Blog Portfolio

Using portfolio owner created content to create a "blog" portfolio

The purpose of this example is to show an instructional designer might be able to create a simple presentation template that a portfolio owner might use to display their content. In this case, we will:

- use a simple OSP form to create structured data
- create a passthrough portfolio template that requires blog entries from portfolio authors
- add blog entries to the passthrough template and inspect the resulting raw XML result
- create a new portfolio template that organizes and presents our data in a useful way
- create a portfolio and publish portfolio created content

One of the simplest use cases for a "portfolio" would involve some sort of "blog-like" activity, where a student may be asked to write a series of journal entries to discuss what they are learning in a class or program. A blog is a reverse chronological presentation of those journal entries, published for a specific audience. The OSP portfolio tool has the interesting feature of allowing the portfolio owner to choose who might be able to see their portfolio presentation and may have some advantages over "public-only" blog tools for some educational environments. Besides, what presentation is more "Hello World"-ish than a blog?

Import the Blog Entry form.

Let's assume that each blog entry will consist of a title and a rich text area. We could add additional form elements (a teaser perhaps), but lets keep it simple with just the one element. Attached you will find a Blog Entry Form that you can use for this example. Import the form and publish it.

Complete the form a few times to create some "Blog Entries" in your resources.

Create a passthrough portfolio template that requires blog entries from portfolio authors

- Create a new Portfolio Template that uses the passthrough.xsl file we used in previous examples or import the attached template.
    - Use the passthrough xsl from the previous examples
    - On Step 3 of the template wizard you have the opportunity to prompt the the portfolio author to supply content to their portfolio of a certain type. For this example we want to just use the "Blog Entry" form.
        - Select the "Blog Entry" form as the "Type"
        - Enter "blogEntry" as the "Name" of this type of content. This will be the name of the element as it appears in the passthrough xml.
        - Enter "Blog Entry" as the "Title" of this type of content.
        - Enter "You should only attach Blog Entry Forms to this Template." as the description. This will presented to the portfolio author as the prompt to supply the right type of content for their portfolio.
        - Select "Yes" to Allow Multiple Selections so that the author can attach many entries to their blog.
        - Click the "Add to list" button
    - Click the "Continue" button to go to the next step of the wizard
    - Complete step 4 (the supporting files page) of the wizard
        - For the "Name (used in xpath)", enter "myLogo" (this name will be an element name in the passthrough file).
        - choose an image from your resources to be your logo.
        - click the "Add to List" button
    - Click the "Finish" button to complete the Template wizard
- Publish the template.

Create a portfolio using the template created above inspect the passthrough XML.

