AUTODESK **UNIVERSITY** 2013

# AutoCAD® Utility Design: Bending the Rules

Dan Leighton – DL Consulting

**UT1625-L**

The new generation of AutoCAD Utility Design introduced rules-driven model-based design to the utility industry. The rules engine within Utility Design provides enormous flexibility with a structured framework to create, edit, and manage the design rules. This lab will provide you with an overview of the rules engine, and then focus on several examples of practical rule definition for styling, material ordering, cost estimation, and feature identifiers. In addition, we will have a discussion of the practical considerations to define and manage rules.

## Learning Objectives

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

- Explain the types of rules supported in Utility Design
- Configure Utility Design rules for styling, material ordering, feature IDs, and more
- Understand the basics of Utility Design rules definition, use, and management
- Demonstrate the steps to edit and test Utility Design rules

## About the Speaker

Dan has spent over 25 years doing CAD, GIS, and analysis work. For the past two decades, Dan has also served in various project management, project marketing, and executive roles in the CAD, GIS, and database industries. Most recently, as the Principal Consultant at DL Consulting, Dan has been providing customized consulting, training, and implementation services to organizations using a variety of Autodesk® products including AutoCAD® Utility Design, Autodesk® Vault Workgroup, AutoCAD®, and AutoCAD® Map 3D. Dan can be reached at: *dan.leighton@dl-consulting.net.*

## Rules Overview

The new generation of AutoCAD Utility Design is "rules driven". A rule in AutoCAD Utility Design is essentially a programming instruction that controls some kind of behavior. Here are some examples of the things that rules within AUD can accomplish:

- Automatically add components. For example, when you draw long overhead runs, AUD rules will automatically add poles at an appropriate spacing, as well as pole heads (crossarms) and guys.
- Design "sanity checking". Rules can be set up to provide warning messages for invalid design decisions, for example, if you try to power a 3 phase transformer with a single phase feeder line
- Analysis and problem reporting. Rules can perform analysis and report on results that exceed limits, for example, to calculate the voltage drop on secondary lines downstream from a transformer and provide warnings if the calculated drop exceed predefined limits.
- Annotation. Rules can select appropriate annotation and drive annotation content.
- Material ordering. Rules can control what appears in the material order.
- Display style. For example symbols or linetypes can be selected based on rules that consider any variable or property within AUD.

## Rule categories

There are nine Categories of Utility Design Rules. When you open the Rule Configuration dialog box, you'll see these nine categories listed on the left side of the dialog box:

- Expressions: These Rules define a library of various values that are assigned to names. These act similarly to variables in traditional programming environments. All the Rule Points within this category evaluate to a value. There are four sub-categories to create Rules that evaluate to integers, real numbers, logical values, or text strings.
- Feature Identifier: These Rules define text values that represent Feature Identifiers of various kinds. By default, all AUD features are identified in Feature Info, Quick Info, Validation Results, and other locations by a combination of the Feature Class name, the Model Name, and the FID (the internal feature identifier). The Feature Identifier Rules override this Default behavior.
- Validation: These Rules are invoked by the Auto Validate feature to generate entries on the Validation Results tab.
- Material: These Rules are used to generate entries that appear in the Material Editor.
- Material Costing: These Rules are used to generate values in the Cost attribute within the Material List (the list viewable in the Material Editor). By Default, this simply maps the cost value from the active Material Catalog to the Cost attribute. Note that if you wish to define a cost based on a combination of Material Catalog factors, you should not try to assign the Cost field using the Set attribute function within the Material Rules, as it will be overridden by the Material Costing Rule even if no Rule is explicitly defined for that Feature Class!
- Annotation: These Rules control how Annotation appears for different components. This function leverages definitions within the Expressions Rules section.
- Sizing: These Rules filter the list of available Models so that only valid Models are considered when AUD is searching for an appropriately sized component. This filtering is also used to define the list of valid Model Names presented as resolution items when resolving validation problems.
- Style (new in AUD2013R2): These Rules determine which Style will be used in 2D views when a given feature is displayed.

- Analysis: These Rules largely define Analysis Variables used in equations to calculate values such as pulling tension and voltage drop.

This lab (at 75 minutes) does not attempt to provide a full, in-depth coverage of all Rules.

## Exercise 1: Basics of Rule Editing

*In this first exercise, you will walk through the basic steps to create and edit a rule.*

**Review status of two vaults that currently have the same style**

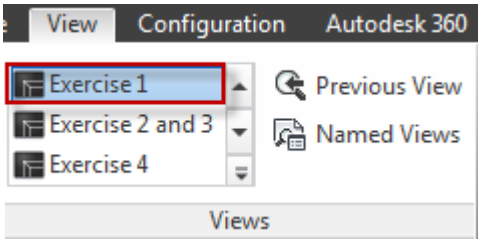
**Step 1:** Start AutoCAD Utility Design.

**Step 2:** Click on  ▶ Open, then navigate to the exercise folder and open the AUD_Rules.dwg file.

*The exercises can be found in the following folder:*
*C:\Datasets\Tuesday\UT1625-L AutoCAD Utility Design Bending the rules*

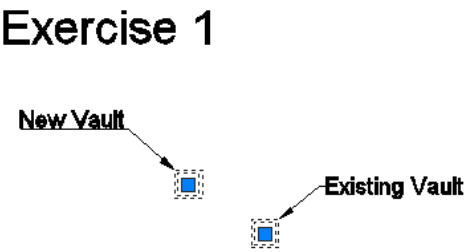
**Step 3:** Switch to the View tab.

**Step 4:** In the Views panel, click on the Exercise 1 view.

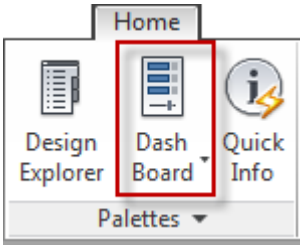



**Step 5:** The Exercise 1 portion of the drawing will appear. You will see two vaults.

**Step 6:** Select both Vaults (just the Vaults, not the annotation!)

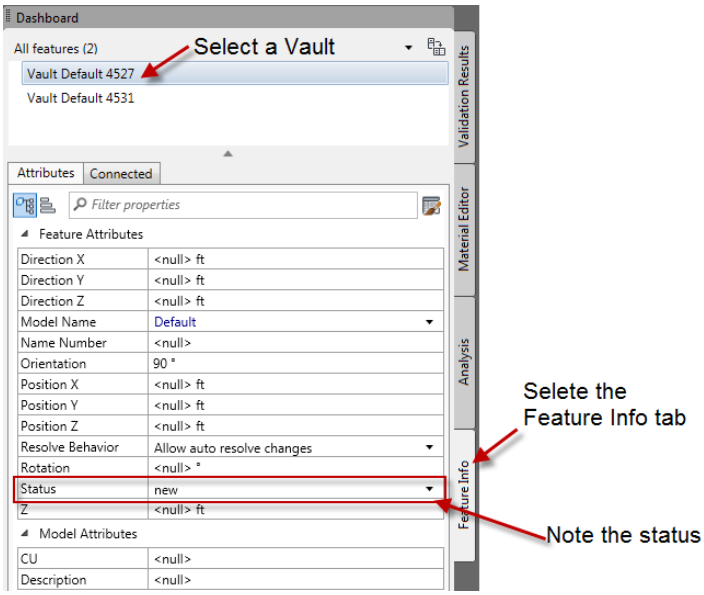



**Step 7:** Turn on the AUD dashboard (if not already visible) by clicking on the Dash Board icon within the Palettes panel of the home tab.



**Step 8:** Switch to the Feature Info tab.

**Step 9:** Select the each of the vaults in turn within the Feature Info display. Note that the first vault has a status value of "new", and the second vault has a status value of "existing".

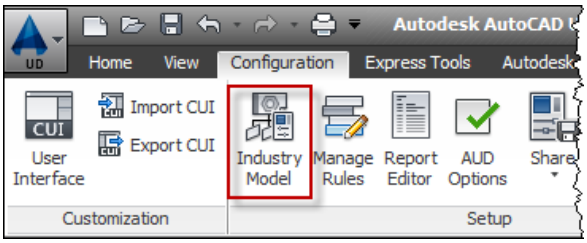



**Review the available styles**

There are at least two reasons why the both Vaults all have the same display style. There might not be any styles defined, or there might not be any styling rules. Let's first check the available styles.

**Step 10:** Switch to the configuration tab.

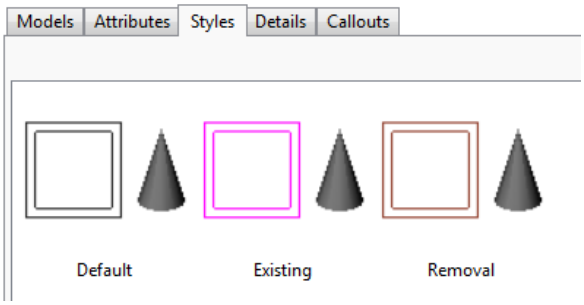
**Step 11:** Select the Industry Model command within the Setup panel




**Step 12:** Expand the section on Structures and select Vault.

**Step 13:** Switch to the Styles tab and observe that there are three styles defined.

**Step 14:** When done, click Cancel to close the Industry Model Configuration dialog box.



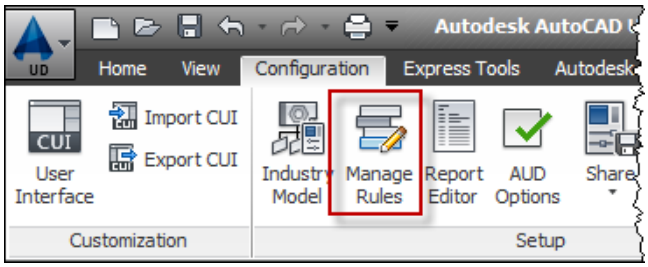

Three styles are shown, so the problem isn't that the style definitions are missing.


*Styles can be defined for any Feature Class within AUD. You can define as many styles as you want. In this case, there are three styles defined. The default is used if there are no styling rules. The other styles are selected based on the styling rules.*

**Review the current styling rules for vaults**

Let's review the current rules associated with vault styling. The Style rules control what defined style is used given different values for status and other AUD properties. Normally there are Style rules for every category of Feature Class, and every Feature Class available.

**Step 15:** Click on the Manage Rules option within the Setup panel of the Configuration tab. This will open the Rule Configuration dialog box.

