



The Decoding Lab: iLogic Intermediate Session

Steve Olson – MESA Inc.

DV1673-L: iLogic is a very powerful part of Inventor that is used for helping users automate segments of their design process. This lab will look at iLogic's ability to copy and replace components and error checking. In this lab, students will create iLogic Rules that turn an Inventor model into a configurator by performing a Save Copy As function on parts and assemblies. The class will also explore iLogic's ability to perform error checks on designs, and show different ways to communicate these errors to the user.

Learning Objectives

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

- Create iLogic Rules that perform Save Copy As operations on parts and assemblies
- Use iLogic Form tools to add Tabs to a Form and Buttons for Rules
- Understand the concepts of adding Error Checking iLogic Rules to your models
- Incorporate verbal warnings for users in iLogic Rules

About the Speaker

Steve Olson is the Manager, Training Services for MESA, Inc. (an Autodesk Silver Partner and an Autodesk Authorized Training Center). He is an Autodesk Certified Instructor and Certified Inventor and AutoCAD Professional. He has been training students in Autodesk Products, for five years. Steve's industry experience comes from five years of service at Fleetwood Folding Trailers. As Manager, Training Services, Steve has created several courses taught at MESA. He also has received accreditation as an Autodesk Consultant Specialist, and has five years of experience working with customers to analyze their utilization of Inventor's capabilities. Steve has presented topics related to Autodesk software at Autodesk University, Western Pennsylvania Inventor User Group Meetings, Northeast Ohio Inventor User Group Meetings, MESA U, and MESA sponsored events.

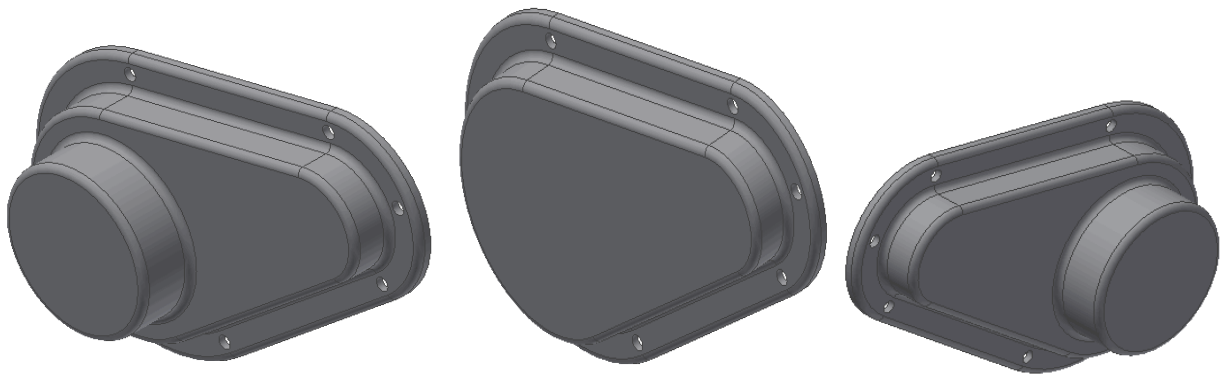
steve.olson@mesa-cad.com

Introduction

This class will look at iLogic's ability to be used as a configurator by looking at two Design Challenges. The first Design Challenge will be to create a configurator for a single part that will allow the user to enter the design parameters and save a copy of the part. The second Design Challenge will investigate a similar process for an assembly. The assembly configurator will need to save the assembly as well as each component. Each Design Challenge solution will include an iLogic Rule and Form.

Design Challenge #1

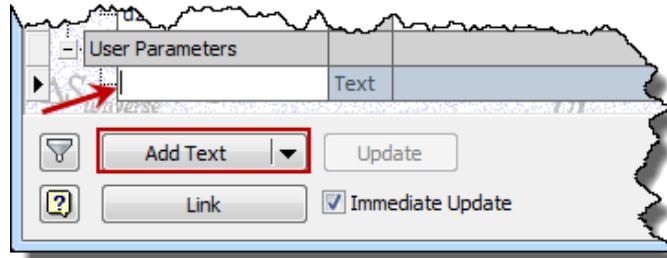
Our company has a standard component that has many variations. We want to create a configurator that will allow users to input key variables, and save that variation of the part. The part has four key model parameters, one optional feature and left and right orientations.



