

## Java Perk Help File

# Java Perk

[What is Java Perk?](#)

[How to Order Java Perk](#)

[Animations](#)

[Billboard Wizard](#)

[Animation Wizard](#)

[The Typewriter applet](#)

[Dynamic Buttons](#)

[Fade-in/Fade-out](#)

[Swirly Text](#)

[Scrolling Status Bar Text](#)

[Password Protection applet](#)

[Clock applet](#)

[Image Fade](#)

[Image Cube](#)

[Image Ripple](#)

[Lake Reflection](#)

[Troubleshooting](#)

## **What Is Java Perk?**

Java Perk is used to create animations and other dynamic effects for Web pages using Java applets. The applets are actually pre-made for your use, and Java Perk makes it simple to create the HTML code which controls them.

Java Perk allows you to input parameters and properties for the different applets and then creates the HTML code with a click of a button! The HTML code is then inserted into HTML documents, and the documents and applets can be uploaded to a Web site for instant use. It's that simple!

Special thanks to Eric Harshbarger and the Catalyst New Media Marketing lab at Sun Microsystems, Inc. for letting McWeb Software include their Java applets in Java Perk!!!!

Also, special thanks to Fabio Ciucci and David Griffiths for their applets.

# What's New in Java Perk

## JAVA PERK 3.1

- The Typewriter applet now supports bold, italic, and two new font styles.
- A Preview function is now built in so you don't have to insert the applet code into your HTML document to test it. Test it right in Java Perk!.
- You can now go back and edit your applet if you don't like the way it looks (Don't have to start all over again).

## JAVA PERK 3.0

- The Animation Wizard which makes it easy and fast to create animations.
- The Billboard Wizard which makes it easy to make scrolling billboard animations for advertising, etc.
- The Image Fade applet which fades in and out between two specified images. It fades from one image to another, then repeats itself.
- The Image Cube applet puts a 256x256 pixel image on each face of a cube and rotates it.
- The Image Ripple applet animates an image so that it appears to have ripples/waves on it(like in water).
- The Lake Reflection applet animates and reflects your image so that it appears to be reflecting off of water.

## JAVA PERK 2.1

- A clock applet which allows you to display the current date and time on your Web pages.
- A password protection applet which allows you allow only selected users to be able to access a Web page. These users need a specific password to view the page.

The background image parameter was fixed in the Typewriter applet. You can now specify a background image for this applet.

The x and y locations of the swirly text can now be specified so the user has better control of where the text will be located.

## **JAVA PERK 2.0**

- A button applet which allows you to specify an image to use as a button on your Web page, a second image to replace the first when the user moves the mouse over it, and a third image to be shown when the user clicks on the button.
- A Swirly Text applet to create text which continuously changes colors (like that shown on the 'Special Offer' on the Java Perk Web page).

A Fade-in/Fade-out applet which creates a fading effect on your Web page when you load or exit the Web page in your browser.

A scrolling text marquee which is displayed in the status bar of the browser.

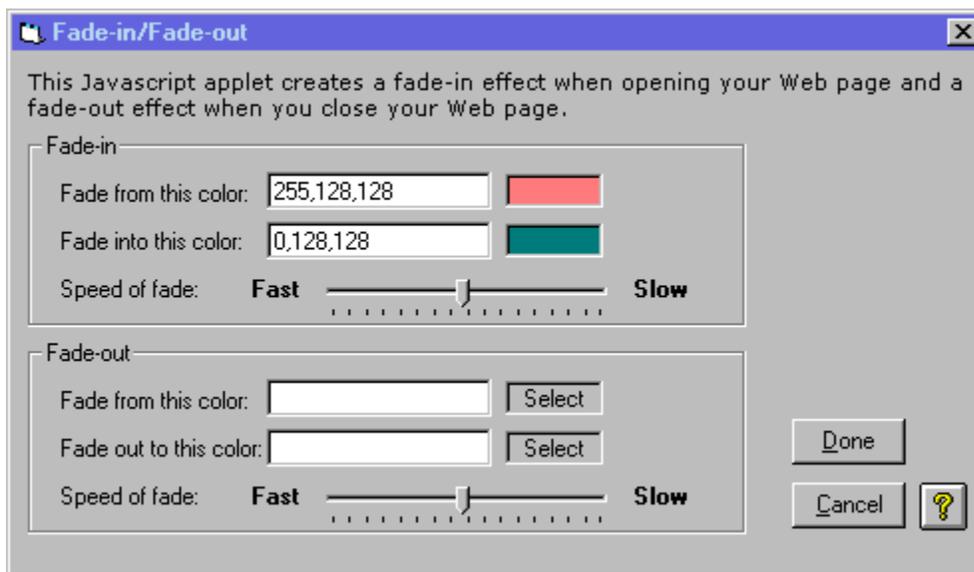
## Fade-in/Fade-out applet

This Javascript applet creates a fade-in and/or a fade-out effect for your Web pages. You can choose the starting screen color and the color to fade into.

Fade-in: When your Web page is loaded into the browser the screen will fade from one color to a second color and then display the contents of your Web page.

Fade-out: When exiting your Web page to view another Web page the screen will fade from one color to a second color and then display the next Web page.

Below is the dialog box for setting Fade-in/fade-out parameters:



You can select the color to fade from, the color to fade into and the fading speed. To select the colors, click on the boxes marked select and the color selection box will appear, allowing you to select a color.

You can have your page only fade-in, only fade-out, or have both effects on your Web page.

### Example:

Suppose you wish to have your Web page fade-in from red to blue when someone loads it in their browser.

- First, open the Fade-in/fade-out applet dialog box.

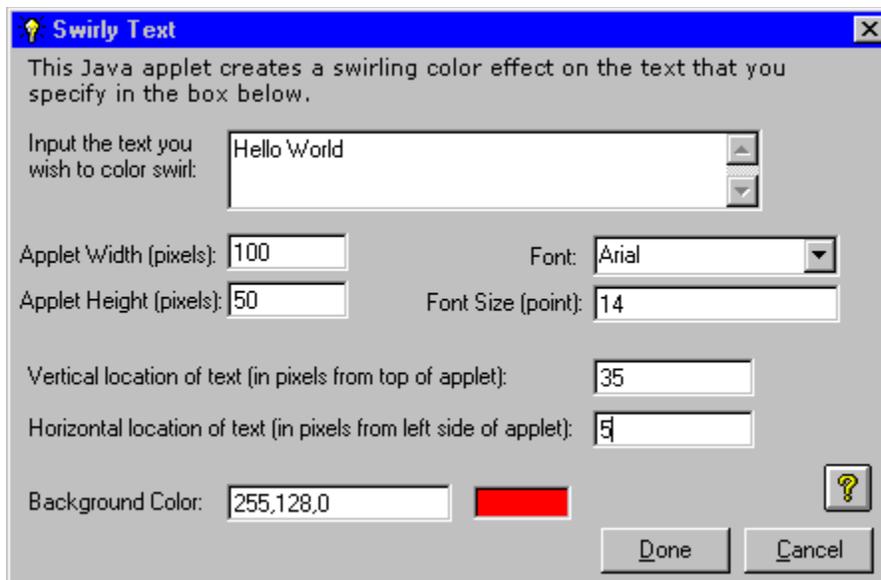
- In the section marked Fade-in, click on the box marked 'Select' next to the text box labeled 'Fade from this color'. This will display the color selection box.
- Choose the color red and click OK. The RGB color code will display in the text box and the color red will be displayed in the box.
  
- Then click on the box marked 'Select' next to the text box labeled 'Fade into this color'. This will display the color selection box.
- Choose the color blue and click OK. The RGB color code will display in the text box and the color blue will be displayed in the box.
  
- Set the speed of the fade. Set it on a slow setting so you can actually see the fade. If you set it too fast you may not see it.
  
  
- When you are done click the OK button on the Fade-in/Fade-out dialog box and the HTML code for this Javascript applet will be displayed in the code preview box for cutting and pasting into an HTML editor.  
The code generated by this particular applet is a full HTML document, so you should paste it into a blank document and insert your other Web page contents between the <BODY></BODY> tags of the text.

## Swirly text applet

This Java applet causes the specified text to change through a range of colors simulating a swirling color effect.

It requires the following applets to function properly:  
**ColorSwirl.class**

Below is the dialog box for setting Swirly text parameters:



The following are the settings for the Swirly text applet:

**Text you wish to color swirl:** In the text box enter the text you wish to display as swirly text in your Web page. This is a required setting.

**Applet Width(pixels):** This is the width of the applet (specified in pixels). The width should be large enough to show the swirly text. This is a required setting.

**Applet Height(pixels):** This is the height of the applet (specified in pixels). The height should be large enough to show the swirly text. This is a required setting.

**Font Name:** This is the name of the font you wish the swirly text to be displayed in. There are a limited number of fonts which you can use.

**Font Size:** This is the size of the font you wish the swirly text to be displayed in (specified in point size e.g., for 14 pt text type 14 into the input box).

**Background Color:** This is the background color of the applet. The specified width and height of the applet will be filled with this color on the Web browser's screen (specified in RGB values).

**X loc:** This is the x-location of the text on the horizontal axis. You can specify (in pixels) where you wish the text to start (measured from the left side of the applet).

**Y loc:** This is the y-location of the text on the vertical axis. You can specify (in pixels) where you wish the text to be located (measured from the top of the applet). For example, if you put Y-loc to be 25 pixels, then the bottom of the text would be located 25 pixels from the top of the applet.

- When you are done click the OK button on the Swirly text dialog box and the HTML code for this Java applet will be displayed in the code preview box for cutting and pasting into an existing HTML document.

## Clock applet

This Java applet displays a working clock (in numeric format) on your Web page. It also has the capability to display the date.

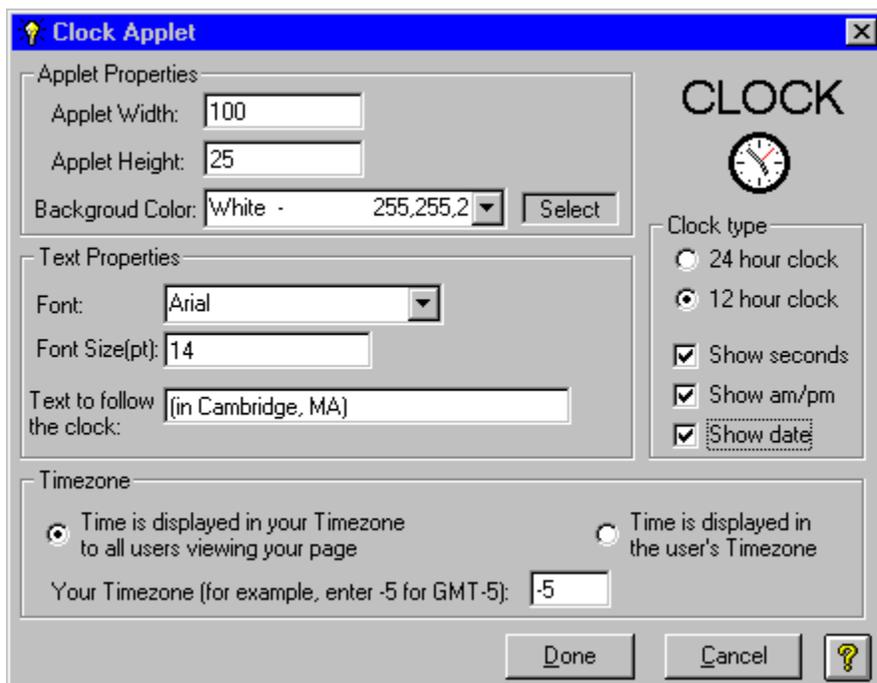
It requires the following applets to function properly:

**Clock1.class**

**AppletSettings.class**

**DetachFrame.class**

Below is the dialog box for setting Clock parameters:



The following are the settings for the Swirly text applet:

**Applet Width(pixels):** This is the width of the applet (specified in pixels). The width should be large enough to show the clock's text. This is a required setting.