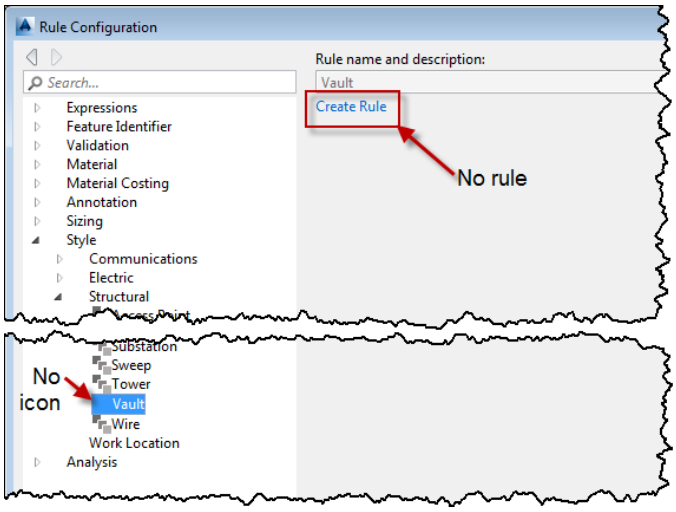



**Step 16:** Expand the Style section, and within that expand the Structural section.

**Step 17:** Note there is no icon to the left of the Vault rule.

**Step 18:** Select the Vault rule.

**Step 19:** Note that there is no rule displayed, and instead you see a Create Rule link.

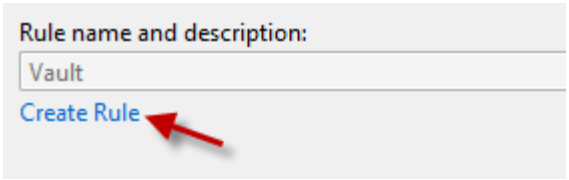



*This explains why the two vaults display the same; there is no rule to affect the styling so both vaults are displaying using the Default style. If there is no rule defined, then the default style is used in all cases.*

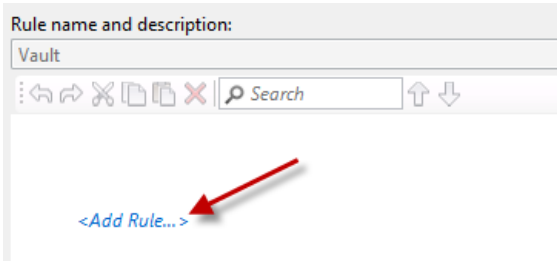
**Define a new Style Rule for vaults**

Since the Vaults have no style rule, you will create one from scratch.

**Step 20:** Click on the Create Rule link.



**Step 21:** The rule editor will now appear as shown to the right. Now click on the Add Rule link.

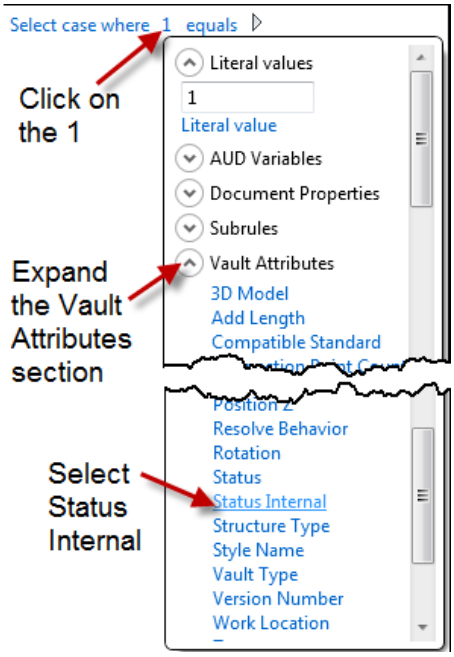



**Step 22:** Now click on the Add Rule link and choose the "Select case where rule". The beginnings of a case statement will appear.


**Step 23:** Click on the "1" (which is a temporary placeholder) within the case statement; a list of all available properties will appear.

**Step 24:** Expand the section under Vault Attributes.

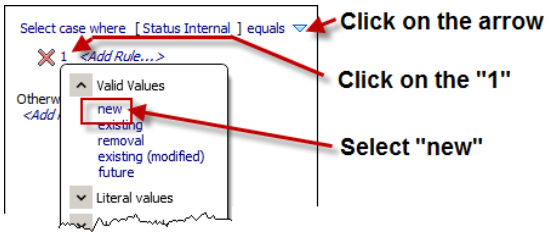
**Step 25:** Select Status Internal.




*What is the difference between "Status" and "Status Internal"? In AUD the "Status Internal" attribute specifies a feature's status from a material ordering perspective – is it new, does it already exist, or is it to be removed. The "Status" value can have more elaboration, for example if it is existing, is it in use (Status = "existing-in use") or has it been abandoned (Status = "existing-abandoned").*

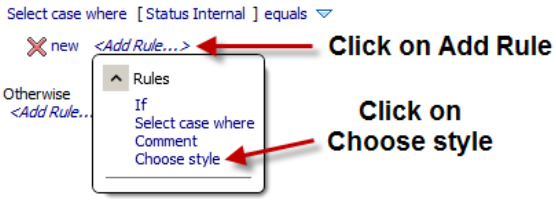
**Step 26:** Now click on the arrow to the right of equals to expand the rule.

**Step 27:** Click on the "1" (another placeholder) and this time you'll see the valid choices for comparison with the Status Internal property.




**Step 28:** Select New.

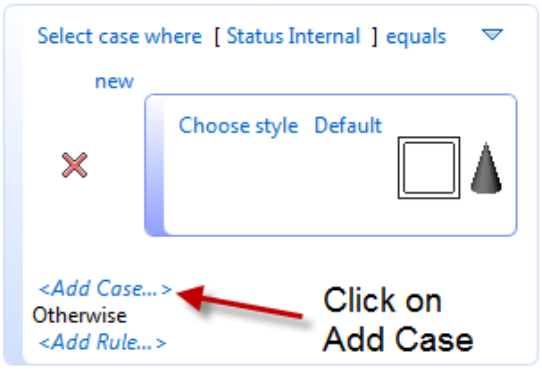
**Step 29:** Now click on the Add Rule link. This time click on "Choose style".




*Note that the Rule editor only allows you to select valid options. In this case, there are only four rules that can be used.*

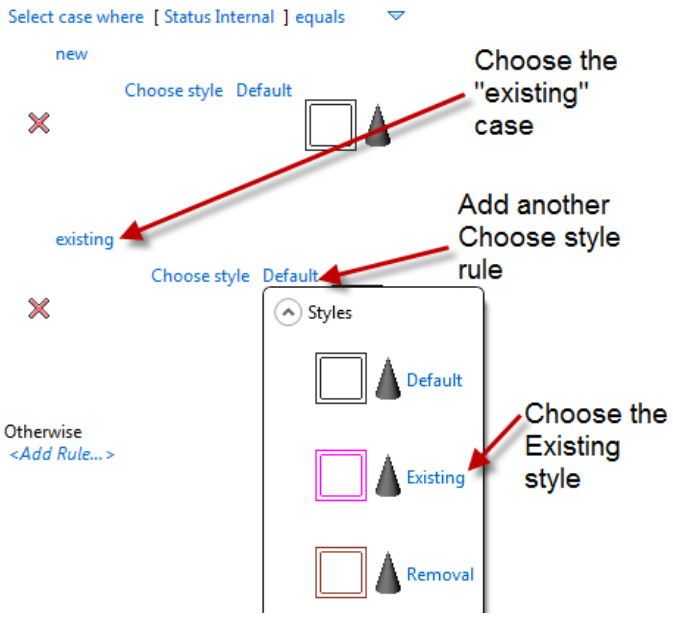
**Step 30:** The Choose style rule automatically defaults to the Default style, which is fine for new vaults.

**Step 31:** Now move the cursor down and you'll see the option to Add Case. Click on the link when it appears to add the next case.



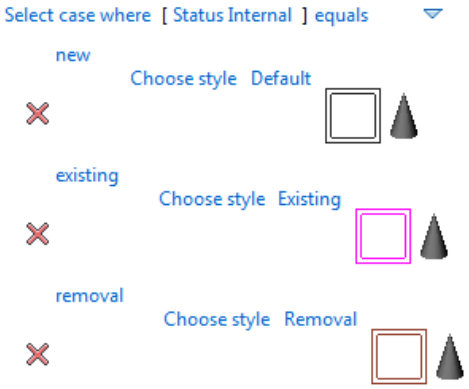
**Step 32:** Choose “existing” for the next case.

**Step 33:** Add another Choose Style rule as you did earlier. Set the style to “Existing”.

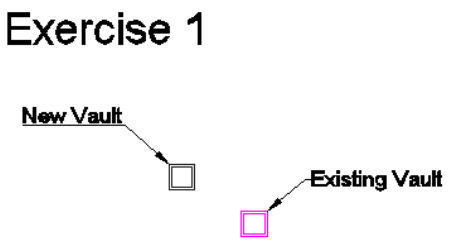



**Step 34:** Under Otherwise, add a final Choose Style rule, this time choosing the Removal Style. When you’re done, the rule should appear as shown to the right.

**Step 35:** To complete your rule editing, click on the Apply button and then click on OK.




**Step 36:** Done correctly, you should see the results shown to the right.




This completes this exercise.

## Properties used in Rules

Many of the AUD rules require properties or variables from components, the catalog, the current project document, or elsewhere. In the previous exercise, when you click on an item to set it (for example the "1" within the "if 1 equals 1" rule on the previous slide) a number of choices appeaedr. Here is a brief summary:

- **Literal values**: These are constants. They can be text strings ("Banana") or numbers ("3.14159").
- **AUD Variables**: These are variables that can be set using the AUD Options configuration command.
- **Subrules**: This provides a list of all the available subrules appropriate within the current context.
- **Object Attributes**: This is the most commonly used type of variable. Please note however that depending on the feature class, or where in the rule hierarchy (which will be explained shortly) the rule is being set, the content type will differ. For specific feature classes, you will see all of the available feature and model attributes. If you are defining a rule higher in the hierarchy, you'll see a more generic name.
- **Operators**: This provides access to all of the available operators and comparisons within the AUD rules engine. Note that if you specify an operator for a variable, you'll then be able to enter two variables, one on each side of the operator.
- **Functions**: This provides access to a large number of (mostly) mathematical functions. There are some other AUD-specific functions as well, for example "attribute is not set", "feature class caption", and things like "number of connected".
- **Catalog**: This provides access to the material catalog, which is useful when creating material ordering rules that need to add a specific part.

## Advanced Style Rules

In Exercise 1 you created a basic styling rule. Styling rules determine which of the styles defined within the Industry Model configuration is selected, given various component properties (or other AUD properties). Here are some points to consider:

- Styles must first be defined using the Industry Model editor.
- Styles must be defined for *every* component display variation (blocks, linetypes, colors, attributes, etc.).
- Case statements are usually used to determine which style to choose.