- Give your blog a title.
- "Add all" of your completed "Blog Entry" forms
- Click the "Finish" button to complete the portfolio wizard
- Click on the name of your portfolio and view the passthrough XML. It should look something like this:

```
<?xml version="1.0" encoding="UTF-8"?>
<ospiPresentation>
    <blogEntry>
        <artifact>
            <metaData>
                <id>2f1d7c8b-a1b9-4e4e-0039-43926cb30fb5</id>
                <displayName>Blog entry #2</displayName>
                <type>
                    <id>file</id>
                    <description>file</description>
                </type>
                <repositoryNode>
                    <created>Mon Jan 01 09:13:09 EST 2007</created>
```

```
                    <modified>Mon Jan 01 09:13:09 EST 2007</modified>
                    <size>157</size>
                    <mimeType>
                        <primary>application</primary>
                        <sub>x-osp</sub>
                    </mimeType>
                </repositoryNode>
            </metaData>
            <structuredData>
                    <entryBody>This is blog entry #2. I sure hope this works.</entryBody>
                </blogEntry>
            </structuredData>
            <schema>
                <instructions>This is a blank text area for you to write a blog
                    entry.&amp;amp;nbsp;&amp;amp;nbsp;</instructions>
                <element name="blogEntry" minOccurs="1" maxOccurs="1">
                    <xs:annotation xmlns:xs="http://www.w3.org/2001/XMLSchema">
                        <xs:documentation source="ospi.label">Blog Entry</xs:documentation>
                        <xs:documentation source="ospi.description">Blog Entry</xs:documentation>
                    </xs:annotation>
                    <children>
                        <element name="entryBody" type="xs:string" minOccurs="1" maxOccurs="1">
                            <xs:annotation xmlns:xs="http://www.w3.org/2001/XMLSchema">
                                <xs:documentation source="ospi.label">Body</xs:documentation>
                                <xs:documentation source="ospi.description">Blog Entry Body</xs:
documentation>
                                <xs:documentation source="ospi.isRichText">true</xs:documentation>
                            </xs:annotation>
                        </element>
                    </children>
                </element>
            </schema>
        </artifact>
        <artifact>
            <metaData>
                <id>7b59c9c0-39fc-4b07-8065-902b53d47a63</id>
                <displayName>Blog entry 1</displayName>
                <type>
                    <id>file</id>
                    <description>file</description>
                </type>
                <repositoryNode>
                    <created>Mon Jan 01 09:11:19 EST 2007</created>
                    <modified>Mon Jan 01 09:11:19 EST 2007</modified>
                    <size>152</size>
                    <mimeType>
                        <primary>application</primary>
                        <sub>x-osp</sub>
                    </mimeType>
                </repositoryNode>
            </metaData>
            <structuredData>
                <blogEntry>
                    <entryBody>This is blog entry #1. I want to see if I can write a
                    blog.</entryBody>
                </blogEntry>
            </structuredData>
            <schema>
                <instructions>This is a blank text area for you to write a blog
                    entry.&amp;amp;nbsp;&amp;amp;nbsp;</instructions>
                <element name="blogEntry" minOccurs="1" maxOccurs="1">
                    <xs:annotation xmlns:xs="http://www.w3.org/2001/XMLSchema">
                        <xs:documentation source="ospi.label">Blog Entry</xs:documentation>
                        <xs:documentation source="ospi.description">Blog Entry</xs:documentation>
                    </xs:annotation>
                    <children>
                        <element name="entryBody" type="xs:string" minOccurs="1" maxOccurs="1">
                            <xs:annotation xmlns:xs="http://www.w3.org/2001/XMLSchema">
                                <xs:documentation source="ospi.label">Body</xs:documentation>
                                <xs:documentation source="ospi.description">Blog Entry Body</xs:
documentation>
```

```
                            <xs:documentation source="ospi.isRichText">true</xs:documentation>
                        </xs:annotation>
                    </element>
                </children>
            </element>
        </schema>
    </artifact>
</blogEntry>
<presentationFiles>
    <myLogo>
        <artifact>
            <metaData>
                <id>361dcb3e-1a57-4d37-0005-8e21d15d8cc8</id>
                <displayName>SOElogo.gif</displayName>
                <type>
                    <id>file</id>
                    <description>file</description>
                </type>
                <repositoryNode>
                    <created>Thu Dec 28 10:38:24 EST 2006</created>
                    <modified>Thu Dec 28 10:38:25 EST 2006</modified>
                    <size>6673</size>
                    <mimeType>
                        <primary>image</primary>
                        <sub>gif</sub>
                    </mimeType>
                </repositoryNode>
            </metaData>
            <fileArtifact>
                <uri>https://eportfolio.syr.edu:8443/access/ospPresentation/7b48753d-d64b-4548-00c1-
7b801c0a31e4/ \
                     CCEF1DE2758DF4E1B3CD195370282C3C/content/user/smkeesle
/A560B424751D6D819170D15FD15B7941/ \
                     SOElogo.gif</uri>
            </fileArtifact>
        </artifact>
    </myLogo>
</presentationFiles>
</ospiPresentation>
```

Create a new portfolio template

We have modified the Hello World xsl file below to query the passthrough XML and display the user content we need:

Element to Display	XPATH
Blog Entry title	/ospiPresentation/blogEntry/artifact/metaData/displayName
Blog Entry post date	/ospiPresentation/blogEntry/artifact/metaData/repositoryNode/created
Blog Entry body	/ospiPresentation/blogEntry/artifact//structuredData/blogEntry/entryBody

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0" xmlns:urwg="http://www.gridforum.
org/2003/ur-wg"
    xmlns:SimpleDateFormat="http://xml.apache.org/xalan/java/java.text.SimpleDateFormat" xmlns:Date="http://xml.
apache.org/xalan/java/java.util.Date">

    <!-- Declaration of parameters -->
    <!-- if you want to use a Sakai value or a passed in parameter from the -->
    <!-- query string, you should state it here -->

    <!-- The id parameter from the query string -->
    <xsl:param name="id"/>

    <!-- The sakai.tool.placement.ia parameter from the query string -->
    <xsl:param name="sakai.tool.placement.id"/>
```

```
    <!-- a number we can switch on to display different content for each page -->
    <xsl:param name="pageNumber"/>

    <!-- if passed, the created variable will be a key to a single blog Entry -->
    <xsl:param name="created"/>

