

Wenatchee Valley College

System Developer Notes

- This implementation contains two phases:
- Phase one correspond to the implementation of all templates for interior and home pages.
- Phase two correspong to the implementation of OU Blog, OU Search, OU Mashup and OU Calender
- It is possible some components or snipptets will be obsolated after the implementation of phase two.

Resources

- Global resources exist in /_resources folder on the [www-2016](#) site
- Javascript: /_resources/js
- CSS: /_resources/css
- Images: /_resources/images
- Fonts: /_resources/fonts
- XSL Templates: /_resources/xsl
- ASP & C# Scripts: /_resources/scripts.
- Templates: /_resources/ou/templates
- Includes: /_resources/includes
- Snippets: /_resources/snippets
- Search: /_resources/search (all scripts, xsl... necessary to implement OU Search)

- Social: /_resources/social (all scripts, xsl... necessary to implement OU Mashup)

Include Files

This include files are global so when publishing it will affect all the pages in all pages.

Headcode Include

- Location: /_resources/includes/heacode.inc(Source Code)

Analytics Include

- Location: /_resources/includes/analytics.inc(Source Code)
- This is a location to place all the analytics of the universay website
- The contents file gets placed at the bottom of every page when publishing it will affect all pages

Footcode Include

- Location: /_resources/includes/footcode.inc (Source Code)
- Contains the scripts that exist before the ending `</body>` tag

Search Include

- Location: `/_resources/includes/search.inc` (Source Code)
- Contain the search html for the search in the header.
- **Important** Form input fields and actions must not be changed in order to have OU Search working. *(See OU Search Section for more information)*

Mobile Search Include

- Location: `/_resources/includes/mobile-search.inc` (Source Code)
- Contain the search html for the search in the header in mobile mode.
- **Important** Form input fields and actions must not be changed in order to have OU Search working. *(See OU Search Section for more information)*

Blog Subscribe Include

- Location: `/_resources/includes/blog-subscribe.inc` (Source Code)
- Contains html code to subscribe to blog news. *(See OU Blog Section for more information)*

PCF TO INCLUDE

Alert PCF

- Location: `*/_resources/includes/alert.pcf*`

- XSL Location: */_resources/xsl/includes/alert.xsl*
- This file allows user to activate/deactivate alert in the web.
- Multiedit content allow user to config alert message:
 - Enable Emergency Alert: Activate emergency message
 - Type: Allow user to choose between a banner at the top of the site or a modal pop up.
 - Level: Allow user to select different levels that will add different classes with different colors for the banner.
Only applicable to banner
 - Alert Title: Title for the alert
 - Alert Learn More: Link to page with more information about the alert.
 - Alert Date/Time: Date and time to display in the alert
 - Alert User Cache: Number of minutes that will last until the message shows again after the user has closed the alert message.

Quick Links Include

- Location: */_resources/includes/quicklinks.pcf*
- XSL Location: */_resources/xsl/includes/quicklinks.xsl*
- Allow user to create quick link panel that will show when user clicks Quick Links buttons on the header.

Page Parameters:

- Logo Image: Image for logo in the quicklinks.

Editable regions:

- One editable region with three table transformations.
- First table allow user to select a list of links with icons.
 - Title: To set title of the list of links
 - Class allows user to select a icon from fontawesome. To add more classes to preview, update quicklink-classes.html
 - Link: Link to the page.
- Other two tables allow user to add list of links without icons
 - First column is the title of the list of links. For example, tool
 - Second column will be a list of links with the following structure

```
<ul>
  <li><a href="#">Student Email</a></li>
  <li><a href="#">Student Handbook</a></li>
  <li><a href="#">Campus Maps</a></li>
</ul>
```

Header Include

- Location: */_resources/includes/header.pcf *
- XSL Location: /_resources/xsl/includes/header.xsl
- Allow user to modify header.

Page Parameters:

- Logo Image: Image for logo in the header.