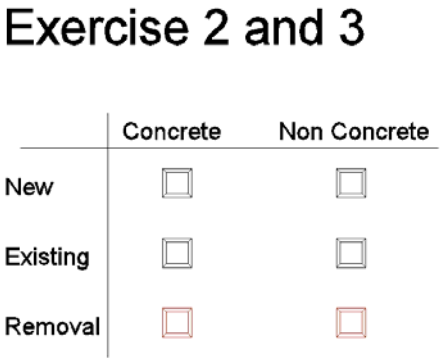
While most styling rules are based on the state of components, there are some that are more complex. In most implementations, the rules for conductors will be the most complex, with rules to select linestyles based on number of status, number of phases, wire use, and more.

This next exercise walks through the steps to create an advanced styling rule.

## Exercise 2: Create an advanced styling rule

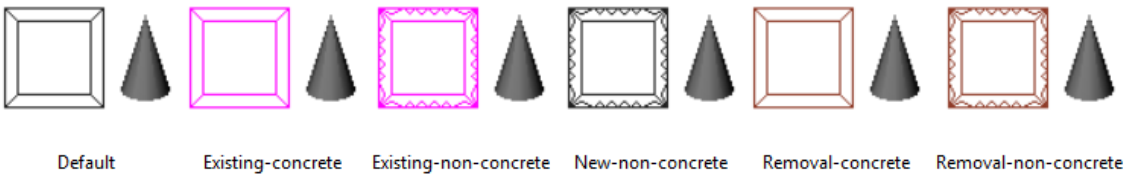
*In this exercise, you'll create an advanced style rule for transformer pads that will consider both Status and Pad Type.*

**Step 1:** In the AUD_Rules.dwg file, switch to the Exercise 2 and 3 view.



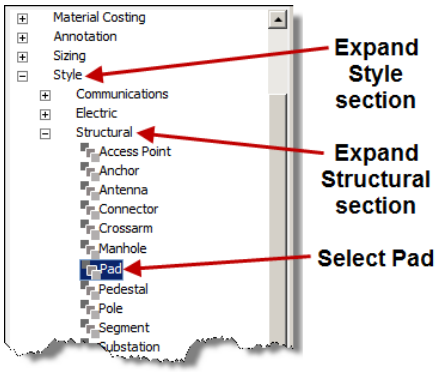

**Step 2:** The drawing contains six pads, side by side. Note the annotation indicates that there are two types of pads, and 3 different status values. The poles with model names ending in "C" are concrete, the "NC" pads are not concrete. They are all displaying with the same symbol and color.

**Step 3:** As you did in Lesson 1, issue the Industry Model configuration command and view the Style tab for pads. You can see there are six styles defined as shown below.

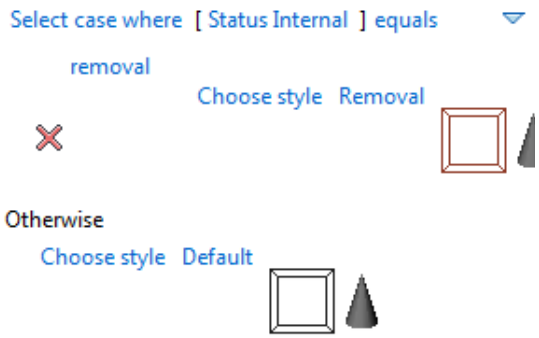



**Step 4:** Click on OK to leave the Industry Model Configuration editor.

**Step 5:** Now click on the Manage Rules icon to return to the Rule Configuration screen. Locate the Style Rule for Pads (Under Style/Structural).



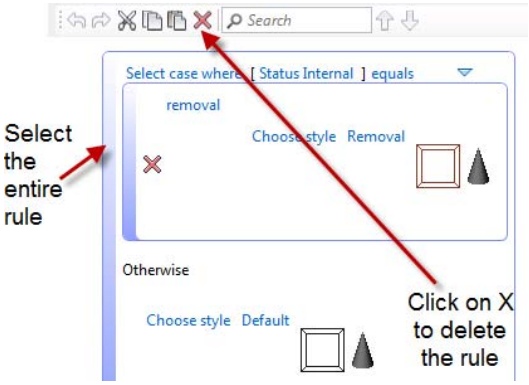
**Step 6:** As you can see, this rule is incomplete.




### Delete the existing styling rules

In this case it is easier to start over and define the full set of rules correctly.

**Step 7:** Delete the existing rules. You can do this by selecting the rules and clicking on the red X at the top.




### Set up a complex IF statement

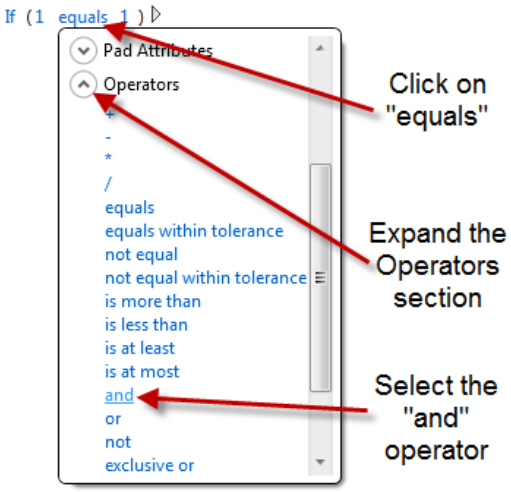
For this exercise, you will create an IF statement with two parts that both must be true. Follow the steps carefully as this can be a bit tricky.

**Step 8:** Start by creating an If rule, which will initially say If (1 = 1):

- Click on <Add Rule…>.
- Select If.

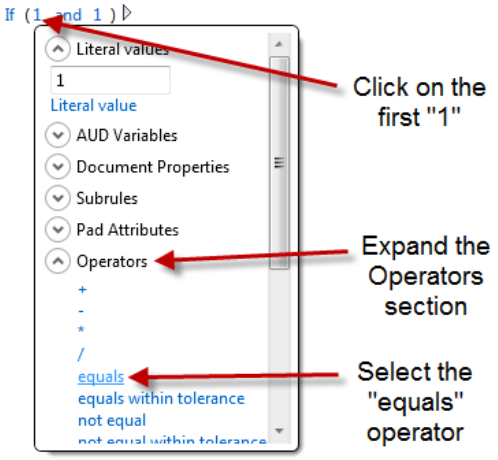
**Step 9:** Now switch to using the “and” operator:

- Click on the “equals” link.
- Expand the Operators section.
- Click select the “and” operator.



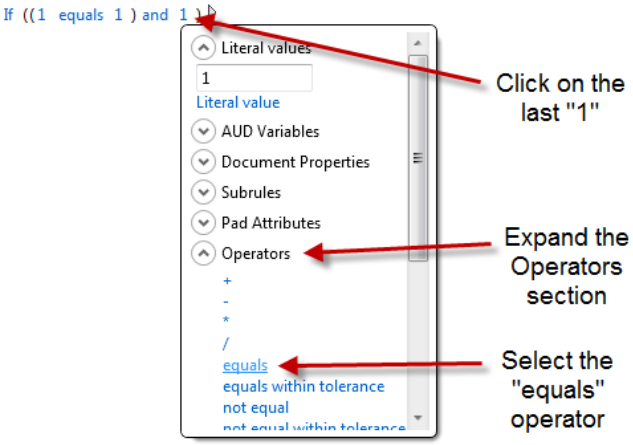

**Step 10:** Next, change the left half to an “equals” clause:

- Click on the leftmost “1” within the rule as shown to the right.
- Expand the Operators section.
- Select the “equals” operator.

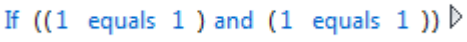



**Step 11:** Do the same thing for the right half, to create another “equals” clause:

- Click on the rightmost “1” within the rule as shown to the right.
- As you did before, expand the Operators section and select the “equals” operator.

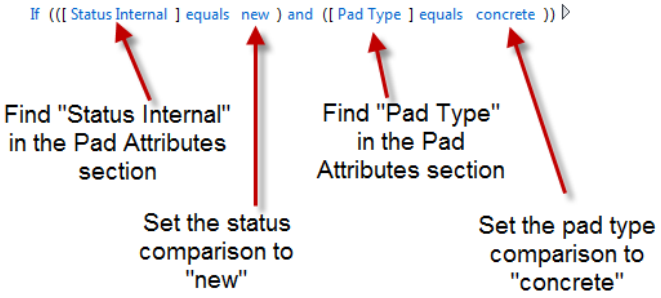



The rule should now appear as shown on the right.



**Step 12:** Now set the values for the left side of the complex IF statement:

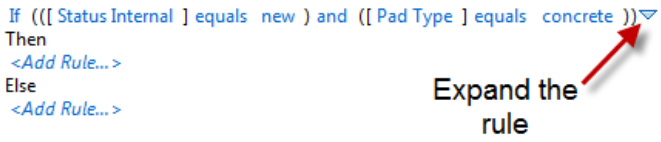
- Click on the first "1"
- Within the Pad Attributes section select the "Status Internal" attribute.
- Set the comparison for the status to "new".




**Step 13:** And continue by setting the values for the right side of the IF statement:

- In the second clause set the first value to Pad Type (also found within the Pad Attributes section).
- Set the comparison for the pad type to "concrete".

