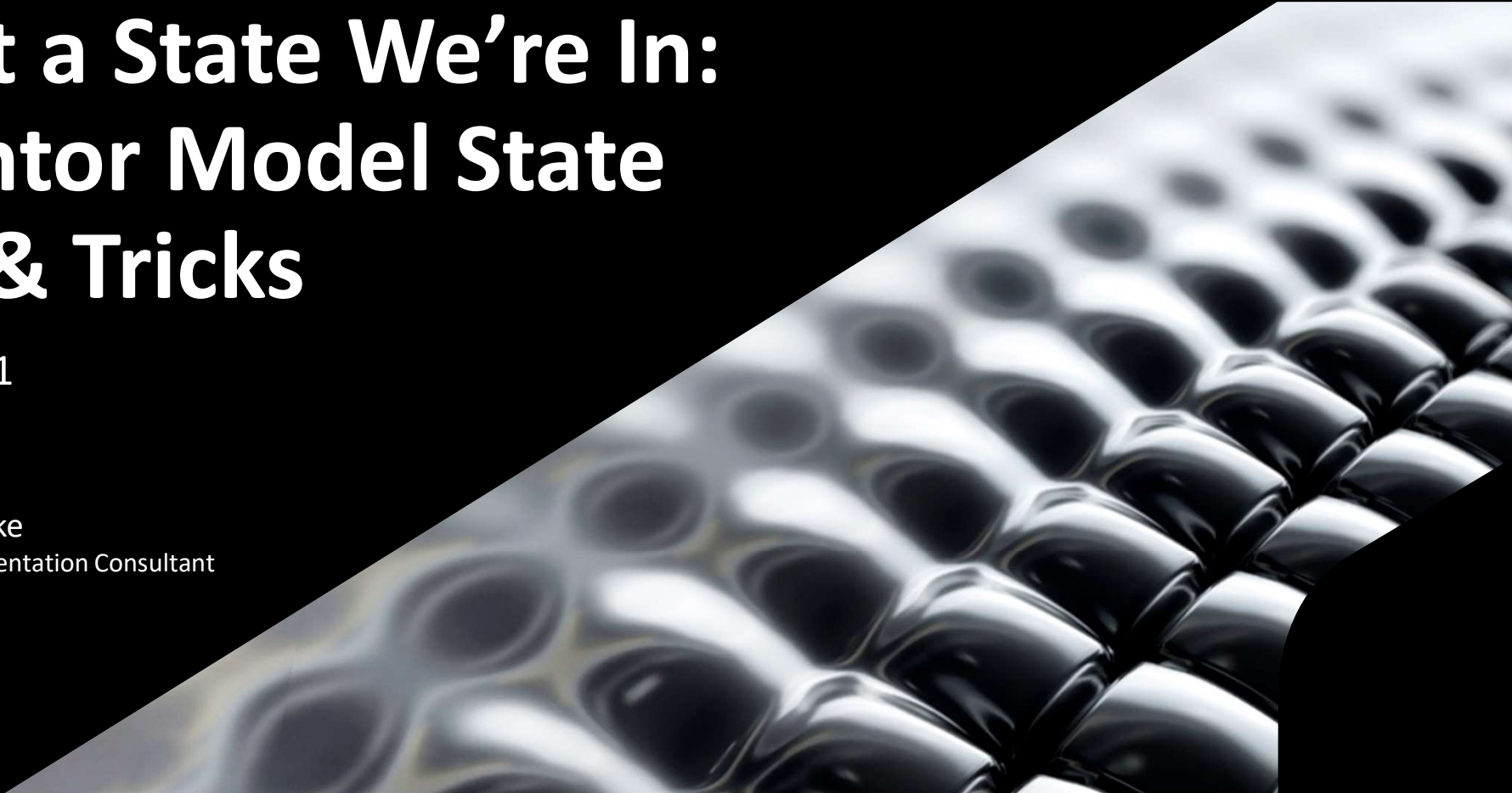




# What a State We're In: Inventor Model State Tips & Tricks

MFG501041

Pete Strycharske  
TeamD3 – Implementation Consultant  
[@petestrycharske](#)





# Who am I???

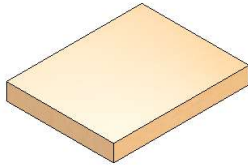
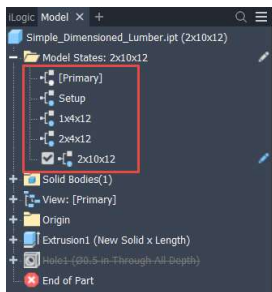
**Pete Strycharske : TeamD3  
Implementation Consultant**

- Autodesk Platinum Partner
- Teach classes on Inventor, AutoCAD, FDU & Fusion 360
- Provide technical support on Autodesk products
- Consult on design workflows and customer content generation
- YouTube content (My wife jokingly calls me a “CAD Influencer”)
- Love God, working with kids, serving in church, Star Wars, the beach and playing basketball!

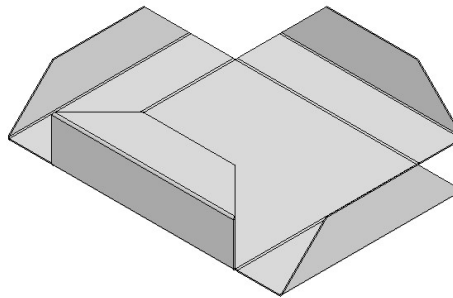
# What's Our Goal???

# Topics Covered in this Class

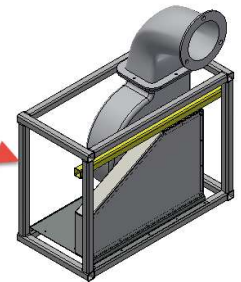
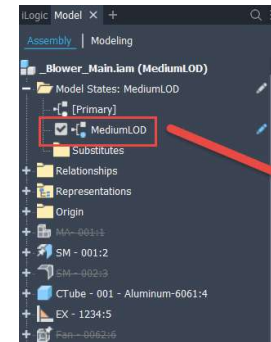
## Model States Tips Including...



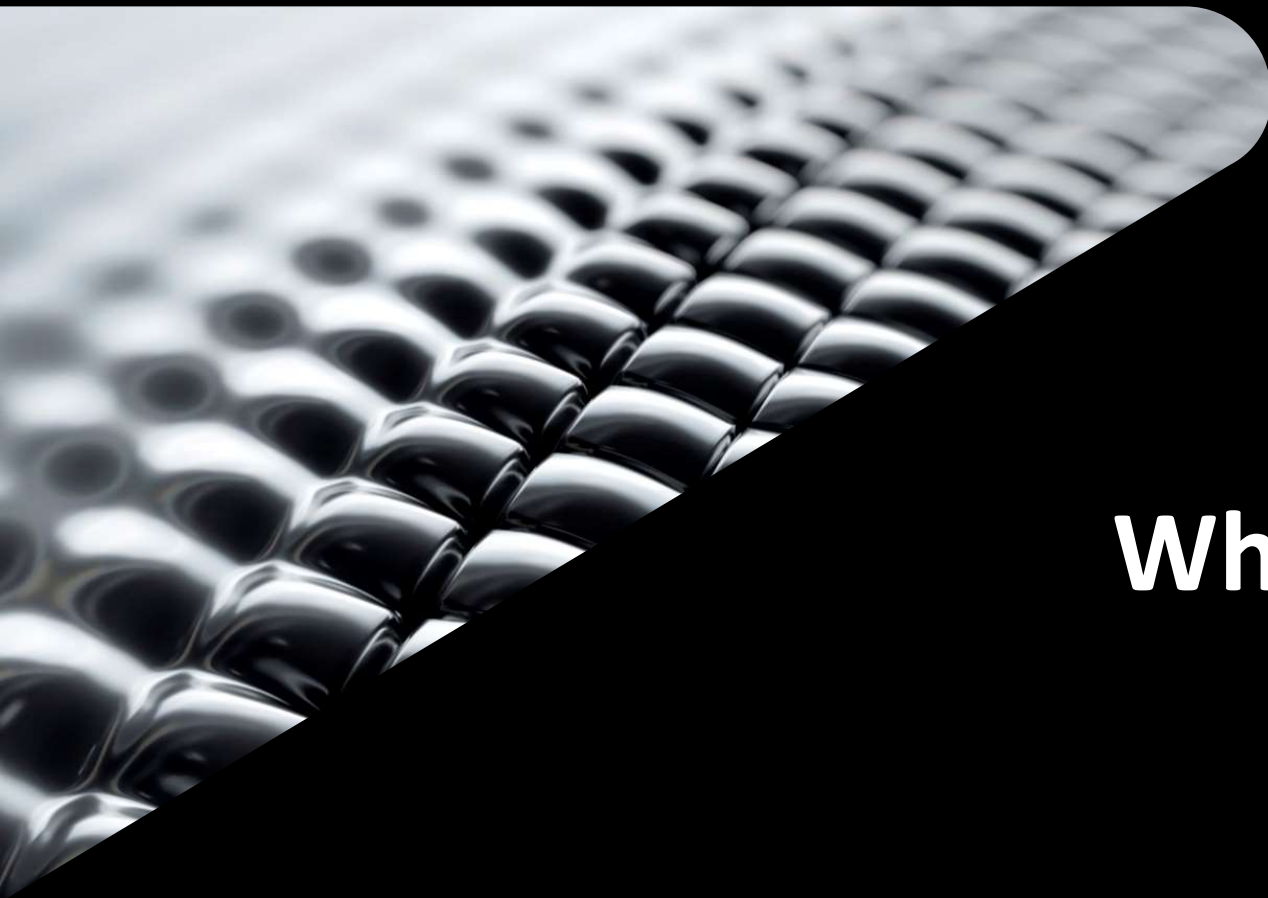
Organization  
(Setting up for Success &  
Reordering Data)