**Applet Height(pixels):** This is the height of the applet (specified in pixels). The height should be large enough to show the clock's text. This is a required setting.

**Background Color:** This is the background color of the applet. The specified width

and height of the applet will be filled with this color on the Web browser's screen (specified in RGB values).

**Font:** This is the name of the font you wish the clock text to be displayed in. There are a limited number of fonts which you can use.

**Font Size:** This is the size of the font you wish the clock text to be displayed in (specified in point size e.g., for 14 pt text type 14 into the input box).

**Text to follow the clock:** This is the text that you wish to accompany the clock. Usually the text has to do with the your location. For example, I want my clock to be displayed in US Eastern Standard Time (GMT-5) for everyone to see. So someone in Australia viewing my page will see the time in Boston, Massachusetts time, not Australia time. I need to type some text like "in Boston Massachusetts" to accompany the clock so that people will know what this particular time signifies. As an example, the text you would type is "in Boston, Massachusetts" or "EST" or "GMT-5".

**Timezone:** This is the timezone you wish to set the clock at. If you wish to have the clock display the time in the user's time zone, then click on the "Time is displayed in the user's timezone" button. If you wish all users to see the time in your time zone, then click on the "Time is displayed in your timezone to all users viewing your page" button and enter your GMT timezone setting in the provided input box. Enter -5 for GMT-5 (Eastern Standard time), -8 for GMT-8 (Pacific Standard Time), etc.

**24 hour/ 12 hour clock:** This allows you to set the clock to display in either 12 hour or 24 hour display.

**Show seconds:** This setting determines whether seconds are shown on the clock.

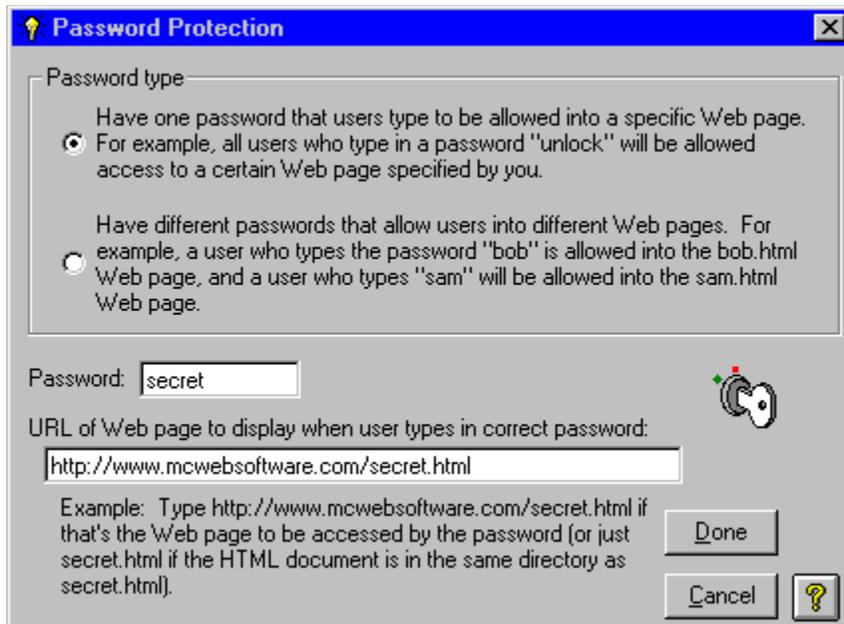
**Show am/pm:** This setting determines whether the words am or pm are shown on the clock.

**Show date:** This setting determines whether the date is shown on the clock.

## Password Protection applet

This Java applet allows the user to use password.protection for their Web pages.

Below is the dialog box for setting Password Protection parameters:



The following are the settings for the Password protection applet:

There are two types of password protected Web pages that you can create.

- 1) You want one password to protect your Web page so that a user must know that password to access the page.
- 2) You want to set up several pages, each with their own passwords. The name of the page must be the same as the entered password. For example, if a user types in "bob" as the password, then the bob.html page is opened. If a user types "tim" as the password, then the tim.html page is opened. This is good for having many user-specific pages that you only want your users to know about.

**IMPORTANT!** The amount of security of this password protection Javascript is not high. It's enough so the average Web user wouldn't be able to figure out how to get past the password.

## If you choose type 1:

You must fill in the following parameters:

**Password:** This is the single password that all users will be using to access the specified Web page.

**URL of Web page to display when correct password is given:** This is the Web page that is password protected. It will be displayed when any user correctly types the password in the input box when viewing your Web page.

## If you choose type 2:

You must fill in the following parameters:

**URL of Web page to display when user types in password:** This is the Web page URL that is password protected. The user-specified Web page will be displayed when any user types the user-specific password in the input box when viewing your Web page. Enter only the URL and directory name into this text box. Do not enter the Web page name.

For example, you have two friends, Bob and Tim. You have two Web pages (one for each of them) called bob.html and tim.html.

The entire URL of these Web pages is  
<http://www.mcwebsoftware.com/friends/bob.html>  
<http://www.mcwebsoftware.com/friends/tim.html>

In this Java Perk input box you should enter the following:

<http://www.mcwebsoftware.com/friends>

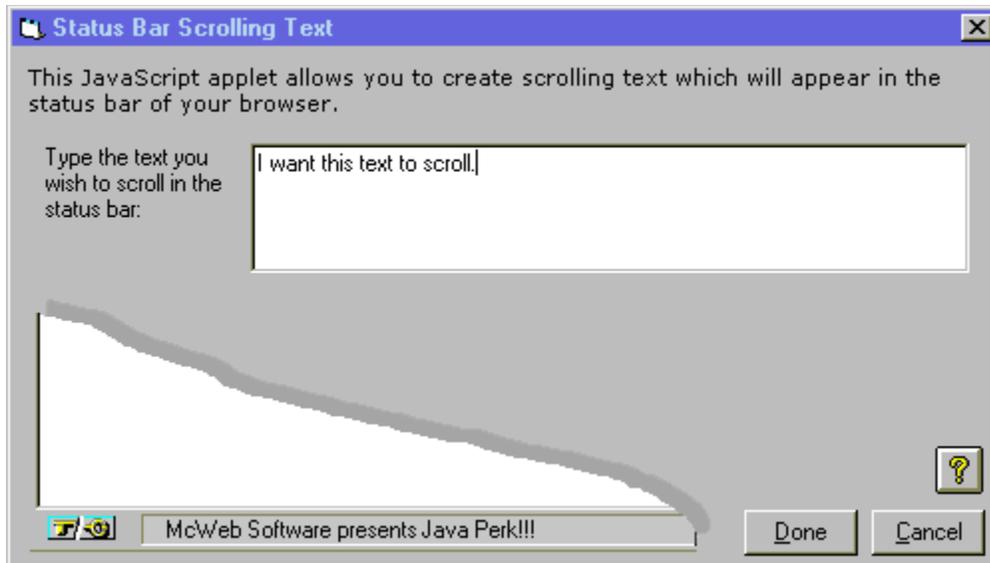
Your password entry Web page is the page that you create with Java Perk. When either friend accesses this page, they will be presented with a pop-up box that asks them for the password. When Bob types the password "bob", the bob.html page will be displayed to him. Likewise for Tim.



## Scrolling Status bar text

This Java applet causes the specified text scroll across the bottom status bar of a Web browser.

Below is the dialog box for setting Scrolling status bar parameters:



The following are the settings for the Scrolling status bar text applet:

**Text you wish to scroll:** Enter the text you wish to scroll across the bottom status bar of the browser's screen.

- When you are done click the OK button on the Scrolling Status bar applet dialog box and the HTML code for this Javascript applet will be displayed in the code preview box for cutting and pasting into an HTML editor.

The code generated by this particular applet is a full HTML document, so you should paste it into a blank document and insert your other Web page contents between the <BODY></BODY> tags of the text.

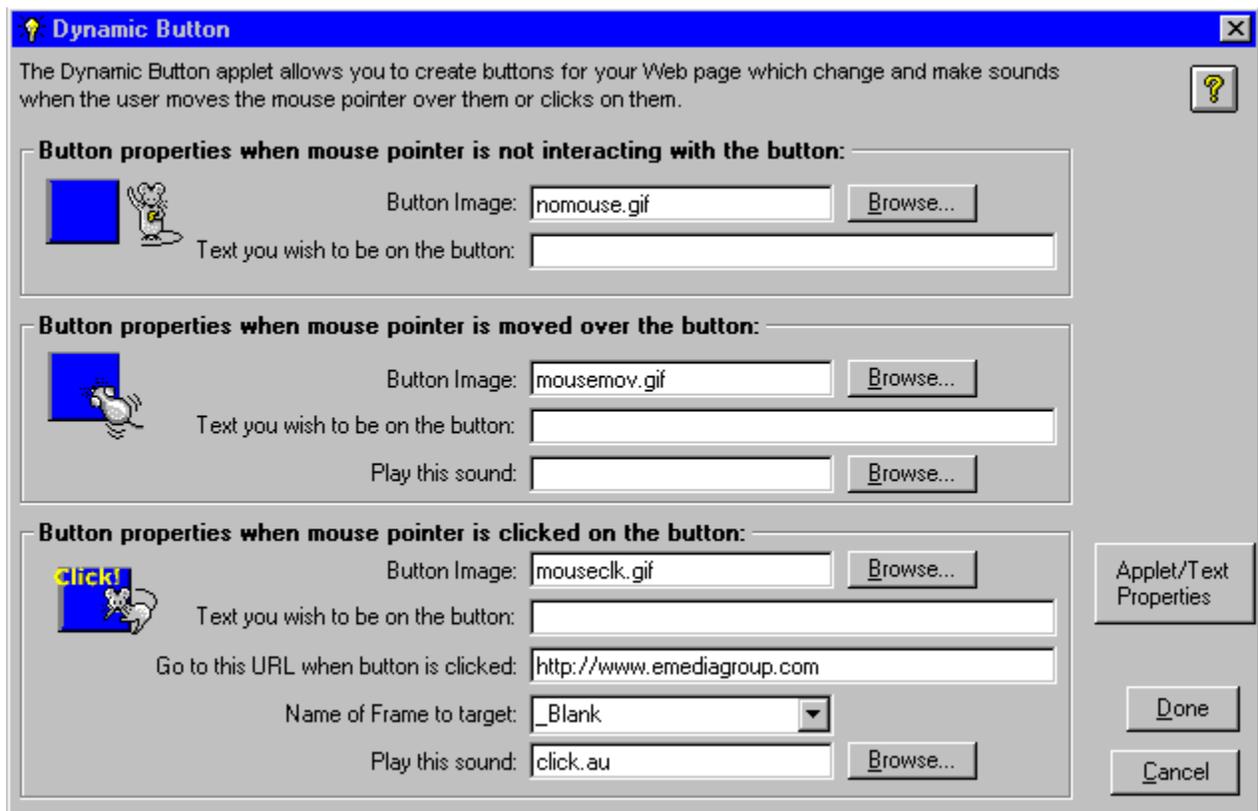
## Dynamic Button applet

The Dynamic Button applet creates buttons which change (and make a sound) when the user moves the mouse pointer over the button and clicks on the button. You can use graphic images for the buttons or create plain buttons with text.