**Step 14:** Now expand the rule to show the Then and Else clauses.



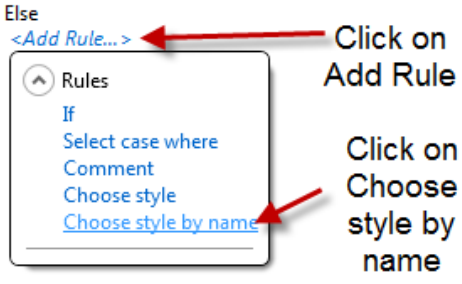

**Step 15:** Click on <Add Rule…> within the Then clause, select the Choose Style rule, and set it to the Default style.

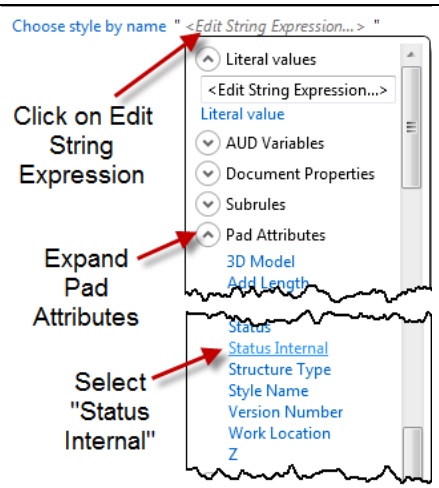
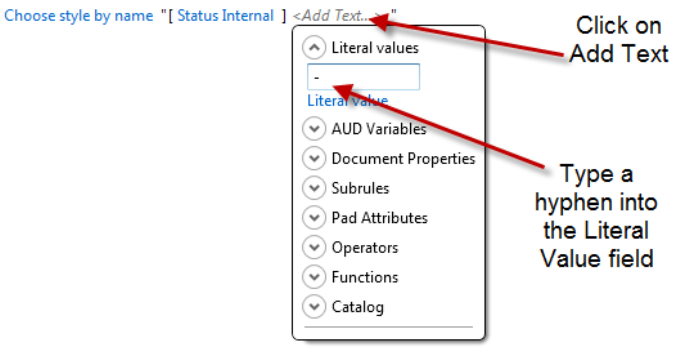
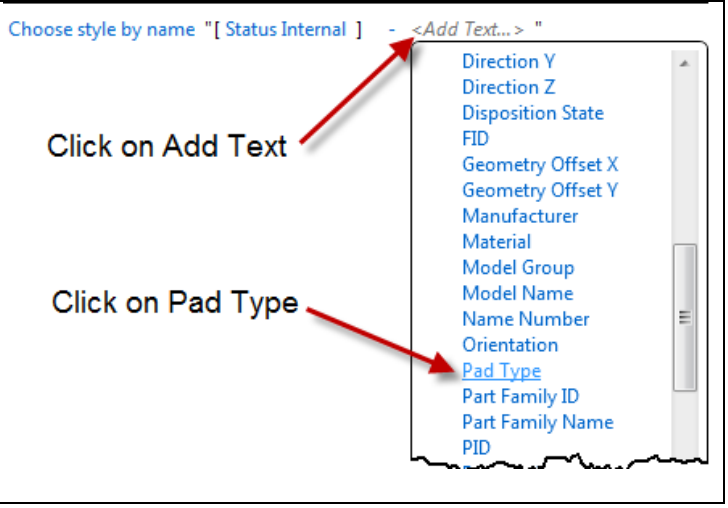

**Create a data-driven styling rule**

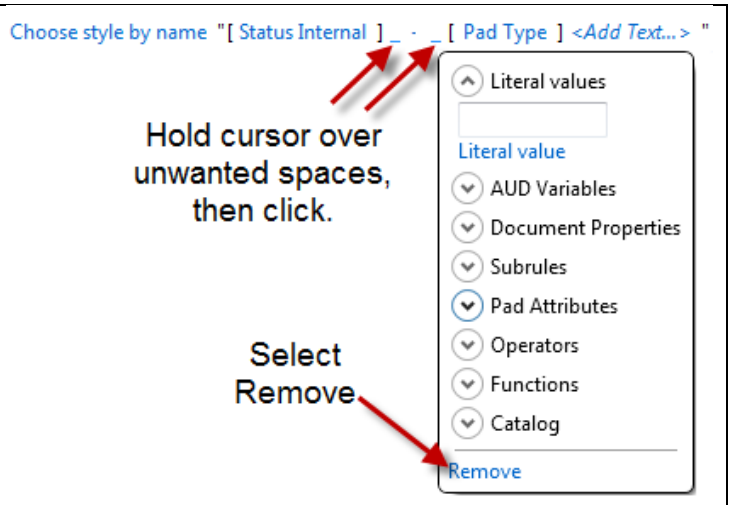
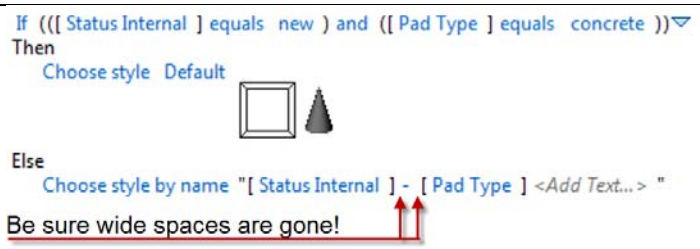
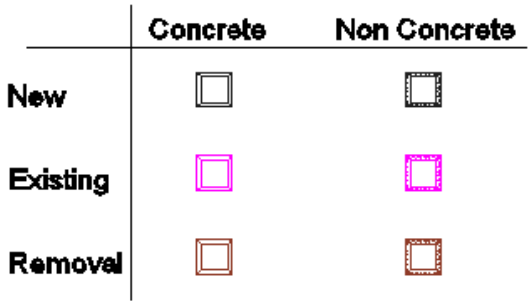
The complex if statement just created was required because new concrete pads use the default style. Now you will use the new "Choose style by name" rule for the remaining styles.

**Step 16:** Click on the "Add Rule" link within the Else clause.

**Step 17:** Select the "Choose style by name" rule.



| | |
|---|---|
| **Step 18:** Click on Edit String Expression.<br><br>**Step 19:** Expand the Pad Attributes section, and choose "Status Internal". |  |
| **Step 20:** Click on Add Text.<br><br>**Step 21:** Type a hyphen into the Literal Value field (press Enter to accept the value). The hyphen will appear in the expression. |  |
| **Step 22:** Click again on Add Text.<br><br>**Step 23:** Within the Pad Attributes section, choose Pad Type. |  |

| | |
|---|---|
| **Step 24:** Finally, locate and clean out any unwanted spaces in the Chose style by name command (these are sometimes added automatically):<br>▪ Hold the cursor over any wide spaces. A small underline will appear indicating an unwanted space insertion.<br>▪ When the underline appears, click, then select remove. |  |
| **Step 25:** The complete rule should now appear as shown on the right. Check to be sure the wide spaces are gone.<br>**Step 26:** Click on Apply, then click on OK to close the Rule editor. |  |
| **Step 27:** Done correctly, you should see the results shown to the right. |  |

This completes this exercise.

## Material ordering rules

Material Rules execute when a design is performed, and they define what appears in the Material List based on components used in the design. These Rules are essential to getting the full value out of AutoCAD Utility Design. There are two basic things these rules do:

- Determine what CU (or combination of CUs) in the Material Catalog should be ordered (for components with Internal Status = “New”) or removed (for components with where the Internal Status = “Removal”).

- Determine value for additional attributes associated with the objects, for example to calculate associated costs.

The heart of material ordering is the "Add material" command as shown in the following figure.


Material rules must consider each component's status, and despite its name, the Add Material Rule can be used for both adding and removing material. Think of it as adding an entry to the Material List.

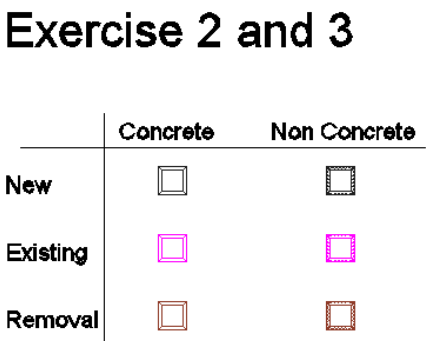
## Exercise 3: Configuring a Material subrule

*In this exercise, you'll add a fairly comprehensive material subrule and use that rule within the material hierarchy.*

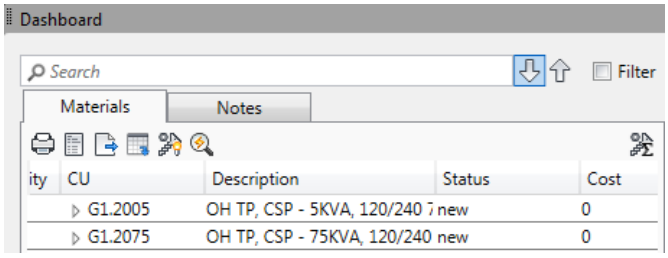
**Review current Pad rule**

As you'll see, there are currently no materials in the material editor for pads.

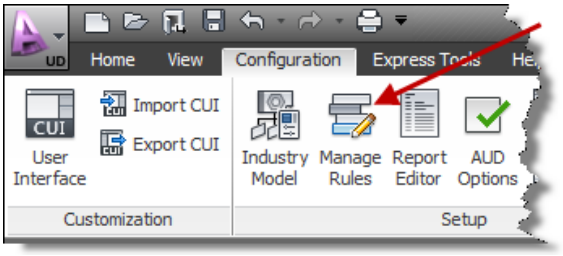
**Step 1:** In the AUD_Rules.dwg file, switch to the Exercise 2 and 3 view.




**Step 2:** Switch to the Materials tab of the dashboard. Note there are only two items listed, and there are currently no pads listed.

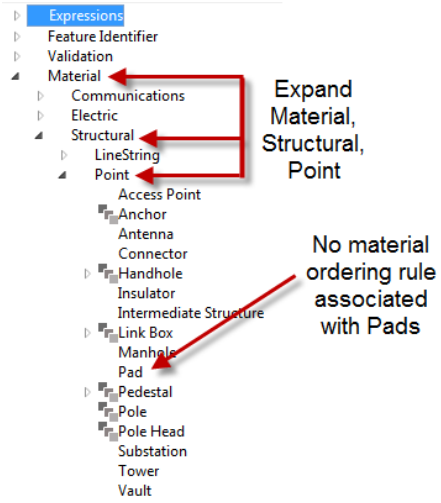


**Step 3:** On the Configuration tab, click on the Manage Rules command within the Setup panel.




**Step 4:** Expand the Material section, then expand the Structural section and the Point section.

**Step 5:** Note there is no rule associated with Pads.

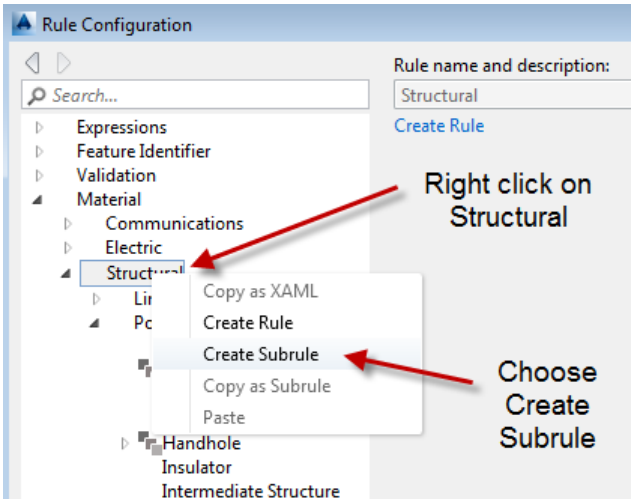



*Rather than creating a rule for pads, you will create a more general purpose Subrule and then use it for the pads.*

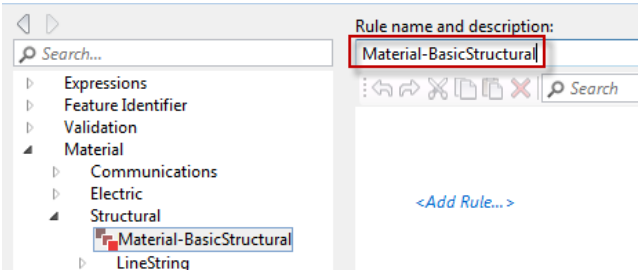
**Add a subrule at the "Structure" level**

Subrules placed higher in the rule hierarchy can be used anywhere underneath.

