

ScrollerPRO

Version 1.0



Copyright 1998 by One Wolf WebArt (formerly Lone Wolf WebArt). All rights reserved.

<http://www.onewolf.com>

Table of Contents

| | |
|--|----|
| <i>Preface</i> | 1 |
| <i>Chapter 1 - Introduction</i> | 2 |
| Features | 2 |
| <i>Chapter 2 - Parameters</i> | 4 |
| LINE1 - LINE500 | 4 |
| BGRED, BGGREEN, BGBLUE | 5 |
| Color Table | 5 |
| FGRED, FGGREEN, FGBLUE | 6 |
| FONTNAME, FONTSIZE, FONTITALIC, FONTBOLD | 6 |
| FONTNAMEx, FONTSIZEx, FONTITALICx, FONTBOLDx | 6 |
| FALSESPACES | 7 |
| SPACING | 7 |
| STARTDELAY | 8 |
| DELAY | 8 |
| STEP | 8 |
| XPOS | 8 |
| DIRECTION | 8 |
| BACKGROUND | 9 |
| <i>Chapter 3 - Advanced Topics</i> | 10 |
| CONTENT | 10 |
| Style Tags | 10 |
| <i>Chapter 4 - FAQ</i> | 11 |

P R E F A C E

In the short time I've been developing with Java, one of the most rewarding experiences has been receiving comments and feedback from people using the applets I've chosen to share with the web community.

One of those applets, tinyScroller, has become an extremely popular applet with both professional web developers and hobbyists. Feedback from the users of tinyScroller has driven its evolution. Hundreds of web sites now utilize tinyScroller or one of the other applets in the tinyApplet series.

My newest applet, scrollerPRO is both a break from, and an advancement of, tinyScroller. Because I chose not to distribute the source code with scrollerPRO, I was able to replace some of the more readable code with smaller, more efficient algorithms while adding functionality.

Enjoy using scrollerPRO. I've had a lot of fun developing it.

C H A P T E R . 1

Introduction

Your newest applet, scrollerPRO, is feature rich, simple to use and compatible with tinyScroller. If you're already using tinyScroller, scrollerPRO will plug right in, no modification necessary.

The following features are new to scrollerPRO (compared with tinyScroller):

- I'm using a more efficient and smoother technique to achieve the animation. Even though tinyScroller double-buffered the image, it used a less practical method of redrawing the image.
- You can, optionally load your content from a separate file on your web site or anywhere in the world (passed as an URL). Not only does this make maintenance easier, but now you can deploy scrollerPRO on many sites. It will pull its content from a single location -- no more redundant maintenance for enterprise or franchise web sites.
- Font, font size, and attributes are all customizable for EACH line. Now you can mix and match styles. I left the FONTNAME and FONTSIZE tags in place for backward compatibility -- they now determine the *default* font style for all content in the applet.
- While you may still hard-code the spacing between lines, if you choose not to, the lines will be spaced proportional for the font type and size chosen for the line(s) being displayed.
- I've increase the number of available lines to 500. While this seems unwieldy, now that you can load your content from a separate file, you don't need to complicate your web pages by maintaining the content in the HTML.

- I've added FALSESPACES. Because parameters passed from HTML are often stripped of leading spaces by browsers (both Netscape and Internet Explorer), I've added an (optional) feature I call false spaces. If you choose, leading periods will be replaced by the applet with spaces. This means you can now better control the layout of your content.
- I've added the STEP parameter. STEP allows you to define the change in position each time the content moves. Increasing the STEP value causes the lines to move in a less smooth (blockier) fashion.
- The content and background graphic are now tiled. This serves two purposes. First, when the text block finishes scrolling into the viewable space, scrollerPRO immediately begins scrolling the lines again, making your content appear seamless. Second, tiling the background allows you to pass smaller (more efficient) graphic files.
- Finally, I've added *real* documentation and examples.

The following scrollerPRO features were inherited from tinyScroller:

- The foreground (font) and background colors are adjustable by passing red, green and blue components (in base-10, decimal).
- The font type and size are definable, but the applet will default to the client system's standard, if the passed font style is unavailable.
- The web designer may control the line spacing.
- The web designer may control the scroll speed.
- The content may be scrolled in either direction (top to bottom or bottom to top).
- The web designer may control of the horizontal position (XPOS).
- A background graphic may be used (and it scrolls).