Editable Regions

- Right Links
 - Edit text and links on the right of the footer.
 - Use quicklinks class on the anchor to trigger Quick Links Panel. *This is the default config*
- Header Bar: Config the links and text of the three buttons over the Search
- BT Links: Edit links under the search

Meganav Include

- Location: **/_resources/includes/meganav.pcf **
- XSL Location: */_resources/xsl/includes/meganav.xsl*
- Allow user to modify the meganav under the header.

Page Parameters:

- Eight tabs to set the number of main tabs to display.
 - Display tab: To hide/display a tab
 - Display subsection: To hide/display subsection for each tab
 - Link Title: Link to show in the tab
 - Link URL: Link to page in the main tab.
- Phone information to use in mobile version.
 - Phone Text Left: Phone text to display on the left (*Wenatchee by default*)
 - Phone Number Left: Phone number on the left
 - Phone Text Right: Phone text to display on the right

(Omak by default)

- Phone Number Right: Phone number on the right

Editable Regions:

- One editable region for each subsection. *Only available if Display Subsection is yes*
- Each subsection have two table transformation:
 - One with five columns with a header and a list of links.
If column is empty it will not displayed
 - Header will display the header on the list
 - Content will be a list of links to other pages. It can contain until two level of nested links on the content section.
 - Extra Content. One panel where user can add any content to display at the right of the links.

Footer Include

- *Location: /_resources/includes/footer.pcf*
- *XSL Location: /_resources/xsl/includes/footer.xsl*
- Please update the standard footer in this page.

Page Parameters:

- Logo Image: To choose an image as logo in the footer
- Section 1 Title: Select the title of the first section in the footer
- Section 2 Title: Select the title of the second section in the footer

Editable regions:

- Two editable regions to modify content in both sections. User can add any content to those sections

Athletics Header Include

- *Location: /_resources/includes/athletics-header.pcf*
- XSL Location: */_resources/xsl/includes/athletics-header.xsl*
- User can edit the header for athletics page in this page. To see this header, use directable variable athletics in your section.
(See Directory Variable section for more information)

Editable Regions:

- Left Links: To edit links at the left of the logo
- Right Links: To edit links at the right of the logo

Athletics Footer Include

- *Location: /_resources/includes/athletics-footer.pcf*
- XSL Location: */_resources/xsl/includes/athletics-footer.xsl*
- User can edit the footer for athletics page in this page. To see this footer, use directable variable athletics in your section.
(See Directory Variable section for more information)
- Same parameters and regions apply as in standard footer.

Templates

Interior Page

- Interior page is the standard page for Wenatchee pages.

Page properties

- Page Title is the title in the browser.
- Description and keywords will be used as metadata in the head of the page
- Page Heading is the header that will display at the top of the page.
- Heading placement allows editor to set the header on the left, center or right of the main container.
- Breadcrumb is the page name that will display in the breadcrumb
- Display sidebar allows editor to display/hide sidebar on the right.
- Select Feature allows user to select different feature options in the top of the page. Three options:
 - Feature Image: Display an image as a banner.
 - Feature Editable Region: Display an editable region where editors can add any content
 - None: Do not display this region
- Feature Image: Image to select as banner. *Only applicable if Feature Image is selected*
- Display Fullwidth: Enable/disable full width section at the bottom of the page.
- LDP Gallery: Allow user to select how LDP Galleries are

displayed in the page (Con Slider or Fancy Box). *See LDP Gallery section for more info*

Editable Regions:

- Main Content: Main container of the site. This region will be always available.
- Feature Content: Region at the top of the page Only available if Feature Editable Region is selected
- Sidebar Content: Sidebar content below the side navigator. Only available if sidebar is selected
- Full width Content: Full width content at the bottom of the main container. Only available if full-width content is selected.

Area of Study

- This is a Interior Page with an additional structure “Areas of Study”.
- This page will not have breadcrumbs
- Same page properties as interior page will apply.

