Converting SmartPlant PID for iTwin

2020-08-24

Contents

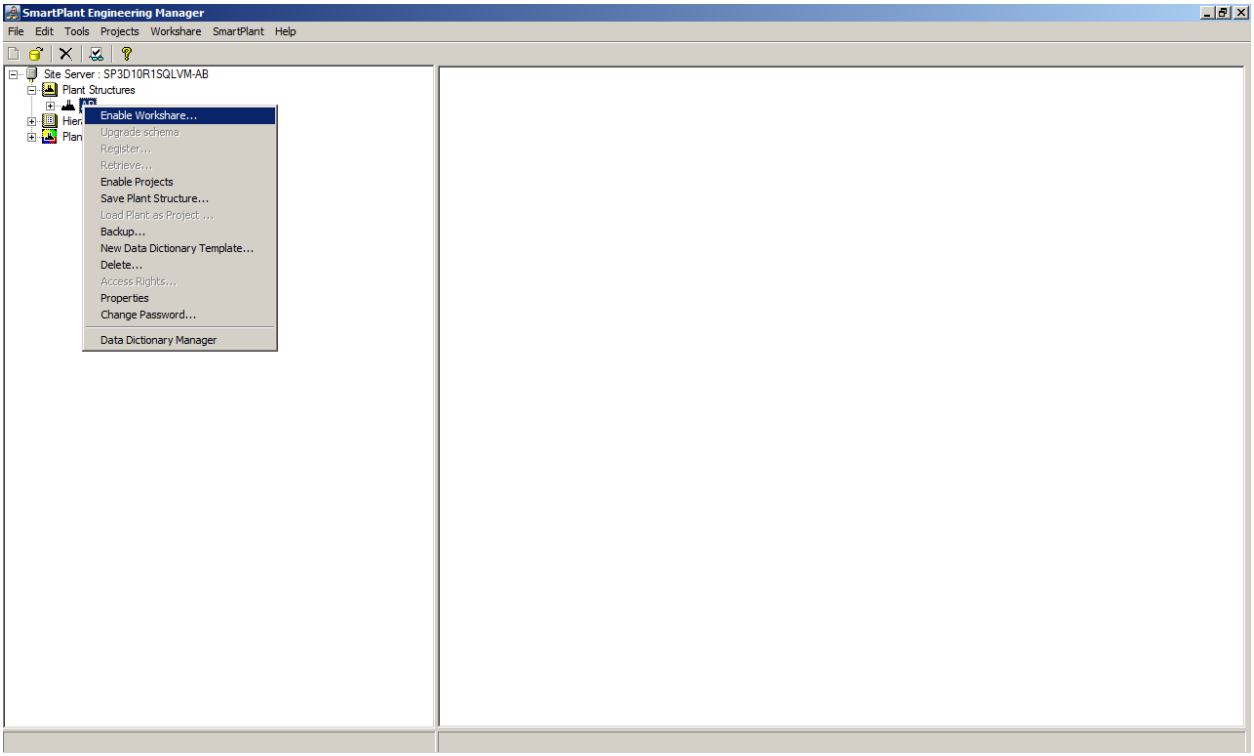
Summary

This document will step you through how to export drawings from SmartPlant PID and then convert these to the iModelHub.

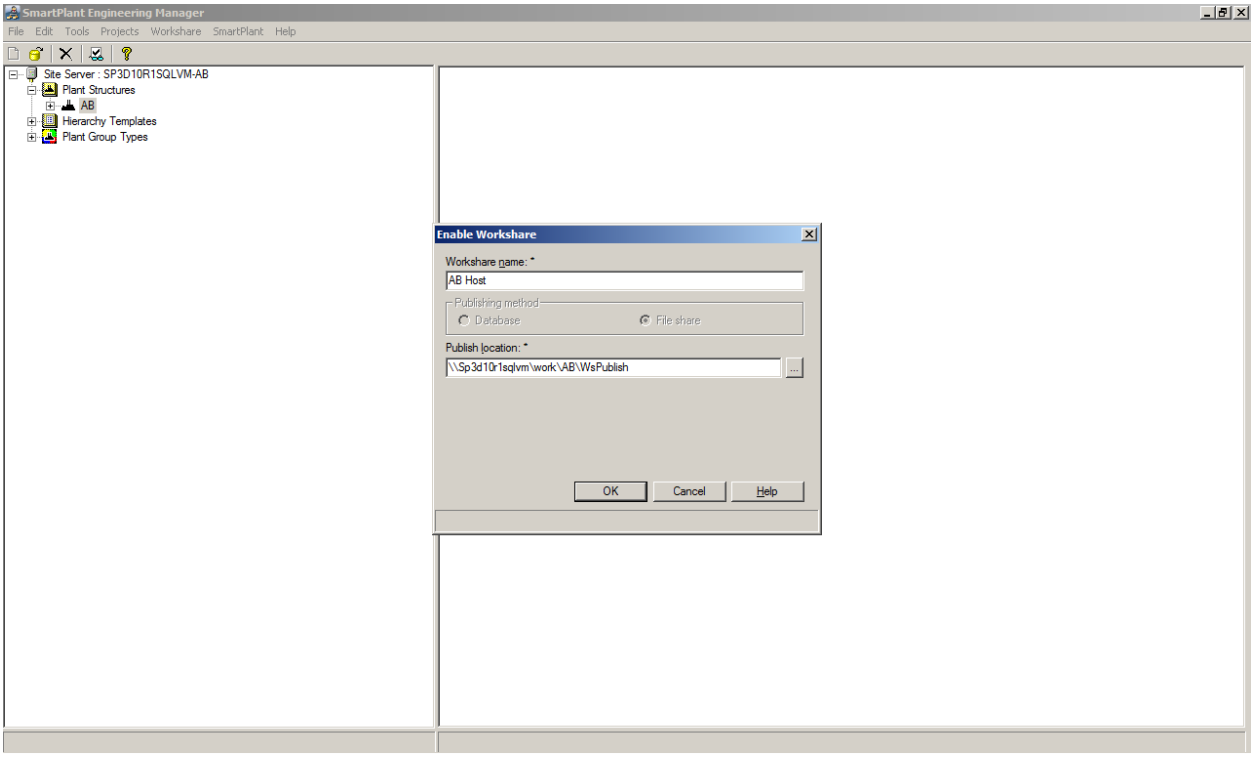
Exporting Drawings from SmartPlant PID

The iModel Bridge for SmartPlant PID works on drawings that are exported using the workshare functionality of SmartPlant PID.

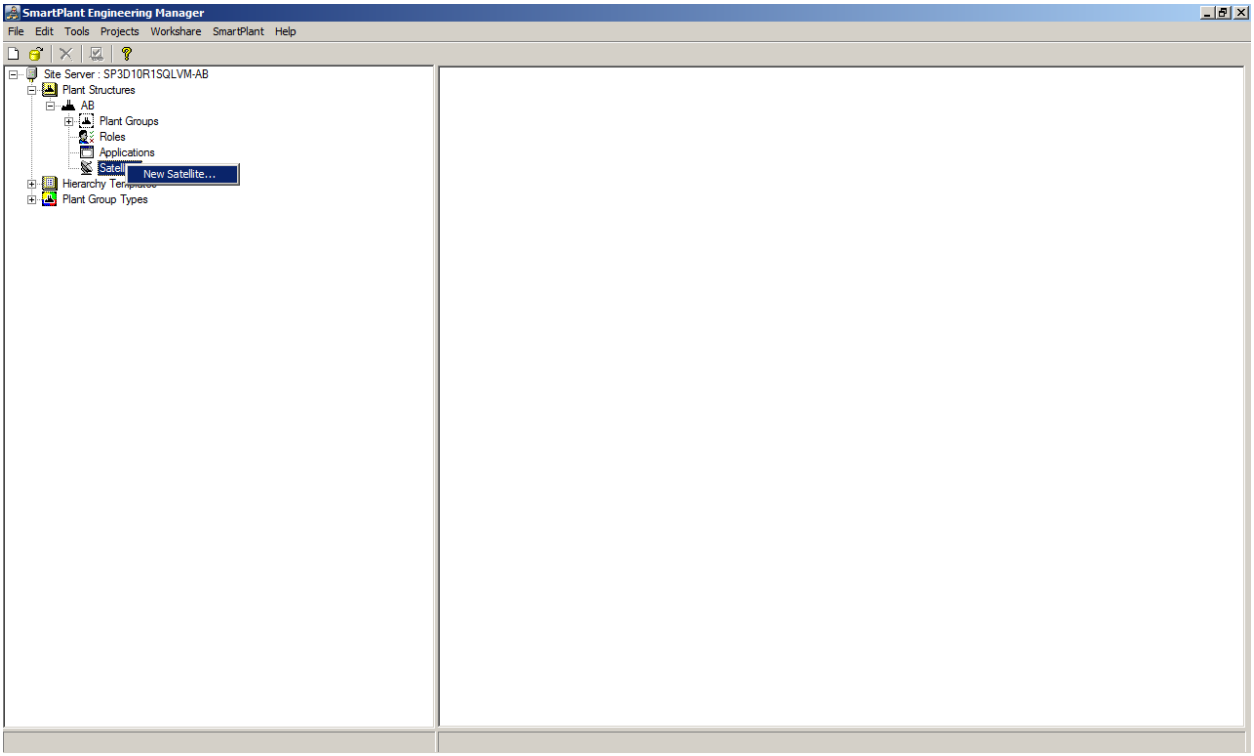
Initially it is required to first enable workshare. To do this, start SmartPlant Engineering Manager. Go to your Plant Structures node as indicated and then either right click on the node or go Workshare \ Enable Workshare… from the menu:



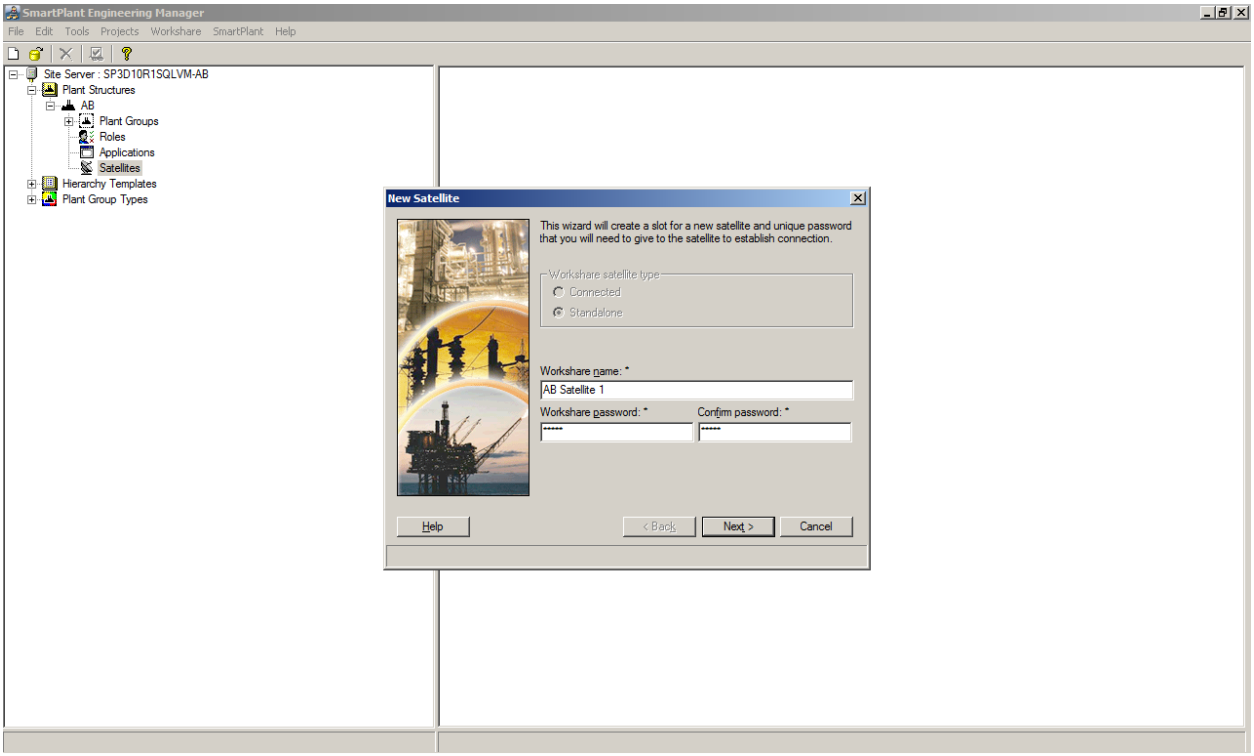
Set the Workshare name and click OK, make sure the file share option is enabled.



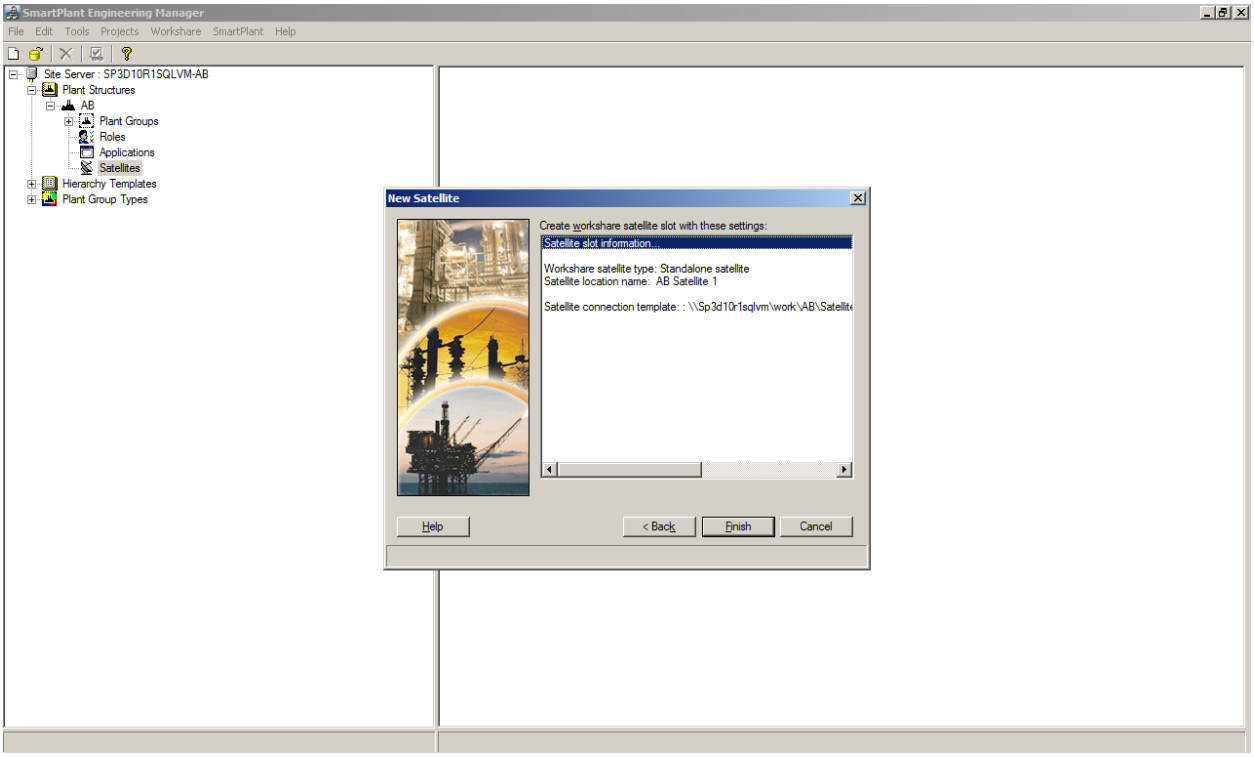
The Satellites node should then be highlighted. Right click and select New Satellite…



Enter a name for the Workshare name and enter a password for the Workshare.

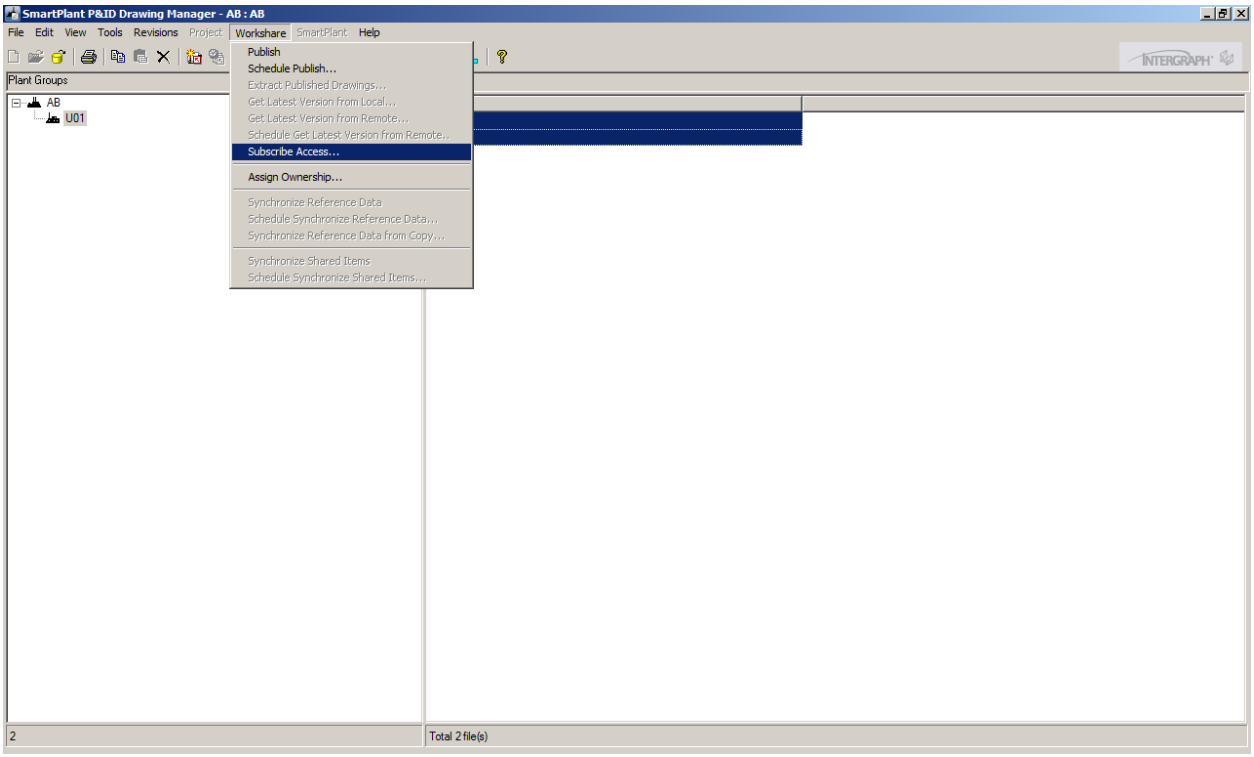


Click Finish

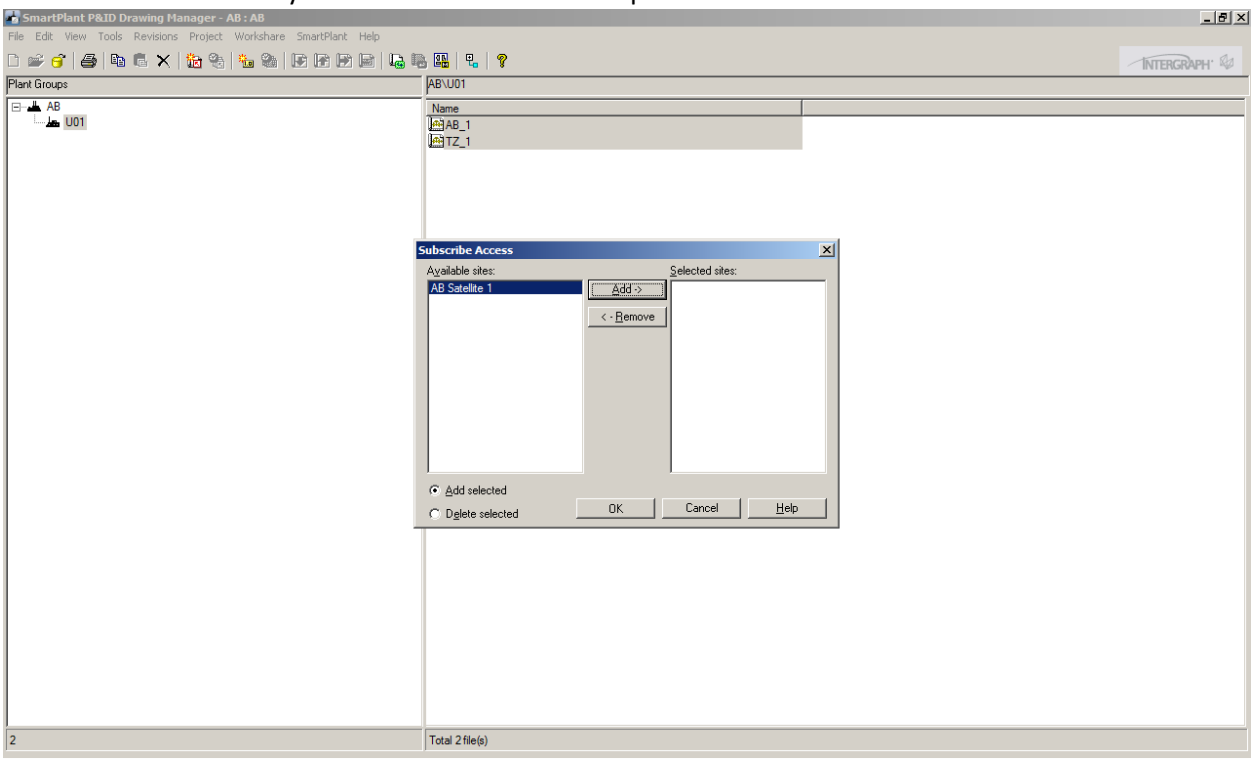


Exit SmartPlant Engineering Manager.