**Step 6:** Right click on Structure and choose Create Subrule.



**Step 7:** Enter "Material-Basic Structural" for the Rule name in the field near the top of the screen.



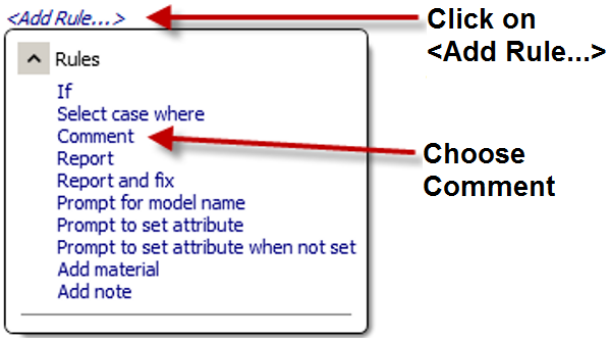

*When you define names for subrules, have a naming strategy so you will know the purpose of each Rule later when you need to choose it from a potentially long list of subrules.*

**Add a comment to help describe the subrule**

Utility Design now has the ability to include comments in Rules to better describe their function. You'll use this ability here to describe this basic electric Rule.

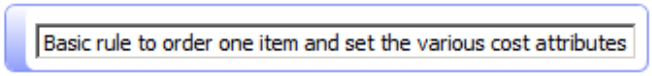
**Step 8:** Click on <Add Rule...> in the Rule definition area.

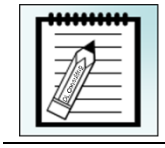
**Step 9:** Choose Comment.




**Step 10:** Enter a description for the subrule as shown on the right.

**Step 11:** Press Enter to finish entering the comment.



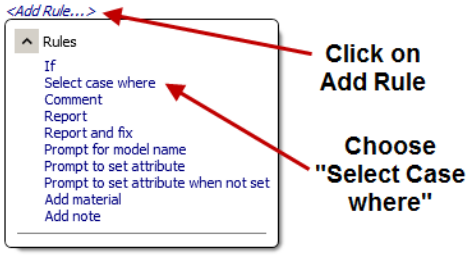
*When you have dozens of comprehensive Rules and subrules, they can become quite confusing. It is good practice to include a comment for all the Rules and subrules you create within Utility Design.*

### Define a case statement in the subrule

Most material ordering Rules start by considering the Status of the component, because what you order and how you calculate it is based on Status. You'll use a case statement for this purpose.

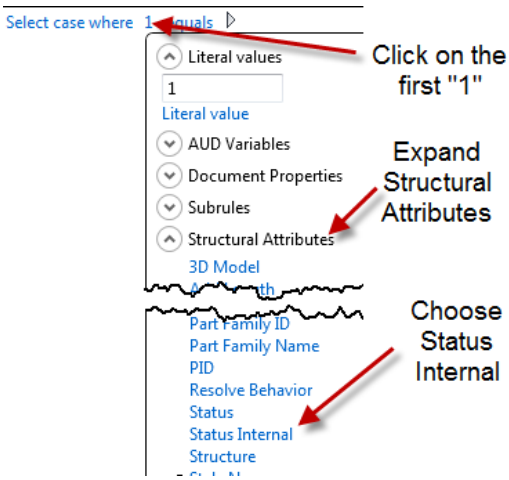
**Step 12:** Add a new case statement:

- Hold the cursor below your new comment until <Add Rule…> appears.
- Click on <Add Rule...> in the Rule definition area.
- Choose the "Select case where" rule.




**Step 13:** Select "Status Internal" as the case statement variable:

- Click on the "1" within the case statement.
- Expand the Structural Attributes section.
- Choose "Status Internal".



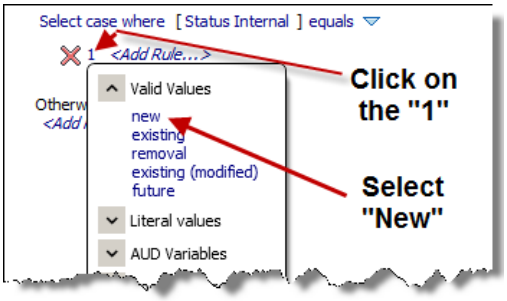

*Note that just below "Subrules" you see "Structural Attributes." If you had selected a specific component type, for example a pole, this would have said "Pole Attributes." Since you are adding this subrule high in the Feature Class hierarchy, you will only have feature attributes that are* ***common to all features below it*** *in the hierarchy.*

### Define the case where Status = New

Now you will define what happens in the case where the value of Status Internal is equal to "new."

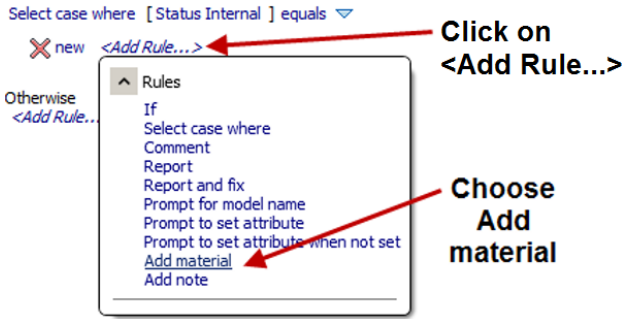
**Step 14:** Click on the arrow to the right to expand the Case statement.

**Step 15:** Click on the "1", and change it to "new".

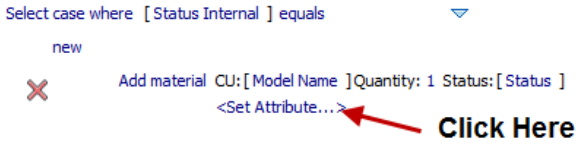


**Step 16:** Add the rule associated with cases where the internal status equals "new":

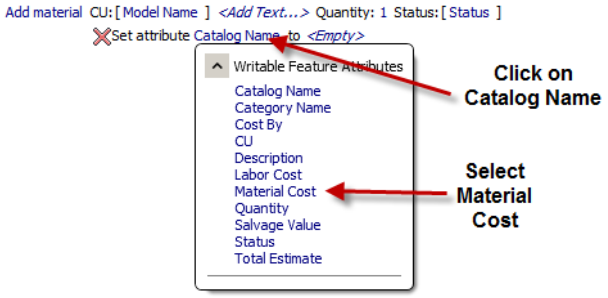
- Click on <Add Rule…> to the right of "new".
- Choose "Add material." The generic material ordering Rule will appear (which in this example is acceptable as-is).




**Step 17:** Click on the <Set Attribute…> clause below the Add Material Rule.



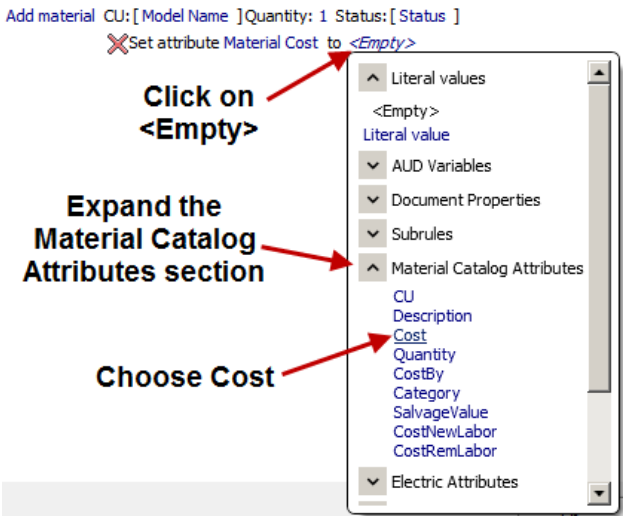

**Step 18:** The attribute will initially be set to Catalog Name (the first one alphabetically in the list). Click on Catalog Name and instead choose Material Cost.




**Step 19:** Now click on the second clause in the Set attribute statement, <Empty>. You'll see the usual list of choices.

**Step 20:** Expand the Material Catalog Attributes section.

**Step 21:** Choose Cost.

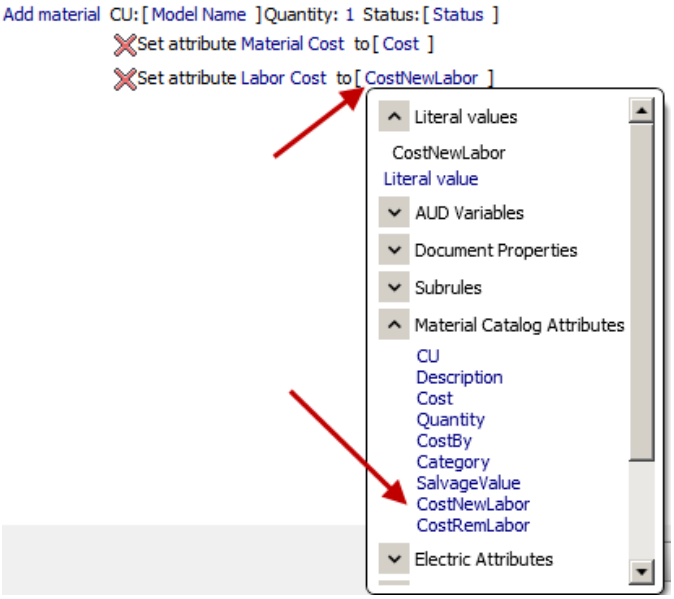


***What are you doing here?***

*The Set attribute clause is used to define the value of an attribute in the Material List (i.e. the list that appears in the Material Editor). In this case, you want to populate the material cost value. You want to assign the material cost based on the cost value in the Material Catalog, which happens to be named "Cost."*

**Step 22:** Now, following similar steps to what you just did, add a second Set attribute clause, this time to set the Labor Cost attribute to CostNewLabor, another custom field in the catalog.

**Step 23:** Click on Apply to accept your subrule.




**Define the rest of the subrule (OPTIONAL)**

If you have time in the class, you can define the rest of the subrule (although it's not necessary to continue with the class!)