Editable Regions

- One new editable region Area of Study that contains a set of table transformations.
- To add a new area of study, user can add an Area of Study Snippet. *(See Snippets section for more information)*
- Those snippets will contain three rows:
 - Image: Image to display as Area of Study
 - Title: Text to be the title of the Area of Study

- Content: Contain a table with rows the following info:
 - Title of head subarea.
 - Content. List of the different subject of this area. To format properly this you will need to apply **Line Style (Area of Study)** from the Styles dropdown in the WYSYGR. Any item may be follow by a list of elements.

```
<p>Computer Science</p>
<ul>
  <li>Programming I</li>
  <li>Programming II</li>
</ul>
```

- Other editable regions as Interior Page will apply. Main content will always display under Area of Science Section

Faculty Directory

- This is a Interior Page with an additional structure “Faculty Directory”.
- This page will not have breadcrumbs
- Same page properties as interior page will apply.

Editable Regions

- One new editable region Faculty Directory that contains a set of table transformations.
- If table is accidentally deleted, user can add an Area of Study

Snippet. *(See Snippets section for more information)*

- Table contain a set of rows with the following columns:
 - Image: Image of the faculty member with 203x204 proportions. Image is automatically crop to be a circle.
 - Name: Name of the faculty member. This may contain a link to other page. This link will apply to both Name and Image. A-Z filter will use second word in the name to filter
 - Department. Department of the faculty member. Department list will be generated from the list of departments in the table, for that reason it is important to spell the department correctly and do not introduce any extra character. When a member belongs to more than one department, editor can list them by adding a break line between them
 - Bio: Contains the bio info of the member. Editor can add any content in here.
 - Office: Text with the office of the faculty member.
 - Phone: Phone of the faculty member. Editor can add any format.
 - E-mail. It may contain a link to the e-mail of the faculty member.
 - Website. It may contain a link to the personal website of the faculty member.
- Other editable regions as Interior Page will apply. Main conten will always display under Faculty Section

Home Page

- This is the standard template for Wenatchee Campus Homepage.

Page properties

- Page Title is the title in the browser.
- Description and keywords will be used as metadata in the head of the page
- Connect Settings contains information about the Connect Section.
 - Display option: Allow user to switch between editable regions or feeds with the information.
 - Facebook Id: This is the id to get the Facebook items
 - Facebook items to display
 - Twitter Handle: Hashtag of twitter to use.
 - Twitter items to display
 - Character per item to display
 - Display order: To display Facebook or Twitter before.
- Video Region: Allow user to select some properties to the video region on the page.
 - Background Image in the container.
 - Video Link: Link to video when clicking play.
 - Icon Image: Icon in of the play button.
 - Text Line 1: Text in the first line
 - Text Line 2: Text in the second line
 - Request a Tour Text: Text to display in the tour link

- Request a Tour Link: Link to request a tour page
- Contact Region: Contact information region of the two campuses
 - Contact Title: Title of the section
 - For each container left and right user will have:
 - Campus Background Image: Image to select as background
 - Campus Title: Title of the Campus. For example, Wenatchee
 - Campus Title Cursive: For example, (Main Campus)
 - Campus Description: Text with description of the campus
 - Address 1: First line of the address.
 - Address 2: Second line of the address
 - Campus Location link: Link to map of the location of the campus
 - Campus Location text: Text of the link to the map of the campus.
 - Link to Facebook. *Icon dissappear if empty*
 - Link to Twitter. *Icon dissappear if empty*
 - Link to Youtube. *Icon dissappear if empty*
 - Link to Instagram. *Icon dissappear if empty*
- LDP Gallery: LDP Gallery type. This does not apply to the LDP-Home gallery that has special format and will always be displayed as a banner. It only applies to other additional galleries that editor may want tot add.