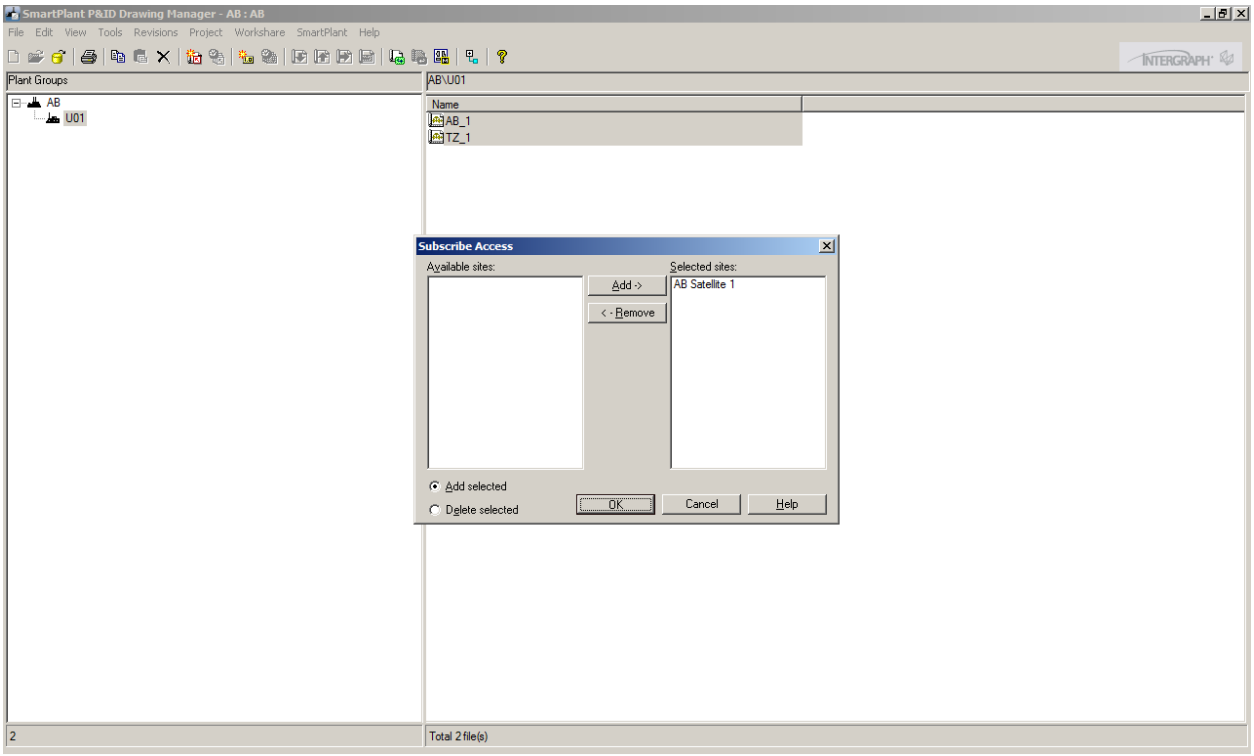
Run SmartPlant P&ID Drawing Manager and go to the node you want exported and select the drawing(s) to be exported. Select Workshare \ Subscribe Access… from the menu



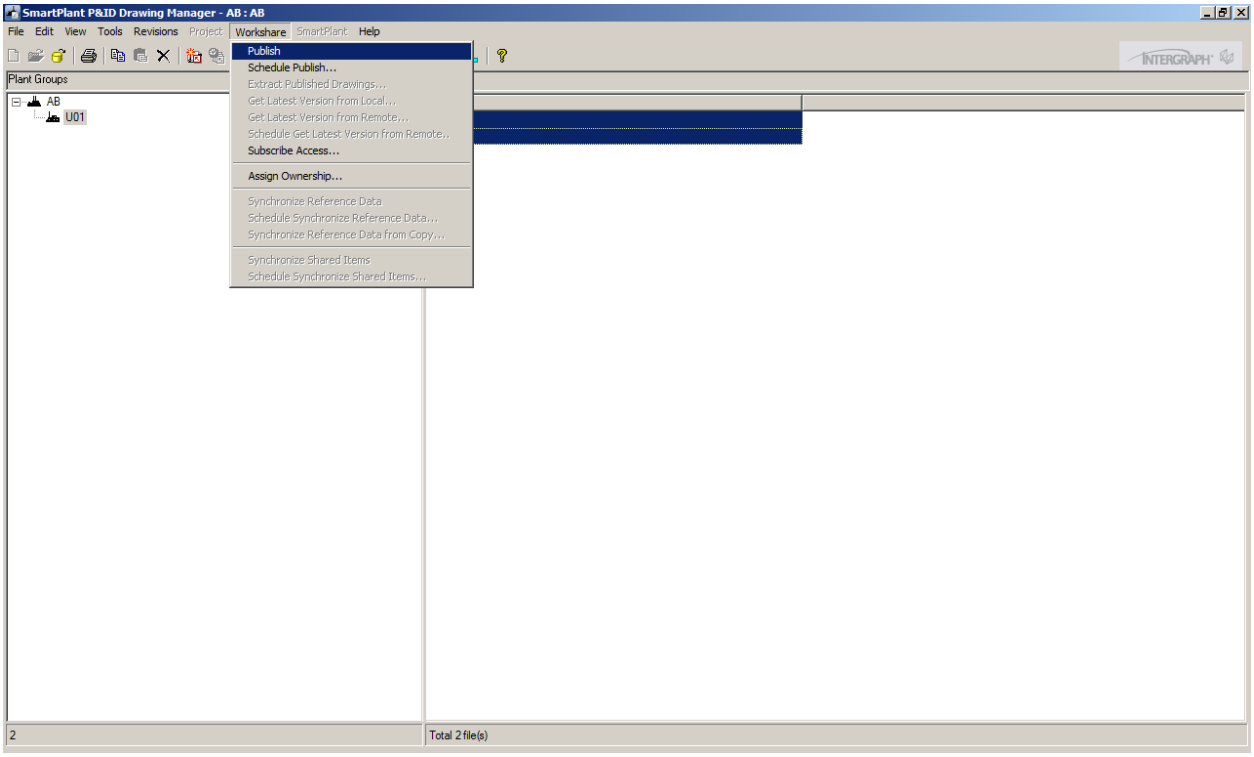
Select the Satellite that you created in the earlier step and then click Add



and click OK.



Then select Workshare \ Publish from the menu.

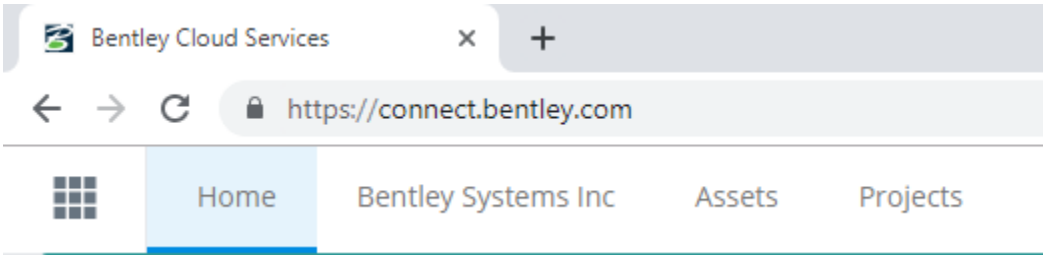


This will create zip files in a folder of Workshare that was created. These zip files will be used as the input to the iModel Bridge. You will also need the satellite file that was created in the Satellite Templates folder of the workshare.

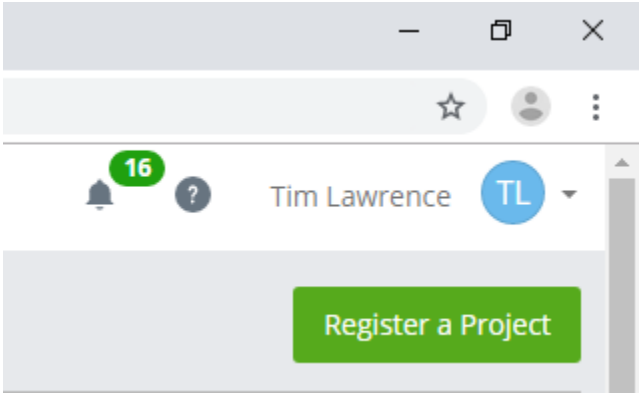
Importing Drawings into the iModelHub

Ensure you have a CONNECT Project

In order to import a PID that can be viewed on the iModelHub, you must first create a project by going to https://connect.bentley.com/, then select Projects



You should have a “Register a Project” button at the top right of your browser window

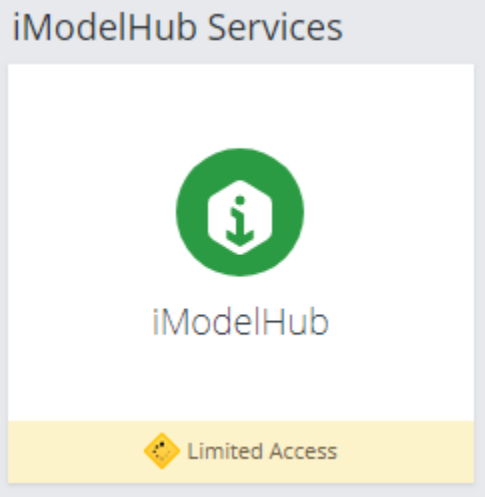


If you don't have a register button, contact your company's administrator for Bentley software.

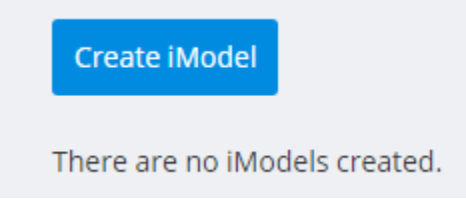
Click Register a Project and fill in the info required and click Register when completed.

Create an iModel

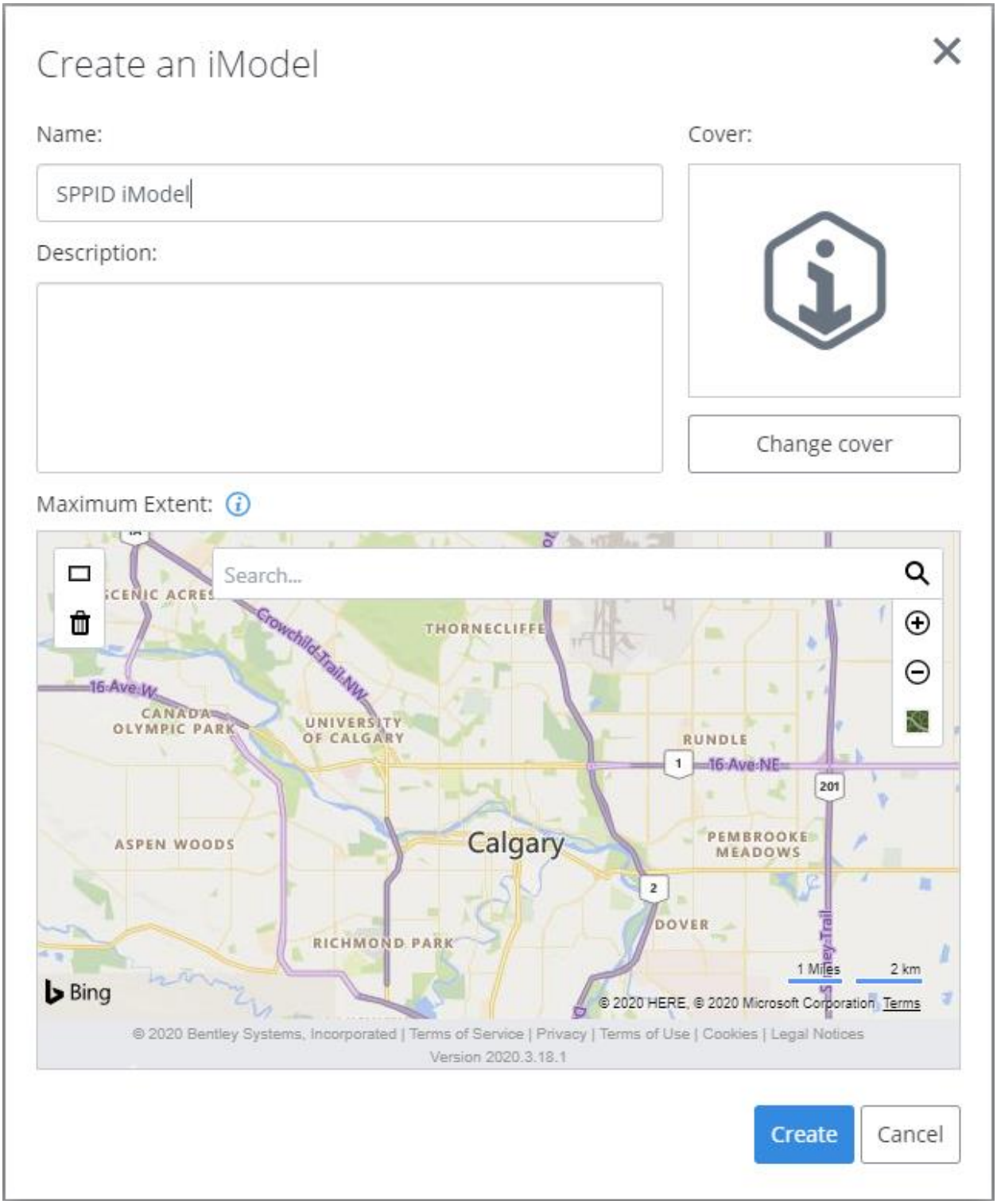
Your project will also require an iModel to be defined. To do this, find the tile labeled "iModelHub Services"



If you have a just created your project, you will only see a button to Create an iModel:



Click the button and name your iModel



Click Create

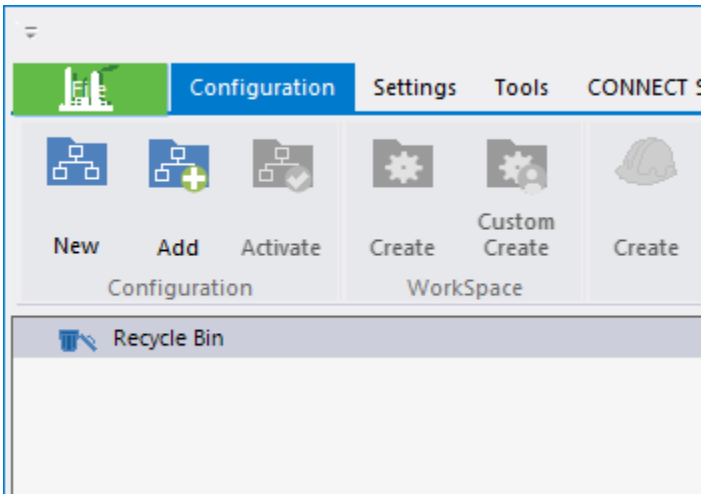
iModel Provisioning

Prior to importing any drawings from SmartPlant PID, you must first provision the iModel. The provisioning will add the schemas to the model. The source for these schemas can currently only come

from using OpenPlant Project Administrator to provision the iModel with the OpenPlant schemas. From OpenPlant Project Administrator 10.08, you do not need to have a separate source for the schemas that previously required the installation of an OpenPlant design application as the WorkSets are now part of the OpenPlant Project Administrator.