**Step 24:** Now define the rest of the subrule as shown to the right.

**Step 25:** When you're done, click on Apply.

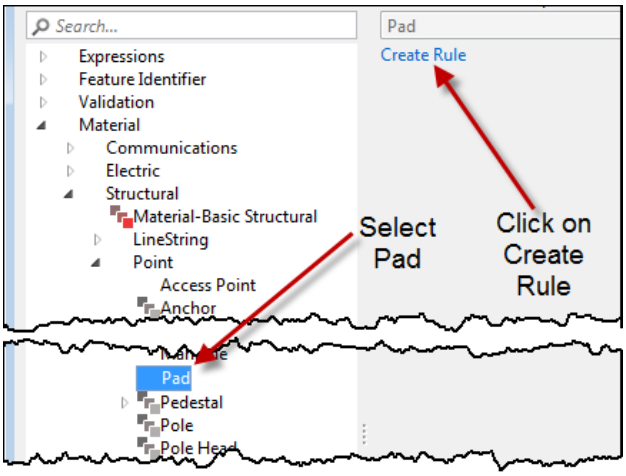
```
Select case where [Status Internal ] equals
    new
        Add material CU:[Model Name ] <Add Text...> Quantity: 1 Status:[Status ]
                Set attribute Material Cost to[Cost ]
                Set attribute Labor Cost to[CostNewLabor ]
                Set attribute Salvage Value to 0
    removal
        Add material CU:[Model Name ] <Add Text...> Quantity: 1 Status:[Status ]
                Set attribute Material Cost to 0
                Set attribute Labor Cost to[CostRemLabor ]
                Set attribute Salvage Value to[SalvageValue ]
    existing (modified)
        Add material CU:[Model Name ] <Add Text...> Quantity: 1 Status:[Status ]
                Set attribute Material Cost to([Cost ] - [SalvageValue ])
                Set attribute Labor Cost to([CostNewLabor ] + [CostRemLabor ])
```

### Use the subrule to define material ordering for all pads

Now that the subrule is complete, you can use it to define material ordering for any component within the structural hierarchy. In this case you'll define the rule for pads.

**Step 26:** Click on Pad (under Structural Point within Material hierarchy.

**Step 27:** Click on Create Rule to create the Pad material ordering rule.

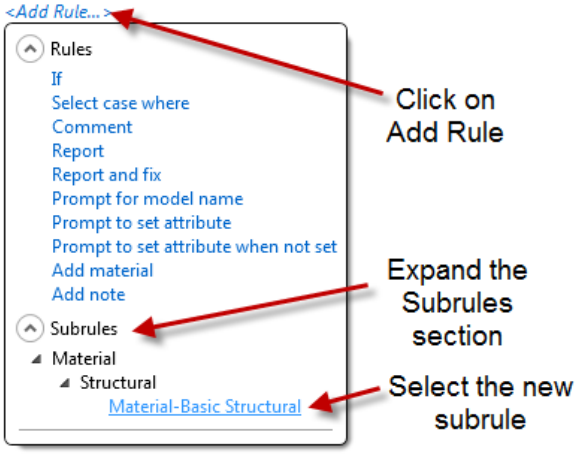



**Step 28:** Now click on <Add Rule...>

**Step 29:** Expand the Subrules section.

**Step 30:** Select the new subrule created earlier in this exercise.

**Step 31:** Click on Apply. A Rule icon will appear next to Pad.



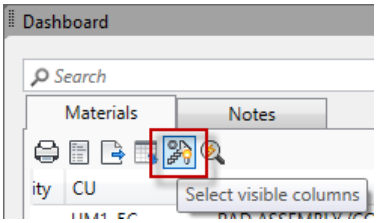

**Step 32:** Once you're done, click on Apply and then OK to save the rule and leave the AUD Rule editor.

### Note the results of the new rule

Now you will observe the effect within the Material Editor.

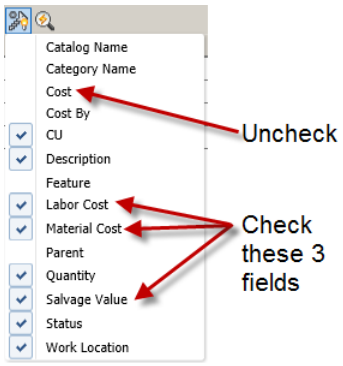
**Step 33:** Be sure that the Material Editor tab on the dashboard is selected.

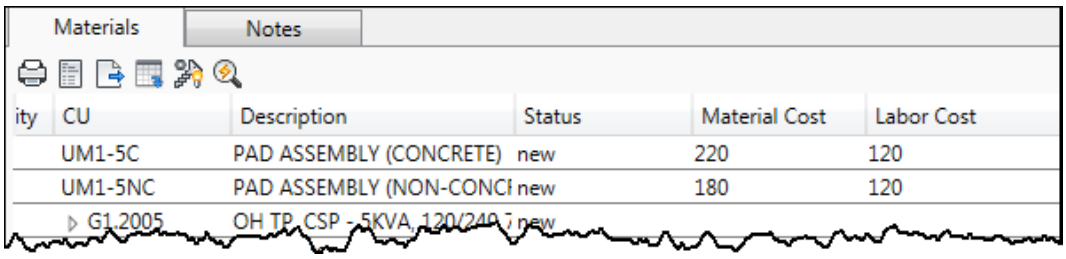
**Step 34:** Click on the icon that looks like a screw with a light bulb.



**Step 35:** In the Select Visible Columns display, uncheck Cost and check Material Cost, Labor Cost, and Salvage Value.

**Step 36:** Note in the Material Editor there will now be entries for pads. If you did the optional step, you'll also see removal entries for the pads listed to be removed.




This completes this exercise.

## Expression rules

The concept of Expressions Rules is similar to function definitions within a programming environment. These Rules let you define a Rule name that acts like a variable within AUD. There are four Expression Rule types corresponding to four types of variables: Booleans (logical true or false), Integers, Real numbers, or Text.

You can define these "variables" for use in other parts of the configuration, including:

- The Visibility control within Industry Model Attribute definitions. An attribute can be visible or not based on a Boolean Expressions Rule using some combination of the object's attributes.
- The block attribute settings within point object Styling. These attributes can be driven by a text Expressions Rule based on a combination of the object's attributes.
- Within other Rules. This is very common, as it's often easier or more practical to refer to a short, understandable expression instead of defining a complex Rule over and over.

There are a number of other places within AUD where you will find that Expressions can be used.

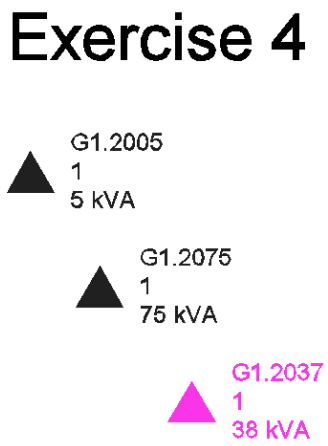
## Exercise 4: Configuring and using a Text Expression rule

*In this exercise, you'll add a simple Text Expression rule to better display a transformer's phase count.*

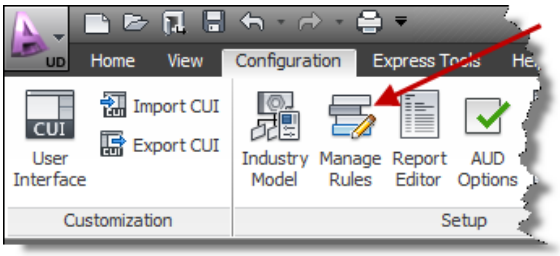
### Define new Text Expression

**Step 1:** In the AUD_Rules.dwg file, switch to the Exercise 4 view.

**Step 2:** Note there are three transformers, each with attributes displaying some transformer attributes.

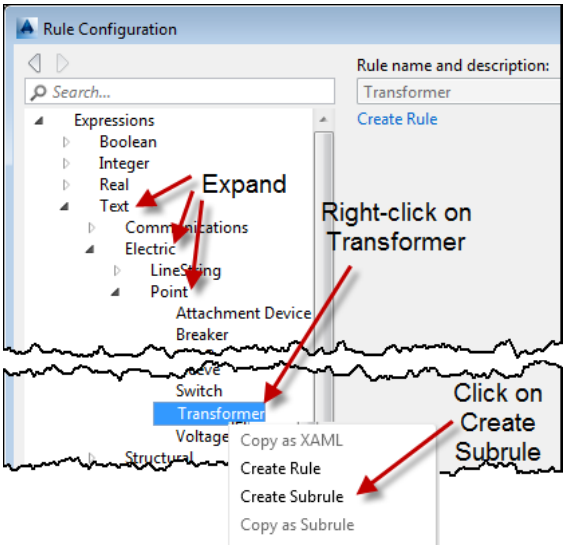



**Step 3:** On the Configuration tab, click on the Manage Rules command within the Setup panel.

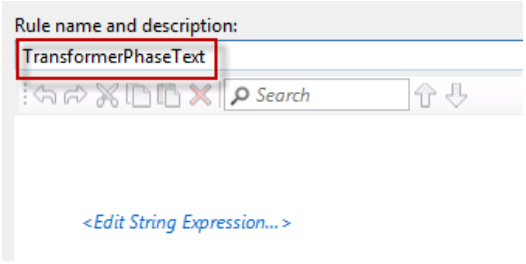



**Step 4:** Begin the steps to create a Transformer subrule as follows:

- Expand the Expressions section.
- Further expand the section under Text, Electric, Point.
- Right-click on Transformer.
- In the context menu, click on Create Subrule.



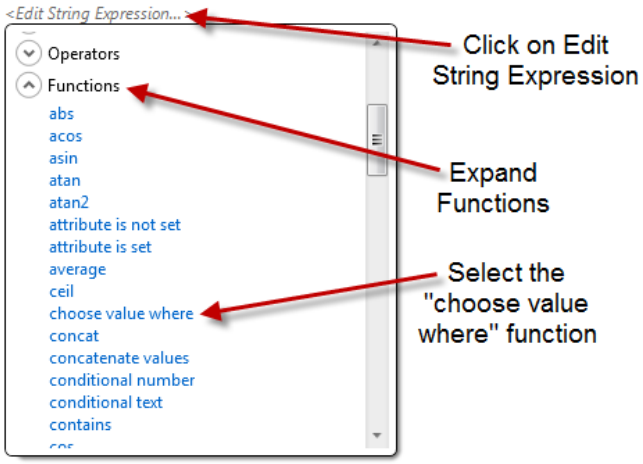
**Step 5:** Near the top, enter the name "TransformerPhaseText".




*Expressions usually have only a single line that evaluates to a single value. In this case, you'll want a logic statement, so to do this you will use a special function.*

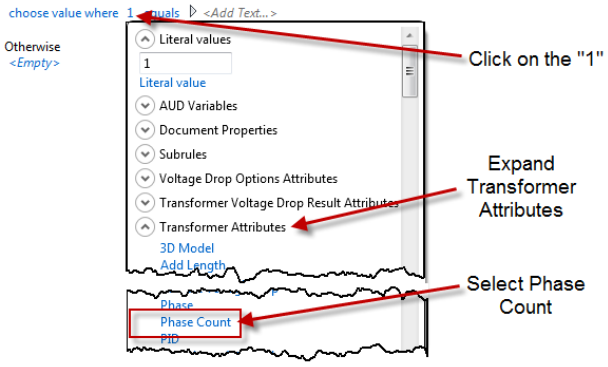
**Step 6:** Select a special function as follows:

- Click on Edit String Expression…
- Expand the section on Functions.
- Select the "choose value where" function.