It requires the following applets to function properly:  
**ButtonPLUS3.class**

Making animated dynamic buttons is easy with Java Perk. The process consists of 3 main steps:

Below is the dialog box for setting Dynamic Button applet parameters:



The screenshot shows a dialog box titled "Dynamic Button" with a blue title bar. The main area contains three sections for configuring button properties:

- Button properties when mouse pointer is not interacting with the button:** Includes a "Button Image" field with "nomouse.gif" and a "Browse..." button, and a "Text you wish to be on the button:" text input field.
- Button properties when mouse pointer is moved over the button:** Includes a "Button Image" field with "mousemov.gif" and a "Browse..." button, a "Text you wish to be on the button:" text input field, and a "Play this sound:" field with a "Browse..." button.
- Button properties when mouse pointer is clicked on the button:** Includes a "Button Image" field with "mouseclk.gif" and a "Browse..." button, a "Text you wish to be on the button:" text input field, a "Go to this URL when button is clicked:" field with "http://www.emediagroup.com", a "Name of Frame to target:" dropdown menu set to "\_Blank", and a "Play this sound:" field with "click.au" and a "Browse..." button.

On the right side of the dialog, there is a "Applet/Text Properties" button, a "Done" button, and a "Cancel" button. A help icon (?) is located in the top right corner of the dialog area.

1. Set the **Button properties when mouse pointer is not interacting with button** settings. If you wish to use an image as your button, then input the filename of the image in the input box. You can also add text to the button by typing it in the 'text you

wish to be on the button' input box.

If you wish to create a plain button (no image) with text you can simply type it in the 'text you wish to be on the button' input box.

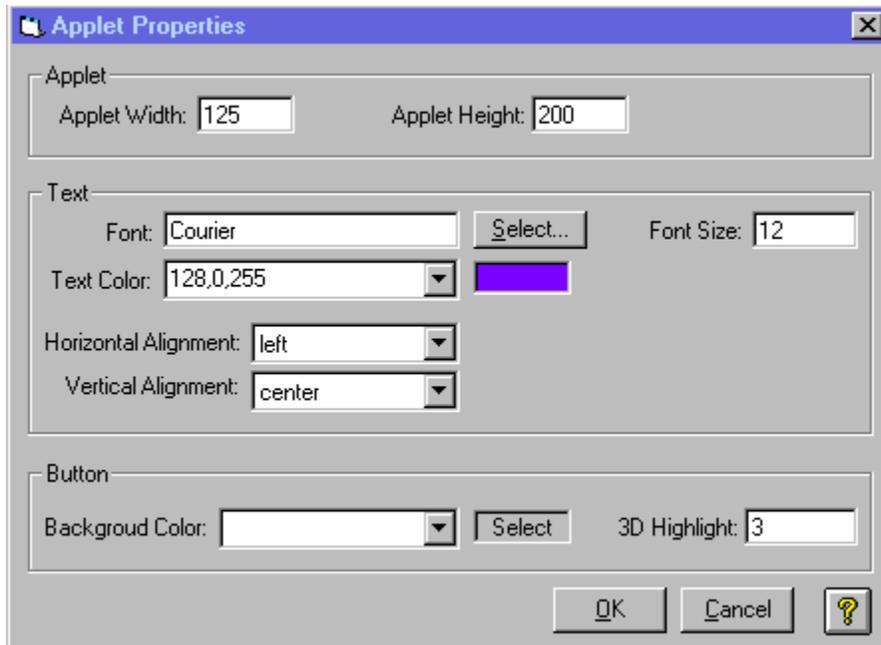
2. Set the **Button properties when mouse pointer is moved over the button** settings. If you are using an image as your button, then input the filename of the image in the input box that you wish to replace the original image when the mouse is moved over the button. You can also add text to the button by typing it in the 'text you wish to be on the button' input box. You can also add a sound file to play when the mouse moves over the button. Enter the sound filename in the 'Play this sound' input box. If you are creating a plain button with text, enter the text (in the input box marked 'Text you wish to be on the button') you wish to appear when the mouse moves over the button. If you wish to play a sound when the mouse moves over the button, then enter the sound filename in the 'Play this sound' input box.

3. Set the **Button properties when mouse pointer is clicked on the button** settings. If you are using an image as your button, then input the filename of the image in the input box that you wish to replace the original image when the mouse is clicked on the button. You can also add text to the button by typing it in the 'text you wish to be on the button' input box. You can also add a sound file to play when the mouse clicks on the button. Enter the sound filename in the 'Play this sound' input box.

If you are creating a plain button with text, enter the text (in the input box marked 'Text you wish to be on the button') you wish to appear when the mouse clicks on the button. If you wish to play a sound when the mouse clicks on the button, then enter the sound filename in the 'Play this sound' input box.

In the 'Go to this URL when button is clicked' input box, enter the Web URL you want the user to go to when he/she clicks on the button.

**NOW** click the **Applet/text properties** button on the Dynamic button dialog box. This displays the following dialog box to enter settings for the Dynamic Button applet:



The following are the settings for the Dynamic Button applet:

**Applet Width(pixels):** This is the width of the applet (specified in pixels). The width should be large enough to show the button image. This is a required setting.

**Applet Height(pixels):** This is the height of the applet (specified in pixels). The height should be large enough to show the button image. This is a required setting.

**Font:** This is the font that the button text will be displayed in.

**FontSize:** This is the size (in points) of the button text.

**Text color:** This is the color of the button text specified in RGB values.

**Horizontal Alignment:** This dictates how the button text will be aligned horizontally on the button. The different settings are LEFT, CENTER, RIGHT.

**Vertical Alignment:** This dictates how the button text will be aligned vertically on the button. The different settings are TOP, CENTER, BOTTOM.

**Background Color:** This sets the background color of the button. Settings are specified in RGB values.

**3D Highlight:** This controls the 3D 'look' of the button. A higher number will result in a more protruding button.

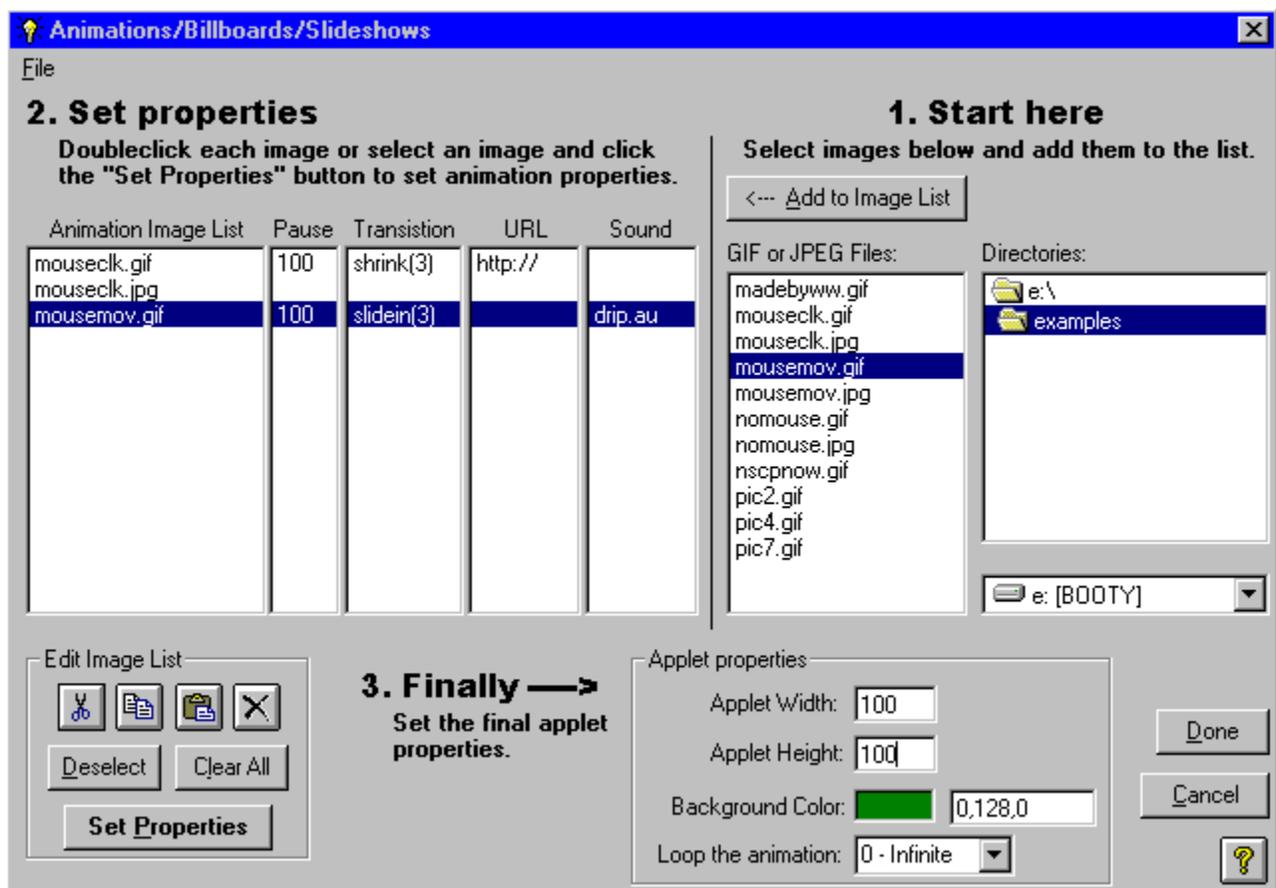
Special thanks to Eric Harshbarger and the Catalyst New Media Marketing lab at Sun Microsystems, Inc. for letting McWeb Software include their Java applets in Java Perk!!!!

# Animator Applet

It requires the following applets to function properly:  
**AnimatePLUS25.class**

Making animations is easy with Java Perk. The process consists of 3 main steps:

1. Using the drive/directory/file list boxes on the right side of the screen, add image files to the Animation image list(left side of screen).
2. Set properties for each image such as pauses, sounds, transitions, URLs to link to, how many times to loop the animation, and applet size
3. Save the animation and view it in your favorite browser (See troubleshooting if you are using Internet Explorer 3.02 or above)!



## Step 1:

Using the drive, directory and file list boxes on the right side of the screen, find the directory where your image files reside. Once the image files are displayed in the file list box, you can insert them into the Animation Image list box on the left side of the screen. You can do this by double-clicking on each image file or by selecting/highlighting an image file and clicking on the 'Add to Image List' button.