Editable Regions

- LDP-Home: Select a LDP Gallery for this region. *See LDP Gallery for more information*
- Column Left: Allow user to manually add Social Media Items if this option is active. Normally it will contain a content social table transformation. *See snippets for more information*
- Column Center: Allow user to manually add News if this option is active. Normally it will contain a content news table transformation. *See snippets for more information*
- Column Right: Allow user to manually add Events if this option is active. Normally it will contain a content events table transformation. *See snippets for more information*
- Box Left: Allow editor to insert any content in this box.
- Box Right: Allow editor to insert any content in this box. By default, it will contain a table transformatin Choose Excel Graph. *See snippets for more information*
- Slider Region: Allow editor to insert key dates in this containeter. By default, it will contain a table transformatin Slider Key Dates. Each two columns will represent a slider. A slider can have until four different key dates. *See snippets for more information*
- Slogan Region: Allow editor to add any content to this slogan container.

Omak Home Page

- This is the standard template for Omak Campus Homepage.

Page properties

- Page Title is the title in the browser.
- Description and keywords will be used as metadata in the head of the page
- Tabs: Eight tabs that may appear below the main header. Each tab contains:
 - Display Tab: To hide/display tab.
 - Link Title: Title of the tab
 - Link URL: Link to other page
- Bottoms Pannels: Three pannels at the bottom of the page that contain the following information.
 - Image Panel: Background Image in the panel
 - Title Panel: Title of the pannel
 - Description Panel.
 - Link Panel: Link to other page.
- LDP Gallery: LDP Gallery type. This does not apply to the LDP-Home gallery that has special format and will always be displayed as a banner. It only applies to other additional galleries that editor may want tot add.
- News Display Mode: Allow user to switch between manual entry or pulling posts from blog and calendar.

Editable Regions:

- LDP-Home: Select a LDP Gallery for this region. *See LDP Gallery for more information*
- Box Left: Container where editor can add any content.
- Box Right: Container where editor can add any content. By defauly, it contains a two column tabs table transformation with

two feeds. One for news and another for events.

Foundation Home Page

- This is the standard template for Foundation Homepage.

Page properties

- Page Title is the title in the browser.
- Description and keywords will be used as metadata in the head of the page
- Top Buttons: Three panels on the top that will display below LDP Gallery. Each button contains:
 - Icon: Icon on the left of the text.
 - Title: Title of the button
 - Description: Text below the title
 - Link: To another page.
 - Alternative Text for the image in the button.
- Tabs: Eight tabs that may appear below the main header. Each tab contains:
 - Display Tab: To hide/display tab.
 - Link Title: Title of the tab
 - Link URL: Link to other page
- Intro Hero: Image and alternative text on the right side of the “Hero Container”
- Slider Container: Text and link in the submit history button on the testimonial slider
- Bottoms Pannels: Three pannels at the bottom of the page that contain the following information.

- Image Panel: Background Image in the panel
- Title Panel: Title of the pannel
- Description Panel.
- Link Panel: Link to other page.
- LDP Gallery: LDP Gallery type. This does not apply to the LDP-Home gallery that has special format and will always be displayed as a banner. It only applies to other additional galleries that editor may want tot add.

Editable Regions:

- LDP-Home: Select a LDP Gallery for this region. *See LDP Gallery for more information*
- Text Content: First region with text in the “Hero Container”
- Slogan Region: Editable region where editor can add any content.
- Slider region: This region is prepare to set an “slider testimonial”. By default it contains a table transformation with a “Slider Testimonial” where editor can add rows with the image, title, description... *See snippets section for more information*

Athletic Homepage

- This is the homepage template for athletics.

Page properties

- Title News: Title of news section
- Title Score: Title of scores & schedules section

- Title Teams: Title of teams sections.
- News Options: Information that comes from social media feed.
 - Display Option: Option to display facebook feed or editable region with table transformation
 - Facebook Id: For Wenatchee
 - Facebook Link: Link to Wenatchee Facebook Page.
 - Character per item: Number of char in the feed description.

Multiedit Content

- A set of images of boxes to display in team section
- They are two rows. Each row can have 3 or 4 boxes.
 - Top Images Number
 - Bottom Images Number
- Each box will have:
 - Image
 - Image Description
 - Text
 - Link

Editable Regions:

- LDP-Home: Select a LDP Gallery for this region. *See LDP Gallery for more information*
- Left Content: This will be only available if the feed content is set to editable region. It contents a table transformation where user can set manual feeds.
- Right Content: This container is set to edit the scores and

schedules. By default, it contains a list of table transformation with three sections. Each table will correspond to a sport. Editor can add more sports by adding them from the snippets button.