1. Open Cover.ipt
2. Create these User Parameters
 - A. Click on the Parameter icon in the Quick Access Tool bar



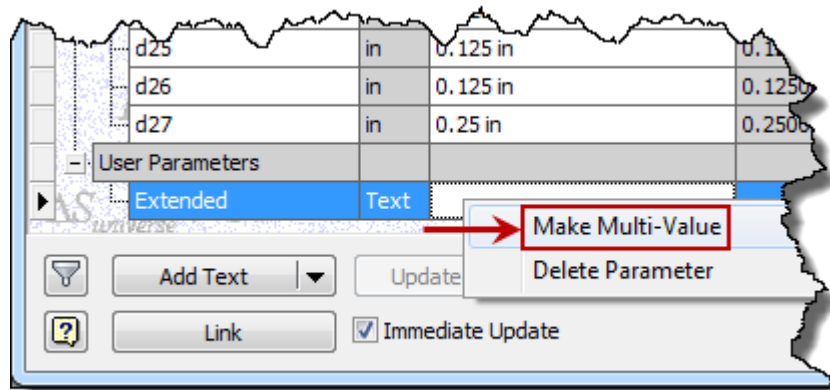
- B. At the bottom of the Parameter Dialog box, click on the drop down button next to “Add Numeric.” From the drop down list, Choose “Add Text”



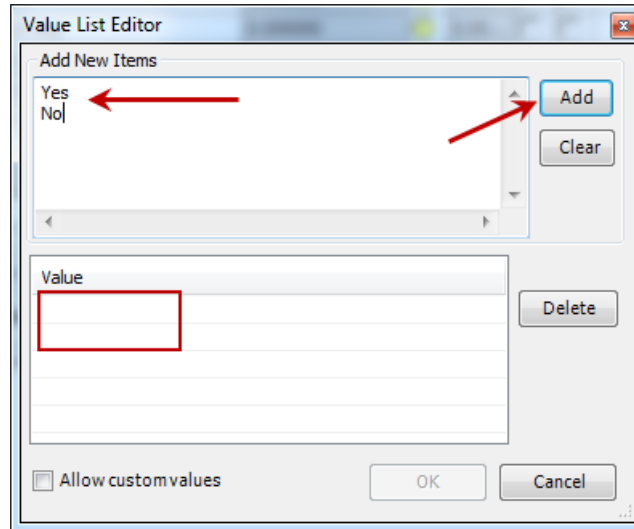
- C. In the line for the new parameter enter a name of one of the following parameters.

Name	Type	Value(s)
Extended	Text	Yes No
Orientation	Text	Left Hand Right Hand
NewPartNumber	Text	<User’s Choice>
ProjectNumber	Text	<User’s Choice>

- D. If the parameter has multiple values, right click in the Equation column of that row and choose “Make Multi-Value.”



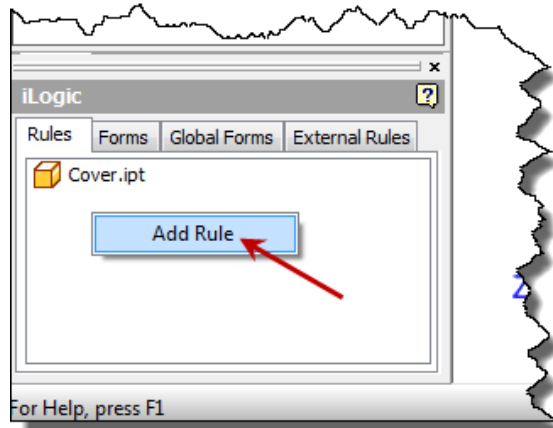
- E. In the Value List Editor Dialog, enter the possible values (one per line) in the “Add New” Items pane. Then click “Add” to move the values to the lower portion of the box.