    <!-- Declare output method -->
    <xsl:output method="html"/>

    <!-- Main Template outputs our HTML page -->
    <xsl:template match="/">
        <xsl:choose>
            <xsl:when test="$pageNumber=4"> Rss! </xsl:when>
            <xsl:otherwise>
                <html>
                    <head>
                        <title>My Blog</title>
                        <link rel="stylesheet" type="text/css">
                            <!--<xsl:attribute name="href">blogStyle.css</xsl:attribute>-->
                            <xsl:attribute name="href">
                                <xsl:value-of select="/ospiPresentation/presentationFiles/style/artifact
/fileArtifact/uri"/>
                            </xsl:attribute>
                        </link>
                    </head>
                    <body>
                        <div id="wrapper">
                            <div id="header">
                                <xsl:attribute name="style">
                                    <xsl:text>background-image:url('</xsl:text>
                                    <xsl:value-of select="/ospiPresentation/presentationFiles/backgroundImage
/artifact/fileArtifact/uri"/>
                                    <xsl:text>')</xsl:text>
                                </xsl:attribute>
                                <img>
                                    <xsl:attribute name="src">
                                        <xsl:value-of select="/ospiPresentation/presentationFiles/myLogo
/artifact/fileArtifact/uri"/>
                                    </xsl:attribute>
                                    <xsl:attribute name="alt">My Institutions Logo</xsl:attribute>
                                    <xsl:attribute name="style">float: right;</xsl:attribute>
                                </img>
                                <h1>My Blog</h1>
                            </div>
                            <!-- a little menu to navigate -->
                            <div id="menu">
                                <ul>
                                    <li>
                                        <xsl:call-template name="menuLink">
                                            <xsl:with-param name="pageNumber"/>
                                            <xsl:with-param name="text">Last 5</xsl:with-param>
                                        </xsl:call-template>
                                    </li>
                                    <li>
                                        <xsl:call-template name="menuLink">
                                            <xsl:with-param name="pageNumber">1</xsl:with-param>
                                            <xsl:with-param name="text">About Me</xsl:with-param>
                                        </xsl:call-template>
                                    </li>
                                    <li>
                                        <xsl:call-template name="menuLink">
                                            <xsl:with-param name="pageNumber">2</xsl:with-param>
                                            <xsl:with-param name="text">Entry Index</xsl:with-param>
                                        </xsl:call-template>
                                    </li>
                                    <li>
                                        <xsl:call-template name="menuLink">
                                            <xsl:with-param name="pageNumber">4</xsl:with-param>
                                            <xsl:with-param name="text">RSS Feed</xsl:with-param>
                                        </xsl:call-template>
```

```
                                    </li>
                                </ul>
                            </div>
                            <div id="contentWrapper">
                                <div id="entryNav">
                                    <h3>Blog Posts</h3>
                                    <ul>
                                        <xsl:for-each select="/ospiPresentation/blogEntry/artifact">
                                            <xsl:sort data-type="number"
                                                select="Date:getTime(SimpleDateFormat:parse(SimpleDateFormat:new
('EEE MMM dd kk:mm:ss zzz yyyy'), ./metaData/repositoryNode/created))"
                                                order="descending"/>
                                            <li>
                                                <a>
                                                    <xsl:attribute name="href">
                                                        <xsl:text disable-output-escaping="yes"
>viewPresentation.osp?sakai.tool.placement.id=</xsl:text>
                                                        <xsl:value-of select="$sakai.tool.placement.id"/>
                                                        <xsl:text disable-output-escaping="yes">&amp;id=</xsl:
text>
                                                        <xsl:value-of select="$id"/>
                                                        <xsl:text disable-output-escaping="yes">&amp;
pageNumber=3&amp;created=</xsl:text>
                                                        <xsl:value-of select="./metaData/repositoryNode/created"
/>
                                                    </xsl:attribute>
                                                    <xsl:value-of select="./metaData/displayName"/>
                                                </a>
                                            </li>
                                        </xsl:for-each>
                                    </ul>
                                </div>
                                <!-- a switch to select content for this page based on the value of pageNumber
-->
                                <div id="content">
                                    <xsl:choose>
                                        <xsl:when test="$pageNumber=1">
                                            <!-- about me --> About me Page </xsl:when>
                                        <xsl:when test="$pageNumber=2">
                                            <!-- display an index of all blog entries -->
                                            <h3>All Entries</h3>
                                            <ul id="datedEntryList">
                                                <xsl:for-each select="/ospiPresentation/blogEntry/artifact">
                                                    <xsl:sort data-type="number"
                                                        select="Date:getTime(SimpleDateFormat:parse
(SimpleDateFormat:new('EEE MMM dd kk:mm:ss zzz yyyy'), ./metaData/repositoryNode/created))"
                                                        order="descending"/>
                                                    <li>
                                                        <a>
                                                            <xsl:attribute name="href">
                                                                <xsl:text disable-output-escaping="yes"
>viewPresentation.osp?sakai.tool.placement.id=</xsl:text>
                                                                <xsl:value-of select="$sakai.tool.placement.id"
/>
                                                                <xsl:text disable-output-escaping="yes">&amp;
id=</xsl:text>
                                                                <xsl:value-of select="$id"/>
                                                                <xsl:text disable-output-escaping="yes">&amp;
pageNumber=3&amp;created=</xsl:text>
                                                                <xsl:value-of select="./metaData/repositoryNode
/created"/>
                                                            </xsl:attribute>
                                                            <xsl:value-of select="./metaData/displayName"/> -
<xsl:value-of select="./metaData/repositoryNode/created"/>
                                                        </a>
                                                    </li>
                                                </xsl:for-each>
                                            </ul>
                                        </xsl:when>
                                        <xsl:when test="$pageNumber=3">
                                            <!-- individual entry displayed -->
```

