere
  if svMakePoint = true
    send playAnimation 1 to ellipse "basketball"
  else
    send playAnimation 2 to ellipse "basketball"
  end if
end buttonClick
```

If you want the animation to suspend all other activity while it plays, send `true` as the third parameter with the `playAnimation` message:

```
to handle buttonClick
  send playAnimation 1, null, true to ellipse "basketball"
end
```

You can have Multimedia ToolBook send messages while it is underway and when the animation is complete, so you can perform dependent actions or reset the animation. For details, see Requesting notification by animations.

You can also perform processing while the animation is running. For details, see Controlling animation with OpenScript messages.

Step-by-step

About the Path Animation utility
Controlling animation with OpenScript messages
Creating a cel animation
Creating a path animation
Distributing a book with animation paths
Getting and setting animation properties
How path animation works
Requesting notification by animations
Setting animation options

Controlling animation with OpenScript messages

Step-by-step

Example...

Animations are controlled by messages. You start an animation by sending a message to the object whose animation you want to play. Then Multimedia ToolBook sends messages to move the object along the animation path and to stop the animation when it finishes. The following messages are used to control an animation:

Message	Sent by	Purpose
playAnimation	You	Begins animation
playStep	Multimedia ToolBook	Moves object one step along animation path
stopAnimation	Multimedia ToolBook or you	Stops animation

Because Multimedia ToolBook sends the `playStep` message to the object for each step of the animation, you can write a handler for that message to modify the animation as it is running. (Always forward the `playStep` message if you want the object to continue moving along the path.)

To determine when an animation is finished or if it is interrupted, you can request that Multimedia ToolBook send the doneAnimatingNotify message notification. You can write a handler for this message to take appropriate action, such as moving the object back to the beginning of the animation path.

If the animation is not running, you can send messages to move the object to the start of the animation path. You can also move the object along the path manually, which is useful if you want to preview the animation. Send one of the following messages:

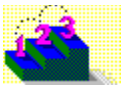
Message	Purpose
restoreAnimation	Moves object to starting point of animation
jumpToPercent	Jumps to a point at the specified percentage of the path
jumpToStep	Jumps to the specified step in the path

Step-by-step

Creating a cel animation
Creating a path animation
Distributing a book with animation paths
Getting and setting animation properties
Requesting notification by animations
Setting animation options
Starting an animation

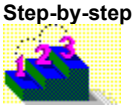
Copy...
Print
Tile windows

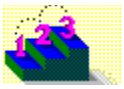
```
--Handler that illustrates how to modify an animation while it is
--running. Changes the color of the animated object as it
--moves along its animation path
to handle playStep
   system colorNumber
   if colorNumber = null then
      colorNumber = 1
   end
   colors = "blue,green,red,yellow,black,white"
   increment colorNumber
   if colorNumber > itemCount(colors) then
      colorNumber = 1
   end
   fillColor of target = evaluate(item colorNumber of colors)
   forward
end


--Handler that illustrates how to move an object along its animation
--path when the animation is not running. Moves an animated object
--along its animation path by 5% each time the button is clicked
to handle buttonClick
   system percent
   if percent is null then
      percent = 0
   end
   increment percent by .05
   if percent > 100 then
      percent = 0
   end
   send jumpToPercent percent to ellipse "basketball"
end
```



Requesting notification by animations

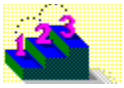
Step-by-step





When you play an animation with the `playAnimation` message, you can request that Multimedia ToolBook send a message when the animation is finished or stops for any reason. For example, you can start an animation and request that a message be sent when it is done so that you can reset the animation to its starting point.

To request notification, include the name of an object to notify when you send the `playAnimation` message. Multimedia ToolBook plays the animation and sends a `doneAnimatingNotify` message to the specified object when the animation is done or interrupted. Write a handler for the `doneAnimatingNotify` message to take appropriate action.



Step-by-step

Controlling animation with OpenScript messages
Creating a cel animation
Creating a path animation
Starting an animation

Copy...
Print
Tile windows

```
--Plays animation and requests that notification be sent
--to the current object when the animation finishes
to handle buttonClick
   send playAnimation 1, self to ellipse "basketball"
end


--Handles the notification message by resetting the animation
--when it is finished
to handle doneAnimatingNotify status, animObject
   send restoreAnimation to animObject
end
```

Distributing a book with animation paths

Step-by-step

When you distribute an application that contains path animations, make sure that you include the file MTB30ANM.SBK as part of the application. This file contains the handlers necessary to run (but not create or change) animations you drew in the Animation window.

Note If you use MTB40.SBK as a system book in your application (the system book that contains handlers for creating animations, indexes, and multimedia widgets), you do not need to also include MTB30ANM.SBK. However, because MTB40.SBK contains authoring tools, your applications probably do not require it; include MTB30ANM.SBK instead.

So that your application can find the animation handlers, make MTB30ANM.SBK a system book in your application. You can do this in one of two ways:

Make the file a startup system book by assigning it to `startupSysBooks` in the [TOOLBOOK] section of the MTB40.INI file. For example:

```
startupSysBooks=MTB30ANM.SBK
```

You can establish settings for the MTB40.INI file if you use the Asymetrix Setup utility. For details, refer to the online Help for that product.

Write an `enterApplication` handler that loads MTB30ANM.SBK into `sysBooks` when users open the book containing the path animation. For example:

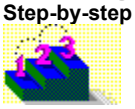
```
to handle enterApplication
  if "MTB40.SBK" is not in sysBooks then
    if "MTB30ANM.SBK" is not in sysBooks then
      push "MTB30ANM.SBK" onto sysBooks
    end if
  end if
  -- ... further statements here
  forward     --Always forward enterApplication handlers
end enterApplication
```

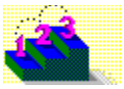
Step-by-step

About the Path Animation utility
Creating a cel animation
Creating a path animation
Creating multiple animation paths for one object
Getting and setting animation properties
Selecting an animation to edit
Starting an animation
The Animation window

Getting and setting animation properties

Step-by-step





Each animation has properties that you can use to determine the animation's settings using OpenScript commands. Some are runtime properties, which means you can set them to affect the animation currently running, or get them to determine the status of the current animation. If you change a runtime property, it does not affect how the animation plays the next time.

Others properties are permanent, meaning you can get their values at any time, even if the animation is not running. If you set a permanent property while an animation is running, you affect the way the current animation runs and the way it runs the next time it is played.

Some properties are get-only, which means you can get their values, but cannot set them.

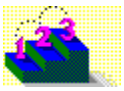
Tip Runtime properties return a value of -1 when the animation is not running, which is useful to determine whether the animation is running.

The complete list of properties is listed in the following table.

Property	Description	Type
`anim_CelAnimation`	If `true`, Multimedia ToolBook hides and displays individual objects in an animated group.	Permanent
`anim_CurrentStep`	The current step number in a running animation.	Runtime
`anim_CurrentTime`	Elapsed time since Windows was started.	Runtime, get-only
`anim_Duration`	Total duration in seconds of the animation (can be overridden by `anim_NumSteps` if `anim_ShowAllSteps` is `true`).	Permanent
`anim_ElapsedTime`	Elapsed time since the animation started.	Runtime, get-only
`anim_EndSize`	Size in page units of object when animation ends.	Permanent, get-only
`anim_NumRepetitions`	Number of times to repeat the animation.	Permanent
`anim_NumSteps`	Total number of steps in the animation.	Permanent
`anim_Offset`	Amount in page units to move the entire animation path from the point specified in `anim_StartPosition`.	Runtime, get-only
`anim_ShowAllSteps`	If `true`, Multimedia ToolBook shows all steps in an animation even if the animation takes longer than specified in the `anim_Duration` property	Permanent
`anim_StartPosition`	Coordinates in page units specifying where the animation begins.	Permanent
`anim_StartSize`	Size in page units of object when animation begins.	Permanent, get-only
`anim_StartTime`	Time at which animation started.	Runtime, get-only
`anim_StepsPerCel`	Number of steps to complete before showing next cel in a cel animation.	Permanent

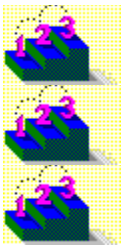
Other information about the animation is available in an array stored as a user property of the object. For details

on how to access this array, see anim_AnimationSettings.



Step-by-step

Controlling animation with OpenScript messages
Setting animation options
Starting an animation