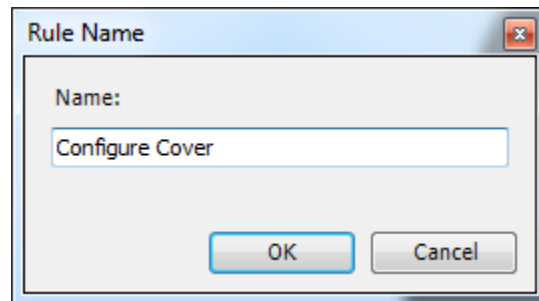
- F. Repeat the previous steps for each parameter.

3. Create Configure Cover Rule

- A. In the Rule tab of the iLogic browser, right click and choose “Add Rule.”



- B. In the “Rule Name” dialog, enter “Configure Cover” as the name for this rule.



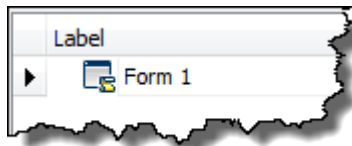
C. In the text pane of the Rule Editor, enter the following text.

Note: Remember to use Snippets and other shortcuts to help enter the text quicker and eliminate the possibilities of misspellings.

```
'Suppress or Inclusion of Large Motor Option and Fillet
If Extended = "Yes" Then
    Feature.IsActive("Large Motor Option") = True
    Feature.IsActive("Large Motor Fillet") = True
Else
    Feature.IsActive("Large Motor Option") = False
    Feature.IsActive("Large Motor Fillet") = False
End If
```

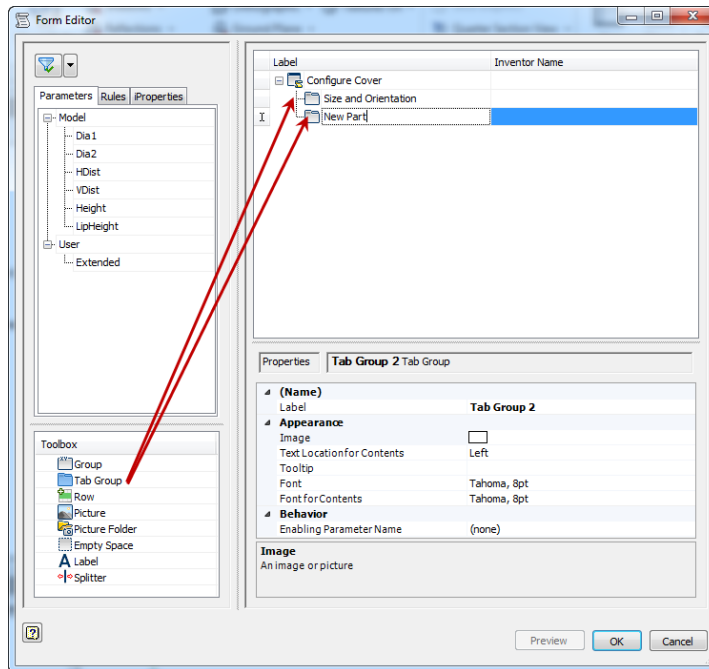
```
'Controls whether the part is left hand or right hand
'through controlling a mirror Feature
If Orientation = "Left Hand" Then
    Feature.IsActive("Mirror1") = False
Else
    Feature.IsActive("Mirror1") = True
End If
```

4. Test Configure Cover Rule by changing parameter values to see if the model reacts in expected ways.
5. Create Configure Cover Form
 - A. In the Forms tab of the iLogic browser, right click and choose "Add Form."
 - B. In the Form Editor, change the name of the form to "Configure Cover" by editing the text under the Label pane of the dialog.



C. Drag and drop two Tab Group element from the Toolbox to the label pane of the dialog.

D. Rename each Tab Group element.



E. Drag and drop Parameter elements from the Parameter tab on the left.

F. Repeat until the Form matches the screen shots below.

Note: If you parameter names are not very descriptive, the labels can be changed by editing the Label column of the dialog box, as seen below.