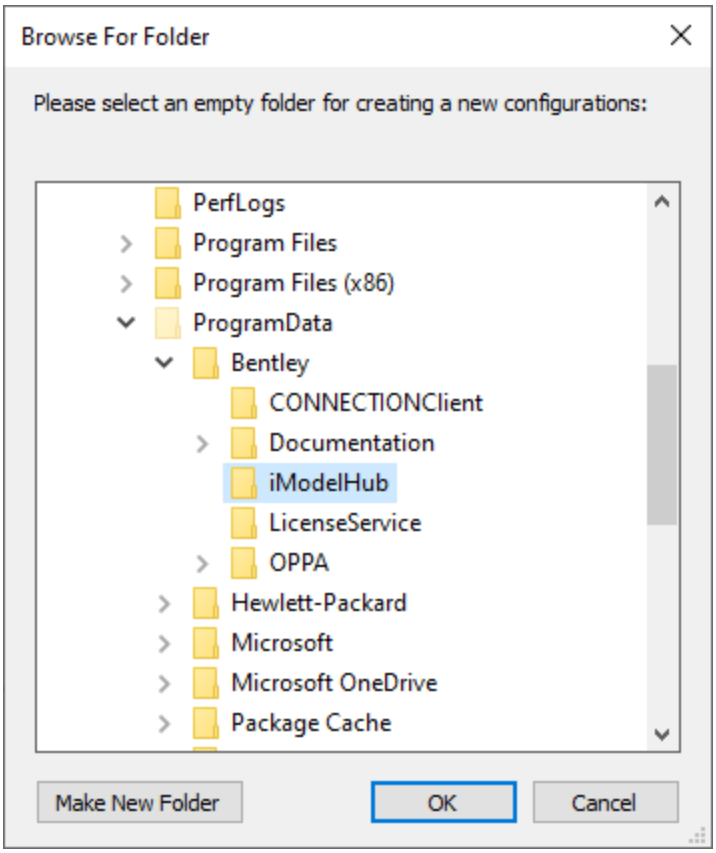
Creating a WorkSet in a clean OpenPlant Project Administrator Setup

Run OpenPlant Project Administrator. If you don't have any design applications installed, you will have an empty set of workspaces

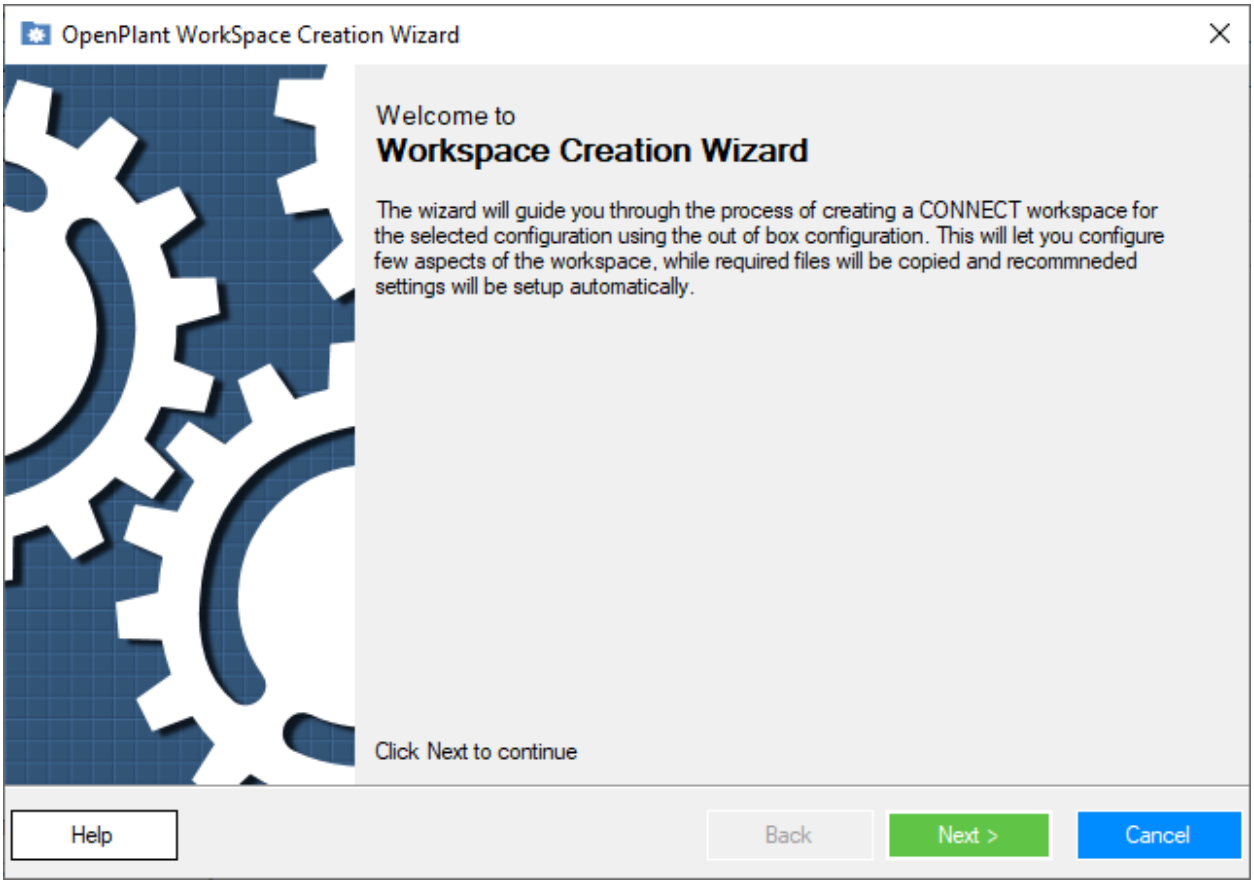


Click New

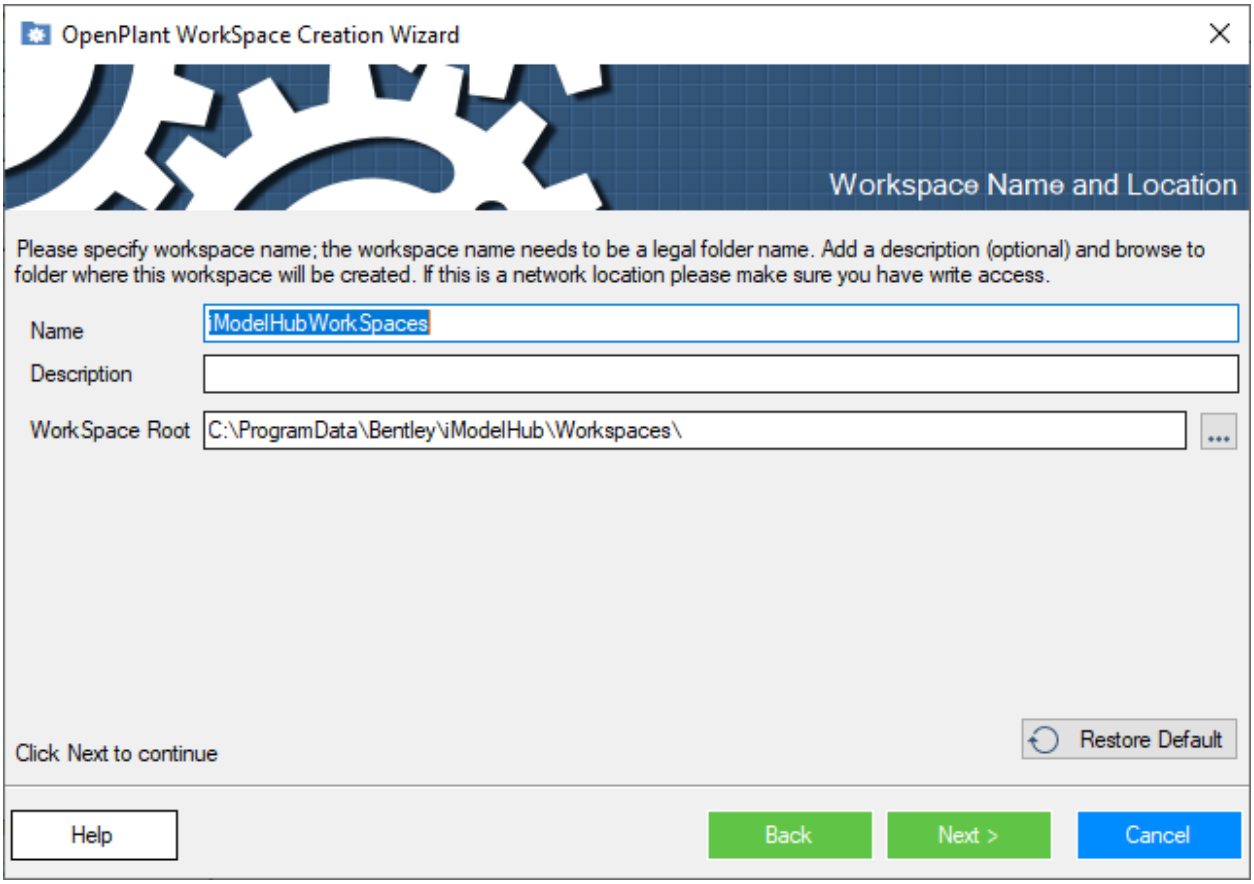
The default location for this data would be under C:\ProgramData\Bentley, but you are free to choose your own location. Make a New Folder and name it:



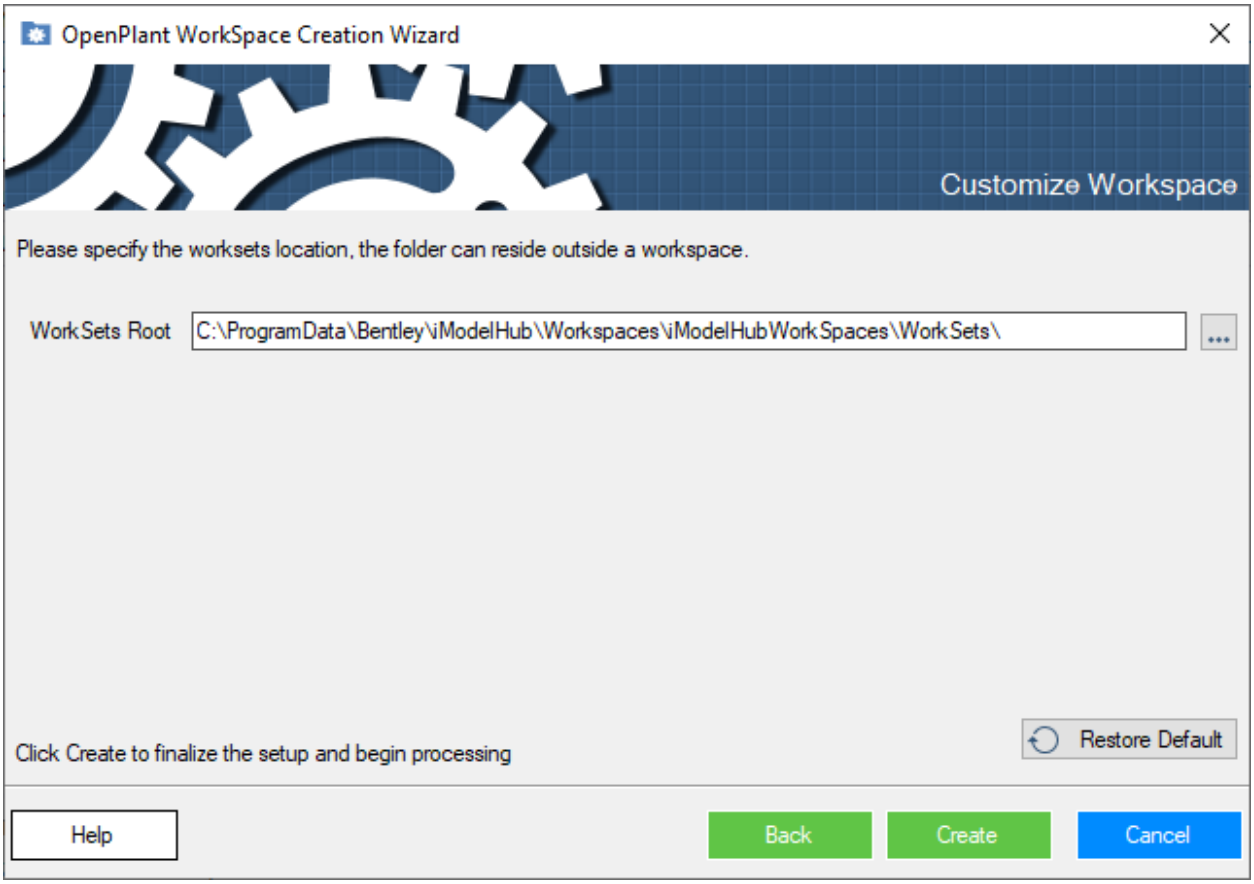
Now Create a new Workspace by clicking on Create which brings up the Workspace Creation Wizard:



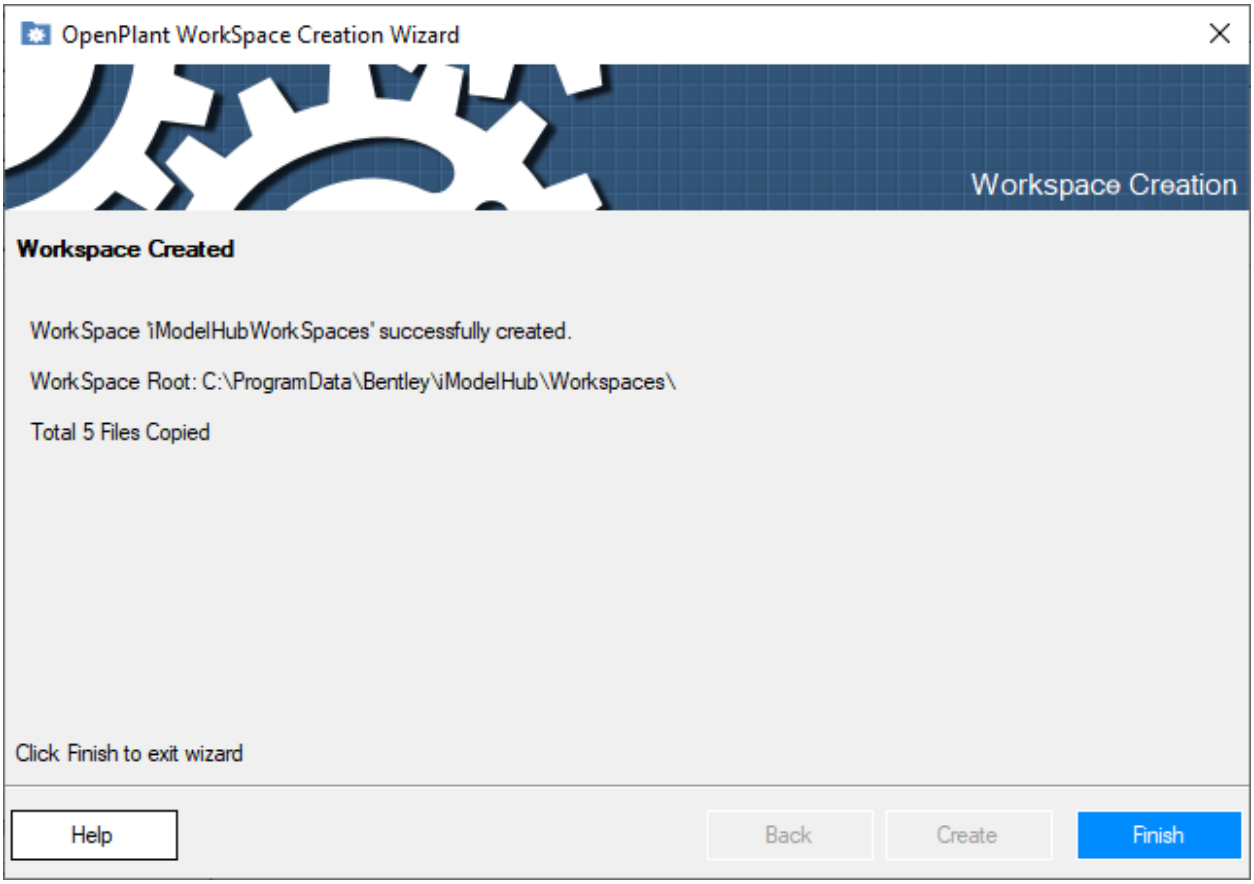
Click Next, then give the WorkSpace a name:


OpenPlant WorkSpace Creation Wizard
Workspace Name and Location
Please specify workspace name; the workspace name needs to be a legal folder name. Add a description (optional) and browse to folder where this workspace will be created. If this is a network location please make sure you have write access.
Name
iModelHubWorkSpaces
Description
WorkSpace Root
C:\ProgramData\Bentley\iModelHub\Workspaces\
Click Next to continue
Restore Default
Help
Back
Next >
Cancel

Click Next

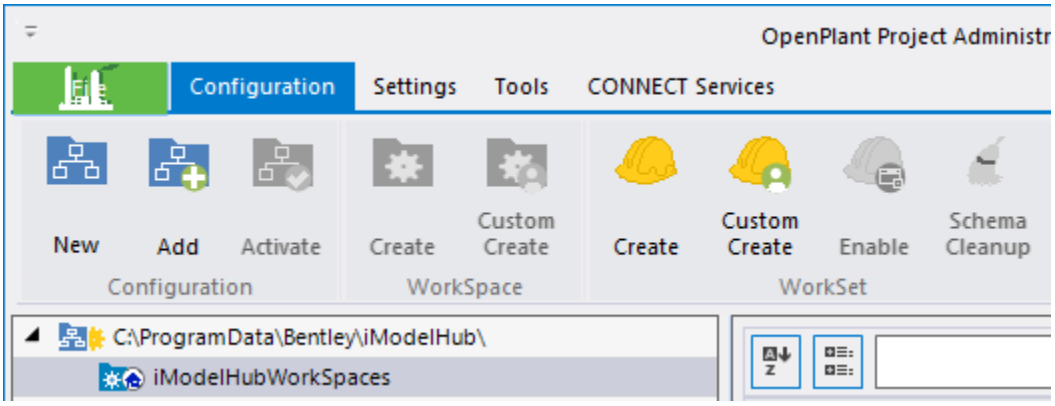

OpenPlant WorkSpace Creation Wizard
Customize Workspace
Please specify the worksets location, the folder can reside outside a workspace.
WorkSets Root
C:\ProgramData\Bentley\iModelHub\Workspaces\iModelHubWorkSpaces\WorkSets\
Click Create to finalize the setup and begin processing
Restore Default
Help
Back
Create
Cancel

Then Create

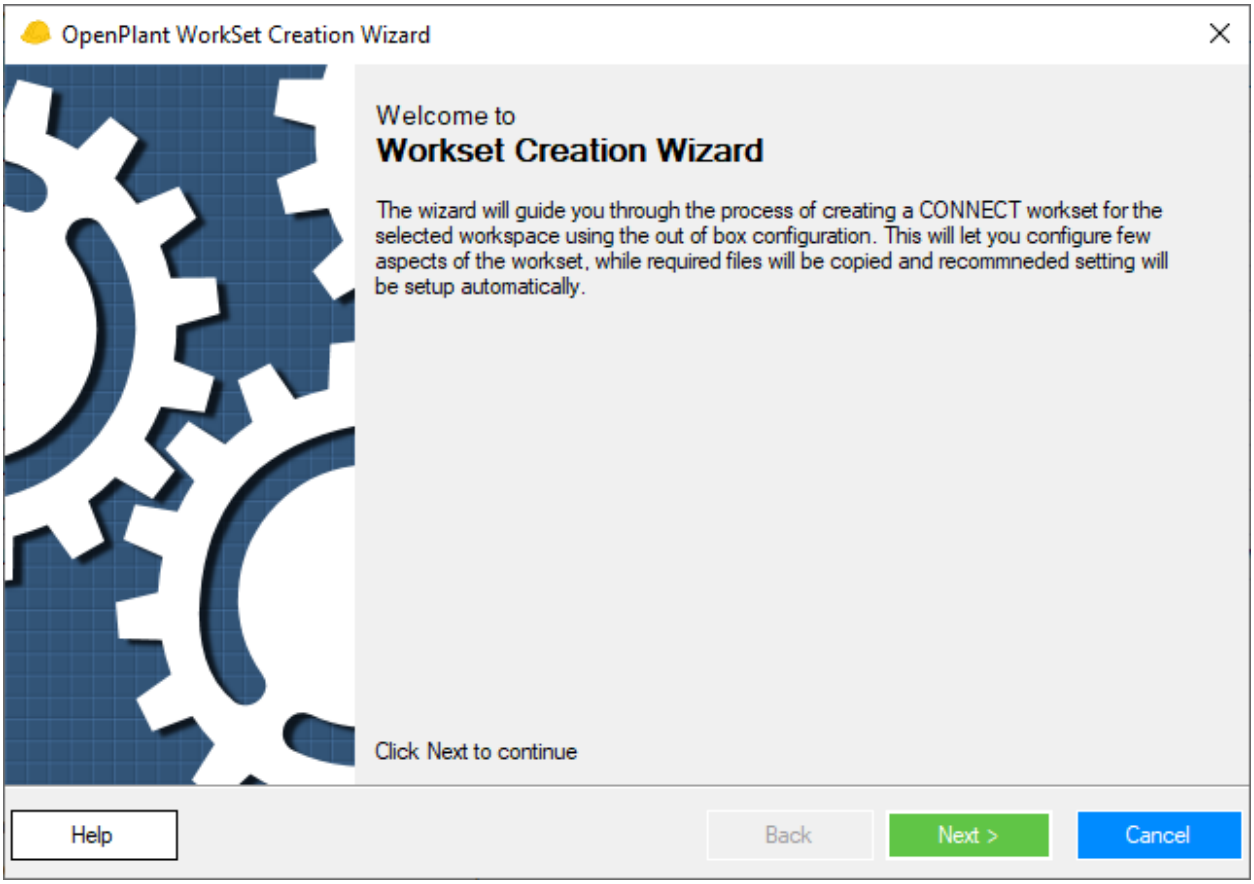


After the WorkSpace has been created, click Finish.