```
--Starts animation after determining total number of
--steps in animation
to handle buttonClick
   system numSteps     --Variable to share with playStep handler
   numSteps = anim_NumSteps of self
   send playAnimation 1 to self
end
--Monitors animation, using animation properties to
--calculate percentage complete
to handle playStep
   system numSteps
   curStep = anim_CurrentStep of target
   if now <> null then
      percent = ceiling((curStep/numSteps)*100)
      caption of statusBar = percent & "%"
   end if
   forward      --Always forward playStep message
end playStep
```

anim_CelAnimation

Permanent property

Syntax

```
get anim_CelAnimation(<animNumber>) of <objectRef>
anim_CelAnimation(<animNumber>) of <objectRef> = <value>
```

Parameters

<animNumber>
The animation number to get or set. If you do not indicate an animation number, Multimedia ToolBook uses the first animation.

<objectRef>
The object whose animation to get or set.

Description

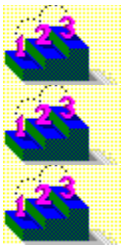
A permanent property that determines whether the animation cycles through the cels as the object moves along the animation path.

Value

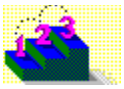
`True` if cel animation is turned on. If `false`, the object moves along its path, but does not display successive cels of a cel animation.

Step-by-step

Controlling animation with OpenScript messages
Creating a cel animation
Getting and setting animation properties
Setting animation options

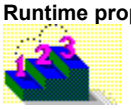


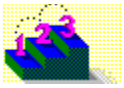
```
--Sets second animation of object to be a cel
--animation (object being animated must be a group)
anim_CelAnimation(2) of ellipse "basketball" = true
```



anim_CurrentStep

Runtime property





Syntax

```
get anim_CurrentStep(<animNumber>) of <objectRef>
anim_CurrentStep(<animNumber>) of <objectRef> = <value>
```

Parameters

<animNumber>
The animation number to get or set. If you do not indicate an animation number, Multimedia ToolBook uses the first animation.

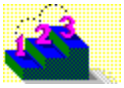
<objectRef>
The object whose animation to get or set.

Description

A runtime property containing the current step of the animation. Set this property to skip to particular points in the path. Setting this property has no effect unless `anim_ShowAllSteps` is set to `true`.

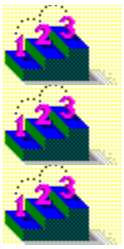
Value

Number of the current step. If the animation is not running, the value is -1.

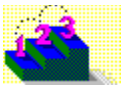


Step-by-step

Controlling animation with OpenScript messages
Getting and setting animation properties
Setting animation options

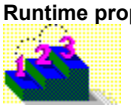


```
--Displays current step in status bar as object
--moves along animation path; put handler in script
--of animated object
to handle playStep
   caption of statusBar = anim_CurrentStep of target
   forward      --Always forward playStep message
end playStep
```



anim_CurrentTime

Runtime property



Syntax `get anim_CurrentTime(<animNumber>) of <objectRef>`

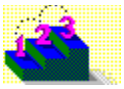
Parameters

`<animNumber>`
The animation number to get. If you do not indicate an animation number, Multimedia ToolBook uses the first animation.

`<objectRef>`
The object whose animation to get.

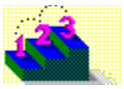
Description A runtime property containing the elapsed time since Windows was started. Use the value of this property as a reference time for calculating elapsed time. You cannot set this property.

Value Integer value indicating elapsed time in milliseconds. If the animation is not running, the value is -1.



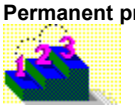
Step-by-step

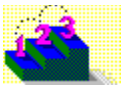
Controlling animation with OpenScript messages
Creating a cel animation
Getting and setting animation properties
Setting animation options



anim_Duration

Permanent property





Syntax

```
get anim_Duration(<animNumber>) of <objectRef>
anim_Duration(<animNumber>) of <objectRef> = <value>
```

Parameters

<animNumber>
The animation number to get or set. If you do not indicate an animation number, Multimedia ToolBook uses the first animation.

<objectRef>
The object whose animation to get or set.

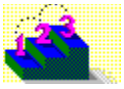
Description

A permanent property specifying the time that one iteration along the path should take. If `anim_ShowAllSteps` is set to `true`, the number of animation steps in `anim_NumSteps` takes precedence over this property, so the animation may take longer than the time specified.

Changes to this property do not take effect until the next time you run the animation.

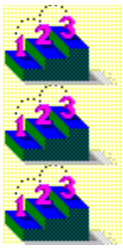
Value

A decimal number specifying the duration in seconds.

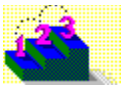


Step-by-step

Controlling animation with OpenScript messages
Getting and setting animation properties
Setting animation options

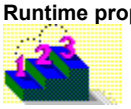


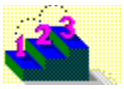
```
--Sets animation speed by changing duration based
--on radio button settings
to handle buttonClick
   conditions
      when checked of button "Slow" = true
         duration = 15
      when checked of button "Medium" = true
         duration = 10
      when checked of button "Fast" = true
         duration = 5
   end conditions
   anim_Duration of target = duration
end
```



anim_ElapsedTime

Runtime property





Syntax `get anim_ElapsedTime(<animNumber>) of <objectRef>`

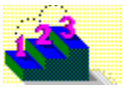
Parameters

`<animNumber>`
The animation number to get. If you do not indicate an animation number, Multimedia ToolBook uses the first animation.

`<objectRef>`
The object whose animation to get.

Description A runtime property containing the elapsed time for the animation currently running. This property is valid only while an animation is running. You cannot set this property.

Value Number of milliseconds since the animation was started. If the animation is not running, the value is -1.

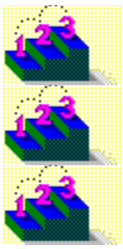


Step-by-step

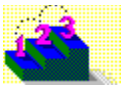
Controlling animation with OpenScript messages
Getting and setting animation properties
Setting animation options

OpenScript reference

`anim_CurrentTime`
`anim_StartTime`

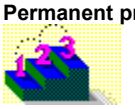


