**MBD Update for EPM Document on Release**

**Date: 14th May 2024**

**Developed By: PLX WORX PRIVATE LIMITED**

**1. Windchill and Dedicated Workspace Integration**

**1.1 Description:**

This document explains how Windchill integrates with a dedicated workspace and client server. When there is a change in Windchill (Change Notice) involving EPM Document, the resulting objects will be added to a dedicated workspace and trigger the CAD customization code on CAD worker server via a socket server connection. The exe on CAD server will asynchronously update the model and drawings with the change table and check in back to Windchill.

**1.2 Files used:**

1.ServiceClass.java

2. Change Notice.txt (code to be pasted in method robot of CN workflow)

3.CadCustomization.properties

4. Change notice 2.txt

**1.3 File path:**

<WT\_HOME>/ src/ext/nimr/MBDtableUpdate/TriggerCADexe.java

<WT\_HOME>/codebase/ext/nimr/MBDtableUpdate/CadCustomization.properties

<WT\_HOME>/codebase/ext/nimr/MBDtableUpdate/CheckReminingResultingObject.java

**1.4 Command to compile java class:**

ant –f bin/tools.xml class -Dclass.includes=ext/nimr/MBDtableUpdate/TriggerCADexe.java

ant –f bin/tools.xml class -Dclass.includes=ext/nimr/MBDtableUpdate/CheckReminingResultingObject.java

**2. Deployment Steps :**

**Step 1:** Add Resulting Objects to Workspace (**Workflow Code**):

* Add the below code in expression robot of dedicated change notice which includes the trigger point of TriggerCADexe.java methods. Same has been put in Change Notice.txt.

|  |
| --- |
| String requestDataString = "";  String Str = "Test";  try { wt.epm.EPMDocument epmDoc = null;  wt.fc.QueryResult qr = wt.change2.ChangeHelper2.service  .getChangeablesAfter((wt.change2.WTChangeOrder2) primaryBusinessObject);  while (qr.hasMoreElements()) {  wt.fc.Persistable pers = (wt.fc.Persistable) qr.nextElement();  if (pers instanceof wt.epm.EPMDocument) {  epmDoc = (wt.epm.EPMDocument) pers;  String DocType = epmDoc.getDocType().toString();  if (DocType.equalsIgnoreCase("CADDRAWING")) {  requestDataString += epmDoc.getCADName() + "$";  System.out.println("## requestDataString :-" + requestDataString);  ext.CadCustomization.ServiceClass.addToWorkspace(pers, Str);// ServiceClass method  }}}  ext.CadCustomization.ServiceClass.sendToServer(requestDataString);// ServiceClass method  } catch (Exception e) { } |

Change notice2.txt code put in conditional robot:-

boolean containsYes=false;

boolean inWorkspace=false;

java.util.ArrayList<String> arrList= new java.util.ArrayList<String>();

try {

wt.epm.EPMDocument epmDoc = null;

wt.fc.QueryResult qr = wt.change2.ChangeHelper2.service

.getChangeablesAfter((wt.change2.WTChangeOrder2) primaryBusinessObject);

String found=null;

while (qr.hasMoreElements()) {

wt.fc.Persistable pers = (wt.fc.Persistable) qr.nextElement();

System.out.println(("@@@ ") + pers.getClass().getName());

if (pers instanceof wt.epm.EPMDocument && !inWorkspace) {

inWorkspace= ext.CadCustomization.CheckReminingResultingObject.CheckReminingObject(pers);

}}

if (inWorkspace) {

System.out.println("The ArrayList contains 'yes'.");

result="yes";

} else {

System.out.println("The ArrayList does not contain 'yes'.");