Fabrication Verification  
(Organizing and Animating  
Model States)



Advanced Topics  
(Using Derived Designs vs.  
Linking Model States,  
Working with Converted  
LODs & Tracking Model State  
Versions in Vault)



# What's in a Name?

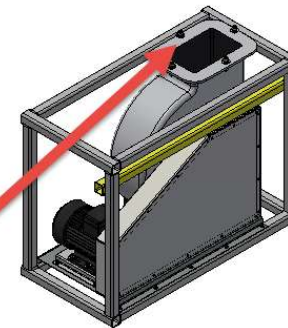
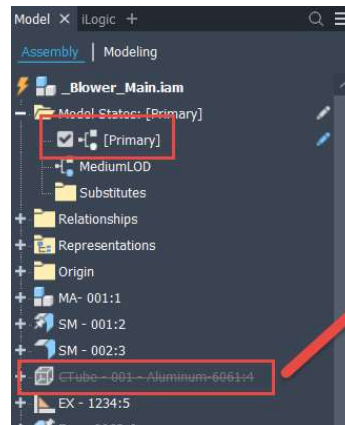
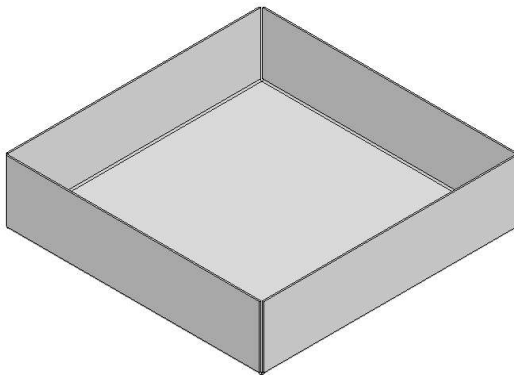
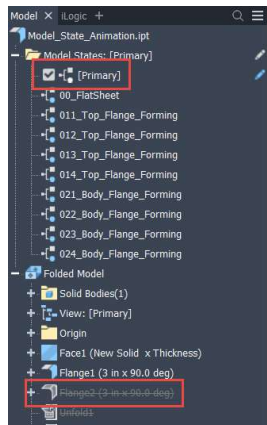
Naming Considerations

# Caution When Using the Primary / Master Model State

**Note:** the “Primary” Model State replaces the “Master” starting in 2023

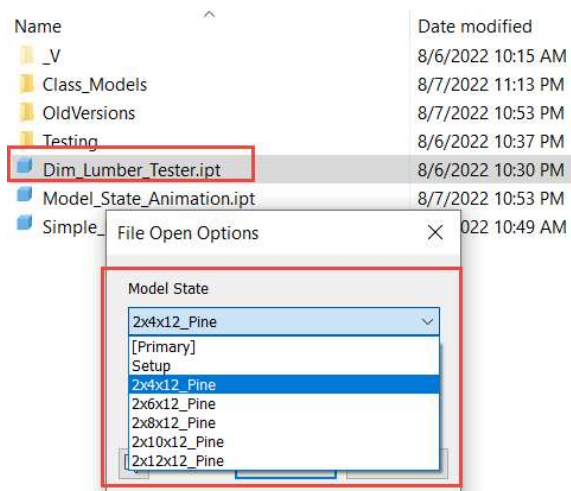
- The Primary State can be deceptive, because features (parts) and components (assemblies) can be suppressed
  - This doesn't mean that all features and components are active
- Particularly challenging when using converted LODs
  - Converted LODs will reference the Primary MS
  - BOMs can be incorrect if Primary contains suppressed components

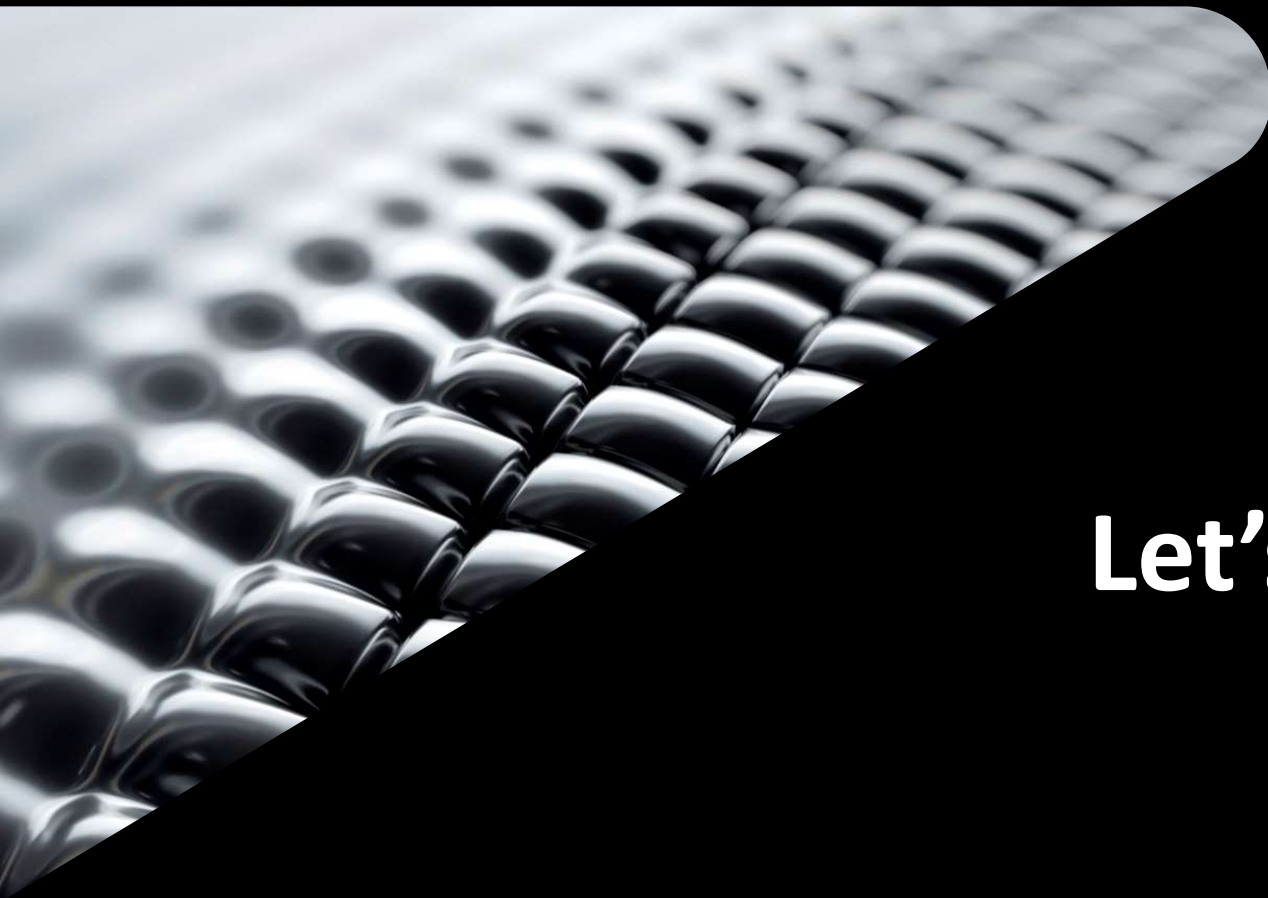
Use “**Final\_Design**” to denote a single completed model version.



