CI121848

Cadd Nuggets with Conditional Dipping Sauce

Speaker: Jeff Frye - HDR, Inc.
Co-Speaker: Spenser Hays - HDR, Inc.

Learning Objectives

- Leveraging 2d Model data
- Production Tools
- Conditional Sub-Assembly examples

Description

This class will be full of nuggets of wisdom from seasoned professionals. We will show users how to use conditional sub-assemblies to minimize the number of regions and assemblies used in corridor design. Add additional value and design information to your model by targeting 2D lines from xref layers. This class will demonstrate a workflow for corridor modeling and sheet production and maximizes the model while keeping all aspects of production in mind. We will use data extraction to help with quantity take offs on 2D linework that historically is not counted and is under-utilized.

Speaker Bios

Jeff is a veteran design professional with over 27 year's experience. He is production oriented and is always looking for faster ways to use the software and tools at his disposal. He has become a valued internal training manager within HDR.

Jeff Frye

HDR, Inc.

Jeff.Frye@hdrinc.com

Spenser is a registered professional engineer with a focus in Transportation. He is a design lead on multi-million dollar roadway corridors that utilize the BIM/CIM process. His passion in Civil 3D lies with any aspect that helps to improve or speed up a process. He has also become a valued internal trainer of Civil 3D within HDR.

Spenser Hays

HDR, Inc.

Spenser.Hays@hdrinc.com

What is a nugget?

- An unexpected tip, trick, time saving command or other click reducing workflow not originally intended to be the main topic at that time.

What is a Nugget?

We have attended many classes, training sessions, viewed online videos and discussed cadd with many other professionals. During these interactions, we always came away with some other tip or trick that was not the actual topic being discussed. We decided that we would put together a class to share some of the more noteworthy with you.

The following pages will walk you thru some very basic modeling components and how to maximize them. We will show you how to add annotation to 2D linework, assemblies, alignments and polylines. We will give you some tools to use the next time you want quick quantities from your files. And finally, we will give you some very valuable tools surrounding conditional sub-assemblies.

We have divided the sections as follows:

- Alignment
- Profile
- Assembly
- Existing Ground
- 2D linework
- Conditional Subassemblies
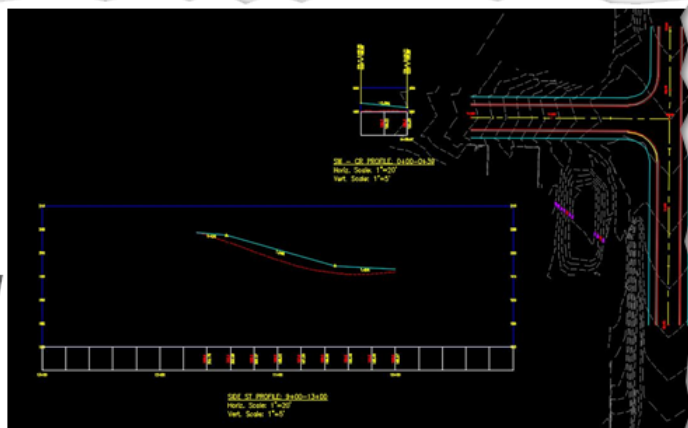
Alignment

Without a label, it is difficult to identify your alignments, especially if you have multiple offset alignments in a modeling environment. You can add a note or alignment label to display the name and description of your alignments to help you and other users navigate your model.
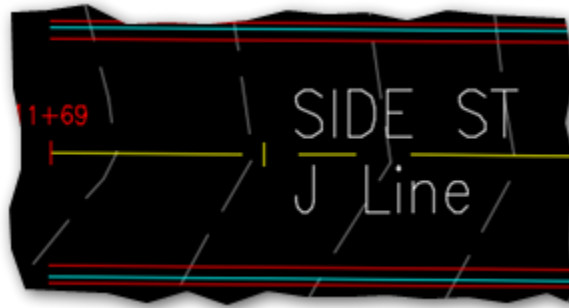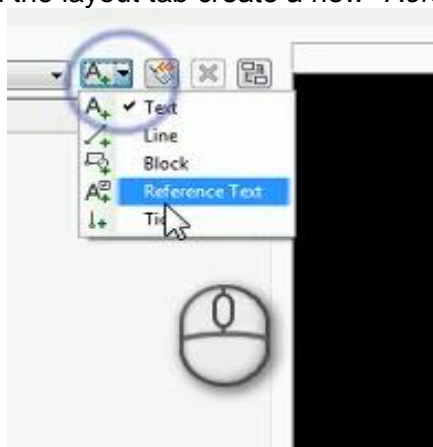

In this exercise we will create a label style and apply it to the alignment.

Steps:

1. Create new **NOTE** label style called "*Alignment Name and Description*"
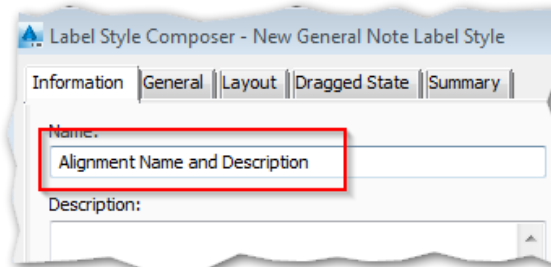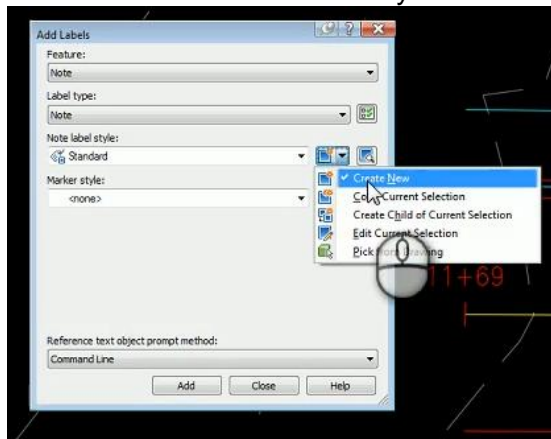
2. On the layout tab create a new "*Reference Text*" component and select Alignment