```
                                        <xsl:for-each select="/ospiPresentation/blogEntry/artifact[metaData
/repositoryNode/created=$created]">
                                            <xsl:call-template name="entry">
                                                <xsl:with-param name="title">
                                                    <xsl:value-of select="./metaData/displayName"/>
                                                </xsl:with-param>
                                                <xsl:with-param name="date">
                                                    <xsl:value-of select="./metaData/repositoryNode/created"
/>
                                                </xsl:with-param>
                                                <xsl:with-param name="body">
                                                    <xsl:value-of select="./structuredData/blogEntry
/entryBody"/>
                                                </xsl:with-param>
                                            </xsl:call-template>
                                        </xsl:for-each>
                                    </xsl:when>
                                    <xsl:otherwise>
                                        <!-- last 5 blog entries --
>
                                        <xsl:for-each select="/ospiPresentation/blogEntry/artifact
[position() &gt; 0]">
                                            <!-- sort by year, moth, day created dates formatted like: Tue
Jan 09 09:59:52 EST 2007 -->
                                            <xsl:sort data-type="number"
                                                select="Date:getTime(SimpleDateFormat:parse
(SimpleDateFormat:new('EEE MMM dd kk:mm:ss zzz yyyy'), ./metaData/repositoryNode/created))"
                                                order="descending"/>
                                            <xsl:call-template name="entry">
                                                <xsl:with-param name="title">
                                                    <xsl:value-of select="./metaData/displayName"/>
                                                </xsl:with-param>
                                                <xsl:with-param name="date">
                                                    <xsl:value-of select="./metaData/repositoryNode/created"
/>
                                                </xsl:with-param>
                                                <xsl:with-param name="body">
                                                    <xsl:value-of select="./structuredData/blogEntry
/entryBody"/>
                                                </xsl:with-param>
                                            </xsl:call-template>
                                        </xsl:for-each>
                                    </xsl:otherwise>
                                </xsl:choose>
                            </div>
                            <!-- content -->
                            <div style="clear: both"/>
                        </div>
                        <!-- contentWrapper -->
                    </div>
                    <!-- end wrapper div -->
                </body>
            </html>

        </xsl:otherwise>
    </xsl:choose>

</xsl:template>

<xsl:template name="entry">
    <xsl:param name="title"/>
    <xsl:param name="date"/>
    <xsl:param name="body"/>
    <div class="entry">
        <h3>
            <xsl:value-of select="$title"/>
        </h3>
        <div class="date">
            <xsl:value-of select="$date"/>
        </div>
        <div class="entryBody">
```

```
                <xsl:value-of select="$body" disable-output-escaping="yes"/>
            </div>
        </div>
    </xsl:template>

    <!-- a named template to menu display links correctly -->
    <xsl:template name="menuLink">
        <xsl:param name="pageNumber"/>
        <xsl:param name="text"/>
        <a>
            <xsl:attribute name="href">
                <xsl:text disable-output-escaping="yes">viewPresentation.osp?sakai.tool.placement.id=</xsl:text>
                <xsl:value-of select="$sakai.tool.placement.id"/>
                <xsl:text disable-output-escaping="yes">&amp;amp;id=</xsl:text>
                <xsl:value-of select="$id"/>
                <xsl:text disable-output-escaping="yes">&amp;amp;pageNumber=</xsl:text>
                <xsl:value-of select="$pageNumber"/>
            </xsl:attribute>
            <xsl:value-of select="$text"/>
        </a>
    </xsl:template>

</xsl:stylesheet>
```