Label	Inventor Name
Configure Cover	
Size and Orientation	
Diameter 1	Dia 1
Diameter 2	Dia 2
Horizontal Center Dist	HDist
Vertical Center Dist	VDist
Extended	Extended
Orientation	Orientation
New Part	
Project Number	ProjectNumber
New Part Number	NewPartNumber

Configure Cover

Size and Orientation | New Part

Diameter 1: 2 in

Diameter 2: 4 in

Horizontal Center Dist: 4 in

Vertical Center Dist: 2 in

Extended: Yes

Orientation: Left Hand

Done

Configure Cover

Size and Orientation | New Part

Project Number: ABC128

New Part Number: 4701-226

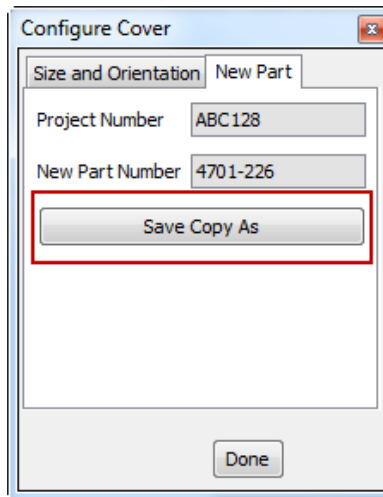
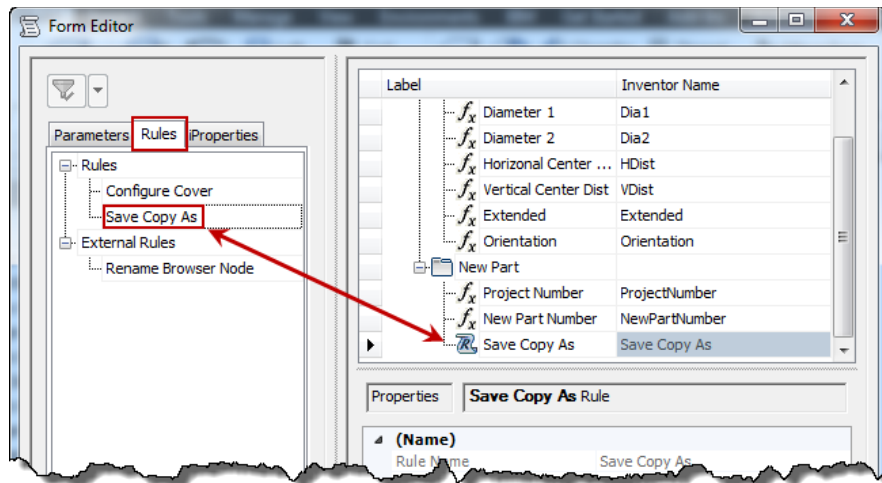
Done

6. Test Configure Cover Form by changing parameter values to see if the model reacts in expected ways.
7. Create Save Copy As Rule
 - A. In the Rule tab of the iLogic browser, right click and choose "Add Rule."
 - B. In the "Rule Name" dialog, enter "Save Copy As" as the name for this rule.
 - C. In the text pane of the Rule Editor, enter the following text.