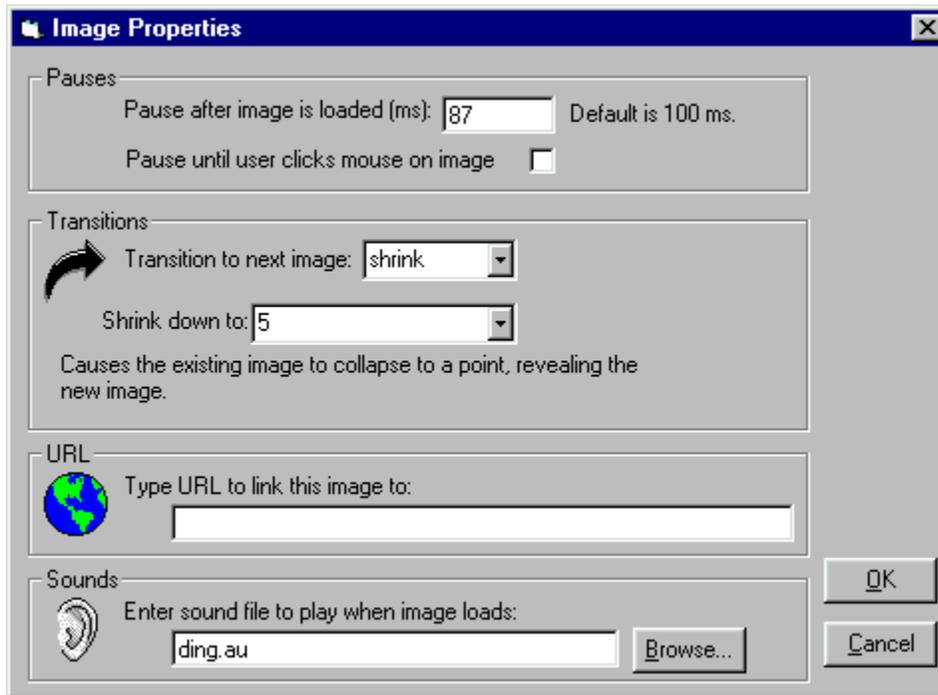
Insert the images in the order in which you want them to display in your animation. If you forget to add an image in your Animation image list, then you can select the position where you want to insert the image, select the desired image from the file list box and double-click it or click the 'Add to Image List' button to insert it into this selected location. You can also use the Copy, Cut, Paste, Delete, Deselect, and Clear All buttons to edit your Animation Image list.

A maximum of 30 images may be used per animation.

## Step 2:

When you have completed inserting the images into the Animation Image list, you can set properties for each image like the following:

- **Pauses** - controls how long to wait (in milliseconds) after the image loads before loading the next image in the sequence (a value of -1 causes the current frame to remain visible until the user clicks on it and a value of -2 causes the current frame to remain visible until the user moves his/her mouse over it,
- **Sounds** - controls what sound files are associated with each frame of the animation. The sound files must be AU files (not WAV files) with the (.au) extension. The associated sounds play as an image loads,
- **Transitions** - controls how the images are loaded into the animation and how they change from one to the next (e.g., an image may slide in from the left, fade away, or scroll down to uncover the next image),



## Descriptions of Transitions

### **overwrite**

This is the default transition which simply writes the new image right overtop the old file; no apparent 'transition' really.

### **slidein(#)**

slidein(#) causes the new frame to enter in slowly from a particular direction and cover up the previous image. The direction of entry is determined by the number in parentheses where '1' is the northwest corner, '2' is the northern edge corner, and progressing clockwise until '8' refers to the western edge. Instead of 1-8, an 'r' may be entered... the applet will then randomly choose a direction each time.

### **slideout(#)**

Similar to slidein(#), slideout(#) causes the old frame to exit the display... uncovering the new image below. The direction of exit is determined by the number in parentheses (numbers correspond to the directions given in slidein(#)). Again, and 'r' may be used in place of a numeric value.

**scroll(#)**

scroll(#) is similar to slidein(#) except that the new frame appears to 'push' the old frame out instead of overlapping it. Note that only directions 2, 4, 6, and 8 are valid for this transition (corresponding to north, east, south, and west respectively).

**expand(#)**

expand(#) causes the new frame to start at a point and slowly 'grow' so that it covers the old frame. The point of origin for the expanding frame is determined by the number in parentheses ('1' is the northwest corner, '2' is the center of the northern edge, and progressing clockwise until '8' refers to the western edge). expand(9) puts the point of origin in the center of the display; expand(r) may be used to let the applet pick a point of origin randomly.

**shrink(#)**

Similar to expand(#) except that the old image collapses to a point, revealing the new image.

**squish(#)**

The new image 'squishes' out the old frame. Number values may only be '2' (north) or '6' (south). East and West squishes do not currently work right because of some image rendering loopiness in Java.

**shutter**

This causes the border of the frame to close over the old image completely then open up, revealing the next frame. The shutter will be the same color as the BORDERCOLORS value for the frame.

**random**

This causes the applet to randomly pick a transition

- **URLs to link to** - this controls which images are associated with Web URLs (addresses) so that the user can click on the image and will be linked to the associated Web page URL,
- **Cycles** - this controls how many times to loop the animation (0 is used to loop the animation indefinitely),
- **Applet Width and Height** - this sets the width and height of the applet in pixels. In general, the size of the applet will be the size of the images contained within the animation, These values are *required*.

and,

- **Background Color** - Click on the 'select' button to select a background color from the palette. The color is specified as the RGB values in this form R,G,B where each color is represented by a number ranging from 0 to 255. For example, gray would be represented by 192,192,192.

### Step 3:

When you have completed inserting all images and setting their properties, you are ready to save your animation and test it!! Click the 'Done' button and a message box will pop up asking if you wish to save your animation.

If you click YES a dialog box will pop up asking you for a filename to save your animation as. It's convenient to be able to save your animation because you may want to edit it later on. Type in a filename with the Java Perk Animation extension (\*.jpa) and click OK. The animation you just created will be saved to your hard drive for safe keeping.

If you select NO to saving the animation then the animation will not be saved to your hard drive, but the HTML animation applet code is still available for you to use.

The HTML applet code Previewer will then appear containing the HTML code necessary for your animation. Select all the code in the Previewer text box copy it to the Windows' clipboard by pressing the copy button. You can then go into your HTML editor and paste the applet code in an appropriate place in an existing HTML document.

Note that the HTML code for the applet which was copied to the Windows' clipboard and the saved animation file are **not the same thing**. The animation file contains all the information that you entered into Java Perk regarding your animation so that you can later reload it for editing or to reproduce the applet HTML code from it. To reproduce the HTML code, just open an animation that you have built and press the 'Save Animation' button again. This will place another copy of the applet HTML code on the Windows' clipboard and then prompt you to save the animation file. You can skip saving the animation file (by pressing 'Cancel' button) if you haven't made any changes to it.

If you want to start a new animation select the 'New' pull-down menu item.

- After you save the animation, open an HTML document in your HTML editor and PASTE the Java Perk Animation HTML code into the document.
- Save the HTML document.
- Copy the AnimatePLUS25.class file into the same directory as the HTML document you just saved.
- Open the HTML document in your favorite browser and watch the animation go!!!

Check Troubleshooting if your animation doesn't work.

Special thanks to Eric Harshbarger and the Catalyst New Media Marketing lab at Sun Microsystems, Inc. for letting McWeb Software include their Java applets in Java Perk!!!!

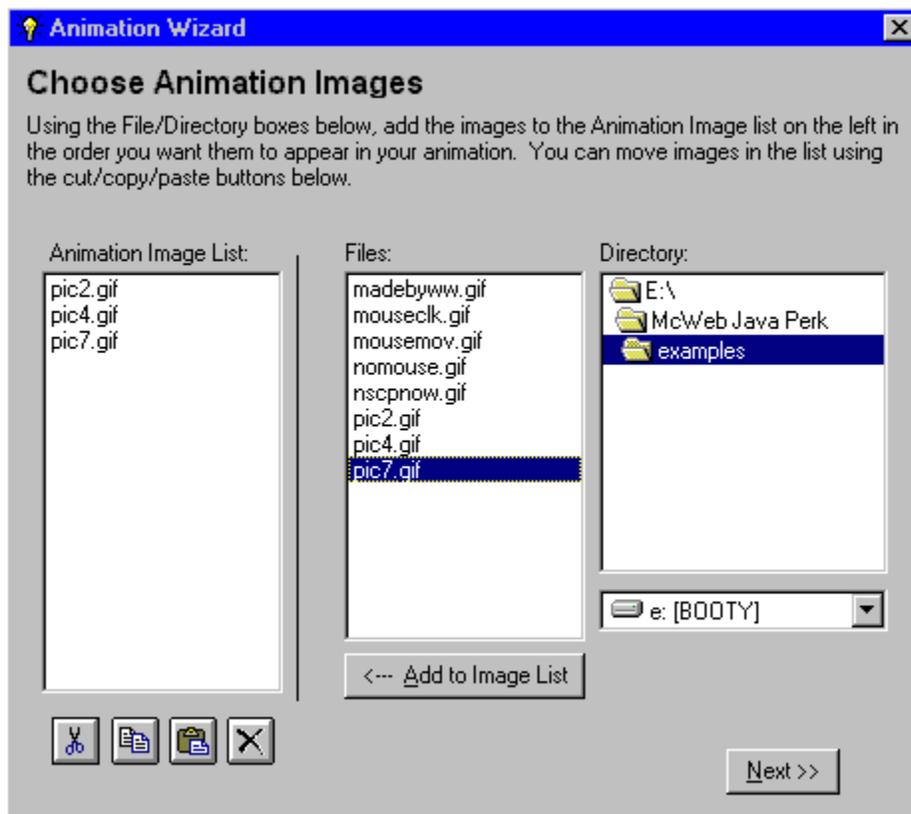
## Animation Wizard

It requires the following applets to function properly:  
**AnimatePLUS25.class**

Making animations is easy with the Animation wizard.

### Step 1: Choosing the Animation Images

Using the file and directory lists on the right side of the screen, select the images (one at a time) for your animation and click on the 'Add to Image List' button (or doubleclick the file) to add it to the Animation image list. It's easiest to insert the images in the order you wish them to be animated in, but you can change the image order by using the cut/copy/paste buttons below the animation image list.



After inserting all the images, move on to the next step.

