How to Build a Web-Based Slide Show

By R. Nicholas Gerlich. Ph.D.

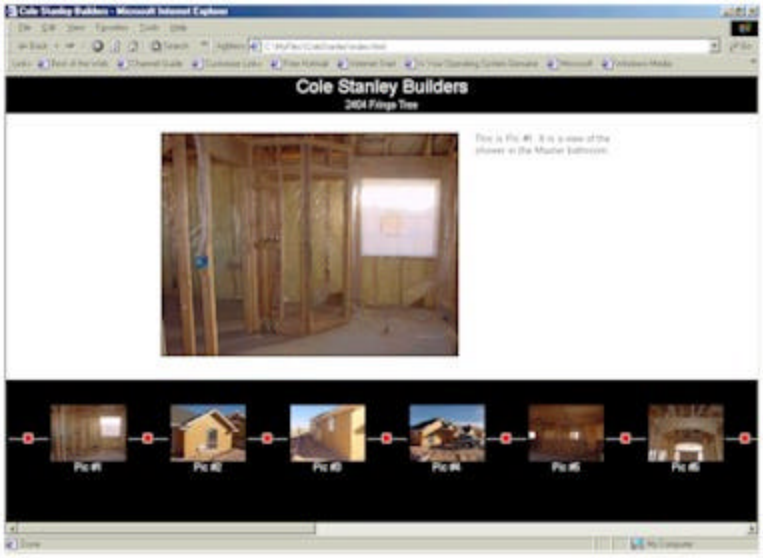
Slide shows are a great way to entertain or inform your web visitors. Unfortunately, pictures by their very nature are “heavy” objects on the web, and unless precautions are taken, can bring the site to its knees.

There are a number of ways that I have built slide shows in the past. These include:

- Automated slide shows, in which the viewer has no control over the pages. Meta tags with a “refresh” attribute cause the individual slides to change at the speed I decide. Once in the show, it cannot be stopped unless the user closes the window.
- User-controlled slide shows, in which each slide has a forward and back button, as well as a stop/close window button (text links could be used instead). This option gives the user control, but as with the first option, the user does not know what’s coming next.
- A third option is to build a grid of thumbnail images on a master page, from which users can select any image. A separate page or a pop-up is then spawned with a larger version of the thumbnail. This gives the user maximum control over viewing options, but it runs the risk of irritating users when the pop-up or second page obscures their view of the thumbnails. This requires the user to constantly be flipping back and forth between thumbnails and larger views.
- A user-friendly option that I will discuss below is to use a frameset design. In this example, I use a three-row design. The top row is for corporate or personal identification. The bottom row is a horizontal scrollbar with thumbnails. The middle row, the largest of the three, is the main viewing area. Upon clicking a thumbnail, the larger image appears in the viewing area, along with any text the programmer wants displayed.

How it Looks

The screenshot below shows how the finished product looks:



The slide show illustrated here was used as a demo for a home builder. The slide show is actually a walking tour of the home, from inside to outside. This method has the advantages that the user is always in control, can view the entire strip of thumbnails, and also see the larger image without having to switch between browser windows.

How to Build It

The first step is to code the frameset. The code below is rather simple, and assumes a basic understanding of HTML frames.

```
<html>
<head>
<title>Cole Stanley Builders</title>
</head>

<frameset rows="50,360,*" frameborder="0" border="no" marginheight="0" marginwidth="0" framespacing="0" scrolling="no">

<frame src="top.html" name=top scrolling="no" frameborder="0" border="no" marginheight="0" marginwidth="0" framespacing="no" noresize border="0">

<frame src="pic01.html" name=middle scrolling="no" frameborder="0" border="no" marginheight="0" marginwidth="0" framespacing="no" noresize border="0">

<frame src="bottom.html" name=bottom scrolling="auto" frameborder="0" border="no" marginheight="0" marginwidth="0" framespacing="no" noresize border="0">

</frameset>

</html>
```

This particular slide show features 15 thumbnails. There are 18 separate HTML files needed for this design: the index file for the frameset, one file each for the top and bottom frames, and 15 different HTML pages for the middle viewing area, one for each larger image. There are also 30 images needed for this site: 15 original images sized to fit the viewing page, and 15 thumbnails that have been significantly reduced so as to improve page-loading time.

You will notice in the screenshot that this is a seamless frameset. This adds elegance to the site, and to the non-programmer will look very sophisticated as pictures change in the main viewing area. It looks like a regular non-frame web page, but it really isn't.

The dead giveaway is the horizontal scrollbar along the bottom. Normally an HTML no-no, the normally-dreaded horizontal scroller is now your best friend. Since this is a slide show and not a standard web page, you can use the scroller to build a thumbnail filmstrip that is fairly long. However, caution must be used to not make it *too* long or else it will slow down page-loading.

The frameset was built to accommodate a thin row at the top for logo and/or text. It need not be a large row; for this reason, only 50 pixels were allocated.