```
--Displays elapsed time over the total duration
to handle playStep
   duration = anim_Duration of target
   et = ceiling(anim_ElapsedTime of target / 1000)
   text of field "status" = et & "/" & duration
   forward
end
```



anim_EndSize

Permanent property



Syntax `get anim_EndSize(<animNumber>) of <objectRef>`

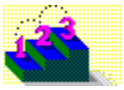
Parameters

`<animNumber>`
The animation number to get. If you do not indicate an animation number, Multimedia ToolBook uses the first animation.

`<objectRef>`
The object whose animation to get.

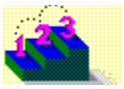
Description A permanent property indicating the size of the object when the animation ends. You cannot set this property.

Value List of two integers indicating the size of the object in page units.



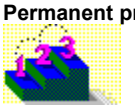
Step-by-step
Changing an object's size during an animation
Getting and setting animation properties

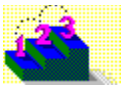
OpenScript reference
`anim_StartSize`



anim_NumRepetitions

Permanent property





Syntax

```
get anim_NumRepetitions(<animNumber>) of <objectRef>
anim_NumRepetitions(<animNumber>) of <objectRef> = <value>
```

Parameters

<animNumber>
The animation number to get or set. If you do not indicate an animation number, Multimedia ToolBook uses the first animation.

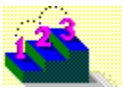
<objectRef>
The object whose animation to get or set.

Description

A permanent property that determines the number of times the object will go along the path before the animation stops.

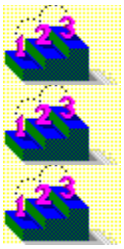
Value

A positive integer indicating the number of times to repeat the animation, or zero to cause the animation to run continuously.

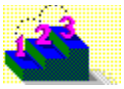


Step-by-step

Controlling animation with OpenScript messages
Getting and setting animation properties
Setting animation options

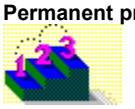


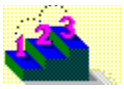
```
to handle buttonClick
   ask "Repeat how many times?"
   if sysError <> "ok" then
      break buttonClick
   end if
   if isType("int",It) then
      anim_NumRepetitions of target = It
   end if
end buttonClick
```



anim_NumSteps

Permanent property





Syntax

```
get anim_NumSteps(<animNumber>) of <objectRef>
anim_NumSteps(<animNumber>) of <objectRef> = <value>
```

Parameters

<animNumber>
The animation number to get or set. If you do not indicate an animation number, Multimedia ToolBook uses the first animation.

<objectRef>
The object whose animation to get or set.

Description

A permanent property containing the number of steps in the compiled path, based on this formula:

```
steps per Second * duration
```

The actual number of steps is adjusted depending on the `rateType` and on whether the animation is looped (where the first vertex is within two pixels of the last vertex). The final number of steps is calculated using this formula:

```
itemCount(compiled path) DIV 4 -1
```

The value in this property is useful for creating status updates such as a percentage-done bar.

Changing the value of this property sets how far along the path the object will move. It also indirectly sets the speed and smoothness of the animation; the object will move the number of specified steps in the time specified in the `anim_Duration` property. The higher the value of `anim_NumSteps`, the further along the path the object moves, and the faster it moves, because it moves more steps in the specified time. Changes to this property do not take effect until the next time you run the animation.

Value

A number denoting the number of steps in the animation.

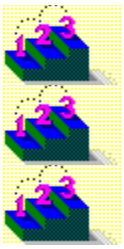
Note You should not set the value of `anim_NumSteps` higher than the value returned by the formula for the final number of steps as described above (the default value for this property). Multimedia ToolBook displays an error when the step count exceeds the final step value calculated by the formula.

Step-by-step

Changing an object's size during an animation
Setting animation options

OpenScript reference

`anim_CurrentStep`
`anim_Duration`
`anim_ShowAllSteps`
`anim_StepsPerCel`



```
--Displays percentage done by current step against
--total number of steps. For efficiency, anim_NumSteps
--can be determined when the animation begins and passed to
--this handler.
to handle playStep
   numSteps = anim_NumSteps of target
   curStep = anim_CurrentStep of target
   if now <> null then
      percent =  ceiling((curStep/numSteps)*100)
      caption of statusBar = percent & "%"
   end if
   forward
end

--Changes number of steps, checking that value does not exceed
--original (default) value
to handle buttonClick
   --Preserve default value of property in user property
   if orig_NumSteps of target = null then
      orig_NumSteps of target = anim_NumSteps of target
   end
   numSteps = anim_NumSteps of target
   msg = "Current # of steps =" && numSteps & crlf & "Enter new value."
   ask msg
   if sysError = "ok" then
      if It > orig_NumSteps of target then
         It = orig_NumSteps of target
      end
      anim_NumSteps of self = It
   end
   send playAnimation 1 to self
end
```

anim_Offset

Runtime property

Syntax

```
get anim_Offset(<animNumber>) of <objectRef>
anim_Offset(<animNumber>) of <objectRef> = <value>
```

Parameters

<animNumber>
The animation number to get or set. If you do not indicate an animation number, Multimedia ToolBook uses the first animation.

<objectRef>
The object whose animation to get or set.

Description

A runtime property containing the amount to offset the step positions while an animation is playing. Set this property to move an animation path while the animation is playing.

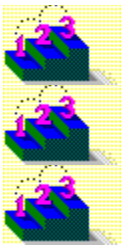
The compiled path of an animation is based on an offset of 0,0 so it can be easily manipulated; when each step is played, Multimedia ToolBook adds the value of the offset to step position stored in the compiled path to get the actual step position. The initial value for this property comes from the object's `anim_startPosition` animation property.

Value

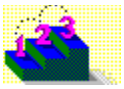
A list specifying the X and Y offset values in page units.

Step-by-step

Controlling animation with OpenScript messages
Getting and setting animation properties
Setting animation options

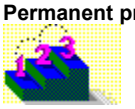


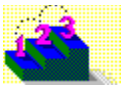
```
--Causes the ellipse "Moon" to animate around the object
--while the object itself is animated
to handle buttonUp
   send playAnimation 1
   send playAnimation 1,null,true to ellipse "Moon"
end

--Makes the ellipse moon "orbit" around self
to handle playStep
   --Stores original position
   oldPos = my position
   --Animates self
   forward
   --Gets new position
   newPos = my position
   --Calculates movement of self
   dx = item 1 of newPos - item 1 of oldPos
   dy = item 2 of newPos - item 2 of oldPos
   --Gets the old offset of the ellipse
   get anim_Offset of ellipse "moon"
   -- -1 means the ellipse isnt being animated
   if it <> -1
      --Changes the offset by the amount of movement
      get item 1 of it + dx, item 2 of it + dy
      set anim_Offset of ellipse "moon" to it
   end
end
```



anim_ShowAllSteps

Permanent property





Syntax

