

Prashank Singh's Blog: New CA PPM UI & How to overcome its challenges

Posted by [Prashank Singh](#) Jun 14, 2018

With increasing adaptability towards new UI, a lot of users are also bumping into the issues in New UI because of the limitations it has. Where New UI seems to be promising to be used for macro management, we still need to grill into classic PPM for the micro management. However, blending these two UI together might not prove to be the best solution. The reason is the level of expectations by the users for the new UI as per the customization in classic UI in their organization. Also, few elements of new UI are not mature as they are expected to improve by every release.

This raises a question as what should be an optimal approach towards UI release. The definition of optimal varies from organization to organization and on the module being rolled out and the customizations being done to classic.

One good solution would be to follow an iterative methodology, where we roll out some features to stakeholders & limited users and based on the feedback, revisit previous steps with some improvement and finally making it available for wider audience.

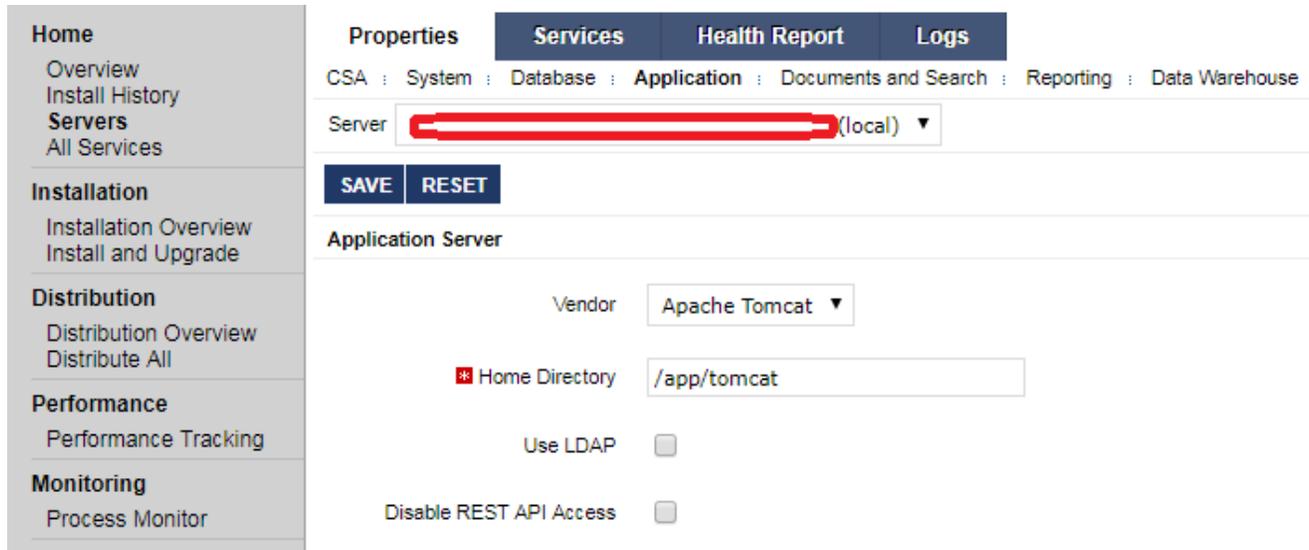
Following steps can be helpful while rolling out the New UI.

Step 1a: Access Control for Module control

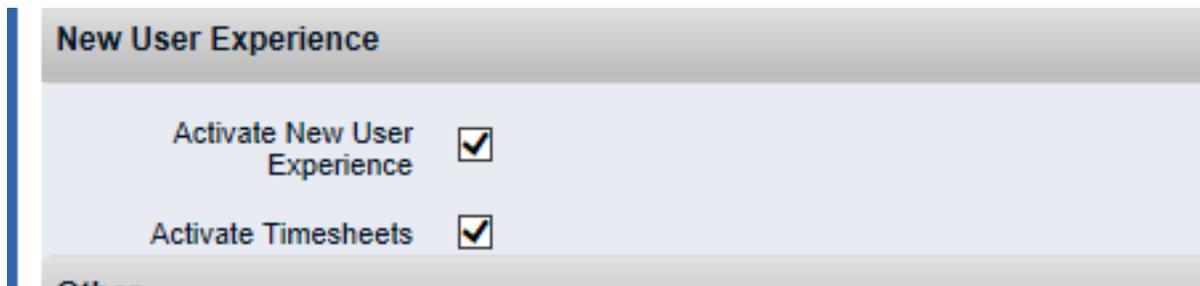
Mature way would be not to rush into exposing all modules to wider audience as this would lead to users facing difficulty with adaptability. For any of the mature implementations in classic, the user might tend to compare for more functionality, which is not present in New UI yet. That might be in pipeline on new UI for future release. Now that we understand as all modules all together might not be an ideal solution while launching new UI, next big thought that arises is which module to pick and how to hide other module links. This leads us to Access Control provided by CA for controlling all these modules, one must carefully go through these.