The middle viewing area was set at 360 pixels, which is sufficient for viewing most images. Since most computer users have 800 X 600 resolution, this size will work well. It also mans that your larger images should have a height less than 360 pixels (or whatever frame size you declare).

The bottom row uses the * to allow it to fill up whatever vertical space remains in the user's browser.

It is important to note that the frame sizes are up to the programmer. Also, it is important to name each frame, especially the middle frame, since this is where you will be targeting your larger images. Without the name="" attribute, you will not be able to target from the bottom frame.

Note the src="" atribute in each frame tag. This specifies the HTML file that should be loaded in that frame upon opening the frameset. The top frame opens with a file called top.html; the bottom frame opens with bottom.html. The middle frame is different, though, because this is where a multitude of HTML files will be targeted. I initialize the viewing with the larger version of the first picture. Thus, a file called pic01.html is set as the source file.

One other note: I have set scrolling=no so that vertical scroll bars in the individual frames are turned off. This adds eye-appeal to the site. But beware that by turning off scrolling, if your frames will not fit into the user's window (or if you image and text are larger than the frame), a truncated version of the frame will appear. In other words, you should tinker with your frame sizes to meet your needs. However, it is desirable to have the finished product work well with scrolling turned off because it looks much better.

The Top Frame

This is the easiest file to build because it includes only a small logo and/or text that serves as an identifier. Remember, this should be kept small, so do not use large images or txt. Logos should be resized so that you can minimize page-loading time.

```
<html>
<head>
<title>Top frame</title>
</head>
<body bgcolor=black>
<center>
<font face="arial" size=5 color=white>Cole Stanley Builders</font><br>
<font face="arial" size=2 color=white>2404 Fringe Tree</font>
</center>

</body>
</html>
```

The Middle Frame

This is the main viewing area. You can pretty much do whatever you want to here. I have used a two-column table design, with the image on the left, and text on the right. The bgcolor here (as well as in the other HTML documents) can be set to suit your preferences. I also used the BLOCKQUOTE tag to indent the entire table and thereby give it some white space on all four sides.

The table is built so that it will fit into the designated frame size. The image is also sized accordingly. The only other concern then would be the text. If you have too much text, it will go beyond the bottom of the image. Because scrolling=no for all frames, you would lose whatever text goes beyond the image.

The 15 different pages (all named pic01.html through pic15.html) are easy to mass produce. The only difference between them is the image name and the text. All f the rest of the HTML code is identical.

```
<html>
<head>
<title>Middle Frame</title>
</head>
<body bgcolor=white>

<center>
<blockquote>
<table width=640 bgcolor=white>

<tr>
<td colspan=2 height=20> </td>
</tr>

<tr>
<td valign=top align=center width=440><img src="images/pic01.JPG" width=400 border=2
hspace=10 ></td>
<td valign=top align=left width=200><font face="arial" size=2 color=black>This is Pic #1. It is a
view of the shower in the Master bathroom.</td>
</tr>

<tr>
<td colspan=2 height=20> </td>
</tr>

</table>

</blockquote>
</center>

</body>
</html>
```

The Bottom Frame

This frame is the meat of the matter. It is the control panel for the entire slideshow. I use the metaphor of a film strip to allow users to select an image for larger viewing. A little window dressing (some rollover effects as well as a small image between the thumbnails) makes the whole package look nice.

The first thing to do is resize the original images. I scaled them all down to width=100 height=75. This makes for quicker page-loading; thumbnails need not be any larger than this.

Next, I created a little “divider” image (named divider.jpg). This was used to “tie together” the thumbnails and serve as a common thread throughout the film strip. Its size is minimal, and spruces up an otherwise mundane strip of pictures.

I used one very wide table to crate the horizontal film strip. Each divider and thumbnail has their own TD in this one-row table. Beneath each thumbnail I put a short text identifier.

The thumbnail and text are both links for the user to click. They will spawn the larger image in the middle frame. Note that, for simplicity, I used a common naming scheme throughout this exercise. Picture #1 has an HTML page called pic01.html. Its thumbnail is pic01sm.jpg, the larger image is called pic01.jpg. All images are stored in an images folder, which makes for a tidier root directory.

To add a little pizzazz to the filmstrip, and installed a few basic embedded styles. These are located within the STYLE container in the HEAD of the document.

The first one sets the text link color to white, and turns off the default underlining that links have. The second sets up the hover effect. When a user places his mouse atop the image or the text, the text turns red and becomes underlined. This is a good visual cue to the user that the link is indeed active.