Select the WorkSpace in the hierarchy, then click Create under WorkSet

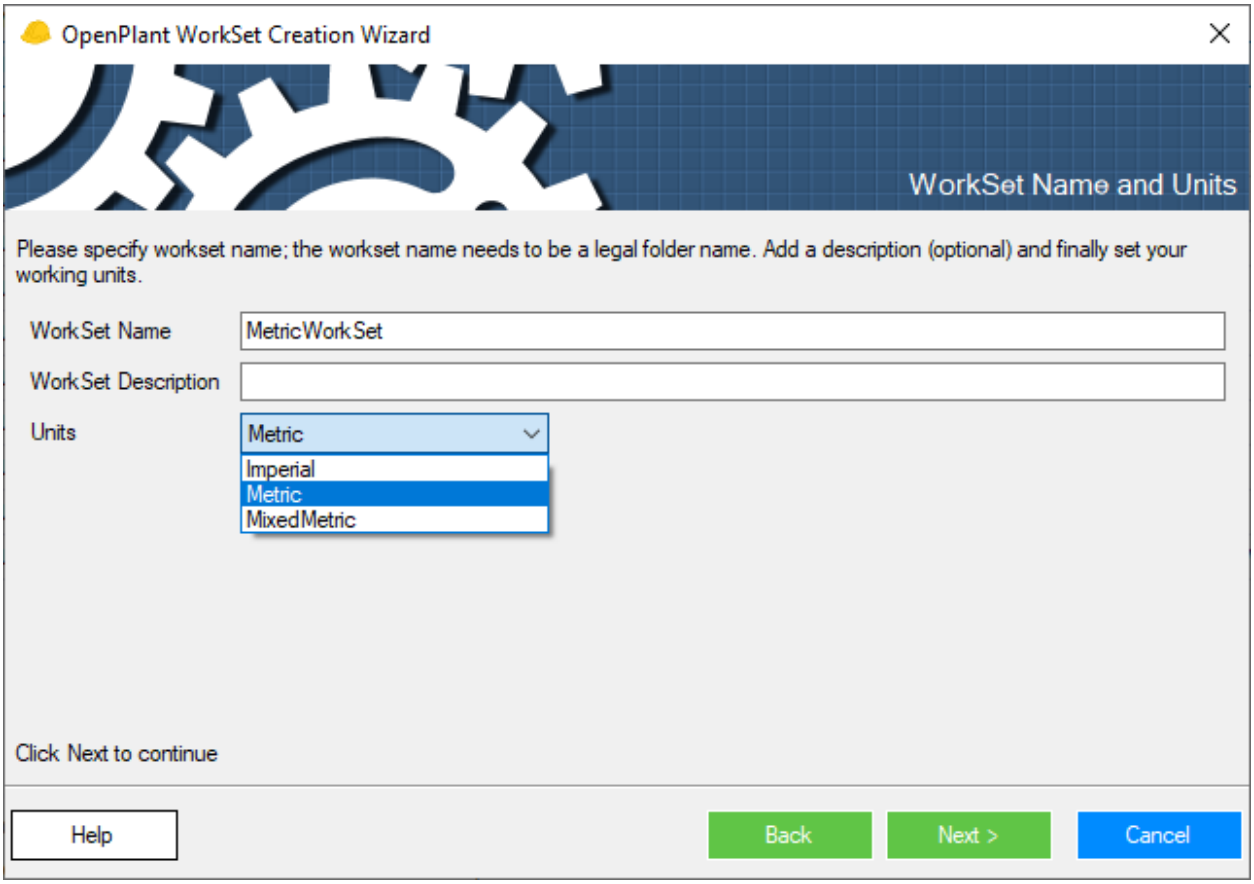


This brings up the WorkSet Creation Wizard:



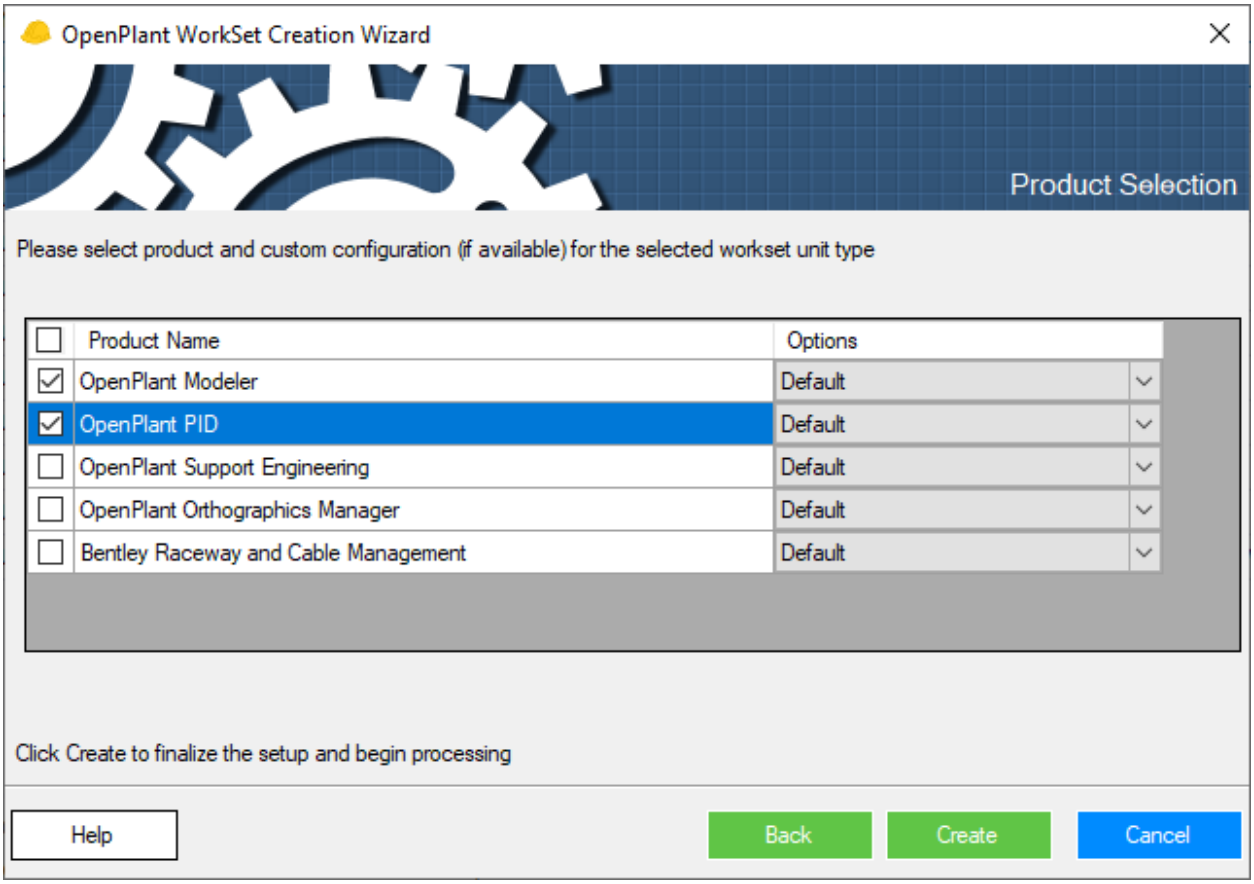
Click Next

Give the WorkSet a name and set the units:

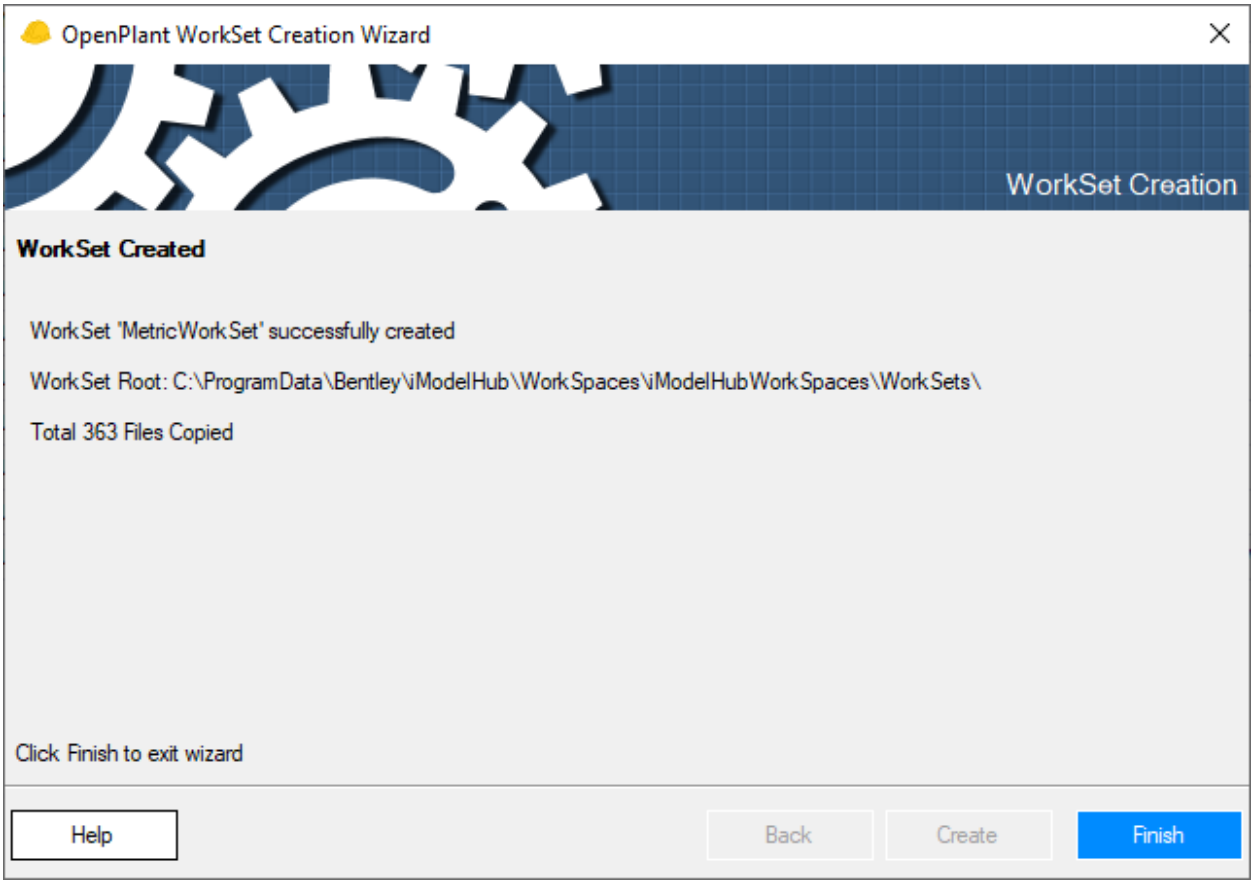


Click Next

Select at least OpenPlant PID and if you are also importing a 3D model that will convert the piping and equipment to use the OpenPlant schema.



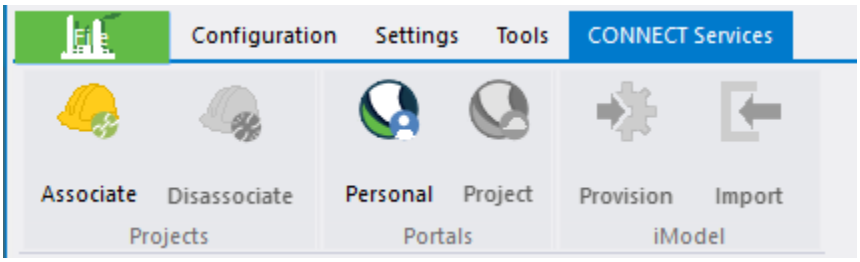
Click Create.



Click Finish after the creation is complete.

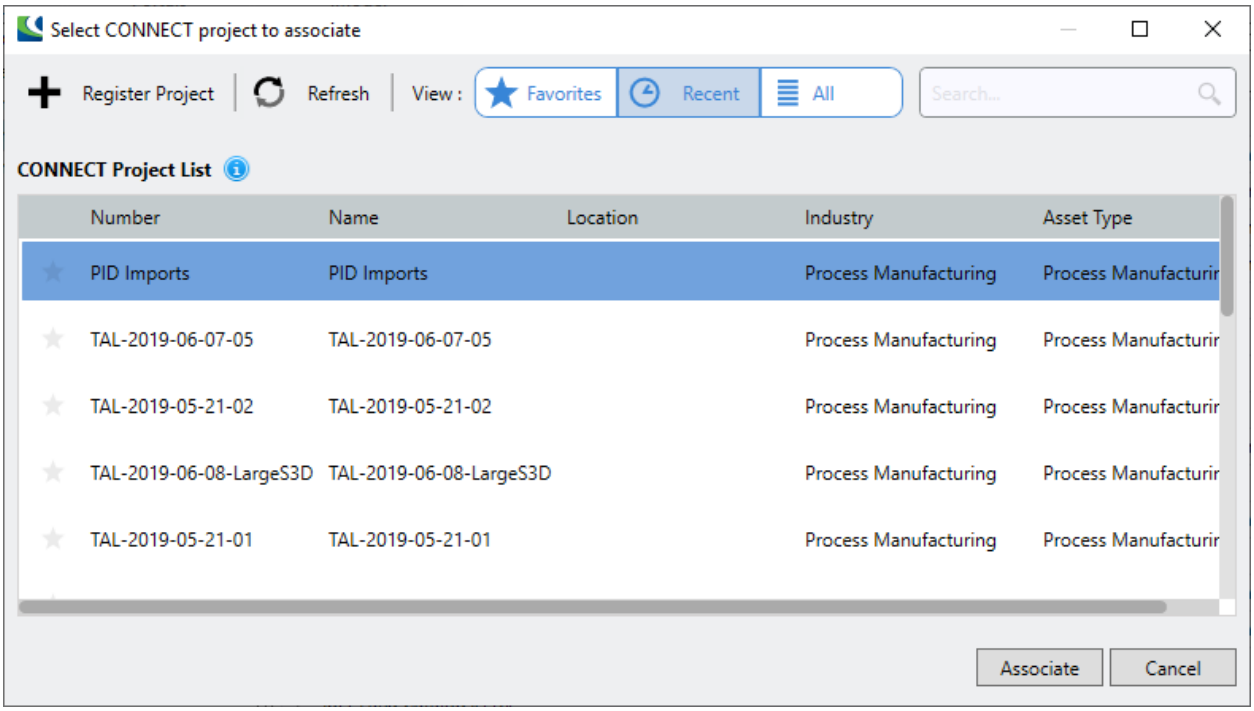
Provision the iModel using the created schema

Make sure the WorkSet you created is selected and then switch to the CONNECT Services tab, then click Associate



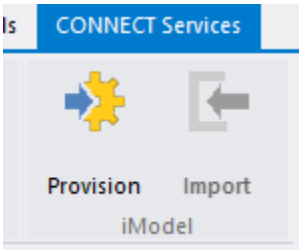
If Associate is not highlighted, then your workspace has not been selected.

Select the project you created:

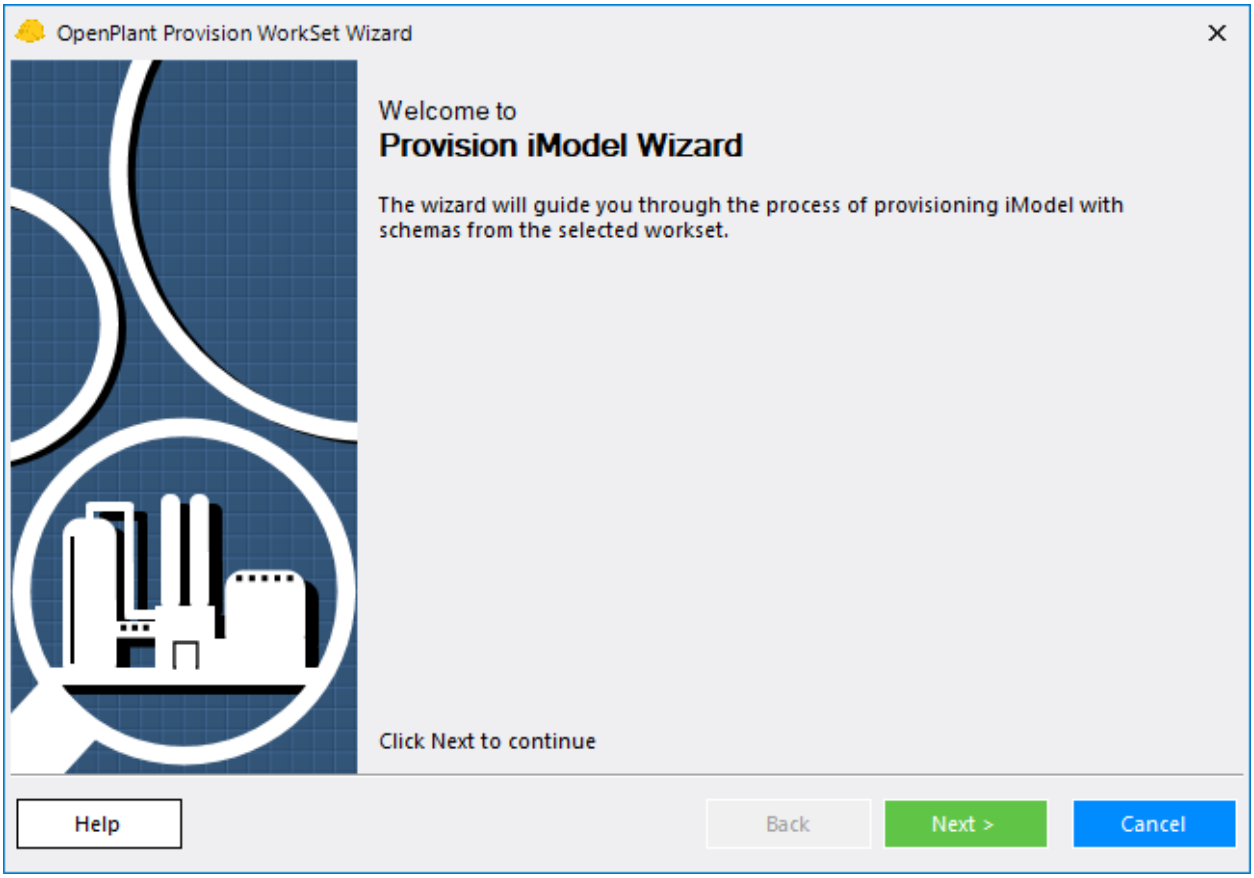


And then click Associate.

