



**PM 10277**

## **Autodesk Factory Design Suite: Asset Development Checklist**

Rusty Belcher – IMAGINiT Technologies

### **Learning Objectives**

At the end of this class, you will be able to:

- Describe the asset checklist approach for developing factory assets
- Describe the different types of basic assets
- Understand the best practices for asset publishing – Keeping your assets simple
- Understand the importance of testing assets prior to publishing

### **Description**

There are so many things to consider when developing assets for Autodesk Factory Design Suite. Creating functional assets for your factory layouts can be challenging unless you have a detailed checklist for asset development. This class offers a step-by-step approach to developing fully functional assets. It accounts for everything from iProperties to iLogic. Join us as we explore the process of asset development while discussing each step in the checklist.

### **About the Speaker**

*Rusty Belcher is a Manufacturing Application Expert working with IMAGINiT Technologies. Rusty provides implementation, training, and support services at every level for all Autodesk Manufacturing products. His specialty involves the integration of 3D design practices into manufacturing production environments. As an instructor and mentor, IMAGINiT regularly receives outstanding reviews of his impact to their organization.*

*Rusty started his career as a structural steel fitter at Newport News Shipbuilding. He is a graduate of the Newport News Shipbuilding Apprentice School and worked in the shipyard's Mold Loft engineering division.*

*Over the past several years Rusty has worked directly with Autodesk to develop and author the current Factory Design Suite training courseware, has also developed, and recorded many of the tips and tutorial videos available on the Factory Design Suite YouTube Channel. He is currently working with Autodesk to develop the Factory Design Suite and Product Design Suite Test Drive for the 2014 release.*

## The Asset Checklist

The true power of Factory Design Suite is the ability for users to create their own library of custom assets that represent the unique machines and capabilities they provide. These assets can be simple representations of equipment or complex parametrically enhanced machines that can be adjusted to any size during placement.

The asset development process can be fairly simple or quite elaborate depending on the asset in question and the desired functionality. There are many options to consider when developing factory assets. It is quite common to test a recently published asset and discover that critical data or functionality was not included.

In order to simplify or standardize the asset development process, I developed a checklist that I use each time I generate a new asset. The checklist serves as a reminder of critical points that must be addressed during the asset development process.



## The Ever Changing Checklist

The asset development checklist changes all the time. The checklist shown on the following page is a good starting point for new users. Once you have the basics of asset development down you can add or remove points on your own version of the checklist. Autodesk is constantly improving the capabilities of the Factory Design Suite. As functionality is added to FDS, your asset development checklist should change accordingly.

Checklist for Asset Publishing	
<input type="checkbox"/>	Model Detail – Low
<input type="checkbox"/>	Shrinkwrap if Necessary
<input type="checkbox"/>	Multi-Body if Necessary
<input type="checkbox"/>	Parameters - Named
<input type="checkbox"/>	Parameters - Key
<input type="checkbox"/>	Parameters - Exported
<input type="checkbox"/>	Parameters – Multi-Value
<input type="checkbox"/>	Parameters - Tested
<input type="checkbox"/>	iProperties – Part Number
<input type="checkbox"/>	iProperties – Description
<input type="checkbox"/>	IProperties – Mapped Parameters
<input type="checkbox"/>	Work Features for Publishing - On
<input type="checkbox"/>	Landing Surface
<input type="checkbox"/>	Insertion Point
<input type="checkbox"/>	Define Connectors
<input type="checkbox"/>	Connector Class Properties
<input type="checkbox"/>	Asset Properties
<input type="checkbox"/>	Asset Variants
<input type="checkbox"/>	Asset Descriptor
<input type="checkbox"/>	iLogic
<input type="checkbox"/>	All Work Features Off
<input type="checkbox"/>	Publish Local
<input type="checkbox"/>	Test
<input type="checkbox"/>	Modify to Suit Testing
<input type="checkbox"/>	Publish Local - Final
<input type="checkbox"/>	Publish to Cloud - Optional

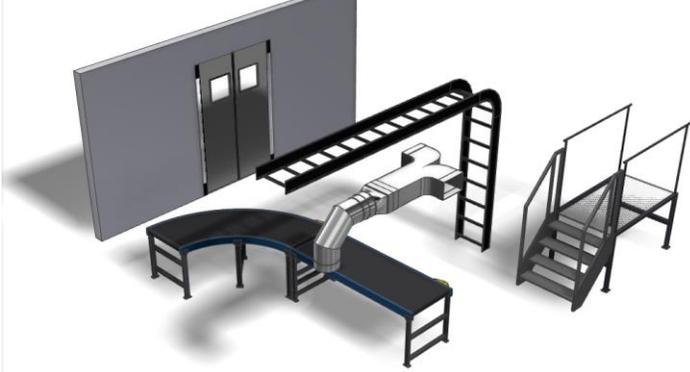


## Asset Types – Static and Dynamic

Assets come in many different shapes and functional forms. The exact process you use to publish them will vary based on the type of asset and the desired functionality. There are two basic types of assets, **Static** and **Dynamic**. In this section, we will discuss the basic process for publishing each of these asset types.

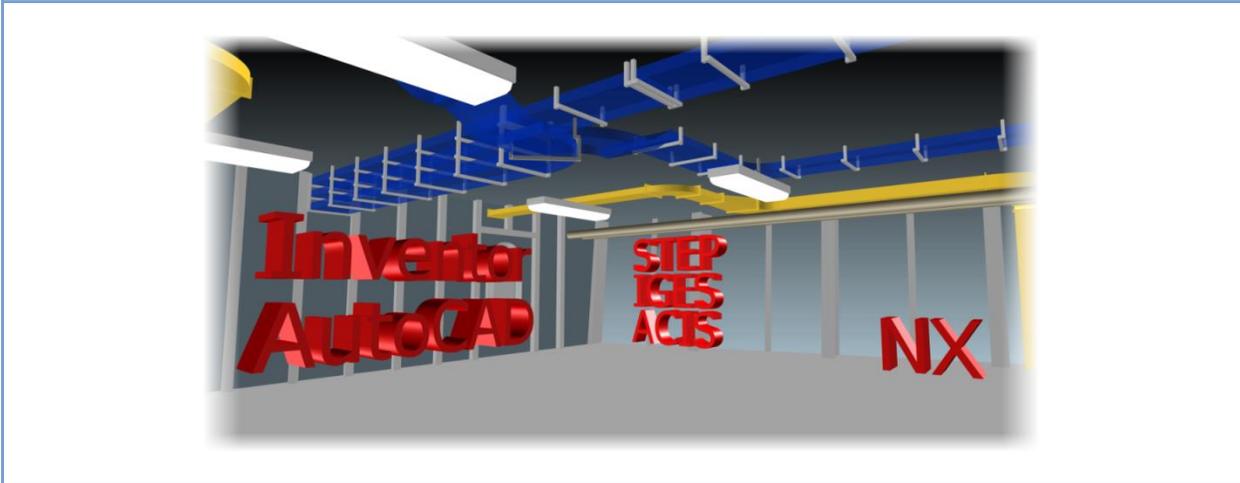
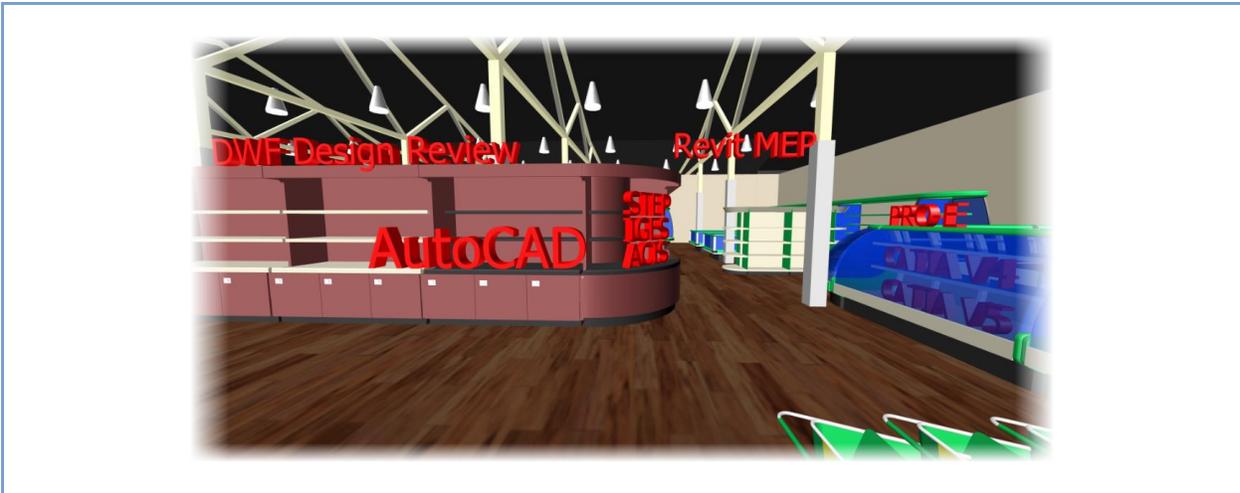
**Static Assets** represent a single unchanging form or shape. Equipment and machines that never change shape can be modeled and published as a simple static asset. It is also possible to download models from numerous CAD sources and use these designs as static assets.

**Dynamic Assets** are designed to parametrically represent forms that will consistently change. Equipment that adjusts in length, width, or height each time it is inserted can be published as a dynamic asset. Dynamic assets must be developed with Autodesk Inventor to include the desired parametric functionality.

Asset Type	Examples
Static Assets	
Dynamic Assets	

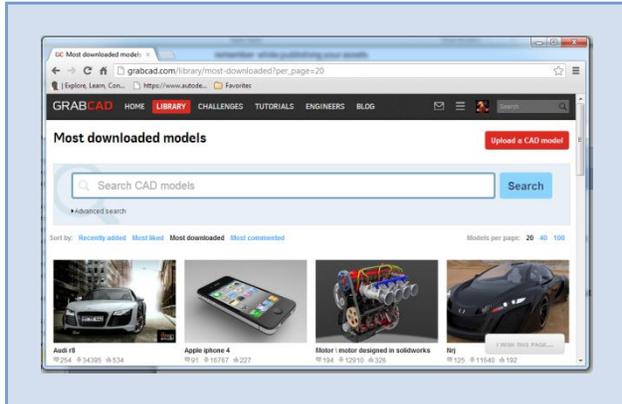
**Asset Origins**

It is important to remember that almost any 3D model can become an asset. The Factory Design Suite allows you to open models from various CAD sources and add the necessary connector points and landing surfaces required to utilize the asset in your layout. This makes online model sites like **GrabCAD** and **PartSpec** perfect places to download models for potential assets.



The Factory Design Suite can utilize almost any 3D model as an asset regardless of the authoring CAD system. The Navisworks component of FDS allows you to append CAD data from even more systems.

The table below offers a few online locations where 3D content is available for free download. Any typical solid model is can usually be used for a FDS asset. Mesh models from tools like **SketchUP** can only be used in the Navisworks component of FDS.



**GRABCAD** is a great place to download models from various CAD sources.



**CAD Register.com** allows you to access thousands of vendor specific CAD designs.



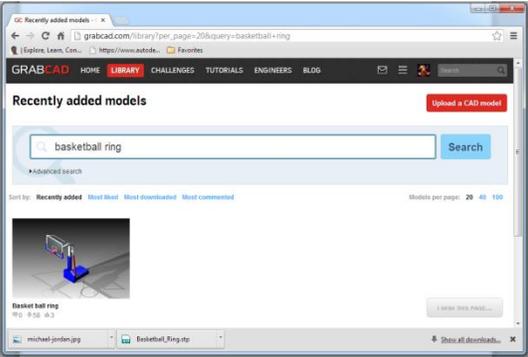
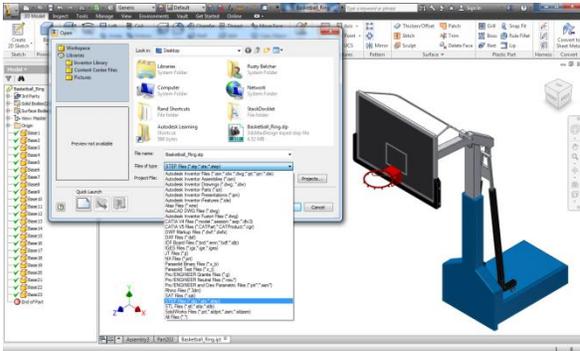
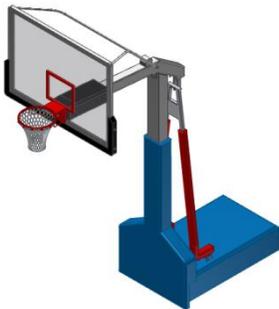
**CBLISS.com** is a great place to download Inventor centric parts and iParts. Please remember that you cannot create an asset from an iPart, but you can utilize it as the basis of an FDS asset.

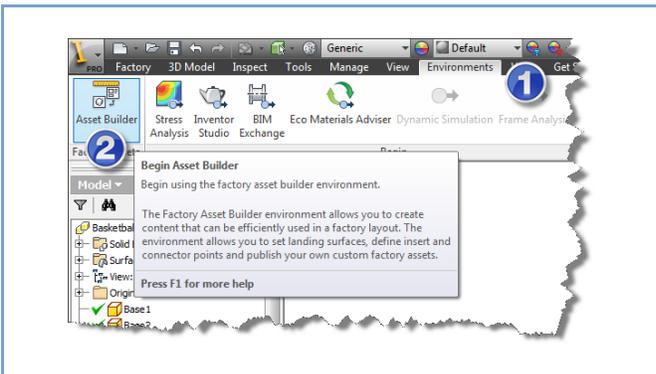


**Trimble 3D Warehouse** (formally Google 3D Warehouse) is a great place to download SketchUP models. These files **cannot** be used as assets but can be used in the Navisworks component of the Factory Design Suite.