- Sport Title: Title of the sport. For example **WOMENS BASKETBALL** .
- Teams section: Teams, Score Date and Time for each game. Add new rows to add more schedules. Those games will be paginated in group of 3.
- Links: Links to schedule, stats and standings pages.

Section

- Folder Name is the folder name of this section.
- Section Title is the section title that will display in the breadcrumb, navigation side bar...
- File Configuration properties will set the properties for the Index Page which will be an Interior Page (see Interior Page template)
- When creating a section `_nav.inc` and `props.pcf` files will be created where user can define the navigation side menu and properties for this section. ***see side navigation section***

Styles Dropdown

- Path to the file `/_resources/ou/editor/styles.txt`
- This file must follow the structure as noted on our Support Site

- Has the styles for the buttons provided, bold Paragraphs, alignment, styling the lists.

_props.pcf file

- This is a section basis file and contains two properties
- Section Title is the value that used to build out the breadcrumb bar under the background image header.
- Sidebar Title is the value that used to build out the breadcrumb bar under sidebar navigation ***see side navigation section***

Directory Variables

- **header-path** - a root relative path to the header file. Use the **New Header File** to create the header, configure as appropriate. Then define the path. Example: You create a header in the /about section called header.pcf and you would like all the files in that folder to have that header so you would define the **header-path** as **/about/header.shtml**. The extension is ..shtml because it pointing to the published version of the header file. More information about this in the Multi Site documentation.
- **nav-start** define a root relative path to where the *_nav.shtml* for that section exists. Example if you want the navigation from */writing-center/* to be the navigation for */admissions/* on the */admissions/* folder you would set the directory variable of **nav-start** to be */writing-center*. The default is empty of it will

pull the `_nav.shtml` file from the current section/directory you are in.

Side navigation

- This is controlled by a server side script that exists here: `/_resources/script/get-nav.ash` The purpose of the script is to recursively parse the `_nav.shtml` files published to the production server to generate nested navigation structure reflecting the navigation. ***This is a global script and will affect all pages when publishing An initial navigation file is loaded and parsed for `` nodes.***
- Given the `/_nav.inc` starting point:
- `/_nav.inc` contents

```
<li><a href="/index.html">Home</a></li>
<li><a href="/section_a/index.html">About Wenatchee Valle
y College</a></li>
<li><a href="/about.html">About</a></li>
```

The script will simply render out the first link, then see the link to a sub-section on the second link and attempt to load the navigation file for that section.

- `/section_a/_nav.inc` contents

```
<li><a href="/section_a/freshmen_college.html">Why colleg
e?</a></li>
```

```
<li><a href="/section_a/freshmen_ready.html">Get ready</a>
</li>
```

- The script will render out the links in the section_a navigation file and then proceed to the third link of the initial navigation file. This happens recursively resulting in the following output:

```
<ul>
<li><a href="/">Home</a></li>
<li><a href="/section_a/">About Wenatchee Valley College</a></li>
<ul><li><a href="/_zz-test/test-directory1/index.html">ZZ-dir</a></li><li><a href="/_zz-test/test1.html">ZZ-dir1</a></li><li><a href="/_zz-test/test1-1.html"> ZZ-dir1.1</a></li><li><a href="/_zz-test/test1-2.html">ZZ-dir2</a></li><li><a href="/_zz-test/test1-3.html">ZZ-dir3</a></li><li><a href="/_zz-test/test1-4.html">ZZ-dir4</a></li><li><a href="/_zz-test/test1-5.html">ZZ-dir5</a></li></ul>
<li><a href="http://ad-wb-wvc01.wvc.edu/untitled.html">Test Page</a></li>
</ul>
```

Snippets

*/resources/snippets/*There are four categories of snippets:

- Area of Study: Complex table transformation to add area of

studies to Area of Study template.