# Consistent Naming is Key

- For multi-step fabrication components to ensure each step is clearly understood
- For multi-version components to ensure that each version is clearly identified
  - Helps to stay organized for parameter, iProperty and feature manipulation
  - Assist users to place the proper version in an assembly





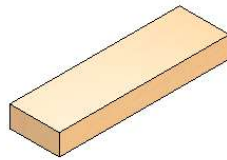
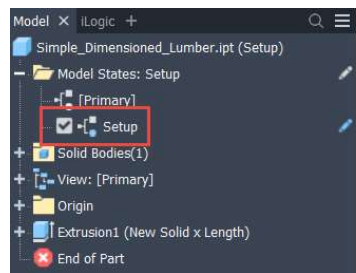
# Let's Get Organized

Initial Steps



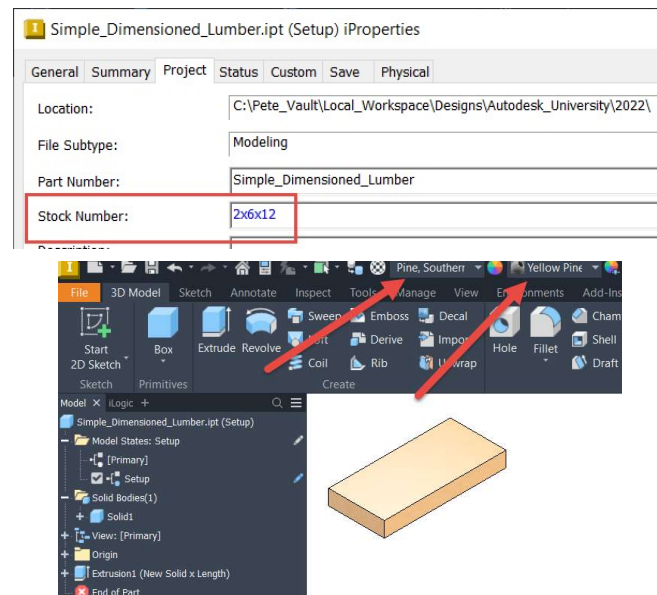
# Create a Setup Model State

Helps think through a design and makes Excel editing MUCH easier

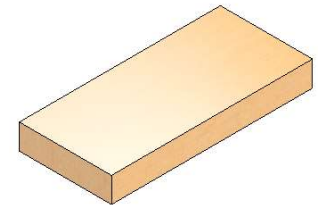
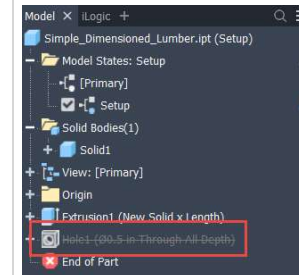


Parameters				
	Parameter Name	Consumed by	Unit/Type	Equation
	Model Parameters			
	User Parameters			
	Thickness	d1	in	1.5 in
	Width	d0	in	5.5 in
	Length	d2	in	12 in

Edit all parameters that change throughout the design




Edit all iProperties and alter materials and appearances
















Suppress / Unsuppress all features that change throughout the design

# Makes for Happier Excel Work!






AutoSave ☐ Off  Model State Worksheet for Simple\_Dimensioned\_Lumber (jf5qure5rx2pw3kv5sxto4pfp)lff.xlsx

File Home Insert Page Layout Formulas Data Review View Help Autodesk Vault



Undo   Paste    





Calibri 11       

Font

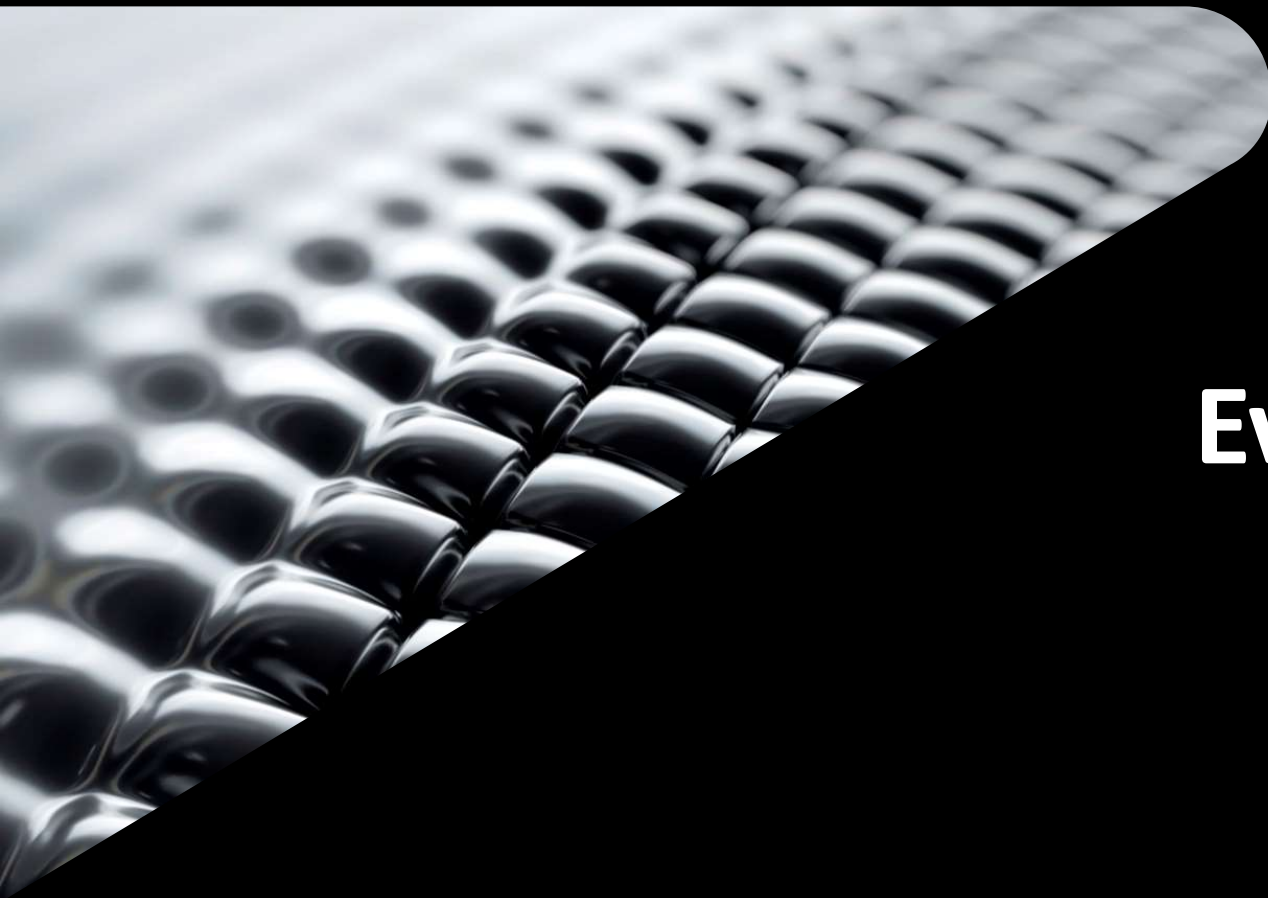
     Wrap Text

Alignment

 Merge & Center 

A1     Member

	A	B	C	D	E	F	G
1	Member	Material [Physical]<material></material>	Width	Stock Number [Project]	d4	d5	Hole1
2	[Primary]	Generic	3.5 in		Width / 2 ul	1.5 in	Compute
3	Setup	Pine, Southern	5.5 in	2x6x12	1.741 in	1.353 in	Suppress
4							



# Everyone Loves a Good Remodel

Reorganizing Data

# Reorder Data to Maximize Efficiency

Makes data modification MUCH faster in Excel

AutoSave Off Model State Worksheet for Simple\_Dimensioned\_Lumber (tnidgvw5upc1pnuhzhcndsjo)ff.xlsx

File Home Insert Page Layout Formulas Data Review View Help Autodesk Vault

Undo Clipboard Font Alignment Number Styles

Calibri 11

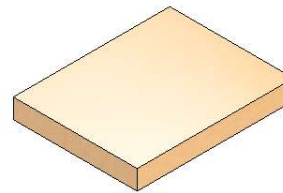
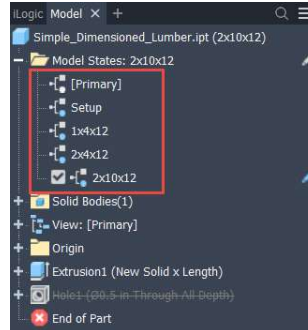
General Conditional Formatting

Format as Table Cell Styles

E7

	A	B	C	D	E	F	G
1	Member	Thickness	Width	Length	Stock Number [Project]	Material [Physical]<material></material>	Hole1
2	[Primary]	1.5 in	3.5 in	12 in		Generic	Compute
3	Setup	3.5 in	5.5 in	18 in	2x6x12	Pine, Southern	Suppress
4	2x4x12	1.5 in	3.5 in	12 in	2x4x12	Pine, Southern	Suppress