**Creating a Static Asset - Downloaded Design**

In this section, we will outline the process of converting a downloaded model into a layout asset. The following table will show the steps necessary to convert a model, downloaded from **GRABCAD**, into a FDS asset.

	<p><b>Download the Model</b></p> <p>The process begins with locating the desired design from one of the many internet locations that supply 3D models. In this example we will download a Basketball Hoop STEP file from <b>GRABCAD.com</b></p>
	<p><b>Open the Model with Inventor</b></p> <p>Inventor allows you to open model files from various CAD sources including the many of the leading modeling products on the market today.</p>
	<p><b>Make Adjustments</b></p> <p>Use Inventor to make any necessary adjustments. In this example, some of the original colors were updated to produce the desired result.</p>



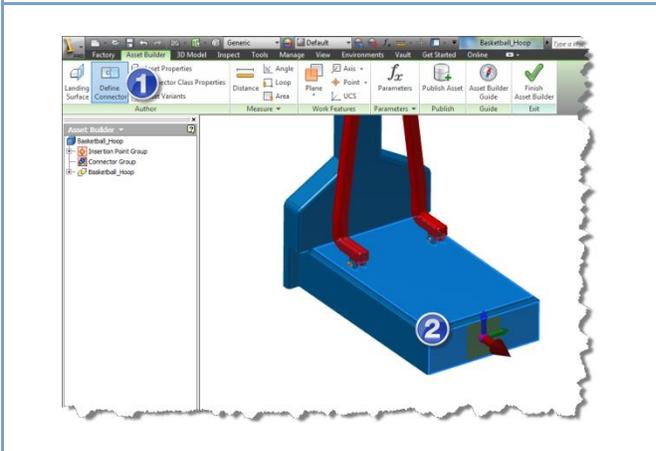
### Open the Asset Builder

When the file is ready to be published, activate the Environments tab (1) and click the Asset Builder (2).



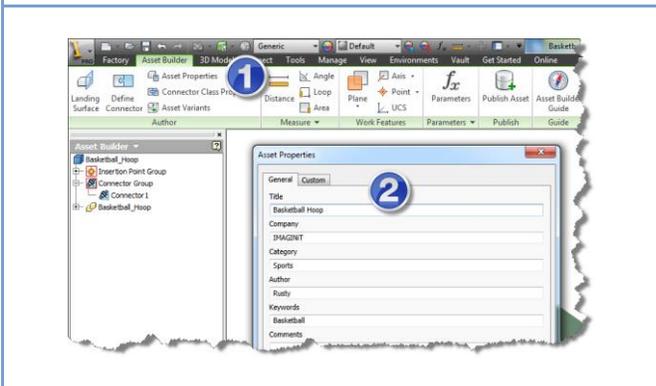
### Set the Landing Surface

On the Asset Builder ribbon, click the Landing Surface command (1). Then select the face that will rest against the floor (2). If necessary, click the Insertion Point option (3) and click the desired point you wish to insert the asset by.



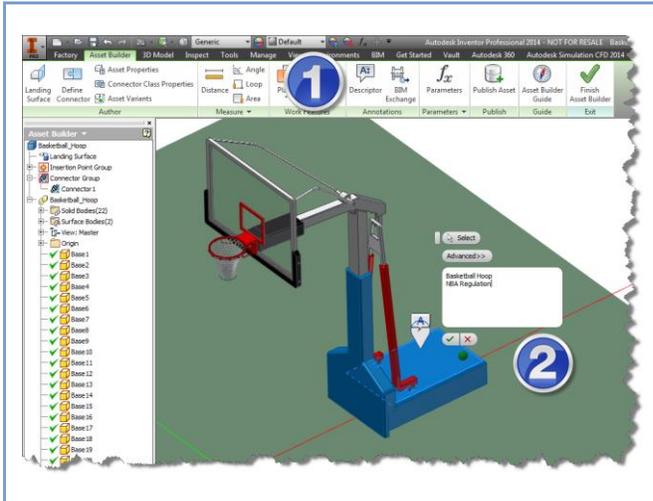
### Define Connectors

On the Asset Builder ribbon, click the Define Connector command (1). Select a face or vertex to use as a connector point (2). Align the blue axis of the triad to the upward direction by selecting the axis and then selecting a face or edge. Right click and select OK to complete the command.



### Define the Asset Properties

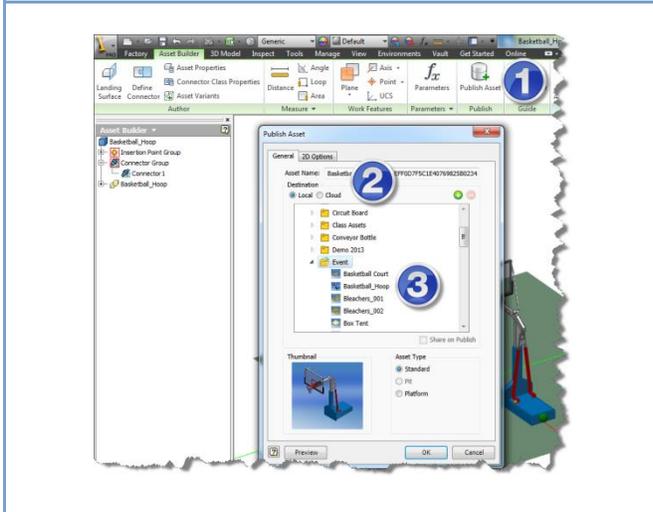
On the Asset Builder ribbon, click the Asset Properties command (1). Fill in the necessary file properties in the dialog (2). Click OK to complete the command.



### Asset Descriptor – (Optional)

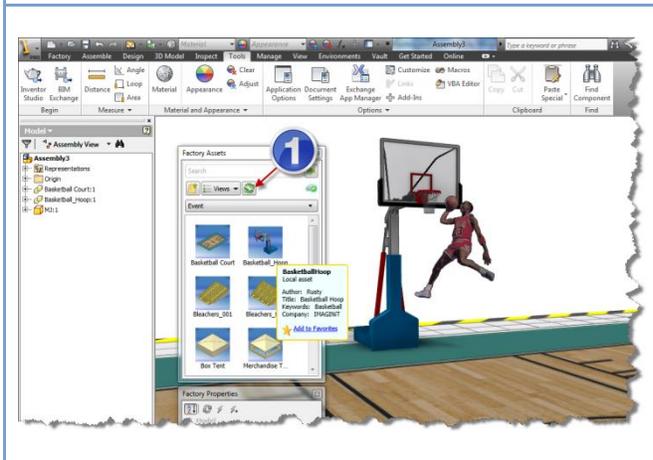
On the Asset Builder ribbon, click Descriptor (1). Enter the description desired in the field provided (2). Click the checkmark to finish the command.

**Note:** The text entered in the asset descriptor will display as attributed text in AutoCAD after synchronizing.



### Publish the Asset

On the Asset Builder ribbon, click the Publish Asset command (1) and save your model if prompted. In the Publish Asset dialog, select the Local option (2) to publish the asset to your local library or select the Cloud option to publish and share the asset in the Asset Warehouse. Then you will create or select the subdirectory to save the asset in (3). Click OK to publish the asset.



### Test your Asset

Start a new layout and select the Refresh command on the Factory Asset browser (1). Then place your asset to test its functionality.

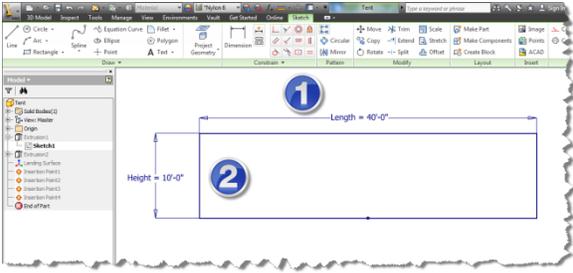
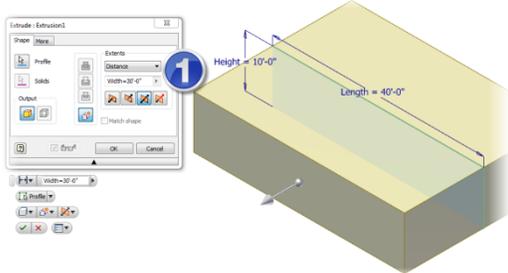
### ***Inventor Based Assets - Parametric Design***

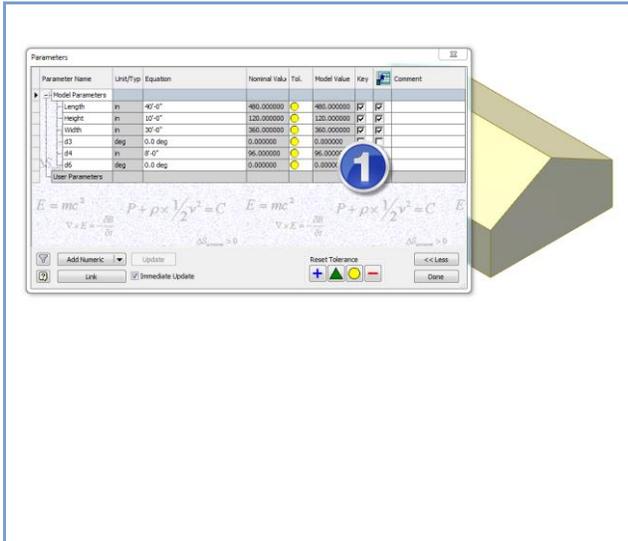
Creating assets from native Inventor models offers many additional advantages and functionalities. You can use named parameters to control key features. These parameters can be marked for use by FDS when the asset is placed in a layout. The factory properties palette allows you to easily modify any key parameter updating the asset automatically.

It is also important to note that typical Inventor parametric practices are all that is needed to create fully functional assets. iLogic can be used to provide additional functionality, but it is not required by default. Anyone familiar with basic Inventor parametric modeling can easily produce layout assets.

### ***Creating a Dynamic Asset - Parametric Design***

In this section, we will outline the process of developing assets for your layout design. In this example, we will be creating an asset from scratch utilizing the full power of Inventor.

	<p><b>Example</b></p> <p>In this example, we will be creating an asset for one of the tents used for an event-planning layout. The asset will include parametric control over the <b>Length</b>, <b>Width</b>, and <b>Height</b> of the tent.</p>
	<p><b>Include Parameters in Sketch</b></p> <p>Parameters are included in the initial sketch to define the Length and Height of the tent. Instead of entering a static number for the dimension value, a simple equation is entered <b>ex. Length=40'</b>.</p>
	<p><b>Include Parameters in Features</b></p> <p>A parameter is also included in the extrusion feature. The distance of the extrusion is set to <b>Width=30'</b> (1).</p>

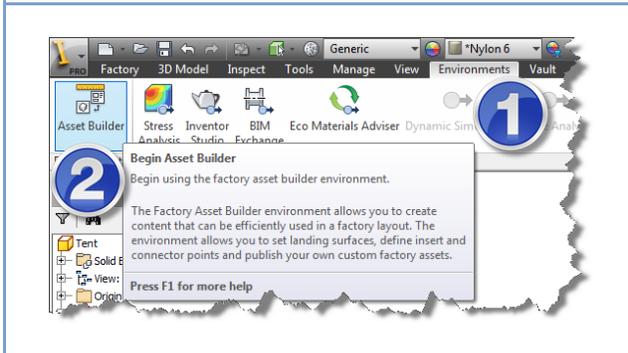


### Mark the Key Parameters

In the Parameters dialog, the Length, Width, and Height parameter are marked as **Key** (1).

**Note:** Test the Parameters to ensure proper functionality.

**Note:** Mark the parameters for **Export** to use the values in downstream processes like asset description, drawing titleblocks, or assembly bill of materials.



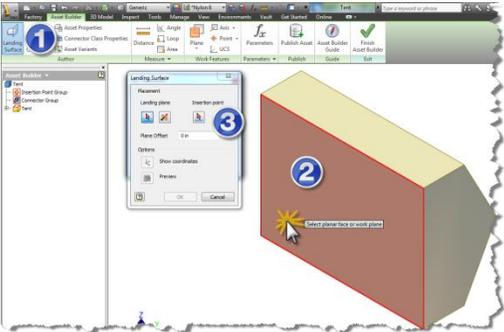
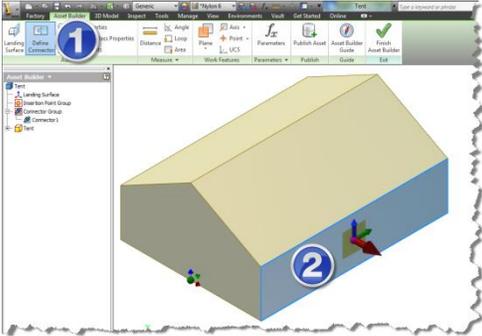
### Open the Asset Builder

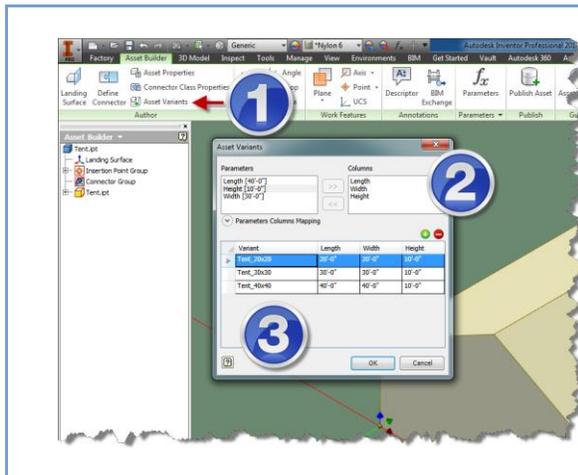
When the file is ready to be published, activate the Environments tab (1) and click the Asset Builder (2).