C H A P T E R . 2

Parameters

If you're not sure how to pass your applet parameters or are not comfortable working with HTML, see the FAQ at the end of this documentation. There is also plenty of good HTML documentation available on the web.

LINE1 - LINE100 (Some Required) - LINE1 through LINE500 make up the content to be displayed. If you're not using all 500 lines you don't have to define them all but the applet will stop reading lines at the first NULL line it encounters.

If you want to jump right in and begin using scrollerPRO, add the LINE_x parameters to your applet tags. This is the only required parameter for scroller PRO.

Example:

```
<HTML>
...
<BODY ... >
<APPLET CODE="scrollerPRO.class" HEIGHT=50 WIDTH=100>
<PARAM NAME="LINE1" VALUE="Hello World">
<PARAM NAME="LINE2" VALUE="This is my ">
<PARAM NAME="LINE3" VALUE="use of ">
<PARAM NAME="LINE4" VALUE=" ">
<PARAM NAME="LINE5" VALUE="scrollerPRO">

Content placed here will appear in browsers
without a Java Virtual Machine

</APPLET>
...
</BODY>
</HTML>
```

You may pass a single space (or period, see the FALSESPACES parameter) on lines that won't have content.

Content will be scrolled continuously, meaning that when the last line has scrolled into the viewable area, the first line will immediately begin to appear.



The content may also be read from a file located on this web server or anywhere on the web. Read more about the CONTENT parameter for this feature.

BGRED, BGGREEN, BGBLUE (Optional) - You may set the background color of the applet by passing the one byte (0-255) RGB (Red, Green and Blue) component values. The background color defaults to white.

Example (results in a black background):

```
<PARAM NAME="BGRED" VALUE="0">
<PARAM NAME="BGGREEN" VALUE="0">
<PARAM NAME="BGBLUE" VALUE="0">
```

Here are some common (browser-safe) colors:

| BGRED | BGGREEN | BGBLUE | COLOR |
|-------|---------|--------|------------------|
| 51 | 51 | 102 | Dark Blue |
| 0 | 0 | 153 | Blue |
| 153 | 204 | 255 | Light Blue |
| 0 | 102 | 51 | Dark Green |
| 0 | 153 | 51 | Green |
| 204 | 255 | 204 | Light Green |
| 153 | 0 | 51 | Dark Red |
| 204 | 0 | 51 | Red |
| 255 | 0 | 102 | Light Red (pink) |
| 255 | 255 | 255 | White |
| 0 | 0 | 0 | Black |

The background color will not matter if you also pass a background graphic since the background graphic will be displayed over the background (color).

FGRED, FGGREEN, FGBLUE (Optional) - You may set the foreground (font) color for the applet by passing the one byte (0-255) RGB (Red, Green and Blue) component values. The foreground color defaults to white.

Example (results in a red foreground/font):

```
<PARAM NAME="FGRED" VALUE="255">
<PARAM NAME="FGGREEN" VALUE="0">
<PARAM NAME="FGBLUE" VALUE="0">
```

For more information about mixing red, green and blue to set the foreground color, see the color table above or refer to your favorite graphics program.

FONTNAME, FONTSIZE, FONTITALIC and FONTBOLD (Optional) - You may force the default font type, size and style by passing these values. If you pass FONTNAME, you'll need to also pass FONTSIZE. FONTITALIC and FONTBOLD are optional parameters if you're forcing the default font.

Example:

```
<param name="FONTNAME" value="Arial">
<param name="FONTSIZE" value="24">
<param name="FONTBOLD" value="1">
<param name="FONTITALIC" value="1">
```

This will force the default font (for all lines without custom FONTNAME_x, FONTSIZE_x, FONTBOLD_x and/or FONTITALIC_x attributes) to be a 24 point, Arial, Bold and Italic type.

FONTNAME and FONTSIZE may be anything. If the FONTNAME you select does not exist on the client PC, the system font will display. Valid values for FONTBOLD and FONTITALIC are 1 (on) or 0 (off).

FONTNAME_x, FONTSIZE_x, FONTITALIC_x, FONTBOLD_x - By setting these values, you can override the default font for a single line. These values work like the setting for the default font. You'll need to define FONTSIZE_x if you define FONTNAME_x, but FONTBOLD and FONTITALIC are optional values.