Group common data columns together in Excel



Rearrange Rows by left-clicking and dragging in the Model Browser

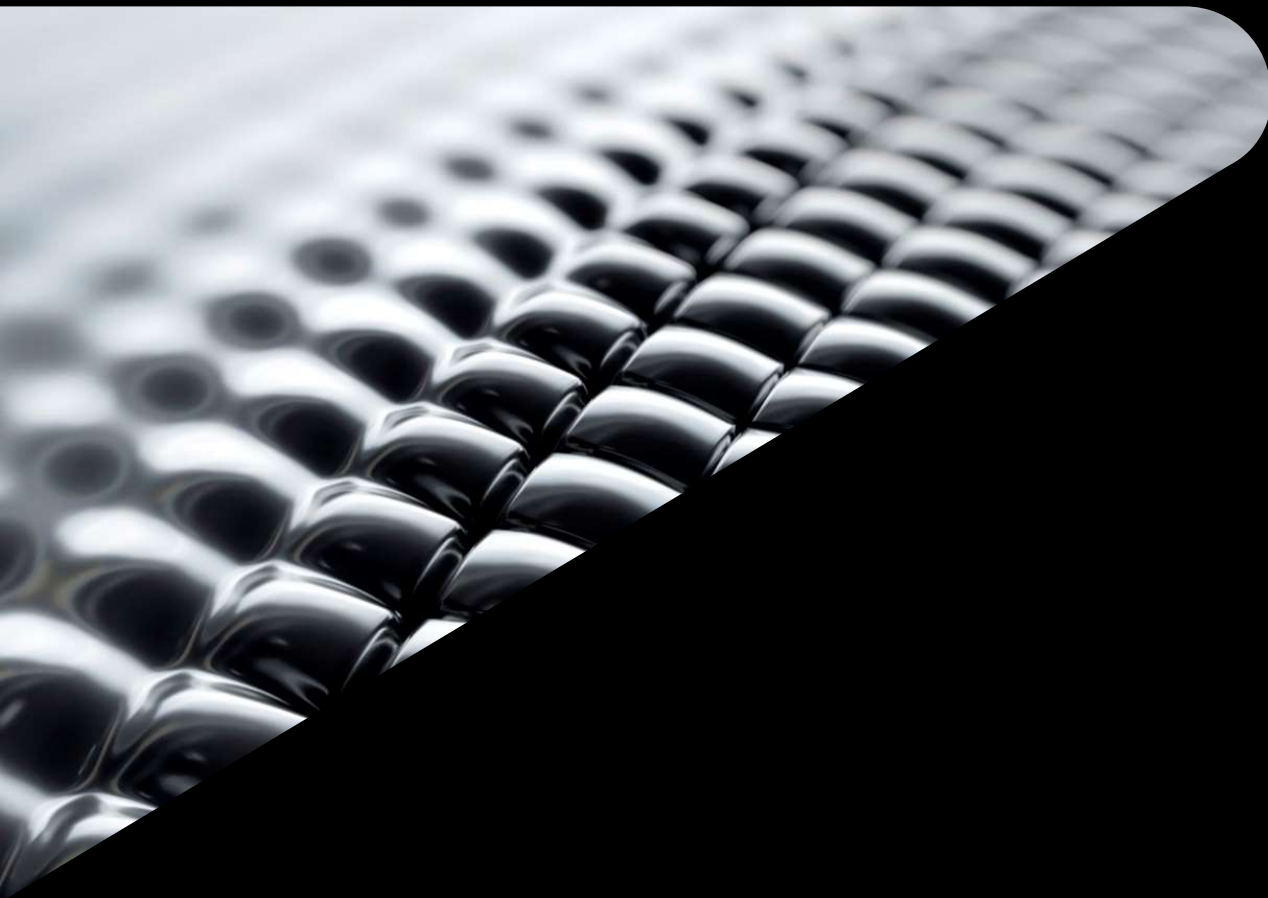
	A	B	C	D	E	F	G
1	Member	Thickness	Width	Length	Stock Number [Project]	Material [Physical]<material></material>	Hole1
2	[Primary]	1.5 in	3.5 in	12 in		Generic	Compute
3	Setup	3.5 in	5.5 in	18 in	2x6x12	Pine, Southern	Suppress
4	1x4x12	0.75 in	3.5 in	12 in	2x4x12	Pine, Southern	Suppress
5	2x4x12	1.5 in	3.5 in	12 in	2x4x12	Pine, Southern	Suppress
6	2x10x12	1.5 in	9.5 in	12 in	2x4x12	Pine, Southern	Suppress

Add and rearrange Rows in Excel



# Demonstration – Dimensioned Lumber

Organizing Model States



# Bend it Like Beckham

Animating Model States

# Access Model States via the Inventor API

- The Model States collection of objects is stored inside a Document's ComponentDefinition
  - These can also be accessed via Occurrences in an assembly

```
'Set up the Document Settings to access the Model States
Dim oDoc = ThisDoc.Document
Dim oCompDef As ComponentDefinition = oDoc.ComponentDefinition

Dim oModelStates As ModelStates = oCompDef.ModelStates
Dim oModelState As ModelState

'Cycle through each Model State
For Each oModelState In oModelStates
```

```
Dim oDoc = ThisDoc.Document

Dim oCompDef As AssemblyComponentDefinition
oCompDef = oDoc.ComponentDefinition

Dim oOccs As ComponentOccurrences
oOccs = oCompDef.Occurrences

Dim oOcc As ComponentOccurrence

'This code will cycle through each ComponentOccurrence
'and display the name of each Model State in that Occurrence
For Each oOcc In oOccs

    'Must access the Occurrence Document
    oOccDoc = oOcc.Definition.Document
    'Must access the ComponentDefinition
    oOccCompDef = oOccDoc.ComponentDefinition
    'Can now access the ModelStates collection from
    'the Component Definition
    oOccModelStateCount = oOccCompDef.ModelStates.Count

    For i = 1 To oOccModelStateCount

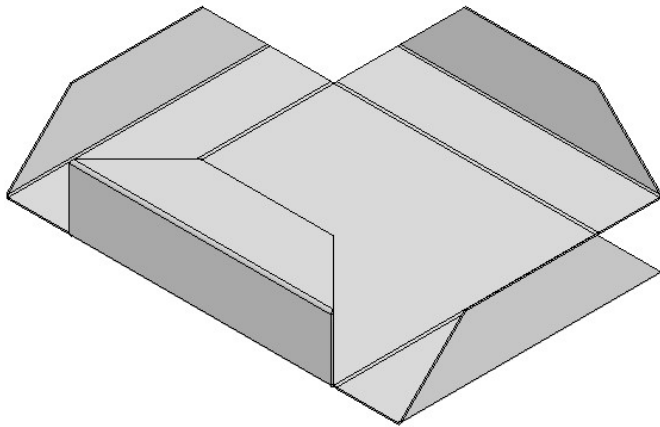
        oModelState = oOccCompDef.ModelStates.Item(i)
        Logger.Info(oModelState.Name)
```

# Set a Timer and Animate those States!

## Utilize the Power of iLogic

- Set a timer delay to control the speed of the animation
- Activate each Model State and update the model via iLogic

```
'Delay the activation of the next Model State (in milliseconds)  
Threading.Thread.Sleep(1500) 'in ms
```



```
'Cycle through each Model State and activate them in order  
For Each oModelState In oModelStates  
    If oModelState.Name <> "[Primary]" Then  
        'If oModelState.Name <> "Master" Then 'For Inventor 2022  
            oModelState.Activate  
        End If  
  
        'Update the document to view the active Model States  
        InventorVb.DocumentUpdate()  
  
        'Delay the activation of the next Model State (in milliseconds)  
        Threading.Thread.Sleep(1500) 'in ms  
    Next
```