### Check your Key Parameters

When you enter the Asset Builder, the Key Parameters dialog is displayed showing your renamed parameters. Make sure to check the parameters you wish to use to control the size of the final asset.

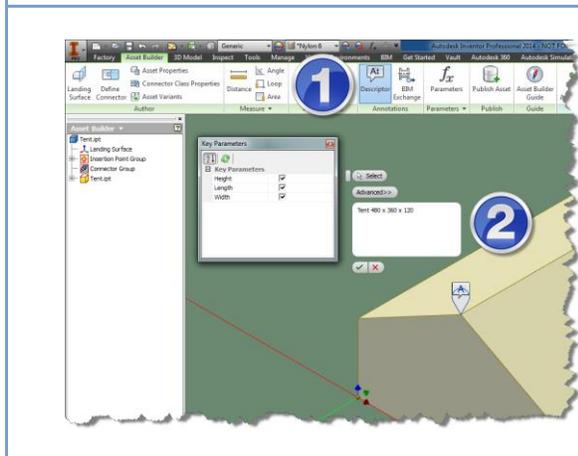
	<h3>Set the Landing Surface</h3> <p>On the Asset Builder ribbon, click the Landing Surface command (1). Then select the face that will rest against the floor (2). If necessary, click the Insertion Point option (3) and click the desired points you wish to insert the asset by.</p> <p><b>Note:</b> you can have multiple Insertion Points.</p>
	<h3>Define Connectors</h3> <p>On the Asset Builder ribbon, click the Define Connector command (1). Select a face or vertex to use as a connector point (2). Align the blue axis of the triad to the upward direction by selecting the axis and then selecting a face or edge. Right click and select OK to complete the command.</p>
	<h3>Define the Asset Properties</h3> <p>On the Asset Builder ribbon, click the Asset Properties command (1). Fill in the necessary file properties in the dialog (2). Click OK to complete the command.</p>



### Asset Variants - Optional

On the Asset Builder ribbon, click Asset Variants (1). Move the necessary Parameters to the Columns field (2). Add rows defining the desired asset variants (3). Click OK to finish.

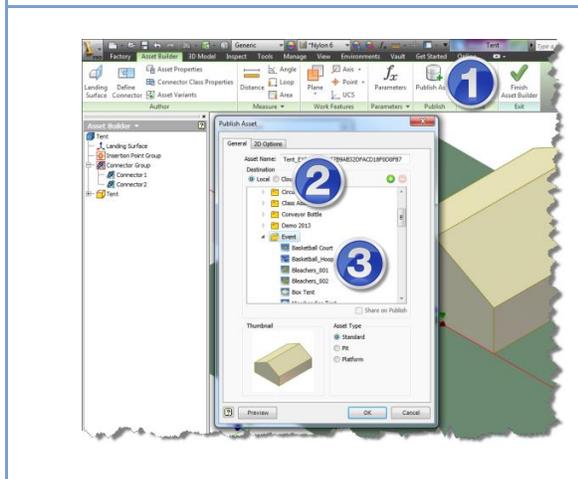
**Note:** Each asset variant will be available to insert into the 3D Inventor layout as well as the 2D AutoCAD layout.



### Asset Descriptor – (Optional)

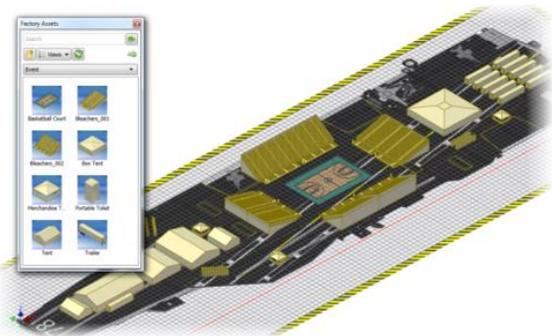
On the Asset Builder ribbon, click Descriptor (1). Enter the description desired in the field provided (2). Click the checkmark to finish the command.

**Note:** The text entered in the asset descriptor will display as attributed text in AutoCAD after synchronizing.



### Publish the Asset

On the Asset Builder ribbon, click the Publish Asset command (1) and save your model if prompted. In the Publish Asset dialog, select the Local option (2) to publish the asset to your local library or select the Cloud option to publish and share the asset in the Asset Warehouse. Then you will create or select the subdirectory to save the asset in (3). Click OK to publish the asset.



The image shows a 3D perspective view of a factory floor layout. A 'Factory Assets' browser window is open on the left side of the screen. The browser window contains a grid of asset icons, including 'Material Cart', 'Machine\_001', 'Machine\_002', 'Bin Tray', 'Worktable 1', and 'Worktable 2'. The 3D view shows several of these assets placed on a grid floor, with yellow and black safety lines indicating aisles or work zones.

### Test your Asset

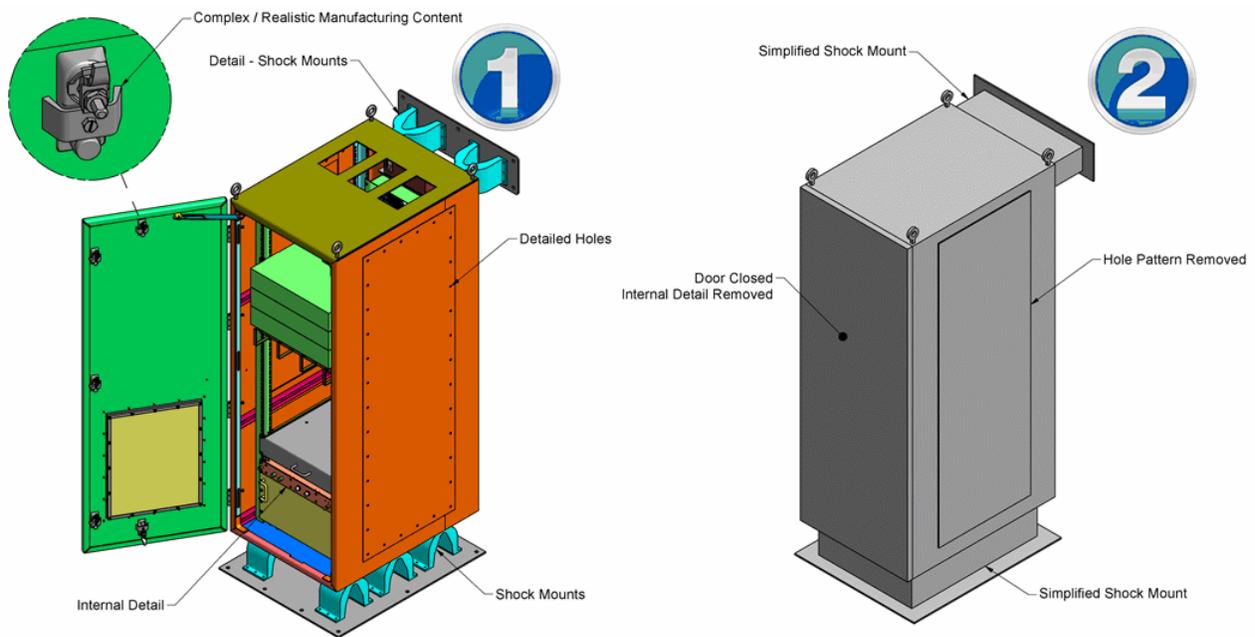
Start a new layout and select the Refresh command on the Factory Asset browser (1). Then place your asset to test its functionality.

## Asset Creation – Best Practices

Assets can provide many valuable downstream benefits, but they must be published in a specific manner with the required settings and metadata in order to provide the maximum performance. This section serves as a recommended “best practice” recommendation for publishing assets in the Autodesk Factory Design Suite.

### Model Detail - Low

It is important to remember that the factory layout process often requires the placement of hundreds or thousands of Assets. For this reason, it is necessary to publish assets with the simplest model detail possible. The image below shows two representations of the same design. The image on the left shows a highly detailed production model of a server cabinet. This assembly contains over 700 individual model elements. The image on the right is a single part file that represents the design envelope of the server cabinet. The simpler form should be used to generate a factory asset.



EX. 1	700 Part Assembly – Production Detail	Not Recommended
EX.2	Single Part – Low Detail	Recommended

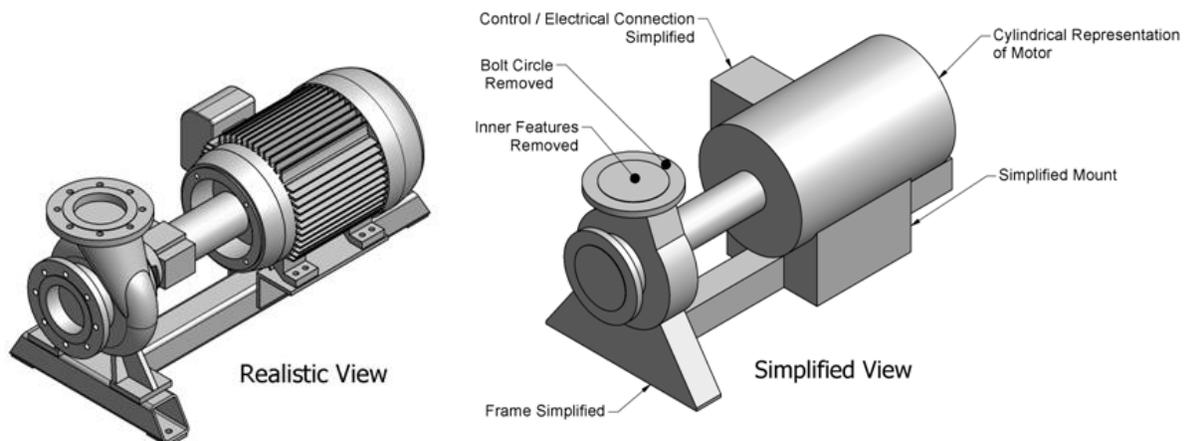
### ***Simplified Level of Detail***

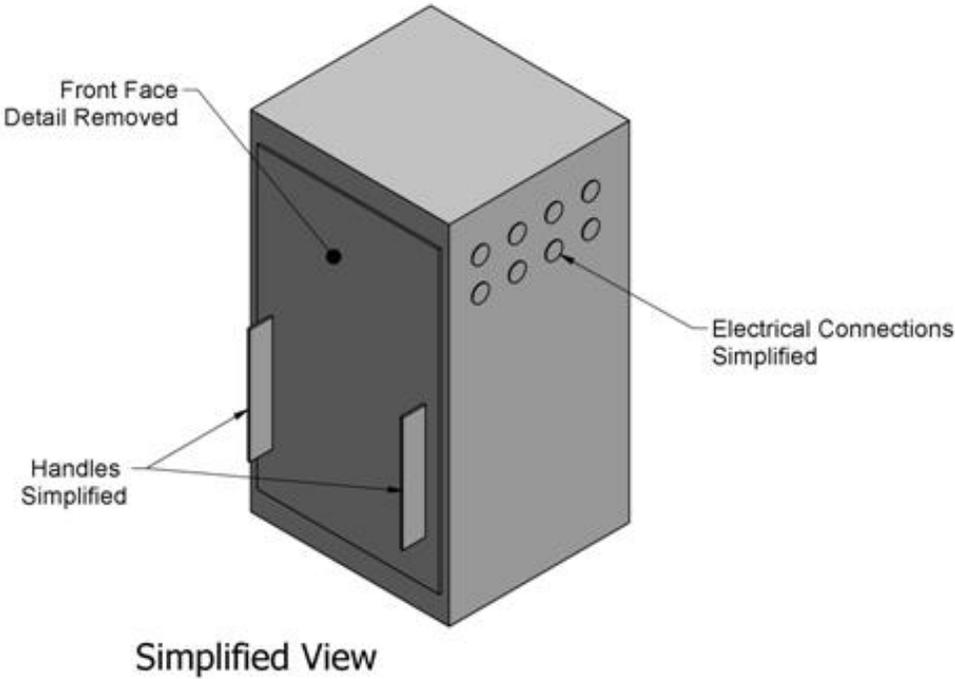
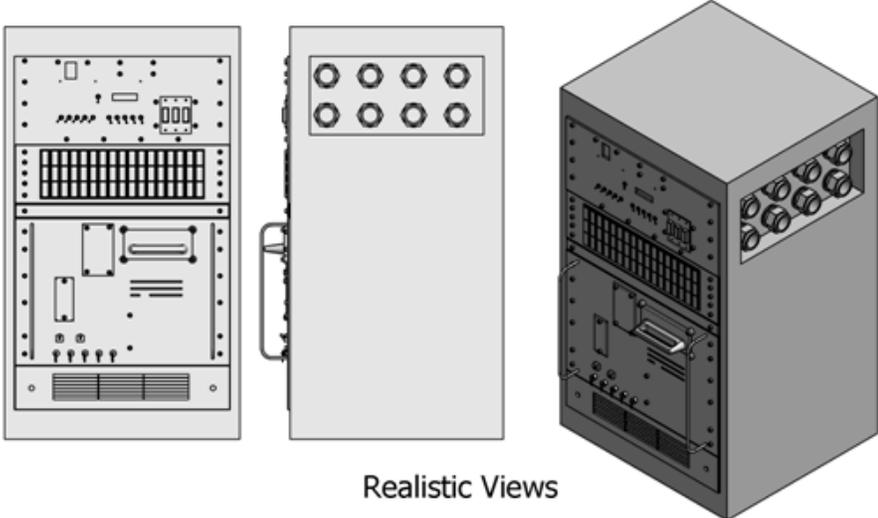
The more features your assets contain, the more graphic resources they require. In addition, your factory design could contain hundreds or thousands of assets. For this reason, it is a recommended practice to generate assets in a simplified form with a minimal amount of detail. Assets should represent the design envelope of the actual component and deliver the necessary design information for the layout drawing.