- Faculty Listing: Complex table transformation to add faculty members to faculty directory template.
- Home Page: Those are table transformations that applied to homepages. They are used to help editors to maintain different components in homepages and they may not work in other regions different to the ones created for.
- Table Transformations: A set of table transformations to insert in any page.

Area of Study Table Transformation

- Each table contain an area of study.
- Image: Image to display as Area of Study
- Title: Text to be the title of the Area of Study
- Content: Contain a table with rows the following info:
 - Title of head subarea.
 - Content. List of the different subject of this area. To format properly this you will need to apply **Line Style (Area of Study)** from the Styles dropdown in the WYSYGR. Any item may be follow by a list of elements.

```
<p>Computer Science</p>
<ul>
  <li>Programming I</li>
  <li>Programming II</li>
</ul>
```

- Add new rows to add new subjects.

Faculty Listing Table Transformation

- Table contain a set of rows with the following columns:
 - Image: Image of the faculty member with 203x204 proportions. Image is automatically crop to be a circle.
 - Name: Name of the faculty member. This may contain a link to other page. This link will apply to both Name and Image. A-Z filter will use second word in the name to filter
 - Department. Department of the faculty member. Department list will be generated from the list of departments in the table, for that reason it is important to spell the department correctly and do not introduce any extra character. When a member belongs to more than one department, editor can list them by adding a break line between them
 - Bio: Contains the bio info of the member. Editor can add any content in here.
 - Office: Text with the office of the faculty member.
 - Phone: Phone of the faculty member. Editor can add any format.
 - E-mail. It may contain a link to the e-mail of the faculty member.
 - Website. It may contain a link to the personal website of the faculty member.

Content Accordion

- Two columns per row. The first column is the heading and the second column is the content of the accordion
- The first column is just the text of the column since it can not contain HTML structure for the accordion to function. The second column you can place any content and style it with the toolbar.

Content Block Text

- Content box with an style
- Title to select title of the box
- Content. Any content than user wants to apply to box.
- Style. To apply to the box. Select text on the field and apply a style from **style dropdown** . Please select one callout style

Content Block Image

- Content box with a background image
- Title to select title of the box
- Content. Any content than user wants to apply to box.
- Image. To apply as background to the box.

Content Accordion

- Two columns per row. The first column is the heading and the

second column is the content of the tab

- The first column is just the text of the column since it can not contain HTML structure for the tab to function. The second column you can place any content and style it with the toolbar.
- First column is always opened by default

Tooltip

- Link text. To select the text of the link
- Tooltip Text: Text to display when user hover on the links

Fancybox

- Trigger text: Text to display in the link
- Box title: Title of the pop up
- Box Content: Content inside the pop up.
- Date & Time: To displaye at the bottom of the pop up

Widget blocks

- Those are widgets to display in the sidebar containter
- Title to select title of the box
- Content. Any content than user wants to apply to box.
- Style. To apply to the box. Select text on the field and apply a style from **style dropdown**. Please select one widget style

LDP

Galleries

- Use the LDP galleries to create an popup image rotator.
- This gallery structure was provided to OmniUpdate by mStoner.
- This gallery can be used anywhere.
- When creating the gallery you may add as many photos you would like but the only field that is used is the description field of the first image. This description field will become the caption of the gallery. All other fields will be ignored.
- On placing the asset on the page, you will see the first image with the caption below it but upon clicking on it, it will bring up a gallery with the images, that can be viewed.