# Demonstration – Sheet Metal Box

Animating Model States



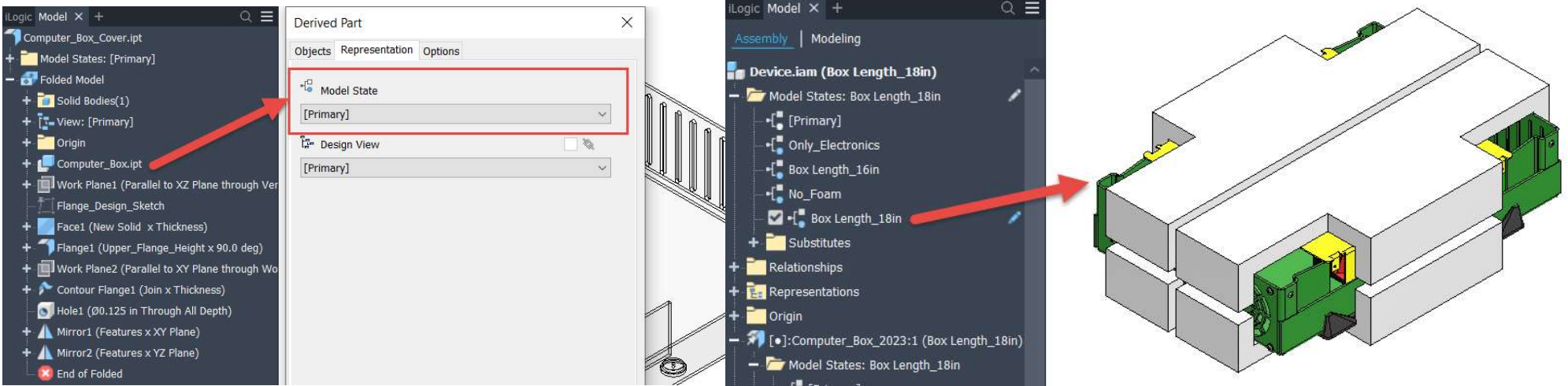
# Extra Fun – Derive vs. Linking Model States

Working Across Models

# Derive Only Utilizes One Model State

Typically the Active Model State when Executing the Derive

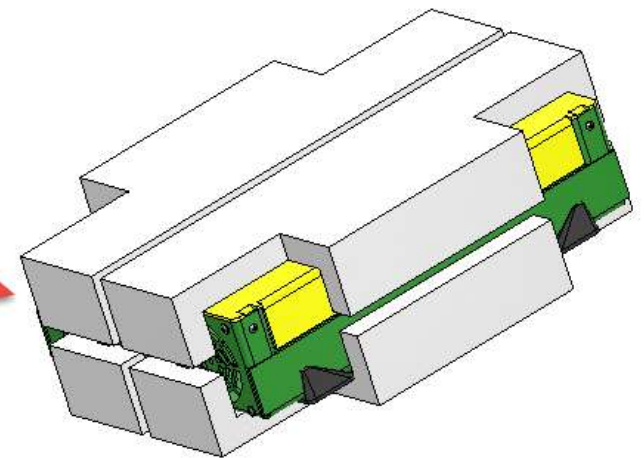
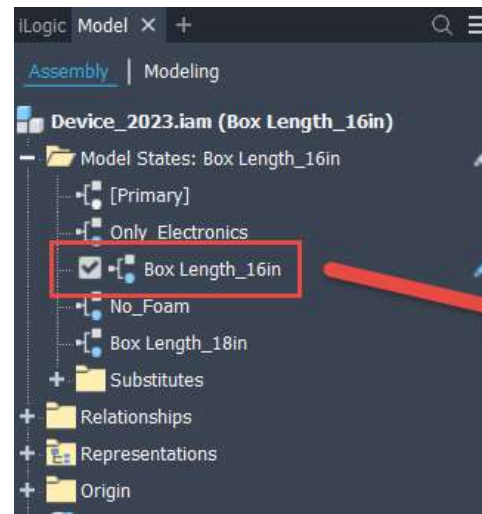
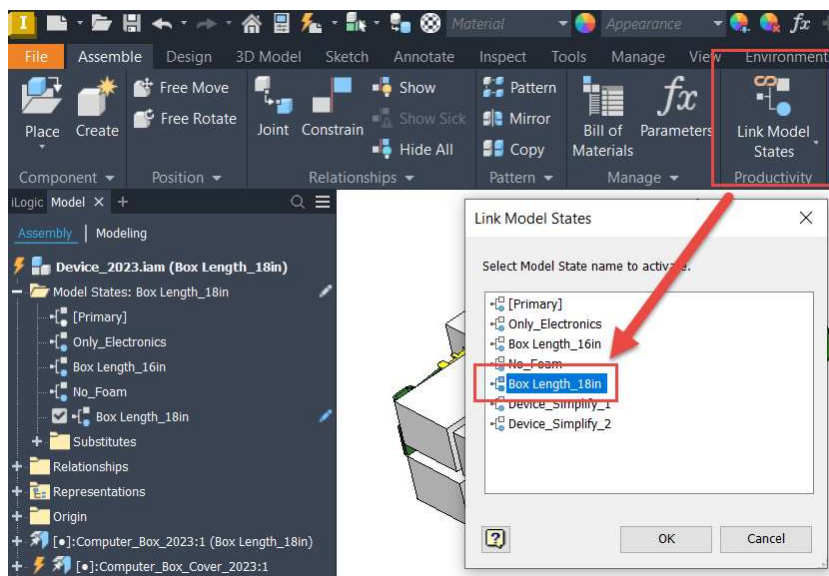
- When Deriving a design into another, only **ONE** Model State can be referenced
- Derived designs will **NOT** update when the Model State is changed in the referenced design
- Best left for designs with a single final variation / version



# Linking Model States is Great for Families of Designs

Sometimes the Setup Time is Worth it

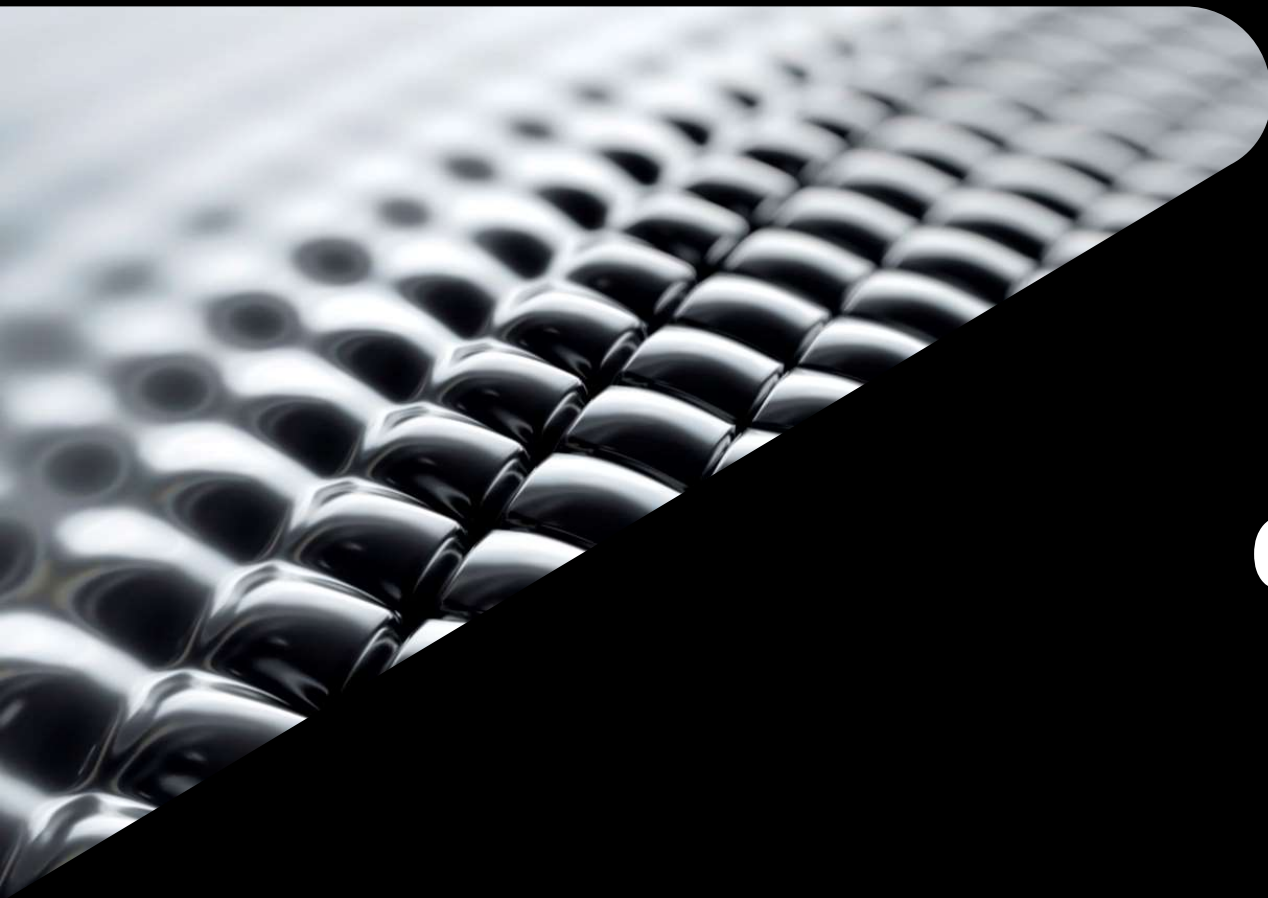
- If using **IDENTICALLY** named Model States, these can be Linked together in assemblies
  - Allows for rapid modification of a design with configurations
  - Can be a bit tedious as components must be linked to subassemblies first, before moving further up the assembly tree