Several general practices should be followed to simplify the design representation and reduce the model file size. Please note these practices are general guidelines and may be adjusted or ignored in certain situations. The best practices for modeling simplified representations are as follows.

- Fit, Form, and Function – Asset components must be modeled as accurately as possible. They must also be modeled to the simplest representation possible. The general guideline of Fit, Form, and Function should be followed. This term is sometimes referred to as F3. In manufacturing and design industries the term refers to the description of an item's identifying characteristics. If the specifications and criteria of an item's Fit, Form and Function are met; all other features are considered extraneous.
- Eliminate Fillets, Chamfers, and Tapers whenever possible.
- Eliminate Holes whenever possible. Some holes are necessary for proper attachment and location in the final layout design.
- Eliminate Shells – Remove all inner features that are not necessary.
- All Contact faces and edges should be modeled to the simplest form possible.
- Clearance and Maintenance spaces can be modeled as separate solid bodies using a transparent color.
- Assemblies should be modeled as, or converted to single components.
- All Adaptive features should be removed from equipment models.
- Do not include fasteners in assembly designs.

The following examples illustrate the realistic view of a component or assembly, and the recommended simplified form that should be modeled for the Factory Asset Library.



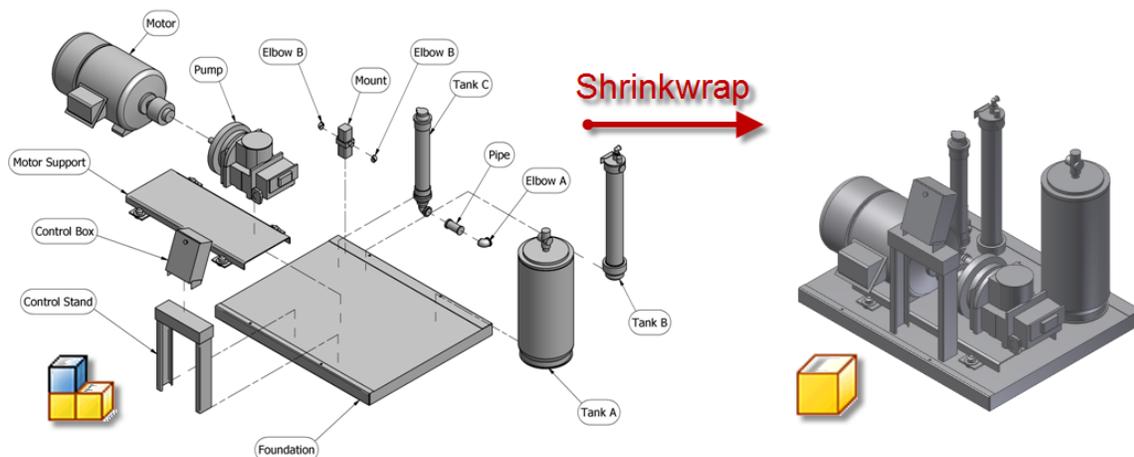


**Remove Model Detail – Convert Production Models or Create New Models**

It is common for designers who are adopting FDS to create their first assets from Production Detailed models they have created in the past. While there are tools that allow you to remove or eliminate detail from an existing model, you must consider the time and effort required to complete this task and compare it to the time required to create a simple version of the asset from scratch. It is often quicker and easier to create a new version of the asset from scratch.

There are tools available that allow you to convert assemblies to single parts or remove voids and detail from existing models. A list of these commands and tools is provided below.

Tool	Function	Location
<b>Shrinkwrap</b>	Converts assembly to single part and removes voids and unnecessary detail.	Inventor Command Located on the Assembly Ribbon.
<b>Multi-body Solid – Import Option</b>	When you Open (Import) a model from an external source, there is an import option that allows you to import assemblies as Multi-Body Parts.	Option in Open (Import) Dialog.
<b>Inventor Simplification</b>	Inventor Simplification is a new environment available in Inventor 2014. The workflows available allow you to generate a simplified version of your complex design from bounding boxes or bounding cylinders.	Available on the simplify tab.



**Parameter Requirements**

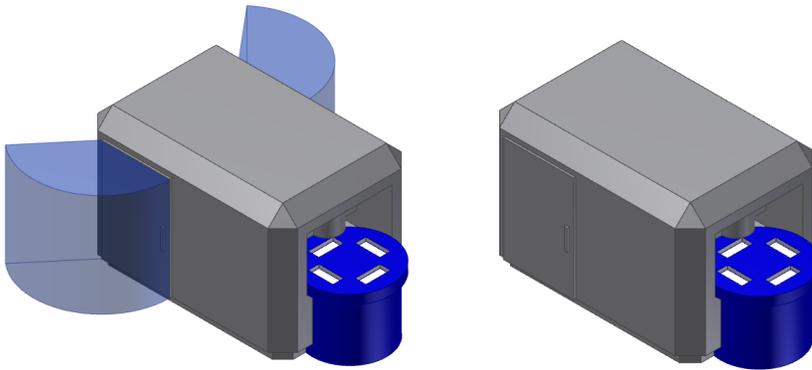
Parameters play a crucial role in the overall function and downstream benefit of factory assets. It is critical that the defining parameters of the asset follow the following guidelines.

<b>Named Parameters</b>	The parameter name should clearly indicate the design value it controls. It is important to note, that the name of the parameter is displayed in the Asset Properties dialog box and is will be utilized by all downstream users to update the asset as needed.
<b>Key Parameters</b>	Only Parameters marked as key are displayed in the Asset Properties dialog. You can mark key parameters in the Inventor Parameters dialog, or during the Asset Publishing Process.
<b>Export Parameters</b>	If you wish to utilize the parameter values in downstream parts lists, or titleblocks, it is necessary to mark the required parameters for export. This option is important for downstream functionality. It is recommended to always mark your key parameters for export.  <b>Note:</b> The Assets provided by Autodesk <b>DO NOT</b> have the key parameters marked for export.
<b>Multi-Value Parameters</b>	If you wish to restrict certain parameters to a limited number of specific values, convert them to Multi-Value parameters. Multi-Value Parameters were introduced as part of the iLogic integration and provide an easy an efficient method to enforce standard values on downstream users.
<b>Testing Parameters</b>	It is recommended to test all parameters prior to publishing to ensure model updates occur as desired.



**Create Clearance Zones Using Parameters**

Clearance and maintenance spaces can be modeled as separate solid bodies and displayed with a transparent color. The appearance of the clearance zones can be controlled by establishing a simple On / Off parameter.



	<p><b>Creating a simple On / Off Parameter</b></p> <ul style="list-style-type: none"> <li>▪ Create a Numeric Parameter setting the unit to unitless and the number to one.</li> <li>▪ Make the value a Multi-Value and add 0 as an alternative.</li> </ul>
	<p><b>Set Suppression for a Feature by Parameter</b></p> <ul style="list-style-type: none"> <li>▪ In the Browser, Right Click the desired feature and select Properties from the menu.</li> <li>▪ In the Feature Properties dialog, set the Suppress option to IF.</li> <li>▪ Select the On / Off Parameter from the drop down list.</li> <li>▪ Set the argument to Not Equal.</li> <li>▪ Enter the value of 1.</li> <li>▪ Click Ok.</li> </ul>

### *iProperties*

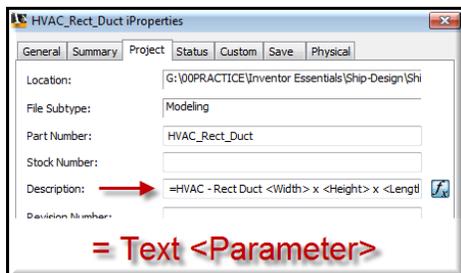
Each Inventor component is equipped with an extensive set of iProperties. The information contained in these properties is utilized in many downstream processes such as Bill of Materials, Parts Lists, Balloons, and Titleblocks. For this reason, it is crucial that the required iProperty fields are filled out prior to asset publishing. The following table indicates the recommended iProperty fields that need to be filled out for each factory asset.

iProperty	Information
<b>Title</b>	Formal Name or Part Name
<b>Part Number</b>	Automatically set to the filename by default
<b>Description</b>	General description of the asset. Could be mapped to parameters like Length, Width, and Height.
<b>Company</b>	Very Helpful for Cloud Based Assets
<b>Category</b>	Your Cloud Based Assets are managed by Categories.
<b>Author</b>	That's You
<b>Keywords</b>	Very Useful for Searching Cloud Based Assets.
<b>Comments</b>	Useful Tips on using the Asset.

### *iProperties and Mapped Parameters*

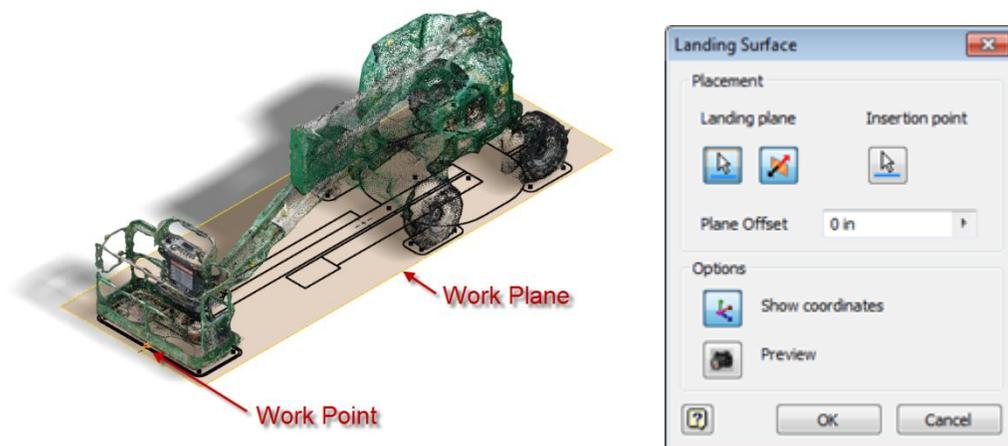
Some fields in the iProperties contain information derived from the various parameter values used to create the asset. It is possible to map parameter values to iProperty fields ensuring the iProperty information reflects the accurate settings for each iteration of the asset.

In a prior step, the practice of marking key parameters for export was recommended. One major benefit of exporting parameters is that they can be mapped to various iProperty values such as Description. To map an exported parameter to an iProperty, you simply enter the parameter name in the formula as shown in the following image.



**Work Features Required for Publishing - ON**

It is occasionally necessary to establish Work Points to use as Insertion Points or Work Planes to use as Landing Surfaces. In these situations, the required work features should be visible (turned on) at the beginning of the publishing process. In the following image, a work point and work plane were established in order to use a point cloud as a factory asset.

**Landing Surface**

Components in a Factory Layout design are usually placed onto the floor. For the component to orient itself properly relative to the floor, it requires a defined landing surface. The landing surface defines the plane on the component that is positioned relative to the floor. When authoring content, it is important to use the Landing Surface command to establish how the component is located relative to the factory floor.

**Note:** Establishing a landing surface on a component is not required for publishing to the Factory Assets library.

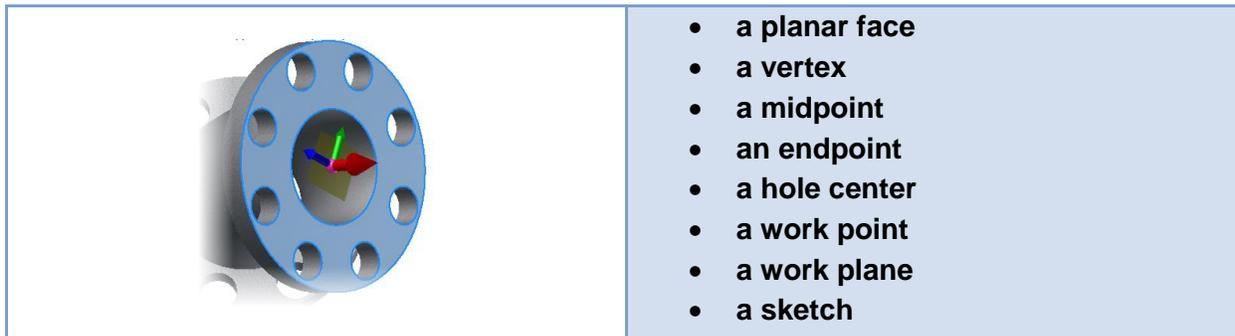
**Insertion Point**

When an asset is placed in a layout, it is initially positioned on the cursor by the first insertion point defined during publishing. You can optionally select one or more insertion points to help you more accurately position the model on the factory floor. An insertion point may be a vertex, work point, or sketch point.

After creating one or more insertion points, the Insert Point Group node appears in the Asset Builder browser. Expand the node and right-click over an insertion point if you wish to edit its location or delete it.