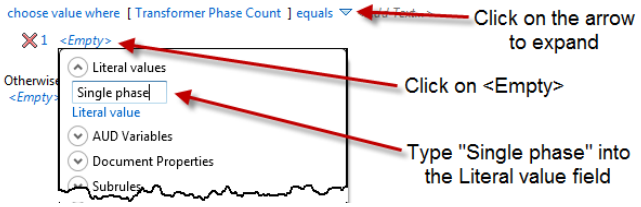

**Step 7:** Set the value to evaluate:

- Click on the "1" in the first line.
- Expand the section with Transformer Attributes.
- Select Phase Count.



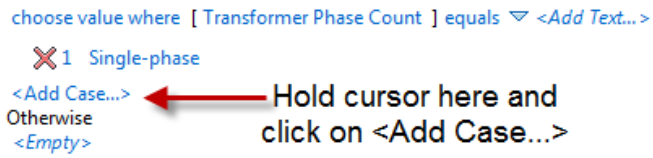
**Step 8:** Now set a value for the first case:

- Click on the arrow to the right of the expression to expand the display.
- Note in this case, for single phase transformers, the "1" value is already correct.
- Click on <Empty>.
- Type "Single phase" into the literal value field and press Enter.

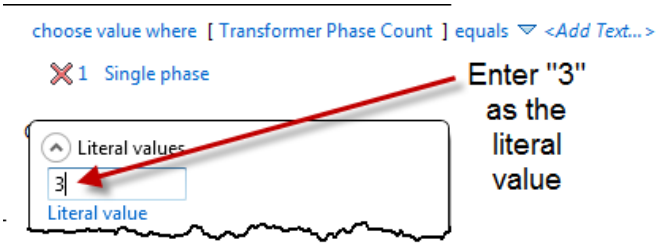



**Step 9:** Add another case as follows:

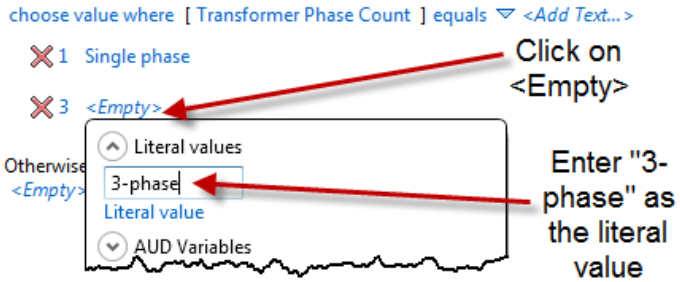
- Hold the cursor just below the 1 Single phase entry.
- When <Add Case…> appears, click on it.




**Step 10:** Enter "3" into the literal value field. Press Enter to accept the value you type.

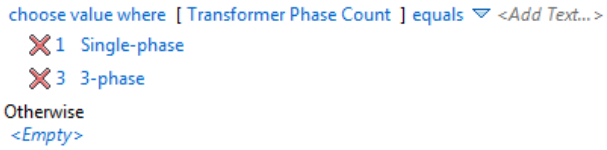



**Step 11:** Set the value associated with 3 phase transformers to "3-phase" as shown.




**Step 12:** The final rule should appear as shown to the right.

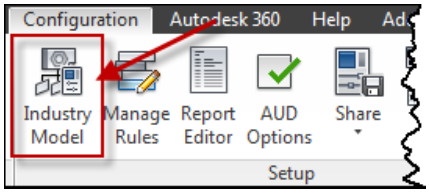
Click on Apply, then click on OK to complete creating the rule.



**Modify the transformer style**

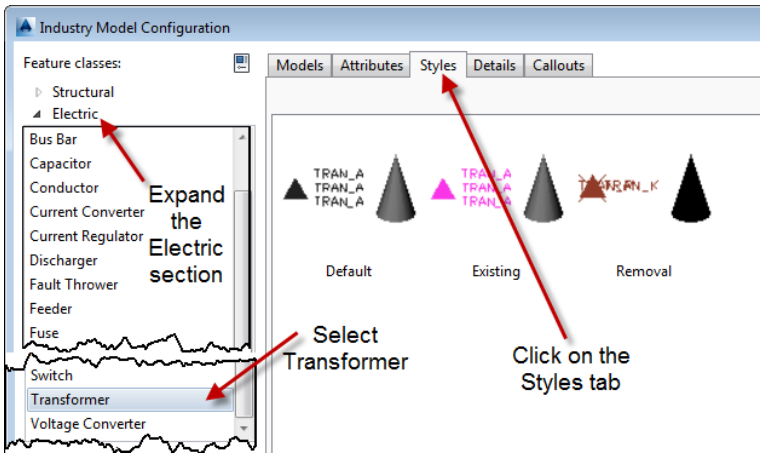
Next you will update the transformer's style so that the display includes the value provided by this expression.

**Step 13:** On the Configuration tab, click on the Industry Model icon.



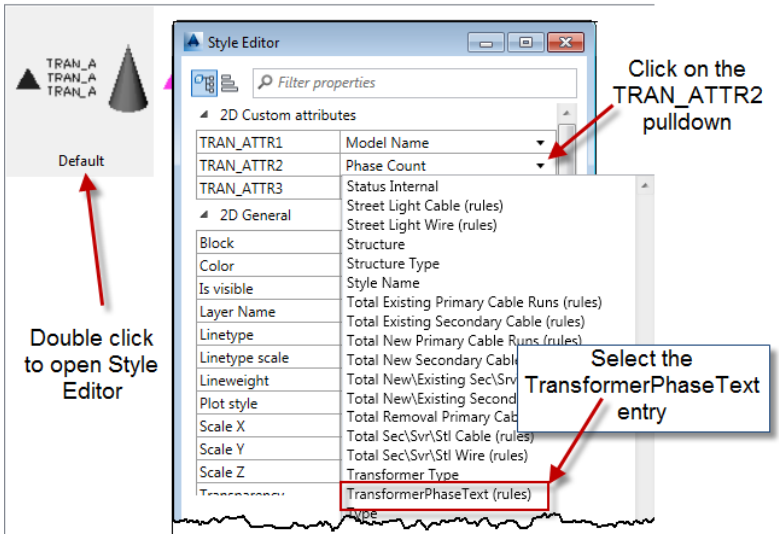

**Step 14:** Display the transformer styles as follows:

- Expand the Electric feature class section on the left.
- Select the Transformer feature class.
- Select the Styles tab.



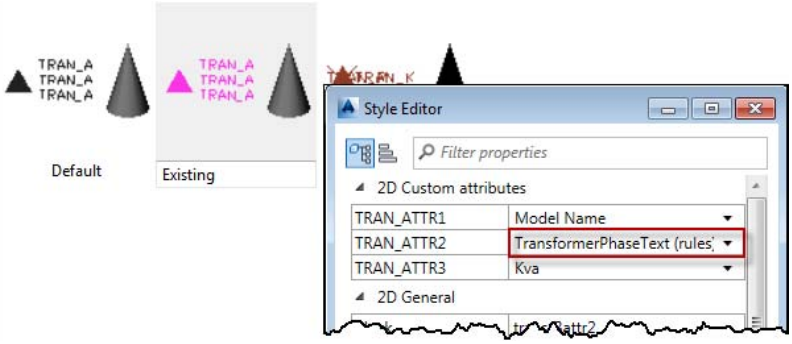

**Step 15:** Update the attribute display for new transformers (the Default entry):

- Double click on the Default entry.
- In the Style Editor, click on the pull down for TRAN_ATTR2.
- Select the new TransformerPhaseText expression created earlier.
- Click on OK to accept the change.



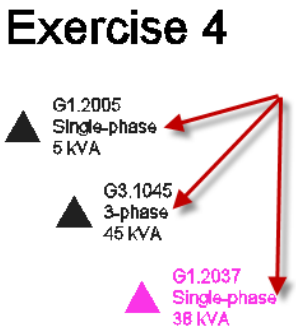
**Step 16:** Make a similar change to the Existing style as shown to the right.

**Step 17:** Click on Apply, then click on OK to accept the changes to the Industry Model configuration.




**Review results**

**Step 18:** The drawing should be changed as shown to the right.




## Annotation rules

The Annotation Rules are associated with selecting Callouts. These Rules control the selection of which Callout definition to use for objects within a Feature Class. Note these Rules can also be placed higher in the Rule hierarchy if appropriate for your implementation. These Rules are fairly simple:

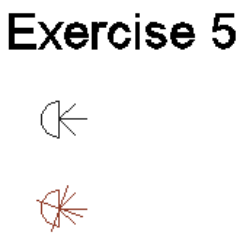
- The user issues the AUDCALLOUTINSERT command. As part of this, the user selects one or more objects.
- AUD checks to see if there's a Rule defined that applies to the Feature Class(s) of selected object(s)
- If a Rule does exist, the Rule is evaluated, and a Callout is inserted based on the results of the Rule.
- If a Rule does not exist, the user is prompted to choose one of the general Callouts, which are presented to the user in the Gallery. Note that this prompting only occurs if a single object is selected.

## Exercise 5: Implement and use a Callout

*In this exercise, you will create definitions for two Callouts for lights, then add Annotation Rules to use these Callouts and test the result.*

**Step 1:** In the AUD_Rules.dwg file, switch to the Exercise 5 view.

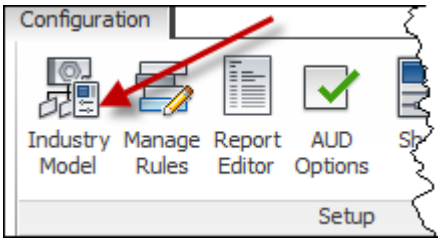
Note there are two lights – one new (black) and one removal (brown). There are no attributes or annotations.




### Create a Callout for new Lights

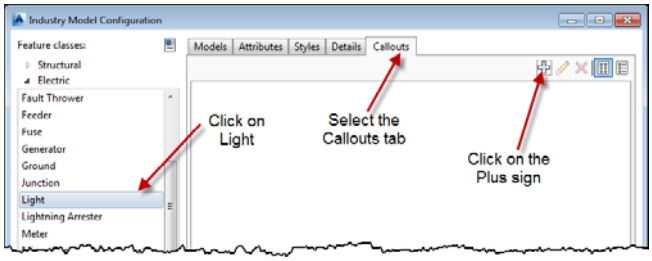
This exercise will use an existing callout block, but will repurpose it for the lights.