# Demonstration – Computer Box / Cover

Linking Model States



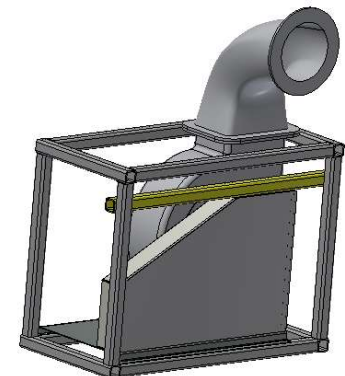
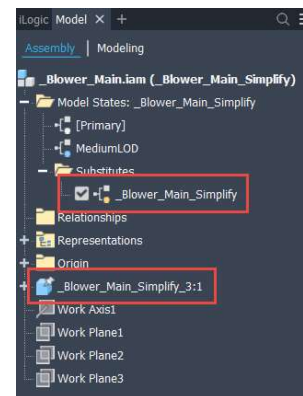
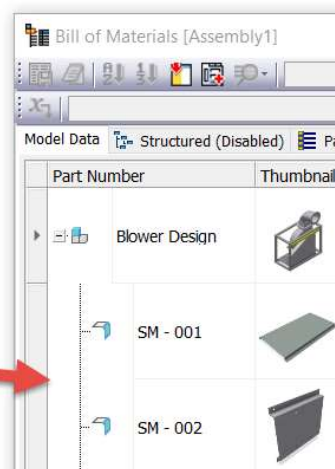
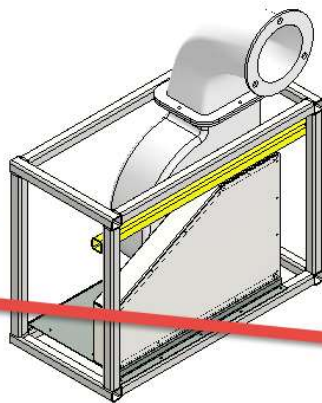
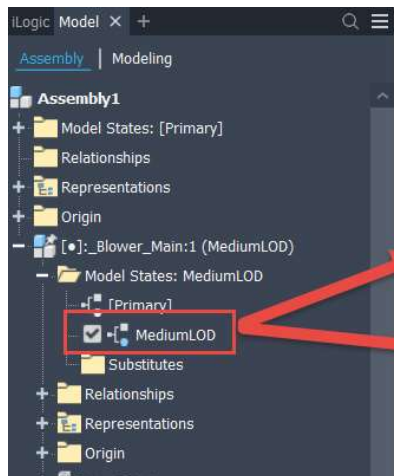
# **Extra Fun – Working with Converted LODs**

Transitioning to 2022 & Beyond

# Converted Custom LODs can be Useful Model States

LOD means “Level of Detail”

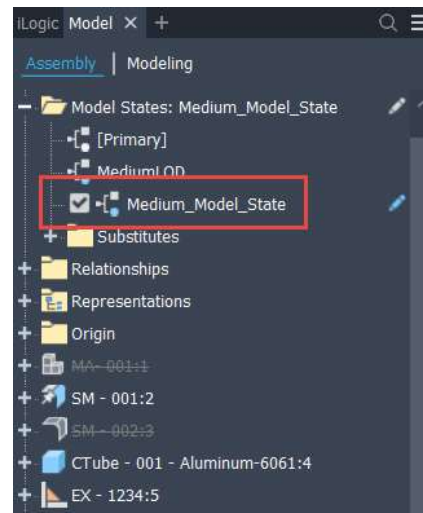
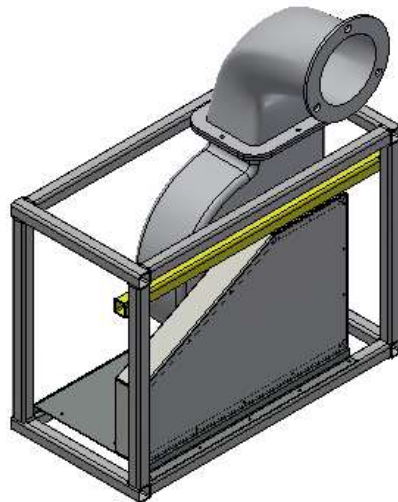
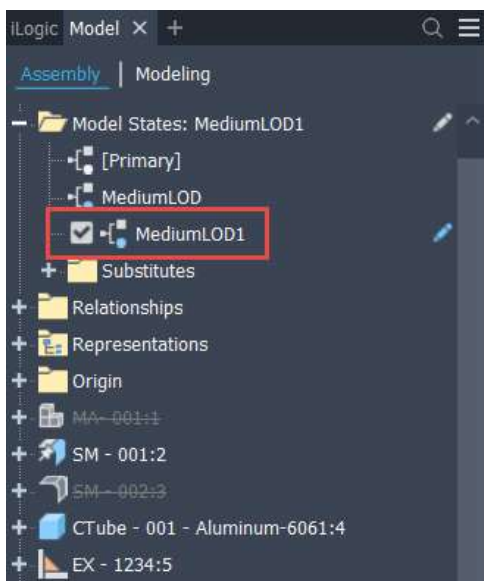
- Custom LODs get converted to Custom Model States starting in 2022
- While these can be a pain in the original assembly, they can be quite useful as subassemblies
  - Converted LODs will launch the “Primary” Model State when viewing the BOM in the original assembly, which is NOT great for large assemblies
  - When placed as subassemblies, will still display the entire BOM, even as a Substitute





# Making a Converted LOD Behave Like a Model State

- By Copying an existing Converted LOD, the resulting Model State actually behaves like a Model State
  - The Primary Model State will no longer be activated to view the BOM
  - Suppressed components will now report a quantity of 0 in the BOM



This screenshot shows the Bill of Materials (BOM) for the blower assembly. The BOM is displayed in a table format, showing the hierarchy of components and their quantities. The 'Medium\_Model\_State' is selected, and the BOM is filtered to show only the parts in this state. The 'SM - 001' part is highlighted with a red box, and its quantity is shown as 1. The BOM also lists other components like 'SM - 002', 'MA - 001', 'CTube - 001 - Aluminum-6061', 'EX - 1234', 'Fan - 0062', 'Shaft Design', 'Propeller Shaft Design', 'Bolted Connection Drive Casing', 'Bolted Connection Motor Mount 1', and 'Bolted Connection Motor Mount 2'.

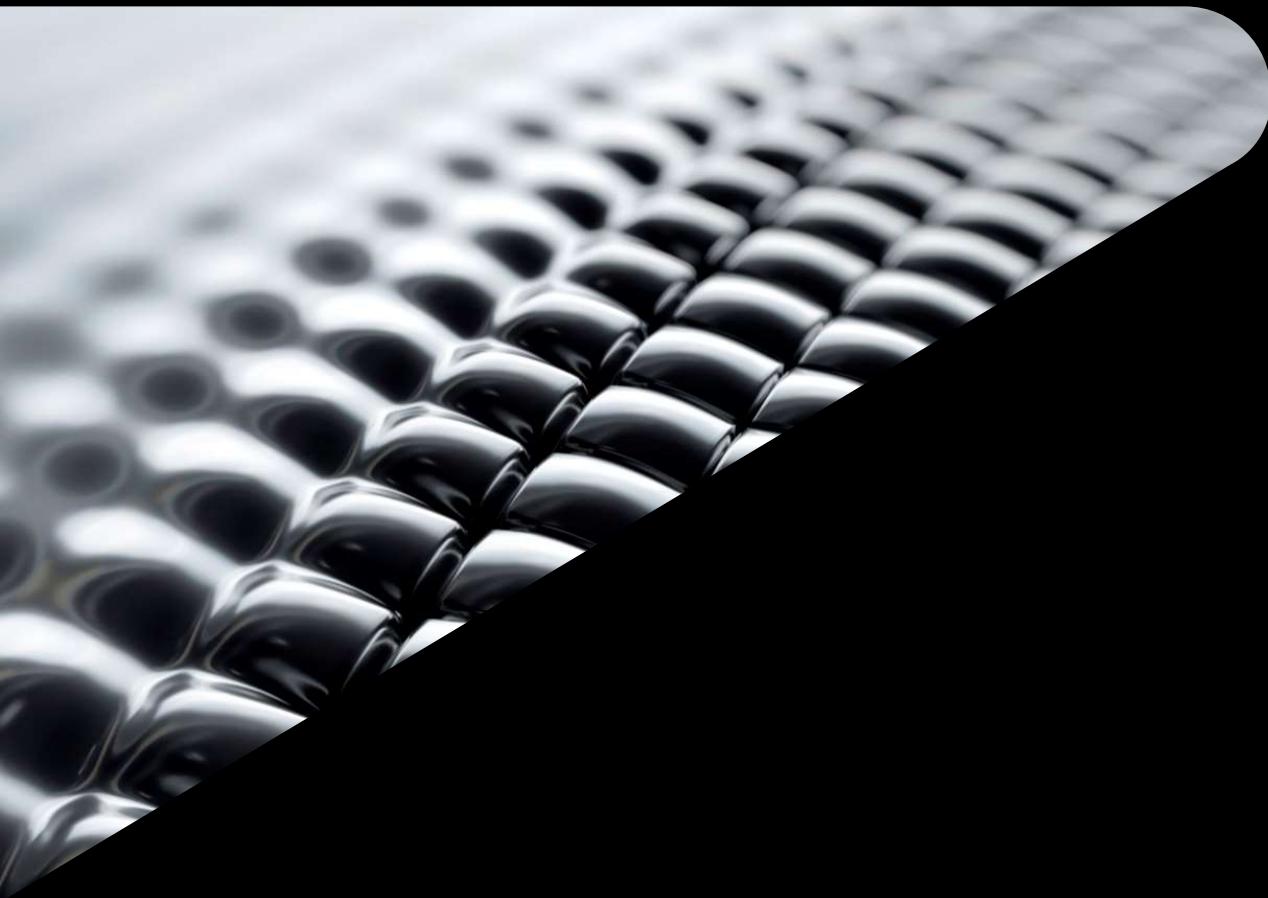
Part Number	BOM Structure	Unit	QTY	QTY	St
SM - 001	Normal	Each	1	1	
SM - 002	Normal	Each	0	0	
MA - 001	Normal	Each	0	0	
CTube - 001 - Aluminum-6061	Normal	Each	1	1	
EX - 1234	Normal	Each	1	1	
SM - 003	Normal	Each	1	1	
Fan - 0062	Normal	Each	0	0	
Shaft Design	Phantom	Each	0	0	
Propeller Shaft Design	Phantom	Each	0	0	
Bolted Connection Drive Casing	Phantom	Each	0	0	
Bolted Connection Motor Mount 1	Phantom	Each	0	0	
Bolted Connection Motor Mount 2	Phantom	Each	0	0	