### Define Connectors – Connector Points

Connectors can be used to help assemble components on the factory floor. Connectors allow snapping between specific points on each component and aligning them based on the axes of the connector points. Connector points may be created on the following:

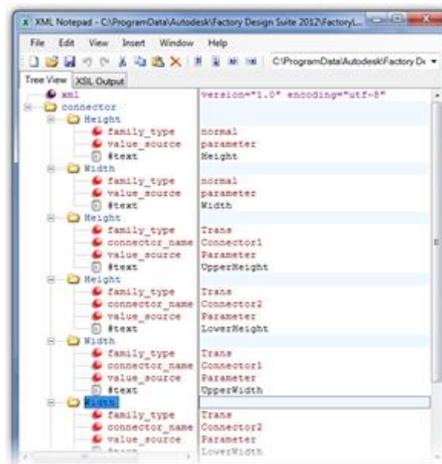
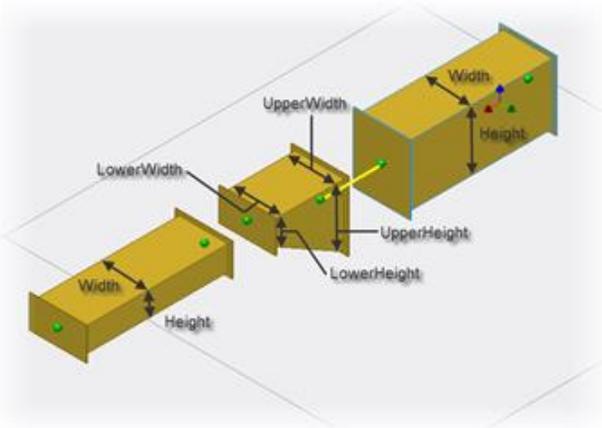


### Connector Class Properties

When creating new assets, it is possible to have them interact, in terms of parameter propagation, with similar assets by assigning connector class properties. These properties control the asset parameters that should be referenced to achieve the desired connection behavior

The Connector Class Properties command lets you choose the connector class to be used for the asset. It also lets you specify the connector class property values that are to be used. These connector class properties are defined in a separate connector class file in .xml format. A set of system connector classes is provided with the product and is used by both the System Assets and Cloud Assets. These connector class files are located on the local hard drive in the C:\ProgramData\Autodesk\Factory Design Suite(version)\FactoryLibrary folder

**Tip:** Use Microsoft XML Notepad to modify or create custom Connector Classes.



### Creating Custom Connector Classes

You can define custom connector classes for new asset types that require connection behavior beyond the product-supplied connector classes. These can then be used to control parameter propagation behavior among assets assigned to the class.

A connector class is defined by an XML file, using the extension .connectorclass. The elements of the connector class define the parameter mapping, and any conditional checks that are used to control the conditions in which the particular mappings are made.

If you are familiar with XML, the structure of a connector class XML file is straightforward. Here is an example of a simple connector class file. The element descriptions follow.

#### Connector Class XML

```
<connector>
<ConveyorWidth value_source="parameter">Width</ConveyorWidth> <ConveyorHeight family_type="normal"
value_source="parameter">Height</ConveyorHeight> <ConveyorHeight family_type="inclined" connector_name="Connector1"
value_source="Parameter">UpperHeight</ConveyorHeight> <ConveyorHeight family_type="inclined" connector_name="Connector2"
value_source="Parameter">LowerHeight</ConveyorHeight> <ConveyorHeight family_type="spiral" connector_name="UpperConnector"
value_source="Parameter">UpperHeight</ConveyorHeight> <ConveyorHeight family_type="spiral" connector_name="LowerConnector"
value_source="Parameter">LowerHeight</ConveyorHeight>
</connector>
```

#### Root Element



Connector class files always use a **Root Element** named connector. This element contains a series of child elements that identify the parameters that are to be propagated between assets when they are connected and belong to the same connector class.

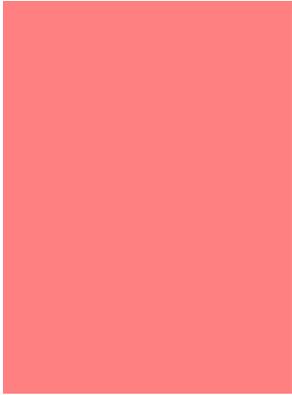
#### Mapping Groups



The child elements define **Mapping Groups**. They can have any name desired and are used according to the following rules:

Each mapping group, which represents a certain parameter, should have a unique element name. In the example above, "**ConveyorWidth**" and "**ConveyorHeight**" represent the two mapping groups.

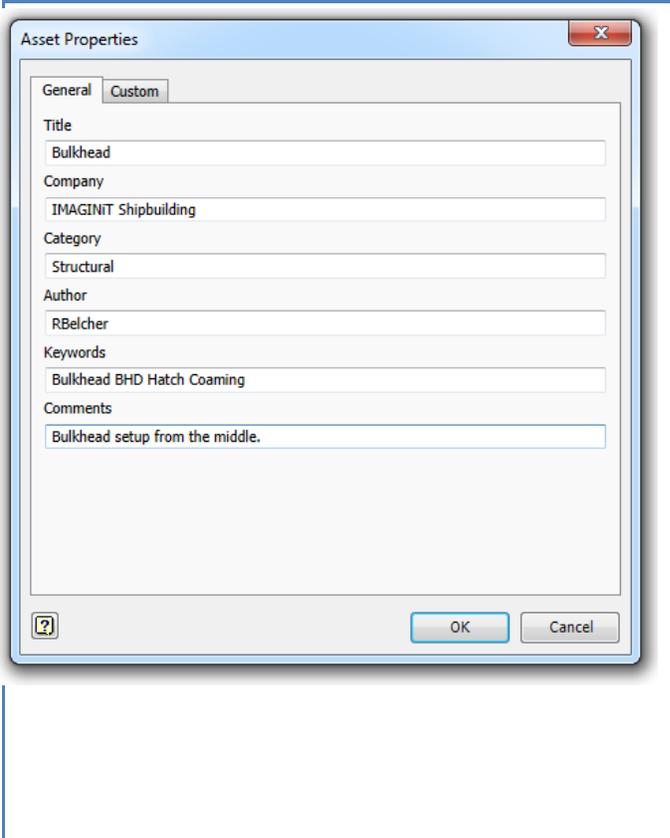
In a given asset, once a match is found for a parameter meeting the matching rules described below, testing of the current mapping group ends, and the next mapping group is processed. Therefore, mapping group elements should be specified in decreasing order of specificity - most specific to least specific

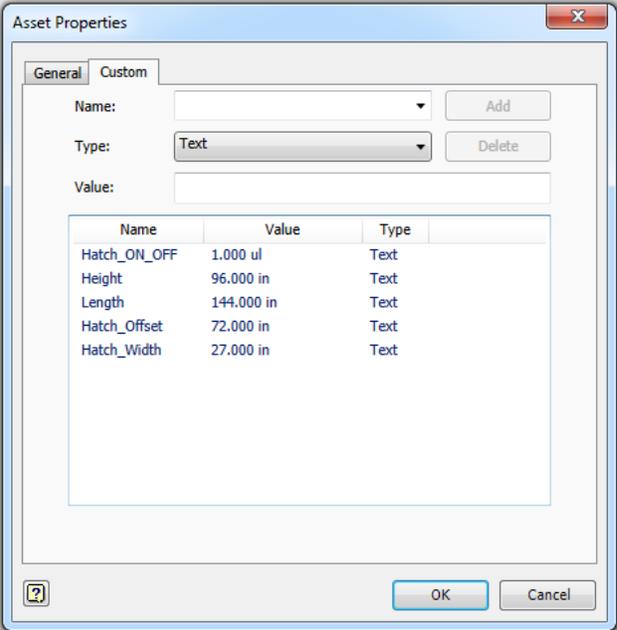
<b>Value Source Attribute</b> 	<p>Noted in the XML code as <b>value_source</b>, this attribute marks the portion of the code that usually maps to a model parameter. The value of these elements usually identifies the parameter name to be mapped (e.g., "Width" above). Both assets involved in the connection must provide a parameter matching this name, or no mapping will be performed.</p>
<b>Value Source Attribute Type</b> 	<p>The Value Source Attribute has two types, Parameter and Constant.</p> <p><b>Parameter</b> - Specifies that the value for the mapping <b>will</b> be provided by the identified parameter in the source asset.</p> <p><b>Constant</b> - Specifies that the value to be assigned will be a constant value, provided in the value attribute</p>
<b>Value Source Parameter Name</b> 	<p>Specifies the name of the model parameter where the numeric value originates.</p> <p><b>Note:</b> Spelling and Capitalization are critical for proper connection.</p>
<b>Family Type Attribute</b> 	<p>Noted in the XML code as <b>family_type</b>, this attribute marks the portion of the code that identifies the subsets of possible connections.</p>
<b>Family Type Name</b> 	<p>Each subset of connections is given a unique name that usually describes the connection situation.</p> <p><b>Examples</b></p> <p><b>Normal</b> – Typical connections where Width equals Width and Height equals Height.</p> <p><b>Inclined</b> – Complex connection representing an incline conveyor where Width could equal UpperWidth or LowerWidth. Height could equal UpperHeight or LowerHeight.</p> <p><b>Spiral</b> – Complex connection representing a spiral conveyor where Width could equal UpperWidth or LowerWidth. Height could equal UpperHeight or LowerHeight.</p>
<b>Connector Name Attribute</b> 	<p>Noted in XML code as <b>connector_name</b>, this attribute marks the portion of the code that identifies the individual connector on the asset model. Some connections require a dedicated connector. This attribute is used to identify the necessary connector by name.</p>
<b>Connector Name in Browser</b> 	<p>Each asset connector has a unique name assigned during the connector creation process. The name appears in the Inventor model browser. These names can be mapped with the connector_name attribute.</p> <p><b>Note:</b> Spelling and Capitalization are critical for proper connection.</p>



**Asset Properties**

All Autodesk Inventor files have properties called *iProperties*. They are used to track and manage files, create reports, and automatically update assembly bills of materials, drawing parts lists, title blocks, and other information. During the Asset Publishing process, you have a final chance to fill out or adjust the *iProperties* of the asset. These properties can be defined using *iProperties* or by using the *Asset Properties* command. The *Asset Properties* command only displays a select set of *iProperty* values.

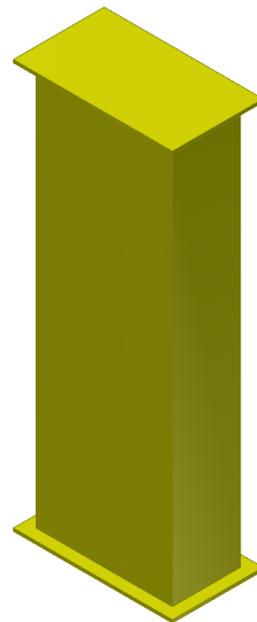
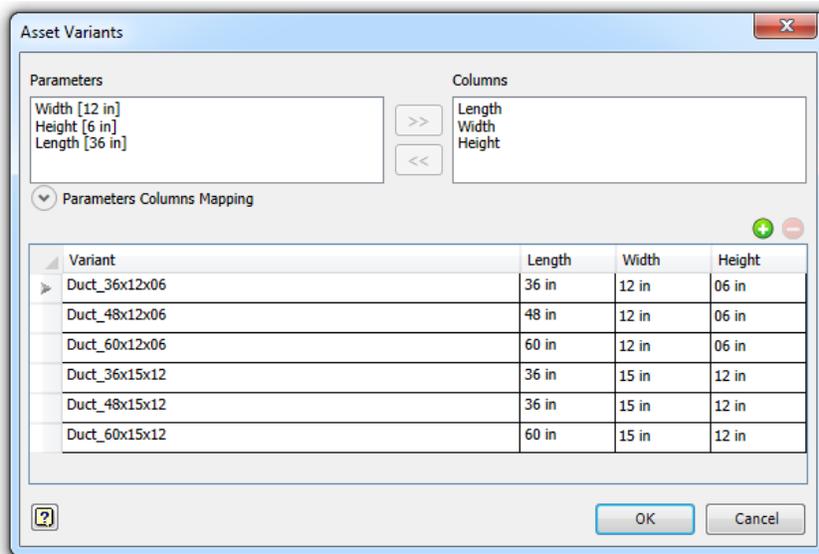
<p><b>Asset Properties General Tab</b></p>	<p>The General tab shows the name, author, and other statistics for the active Inventor asset. Although several of these fields are populated automatically, they can be changed.</p>	
	<p><b>Title</b></p>	<p>Assigns a title for the asset.</p>
	<p><b>Company</b></p>	<p>Identifies the company.</p>
	<p><b>Category</b></p>	<p>Specifies a category for the asset.</p>
	<p><b>Author</b></p>	<p>Identifies the creator of the asset. You can enter a different name.  Note: The author name is set in the General tab of the Autodesk Inventor Application Options dialog box.</p>
	<p><b>Keywords</b></p>	<p>Adds keywords to the asset. You can use keywords to search for related assets.</p>
	<p><b>Comments</b></p>	<p>Adds comments to the asset.</p>