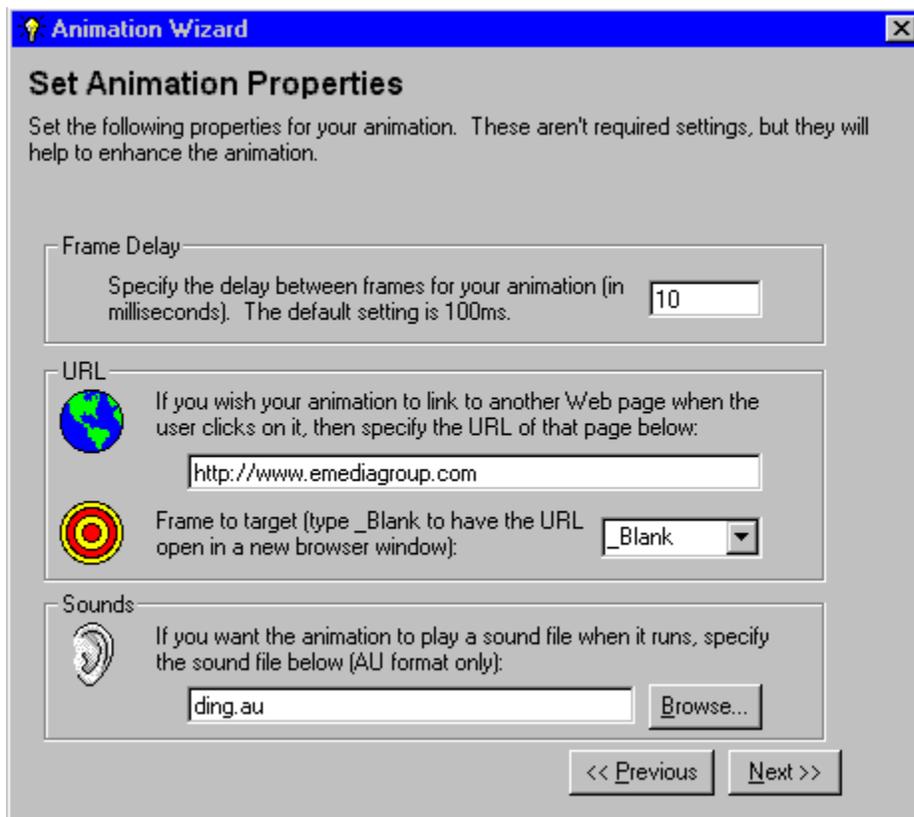
### Step 2: Setting Animation Properties

The following properties can be set for your animation:

**Frame delay:** - the amount of time between frames specified in milliseconds

**URL:** You can specify the URL to associate with your animation. When a user clicks on your animation the specified URL will load. The Target frame for the URL is also specified here.

**Sounds:** Specify a sound to play when your animation begins. If your animation loops several times, then it will play at the beginning of each loop. Only .au format is supported. Use an audio editor such as CoolEdit to convert your WAV files to AU files.



The screenshot shows a window titled "Animation Wizard" with a close button in the top right corner. The main heading is "Set Animation Properties". Below this, a subtitle reads: "Set the following properties for your animation. These aren't required settings, but they will help to enhance the animation." The dialog is divided into three sections:

- Frame Delay:** A text box with the label "Frame Delay" and a description: "Specify the delay between frames for your animation (in milliseconds). The default setting is 100ms." The text box contains the value "10".
- URL:** A section with a globe icon and the text: "If you wish your animation to link to another Web page when the user clicks on it, then specify the URL of that page below:". Below this is a text box containing "http://www.emediagroup.com". Underneath is a "Frame to target" section with a target icon and the text: "Frame to target (type \_Blank to have the URL open in a new browser window):". A dropdown menu is set to "\_Blank".
- Sounds:** A section with an ear icon and the text: "If you want the animation to play a sound file when it runs, specify the sound file below (AU format only):". Below this is a text box containing "ding.au" and a "Browse..." button.

At the bottom of the dialog are two buttons: "<< Previous" and "Next >>".

After setting these properties, move on to the next step.

### Step 3: Setting Applet Properties

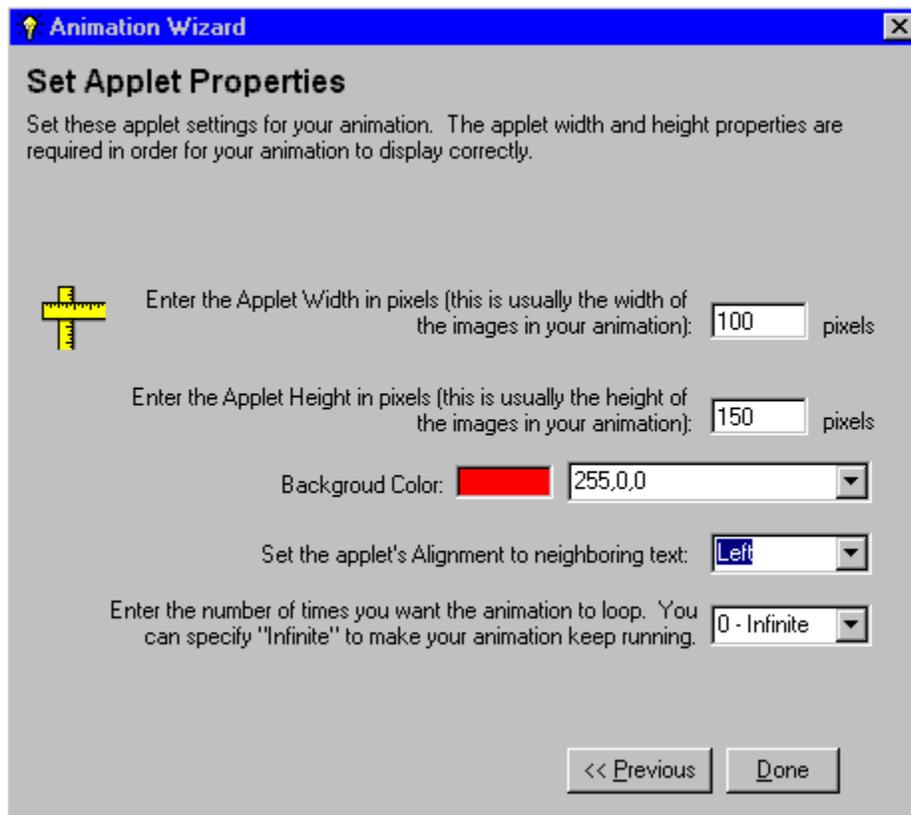
The following properties are set using this step:

**Applet height and width** specified in pixels. Usually the applet height and width are the same size as the height and width of the animation images.

**Background color:** Click on the 'select' button to select a background color from the palette. The color is specified as the RGB values in this form R,G,B where each color is represented by a number ranging from 0 to 255. For example, gray would be represented by 192,192,192.

**Alignment:** The alignment to neighboring text is similar to the alignment used by Web images in Web pages. The same options apply to applets: top, bottom, middle, baseline, texttop, absbottom, left and right.

**Loops:** Set the number of times you want your animation to loop (iterate). Setting this value to 0 will keep the animation running forever.



The screenshot shows a dialog box titled "Animation Wizard" with a sub-title "Set Applet Properties". The main text reads: "Set these applet settings for your animation. The applet width and height properties are required in order for your animation to display correctly." Below this, there are several input fields: "Enter the Applet Width in pixels (this is usually the width of the images in your animation):" with a text box containing "100" and "pixels" to its right; "Enter the Applet Height in pixels (this is usually the height of the images in your animation):" with a text box containing "150" and "pixels" to its right; "Background Color:" with a red color swatch and a dropdown menu showing "255,0,0"; "Set the applet's Alignment to neighboring text:" with a dropdown menu showing "Left"; and "Enter the number of times you want the animation to loop. You can specify 'Infinite' to make your animation keep running." with a dropdown menu showing "0 - Infinite". At the bottom, there are two buttons: "<< Previous" and "Done".

After setting these properties, you can click the 'Done' button and you will be presented with the HTML code that controls the applet. Copy and paste this code into an HTML document that you are creating with your HTML editor.

Put this HTML document, the AnimatorPLUS25.class applet, all the images and sounds into the same directory and test it with your favorite browser. Remember that Internet Explorer 3.02 cannot access files from the local hard drive so you must upload your animation files to your Web server in order to preview it. Netscape Navigator allows you to preview the animation from your local hard disk.

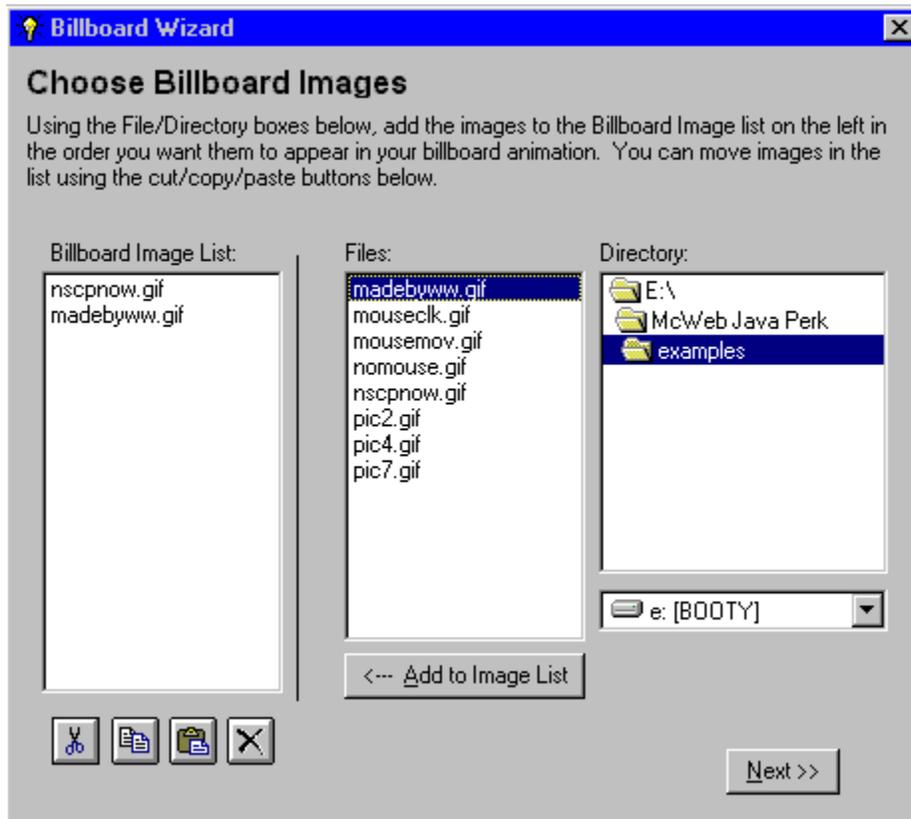
## Billboard Wizard

It requires the following applets to function properly:  
**AnimatePLUS25.class**

Making scrolling Billboards is easy with the Billboard wizard.

### Step 1: Choosing the Billboard Images

Using the file and directory lists on the right side of the screen, select the images (one at a time) for your billboard and click on the 'Add to Image List' button (or doubleclick the file) to add it to the Billboard image list. It's easiest to insert the images in the order you wish them to be animated in, but you can change the image order by using the cut/copy/paste buttons below the Billboard image list.