The final style is for the images. I have set a black border for the images. In a page that is already black, this means that the border is actually invisible. But without setting it to black, the default blue border will appear because the image is itself a clickable object. So, for aesthetics, I have removed the blue and replaced it with something that no one can see.

```
<html>
<head>
<title>Thumbnails</title>
<style>
a {color:"#ffffff"; text-decoration:none;}
a:hover {color:"#ff0000"; text-decoration:underline;}
img {border:black 2px}
</style>
</head>
<body bgcolor=black>
<basefont face="arial" size=2 color=white>
<p>
 
<p>
<table border=0 cellspacing=0 cellpadding=0>
<tr>

<td valign=center><img src="images/divider.jpg"></td>
<td valign=center align=center><a href="pic01.html" target="middle"><img
src="images/pic01sm.jpg" width=100 height=75><font face="arial" size=2 color=white><br>Pic
#1</a></td>
<td valign=center><img src="images/divider.jpg"></td>
<td valign=center align=center><a href="pic02.html" target="middle"><img
src="images/pic02sm.jpg" width=100 height=75><font face="arial" size=2 color=white><br>Pic
#2</a></td>
<td valign=center><img src="images/divider.jpg"></td>
```

```
<td valign=center align=center><a href="pic03.html" target="middle"><img
src="images/pic03sm.jpg" width=100 height=75><font face="arial" size=2 color=white><br>Pic
#3</a></td>
<td valign=center><img src="images/divider.jpg"></td>
<td valign=center align=center><a href="pic04.html" target="middle"><img
src="images/pic04sm.jpg" width=100 height=75><font face="arial" size=2 color=white><br>Pic
#4</a></td>
<td valign=center><img src="images/divider.jpg"></td>
<td valign=center align=center><a href="pic05.html" target="middle"><img
src="images/pic05sm.jpg" width=100 height=75><font face="arial" size=2 color=white><br>Pic
#5</a></td>
<td valign=center><img src="images/divider.jpg"></td>
<td valign=center align=center><a href="pic06.html" target="middle"><img
src="images/pic06sm.jpg" width=100 height=75><font face="arial" size=2 color=white><br>Pic
#6</a></td>
<td valign=center><img src="images/divider.jpg"></td>
<td valign=center align=center><a href="pic07.html" target="middle"><img
src="images/pic07sm.jpg" width=100 height=75><font face="arial" size=2 color=white><br>Pic
#7</a></td>
<td valign=center><img src="images/divider.jpg"></td>
<td valign=center align=center><a href="pic08.html" target="middle"><img
src="images/pic08sm.jpg" width=100 height=75><font face="arial" size=2 color=white><br>Pic
#8</a></td>
<td valign=center><img src="images/divider.jpg"></td>
<td valign=center align=center><a href="pic09.html" target="middle"><img
src="images/pic09sm.jpg" width=100 height=75><font face="arial" size=2 color=white><br>Pic
#9</a></td>
<td valign=center><img src="images/divider.jpg"></td>
<td valign=center align=center><a href="pic10.html" target="middle"><img
src="images/pic10sm.jpg" width=100 height=75><font face="arial" size=2 color=white><br>Pic
#10</a></td>
<td valign=center><img src="images/divider.jpg"></td>
<td valign=center align=center><a href="pic11.html" target="middle"><img
src="images/pic11sm.jpg" width=100 height=75><font face="arial" size=2 color=white><br>Pic
#11</a></td>
<td valign=center><img src="images/divider.jpg"></td>
<td valign=center align=center><a href="pic12.html" target="middle"><img
src="images/pic12sm.jpg" width=100 height=75><font face="arial" size=2 color=white><br>Pic
#12</a></td>
<td valign=center><img src="images/divider.jpg"></td>
<td valign=center align=center><a href="pic13.html" target="middle"><img
src="images/pic13sm.jpg" width=100 height=75><font face="arial" size=2 color=white><br>Pic
#13</a></td>
<td valign=center><img src="images/divider.jpg"></td>
<td valign=center align=center><a href="pic14.html" target="middle"><img
src="images/pic14sm.jpg" width=100 height=75><font face="arial" size=2 color=white><br>Pic
#14</a></td>
<td valign=center><img src="images/divider.jpg"></td>
<td valign=center align=center><a href="pic15.html" target="middle"><img
src="images/pic15sm.jpg" width=100 height=75><font face="arial" size=2 color=white><br>Pic
#15</a></td>
<td valign=center><img src="images/divider.jpg"></td>


</tr>
</table
```

</body>
</html>

I would not recommend a film strip going much beyond 15 thumbnails, because page-loading time will become an issue. The coding presented above is totally scalable to whatever number of thumbnails you wish to include.

Site Navigation

One thing this example does not have is a way to get back to wherever the user originated. Assuming you add a slide show to an existing site you will want to provide an easy return for your users. But you will have to explode the frameset design to do so. In addition, you will need to add a “back” link to each of the middle frame documents. It would look something like this:

Back

Of course, the name of the document to which you are returning will be determined by the programmer. I would put this “back” link beneath the text in ach of the documents that appear in the middle frame.

Summary

This film strip is fairly easy to build. Not counting the time spent actually taking the pictures, the entire project took about two hours. This included resizing images from their original format (in this case, I has to downsize to create both the larger and thumbnail images, because my original images were 640 X 480). The rest of the time was spent coding the HTML.

While this slide show is not perfect, it does offer an easy alternative, and one that offers great ease of use.