Forms

- The regular LDP forms with styling updates to fit the design of the university branding

Other Assets

Comments

- Place the **Disqus Comments** Asset on any page after replacing the following line:
 - `var shortname = 'EXAMPLE'` ; with the shortname of the university, it has to be in quotes

Social Media - Assets to insert social media. They are three different size. Small, medium or large. User can copy and edit the links and social media icons to display

```
<ul class="social sm">
<li class="facebook"><a href="#"></a></li>
<li class="vimeo"><a href="#"><i class="fa fa-vimeo-square"></i></a></li>
<li class="twitter"><a href="#"></a></li>
<li class="instagram"><a href="#"></a></li>
<li class="youtube"><a href="#"></a></li>
<li class="linked-in"><a href="#"></a></li>
<li class="google-plus"><a href="#"></a></li>
<li class="tumblr"><a href="#"></a></li>
<li class="sellsy"><a href="#"></a></li>
<li class="stack-overflow"><a href="#"></a></li>
</ul>
```

OU SEARCH

- Location *_/resources/search/*
- User can edit the css of the results.

- PCF Location is location in /search/index.pcf.
- This is a unique page with interior page format.
- This page can be copy to a new folder but then, editor will need to modify action in:
 - *_/resources/includes/search.inc*
 - *_/resources/includes/mobile-search.inc*
- Pages will be automatically index by OU Search
- Login in <https://ousearch.omniupdate.com/teaxis/dowalk/>? with your user to modify search parametric options.

OU BLOGS

Outline

A blog consists of three main parts, listing pages/assets, post files, and scripts. The blog listing page is a PCF which OUCampus outputs as an ASPX page with function calls to internal scripts, based on page parameters. See [Blog Listing Template](#) below for specific page params. OUCampus holds each post as a PCF, XSL outputs two types of files, content (HTML) and data (XML). This data is taken upon page creation (ou-blog-post.tcf) and manually input via page properties and multi-edit. Scripts are XSL and C#. XSL handles the XML output (.blog extension) and to a lesser extent the HTML - it's mostly just boring content. Blog Assets are handled with a template-match, see [Assets](#) for more detail. C# is used for server side query string based pagination, splitting up the full list. C# does the grunt work by crawling a given directory structure for XML post files, and displaying content with a

bunch of different functions, more below.

Structure

This is only a suggested structure, change it to suit your implementation.

- PCF
 - /blog/index.pcf
 - /blog/post/blog-post.pcf
- XSL
 - /_resources/xsl/ou-blog/blog-functions.xsl
 - /_resources/xsl/ou-blog/listing.xsl
 - /_resources/xsl/ou-blog/post-xml.xsl
 - /_resources/xsl/ou-blog/post.xsl
- C#
 - /_resources/asp_scripts/blog/get-blog-rss.ashx
 - /App_Code/OUBlog.cs

Templates

- Post page - news-post.tcf | news-post.tmpl

PCF options

- _props.pcf

- All sidebar widgets (featured posts, categories, tag cloud, subscribe, social media, upcoming events) can be enabled/disabled for the entire blog here.
- School social media links are entered in this file.
- `Listing page parameters`
 - "Blog Title" the title of the blog (broken down into "Left Title Text" and "Right Title Text" to accommodate styling.
 - "Title Description" the text below Blog Title.
 - "Year Filter" default is empty - all years.
 - "Posts per page" limits the number of posts shown before pagination.
- Post page
 - Parameters
 - "Share Buttons" enable share buttons
 - "Action Buttons" enable action buttons (print/download)
 - "Post Tags" display post tags
 - Multi-Edit (stores post data)
 - TCF Variables: title, description, image, president post option, featured post option.
 - PCF Multi-Edit (not passed by TCF): display toggle and image.
- Tags

- Tags are handled with OUCampus Tag Management and are entered via page Properties.
- Tags are a tag collection entitled “news tag cloud” and can be removed/added to this collection via Setup > Tags.
- Categories
 - Categories are a separate tag collection entitled “news categories” and be removed/added to this collection via Setup > Tags.

Assets

Are meant to compliment the main listing of blog posts, with list of tags/categories, or recent posts, etc. They can be dropped on to any blog page and function.