Example:

```
<param name="FONTNAME" value="Courier">
<param name="FONTSIZE" value="12">

<param name="LINE1" value="In the news:">
<param name="FONTNAME1" value="Arial">
<param name="FONTSIZE1" value="14">
<param name="BOLD1" value="1">
```

For this example, the default font for all lines will be a 12 point Courier, plain font. However, line one will have a 14 point, Arial bold font. Line two will revert back to the default font.

The same rules apply for FONTNAME_x, FONTSIZE_x, FONTBOLD_x and FONTITALIC_x, as for FONTNAME, FONTSIZE, FONTBOLD and FONTITALIC.

FALSESPACES (Optional) - Because parameters passed from HTML are often stripped of leading spaces (in Netscape and Internet Explorer), I've added a feature I call false spaces. If you choose, leading periods will be replaced in the applet with spaces. FALSESPACES defaults to 1 (on), the only other valid value for FALSESPACES is 0 (off).

Using FALSESPACES means you can better control the indent of your content. Additionally, adding the spaces makes the HTML (parameters) easier to read.

Example:

```
<PARAM NAME="LINE1" VALUE="Hello World">
<PARAM NAME="LINE2" VALUE="...This is my ">
<PARAM NAME="LINE3" VALUE="...use of ">
<PARAM NAME="LINE4" VALUE=".">
```

In this example, lines two and three will be indented three spaces and line four will be blank.

SPACING (Optional) - Spacing will normally default to whatever is appropriate for the fonts being used in the lines around the gap. In other words, the spacing will be proportional. However, you may hard-code the line spacing by passing it as a parameter (in pixels).

STARTDELAY (Optional) - By passing a value (milliseconds) in the STARTDELAY parameter, you can control how long after the applet loads and runs before the first lines start appearing in the viewable space. The applet will be “on hold” for as long as you need it to be.

Using this parameter sometimes cuts down on CPU load while the page is loading, allowing graphics and content to finish loading before the applet begins running.

DELAY (Optional) - Delay controls the time interval between changes in the line positions. Delay defaults to 100 milliseconds. The value you are passing is in milliseconds.

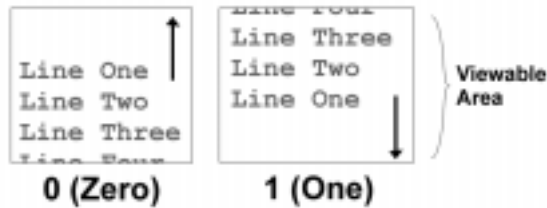
A lower DELAY value translates into a faster scroll. However, setting the DELAY value to an extremely low value (fast scroll), the quality of the scroll (smoothness) becomes dependent upon the load of the client system. Generally 50-150 is a good range.

STEP (Optional) - The step value is the number of pixels the text block moves as it scrolls. STEP defaults to 1 (one) pixel. Overriding this value causes the scroll to be less smooth, but allows the web page designer to dabble with some alternative effects.

XPOS (Optional) - XPOS stands for X position, this controls the horizontal position of the lines within the applet. XPOS defaults to five (pixels). Tailor this value to your background graphic.

DIRECTION (Optional) - DIRECTION allows you to specify whether you want the applet to scroll up (DIRECTION=0) or to scroll down (DIRECTION=1). DIRECTION defaults to zero.

When you consider the setting the DIRECTION to one (scroll down), keep in mind what order the lines will appear in the applet. When the text is scrolling up, LINE1 appears before LINE2 and so on. I’m using the same thinking when scrolling down. LINE1 appears before LINE2, which appears before LINE3 and so on.



BACKGROUND (Optional) - If you wish to use a graphic for the background of the applet, pass the filename here.

Displaying a graphic in scrollerPRO does not have the same impact (flicker) as it did in tinyScroller. I'm also tiling the background graphic in scrollerPRO so you no longer need to pass an extremely large graphic. Consider using graphics one pixel high and just wide enough to fill the applet.