# Demonstration – Blower Assembly

Converted LODs to Model States



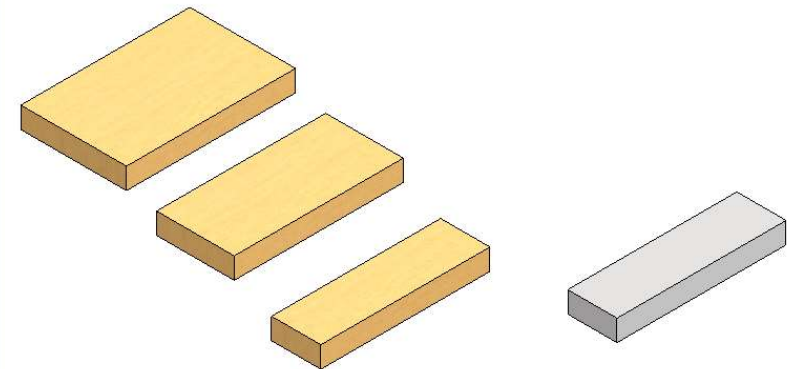
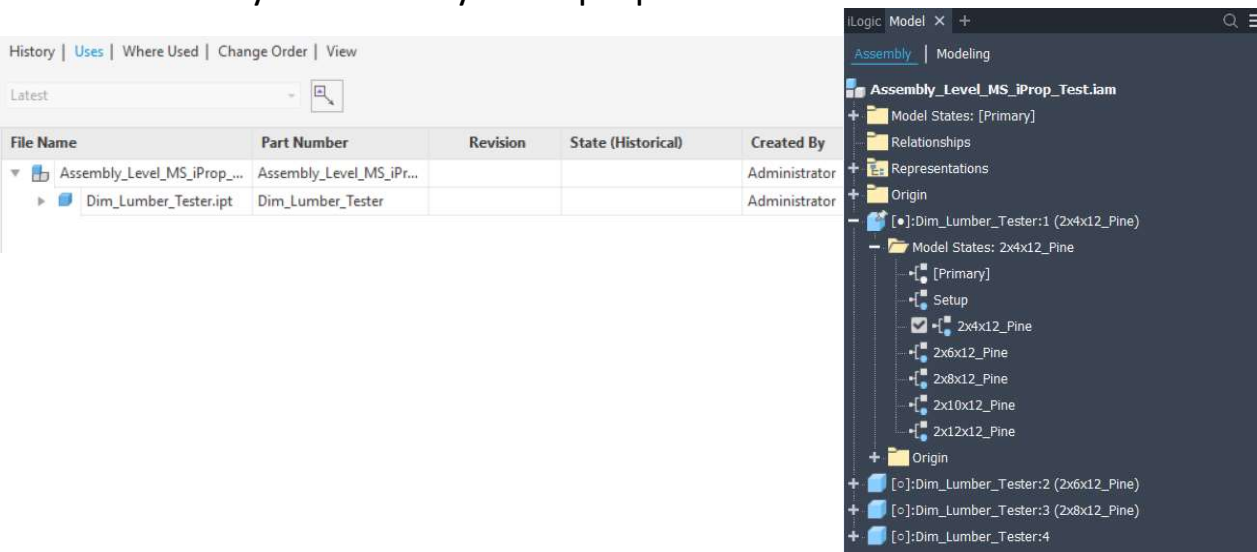
# Extra Fun – Tracking Model States in Vault

Without Using Vault Items

# Vault has Some Limitations Viewing Model States

## Can Only View Properties of the Primary State

- When viewing the “Uses” tab of a Vaulted assembly, one will only view the properties of the Primary Model State
  - Regardless of a single occurrence or multiple occurrences, only one component row is visible
  - Only the Primary State properties are visible



# Using Inventor API – Gather the Active Model States

- By cycling through all Occurrences in an assembly, we can find the active Model States
  - Will be searching only top-level components, as subassemblies can be searched individually
- Find components with unique active Model States
- Assign these to an ArrayList, to be used later

```
'Cycle through all the top-level occurrences and collect the components into the lists
For Each oOcc In oOCCs

    'Check against the Primary, Master and Final Design Model States
    If Not oOcc.ActiveModelState = "[Primary]" And Not oOcc.ActiveModelState =
"Master" -
        And Not oOcc.ActiveModelState = "Setup" And Not oOcc.ActiveModelState =
"Final_Design" Then
        'Find the file name, including the file extension to later find in Vault
        oOccDoc = oOcc.Definition.Document
        'Find the position of the last backslash in the path
        FNamePos = InStrRev(oOccDoc.FullFileName, "\", -1)
        'Get the file name with the file extension
        oOccDocName = Right(oOccDoc.FullFileName, Len(oOccDoc.FullFileName) -
FNamePos)
        'Add the component names and corresponding unique Model States to the
array lists

        list

        'Check the current combination of name and model state against the current
        'Gather the current combination
        Dim ModelState_ID As String = oOccDocName & "_" & oOcc.ActiveModelState

        'Check to see if the ArrayList contains the ModelState_ID and add if it
        does not.
        If File_Names_Active_Model_States.Contains(ModelState_ID) = False Then
            File_Names_Active_Model_States.Add(ModelState_ID)
        End If
    End If
Next
```

# Deposit the ArrayList to a Custom Assembly iProperty

Either to a Single iProperty or Multiple iProperties

```
'Assign the values from the array lists to an iProperty in the Assembly
iProperties.Value("Custom", "Active_Component_Model_States") = ""

For i = 0 To File_Names_Active_Model_States.Count - 1
'    Logger.Info(i) 'For testing only
'    Logger.Info(File_Names(i)) 'For testing only
'    'Get the current value for the custom iProperty
    Initial_String = iProperties.Value("Custom",
"Active_Component_Model_States")

    'Add the new values to the text string
    iProperties.Value("Custom", "Active_Component_Model_States") =
Initial_String & File_Names_Active_Model_States.Item(i) & ", "
Next
```