For basic setup following list might be helpful:

- 1) Enable REST API from CSA



2) Enable Timesheet UI



3) Enable Timesheet UI individually by restricting access via Classic CA PPM.

4) Access Control for Project, Staffing and other elements as per the need and feedback from limited users.

5) Creation of Menu Link to switch between both the UIs.

Step1b: Timesheet module, might be your savior of the day

Timesheet module will be nice to start with as this module is least customized as far as user interface go and any other process/java validation stuff will work almost as similar as it works in classic. So, less modification and quick to start for wider audience to accumulate user responses on the behavior of new UI.

Step2: Kicking other modules, utilizing them carefully

This step requires a detailed analysis and understanding of each module and try to bridge gaps between any missing functionality which could be there in classic as OOTB or as a custom item specific to your organization. One of the way would be to use links in the project module for solving any issues around dynamic html page, Classic CA PPM link or Jasper report in case of dashboard like view.

This will ensure you that any missing functionality which you have built in classic is one click away. Also you need to make sure to give a way to link it back to either classic r to new UI.

Step3: CSS Can be magical to blend both UI's

Changes to your classic UI to look a bit similar to that of new UI by changing the CSS in classic can work really magical.

This will not leave your user in the impression of using two different systems for PPM and the switching between the UIs will not be hurdle to work with.

Step4: Setting Right Expectation for User

With each release, set context on what to expect in new UI and how to use the modules. Giving them the idea of what they might get in the next release would make their perception positive about the New UI.

