

# HtmlTemplatesAndBorderComponent

NOTE: This is outdated information that applies only to Tapestry 4. For current information see <http://tapestry.apache.org/layout-component.html>.

In [Mid 2005 I was] bitten by the HTML/CSS standards bug. I wanted to write semantically rich HTML code that would validate with validator.w3.org (except for the minor jwcids), I could look at direct in my browser of choice, and that Tapestry would take and generate similarly good HTML.

The standard HTML page example is the one in section 6.6.3 of Tapestry in Action:

```
<html>
<head>
<title>Tapestry Hangman</title>
<link rel="stylesheet" type="text/css" href="css/hangman.css"/>
</head>
<body jwcid="$content$"
<span jwcid="@Border">

. . .

</span>
</body>
</html>
```

My solution is to move the \$content\$ up to html tag, and the Border component moves to the body tag:

```
<html jwcid="$content$"
<head>
<title>Tapestry Hangman</title>
<link rel="stylesheet" type="text/css" href="css/hangman.css"/>
</head>
<body jwcid="@Border">

. . .

</body>
</html>
```

My full version adds the following bells and whistles:

- DOCTYPE for xhtml, so that IE6 and Firefox stay away from quirks mode.
- The html tag has xmlns and lang attributes. The lang attribute is set using the page's locale.
- Snazzy use of conditional comments to import browser specific css files. This requires a special renderer.

Here's what a standard html page looks like:

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" lang="en" xml:lang="en" jwcid="$content$">
    <head jwcid="$remove$">
        <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
        <title>Login</title>
        <link rel="stylesheet" type="text/css" href="css/emv2.css"/>
        <!--[if IE 6]>
            <link rel="stylesheet" type="text/css" href="css/ie6.css"/>
        <![endif]-->
        <!--[if IE 7]>
            <link rel="stylesheet" type="text/css" href="css/ie7.css"/>
        <![endif]-->
        <![if !IE]>
            <link rel="stylesheet" type="text/css" href="css/nonIE.css"/>
        <![endif]>
    </head>
    <body jwcid="@Border" baseStylesheet="ognl:assets.baseStylesheet">

    . . .

    </body>
</html>

```

My Border component html looks like this:

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html jwcid="@Shell" title="ognl:page.pageTitle" doctype="
    stylesheet="ognl:stylesheet" delegate="ognl:headerDelegate"
    xmlns="http://www.w3.org/1999/xhtml"
    lang="ognl:page.locale.language" xml:lang="ognl:page.locale.language">
    <head jwcid="$remove$">
        <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
        <link rel="stylesheet" type="text/css" href="css/base.css"/>
        <!--[if IE 6]>
            <link rel="stylesheet" type="text/css" href="css/ie6.css"/>
        <![endif]-->
        <!--[if IE 7]>
            <link rel="stylesheet" type="text/css" href="css/ie7.css"/>
        <![endif]-->
        <![if !IE]>
            <link rel="stylesheet" type="text/css" href="css/nonIE.css"/>
        <![endif]>
        <title>Border Component</title>
    </head>
    <body jwcid="@Body">

    <div id="header">
    </div>

    <div id="contentBlock">
        <span jwcid="@RenderBody">
            Body content goes here.
        </span>
    </div>

    <div id="footer">
    </div>
    </body>
</html>

```

My IRender to do the conditional comments in the head tag takes 4 stylesheet assets:

1. baseStylesheet is a standard stylesheet that is used in all pages for all browsers.
2. ie6Stylesheet is a stylesheet for IE 6 only.
3. ie7Stylesheet is likewise for IE7 (nothing like staying ahead of the game).
4. nonIEStylesheet is for all non-IE browsers, e.g. Firefox, Opera and Safari. As these are more standards compliant, there should be less need for hacks here.

Here is the code:

```
package de.xcom.emv.tapestry;

import org.apache.tapestry.IAsset;
import org.apache.tapestry.IMarkupWriter;
import org.apache.tapestry.IRender;
import org.apache.tapestry.IRequestCycle;

/**
 * This delegate is invoked before the </head> tag of the rendering @Shell component.
 * It is used to write out the correct styleheets using conditional comments.
 *
 * It also supports calling additional delegates to continue the good work.
 *
 * @author joconnor
 */
public class StylesheetDelegate implements IRender {
    private IRender nextDelegate;
    private IAsset ie6Stylesheet;
    private IAsset ie7Stylesheet;
    private IAsset nonIEStylesheet;
    private IAsset baseStylesheet;

    public StylesheetDelegate(IAsset ie6, IAsset ie7, IAsset nonIE, IAsset base, IRender nextDelegate) {
        this.ie6Stylesheet = ie6;
        this.ie7Stylesheet = ie7;
        this.nonIEStylesheet = nonIE;
        this.baseStylesheet = base;
        this.nextDelegate = nextDelegate;
    }

    public void render(IMarkupWriter writer, IRequestCycle cycle) {
        writeStylesheetLink(writer, cycle, baseStylesheet);
        writeStylesheetLink(writer, cycle, "<!--[if IE 6]>\n", ie6Stylesheet, "<![endif]-->\n");
        writeStylesheetLink(writer, cycle, "<!--[if IE 7]>\n", ie7Stylesheet, "<![endif]-->\n");
        writeStylesheetLink(writer, cycle, "<![if !IE]>\n", nonIEStylesheet, "<![endif]>\n");

        if (nextDelegate != null) {
            nextDelegate.render(writer, cycle);
        }
    }

    private void writeStylesheetLink(IMarkupWriter writer, IRequestCycle cycle, String ifCond, IAsset
stylesheet, String endIf) {
        if (stylesheet != null) {
            writer.printRaw(ifCond);
            writeStylesheetLink(writer, cycle, stylesheet);
            writer.printRaw(endIf);
        }
    }

    private void writeStylesheetLink(IMarkupWriter writer, IRequestCycle cycle, IAsset stylesheet) {
        if (stylesheet != null) {
            writer.beginEmpty("link");
            writer.attribute("rel", "stylesheet");
            writer.attribute("type", "text/css");
            writer.attribute("href", stylesheet.buildURL(cycle));
            writer.println();
        }
    }
}
```