CrownPeak ™

MOVING PORTED TEMPLATES TO EXISTING CROWNPEAK CMS CUSTOMER: PROCESS

Date: 10.19.2012

<mark>Second Draft Version</mark>

PROCESS V2

WAVE 0

1. Copy Libraries (and all shared/global standalone code) from the development to the production instance.

2. Create a new C# template on the production site.

3. Copy all the contents of an existing production template, including all template files (that has a development equivalent) to the newly created template. *(Example: Homepage template)*

   a. Clone the Output.asp file and rename it as Output_stage.asp

   b. Rename the current Output.asp to Output_live.asp

   c. Create a new Output.aspx file that will contain the logic of which file will be used depending on the workflow state. (This functionality can be on a library function so it can be modified in a central location).

   d. Rename the Input.asp template file to Input_old.asp

4. Copy *(Replace)* the **Input.aspx** file from the development instance to the newly created production template. *(Example: Input.aspx file from the development Homepage template to the newly created production template).*

5. Do a comparison between the **Output.asp** from the development instance to the **Output_stage.asp** of the production site (of the newly created template):

   a. **If they match** (with the exception of the code that enables ICE), copy the **Output.asp** file from the development instance to the newly created production template. *(Example: Input.aspx file from the development Homepage template to the newly created production template).*

   b. **If they don't match**, we need to keep the original file intact and manually start re-tagging the individual fields to enable ICE.

6. Clone an asset that is using the same template from which the contents of the production template were copied *(Example: Original production Homepage template).*

   a. Change the current template of the cloned asset to point to the new production template.

7. Verify that the functionality of the new production template matches the functionality of the original production template:

   a. Edit and save the content of the new asset in both Classic and Volte.

   b. Verify the look and feel of both templates in both Classic and Volte.

8. Publish the test asset to Stage and review that all the functionality is in place.

9. Change all the assets in the production instance that are using the current production template *(Example: Homepage)* to the newly created template (from step #2) and the alternate workflow. (This alternate workflow has publishing disabled on Live).

10. Re-publish a few assets to Stage and review that all the functionality is still in place and working correctly.

11. Delete the test asset.

12. Once the customer has reviewed and approved the template. Change the workflow of all the assets that are using this particular template back to the original workflow.

Subsequent Waves: Repeat steps #2 through #12 for all the remaining templates.

13. Once all the templates are in place, publish the entire site to STAGE and review that all functionality is in place.

14. Once approved,

    a. The Ouput library function will point the assets to the new location right location.

    b. The Output.asp will be renamed to Output_live.asp to Ouput_old.asp and Output_stage to Output_live.asp.