```
get anim_ShowAllSteps(<animNumber>) of <objectRef>
anim_ShowAllSteps(<animNumber>) of <objectRef> = <value>
```

Parameters

<animNumber>
The animation number to get or set. If you do not indicate an animation number, Multimedia ToolBook uses the first animation.

<objectRef>
The object whose animation to get or set.

Description

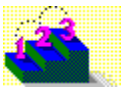
A permanent property that determines whether to skip steps when the animation gets behind. If this property is `true`, Multimedia ToolBook shows all steps in the animation path. If this property is `false`, the animation runs for the amount of time specified in `anim_Duration`, even if Multimedia ToolBook has to skip some steps to complete the animation in time.

Setting `anim_ShowAllSteps` to `true` is useful if the animation would look bad with missing steps, while setting it to `false` is useful if it is important that the animation finish within a specified time.

Changes to this property do not take effect until the next time you run the animation.

Value

`True` or `false`.

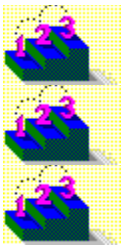


Step-by-step

Controlling animation with OpenScript messages
Getting and setting animation properties
Setting animation options

OpenScript reference

`anim_CurrentStep`
`anim_Duration`
`anim_StepsPerCel`



anim_showAllSteps of ellipse "basketball" = true

anim_StartPosition

Permanent property

Syntax

```
get anim_StartPosition(<animNumber>) of <objectRef>
anim_StartPosition(<animNumber>) of <objectRef> = <value>
```

Parameters

<animNumber>
The animation number to get or set. If you do not indicate an animation number, Multimedia ToolBook uses the first animation.

<objectRef>
The object whose animation to get or set.

Description

A permanent property containing the starting position of an animation. The value in this property is added to the values in `anim_Offset` to determine where to start the animation.

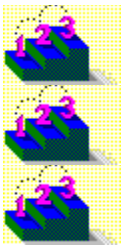
Changes to this property do not take effect until the next time you run the animation.

Value

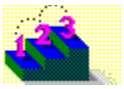
A list in page units of the start position for the animation.

Step-by-step

Changing an object's size during an animation
Controlling animation with OpenScript messages
Getting and setting animation properties
Moving an animation path
Setting animation options

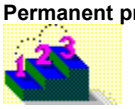


```
--Book script to move the starting position of an animation
--when the user presses Shift+arrow keys
to handle keyUp key, isShift, isCtrl
   startPos = anim_StartPosition(1) of ellipse "basketball"
   conditions
      when key = keyLeftArrow and isShift = true
         decrement item 1 of startPos by 720
      when key = keyRightArrow and isShift = true
         increment item 1 of startPos by 720
      when key = keyUpArrow and isShift = true
         decrement item 2 of startPos by 720
      when key = keyDownArrow and isShift = true
         increment item 2 of startPos by 720
      else
         forward
   end conditions
   anim_StartPosition(1) of ellipse "basketball" = startPos
end
```



anim_StartSize

Permanent property



Syntax `get anim_StartSize(<animNumber>) of <objectRef>`

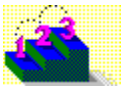
Parameters

`<animNumber>`
The animation number to get. If you do not indicate an animation number, Multimedia ToolBook uses the first animation.

`<objectRef>`
The object whose animation to get.

Description A permanent property indicating the size of the object when the animation begins. You cannot set this property.

Value List of two integers indicating the size of the object in page units.

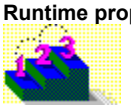


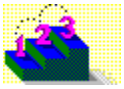
Step-by-step
Changing an object's size during an animation
Getting and setting animation properties

OpenScript reference
`anim_EndSize`

anim_StartTime

Runtime property





Syntax `get anim_StartTime(<animNumber>) of <objectRef>`

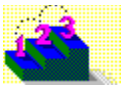
Parameters

`<animNumber>`
The animation number to get. If you do not indicate an animation number, Multimedia ToolBook uses the first animation.

`<objectRef>`
The object whose animation to get.

Description A runtime property containing the start time of the object's current animation. The value of this property is compared to the current time and the duration of the animation to calculate which step to show next. You cannot set this property.

Value Number in milliseconds based on the system time. If the animation is not running, the value is -1.

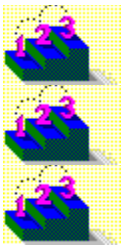


Step-by-step

Controlling animation with OpenScript messages
Getting and setting animation properties
Setting animation options

OpenScript reference

`anim_Duration`
`anim_CurrentStep`
`anim_ShowAllSteps`
`anim_StepsPerCel`



```
get anim_StartTime of ellipse "basketball"
```

anim_StepsPerCel

Permanent property

Syntax

```
get anim_StepsPerCel(<animNumber>) of <objectRef>
anim_StepsPerCel(<animNumber>) of <objectRef> = <value>
```

Parameters

<animNumber>
The animation number to get or set. If you do not indicate an animation number, Multimedia ToolBook uses the first animation.

<objectRef>
The object whose animation to get or set.

Description

A permanent property that determines how many steps along the path the object moves before the next cel in a cel animation is shown. For example, if the value is 5, Multimedia ToolBook completes five steps in the animation before showing the next cel. To change cels more slowly, use a higher value; to change cels more quickly use a lower value.

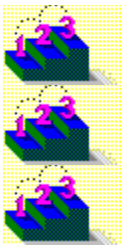
Changes to this property do not take effect until the next time you run the animation.

Value

Integer greater than zero indicating the number of steps to take before showing the next cel.

Step-by-step

Controlling animation with OpenScript messages
Creating a cel animation
Getting and setting animation properties
Setting animation options



```
to handle buttonClick
   --Sets speed at which cel animation changes cels based
   --on radio button settings
   conditions
      when checked of button "Slow" = true
         speed = 10
      when checked of button "Medium" = true
         speed = 5
      when checked of button "Fast" = true
         speed = 1
   end conditions
   anim_StepsPerCell(1) of group "globe" = speed
   send playAnimation 1 to group "globe"
end
```

doneAnimatingNotify

Message

Syntax `doneAnimatingNotify <status>,<object identifier>`

Parameters

`<status>`
The value `successful` if the animation completed; `aborted` otherwise.

`<object identifier>`
A reference to the object whose animation just finished.

Description Sent by Multimedia ToolBook when an object finishes its animation to the object specified for notification in the `playAnimation` message.

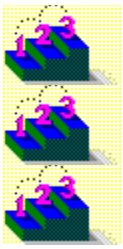
Note The `doneAnimatingNotify` script belongs to the object specified as the `notifyObject` in the `playAnimation` call that started the animation.

Step-by-step

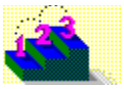
Controlling animation with OpenScript messages
Requesting notification by animations
Starting an animation

OpenScript reference

`jumpToPercent`
`jumpToStep`
`playAnimation`
`playStep`
`restoreAnimation`
`stopAnimation`

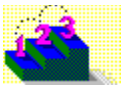


```
--Hides objects after they finish animating
to handle doneAnimatingNotify status, whatObject
   if status is "successful"
      send restoreAnimation to whatObject
   end
end
```



jumpToPercent

Message



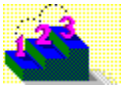
Syntax `jumpToPercent <percentComplete>, <animNumber>`

Parameters

`<percentComplete>`
How far along the path to move as a decimal percent. The number should be between 0 and 1.

`<animNumber>`
The number of the animation along which the object should move.

Description Sent to an object whose animation is not running to move the object to the point along its path nearest the specified percentage. This is useful if you are manually stepping through an animation. This message does not start the animation.

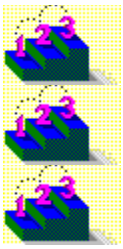


Step-by-step

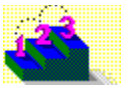
Controlling animation with OpenScript messages
Starting an animation

OpenScript reference

`doneAnimating`
`jumpToStep`
`playAnimation`
`playStep`
`restoreAnimation`
`stopAnimation`

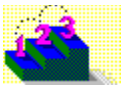


