

MOSS UI Development

Applying custom visual design to a MOSS website using Master Pages and CSS

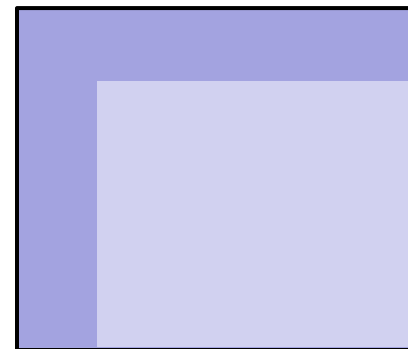
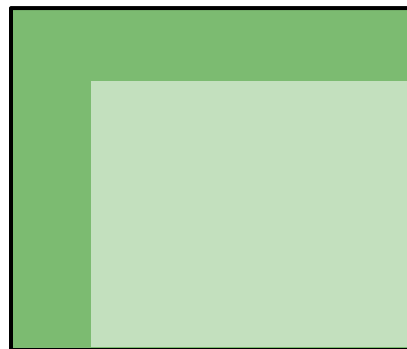
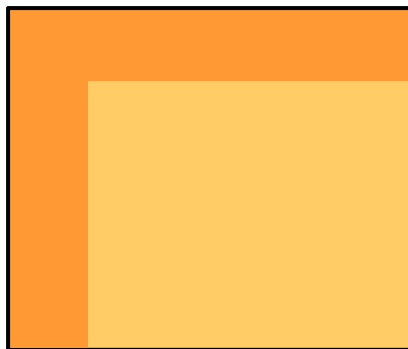
10: CSS Variations in a MOSS Website

February 2008

e.magination

CSS Variations

To achieve different **color schemes** or **variations in layout**, additional CSS may be applied at the **page layout** level or **subsite** level.



CSS Override Methods

The purpose of adding CSS at the Page Layout level or Subsite level is to **override** various existing styles -- not only in Core.css, but also in your own custom CSS file.

- **Page Layout Method:** CSS classes are added directly in the Page Layout code. This change will be reflected on *any page based on this layout*, no matter where it resides in the website.
- **Subsite Method:** An additional CSS file is assigned via the Subsite settings. This change will be reflected on *all pages within the subsite*.



Adding CSS to a Page Layout

Here are a few examples of how adding CSS into a **Page Layout** could be useful:

- If the homepage needs different column widths or column layouts than the interior pages
- If the homepage needs a different color scheme than interior pages
- If different landing pages need different color schemes
- If certain elements need to be hidden in certain layouts
- ... and much more



Adding CSS to a Page Layout

Use the preexisting Sharepoint Content Placeholder called **PlaceholderAdditionalPageHead** to place CSS code directly into the Page Layout. The code will appear in the **<head>** of the rendered page, and *it will override your main CSS file.*

Example Page Layout code in SPD:

```
<asp:Content ContentPlaceholderID="PlaceHolderAdditionalPageHead"
runat="server">
    <style type="text/css">
        #EMAG_leftCol {
            width: 600px;
        }
    </style>
</asp:Content>
```

Adding CSS to a Subsite

Here are a few examples of how adding CSS to a **Subsite** could be useful:

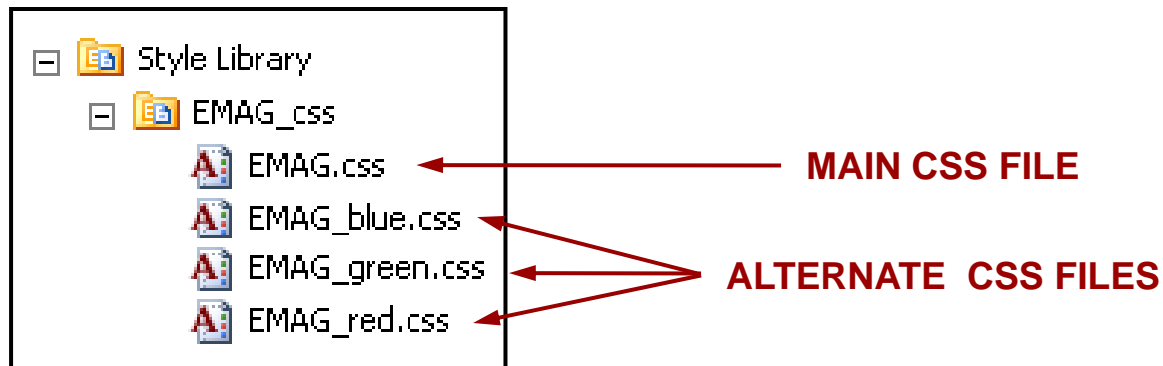
- If all pages in that subsite need a different color scheme than the rest of the site
- If all pages in that subsite need a different logo or graphic to appear in the background
- If certain elements need to be hidden in all pages of that subsite
- ... and much more



Adding CSS to a Subsite

Subsites generally inherit the CSS file setting that you applied to the root site. If you would like to apply a different CSS file to a subsite, you can change the setting for that subsite.

Begin by creating the CSS file that will be used for the subsite. Place the files in the appropriate Style Library folder in SPD.



Adding CSS to a Subsite

Do not repeat all styles from the main CSS in your alternate CSS. You only need to include those styles that you are overriding.

Use the `@import` rule to capture all the styles from the main CSS.

Example content of an alternate file called **EMAG_green.css**:

```
@import url(EMAG.css);

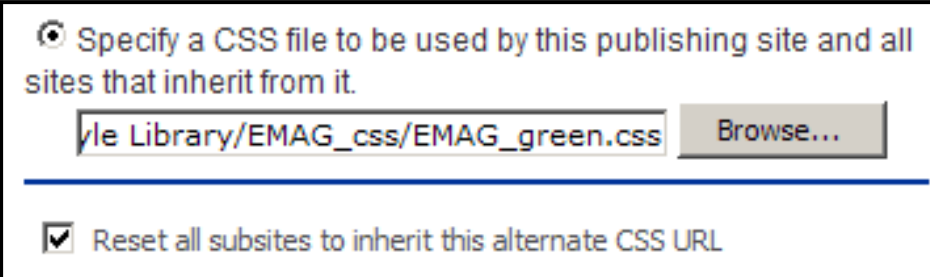
#EMAG_rightCol {
    background: url(../EMAG_images/rCol_bkgGreen.gif) repeat-y;
}

#EMAG_footer {
    background-color: #006633;
    color: #99FFCC;
}
```

Adding CSS to a Subsite

Once the alternate CSS file has been created, you may assign the file to the desired Subsite.

1. Open the MOSS website URL in a browser.
2. Navigate to the desired Subsite.
3. Go to *Site Actions > Site Settings > Look and Feel > Master Page*
4. The CSS Settings are near the bottom of the Master Page Settings.
5. Choose the option to Specify a CSS file to be used by this site.
6. You may browse to your alternate CSS file, or type in the URL
7. Check the box to reset all subsites to use the same CSS file (this is optional, and it will only effect subsites *within* the current subsite)
8. Click OK to save your CSS settings.



Specify a CSS file to be used by this publishing site and all sites that inherit from it.

Reset all subsites to inherit this alternate CSS URL

END of 10:
CSS Variations
in a MOSS Website

February 2008

e.magination