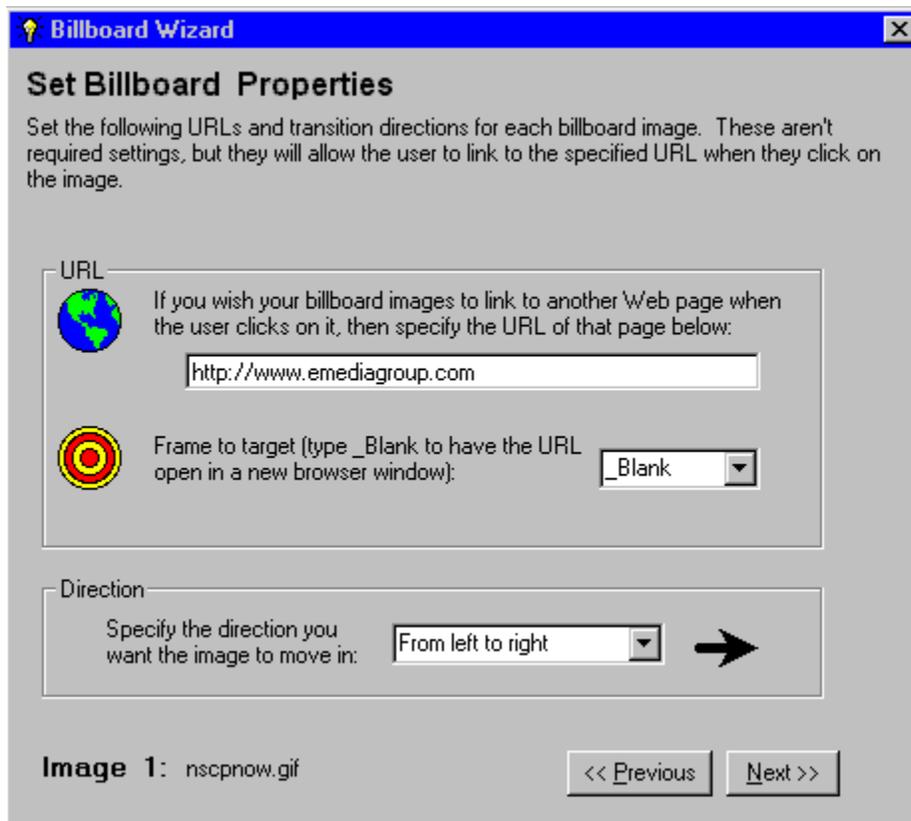
After inserting all the images, move on to the next step.

### Step 2: Setting Billboard Properties

The following properties can be set for each image in your billboard:

**URL:** You can specify the URL to associate with each image of your billboard. When a user clicks on a particular image in your billboard the URL associated with that image will load. The Target frame for the URL is also specified here.

**Direction:** This controls the direction of motion of each image in your billboard. There are 4 different settings. An image can scroll in from the top of the applet to the bottom of the applet, from the bottom to the top, from the left side to the right side, and from the right side to the left side. The default is scrolling from the top to the bottom.



After setting these properties for each image, move on to the next step.

### Step 3: Setting Applet Properties

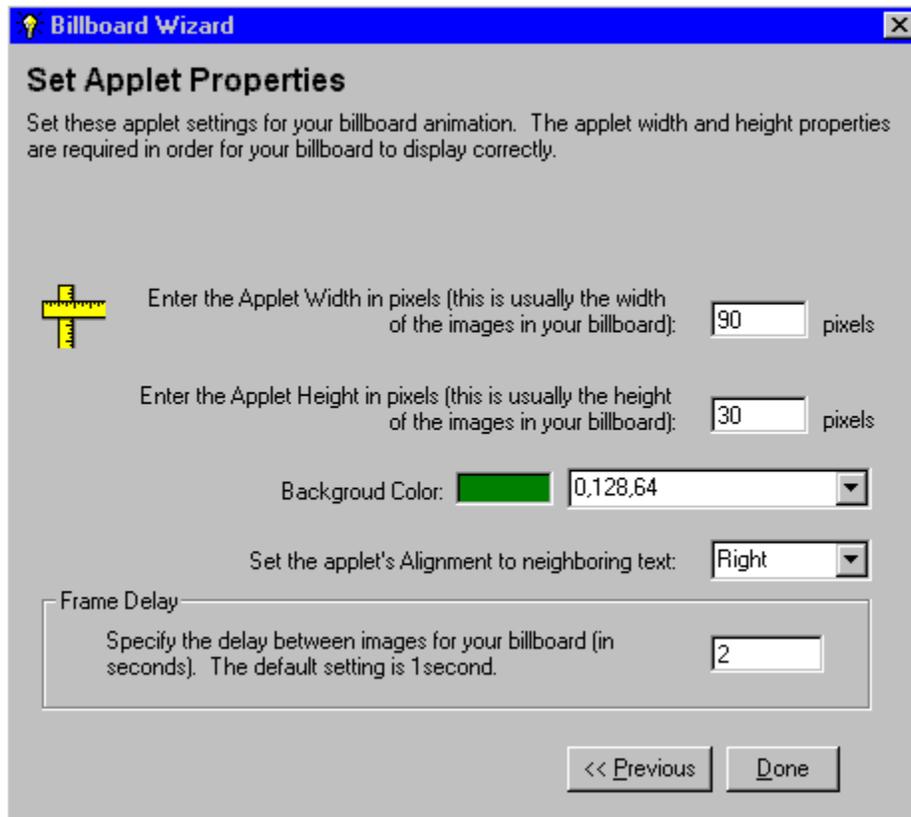
The following properties are set using this step:

**Applet height and width** (specified in pixels). Usually the applet height and width are the same size as the height and width of the images in your billboard.

**Background color:** Click on the 'select' button to select a background color from the palette. The color is specified as the RGB values in this form R,G,B where each color is represented by a number ranging from 0 to 255. For example, gray would be represented by 192,192,192.

**Alignment:** The alignment to neighboring text is similar to the alignment used by Web images in Web pages. The same options apply to applets: top, bottom, middle, baseline, texttop, absbottom, left and right.

**Frame delay** (specified in seconds): This sets the delay between each image frame in your billboard. Setting the delay to be 1 second in length will result in each image being displayed for 1 second until the next image scrolls in.



The screenshot shows a dialog box titled "Billboard Wizard" with a sub-header "Set Applet Properties". Below the sub-header is a descriptive sentence: "Set these applet settings for your billboard animation. The applet width and height properties are required in order for your billboard to display correctly." The dialog contains several input fields: "Enter the Applet Width in pixels (this is usually the width of the images in your billboard):" with a value of 90; "Enter the Applet Height in pixels (this is usually the height of the images in your billboard):" with a value of 30; "Background Color:" with a green color swatch and a text box containing "0,128,64"; "Set the applet's Alignment to neighboring text:" with a dropdown menu set to "Right"; and a "Frame Delay" section with a text box containing "2" and a note: "Specify the delay between images for your billboard (in seconds). The default setting is 1 second." At the bottom are two buttons: "<< Previous" and "Done".

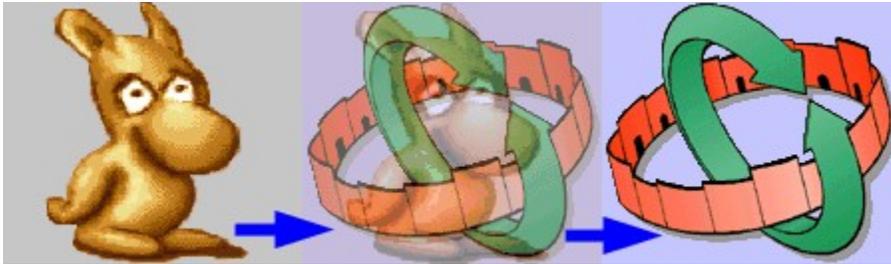
After setting these properties, you can click the 'Done' button and you will be presented with the HTML code that controls the applet. Copy and paste this code into an HTML document that you are creating with your HTML editor.

Put this HTML document, the AnimatorPLUS25.class applet, all the images into the same directory and test it with your favorite browser. Remember that Internet Explorer 3.02 cannot access files from the local hard drive so you must upload your animation files to your Web server in order to preview it. Netscape Navigator allows you to preview the animation from your local hard disk.

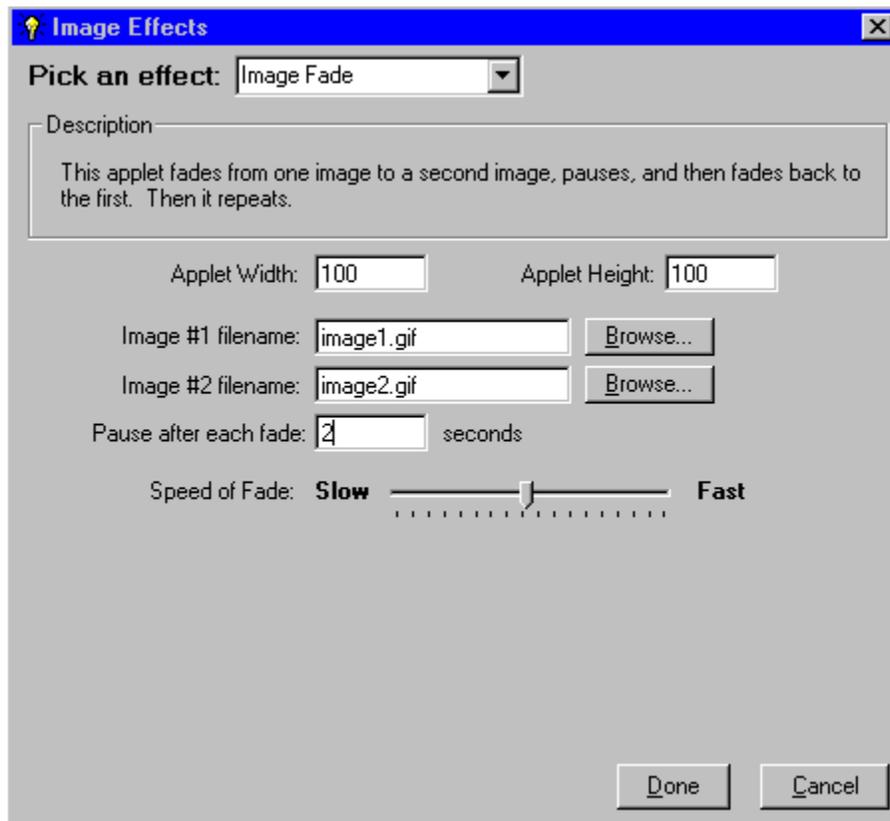
## Image Fade

It requires the following applets to function properly:  
**AnFade.class**

This applet fades from one image into another and then fades back to the original image. Then it repeats.



This is the dialog box that controls your input.



Simply enter the following properties for the applet:

**Applet height and width:** These are the measurements of the height and width of the applet in pixels. It should be the same as the height and width of the images. The images must be the same size.

**Image #1 filename:** This is the filename of the first image to fade from.

**Image #2 filename:** This is the filename of the second image to fade to.

**Pause:** This is the amount of time (in seconds) to wait after each image has faded to the other image. Inputting a 2 second pause would result in the first image fading into the second image, wait 2 seconds, then fade back to the first image, wait 2 seconds, and so on.

**Speed of fade:** This sets how fast the images fade.

After setting these properties, you can click the 'Done' button and you will be presented with the HTML code that controls the applet. Copy and paste this code into an HTML document that you are creating with your HTML editor.

Put this HTML document, the AnFade.class applet, and the two images into the same directory and test it with your favorite browser. Remember that Internet Explorer 3.02 cannot access files from the local hard drive so you must upload your animation files to your Web server in order to preview it. Netscape Navigator allows you to preview the animation from your local hard disk.