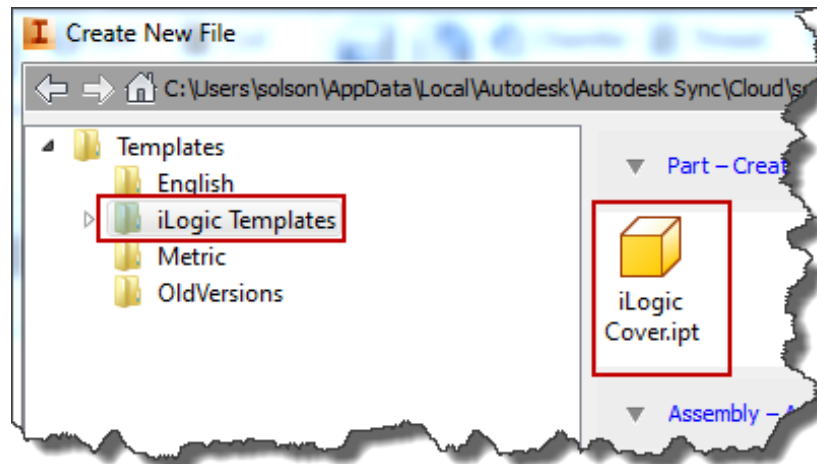
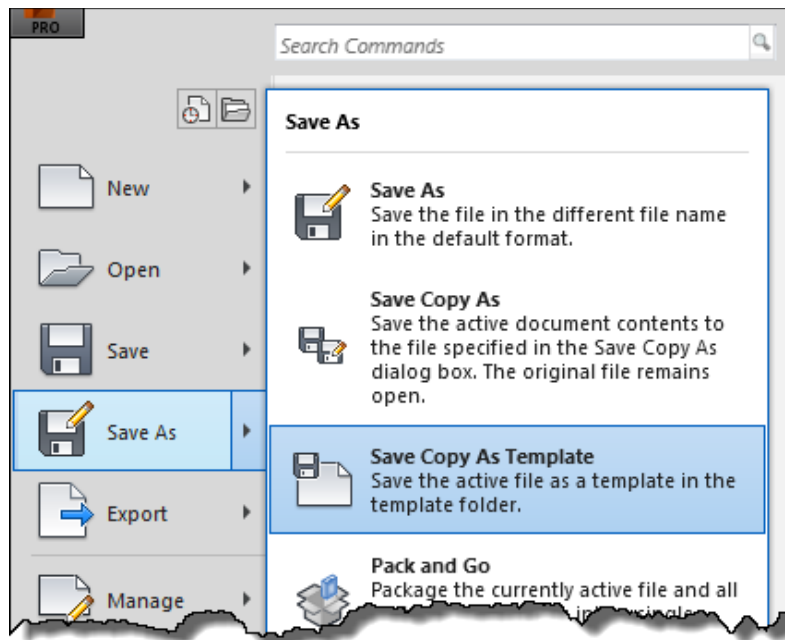
```
'Sets the Path for the New File  
NewPath = ThisDoc.WorkspacePath() & "\" & ProjectNumber  
  
'Checks to see if the path to save to exists, if not  
creates the path  
If (Not System.IO.Directory.Exists(NewPath)) Then  
    System.IO.Directory.CreateDirectory(NewPath)  
End If  
  
'Saves a Copy of the File with the New Name  
ThisDoc.Document.SaveAs(NewPath & "\" & NewPartNumber .  
& ".ipt" , True)
```

8. Add a Button to the New Part Tab of the Configure Cover Form

- A. In the Forms tab of the iLogic browser, right click on the “Configure Cover” button and choose “Edit.”
- B. From the “Rule” tab, drag and drop the “Save Copy As” rule to the “New Part” tab of the form.