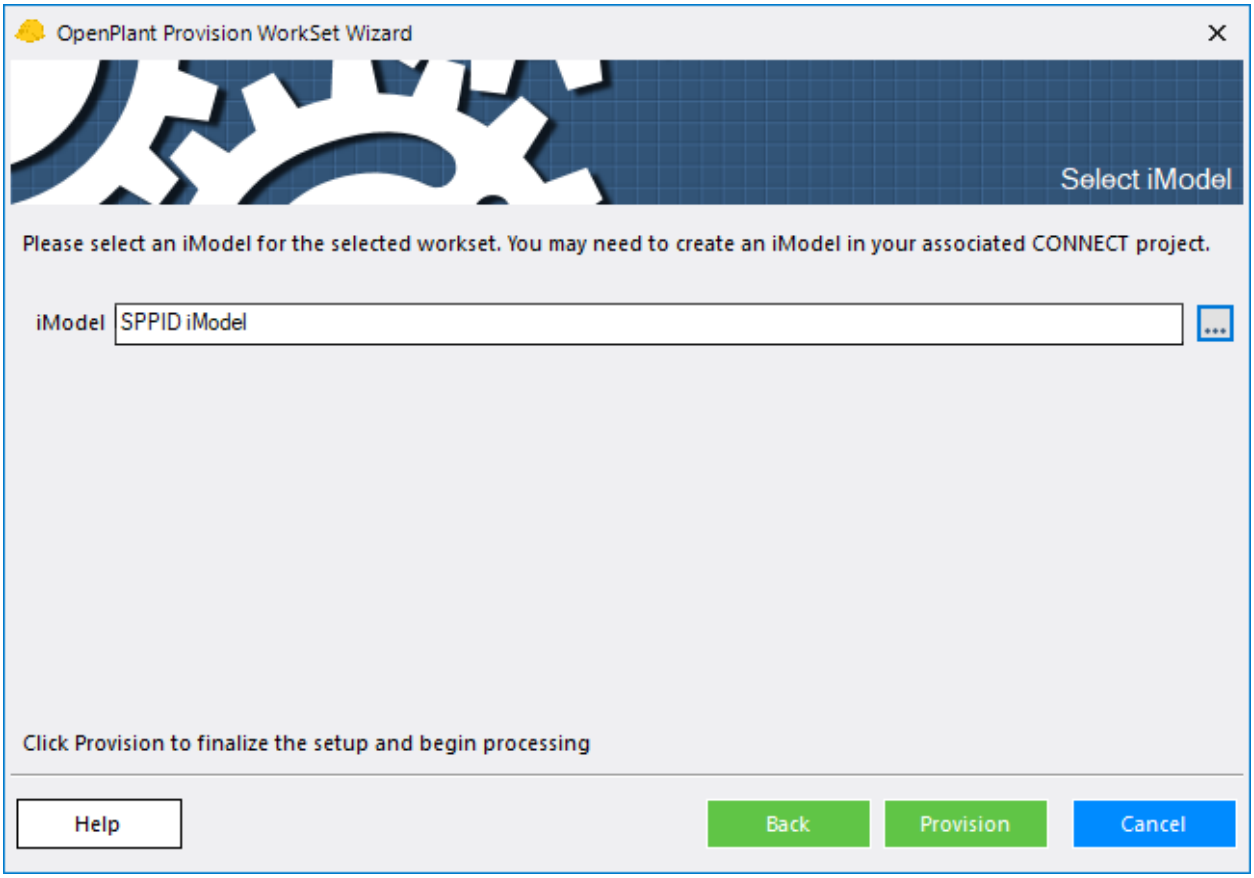
Then click Provision



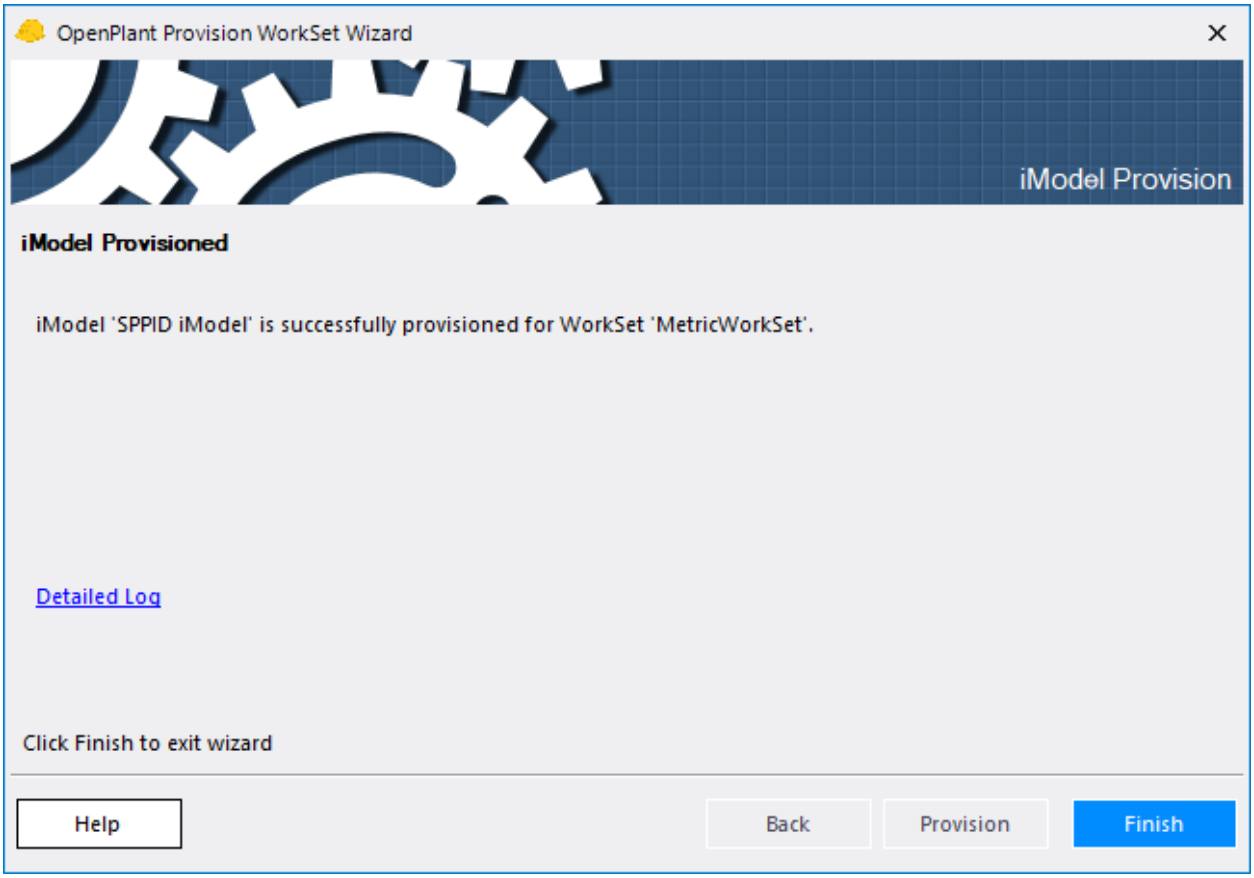
This will bring up the Provision Workset Wizard



Click Next and then select the iModel that you created earlier. One you have selected the iModel, click Next which will provision the iModel.



When the provisioning has finished, click Finish.



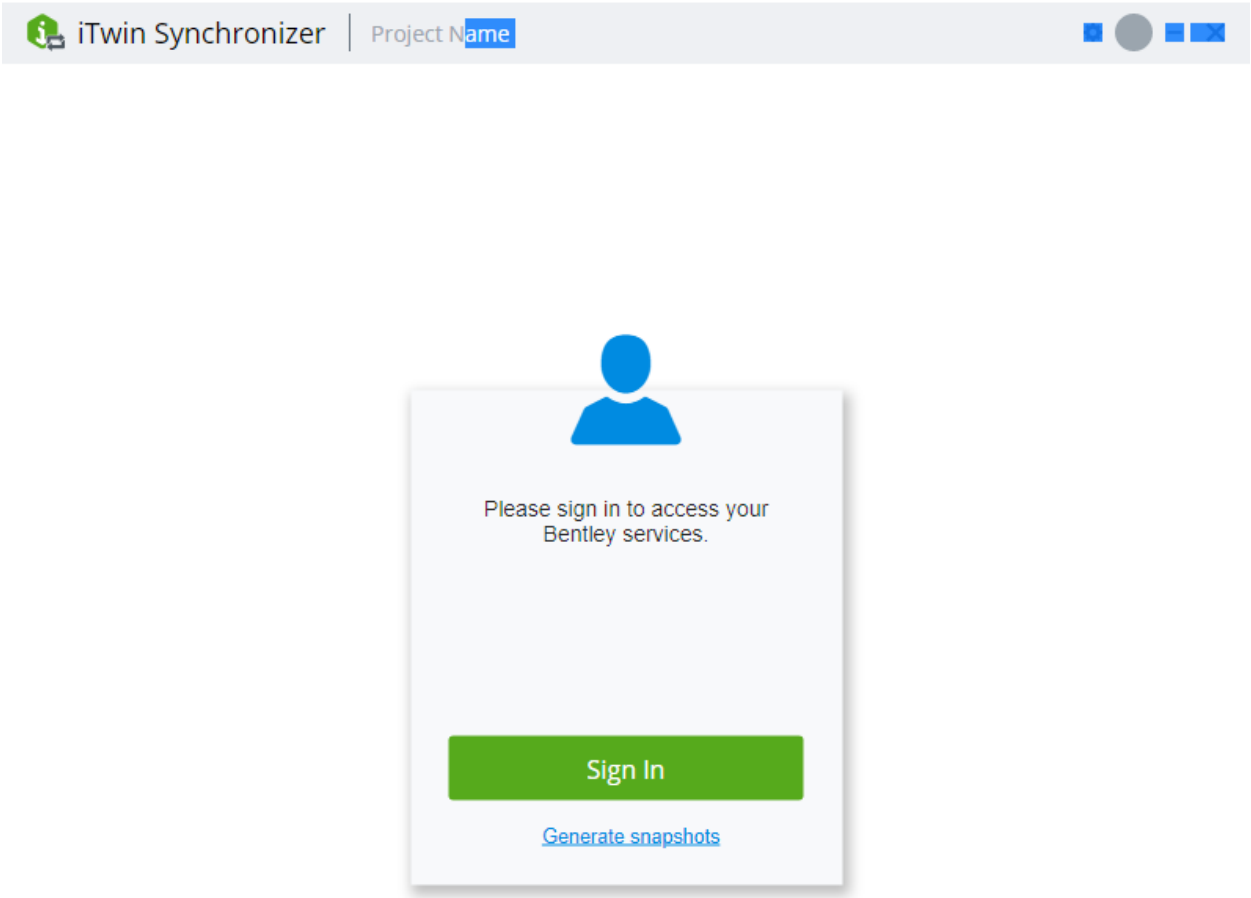
You are now ready to import drawings into your iModel.

Using the iTwin Synchronizer

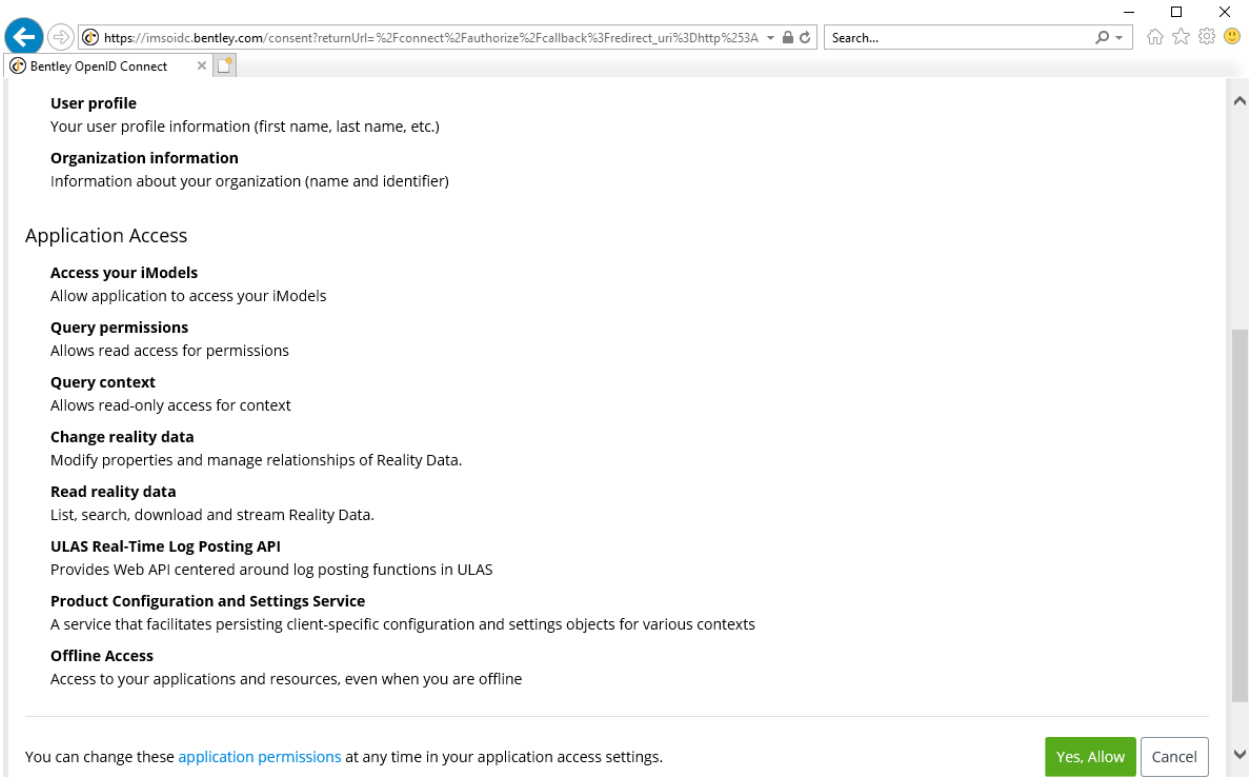
Create the Project and iModel you wish to use and make sure that it is provisioned using the steps outlined in previous sections then you can convert drawings and models from your desktop by downloading the iTwin Synchronizer:

https://www.bentley.com/en/products/product-line/digital-twins/itwin-synchronizer

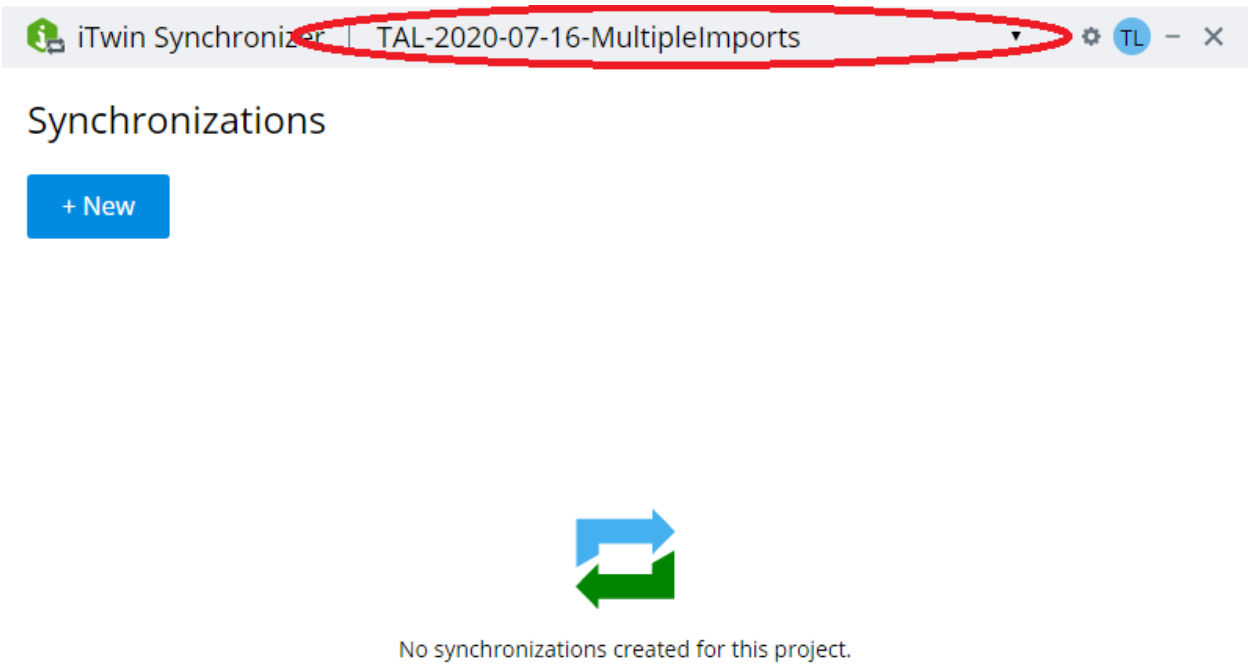
When you first run the iTwin Synchronizer, you need to sign in:



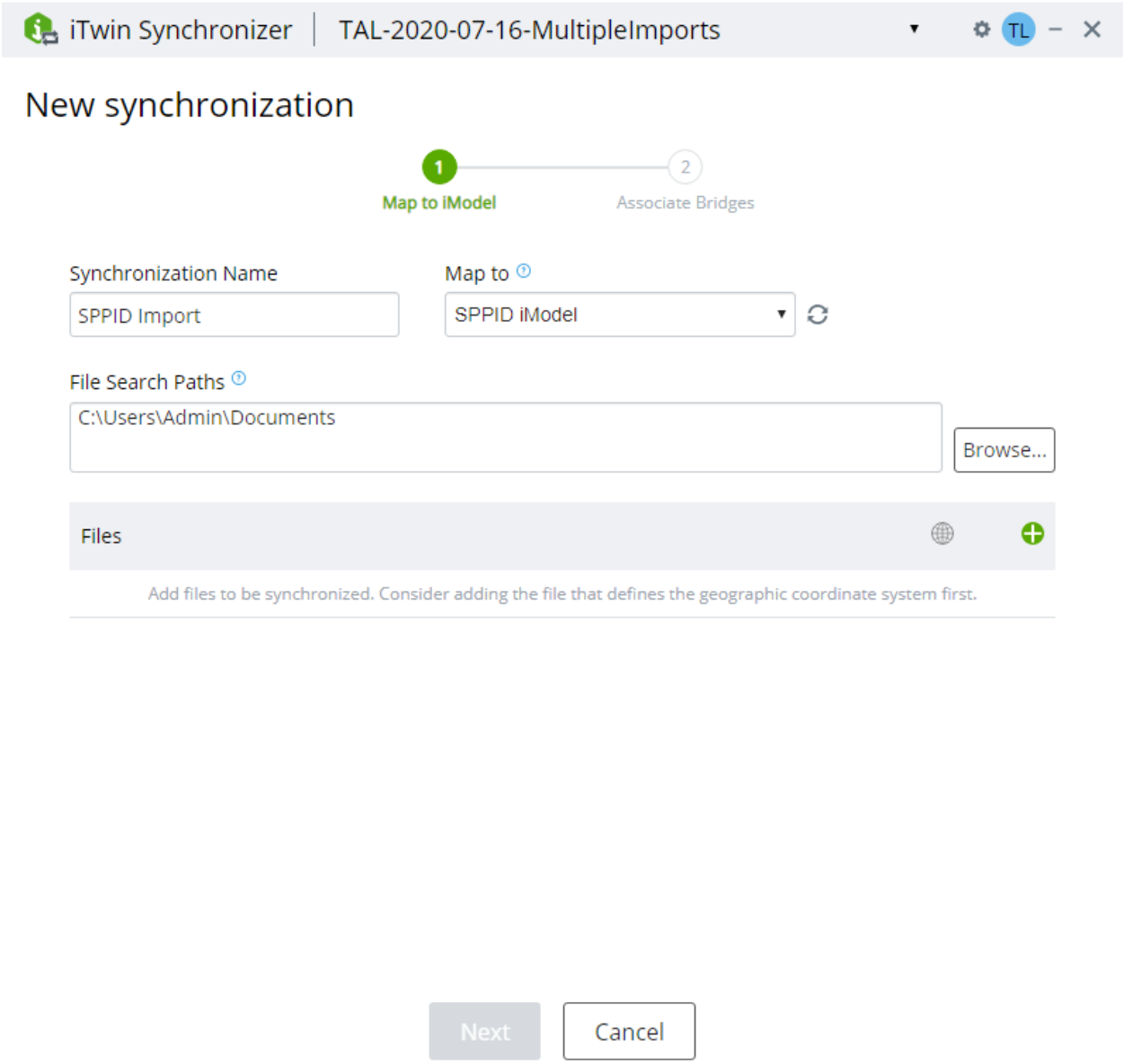
After you sign in, you will need to Allow the application various permissions:



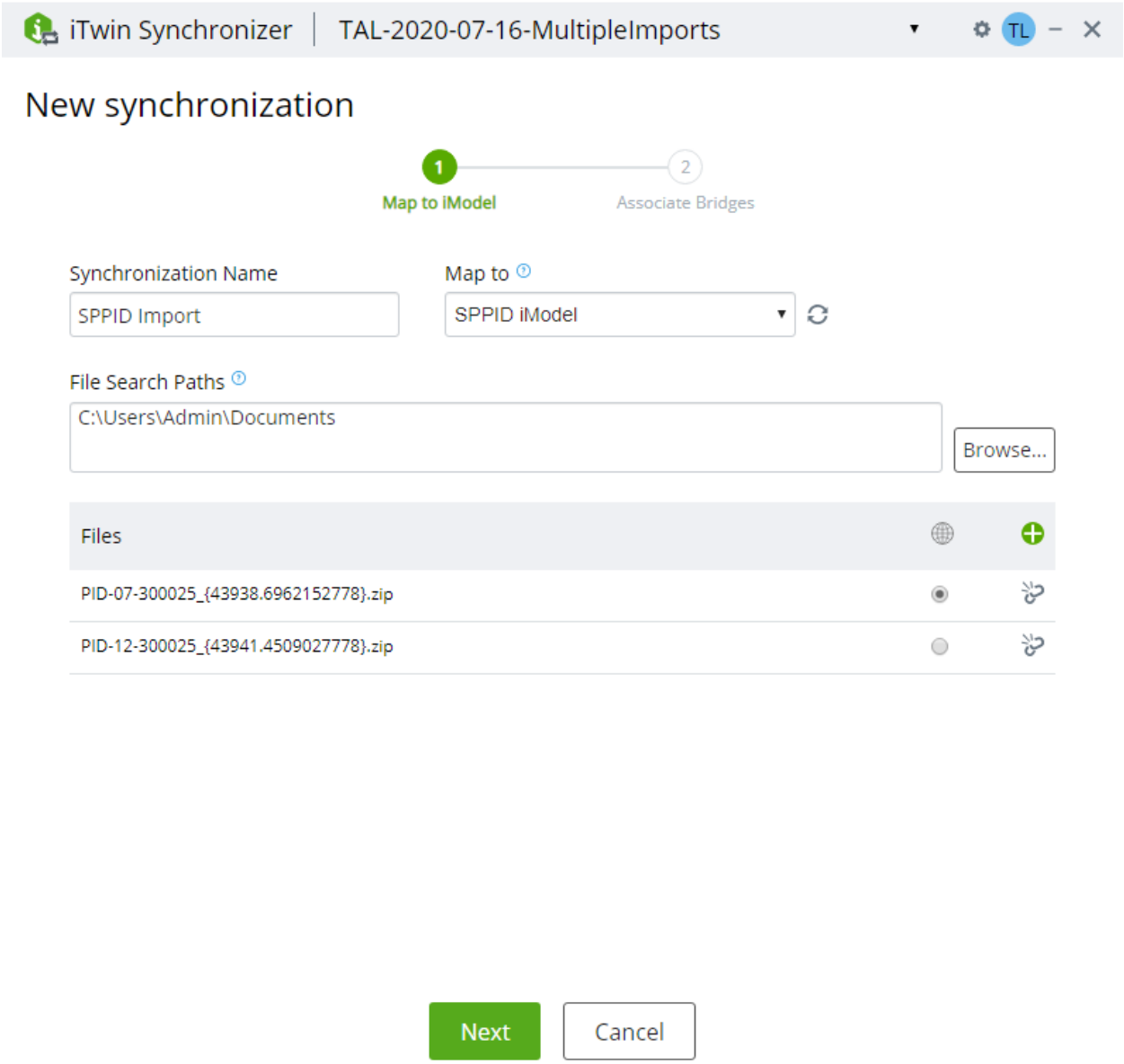
Then select your iModelHub Project:



Click New, then fill in the name of the synchronization, select the iModel from the available iModels and select the path to find the .zip files.

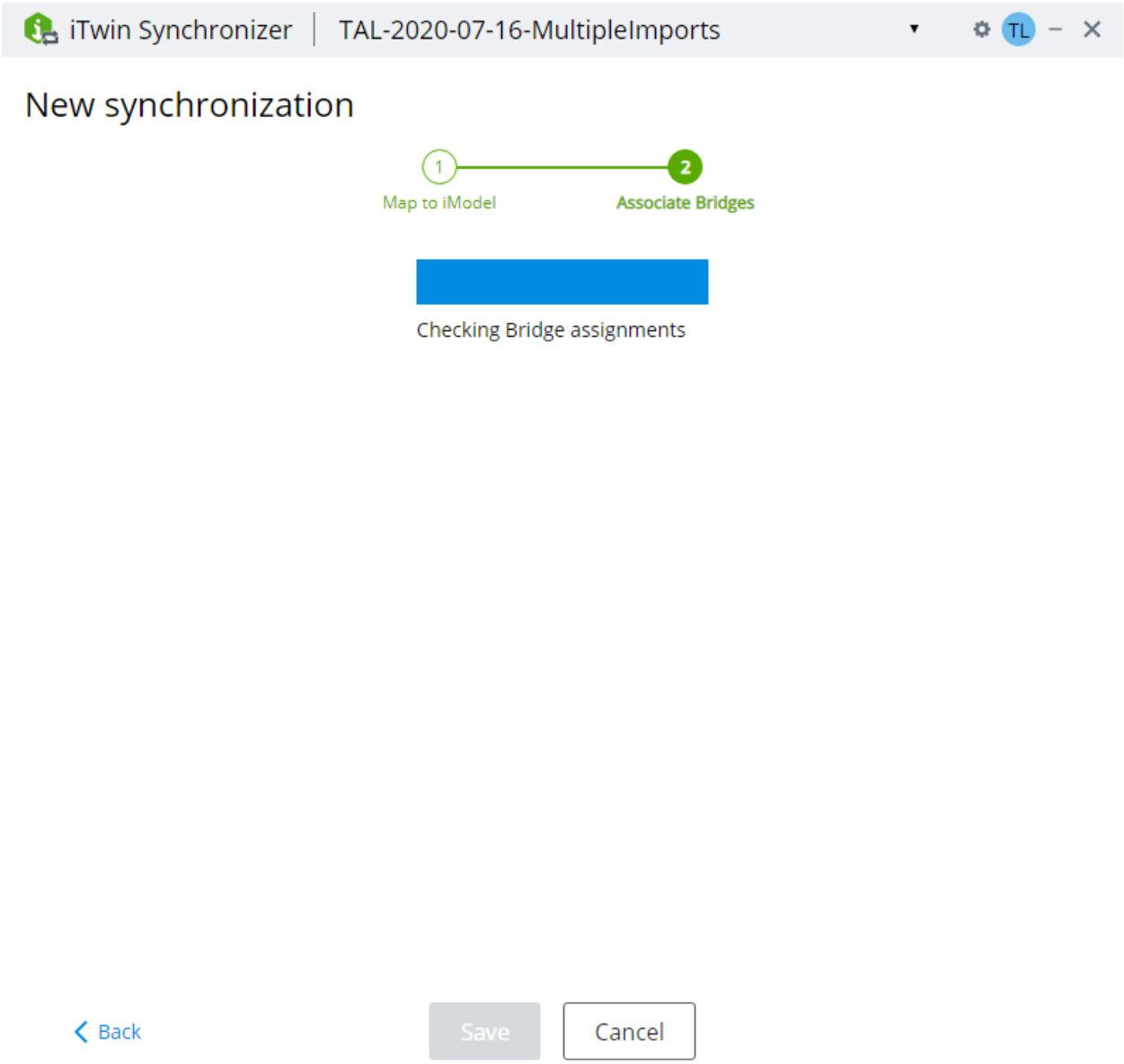


Then click the green + and select one or more SPPID zip exports.

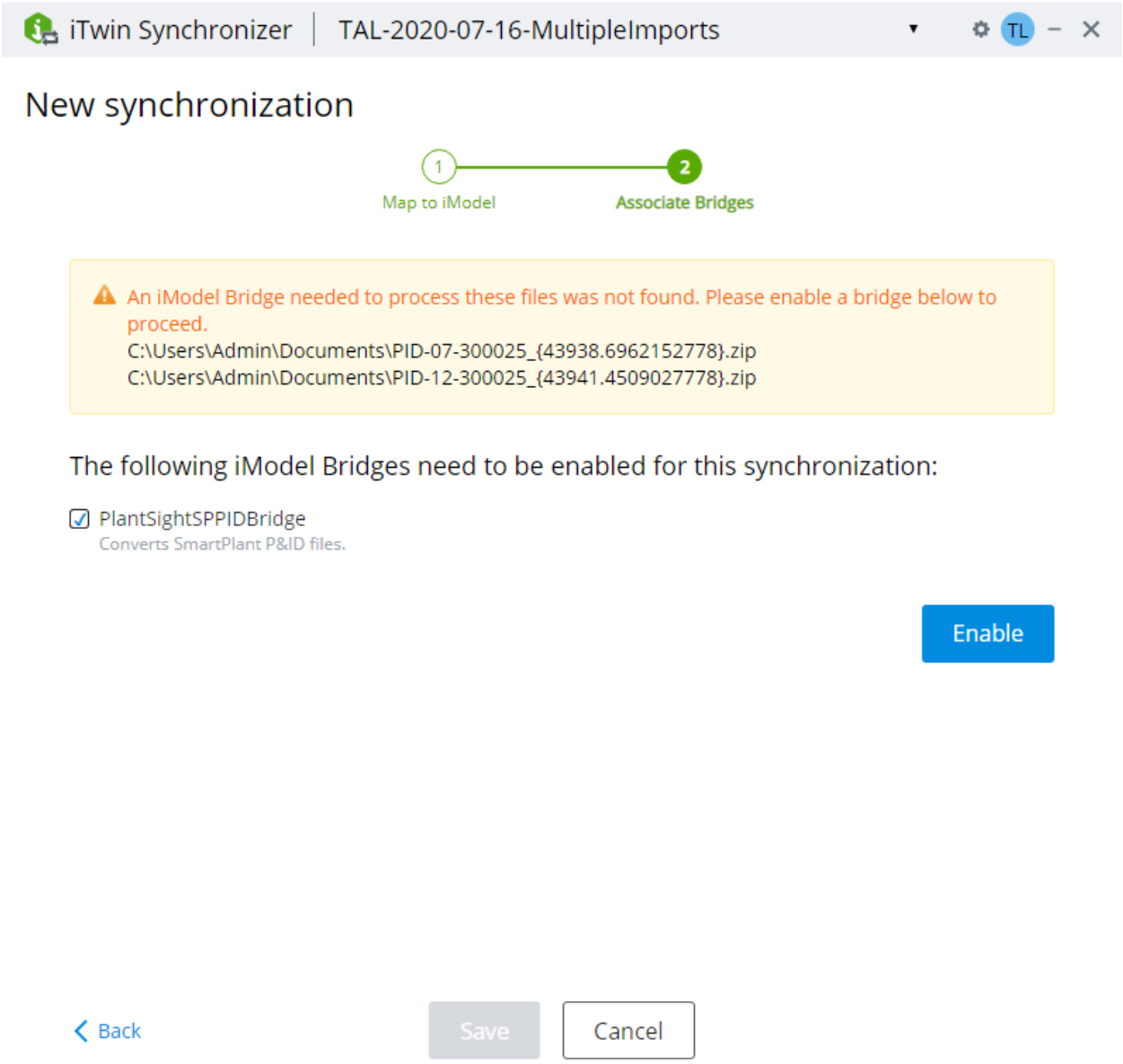


Click Next

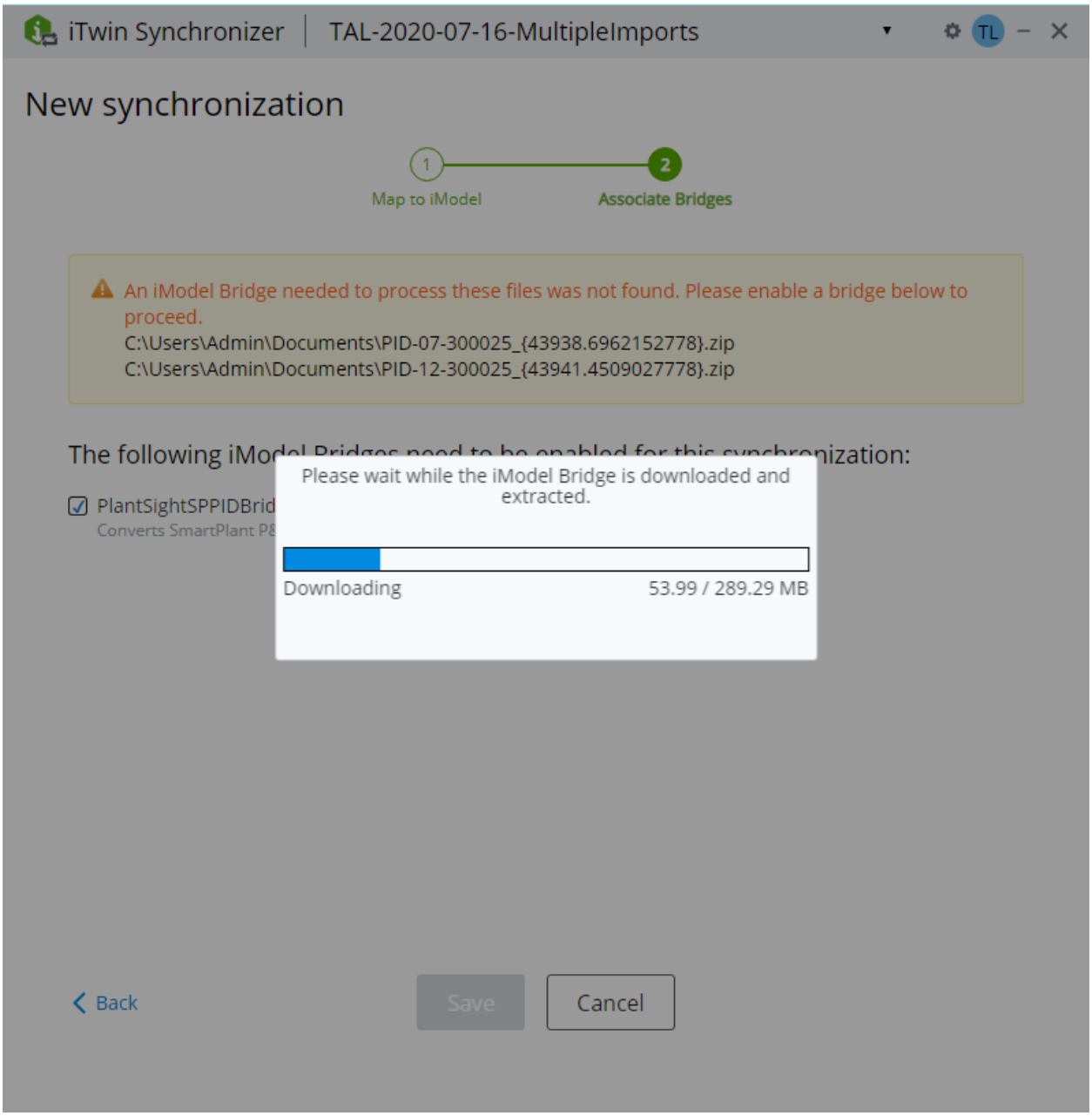
The iTwin Synchronizer will check to see if a Bridge is available for these files:



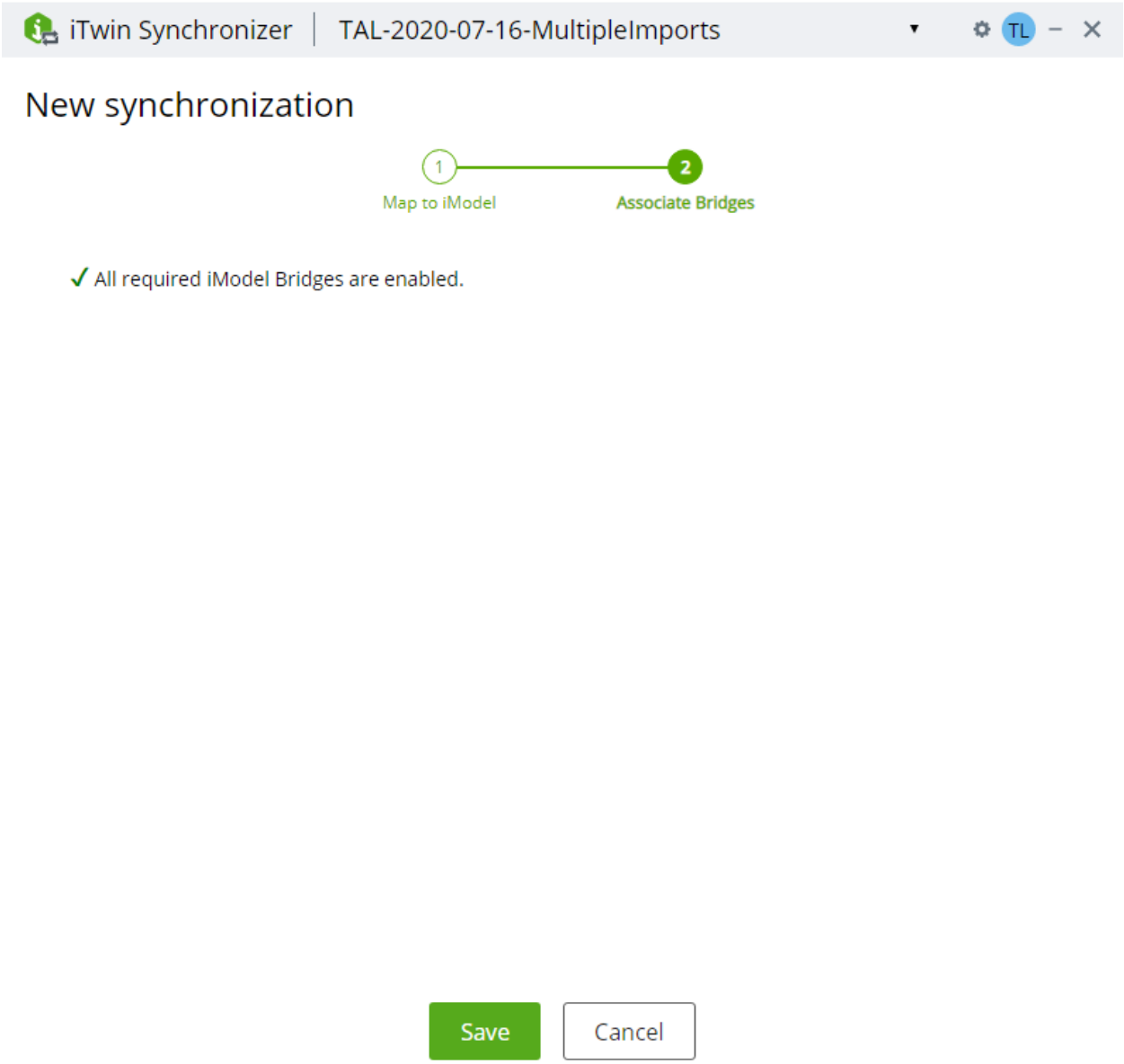
Since this is a new installation, there are no bridges available, so the SPPID Bridge is required to be downloaded.



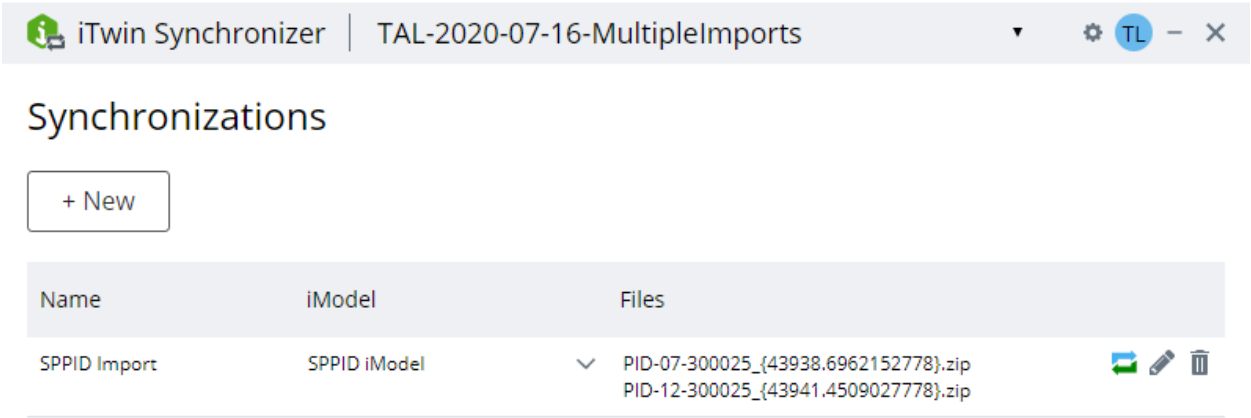
Click Enable.