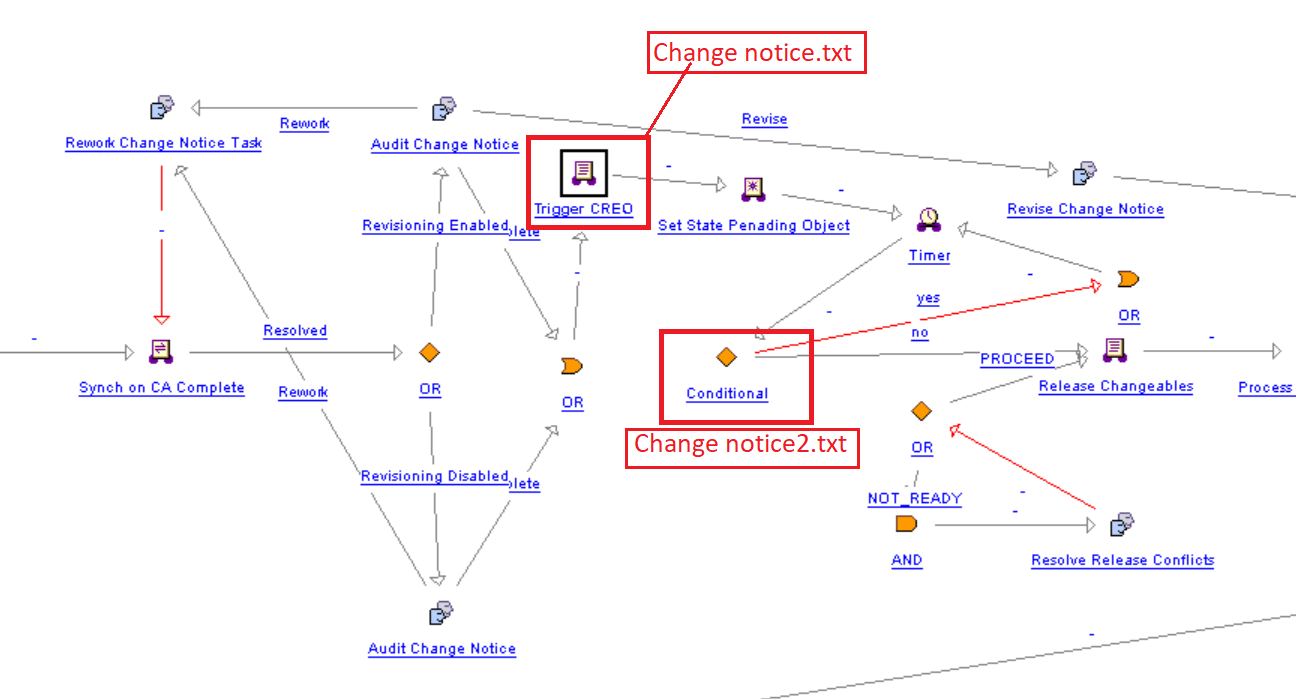
result="no";

}}

catch (Exception e) {

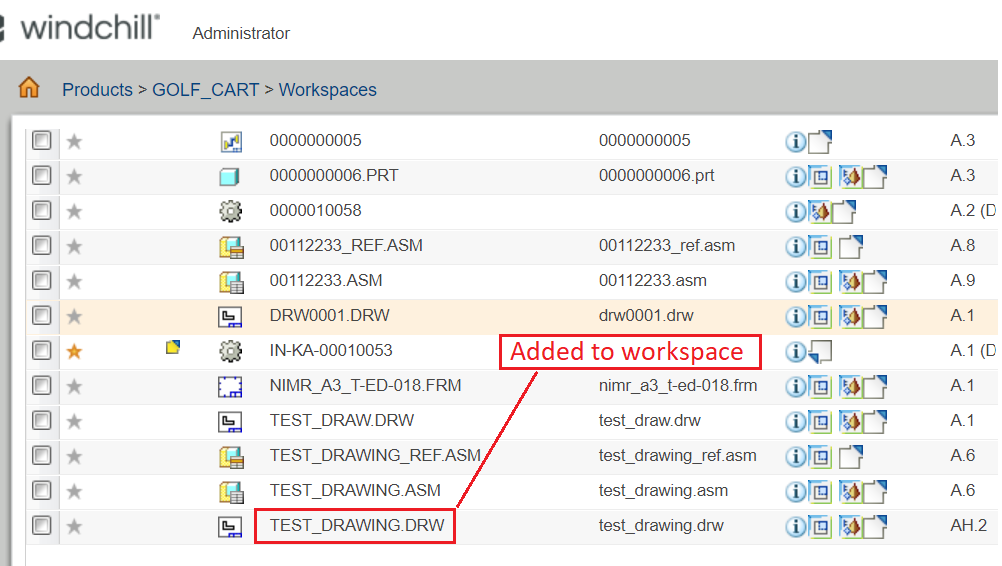
}

* Workflow snippet:-



**Step 2:**

* Create a dedicated workspace to manage CAD table update activities.
* Put workspace name, Windchill CAD worker server name, and port number in CadCustomization.properties file.
* Compile the ServiceClass.java with command: - ant –f bin/tools.xml class –Dclass.includes=ext/ RemoteServer /ServiceClass.java
* After change notice is resolved, check latest version of resulting objects are added to your dedicated workspace



* Incorporate functionality in the Java class to initiate communication with the socket server for client-server interaction and send CAD file names to client.

**Step 3:** CAD EXE file details to trigger once request is received from Windchill.

* Implemented a socket server communication between Windchill and the client server.
* Put port number in CadCustomization.properties file & .exe path mention in environment variable.
* Configure the java file to send the file name of the added EPMDocument to the client server.
* Ensure both windchill server and CAD worker server can ping to each other.