9. (Optional) Save Cover.ipt as a template and test



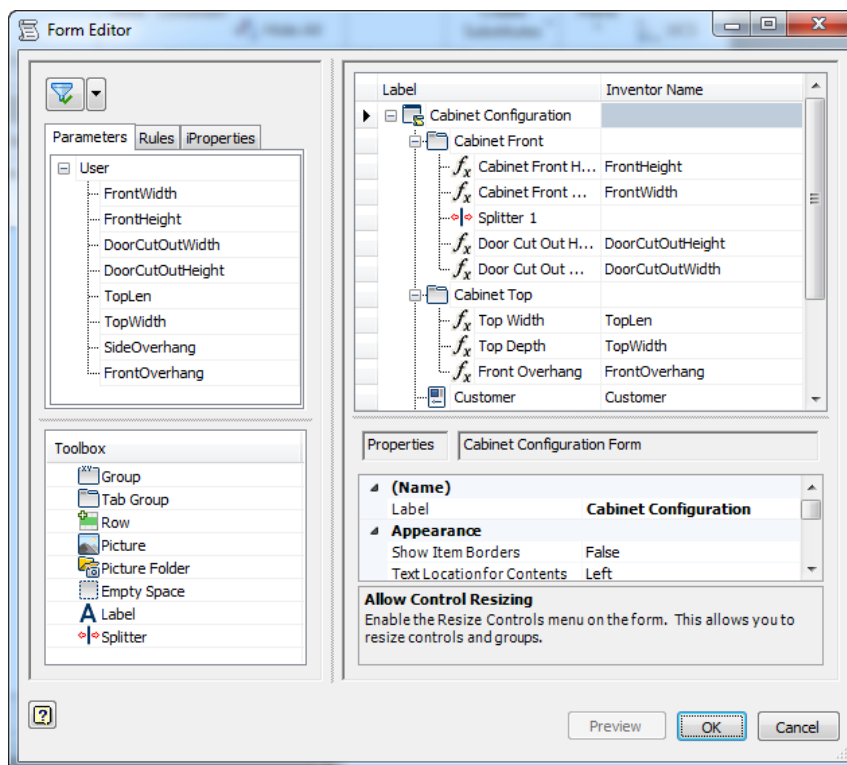
Design Challenge #2

Our company makes furniture for the travel trailer manufactures. One common piece of furniture we make is a storage cabinet subassembly. The cabinets can vary in size between units. We want to create a configurator that will allow users to input key variables, and save that variation of the assembly and its components. The cabinet has several key variables and doors that cover the openings. Our company only purchases certain sizes of cabinet doors; therefore, part of our solution should be to verify if one of the standard door panels can be used to cover the opening in the front of the cabinet.

1. Open iLogic Cabinet.iam
2. Note these parameters have already been created

Name	Type	Value(s)
Customer	Text	<User's Choice>
NewAssemblyPartNumber	Text	<User's Choice>
NewFrontPartNumber	Text	<User's Choice>
NewSidePartNumber	Text	<User's Choice>
NewTopPartNumber	Text	<User's Choice>

3. Note Cabinet Configuration Form has been previously created



Cabinet Configuration

Cabinet Front | Cabinet Top

Cabinet Front Height 26 in

Cabinet Front Width 30 in

Door Cut Out Height 14 in

Door Cut Out Width 13 in

Customer MESA

Assembly Part Number 4701-987

Front Part Number 4701-654

Top Part Number 4701-321

Side Part Number 4701-963

Done

Cabinet Configuration

Cabinet Front | Cabinet Top

Top Width 34 in

Top Depth 24 in

Front Overhang 1 in

Customer MESA

Assembly Part Number 4701-987

Front Part Number 4701-654

Top Part Number 4701-321

Side Part Number 4701-963

Done

4. Open Save Copy As Rule to Edit

- A. In the Rule tab of the iLogic browser, right click on the “Save Copy As” rule and choose “Edit Rule.”
- B. The rule has been started. Edit the text of the rule to match the text below.

```

'Saves the Active Document
ThisDoc.Save

'Sets the Directory for Saving the New Files
AssemDir = ThisDoc.WorkspacePath() & "\" & Customer & "\"
CompDir = AssemDir & "Components\"

'Checks to see if that directory exists, if not, it is created
If(Not System.IO.Directory.Exists(CompDir)) Then
    System.IO.Directory.CreateDirectory(CompDir)
End If

'Identifies the Front Component, Grabs the Document, and Grabs the File Name
compoFront = Component.InventorComponent("Front")
docFront = compoFront.Definition.Document
OldFront = docFront.FullFileName

'Identifies the Side Component, Grabs the Document, and Grabs the File Name
compoSide = Component.InventorComponent("RightSide")
docSide = compoSide.Definition.Document
OldSide = docSide.FullFileName

'Identifies the Top Component, Grabs the Document, and Grabs the File Name
compoTop = Component.InventorComponent("Cabinet Top")
docTop = compoTop.Definition.Document
OldTop = docTop.FullFileName

'Generates New Assembly and Component File Name
AssemblyFile = AssemDir & NewAssemblyPartNumber & ".iam"
FrontFile = CompDir & NewFrontPartNumber & ".ipt"
SideFile = CompDir & NewSidePartNumber & ".ipt"
TopFile = CompDir & NewTopPartNumber & ".ipt"

'Copies Original Components to New File Names
System.IO.File.Copy( OldFront, FrontFile )
System.IO.File.Copy( OldSide, SideFile )
System.IO.File.Copy( OldTop, TopFile )

'Copies the Active Assembly
ThisDoc.Document.SaveAs(AssemblyFile , True)

'Replaces Old Components with New
Component.Replace("Front", FrontFile, True)
Component.Replace("RightSide", SideFile, True)