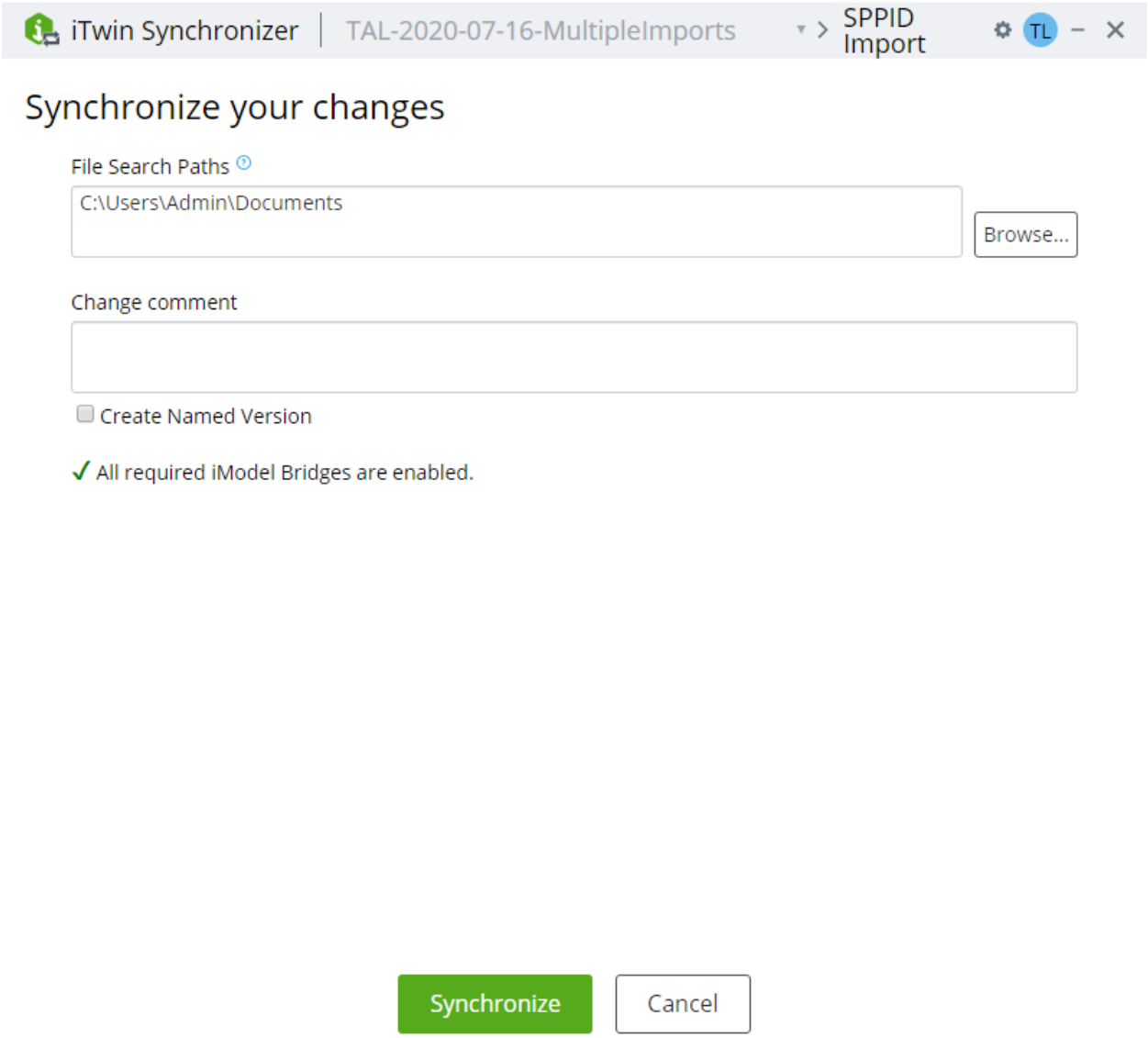
The bridge will be downloaded and extracted.



Click Save to continue once the Bridge was enabled.



Click on the synchronize symbol  to start the synchronization.

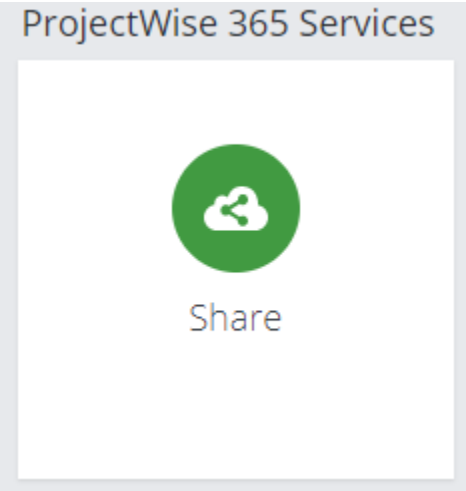


Click Synchronize and the drawings will be converted.

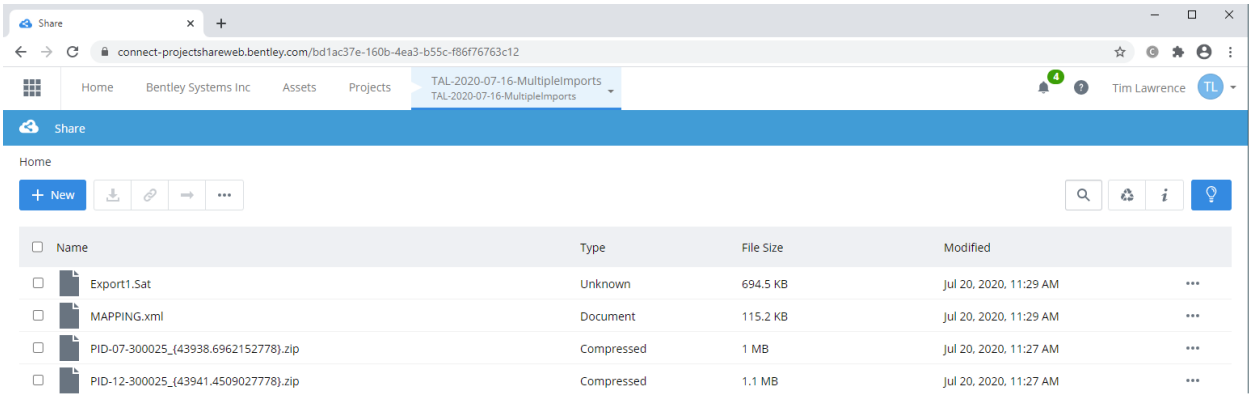
Using Synchronizer from the portal

Create the Project and iModel you wish to use and make sure that it is provisioned using the steps outlined in previous sections.

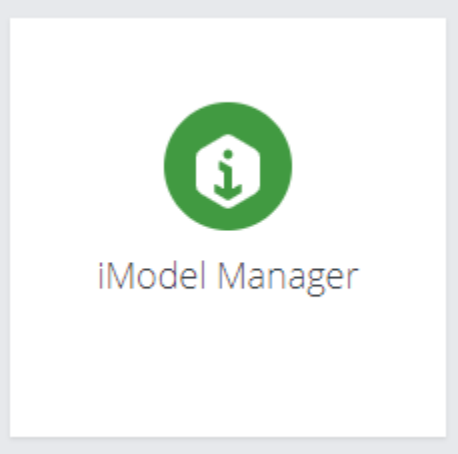
Go to the Share tile on the project you have created:



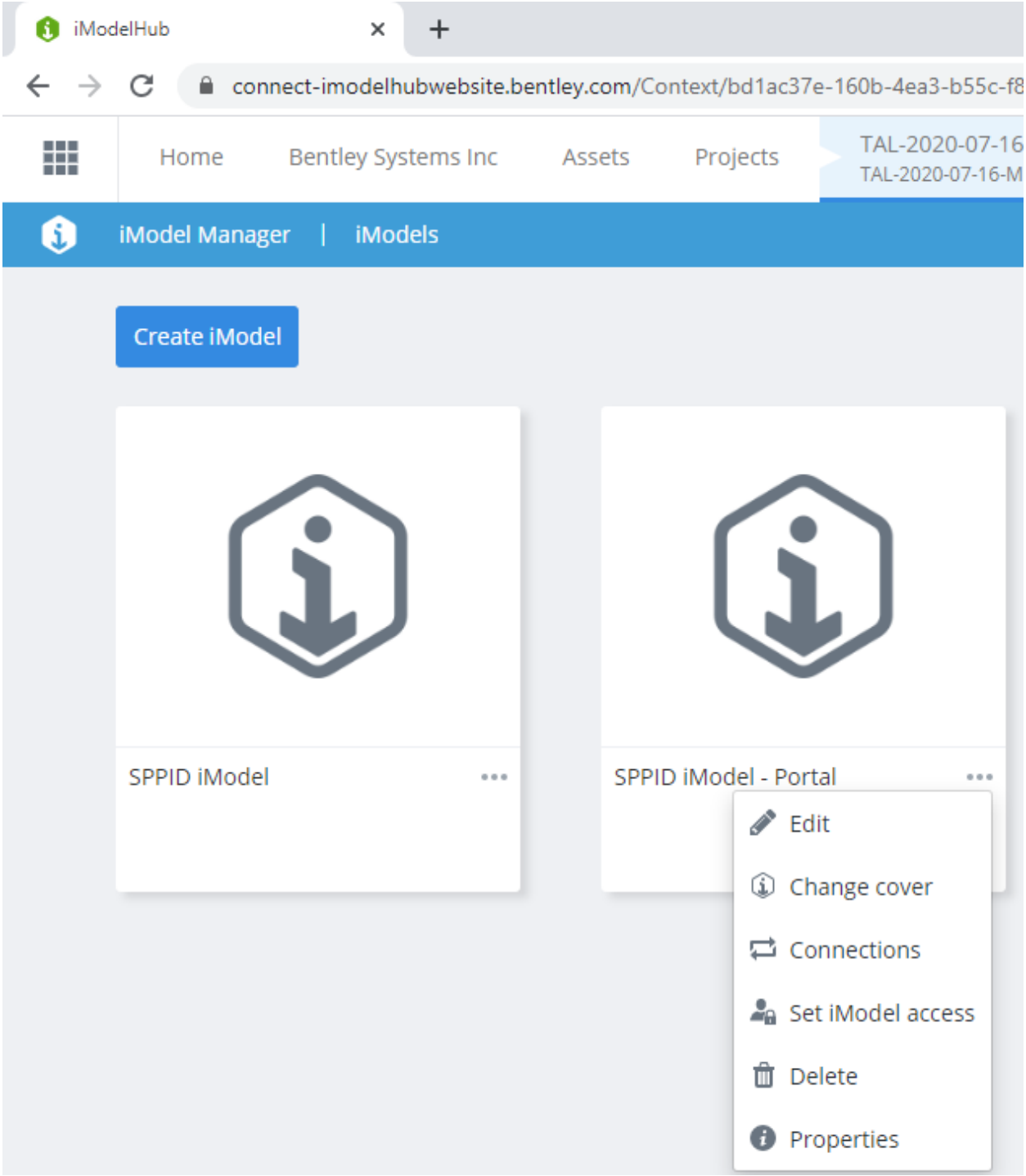
In the ProjectWise Share, add the .zip files that were exported from SmartPlant PID, as well as the Satellite file and if you have made any mapping modifications, that file should also be placed in this folder



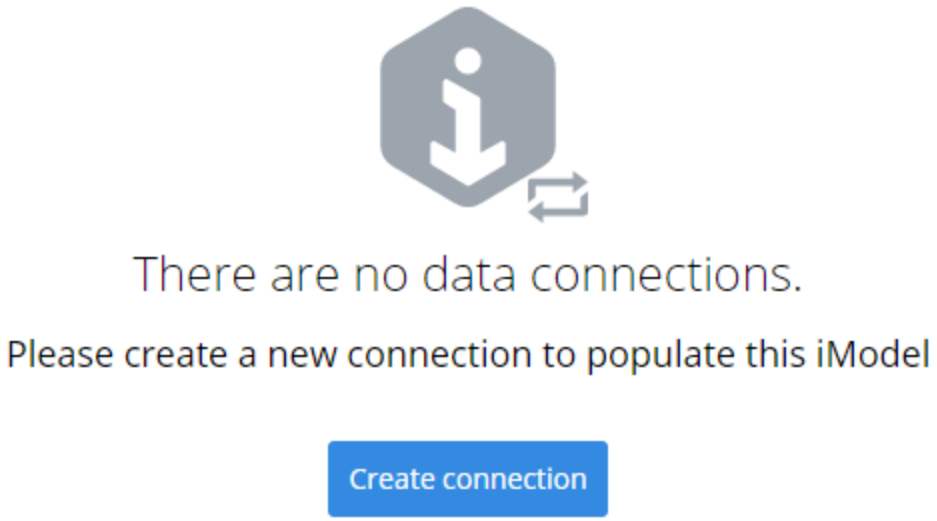
Go back to the main page of this project in the Portal, then select the iModel Manager tile:



Click on the … for the iModel you wish to use, then select Connections

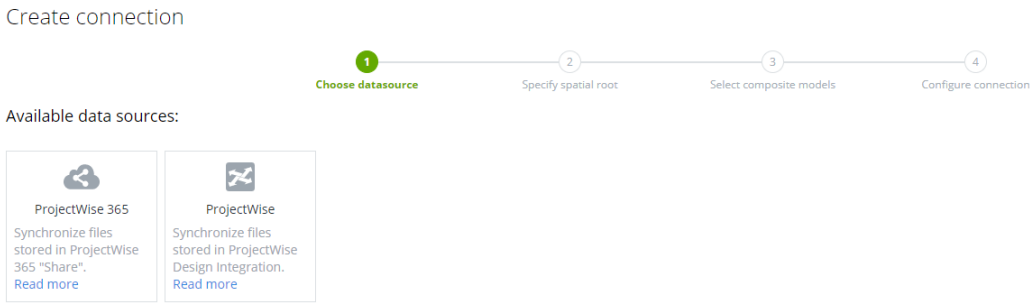


Since this is the first time you will have to create a new connection:



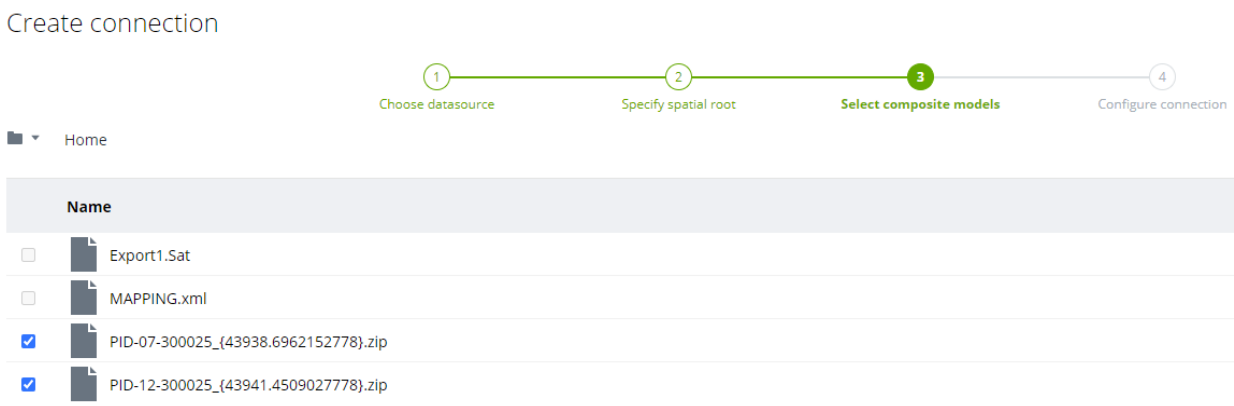
Click Create connection.

Click on ProjectWise 365



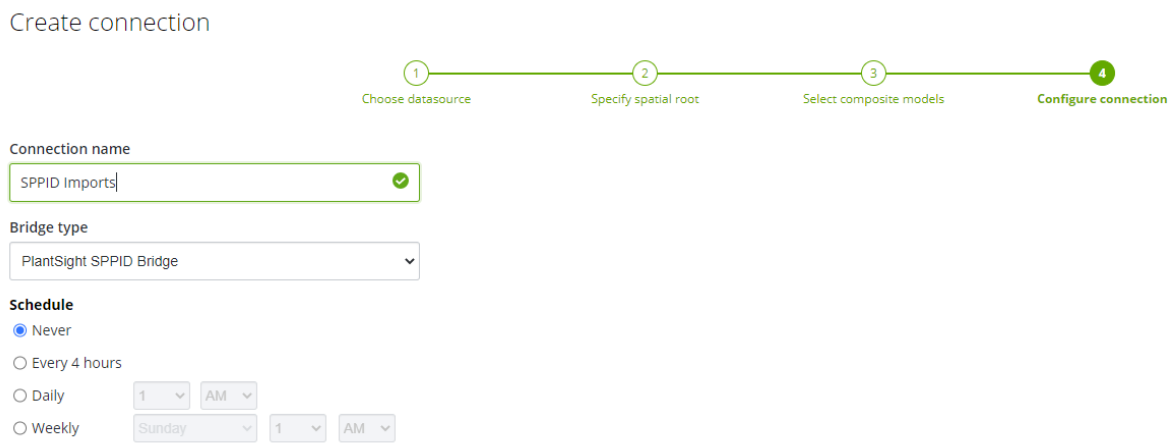
Then Next

At the next step, click Skip as these are 2D drawings and there is no need to set a spatial model.



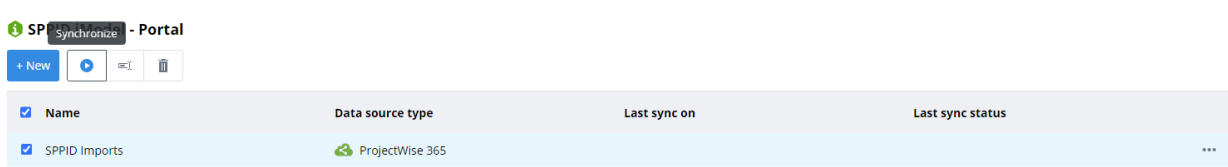
Select the 2 zip files.

Name the import job and if the file type supported multiple bridge types, you would need to select the bridge. Optionally you can also set a schedule for the bridge to run.