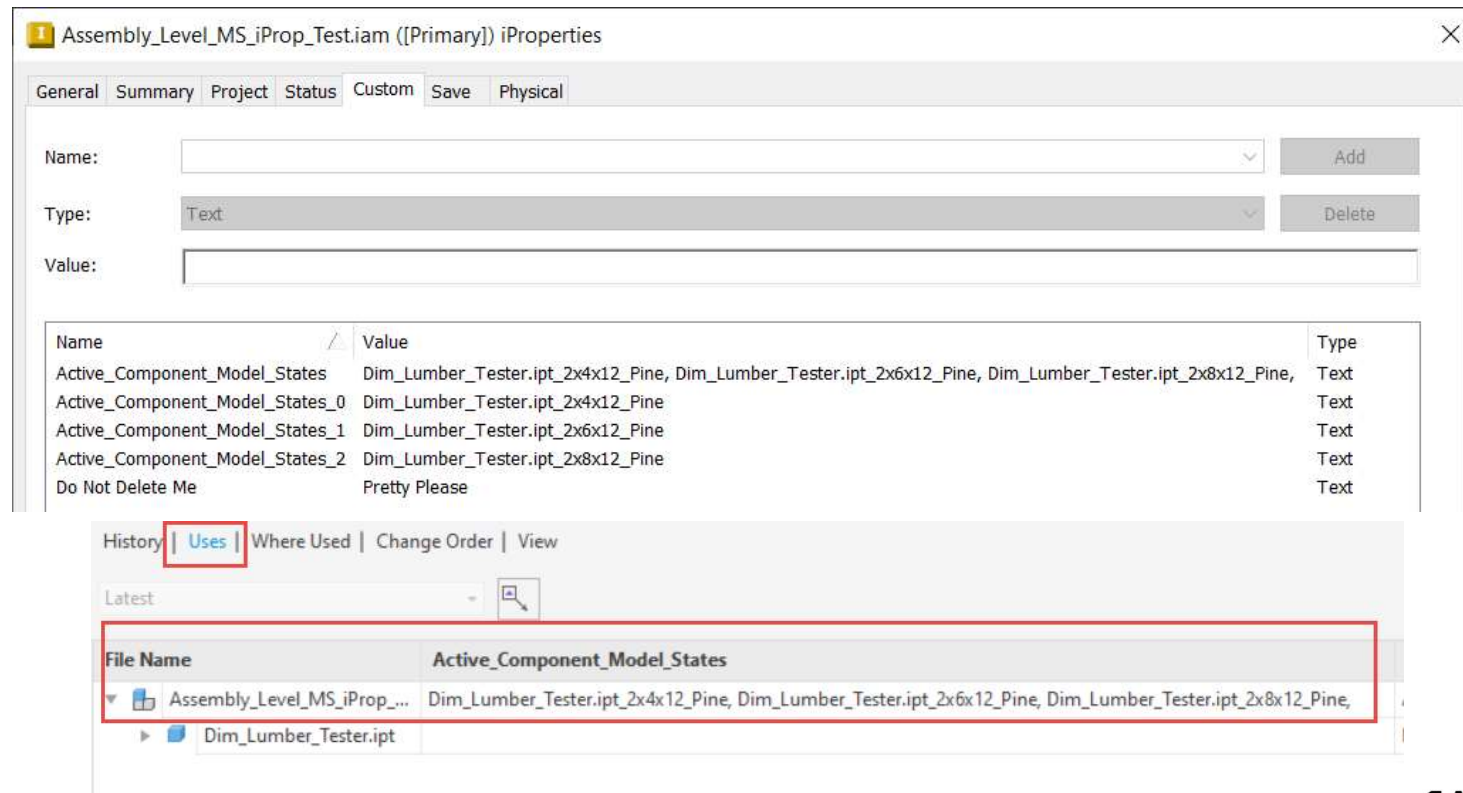
```
'Remove any pre-existing iProperties of this type, in case someone switches
'a component to use "[Primary]", "Master" or the "Final_Design" Model States
Dim oPropSets As PropertySets = oDoc.PropertySets
Dim oUserDefProps As PropertySet = oPropSets.Item("Inventor User Defined
Properties")
For Each oiProp In oUserDefProps
    If oiProp.Name.Contains("Active_Component_Model_States") Then
        oiProp.Delete
    End If
Next

'Assign the values from the Array to an iProperty in the Assembly
For i = 0 To File_Names_Active_Model_States.Count - 1
'    Logger.Info(i) 'For testing only
'    Logger.Info(File_Names(i)) 'For testing only
'    'Create a new iProperty for each component in the array lists
    iProperties.Value("Custom", "Active_Component_Model_States" & "_" & i)
= File_Names_Active_Model_States.Item(i)
Next
```

# The Assembly iProperty(ies) Make Viewing MUCH Easier

## Both within Inventor and in Vault

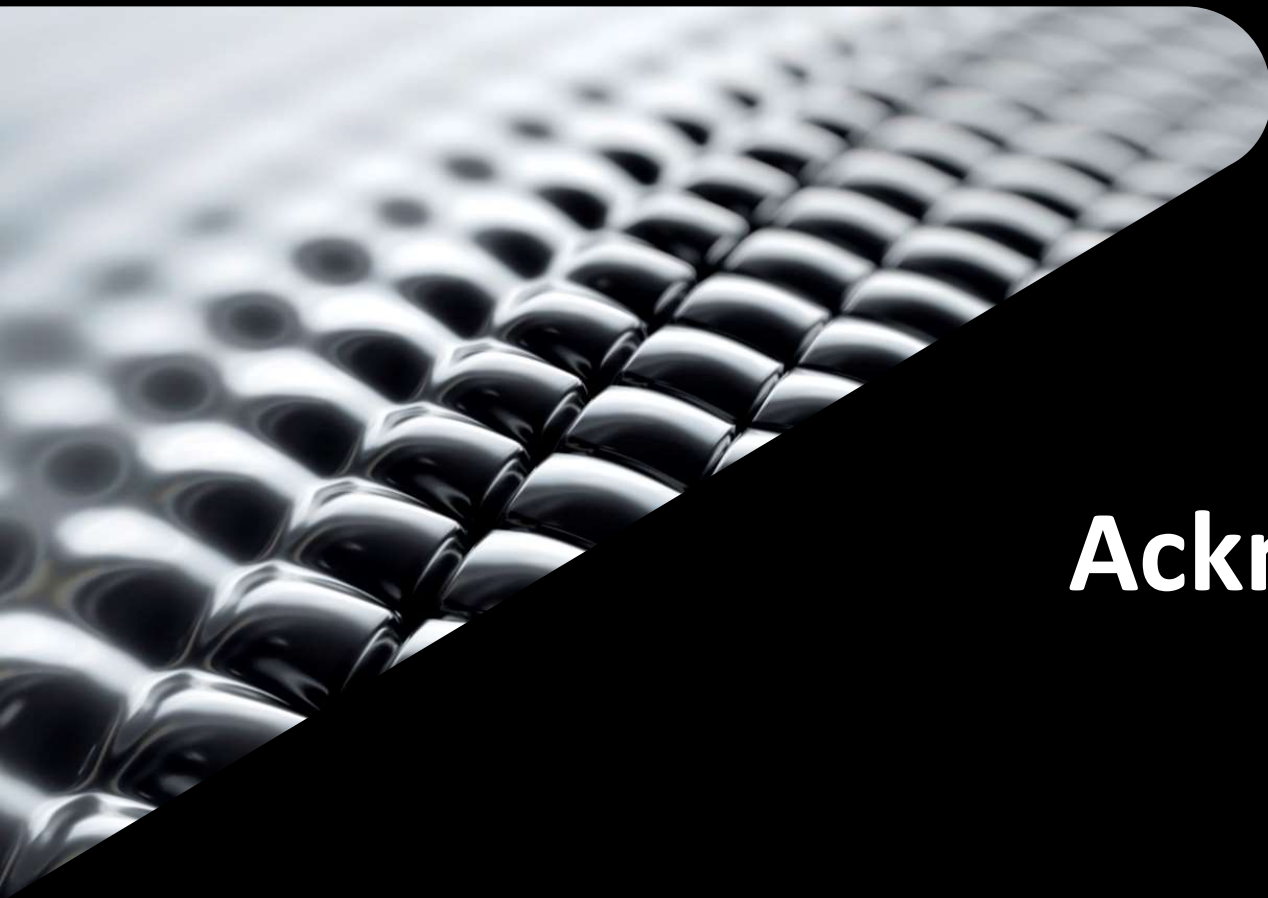
- The properties must be added and mapped in Vault to become visible





# Demonstration – Modular Wall

Displaying Model States in Vault



# Acknowledgements

Nothing is Done in a Vacuum



# Special Thanks!

- God for this wonderful opportunity and truly every breath that I'm given
- Tim Wilson, and all my former bosses, for allowing me time and space to get crazy
- James Jung, my colleague who assisted with my Vault property mapping
- All my customers, who asked great MS questions this past year
- Andrew Humiston, who was the genesis for the MS iProperty idea

Thank  
you



**Any Questions**



Autodesk and the Autodesk logo are registered trademarks or trademarks of Autodesk, Inc., and/or its subsidiaries and/or affiliates in the USA and/or other countries. All other brand names, product names, or trademarks belong to their respective holders. Autodesk reserves the right to alter product and services offerings, and specifications and pricing at any time without notice, and is not responsible for typographical or graphical errors that may appear in this document.

© 2022 Autodesk. All rights reserved.