<p><b>Asset Properties Custom Tab</b></p>	<p>The Custom tab adds custom properties to the selected asset. Use custom properties to classify and manage your layout assets, search for assets, create reports, and automatically update parts lists in drawings and bills of materials in assemblies.</p>														
	<table border="1"> <tr> <td data-bbox="881 569 1076 835"><b>Name</b></td> <td data-bbox="1076 569 1430 835">Specifies a name for a new custom property or selects an existing custom property for editing. Enter the name or click the arrow and select from the drop-down list.</td> </tr> <tr> <td data-bbox="881 835 1076 989"><b>Type</b></td> <td data-bbox="1076 835 1430 989">Sets the data type for the property. Click the arrow and select from the drop-down list.</td> </tr> <tr> <td data-bbox="881 989 1076 1142"><b>Value</b></td> <td data-bbox="1076 989 1430 1142">Specifies the value for the property for the asset. The value must conform to the selected data type.</td> </tr> <tr> <td data-bbox="881 1142 1076 1295"><b>Properties list</b></td> <td data-bbox="1076 1142 1430 1295">Displays the Name, Value, and Type of custom properties currently defined for the asset.</td> </tr> <tr> <td data-bbox="881 1295 1076 1409"><b>Add button</b></td> <td data-bbox="1076 1295 1430 1409">Allows you to add your own custom property name.</td> </tr> <tr> <td data-bbox="881 1409 1076 1522"><b>Delete button</b></td> <td data-bbox="1076 1409 1430 1522">Deletes the selected property from the Properties list.</td> </tr> <tr> <td data-bbox="881 1522 1076 1709"><b>Exported Parameters</b></td> <td data-bbox="1076 1522 1430 1709">Parameters marked for export in the parameters dialog box are automatically added as Custom Properties.</td> </tr> </table>	<b>Name</b>	Specifies a name for a new custom property or selects an existing custom property for editing. Enter the name or click the arrow and select from the drop-down list.	<b>Type</b>	Sets the data type for the property. Click the arrow and select from the drop-down list.	<b>Value</b>	Specifies the value for the property for the asset. The value must conform to the selected data type.	<b>Properties list</b>	Displays the Name, Value, and Type of custom properties currently defined for the asset.	<b>Add button</b>	Allows you to add your own custom property name.	<b>Delete button</b>	Deletes the selected property from the Properties list.	<b>Exported Parameters</b>	Parameters marked for export in the parameters dialog box are automatically added as Custom Properties.
<b>Name</b>	Specifies a name for a new custom property or selects an existing custom property for editing. Enter the name or click the arrow and select from the drop-down list.														
<b>Type</b>	Sets the data type for the property. Click the arrow and select from the drop-down list.														
<b>Value</b>	Specifies the value for the property for the asset. The value must conform to the selected data type.														
<b>Properties list</b>	Displays the Name, Value, and Type of custom properties currently defined for the asset.														
<b>Add button</b>	Allows you to add your own custom property name.														
<b>Delete button</b>	Deletes the selected property from the Properties list.														
<b>Exported Parameters</b>	Parameters marked for export in the parameters dialog box are automatically added as Custom Properties.														

### Asset Variants

With all the parametric control you can establish over factory assets, it is inevitable that standard values will need to be set or enforced. These values need to be readily available in the Inventor and AutoCAD environments. Asset Variants can be set during the asset publishing process. Similar to the old iPart process, this command allows you to define a pre-defined set of values for a given parameter. These values can be chosen prior to the asset placement or change the standard size of an existing asset.

User assets published with the Publish 2D Asset option appear in the AutoCAD Factory Asset Browser ready for use in 2D factory floor layouts. However, these 2D assets are difficult to modify if variations are required. A very powerful solution is to create and publish the asset with the necessary dimensional variations that can easily be selected and placed on the AutoCAD factory floor.

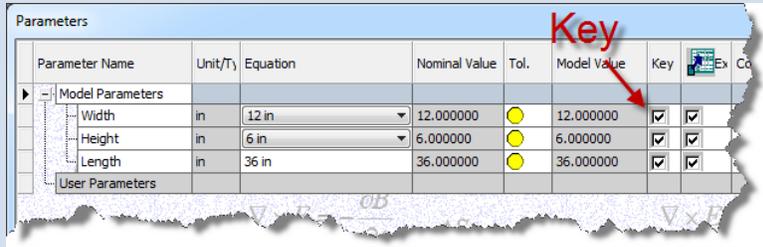


Asset variants are distinguished from other assets in the Assets Browser with this icon  that appears at the upper left of the asset. The numeral in the icon represents the number of variants defined for that asset.

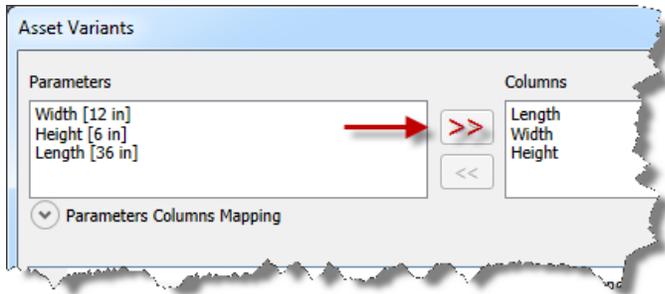
#### Note:

The asset variant workflow can produce many preset standard variations of the original model. Veteran Inventor users will recognize this functionality as being similar to the iPart design process. The variant process allows you to establish a unique name for each variation and specify the required values for the design parameters.

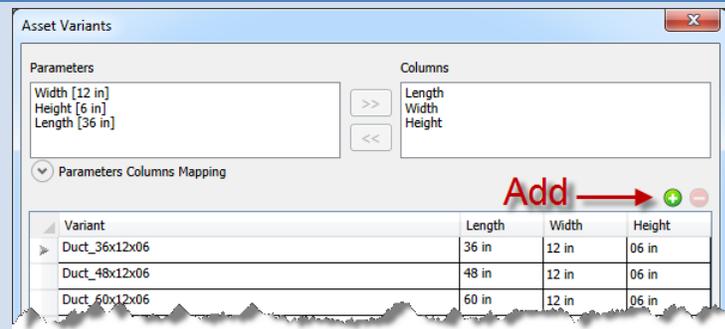
## Asset Variants



Variants can be created only if User parameters are created and if they are identified as “**Key**” in the Parameters dialog. You can rename existing parameters or create new parameters by clicking the Add Numeric button in the Parameters dialog.



Once parameters are defined and marked as key, they will appear in the Asset Variants dialog box. Use the >> button to add the desired parameters as columns in the variant table.



Use the **Add** button to create a new variant row in the table. Each row allows you to establish a unique name and modify the values of the design parameters.

You can add as many rows as you wish.



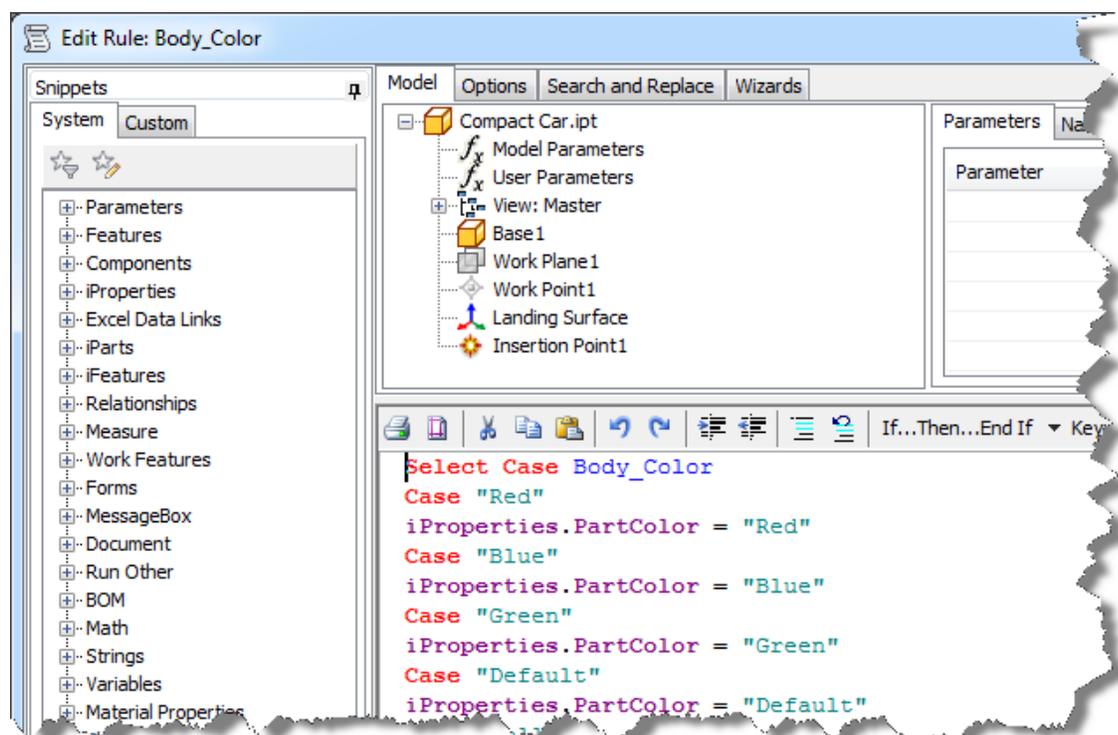
All variants of the asset will be available after the asset is published.

Assets with Variants will appear in the Asset Browser with a circular number in the upper corner of the thumbnail image. The number in the circle indicates the number of variants available for that asset.

### Using iLogic with Assets

The parametric functionality that can be established for an asset is considerable, but there are situations that require a more robust and complex set of controls. Some modeling features require a logical argument, or an “if this, then this” manner of control. This functionality is available through Inventor’s iLogic environment. The iLogic environment allows designers to include coded functionality into an Inventor model or asset.

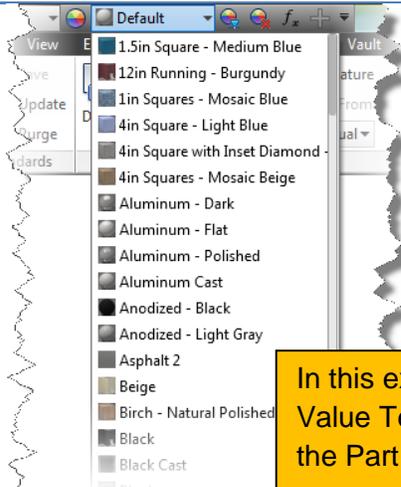
This section will demonstrate a few simple iLogic codes that deliver functionality often required in asset design. The overall objective of this section is not to deliver in depth instruction on the use of iLogic. This section will simply provide examples of how iLogic can be used to deliver model functionality that is often required in asset design.



## Rules for Part Color

It is often necessary to control part or feature color with a parameter. This can be accomplished with the following iLogic process.

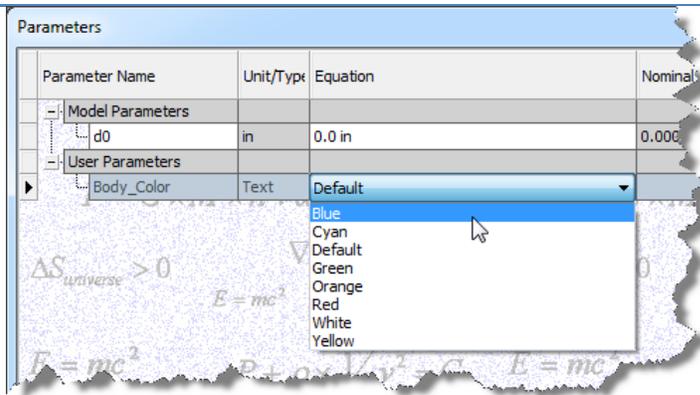
1. Make sure the desired colors are built into the model file.



In this example, a single Multi-Value Text parameter will control the Part Color of the component.

2. Create a Multi-Value Text parameter with the color choices.

**Note:** The spelling and capitalization of the color names must be exact.

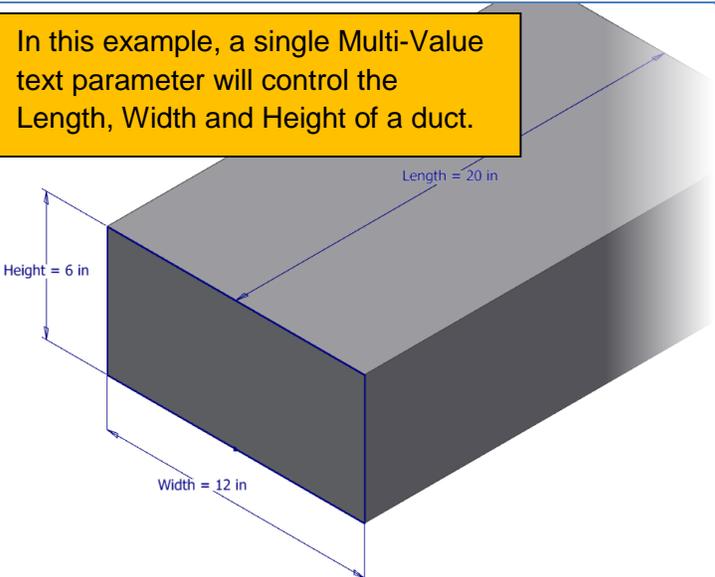
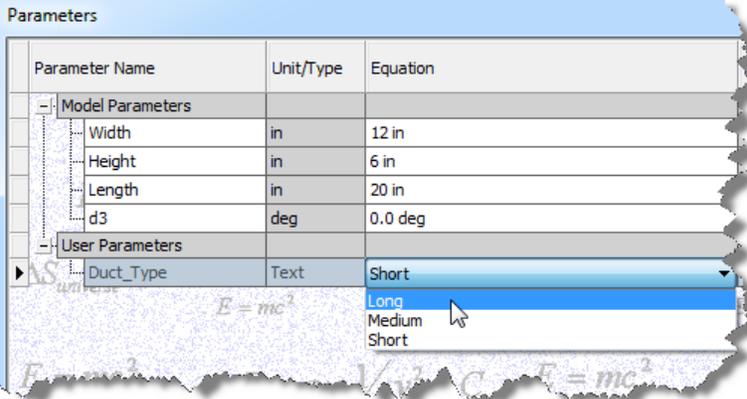