```
--Moves the object half way to the end of path 3
send jumpToPercent 0.5, 3 to ellipse "Moon"
```



jumpToStep

Message



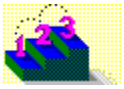
Syntax `jumpToStep <stepNumber>, <animNumber>`

Parameters

`<stepNumber>`
The step to show.

`<animNumber>`
The number of the animation along which the object should move.

Description Sent to an object whose animation is not currently running to position it at the point along its path specified by `stepNumber`. This is useful if you are manually stepping through an animation. This message does not start the animation.

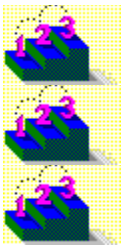


Step-by-step

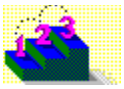
Controlling animation with OpenScript messages
Starting an animation

OpenScript reference

`doneAnimating`
`jumpToPercent`
`playAnimation`
`playStep`
`restoreAnimation`
`stopAnimation`

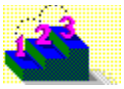


```
send jumpToStep 26, 1 to ellipse "Moon"
```



playAnimation

Message



Syntax `playAnimation <animNumber>[,<notifyObject>[,<wait>]]`

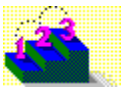
Parameters

`<animNumber>`
The number from among the object's animations to play.

`<notifyObject>`
An object to send the doneAnimatingNotify message to when the object's animation is done playing. This parameter can be null, but you must include a placeholder for this parameter to use the <wait> option.

`<wait>`
If `true`, specifies that the animation should complete before returning control to the handler that sent the message. If `false` or `null`, the animation begins and control returns to the calling handler

Description Sent to an object to cause the object's specified animation to start playing.

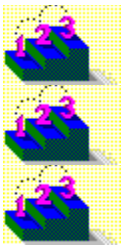


Step-by-step

Controlling animation with OpenScript messages
Requesting notification by animations
Starting an animation

OpenScript reference

`doneAnimating`
`jumpToPercent`
`jumpToStep`
`playStep`
`restoreAnimation`
`stopAnimation`



```
send playAnimation 1,this page,true to ellipse "Moon"
```

playStep

Message

Syntax `playStep <animNumber>`

Parameter **`<animNumber>`**
The number from among the object's animations currently playing.

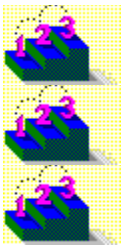
Description Sent by Multimedia ToolBook to an object currently running an animation to advance the object to the next step in the path. You can write a handler for this message to update status counters or perform other processing; be sure to forward the message. Do not send this message yourself.

Step-by-step

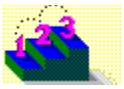
Controlling animation with OpenScript messages
Requesting notification by animations
Starting an animation

OpenScript reference

`doneAnimating`
`jumpToPercent`
`jumpToStep`
`playAnimation`
`restoreAnimation`
`stopAnimation`

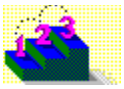


```
--Starts object's animation when object is clicked
to handle buttonClick
   system halfway    --Establishes variable used in playStep handler
   system soundAlreadyPlayed   --Flag indicating if sound clip played
   soundAlreadyPlayed = false
   mmOpen clip "chimes"
   halfway = (anim_Duration of self) * 1000 / 2
   send playAnimation 1 to self
end

--Monitors the animation at each step. When the
--animation is half completed, plays a .WAV file
to handle playStep
   system halfway         --Variable set earlier when animation began
   system soundAlreadyPlayed
   if anim_ElapsedTime of target > halfway then
      if soundAlreadyPlayed = false then
         mmPlay clip "chimes"
         soundAlreadyPlayed = true
      end if
   end if
   forward
end playStep
```



restoreAnimation

Message



Syntax `restoreAnimation [<objectList>, <animationList>]`

Parameters

`<objectList>`
A list of objects whose animations should be reset.

`<animationList>`
A list of animations corresponding to the items in `<objectList>`, which allows you to move an object to the starting position of a particular animation.

Description Sent to an object to reset the object to the starting point of an animation. If the `<objectList>` parameter is set, the objects in the list are reset instead of the object that is the target of the message.

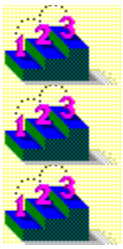
Use the `restoreAnimation` message to reset an animation so it is at its starting point when started again. Otherwise, when an animation is started again, it moves to the starting point from its current location.

Step-by-step

Controlling animation with OpenScript messages
Requesting notification by animations
Starting an animation

OpenScript reference

`doneAnimating`
`jumpToPercent`
`jumpToStep`
`playAnimation`
`playStep`
`stopAnimation`



```
send restoreAnimation to ellipse "Moon"
send restoreAnimation objects of this page
send restoreAnimation button "Guy" 3
```

stopAnimation

Message

Syntax `stopAnimation <status>`

Parameter **`<status>`**
Contains the value `successful` if the animation reaches its end, or `aborted` if the animation is stopped early.

Description Sent to an object to stop the animation in progress and leave the object where it was when the message was sent.

Multimedia ToolBook sends this message to objects when

- the object reaches the end of the animation path.
- the user presses the Esc key.
- the user moves to a page that does not display the object being animated.

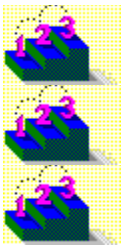
You cannot send this message to stop animations played with the `wait` parameter because the script running the animation must finish before any other OpenScript statements can be executed.

Step-by-step

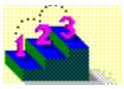
Controlling animation with OpenScript messages
Requesting notification by animations
Starting an animation

OpenScript reference

`doneAnimating`
`jumpToPercent`
`jumpToStep`
`playAnimation`
`playStep`
`restoreAnimation`

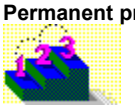