Example:

```
<param name="BACKGROUND" value="bg.gif">
```

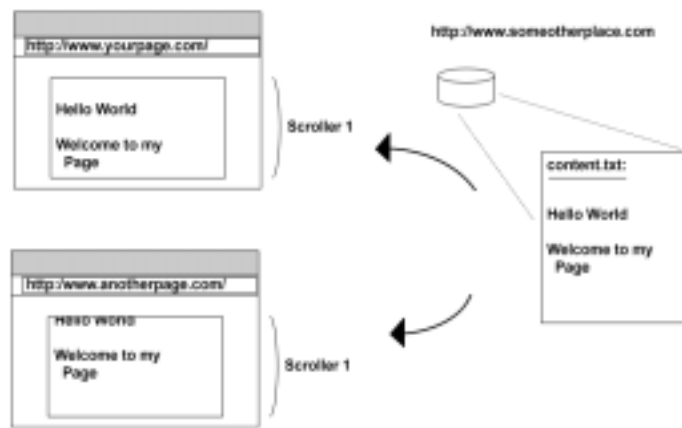
NOTE: Because the background graphic is handled a little differently in scrollerPRO, it does not support fixed (non-scrolling) background graphics like tinyScroller. If you need to implement a scroller with a fixed background graphic, perhaps tinyScroller is better suited to your needs.

C H A P T E R . 3

Advanced Topics

This applet, scrollerPRO, offers many new features. The most requested feature, the ability to load content from an external file, is probably the most valuable to web designers.

CONTENT (Optional) - You may pass the content of the applet (LINE1 - LINE500) as an URL. The CONTENT file must exist on a web server to be read correctly; it only needs to be available on the same network as the client computer (i.e. the internet or intranet).



This allows you to load your content from any location on the web. ScrollerPRO may be running on hundreds of websites around the world, but the actual message only needs to be updated in one place. This drastically reduces the amount of maintenance necessary for corporate intranet sites or franchise sites.

Example:

```
<param name="CONTENT" value="http://someplace.com/content.txt">
```

C H A P T E R . 4

Frequently Asked Questions

This is a copy of the FAQ I keep and maintain on my web site in an effort to answer questions that arise as users try to incorporate the One Wolf WebArt applets into their page.

How do I add an applet to my web page?

There are plenty of ways I could answer this question. The easiest way, I feel, to learn how to incorporate Java applets into your web page is to look at someone else's HTML. You're certainly welcome to view mine.

At a minimum, the following lines need to be added to your HTML:

```
<applet code="class_file_name_here.class"
        height="pixel_heigh" width="pixel_width">

    Your parameters go here

</applet>
```

This assumes that your applet (filename.class) exists in the same directory as your HTML. If your applet is saved somewhere else, you'll need to include the directory (or web address) with your code string. Alternatively, you could include the codebase attribute in your applet tag.

Any HTML that appears between the applet tags would appear in the space of an applet on browsers that don't support Java. The parameters for an applet are documented and come with the ZIP archive in a text file with the same name as the applet.

A parameter tag looks something like this:

```
<param name="parameter_name" value="parameter_value">
```

When I, or another developer, make reference to the parameter (passed to the applet), I'm referring to what you put in the name string. The actual value for the parameter goes in the value string.

I still don't get it....

Okay. Copy and paste my HTML into your web page. In Netscape, click on the View menu, then Page Source. In Internet Explorer, click on the View menu, then Source. Both browsers now support the right (mouse) click in Windows 9x so you could also use the right-click menu to view a page's HTML source.

Once you've pasted the HTML into your source. Change the parameter values. Refer to an applet's documentation for more information about the parameters.

But you're not using the applet the way I want to.

These applets are now being used by hundreds of web developers all over the world. To get an idea for how they may be used, try searching for the applet on Altavista (altavista.digital.com).

Altavista is unique among search engines because it indexes the HTML source. All you need to do is key the applet name in the search string text box; Altavista will return a list of the pages it knows are using the applet.

What is a Magic Number error?

When using FTP to upload an applet to your web server, make sure your FTP application is using the binary mode when transferring the applet. If it's not, figure out how to force it. The Magic Number error is usually caused by a corrupted class file. Most of the time, this corruption is caused by the FTP send method.

I'm getting an Applet not Found error but I know that it's there.

Make sure you spelled the applet (class file) name correctly in your HTML. Also, double check how you capitalized the filename in your HTML and how the class file is capitalized in the source directory.