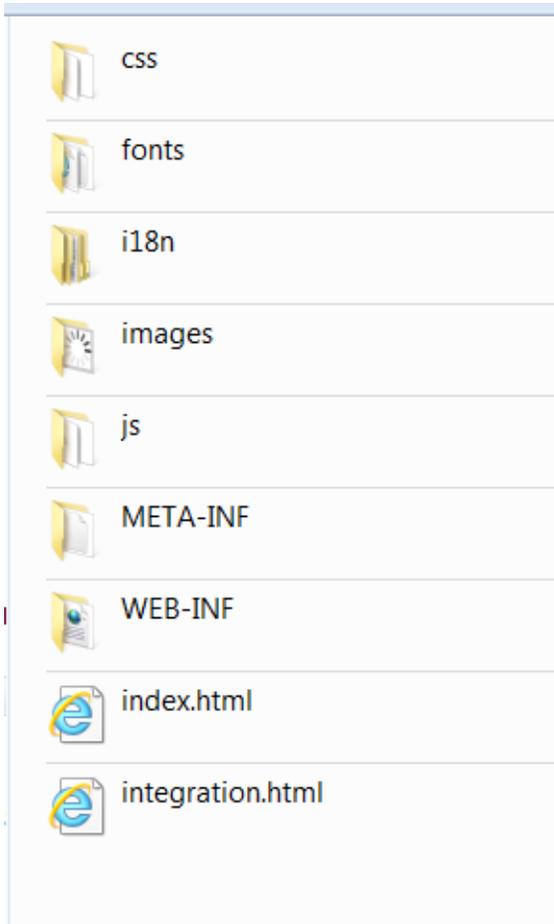
Step5: The Last one

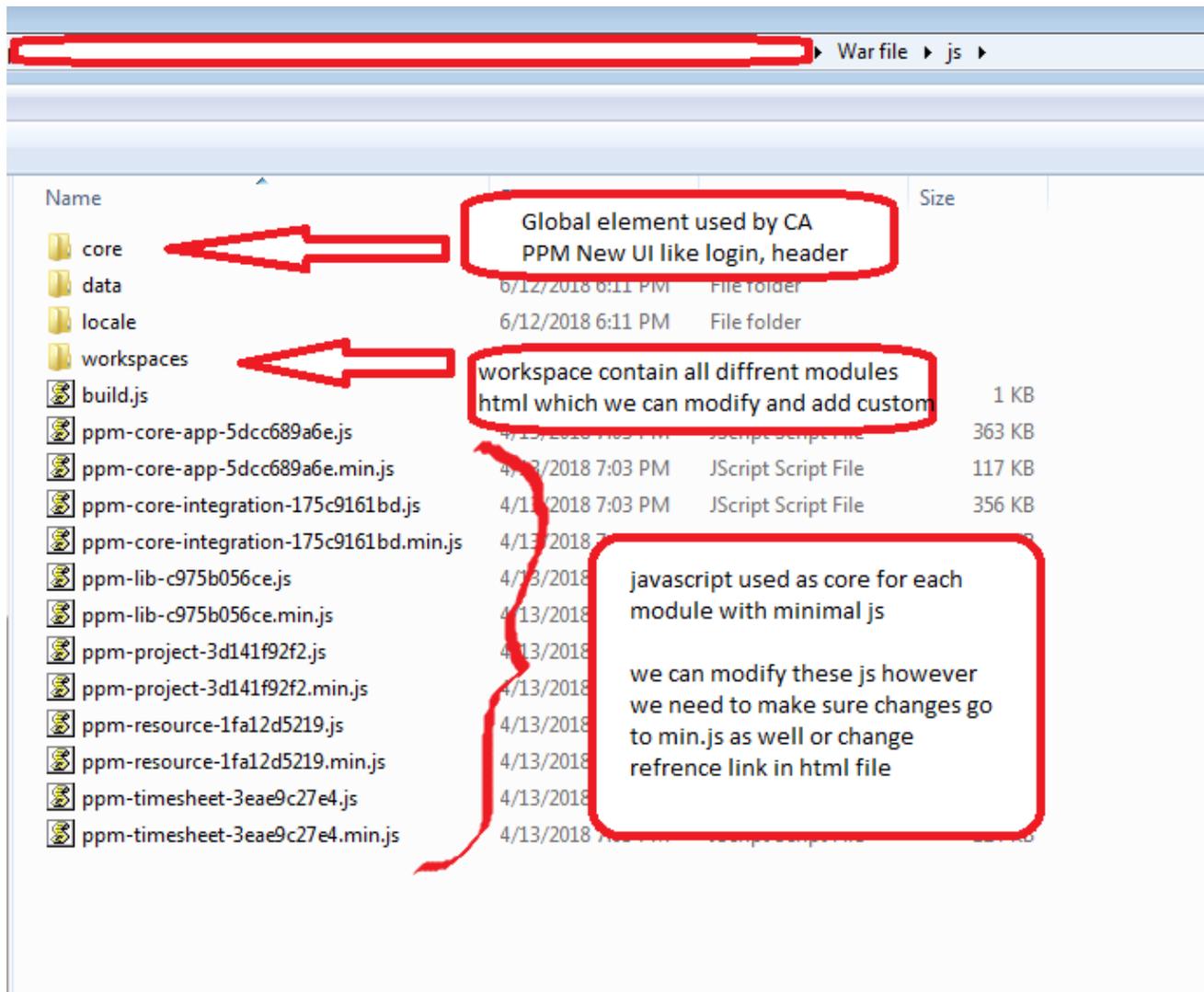
Direct modification of UI might be a powerful approach to achieve almost anything to mitigate the impact of using different UIs. However, **you need to be aware over this as any modification directly to the .war file used to load new UI is not supported by CA.** So, this should be your last step and be avoided.

However, if you are still looking for change then below steps for changing a **User Value 1 label** might be helpful.

For changing label "**User value 1**" in new UI, the file and its path are as below:

- 1) First you have to identify a .war file placed in CA PPM server folder, war file name: **ppm-ux.war**
War file location: %CAPPHome_Path%\META-INF\ui\wars
- 2) Copy and extract this war file in your machine. Below is a sample of extract for War file





3) Look for a localization file path as below:

ppm-ux\i18n\en\timesheet.json

4) You need to change the column definition for the label "User Value 1" which you want to change, will look as below:

"user_lov1": "Modified User Value 1",

5) Then make a .war file again with the modifications included. Then you need to redeploy all the services. You will be able to see the new changed label for the field.

One challenge that I have faced while updating .war file is minimized file generation for JavaScript. For any major change in JavaScript for any module, we need to generate minimized file and that needs to be compatible with CA's. For that you must decide which tool to use to extract and generate the .war file. To avoid that scenario, you can change script path from min.js file to relevant .js file in integration.html.

Warning again: This is not supported by CA.

Hope this might have helped few in achieving transition to new UI.

Prashank Singh's Blog: New CA PPM UI & How to overcome its challenges

Don't forget to share you challenges and the way you took to overcome those.

1066 Views Tags: ca ppm, new ui, 15.x, #ca ppm new ui theme, ca ppm 15.4, ppm approach