**Step 4:** Create .exe file

1. Download Launch4j: Get it from the Launch4j website and install it.
2. Prepare Your Java Code: Make sure you have your Java code ready and compile it into a .class file.
3. Create a JAR File: Package your Java code and its dependencies into a JAR file.
4. Open Launch4j: Launch the Launch4j program.
5. Configure Launch4j:
6. Set the "Output File" to specify where the .exe file will be saved.
7. Select the JAR file you created earlier.
8. Build the Executable: Click the "Build Wrapper" button in Launch4j.

* Test the generated .exe file to ensure proper functionality.
* Trigger the .exe whenever restart the system.

**3. Deployment on CAD Worker Server:**

**Description:**

The Automatic Parameter Updating on Model Family Table simplifies parameter management in both Drawing and Assembly models. Upon execution of the .exe, it retrieves Creo from system environmental variables. Essential data including Windchill Browser username, password, workspace name and server name are stored in designated folders. Our program reads these details from the folder to update the table. Subsequently, it saves, uploads, and checks in the updated model in the Windchill workspace.

**File we Use**

⦁ Change\_recoed\_block.sym.1

⦁ mbdtable.exe

⦁ mbdtable.lib

⦁ mbdtable.pdb

⦁ mbdtable.exp

⦁ Log(Folder)

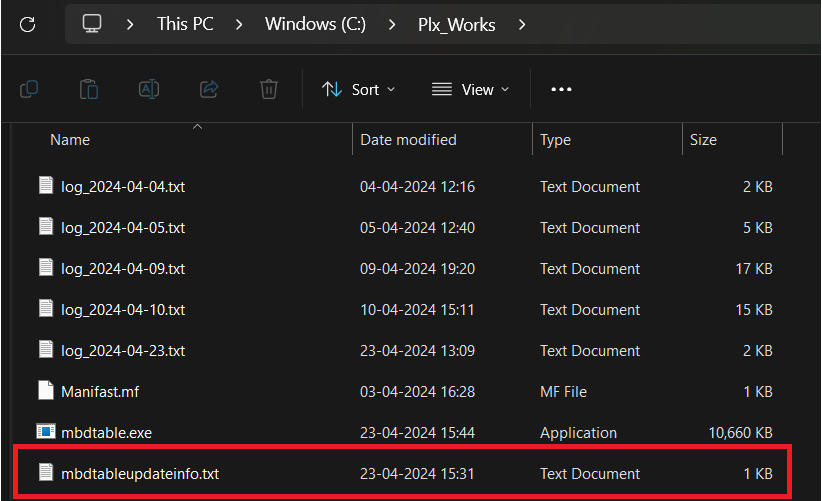
⦁ mbdtableupdateinfo.txt

**Input Details**

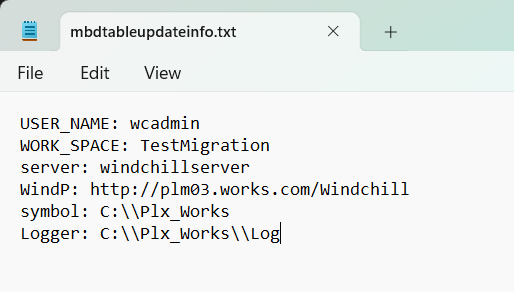
In this text file, we need to store the data such as

* Windchill Username.
* Windchill URL.
* Workspace Name.
* Server Name.

**This text file should be stored in the Executable Directory Path**.

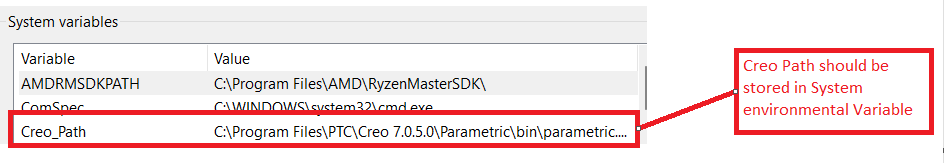


**We shouldn’t change the input names it should be as it is.**

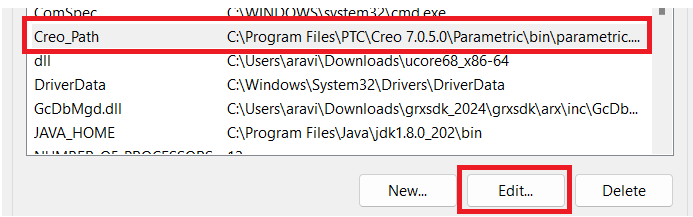


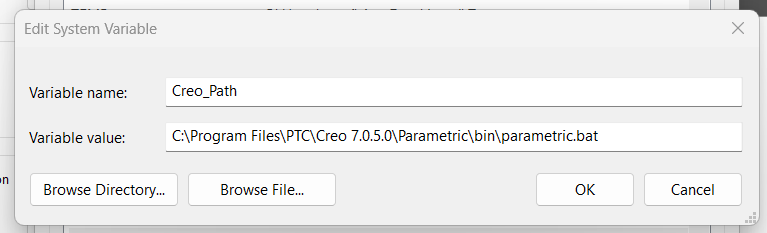
**Creo Path**

Our tool will read the Creo path from the system environmental variable using the given name "Creo\_Path."