3. Create an iLogic rule that controls the part color using the format in the image on the right.

**Note:** This rule uses the Select Case format, which is useful for controlling a single feature with a Multi-Value Text Parameter.

```

Select Case Body_Color
Case "Red"
iProperties.PartColor = "Red"
Case "Blue"
iProperties.PartColor = "Blue"
Case "Green"
iProperties.PartColor = "Green"
Case "Default"
iProperties.PartColor = "Default"
Case "Yellow"
iProperties.PartColor = "Yellow"
Case "Cyan"
iProperties.PartColor = "Cyan"
Case "White"
iProperties.PartColor = "White"
Case "Orange"
iProperties.PartColor = "Orange"
End Select
    
```

<h2>Rules for Multiple Parameters</h2>	<p>In this example, a single Multi-Value text parameter will control the Length, Width and Height of a duct.</p> 																																	
<p>It is often necessary to control multiple features with a single parameter. This can be accomplished with the following iLogic process.</p> <ol style="list-style-type: none"> <li>1. Make sure the required model parameters are functioning as desired.</li> </ol>																																		
<ol style="list-style-type: none"> <li>2. Create a Multi-Value Text parameter with the desired configuration names.                     <p><b>Note:</b> The spelling and capitalization of the configuration names must be consistent with the iLogic code.</p> </li> </ol>	 <table border="1"> <thead> <tr> <th>Parameter Name</th> <th>Unit/Type</th> <th>Equation</th> </tr> </thead> <tbody> <tr> <td colspan="3">Model Parameters</td> </tr> <tr> <td>Width</td> <td>in</td> <td>12 in</td> </tr> <tr> <td>Height</td> <td>in</td> <td>6 in</td> </tr> <tr> <td>Length</td> <td>in</td> <td>20 in</td> </tr> <tr> <td>d3</td> <td>deg</td> <td>0.0 deg</td> </tr> <tr> <td colspan="3">User Parameters</td> </tr> <tr> <td>Duct_Type</td> <td>Text</td> <td>Short</td> </tr> <tr> <td></td> <td></td> <td>Long</td> </tr> <tr> <td></td> <td></td> <td>Medium</td> </tr> <tr> <td></td> <td></td> <td>Short</td> </tr> </tbody> </table>	Parameter Name	Unit/Type	Equation	Model Parameters			Width	in	12 in	Height	in	6 in	Length	in	20 in	d3	deg	0.0 deg	User Parameters			Duct_Type	Text	Short			Long			Medium			Short
Parameter Name	Unit/Type	Equation																																
Model Parameters																																		
Width	in	12 in																																
Height	in	6 in																																
Length	in	20 in																																
d3	deg	0.0 deg																																
User Parameters																																		
Duct_Type	Text	Short																																
		Long																																
		Medium																																
		Short																																
<ol style="list-style-type: none"> <li>3. Create an iLogic rule that controls multiple values from a single parameter using the format in the image on the right.                     <p><b>Note:</b> This rule uses the Select Case format, which is useful for controlling a single feature with a Multi-Value Text Parameter.</p> </li> </ol>	<pre> Select Case Duct_Type Case "Short" Parameter("Length") = 20 Parameter("Width") = 12 Parameter("Height") = 6 Case "Medium" Parameter("Length") = 40 Parameter("Width") = 18 Parameter("Height") = 10 Case "Long" Parameter("Length") = 60 Parameter("Width") = 20 Parameter("Height") = 15 End Select                     </pre>																																	

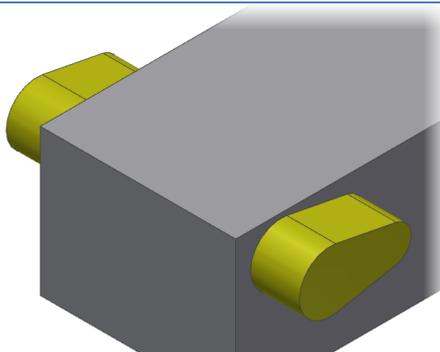
## Rules for Feature Suppression

It is often necessary to control feature suppression based on the value of other parameters. This can be accomplished with the following iLogic process.

1. Make sure the required model parameters and features are functioning as desired.

**Note:**

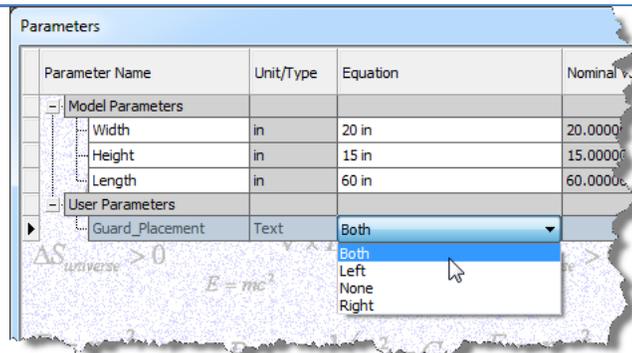
It may be necessary to model one feature directly on another.



In this example, a Multi-Value text parameter will control the placement of the chain guard.

2. Create a Multi-Value Text parameter with the desired configuration names.

**Note:** The spelling and capitalization of the configuration names must be consistent with the iLogic code.



3. Create an iLogic rule that controls feature suppression from a single parameter using the iLogic format in the image on the right.

**Note:**

This rule uses the If Then / Else If format, which is useful for controlling a single feature with a Multi-Value Text Parameter.

```

If Guard_Placement = "Both" Then
Feature.IsActive("Guard_Right") = True
Feature.IsActive("Guard_Left") = True

ElseIf Guard_Placement = "None" Then
Feature.IsActive("Guard_Right") = False
Feature.IsActive("Guard_Left") = False

ElseIf Guard_Placement = "Left" Then
Feature.IsActive("Guard_Right") = False
Feature.IsActive("Guard_Left") = True

ElseIf Guard_Placement = "Right" Then
Feature.IsActive("Guard_Right") = True
Feature.IsActive("Guard_Left") = False

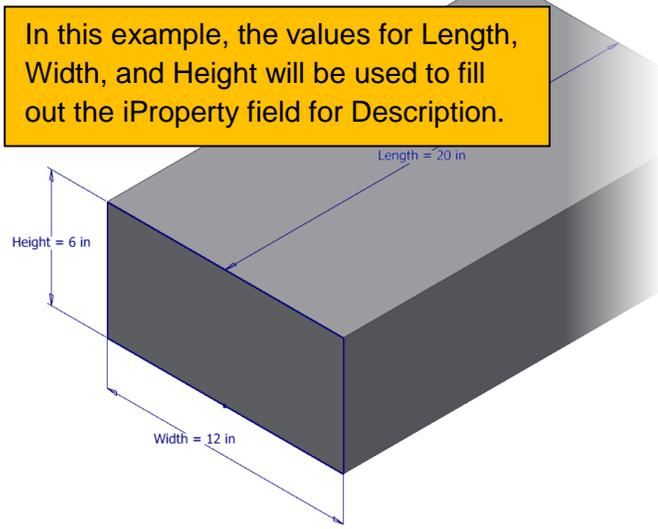
End If

```

## Rules for File Properties

It is often necessary to control the file properties based on the value of parameters. This can be accomplished with the following iLogic process.

1. Make sure the required model parameters and features are functioning as desired.



2. Review the Parameters Dialog for correct names.

**Note:** The spelling and capitalization of the configuration names must be consistent with the iLogic code.

Parameters

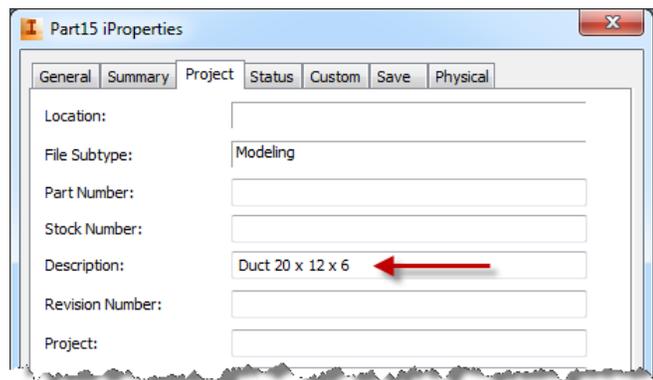
Parameter Name	Unit/Type	Equation	Nominal Value	Dr
<b>Model Parameters</b>				
Width	in	12 in	12.000000	
Height	in	6 in	6.000000	
Length	in	20 in	20.000000	
<b>User Parameters</b>				

3. Create an iLogic rule that fills out the Description iProperty field.

**Note:** The iProperty snippet for Description is all that is needed. Plain Text is placed between quotes and the ampersand is used to add each addition element in the string.

Confirm the results by opening the iProperties and reviewing the Description property.

**Note:** This rule can be combined with the Factory Descriptor rule allowing the Descriptor and the Description to be set by the same rule.



```
iProperties.Value("Project", "Description") = "Duct " & Length & " x " & Width & " x " & Height
```

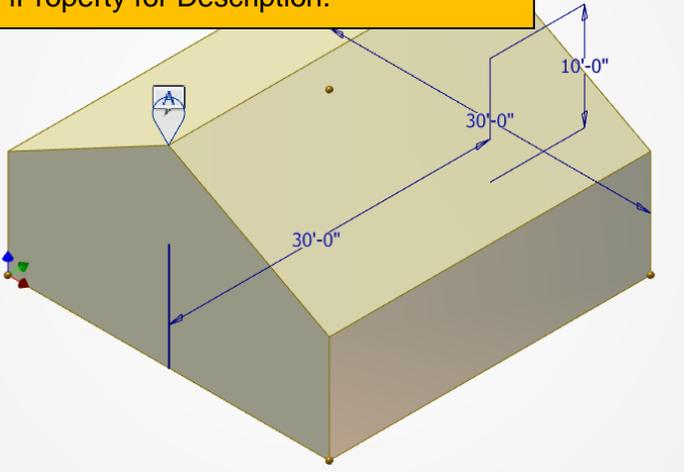
## Rules for Descriptor and Description

Inventor 2014 introduces a new asset feature called a Descriptor. The Descriptor allows custom identification information to be added to the asset. This information can be displayed in Inventor or AutoCAD.

The Descriptor has an advanced option to format the information displayed with an iLogic Rule.

1. Make sure the required model parameters and features are functioning as desired.

In this example, values for Length, Width, and Height will be used to fill out the Factory Descriptor and the iProperty for Description.



2. Review the Parameters Dialog for correct names.

**Note:** The spelling and capitalization of the configuration names must be consistent with the iLogic code.

Parameter Name	Unit/Ty	Equation	Nominal Vali	Driving Rule
Model Parameters				
Length	in	30'-0"	360.000...	
Height	in	10'-0"	120.000...	
Width	in	30'-0"	360.000...	
d3	deg	0.0 deg	0.000000	
	in	8'-0"	96.000000	

3. During Asset Publishing, Click the Descriptor command and select the Advanced button.

Create an iLogic rule that fills out the Descriptor and the Description iProperty field.

**Note:** The basic snippet for the Factory Descriptor is provided automatically.

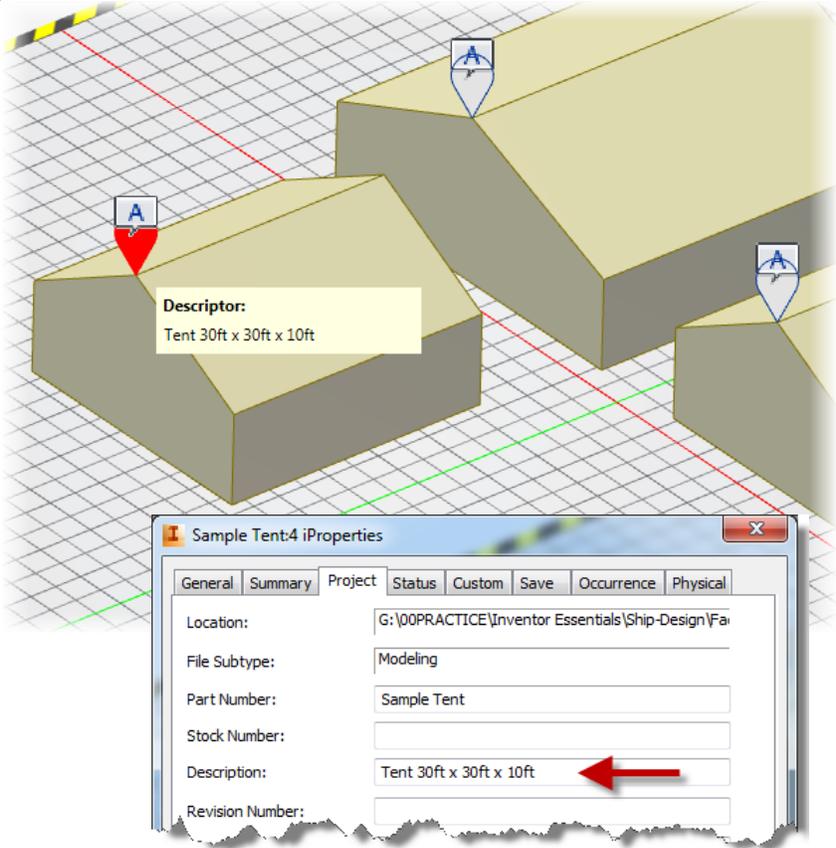
**Note:** In this example, simple math operations establish the desired units.