Applet programmer: Fabio Ciucci

## Image Cube

It requires the following applets to function properly:

**TmapCube.class**

**TmapCubeb.class**

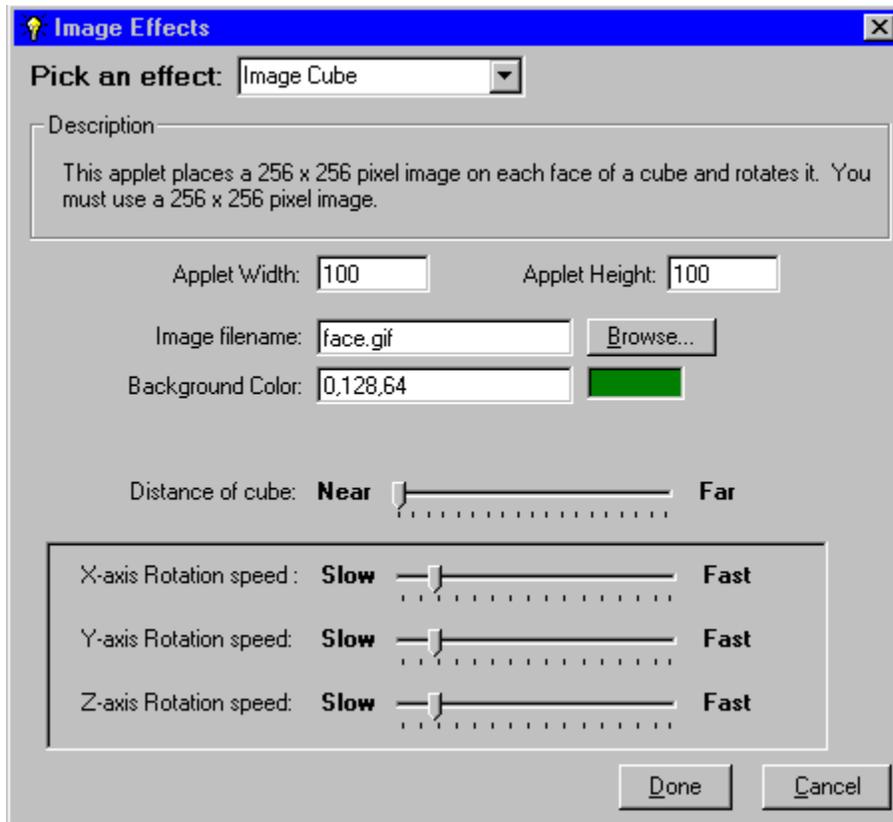
This applet applies an image to each face of a cube and then rotates it. **The image must be 256 x 256 pixels in size.**



Original image



This is the dialog box that controls your input.



Simply enter the following properties for the applet:

**Applet height and width:** These are the measurements of the height and width of the applet in pixels.

**Image filename:** This is the filename of the image that is applied to each face of the cube. **The image must be 256 x 256 pixels in size.**

**Background Color:** Click on the 'select' button to select a background color from the palette. The color is specified as the RGB values in this form R,G,B where each color is represented by a number ranging from 0 to 255. For example, gray would be represented by 192,192,192.

**Distance of cube:** This sets the distance of the cube. 'Near' causes the cube to appear large as if it were close to the screen. 'Far' causes the cube to appear smaller as if it were far away from the screen.

**X-axis Rotation speed:** This sets the cube's speed of rotation about its X-axis.

**Y-axis Rotation speed:** This sets the cube's speed of rotation about its Y-axis.

**Z-axis Rotation speed:** This sets the cube's speed of rotation about its Z-axis.

After setting these properties, you can click the 'Done' button and you will be presented with the HTML code that controls the applet. Copy and paste this code into an HTML

document that you are creating with your HTML editor.

Put this HTML document, the TmapCube.class and TmapCubeb.class applets, and the image into the same directory and test it with your favorite browser. Remember that Internet Explorer 3.02 cannot access files from the local hard drive so you must upload your animation files to your Web server in order to preview it. Netscape Navigator allows you to preview the animation from your local hard disk.

Applet programmer: Fabio Ciucci

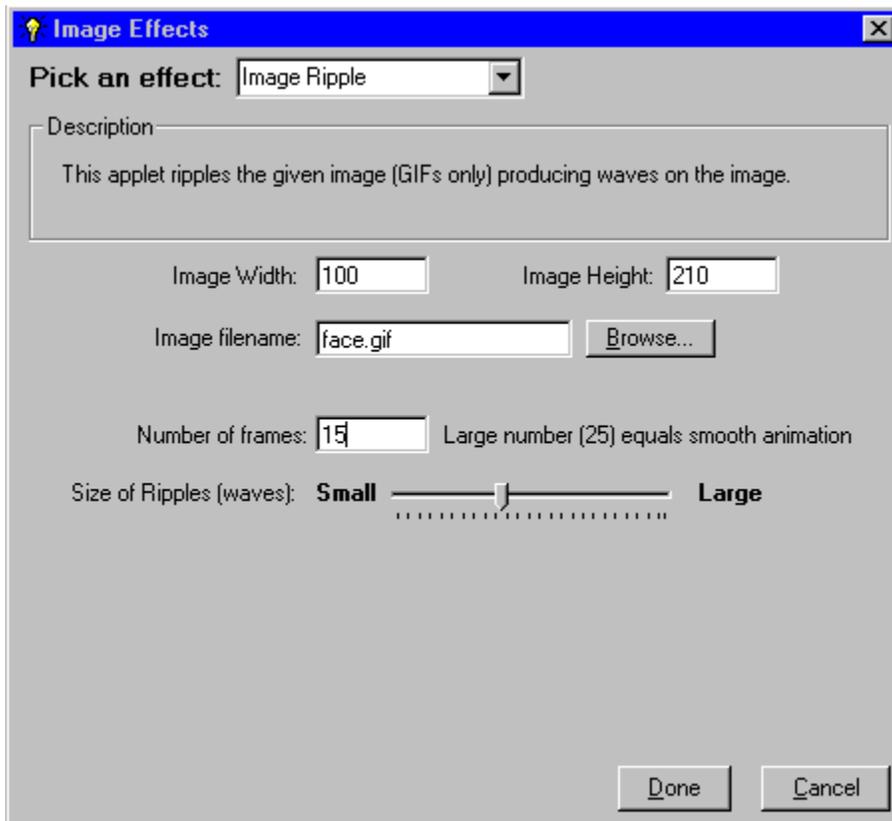
## Image Ripple

It requires the following applets to function properly:  
**ripple.class**

This applet produces a ripple effect on an image to make it look like it's made of water.



This is the dialog box that controls your input.



Simply enter the following properties for the applet:

**Applet height and width:** These are the measurements of the height and width of the applet in pixels. It should be the same as the height and width of the images. The images must be the same size.

**Image filename:** This is the filename of the image you want to ripple (**GIFs only**)!!.

**Number of frames (speed):** This is the number of frames used to animate the ripple effect. A low number of frames will result in a fast, choppy animation. A higher number of frames will result in a smoother and slower animation. A frame number that is too high may result in a crash or you may have to wait a long time for the applet to calculate.

**Size of ripples:** This sets how large the ripples(waves) on the image are made.

After setting these properties, you can click the 'Done' button and you will be presented with the HTML code that controls the applet. Copy and paste this code into an HTML document that you are creating with your HTML editor.

Put this HTML document, the ripple.class applet, and the two images into the same directory and test it with your favorite browser. Remember that Internet Explorer 3.02 cannot access files from the local hard drive so you must upload your animation files to your Web server in order to preview it. Netscape Navigator allows you to preview the animation from your local hard disk.

Applet programmer: David Griffiths

## Lake Reflection

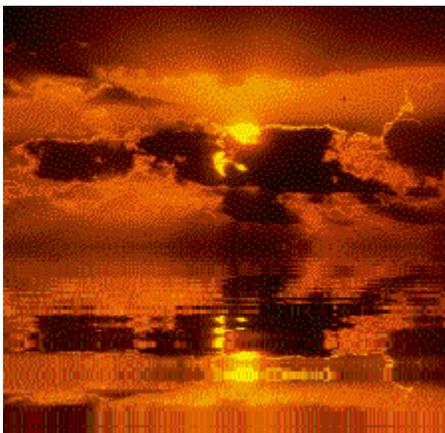
It requires the following applets to function properly:

**Lake.class**

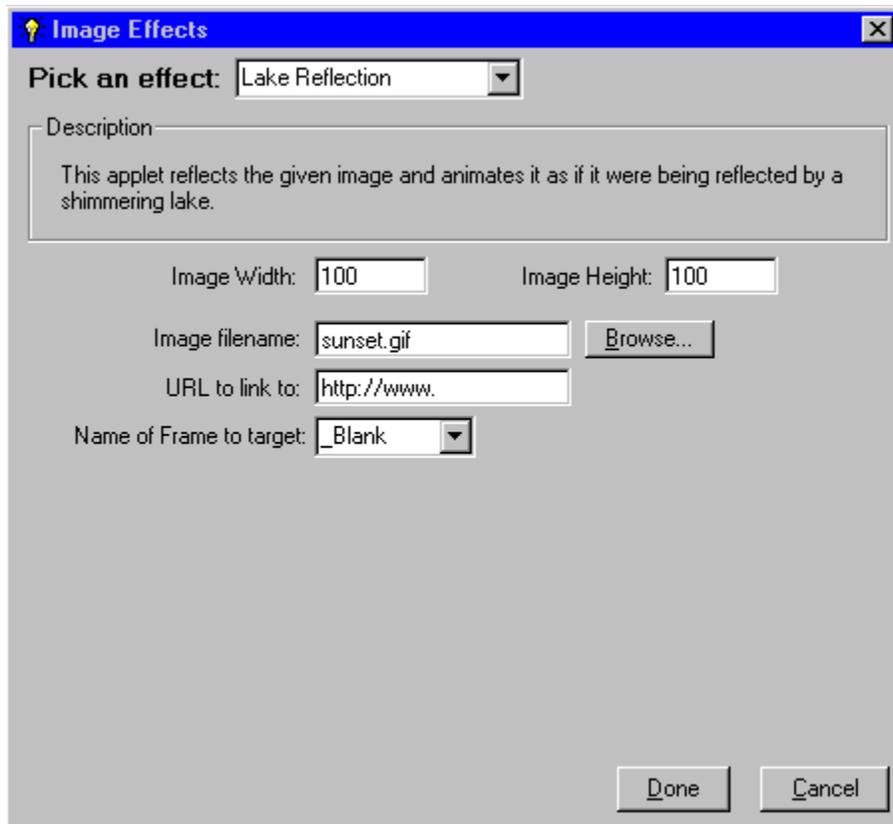
This applet makes your image appear to be reflecting on a pool of water.



Original image



This is the dialog box that controls your input.



Simply enter the following properties for the applet:

**Applet height and width:** These are the measurements of the height and width of the applet in pixels. It should be the same as the height and width of the images. The images must be the same size.