Select the path and click the Edit Button.

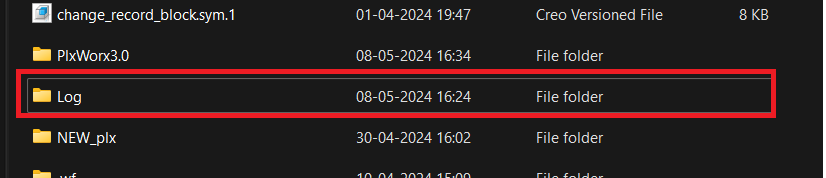


Here we can see the Full path.

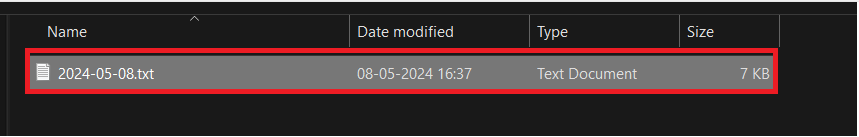
After triggering the .exe, Creo will open, update the model, save the updated model, and upload it to the Windchill Workspace and finally it will delete the updated models from the workspace.

**Model details and their names.**

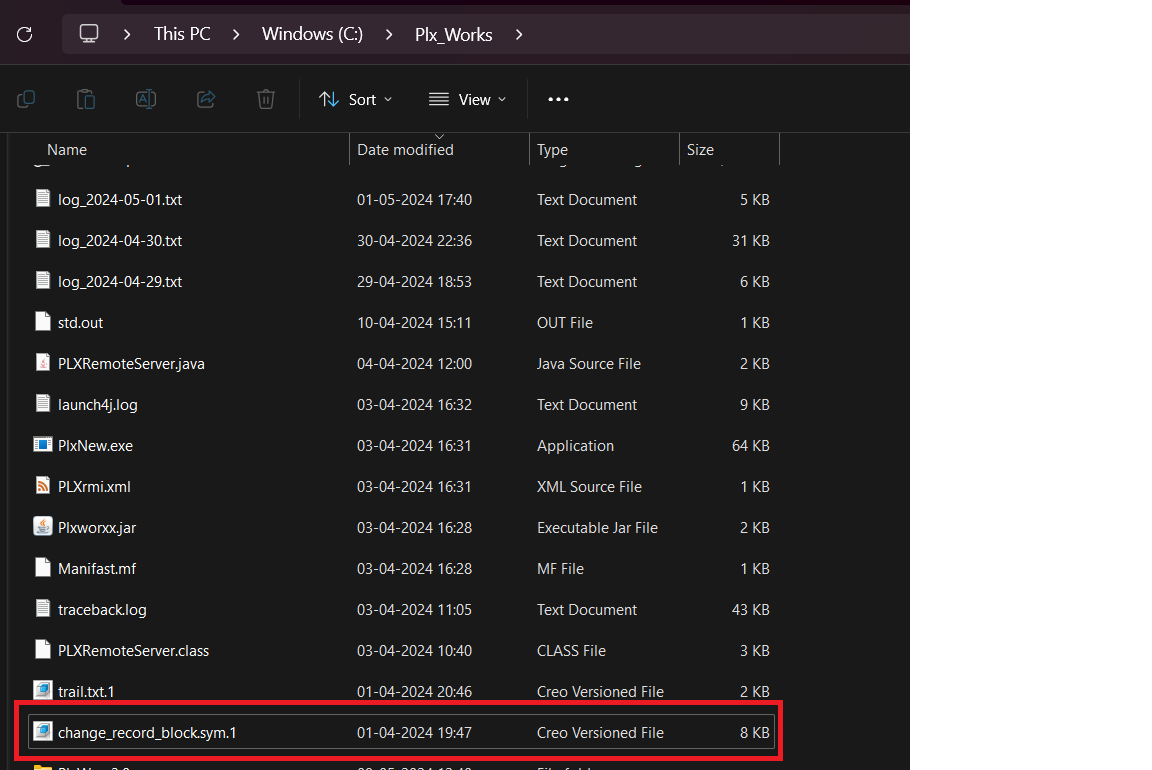
Log file will be created in .exe directory



Once you open this .txt file there we can see the status.



**Table Update for Assembly Model**



This “change\_record\_block.sym.1” file important so where ever you save this file just copy the file path and paste it here.