```

```
Component.Replace("Cabinet Top", TopFile, True)
```

```
'Sets the Part Number iProperty Of the New Files
```

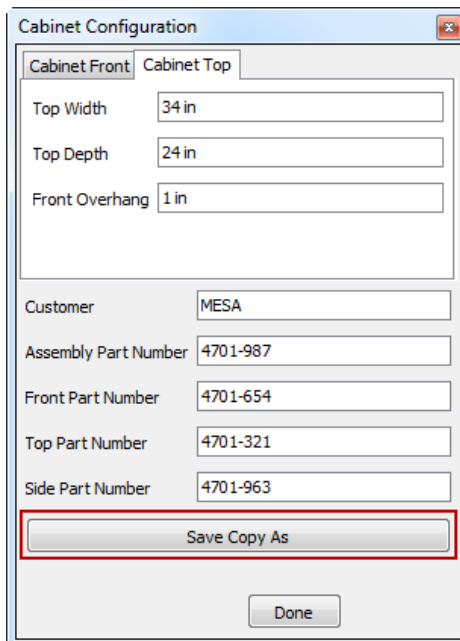
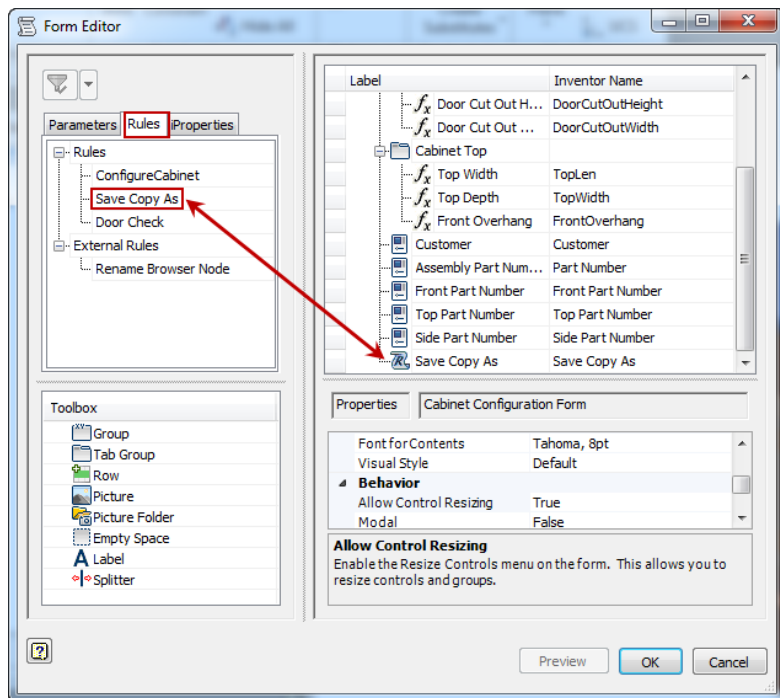
```
iProperties.Value("Front", "Project", "Part Number") = NewFrontPartNumber
```

```
iProperties.Value("RightSide", "Project", "Part Number") =  
NewSidePartNumber
```


```
iProperties.Value("Cabinet Top", "Project", "Part Number") =  
NewTopPartNumber
```