```
send stopAnimation to ellipse "Moon"
```



anim_AnimationSettings

Permanent property



Description A user property of an animated object containing a table of the animation paths of the object and their settings. The table consists of a two-dimensional dynamic array in which the first element identifies an animation and the second element an animation setting.

You can make changes to the settings in this array by editing the animation in the Animation window. Alternatively, some of the values in the array are accessible via properties of the animated object.

If you require access to values in the table that are not already available via properties, you can write `to get` and `to set` handlers that access the array directly. For details about accessing the array directly, see Creating handlers to access the animation array.

Although you can change values in the array using `to set` handlers, some changes do not take effect unless you recompile the array by editing it in the Animation window. For example, you can change element 9 (number of steps), but Multimedia ToolBook calculates the actual number of steps based on the number of bounds in element 10 (compiled path).

Note Before making changes to the contents of the animation array, be sure you understand the structure and purpose of each element. If you make an error when setting the value of an element in the array, the animation might not run, and you might have to redraw it from scratch.

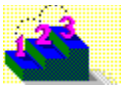
Value `null` if the object does not have any animations or an [n] by 16-element array of animation settings, where "n" is the number of animations you created for the object. The structure of the table is as follows:

Elements in* `anim_AnimationSettings` *array

Value	Description	For more details, see
[n][1]	Path; list of coordinates for the vertices in the path.	
[n][2]	Curved vertices; list of `true` or `false` corresponding to each vertex in the path indicating whether the vertex defines a curve.	
[n][3]	Rate type; `constant` if the object moves along the path at a steady speed, `variable` if the animation speed depends on the length of the segment.	"Animation rates" under Setting animation options
[n][4]	Cel animation; `true` or `false` to determine whether an animation plays cels. You can set this value by checking Turn On in the Animation Settings dialog box	"Cel Animation On" under Setting animation options; `anim_CelAnimation` property
[n][5]	Show all steps; determines whether to skip steps when the animation gets behind.	"Show All Steps" under Setting animation options; `anim_ShowAllSteps` property
[n][6]	Start position; a list in page units of an animation's starting position.	`anim_StartPosition` property
[n][7]	Repetitions; the number of times the object will go along the path before the animation stops.	"Repetitions" under Setting animation options; `anim_NumRepetitions` property
[n][8]	Duration; the time that one iteration	"Duration in Seconds" under Setting

	along the path should take.	animation options; `anim_Duration` property
[n][9]	Number of steps; the total number of steps in the animation.	`anim_NumSteps` property
[n][10]	Compiled path; list of the bounds of the object at each step of the animation.	
[n][11]	Steps per cel; how many steps along the path the object moves before the next cel in a cel animation is shown.	"Steps per Cel" under Setting animation options; `anim_StepsPerCel` property
[n][12]	Step rate; how many steps along the path the object will move each second.	"Steps per Second" under Setting animation options
[n][13]	Fixed path; `false` if path moves when object is moved at Author level; `true` if path remains in place.	"Update Path on Move" under Setting animation options
[n][14]	Row number; the number of the current animation (corresponds to the value of the first element of the `anim_AnimationSettings` array).	
[n][15]	Start size; the size of the object in page units at the first step of the animation. (This size is not necessarily the same as the size of the object before the animation starts, because the object may be resized when the animation begins.)	
[n][16]	End size; the size of the object in page units at the last step of the animation.	

Note This array is subject to change in future versions of the Path Animation utility.



Step-by-step
Setting animation options

OpenScript reference
anim_CelAnimation
anim_CurrentStep
anim_CurrentTime
anim_Duration
anim_ElapsedTime
anim_NumRepetitions
anim_NumSteps
anim_Offset
anim_ShowAllSteps
anim_StartPosition
anim_StartTime
anim_StepsPerCel

Creating handlers to access the animation array

Step-by-step

Multimedia ToolBook stores information about animation settings in a two-dimensional array as a user property (called `anim_AnimationSettings)` of the animated object. You can get and set many of the values in the array using animation properties such as `anim_Duration` and `anim_NumSteps`.

However, if you want access to the array values for which there is no corresponding property, you can write `to get` and `to set` handlers. One method is to write a single `to get` handler for each element of the array you want to be able to access, using a passed parameter to indicate which of the object's animations you want to access.

Note Before making changes to the contents of the animation array, be sure you understand the structure and purpose of each element. If you make an error when setting the value of an element in the array, the animation might not run, and you might have to redraw it from scratch.

To write the handler, you must know what element of the array contains the information you want. Declare a local two-dimensional, dynamic array, copy the user property into it, then extract the value you want. For details about the layout of the array, see `anim_AnimationSettings`. For example, the following handler illustrates how you can get element 15 (start size) of the array.

```
to get anim_StartSize rowNum
  local settingsTable[][]
  --Defaults to animation 1 if no other is specified
  if rowNum = null then
    rowNum = 1
  end
  settingsTable = anim_animationSettings of target
  return settingsTable[rowNum][15]
end
```

Put the handler in the script of the book that contains the object being animated. You can then get the array value using an OpenScript command such as this one:

```
get anim_StartSize of ellipse "basketball"
```

To get the value of the start size for the object's second animation, you can use a command such as this:

```
get anim_StartSize of ellipse "basketball"
```

If you write a corresponding `to set` handler, you can change the value of the array element as well using a statement such as this:

```
anim_StartSize of ellipse "basketball" = 1000,1000
```

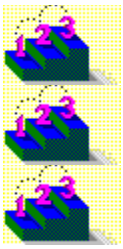
However, remember that although you can change values in the animation array, some of them do not take effect until you recompile the animation by editing it in the Animation window.

For further examples of `to get` and `to set` handlers that access the animation array, click Example.

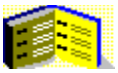
Note The `anim_AnimationSettings` array is subject to change in future versions of the Path Animation tool.

Step-by-step
Setting animation options

OpenScript reference
`anim_AnimationSettings`