**Step 2:** On the Configuration tab, click on the Industry Model command within the Setup panel



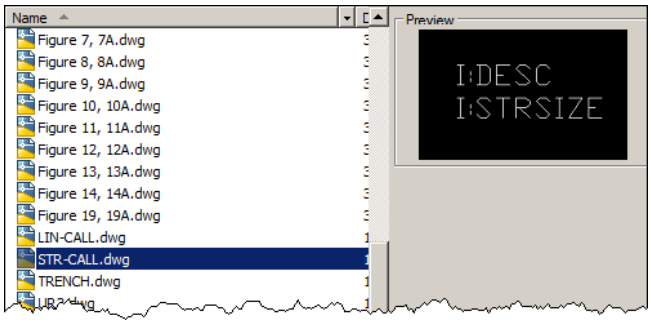

**Step 3:** Create a new callout definition for Lights as follows:

- Expand the Electric section.
- Click on Light within the Electrical section.
- Select the Callouts tab. Note that no Callouts are currently defined.
- Click on the Plus sign to create a new Callout.




**Step 4:** Select the STR-CALL.dwg file to use as the callout block:

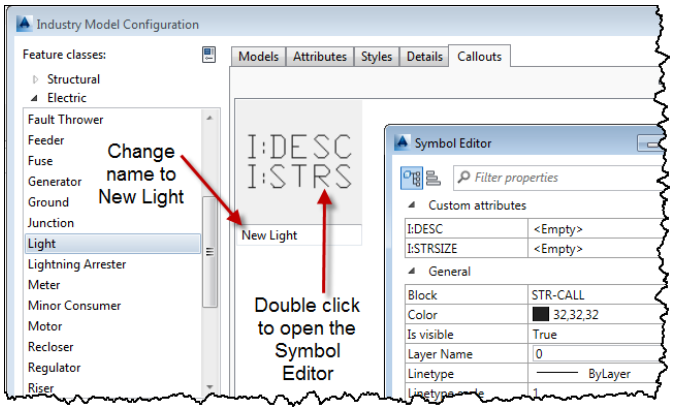
- From the list of DWG scroll down and select the STR-CALL.dwg file.
- Click on Open.



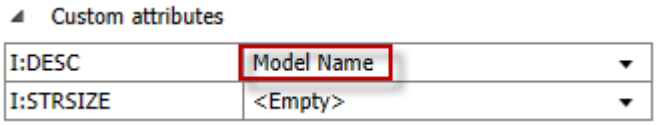
*Although the preview shows two fields for a description and a structure size, these can be repurposed for other uses.*

**Step 5:** Click on the name below the new Callout and change it to "New Light."

**Step 6:** Double click on the new Callout that appears. The Symbol Editor will display.

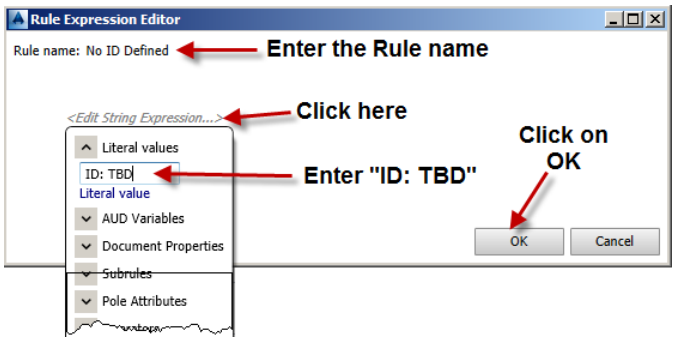



**Step 7:** Click in the first custom attribute field (I:DESC), and select Model Name.




**Step 8:** Create a new expression for the second field as follows:

- Click in the second attribute field (I:STRSIZE) and select <New Expression…>. The Rule Expression Editor will appear.
- Enter a new Rule name: "No ID Defined."
- Click on <Edit String Expression…>.
- In the text field under Literal Values, type in "ID: TBD" and press Enter.
- Click on OK.




**Step 9:** Back in the Symbol Editor, click on OK.

*What you've actually done is define an Expression subrule under the Text category.*

### Create a second Callout for lights that aren't new

Now repeat the steps to create a Callout for lights where the ID is known.

**Step 10:** Follow the earlier steps to create a second Callout definition:

- Click on the Plus sign near the upper right.
- This time select the LIN-CALL.dwg file. This one has three lines.
- Give the new Callout the name "Existing/Removal Light".


**Step 11:** Now set the attributes for the new callout:

- Double click on the new Existing/Removal Light Callout to open the Style Editor.
- Set I:DESC1 to Model Name.
- Set I:DESC2 to Name Number.
- Set I:WIRELEN to Owner.

| | |
|---|---|
| I:DESC1 | Model Name |
| I:DESC2 | Name Number |
| I:WIRELEN | Owner |

**Step 12:** Click on OK, accept the changes, and close the Industry Model Configuration dialog box.

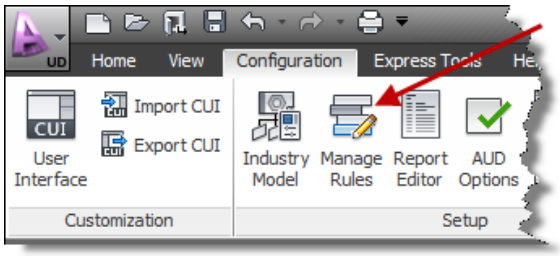
***What are you doing here?***

*When lights are first specified, the correct ID number is not known – this is not established until the construction crew installs the light. So for new lights, the ID field will say ID: TBD. For other lights (existing and removal), the ID field will instead display the contents of the Name Number field, which is AUD's field for unique, component-specific identifiers. In addition, by using the three line block, we can display a different number of attributes for existing and removal lights.*

### Define an Annotation Rule for Lights

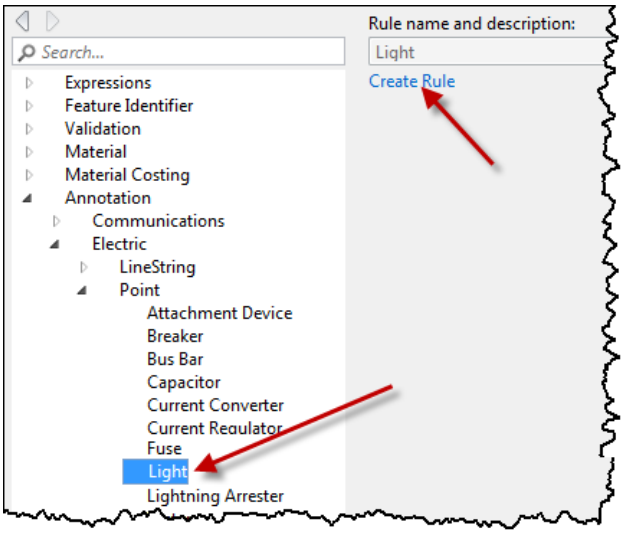
Now you define an actual Annotation Rule to choose the appropriate Callout based on Light Status.

**Step 13:** Click on the Manage Rules command in the Setup panel on the Configuration tab.



**Step 14:** Expand the Annotation Rules under Electric and Point, and select Light.

**Step 15:** Click on Create Rule.



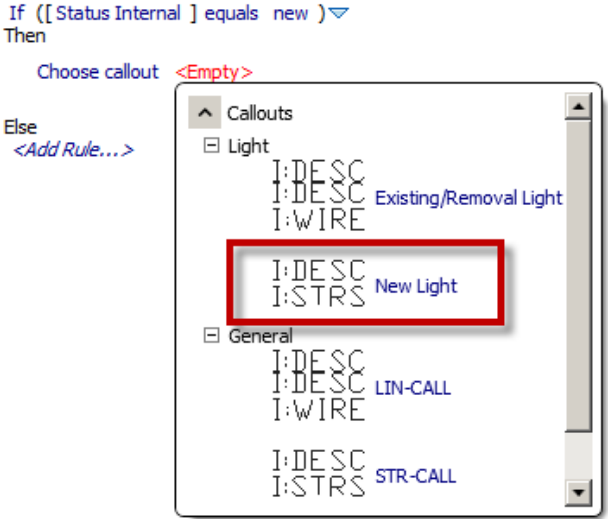

**Step 16:** Click on <Add Rule…> and insert an "If" Rule with the following properties:

- Look under Light Attributes to find the property "Status Internal".
- Leave comparison as "equals".
- Set the second value to "new" (it will likely default to this).

**Step 17:** Expand the rule to see the Then and Else clauses.

**Step 18:** If the light is new, we'll use the first callout as follows:

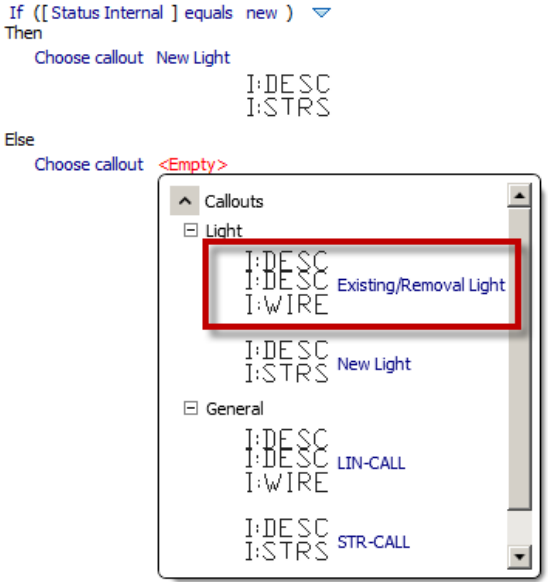
- Click on <Add Rule…> for the Then clause, and select Choose callout.
- On the right, choose the New Light callout from the list presented.



**Step 19:** If the light is not new, we'll choose the Existing/Removal Light callout:

- Click on <Add Rule…> for the Else clause.
- Select Choose callout.
- Choose the Existing/Removal Light callout

**Step 20:** Click on Apply once the Rule is complete, then click on OK.




**Test the Callouts**

Now you can test the Callouts we've created and the associated Rules.

**Step 21:** On the Home tab, click on the Insert Callout command in the Annotation panel.

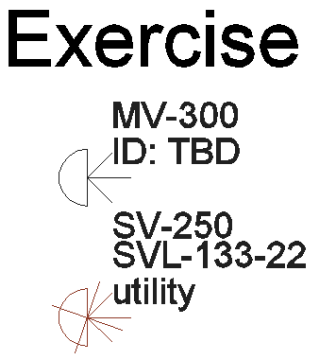

**Step 22:** AUD will prompt you to select features to insert Callout. Select the two Lights and press Enter.

**Step 23:** AUD will prompt for three values. Enter as shown here:

- X Offset: 6
- Y Offset: 1
- Rotation: 0

**Step 24:** At the leader prompt, enter "No."

**Step 25:** The Callouts should appear as shown to the right.



This completes this exercise.