ProjectFlow version 1.1 - System Configuration Tool

Contents

[ProjectFlow Custom References Update 2](#_Toc480206860)

[Overview 2](#_Toc480206861)

[When to Use the Tool 2](#_Toc480206862)

[What Does the Tool Update 2](#_Toc480206863)

[WFlowDLLCache Folders 3](#_Toc480206864)

[Updating Workflow Instances 5](#_Toc480206865)

[Updating Workflow Definitions 6](#_Toc480206866)

# ProjectFlow Custom References Update

## Overview

Pre-upgrade checklist will now include the Implementation Team for reviewing the workflows and determining if additional work will be required by the Implementation Team to ensure the running and new workflows will be successful.

### When to Use the Tool

Updating the custom code is only done under two circumstances, it is not part of the standard upgrade process:

1. Development included instructions in the release notes that a specific ProjectFlow version now requires DLL updates, for example:
* If customer is running a build >= ProjectDox.Web.API 2.1.1.6, ProjectDox.Web.UI 1.1.1.7 then there is no requirement to update the references
* The DLLs may vary and will be explicitly listed for the release
1. The pre-upgrade meeting alerted the Implementation Team to take action following an upgrade. The cause may be integration, custom formlets or custom code are affected by the upgrade.

### What Does the Tool Update

When the Update References button is clicked, the system will get all the WFlowScriptID’s from all the workflow definitions, then use those WFlowScriptID’s to get all the WFlowInstance records which have a custom code scripts. The query will include only those workflow instances with a state of:

|  |  |
| --- | --- |
| **WFlowInstanceStateName** | **WFlowActivityStateID** |
| Active | 1 |
| Paused | 3 |
| Error | 4 |
| Pending | 5 |
| Executing | 6 |

Upgrading the active workflow definitions is accomplished in the workflow Designer via the ProjectDox User interface. The systemconfig.aspx has no effect on the workflow definitions.

* Each active workflow definition must be opened individually and each Activity with custom code requires an action.

### WFlowDLLCache Folders

There are three types of folders in the WFlowDLLCache folder, Editor, Config, and Workflow Instance Custom Script.



1. The Editor Folder contains the DLLs for the custom code scripts in the workflow definitions. The Editor folder contains the WFlowScriptID as the name. For instance, a workflow definition may have 3 custom code scripts on 3 different activities, they would then have 3 folders under Editor, all with the WFlowScriptID stored in the database for each script, such as:
* \WFlowDLLCache\Editor\44
* \WFlowDLLCache\Editor\41
* \WFlowDLLCache\Editor\39

 

1. Workflow Instance Custom Script Folders - Each activity on the designer that has a custom code script will have its own folder. The list of folders are numbered according to the WFlowInstanceID that consumes them. The folders are named with the WFlowInstanceID\WFlowActivityID\WFlowScriptID. For instance, if there was a workflow instance with a WFlowInstanceID = 1107 and it had an activity with a WFlowActivityID = 30700 and it consumed a custom code script with a WFlowScriptID = 43, the path would look like this:
	* \WFlowDLLCache\1107\30700\43



1. The Config folder is not used and will be removed in the future. The configuration file data is kept in the accompanying app.config file such as ProjectFlowScript\_2065.dll.config in the same folder as the DLLs.



## Updating Workflow Instances

After the environment is upgraded, specifically the ProjectFlow.Web.UI, the systemconfig.aspx page is launched. The ProjectFlow Custom References Update section provides a tool to update the workflow instances.

You **MUST** stop the ProjectDox Utility Service prior to updating the references, if the Utility Service is currently running, logon to the Application Server and stop the service and set it to Manual.

The DLL files listed in the release notes will be selected for update in the workflow instance custom script folders. Each custom code script for each activity in all the instances that have custom code, will have their own folder.



1. The ProjectFlow Custom References Updates tool will copy the latest DLLs in the ProjectFlow.Web.UI\bin to the custom script subfolders of the shared WFlowDLLCache
	1. Force Update checkbox is required to update the running instances. It would be uncommon to uncheck it
	2. Force Update will copy the DLLs to the workflow instance folders, if the files do not already exist
	3. Force Update will overwrite the DLLs if they already exist in the workflow instance folders

## Updating Workflow Definitions

Only active workflow definitions require refresh, recompile, and save. There is no need to refresh inactive workflow definitions. This script can be executed to pinpoint the workflow definitions that contain custom code. If the customer has a few active definitions, the Installer might choose to do this manually.

SELECT WFlowInstances.InstanceName, DllPath

FROM WFlowInstances INNER JOIN

WFlowActivities ON WFlowInstances.WFlowInstanceID = WFlowActivities.WFlowInstanceID INNER JOIN

WFlowScripts ON WFlowActivities.WFlowActivityID = WFlowScripts.WFlowActivityID

WHERE WFlowScripts.WFlowActivityID IS NOT NULL

1. Open the ProjectFlow Administration window and select an active workflow definition whose name is returned in the above query. Click the Edit icon to open the workflow in the Workflow Designer.
2. Select the Activity with the Gears then click on the Gears icon.
 
3. Click Refresh Custom References, Compile, then Save.



1. Repeat this for all custom references in all active workflow definitions.