```
--Returns value out of the anim_AnimationSettings array
to get anim_FixedPath rowNum
   local settingsTable[][]
   if rowNum = null then
      rowNum = 1
   end
   settingsTable = anim_animationSettings of target
   return settingsTable[rowNum][13]
end
--Sets value in anim_AnimationSettings array
to set anim_FixedPath rowNum to value
   local settingsTable[][]
   if rowNum = null then
      rowNum = 1
   end
   settingsTable = anim_animationSettings of target
   settingsTable[rowNum][13] = value
   anim_animationSettings of target = settingsTable
end
```



Glossary

The topics below provide definitions of terms related to path animation.

cel animation
compiled path
path
permanent
runtime
segment
step
vertex, vertices

cel animation

A type of animation in which individual views of an object, called cels, are shown in rapid sequence to make the object look as if it is moving or changing. For example, a cel animation of a spinning globe might consist of 16 views of the earth, each slightly different. If you see the individual cels in rapid sequence, it looks as if the globe is spinning. In Multimedia ToolBook you create cel animations by drawing individual objects to act as cels, then grouping them.

compiled path

A property of an animated object containing the coordinates for the object at each step of the animation. Multimedia ToolBook plays the animation by using the next set of coordinates (four coordinates per step) from the compiled path and setting the object's bounds to them.

The number of steps in the compiled path is based on the duration and steps per second specified when you saved the animation in the Animation window. A compiled path with 30 steps (including the start and end point) will have 120 items in its compiled path.

path

The course that the object will follow as it moves during the animation. You create a path in the Animation window by clicking to create vertices that define the path. By setting the number of steps per second, the total duration of the animation, and the animation rate, you can control how quickly and smoothly the object moves along the path. To play the animation and move it along its path, send the `playAnimation` message to the object.

The path is stored as a list of coordinates for each vertex in the object at each step of the animation.

permanent

Of a property, that you can get its value at any time, even if the animation is not running. If you set a permanent property while an animation is running, you affect the way the current animation runs and the way it runs the next time it is played.

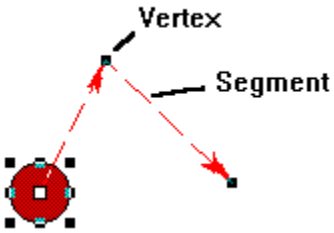
runtime

Of a property, that it contains a valid value only when the animation is running. When the animation is not running, the value of a runtime property is -1. If you set a runtime property, it affects only the animation currently running.

segment

One leg of the path along which the object moves. When you draw an animation path, you click to create a new vertex. Multimedia ToolBook then draws a new segment from the end of the existing path to where you clicked.

By default segments are straight lines, but you can convert them to curves.



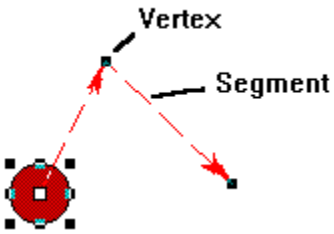
step

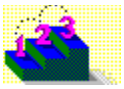
One increment along the animation path. To make the animation look smooth, specify a high number of steps per second so that the object moves only a small distance with each step.

If you are creating a cel animation, you can specify how many steps the object moves before Multimedia ToolBook shows the next cel.

vertex, vertices

The points defining the beginning and end of a segment. When you draw a path animation, you click to position a vertex, and Multimedia ToolBook draws a segment between the vertex at the end of the path and the new vertex.





Technical Support

Help menu

You can receive Asymetrix Technical Support in a variety of ways. Click a topic below for step-by-step instructions about Technical Support.

Calling Technical Support

Using the Fax Back System

Using America Online

Using CompuServe

Using Internet

Using the Asymetrix Bulletin Board System (BBS)

Using the World Wide Web

Calling Technical Support

Step by step

Registered Asymetrix product users receive 30 days of complimentary technical support, beginning with their first call to Asymetrix Technical Support. Phone support can be extended beyond the complimentary period by purchasing a technical support contract.

Use the appropriate Asymetrix technical support telephone numbers for your location as listed below.

Australia/Asia Pacific

Infotainment Asia Pacific Pty Ltd.

(61+3) 5255-471 (direct) 9 a.m. to 6 p.m. Eastern Standard Time
(61+3) 5255-482 (fax)

XLTECH Pty Ltd.

(61+2) 9752-111 (direct) 9 a.m. to 6 p.m. Eastern Standard Time
(61+2) 9752-167 (fax)

Europe (except France, Germany, and United Kingdom); Middle East; Africa; Russia

(44) 1923-208433 9:00 to 17:00 GMT
(44) 1923-208419 (fax)

France

05-90-83-19 (freephone) 9:00 to 17:00 GMT
(44) 1923-208419 (fax)

Germany

01-30-81-27-07 (freephone) 9:00 to 17:00 GMT
(44) 1923-208419 (fax)

United Kingdom

0800-716957 (freephone) 9:00 to 17:00 GMT
(44) 1923-208419 (fax)

U.S.A. and rest of world

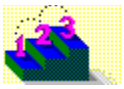
(206) 637-1600 (direct) 6 a.m. to 6 p.m. Pacific time, Monday through Thursday; 6
(206) 454-0672 (fax) a.m. to 3 p.m. Friday

When you call Technical Support, please be at your computer with your Asymetrix documentation and have the following information:

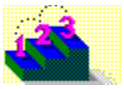
Your Asymetrix product serial number (found on the first installation disk, your license agreement envelope, or your Asymetrix product box).

The version of the Asymetrix product you are running.

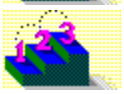
Your system configuration information (you can find this information by choosing System Info from the Help menu, or by double-clicking the System Info icon in the Asymetrix product program group, located in the Program Manager; or by double-clicking Windows Setup, located in the Program Manager in the Main group).



The exact wording of any error message you have encountered.

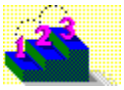


What happened and what you were doing when the problem occurred.



How you tried to solve the problem.

When you leave a message or send a fax, please include the information listed above. A technical support representative will respond within two business days to messages sent and faxes left before or after business hours.



Step by step
Using America Online
Using CompuServe
Using Internet
Using the Asymetrix Bulletin Board System (BBS)
Using the Fax Back System
Using the World Wide Web

Using the Asymetrix Bulletin Board System (BBS)

Step by step

You can receive online technical support on the Asymetrix Bulletin Board System (BBS) by leaving a message for the Asymetrix System Operator. The System Operator checks the BBS daily for new messages and answers them within two business days (Monday through Friday). To help the System Operator answer your question, gather information about your system as described in Step 5 below.

The data configuration for the BBS is: no parity, 8 data bits, 1 stop bit.

To connect to the Asymetrix Bulletin Board System:

1 Call the Asymetrix BBS number:

If you have a 1200 to 9600 baud modem (v.32bis), call (206) 451-1173.

If you have a 9600 to 14,400 baud modem (v.32bis), call (206) 451-8290.

2 Enter your first and last names at the prompts.

If you are a new BBS user and you see a welcome message, there is probably another user with the same name. Hang up the phone, redial, and use a different name, such as "Jim" instead of "James," or include your middle initial.

3 Enter your password, then retype it to confirm its spelling.

4 If you are a new user, complete the new user questionnaire.

Make sure you have your Asymetrix product serial number (found on the first installation disk, your license agreement envelope, or your Asymetrix product box). You will be asked about your software and hardware. Where applicable, it is recommended that you accept the defaults.

5 To leave a message for the System Operator, type C for "Comments to the sysop," then leave a message containing the following information:

All the information gathered about your system configuration. For details on gathering this information, see instructions below.

The version of DOS you are running.

The version of Windows you are running.

The version of the Asymetrix product you are running.

The exact wording of any error message you have encountered.

What happened and what you were doing when the problem occurred.

How you tried to solve the problem.

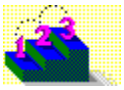
6 To log off the BBS, type G for "Goodbye," then type Y to confirm.

Tip At any time, you can type "?" to get online command help. If you have any problems or difficulties with the Asymetrix BBS, leave a message for the BBS System Operator or call the Asymetrix Technical Support line at

(206) 637-1600.

To gather system configuration information for the System Operator:

1. Choose System Info from the Help menu. If you cannot run your Asymetrix product, double-click the System Info icon in the product's program group, located in the Program Manager.
2. Copy down all the information listed in the System Status box.



Step by step
Calling Technical Support
Using America Online
Using CompuServe
Using Internet
Using the Fax Back System
Using the World Wide Web

Using CompuServe

Step by step

If you are a CompuServe member, you can receive online technical support by leaving a message for the Asymetrix Forum Moderator. The Forum Moderator checks CompuServe daily for new messages and answers them within two business days (Monday through Friday). To help the Forum Moderator answer your question, gather information about your system as described in Step 3 below.