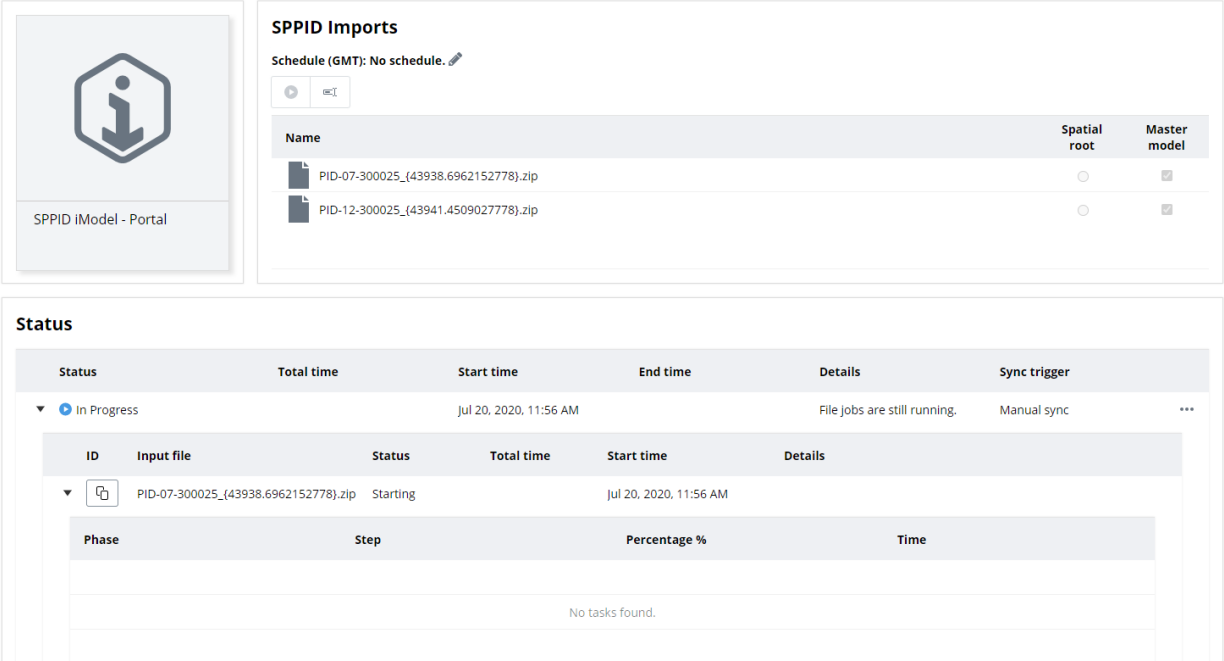


Click Create.

Select the Import job and click the play button to synchronize.



As the jobs run, you can monitor the progress:



Changing the Class and Property Mappings

This is default mapping for classes and properties when you run the AVEVA PID bridge. But you can customize these mappings as you may have different properties in AVEVA PID. To modify the mappings, download this:

SPPID.zip

This zip file contains 2 files:

Mapping.xml

Spec_SPPID.xlsm

Class Mapping

There are two different sheets for mapping of classes, BaseClassMapping and LeafClassMapping.

Base Class Mapping

The BaseClassMapping sheet has a list of all SmartPlant PID "Class name" - s (SPTypes), and the corresponding OpenPlant class name value. Values in a column OPPID "Class name2" can be changed.

SPPID				OPPID
Class name	Base class	IsDisplayed	Conditional property	Class mapped
AreaBreak	PlantItemGroup	1		
AreaBreakAttribute		1		
BoundedShape	Representation	1		
BoundedShapeVertex		1		
Case		1		GENERIC_EQUIPMENT
CaseControl		1		CONTROL_DESK
CaseProcess		1		PROCESS_INSTRUMENT
Connector	Representation	1		
ConnectorVertex		1		PROCESS_CONNECTOR_SYMBOL
Drawing		1		PID_DOCUMENT
DrawingProject		1		
DrawingVersion		1		
DuctingComp	PlantItem	1		INLINE_PIPE_COMPONENT

The "Conditional property Column" value is used to Map components to leaf OpenPlant classes depending on the Property value. If the matching OpenPlant LeafClass is not found, the value from the Base class mapping will be used.

Leaf Class Mapping

Due to the different component hierarchy of SmartPlant PID components, Leaf Class Mapping was added to specify more precise OpenPlant class name than a mapped Base class name.

SPPID				OPPID
Conditional property	Property value	Subclass	Class	Class mapped
EquipmentType	Base plate	Equipment supports	Equipment Component	PLATE
EquipmentType	Conical skirt	Equipment supports	Equipment Component	VESSEL
EquipmentType	Cylindrical skirt	Equipment supports	Equipment Component	VESSEL
EquipmentType	Leg	Equipment supports	Equipment Component	VESSEL
EquipmentType	Saddle	Equipment supports	Equipment Component	VESSEL
EquipmentType	Support	Equipment supports	Equipment Component	VESSEL
EquipmentType	Trunnion	Equipment supports	Equipment Component	VESSEL
EquipmentType	Anode	General equipment components	Equipment Component	
EquipmentType	Boot w/body flange	General equipment components	Equipment Component	
EquipmentType	Boot w/cone head	General equipment components	Equipment Component	
EquipmentType	Boot w/head	General equipment components	Equipment Component	
EquipmentType	Continuation	General equipment components	Equipment Component	
EquipmentType	Demister	General equipment components	Equipment Component	
EquipmentType	Demister Type 1	General equipment components	Equipment Component	
EquipmentType	Demister Type 2	General equipment components	Equipment Component	
EquipmentType	Dome w/body flange	General equipment components	Equipment Component	

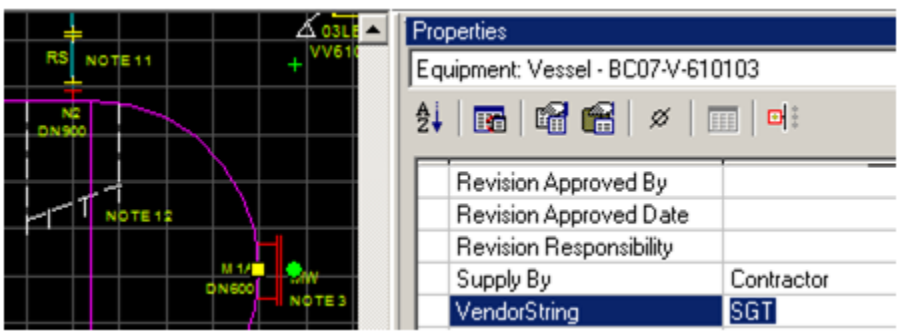
Properties Mapping

The properties mapping sheet lists the properties from the OpenPlant functional schema. The matching property from SmartPlant PID is filled into column D. If the schema is extended, the additional properties can be added to this sheet.

OPPID			SPPID
Class	Base classes	Property name	Property name mapped
PLANT_BASE_OBJECT			
	func:FunctionalComponentElement		
		OpenPlantTypeName	ItemTypeName
		DesignState	ProcessDesign.CaseClass
NAMED_ITEM			
	PLANT_BASE_OBJECT		
		DESCRIPTION	Description
		ALIAS	Name
DEVICE			
	NAMED_ITEM		
		DESIGNER	DesignBy
		SIZE	
		MATERIAL	Material
		ELEVATION	SP_ElevationSI

One SmartPlant property can be mapped to multiple OpenPlant properties by separating them with a semi colon <;>.

PIPE_REDUCER			
	PIPING_COMPONENT		
		DIRECTION	FlowDirection
		DESIGN_LENGTH_END_TO_END	
		SUPPLIER	SupplyBy
		MATERIAL_MARK	PipingMaterialsClass
		WALL_THICKNESS	SP_ScheduleOrThicknessSI
		END_TO_END_LENGTH	
		NOMINAL_DIAMETER_REDUCING_END;REDUCING_SIZE	PipingPoint2.NominalDiameter
		MAIN_SIZE	PipingPoint1.NominalDiameter
		REDUCING_SIZE	



A heuristic description for finding a proper SmartPlantPID property name is described further here:

Example: Some SmartPlant PID EQUIPMENT components have a property VendorString not displayed in the resulting iModel properties. An OpenPlant property name can be chosen to display this information, such as "Manufacturer".

The SmartPlant PID original property names can be taken from the published .ZIP file.

Unpack the published archive and in Drawing.xml, find the exact name of the property by the component SP_ID

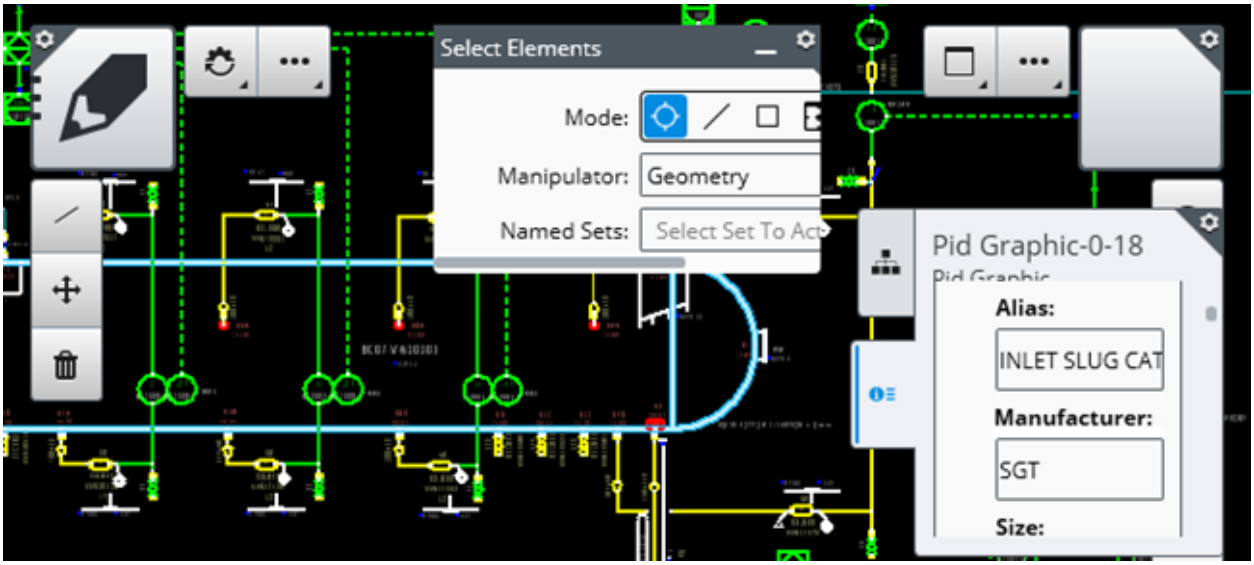
```
53    <SPItem ID="Vessel-888A10D8B14B4AACB6DB8BEC133776D6" SPType="Vessel">
54        <Prop Name="SP_ID">888A10D8B14B4AACB6DB8BEC133776D6</Prop>
55        <Prop Name="LengthTanToTan">16.5</Prop>
56        <Prop Name="SP_LengthTanToTanSI"> 16.5</Prop>
57        <Prop Name="Class" EnumIndex="2">Vessel Equipment</Prop>
58        <Prop Name="EquipmentType" EnumIndex="10033">Hemispherical Head Hor

83        <Prop Name="VendorString">SGT</Prop>
```

Add the "VendorString" value to the SmartPlant PID property name, corresponding to the OPPID property "Manufacturer".

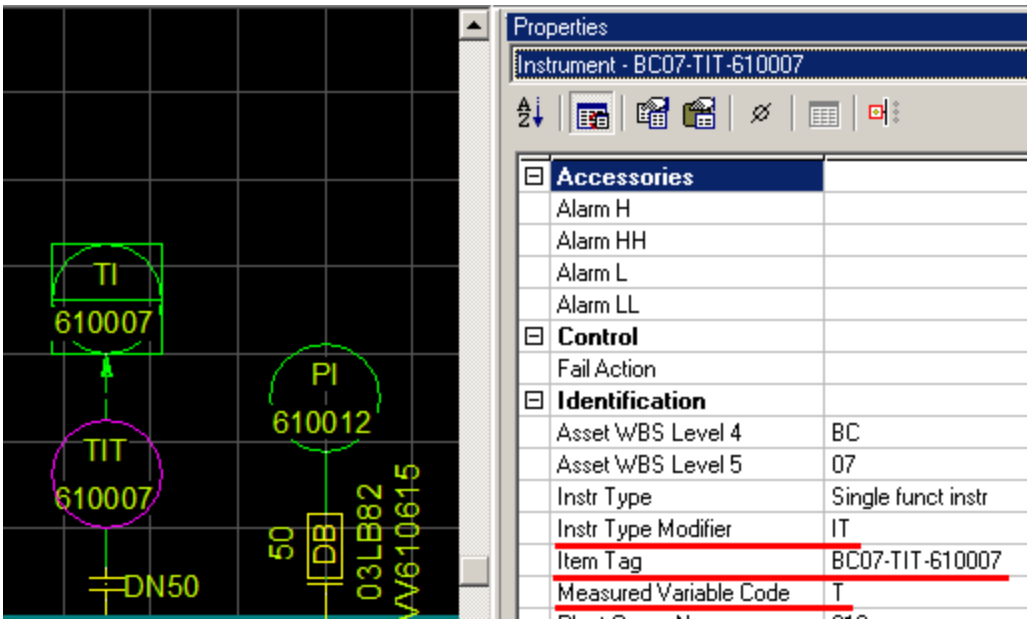
13	DEVICE			
14		NAMED_ITEM		
23			MANUFACTURER	VendorString

Information: Why would a property of a VESSEL be defined in DEVICE properties? That is due to OpenPlant ECClass inheritance hierarchy. In OpenPlant logic, the class incorporates properties of all its base classes. The base OpenPlant class is displayed in the second column of the PropertiesMapping sheet. For instance: Vessel aggregates properties of VESSEL;CONTAINER; EQUIPMENT; DEVICE; NAMED_ITEM; PLANT_BASE_OBJECT classes.

After saving the changes, re-Creating XML and running export - the new property is displayed.



INSTRUMENT Example: Some INSTRUMENTs have properties that are not displayed in the resulting conversion.



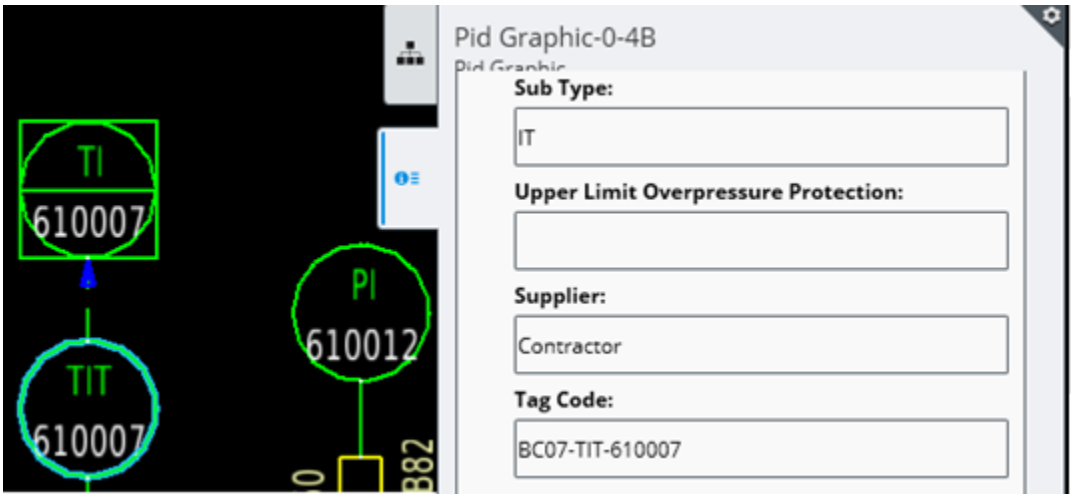
Find the values of these properties by SP_ID objects inthe Drawing.xml file (from published *.zip)

```
157    <SPItem ID="Instrument-0D009ADD50904524B36DBF0EF8A49987" SPType="Instrument">
158        <Prop Name="InstrumentTypeModifier">IT</Prop>

166        <Prop Name="SP_PipeRunID">C97CDE738C7F492293C9EE90DE552C62</Prop>
167        <Prop Name="SPIInstrClass" EnumIndex="1">C</Prop>
168        <Prop Name="ItemTag">BC07-TIT-610007</Prop>
```

Add the SmartPlant PID property names to the mapping, create the XML, and run the conversion.

177	INSTRUMENT			
190		SET_POINT	PointIndex	PointIndex
191		SUB_TYPE	InstrumentType	InstrumentTypeModifier
192		UPPER_LIMIT_OVERPRESSURE_PROTECT	SP_CriticalPressureSI	
193		SUPPLIER	SupplyBy	SupplyBy
194		TAG_CODE		ItemTag
195		PIPE_LINE_NUMBER		
196		PID_NUMBER		
197		HOUSING_MATERIAL	PipingMaterialsClass	PipingMaterialsClass
198		CALIBRATION_DATA_REQUIREMENTS		
199		INTRINSICALLY_SAFE_INSTALLATION		

These SmartPlant PID properties are shown in the iModel Property palette.



The original SmartPlant PID property names can be take neither from Drawing.xml, or from the "FullSPPIDproperties" Excel sheet which is hidden by default. The FullSPPIDproperties sheet contains all possible properties with display names from the SmartPlant PID resources.

Relation Mapping

The relation mapping consists of two sheets: RelationMapping and ConnectByFields. These sheets are used for adding/removing relationships between components.

RelationMapping is based on the OPPIDProcessFunctional ECRelationshipClasses of the OpenPlant Schema. These sheets help to specify relationships: "Relation" name and a pair of "Source" - "Target" OpenPlant ECClass names.

If a value of 1 is set in the "Disable" column, the value is not processed.

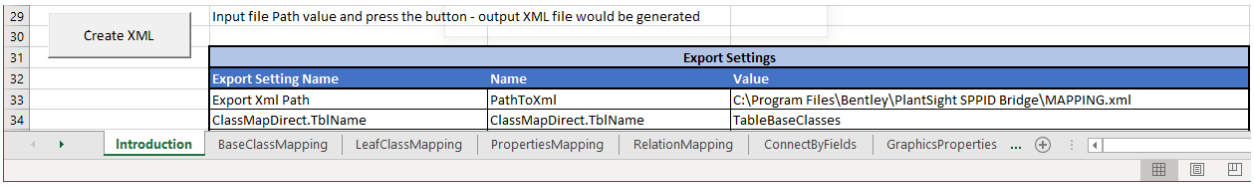
Relation	Source	Target	Disable
LOOP_HAS_INSTRUMENT	LOOP	INSTRUMENT	
LOOP_HAS_OBJECT	LOOP	PLANT_BASE_OBJECT	
NOZZLE_CONNECTS_TO_SEGMENT	NOZZLE	PIPING_NETWORK_SEGMENT	
Object_Connects_To_Object	PLANT_BASE_OBJECT	PLANT_BASE_OBJECT	1
OBJECT_HAS_INSTRUMENT	PLANT_BASE_OBJECT	INSTRUMENT	1
Object_Has_Object	PLANT_BASE_OBJECT	PLANT_BASE_OBJECT	1
Object_Is_Driven_By_Object	PLANT_BASE_OBJECT	PLANT_BASE_OBJECT	1
Object_Is_Related_To_Object	PLANT_BASE_OBJECT	PLANT_BASE_OBJECT	1

The ConnectByFields sheet is for specific cases when objects are connected through a reference property (Component ID). The direction can be Backward (Component with Property will be Source and Class will be Target) and Forward.

OPPID	SPPID			
Relation	Class	Property	Direction	Disable
SEGMENT_HAS_INSTRUMENT	Instrument	SP_PipeRunID	Backward	
SEGMENT_HAS_INSTRUMENT	Instrument	InlineComp.SP_PipeRunID	Backward	
SIGNALLINE_CONNECTS_TO_INSTRUMENT	Instrument	SP_SignalRunID	Backward	
EQUIPMENT_HAS_NOZZLE	Nozzle	SP_EquipmentID	Backward	

Updating Mapping.xml

Once you complete modifications to the mapping sheets, click the Create XML button on the Introduction sheet. This generates the mapping.xml file in the specified path (check that the path exists and that you have write permission to that location). VBA macros are used to create the xml from Excel sheets.



Using the results of the mapping

If you update the mapping.xml and want to use these in your conversions, the mapping file needs to be available for the iModelBridge.

iTwin Synchronizer:

Place the mapping.xml file in the same folder as the zip file that you select

CONNECT Portal:

Place the mapping.xml in the same folder as your .zip files to be converted. This will be either in the ProjectWise Share or folder on ProjectWise.