**Optional:** The code for the Description iProperty can be added below the Factory Descriptor function.

```
LengthFeet = Length / 12
WidthFeet = Width / 12
HeightFeet = Height / 12
```

```
Factory_Descriptor = "Tent " & LengthFeet & "ft" & " x " & WidthFeet & "ft" & " x " & HeightFeet & "ft"
iProperties.Value("Project", "Description")= "Tent " & LengthFeet & "ft" & " x " & WidthFeet & "ft" & " x " & HeightFeet & "ft"
```

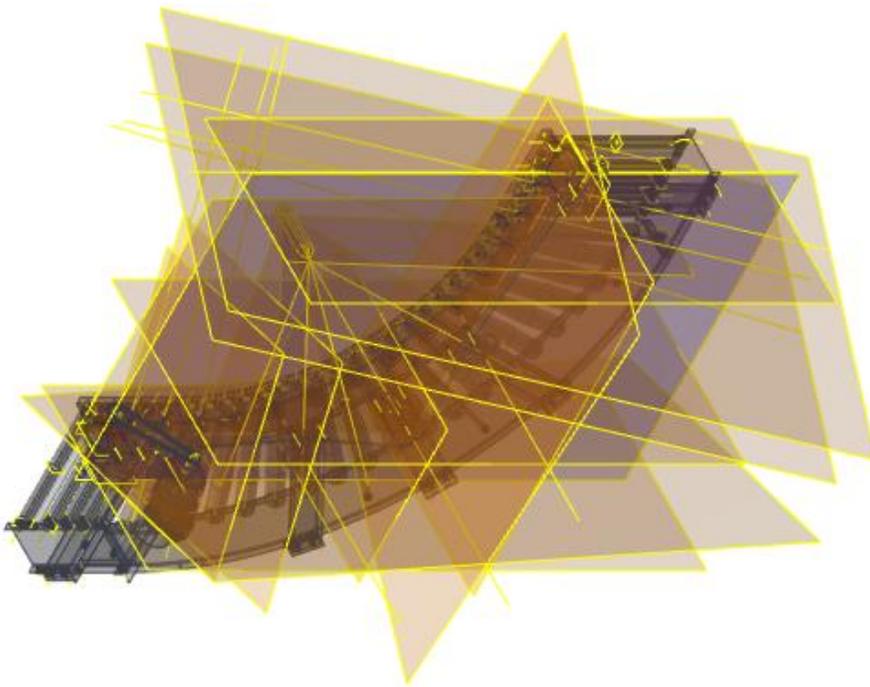
4. Confirm the rule by hovering over the descriptor and reviewing the Description iProperty.



***All Work Features OFF***

The nature of Factory Assets does not allow designers to make changes to the assets once they are published. Therefore, it is important to make sure all parametric and aesthetic settings are precise, functional, and acceptable before publishing. It is quite annoying to use a part created by another designer and find that they failed to turn off all the work features. This issue is compounded by the fact that Work Features are often required to establish landing surfaces, insertion points, or connectors.

It is recommended that the last thing you do, before the final publish command, is to turn off all Work Features.



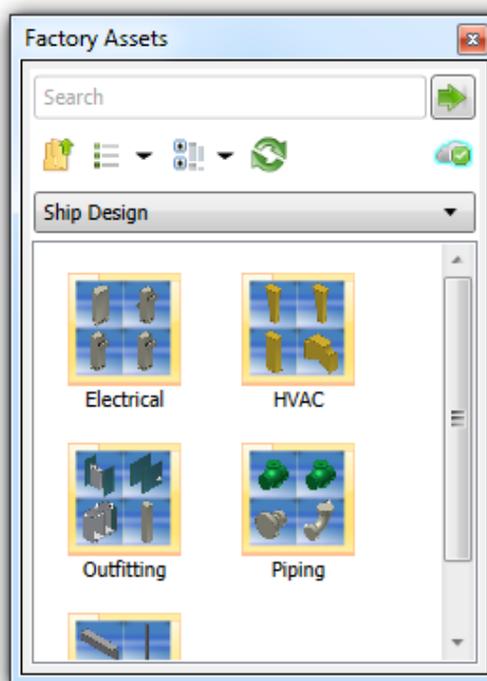
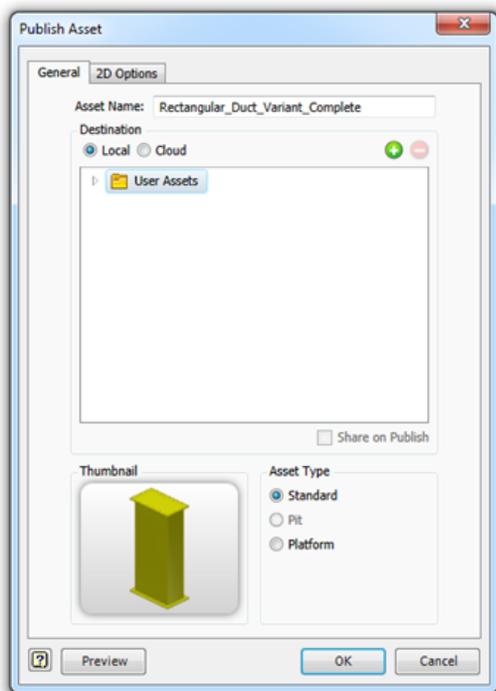
## Publishing Assets – Initial Publish

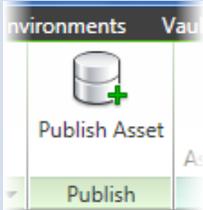
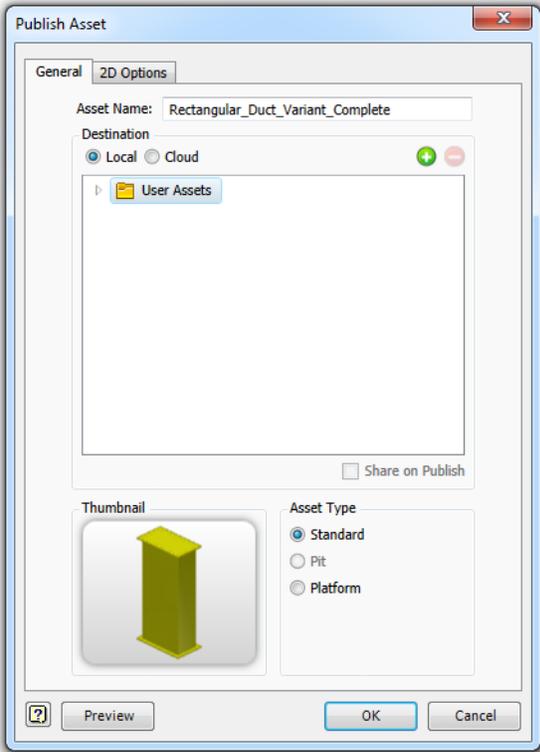
Publishing the asset adds the model to the User Assets Folder of the Factory Asset library. Once an asset is published, it can be inserted from the Asset Browser into the factory layout.

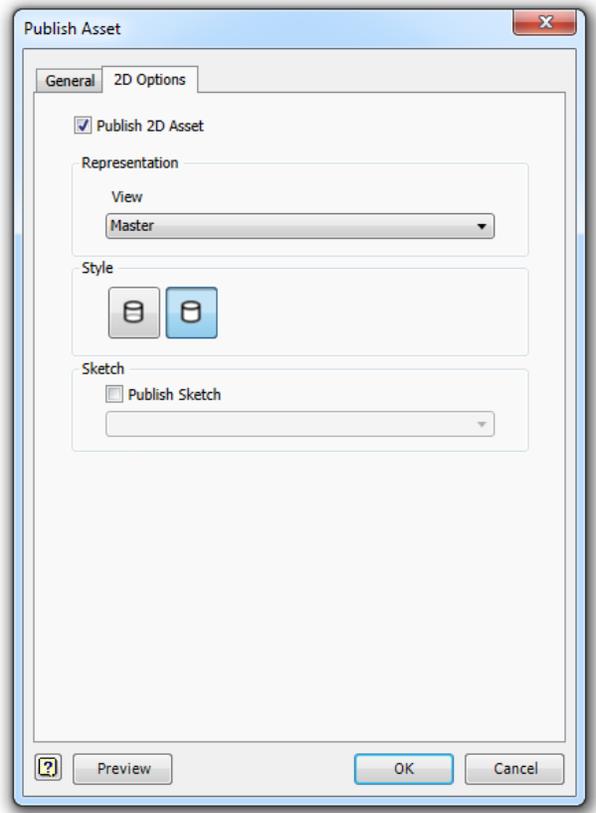
### Note:

Before authoring and publishing a component as a User Asset, ensure that the model is fully designed and modeled with the desired parametric functionality. All parameters that control different model sizes or configurations must be named and tested before authoring. The design parameters should also be marked as Key prior to publishing. If necessary, all property information, iLogic rules, and variant information must be filled out prior to publishing.

Before publishing an asset, refer to the **Asset Publishing Checklist** at the end of this section.



<p><b>Asset Publishing</b></p>	
	<p>The <b>Publish Asset</b> tool is located on the Asset Builder ribbon. This command launches the Publish Asset dialog.</p> <p><b>Note:</b> A model or assembly cannot be published if it has a pending save.</p>
	<p><b>General Tab</b></p> <p>The General Tab allows users to set the asset file name and determine the storage location for the asset. (By default, authored assets are published to the <i>User Assets</i> library.)</p> <p>If you want to specify a different folder for your published asset, right-click the <i>User Assets</i> folder and select <i>New</i> from the pop-up context menu. The new folder appears as a subfolder in the <i>User Assets</i> folder.</p>



**2D Options Tab**

The 2D Options tab allows you to control the 2D version of the published asset. This version will be used in the AutoCAD environment to develop a classic top down view of the layout or compartment. Options Include.

**Publish 2D Asset**  
Indicates whether a 2D version should be created or not.

**Representation**  
Determines which View Representation will be used to create the 2D asset.

**Style**  
Hidden Lines or No Hidden Lines.

**Sketch**  
Provides the option to generate an independent sketch of the desired top view.

## Publish – Local Testing

It is important to test the asset thoroughly before releasing it into the production environment. Publish the asset to your local workstation and put it through its paces, testing all functionality. A checklist of common testing criteria is provided below.



**Testing Checklist**

Checklist for Asset Testing	
<input type="checkbox"/>	Parameter Functions – Model Update
<input type="checkbox"/>	iProperty Settings – In Way of Mapped Values.
<input type="checkbox"/>	Connector Functionality
<input type="checkbox"/>	Connector Class Parameter Propagation.
<input type="checkbox"/>	Parameters - Key
<input type="checkbox"/>	Asset
<input type="checkbox"/>	iLogic
<input type="checkbox"/>	Variants FDS General Layout Functionality - Inventor
<input type="checkbox"/>	FDS General Layout Functionality - AutoCAD
<input type="checkbox"/>	Inventor Sync - AutoCAD Sync

**Modify Asset to Suit Testing Results**

Based on your testing results you may need to make changes or updates to the Factory Asset before final publishing.

Tip: Keep the asset file open during the testing phase. If you find an issue, close all testing files except the actual asset. Make the modifications and republish. Keeping the asset file open saves you time and the extra steps of opening the asset and accessing the necessary dialogs. Also the publish dialog will “remember” your settings as long as the asset file remains open.

**Publish to Cloud or Production Environment**

Once you are completely satisfied with the performance of the asset, you can publish it to your production environment or to the Autodesk Cloud. Assets published to the cloud are available to the entire Factory Design Suite community. When any user of the Factory Design Suite executes a search in the Asset Browser, your cloud-based assets will be included in the results.

**Delete Local Asset**

Once the asset is in service, you can delete the local copy you created for testing. Your factory layouts should only use production ready assets. Mixing Production and Testing assets together in a factory layout can cause a variety of issues in downstream data management processes.

**Asset Publishing Checklist**

The Asset Publishing Checklist is provided on the following page as a final reference. Please feel free to adjust it to your specific needs or standards.



Checklist for Asset Publishing	
<input type="checkbox"/>	Model Detail – Low
<input type="checkbox"/>	Shrinkwrap if Necessary
<input type="checkbox"/>	Multi-Body if Necessary
<input type="checkbox"/>	Parameters - Named
<input type="checkbox"/>	Parameters - Key
<input type="checkbox"/>	Parameters - Exported
<input type="checkbox"/>	Parameters – Multi-Value
<input type="checkbox"/>	Parameters - Tested
<input type="checkbox"/>	iProperties – Part Number
<input type="checkbox"/>	iProperties – Description
<input type="checkbox"/>	IProperties – Mapped Parameters
<input type="checkbox"/>	Work Features for Publishing - On
<input type="checkbox"/>	Landing Surface
<input type="checkbox"/>	Insertion Point
<input type="checkbox"/>	Define Connectors
<input type="checkbox"/>	Connector Class Properties
<input type="checkbox"/>	Asset Properties
<input type="checkbox"/>	Asset Variants
<input type="checkbox"/>	Asset Descriptor
<input type="checkbox"/>	iLogic
<input type="checkbox"/>	All Work Features Off
<input type="checkbox"/>	Publish Local
<input type="checkbox"/>	Test
<input type="checkbox"/>	Modify to Suit Testing
<input type="checkbox"/>	Publish Local - Final
<input type="checkbox"/>	Publish to Cloud - Optional



**Summary**

The basic process of developing assets for Factory Design Suite is fairly simple and straight forward, but there are many options for asset functionality to consider. The goal of this course is to provide a general step-by-step checklist approach for the asset development process. During this course, we reviewed each step in the checklist process and explained the possible asset enhancements available. The next time you're developing a factory asset, I hope you remember to use this checklist, or better yet, develop a checklist of your own.

## **Autodesk Factory Design Suite: Asset Development Checklist**

Rusty Belcher – IMAGINiT Technologies

**PM 10277**