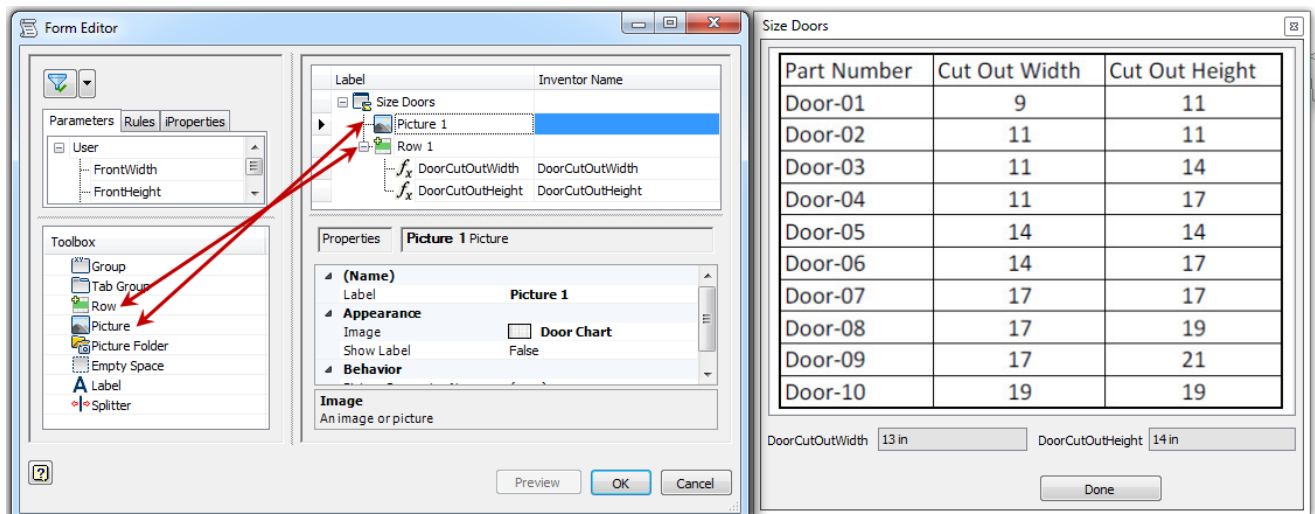
```
iLogicVb.RunRule("Door Check")
```

5. Add a button to the Configure Cabinet Form for the Save Copy As Rule and Test
 - A. In the Forms tab of the iLogic browser, right click on the “Configure Cabinet” button and choose “Edit.”
 - B. From the “Rule” tab, drag and drop the “Save Copy As” rule to the form.



6. Create a Size Door Form

- A. In the Forms tab of the iLogic browser, right click in the open space and choose "Add Form."
- B. Under the Label section, change the name of the form to "Size Doors."
- C. From the Toolbox, drag and drop a Picture element. Under the properties of the picture, click on the Image property. Then click on the  button to browser for the image. Find and select the Door Chart.jpg in the Documentation folder of the data set.
- D. From the Toolbox, drag and drop a Row element to the form. Then drag and drop the DoorCutOutWidth and DoorCutoutLength parameters from the parameter tab on the left. Drop them on the Row element.



7. Create Door Check Rule and Test

- A. In the Rule tab of the iLogic browser, right click and choose “Add Rule.”
- B. In the “Rule Name” dialog, enter “Door Check” as the name for this rule.
- C. In the text pane of the Rule Editor, enter the following text.

```

DoorWidth = DoorCutOutWidth + 1
DoorHeight = DoorCutOutHeight +1

i = iPart.FindRow("LeftDoor", "DoorWidth", "=", DoorWidth,
"DoorHeight", "=", DoorHeight)

i = iPart.FindRow("RightDoor", "DoorWidth", "=", DoorWidth,
"DoorHeight", "=", DoorHeight)

If i = -1 Then
    '___ Use windows voice command _____
    Dim objSPVoice, colVoices
    objSPVoice = CreateObject("SAPI.SpVoice")
    objSPVoice.Speak ("Unable to find appropriate Door
Assembly")

    i = MessageBox.Show("Unable to find the appropriate door.
Please change the door cutouts, And rerun the rule.", "Size
Doors", MessageBoxButtons.OK, MessageBoxIcon.Hand,
MessageBoxDefaultButton.Button1)

    iLogicForm.Show("Size Doors", FormMode.Modal)

End If

```

8. Edit “Edit Configure Cabinet” rule and add the following statement as the last line

- A. In the Rule tab of the iLogic browser, right click on the “Save Copy As” rule and choose “Edit Rule.”
- B. Add this text as the last line of the rule.

```
iLogicVb.RunRule ("Door Check")
```

9. (Optional) Use Save Copy As Template to save as a template and test