Each asset is a single blog node with attributes defining it’s output: **type**, **dir**, and **limit**. Example:

```
<blog type="tags" dir="/" limit="0"/>
```

- Dir tells the C# scripts where to begin looking for (*.blog*) post files which are XML.
- Type tells the XSL which C# display function to output.
- Limit is the number of items to display.
- Optional **tags** attribute for type=“byTag” as CSV, eg.

tags="consetetur, dolor sit amet"

A template match converts this node as help text, for staging, and a C# statement for production. Each output is a simple `` list.

Types available: (all gather posts from defined *dir*, with respect to *limit*)

1. "tags" - posts by tags found in posts.
2. "recent" - posts by pubDate.
3. "featured" - posts marked as *featured*.
4. "related" - for **posts only**, takes current PCF *tags* and displays all posts with matching tag.

Example: you want to display the five most recent posts from the academics blog:

```
<blog type="recent" dir="/academics/blog/" limit="5"/>
```

Outputs:

```
<%  
Response.Write(OUK.Blog.DisplayRecentPosts("/academics/bl  
og/", "5"));  
%>
```

Scripts

Each post PCF is processed by XSL to output only the essential data as XML. C# scripts combine to output different lists of posts, featured, recent, related, etc.

XSL

pubDate = dateTime

To prep the date variable “pubDate” an XSL function (*ou:toDateTime*) converts the date given by the OU property *type=“datetime”* to a valid XSL dateTime object. This allows us to use XSL functions for dateTimes, namely *format-dateTime()*.

Note: post.tpl uses the OU System Variable \$CurrentDate to output “Friday, January 30, 2015 3:17:33 PM PST”. However accessing and saving via parameters converts this date format to “01/30/2015 03:17:33 PM”. (*ou:toDateTime*) accounts for both formats.

Once we have a dateTime object we use *ou:pubDate()* to output a valid “RFC822” pubDate string. This is required for a [valid RSS string](#) and can be interpreted by C# easily. There are also smaller functions available to convert dates and times to friendly formats, *ou:displayLongDate()*, *ou:dayOfWeekAbbrevEn()*, etc.

C# Classes in OUBlog.cs and the RSS WebHandler get-blog-rss.ashx

Your XSL should only be concerned with calling the appropriate Display functions within class Blog.

OUC.Blog

Display functions take parameters from XSL to crawl defined directories and extract data from posts XMLs.

OUC.BlogPosts

Used to maintain a total count of available blog posts necessary for pagination

OUC.BlogPost

A blog post data model/view class initialized by the root post node of the blog files on the server. Blog posts can render themselves depending on the passed in conditions.

get-blog-rss.ashx

Output a valid RSS feed based of URL parameters. By default, will return all blog posts. Valid URL params:

- “dir” - The directory where the script should begin crawling for post files, defaults to blog root.
- “limit” - Maximum number of posts returned, default “” returns all.
- “tags” - Only return posts from specific tags, default “” no

categorization.

- “featured” - “true” to return only featured posts, default “false”.
- “year” - return posts from a specific year, default “” return all years.

OU Mashup

- All the script files for the mashup are located in *_/resources/social/* which has the CSS and JS as well as the social media files that do the calls to the Social Media API's.
- Mashup page will have the same forma as other interior pages but with mashup content
- Config File in *_/social/oumashup* needs to be modify and publish to get mashup information
- Multiedit region with social media settings.
- Twitter:
 - Show: to display/hide those feeds
 - Handle: Hashtag page
 - Items to display
- Facebook:

- Show: to display/hide those feeds
 - User/Page ID: Facebook info
 - Items to display
- Youtube:
 - Show: to display/hide those feeds
 - Channel name
 - Items to display
- Instagram:
 - Show: to display/hide those feeds
 - User/Page ID: Facebook info
 - Items to display
- Example page: <http://ad-wb-wvc01.wvc.edu/social/index.html>
- They can rename/move the page to whatever they would like.
- Inside of the page source of OU Campus you will see the script references These should only be touched if those script files are moved. `<headcode><script type="text/javascript" src="_oumashup.js"></script></headcode>` -Some of the social media items will not show up inside of OU Campus due to the Social Media API's so make to view the published version to see the final result.