**Image filename:** This is the filename of the image to be reflected on the pool of water.

**URL:** This is URL of the Web page that the applet will link to. When the user clicks on the applet, the specified URL will load.

**Target frame:** This specifies the target frame in which the URL will be opened..

After setting these properties, you can click the 'Done' button and you will be presented with the HTML code that controls the applet. Copy and paste this code into an HTML document that you are creating with your HTML editor.

Put this HTML document, the Lake.class applet, and the two images into the same directory and test it with your favorite browser. Remember that Internet Explorer 3.02 cannot access files from the local hard drive so you must upload your animation files to your Web server in order to preview it. Netscape Navigator allows you to preview the

animation from your local hard disk.

Applet programmer: David Griffiths

# Java Perk Order Form

If ordering with Credit Card, please use the online order form at:

<http://www.mcwebsoftware.com/order.html>

Call in your order at (617) 975-3738

Fax in your order at 1-888-329-0115

If ordering with cash, check or money order:

Please fill out the following registration form, enclose \$25 (U.S.) (plus \$5 shipping & handling if you wish the software sent by U.S. Mail) in form of check or money order, and send to the address at the bottom of this page. The \$25 fee will license one copy for use on one computer at any one time. A full copy of the software will be sent to you along with a license number, help files, sample images and JavaScripts for you to use, etc. Also, subsequent versions and updates will be sent to you free of charge only if you receive the program by email. If you order both Web Weaver and Java Perk then the cost is **only** \$40 total (plus shipping and handling if shipped via snail mail).

PLEASE PRINT CLEARLY!

Name: \_\_\_\_\_ Date: \_\_\_\_\_

Company: \_\_\_\_\_

Address: \_\_\_\_\_

City, State, Zip: \_\_\_\_\_

Country: \_\_\_\_\_ Phone: \_\_\_\_\_

Email address: \_\_\_\_\_ (Your email address will NOT be given to any mailing lists, etc.)

---

__Quantity__	Product	Total
____@ \$25 each	Web Weaver for Windows 3.1/3.11	_____
____@ \$25 each	Web Weaver for Windows 95/NT	_____
____@ \$25 each	Java Perk for Windows 95/NT (only \$15 if ordered w/ Web Weaver)	_____
Add \$5 Shipping & Handling if sent via postal "snail" mail)		_____
TOTAL		_____

Pick shipping method: 3.5" floppy disk \_\_\_\_ Download \_\_\_\_

---

Make checks payable to McWeb Software.

Mail check and registration form to:

McWeb Software  
232 Kent Street  
Brookline, MA 02146  
USA

Email McWeb Software at: [info@mcwebsoftware.com](mailto:info@mcwebsoftware.com) if you have any questions.

IN NO EVENT SHALL LICENSOR OR ITS SUPPLIERS BE LIABLE FOR ANY  
CONSEQUENTIAL OR INCIDENTAL DAMAGES ARISING OUT OF THE USE OF THE  
LICENSED PROGRAM OR RELATED TO THIS LICENSE.

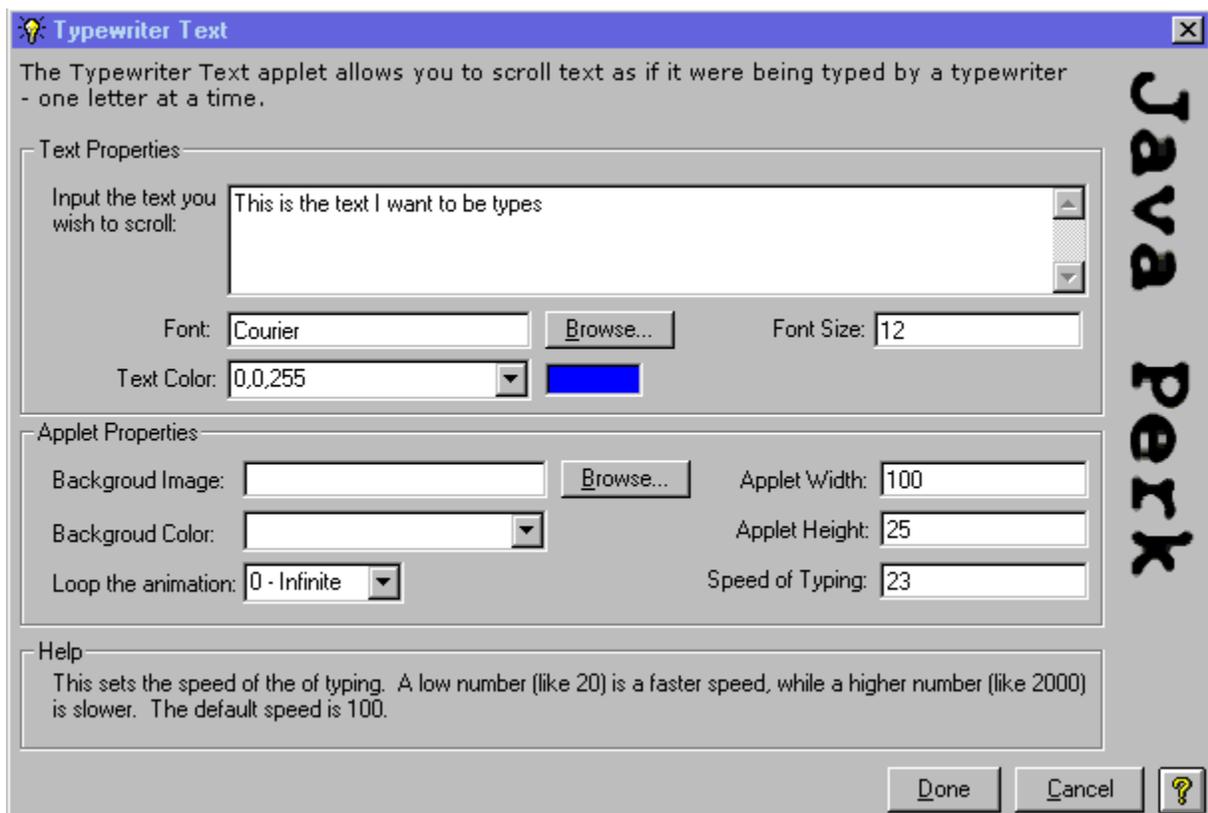
McWeb Software  
Copyright 1995-1997

# Typewriter Applet

It requires the following applets to function properly:  
**Typewriter3.class**

The Typewriter applet allows you to display text on a Web page and animate it as if it were being typed on a typewriter. Many properties can be set to control how the text behaves, like:

- Typing speed,
- Text color,
- Background color,
- Background Image
- Font style and size,
- Cycles, and
- Applet dimensions.



It's a very simple applet to create! Simply type in the text you wish to be typewritten (animated) on your Web page in the large text input box.

Then enter in all the other properties mentioned above. Remember that the number of cycles are the number of times the typewriter animation will loop (a value of zero will equal an infinite loop).

The typing speed defaults to a value of 100. The lower the number, the faster the typing. So, a value of 25 would result in fast typing while a value of 2000 would result in a very slow typing speed.

Be sure to include an appropriate applet width and height which will adequately display the animation. These values should be entered in pixels.

Once you've input all the properties (some are optional) you can save the typewriter applet HTML code by clicking on the 'Done' button. The HTML applet code Previewer will then appear containing the HTML code necessary for your typewriter applet. Select all the code in the Previewer text box copy it to the Windows' clipboard by pressing the copy button. You can then go into your HTML editor and paste the applet code in an appropriate place in an existing HTML document. .

Special thanks to Eric Harshbarger and the Catalyst New Media Marketing lab at Sun Microsystems, Inc. for letting McWeb Software include their Java applets in Java Perk!!!!

# TROUBLESHOOTING

## ***Why doesn't the animation work? I don't see any pictures.***

1) You may be viewing the animation using Internet Explorer 3.02 or 4.0. Microsoft has prevented the loading of files/images from the local hard drive for security reasons. You can download the bug fix (updated Java virtual machine) from Microsoft's site to correct this problem. Go to <http://www.microsoft.com/java> and select the 'Java VM Update' hyperlink which will take you to the update download page. Download the appropriate Build of the Java VM (Build 1518 for Windows 95 as of 6/16/97). This will correct the problems.

Otherwise, if you wish to view the animation locally from your hard drive then you must use Netscape Navigator or another browser (IE prior to version 3.02). You can use Internet Explorer 3.02 if you upload the animation to a Web server and view it from the Web.

2) You may have put the files in a directory which has a long filename with spaces in it (like c:\Java Perk\files). The animator applet doesn't understand the space between the words 'Java' and 'Perk'.

**Solution:** Put your HTML, image and sound files in a directory which doesn't have any spaces in its name (like c:\animations\files).

3) You may have put your HTML file with the applet code in one directory and your image files in a separate directory.

**Solution:** This is OK as long as you specify directory names while creating the applet code in Java Perk.

The easy way to make your animations work is: Be sure that you have the following things in the same directory:

- the HTML document containing the animation applet code (which is created by Java Perk)
- the AnimatorPLUS25.class Java applet
- the images and sound files (unless you have specified different directories for these in Java Perk).

The default is for the HTML document (that has the applet code embedded in it) to be in the same directory as the images and sounds which are used by the animation (no image or sound directories need to be specified in this case). If these files are all in the same directory, then you **should not** specify any directory names during creation of the applet in Java Perk (specifying directory names in this case will result in errors).

4) Putting the images and sound files in different directories than the AnimatePlus25.class file and the HTML file with the applet code can be done, but it involves specification of these directories. For example, if you have the HTML document in a directory such as C:\WEBPAGE, you have the images in a directory C:\WEBPAGE\IMAGES and the sounds in the directory C:\WEBPAGE\SOUNDS, then you must specify these directories in Java Perk. The directories that you specify **should** be specified relative to the location of the HTML document containing the animation applet. You **can** specify the entire directories, but this may be inconvenient if you have to move your Web pages to another site or from your local hard disk to disk space on an Internet Service Provider's server.

As recommended, specify the directories relative to the directory which contains the AnimatePlus25.class and HTML files. Continuing our example, specify the image directory to be IMAGES/ and the sound directory to be SOUNDS/.

### ***Why do my images not look clear in the animation? Why are they pixelated or have strange colors?***

The use of GIF images which have transparent background colors set will show the background colors of the HTML web page in which the applet is embedded. Also, Netscape seems to have trouble with GIF images in these animations. Sometimes they appear pixelated (not perfectly clear).

It is best to use JPEG images in your animations. They are usually smaller in file size and you won't have problems with transparency settings as you do with GIFs.

Also, Netscape seems to have a 256 color palette when it displays Java applets. this may dither or pixelate your images if they have more than 256 colors.

### ***Why doesn't the fading applet work?***

You may have specified a very fast speed. This will result in the fade occurring at a speed which is 'faster than the human eye'.

**Solution:** Slow it down.

If you set the different colors on a computer with a 24 bit (16.7 million) color setting and you view it on a computer which has 8 bit (256 color) setting then you may not see a fade because the two colors may have been too similar. On a 256 color machine these colors will be converted to the nearest color in the 256 color palette, and if these colors were similar then they may have been converted to the same color (resulting in no fading).

**Solution:** Choose colors from the 256 color palette.