Refer to your CompuServe documentation for step-by-step instructions for logging onto CompuServe, leaving messages for the Forum Moderator, and logging off CompuServe.

To connect to CompuServe:

1. Log on to CompuServe.
2. To connect to

Windows Third Party Developer A forum, Section 1, type `GO ASYMETRIX` or `GO WINAPA` at the prompt.

Multimedia Vendors forum, Section 15, type `GO MULTIVEN` at the prompt.

IBM Ultimedia Tools A forum, Section 5, type `GO ULTIATOOLS` at the prompt.

3. In the Asymetrix section (Section 1), leave a message for the Forum Moderator containing the following information:

All the information gathered about your system configuration. For details on gathering this information, see instructions below.

Your Asymetrix product serial number (found on the first installation disk, your license agreement envelope, or your Asymetrix product box) and version.

The version of DOS you are running.

The version of Windows you are running.

The exact wording of any error message you have encountered.

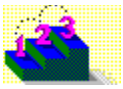
What happened and what you were doing when the problem occurred.

How you tried to solve the problem.

4. Log off CompuServe.

To gather system configuration information for the Forum Moderator:

1. Choose System Info from the Help menu. If you cannot run your Asymetrix product, double-click the System Info icon in the product's program group, located in the Program Manager.
2. Copy down all the information listed in the System Status box.


1
2
3

Step by step

Calling Technical Support
Using America Online
Using Internet
Using the Asymetrix Bulletin Board System (BBS)
Using the Fax Back System
Using the World Wide Web

Using America Online

Step by step

If you are an America Online member, you can receive online technical support by leaving a message for the Asymetrix Forum Moderator. The Forum Moderator checks America Online daily for new messages and answers them within two business days (Monday through Friday). To help the Forum Moderator answer your question, gather information about your system as described in Step 3 below.

Refer to your America Online documentation for step-by-step instructions for logging onto America Online, leaving messages for the Forum Moderator, and logging off America Online.

Note America Online operates a text telephone (TDD) for deaf or hearing-impaired members.

To connect to America Online:

1 Log on to America Online.

2 Do one of the following:

- From the Go To menu, select Keyword, then type `Asymetrix`.
- Go to the Computing and Software area, select Industry Connection, then select Asymetrix.

3 Leave a message for the Forum Moderator containing the following information:

- All the information gathered about your system configuration. For details on gathering this information, see instructions below.
- Your Asymetrix product serial number (found on the first installation disk, your license agreement envelope, or your Asymetrix product box) and version.
- The version of DOS you are running.
- The version of Windows you are running.
- The exact wording of any error message you have encountered.
- What happened and what you were doing when the problem occurred.
- How you tried to solve the problem.

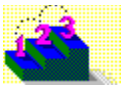
4 Log off America Online.

Tip You can receive the software required to use America Online at no cost, plus five complimentary hours of online time. For details about America Online, call (800) 827-6364 or (703) 893-6288.

To gather system configuration information for the Forum Moderator:

1 Choose System Info from the Help menu. If you cannot run your Asymetrix product, double-click the System Info icon in the product's program group, located in the Program Manager.

2 Copy down all the information listed in the System Status box.


1
2
3

Step by step
Calling Technical Support
Using CompuServe
Using Internet
Using the Asymetrix Bulletin Board System (BBS)
Using the Fax Back System
Using the World Wide Web

Using Internet

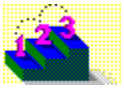
Step by step

If you have access to Internet mail, you can receive online technical support by leaving a message for the Asymetrix Forum Moderator. The Forum Moderator checks Internet daily for new messages and answers them within two business days (Monday through Friday). To help the Forum Moderator answer your question, gather information about your system as described in Step 3 below.

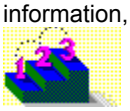
Refer to your electronic mail documentation for step-by-step instructions about starting mail and sending messages on Internet.

To access Internet:

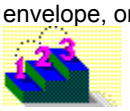
1. Start your electronic mail application.
2. In the box where you enter the receiver's address, type `support@asymetrix.com` or `techsup@asymetrix.com`.
3. Leave a message for the Forum Moderator containing the following information:



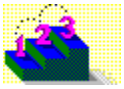
All the information gathered about your system configuration. For details on gathering this information, see instructions below.



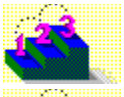
Your Asymetrix product serial number (found on the first installation disk, your license agreement envelope, or your Asymetrix product box) and version.



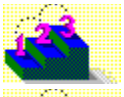
The version of DOS you are running.



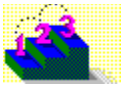
The version of Windows you are running.



The exact wording of any error message you have encountered.



What happened and what you were doing when the problem occurred.

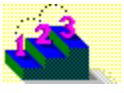


How you tried to solve the problem.

4. Send your message.

To gather system configuration information for the Forum Moderator:

1. Choose System Info from the Help menu. If you cannot run your Asymetrix product, double-click the System Info icon in the product's program group, located in the Program Manager.
2. Copy down all the information listed in the System Status box.



Step by step

Calling Technical Support
Using America Online
Using CompuServe
Using the Asymetrix Bulletin Board System (BBS)
Using the Fax Back System
Using the World Wide Web

Using the Fax on Demand System

Step by step

If you have a fax machine, you can receive answers to frequently asked questions from the Asymetrix Fax on Demand System, which is available 24 hours a day.

To use the Fax on Demand System:

1. From your fax machine or regular phone, dial (800) 770-5444 or (206) 637-5833.
2. If you do not have a list of available fax documents, or if the list you have is old, follow the directions in the Fax on Demand System's recording to receive a list. The list is updated frequently.

   If you have a current list of available documents, follow the directions in the Fax on Demand System's recording to receive up to three fax documents.
3. If you are calling from a regular phone, the Fax Back System will prompt you to enter your fax machine phone number. Enter a "'1" and your area code, followed by your phone number.

   If you are calling from a fax machine phone, the system will automatically fax to that number.

Step by step

Calling Technical Support
Using America Online
Using CompuServe
Using Internet
Using the Asymetrix Bulletin Board System (BBS)
Using the World Wide Web

Using the World Wide Web

Step by step

If you are have access to the World Wide Web (WWW), you can receive online technical information and technical support at the Asymetrix Web site. The Asymetrix Internet Tech Center Web page provides access to

answers to frequently asked questions (FAQ), organized by product.

the Asymetrix FTP (file transfer protocol) site, which includes an index of the names, size, details, and locations of all Asymetrix FTP files.

Asymetrix online technical support, where you can leave a message. You message will be answered within two business days (Monday through Friday).

To access the Asymetrix web site:

1. Load your Windows Web browser (for example, NetScape or Mosaic).
2. Enter the location (URL) of the Asymetrix web site:
   `http://www.asymetrix.com`.

To reach the Asymetrix Internet Tech Center Web page:

1. Click the Technical Services button on the Asymetrix main menu graphic.
2. Click the "Internet Tech Center" hyperlink.
3. Click the appropriate hyperlink to go to FAQ, the Asymetrix FTP site, or technical support.
4. If you click the "Submit a problem or suggestion to Asymetrix's Technical Support" hyperlink to go to technical support, enter your contact information, as well as the details of your problem, including the following information:

All the information gathered about your system configuration. For details on gathering this information, see instructions below.

Your Asymetrix product serial number (found on the first installation disk, your license agreement envelope, or your Asymetrix product box) and version.

The version of DOS you are running

The version of Windows you are running

The exact wording of any error message you have encountered

What happened and what you were doing when the problem occurred

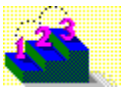
How you tried to solve the problem

To gather system configuration information for the Forum Moderator:

1. Choose System Info from the Help menu. If you cannot run your Asymetrix product, double-click the System

Info icon in the product's program group, located in the Program Manager.

2. Copy down all the information listed in the System Status box.



Step by step

Calling Technical Support
Using America Online
Using CompuServe
Using Internet
Using the Asymetrix Bulletin Board System (BBS)
Using the Fax Back System