3. In the Contents text box change the default "*label text*" to display the alignment name over the description as shown below by selecting the *Alignment Name* property and hitting the right facing arrow repeat with the *Alignment Description*
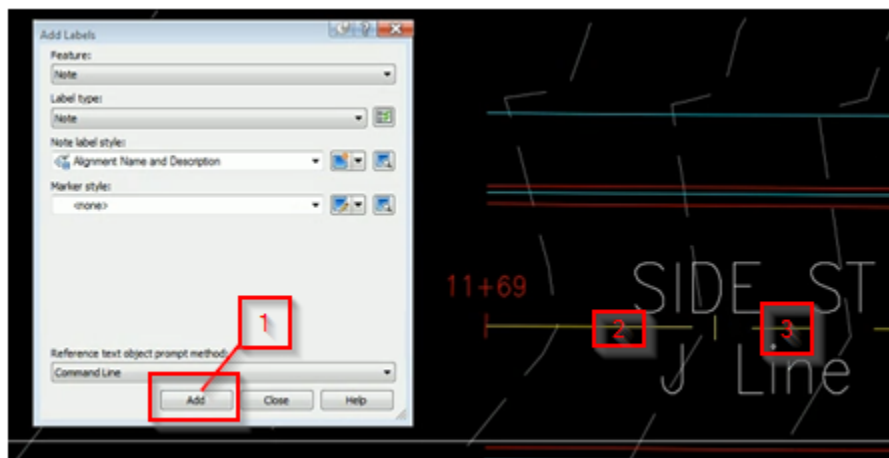

Follow the numbered steps below to create the label on your alignment

Step 1 Select "Add"

Step 2 Select the alignment

Step 3 Select the location to place label

Superimposed Intersecting Profiles

Design profiles at intersections are very difficult to manage. One method of tracking what elevation your side streets are at relative to your Mainline (and vice versa) is to use "*super imposed Profiles*".

In this exercise we will superimpose the "Side Street" profile onto the profile view of "Main Street". This will help us see where the side street profile intersects the mainline in profile view.

To begin we must have a profile style that displays markers at the beginning and end of the profile.
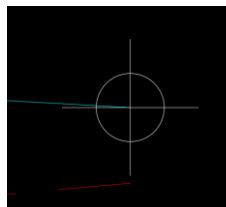Steps:
1. Create a new profile style named "FG with Markers"
    On the Marker tab, change Begin Profile and End Profile to Basic Circle with Cross:
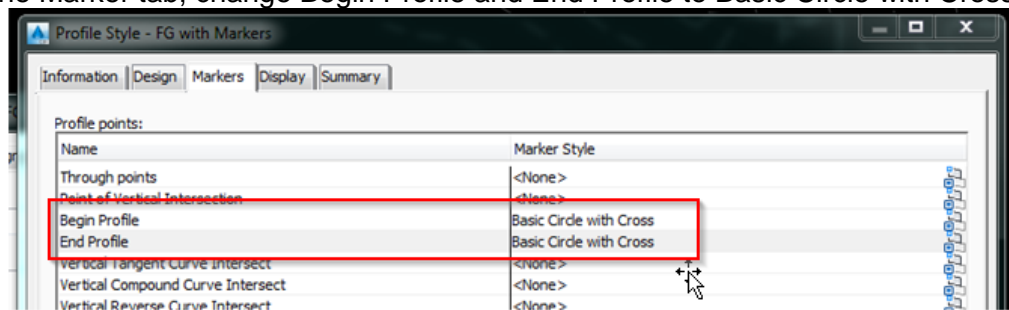

2. Set the profiles to this new style. This marker should appear now at each end of your profiles:

3. Click on the profile view and select "*super imposed profile*" from the ribbon
3a.   Select the source profile (the one you want to show on the other profile view).
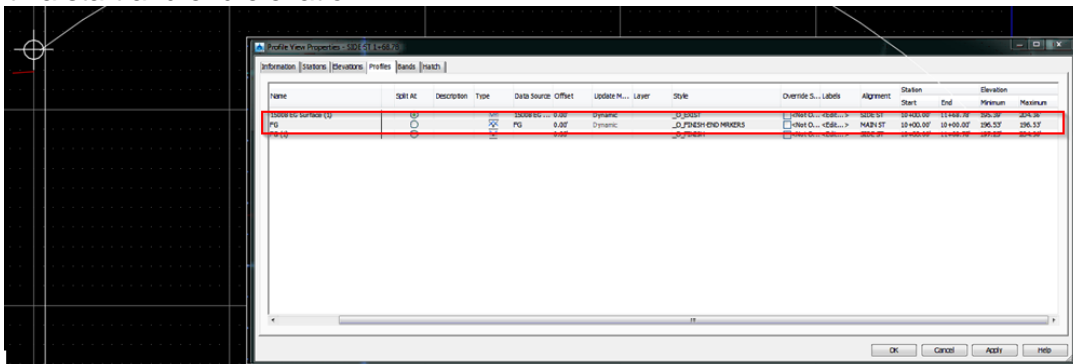3b.   Select the destination profile view.

4. You will only want a small portion of the profile to show on the destination profile view. Pick station 10+00.00 as the starting station and 10+00.01 as the ending station to display:
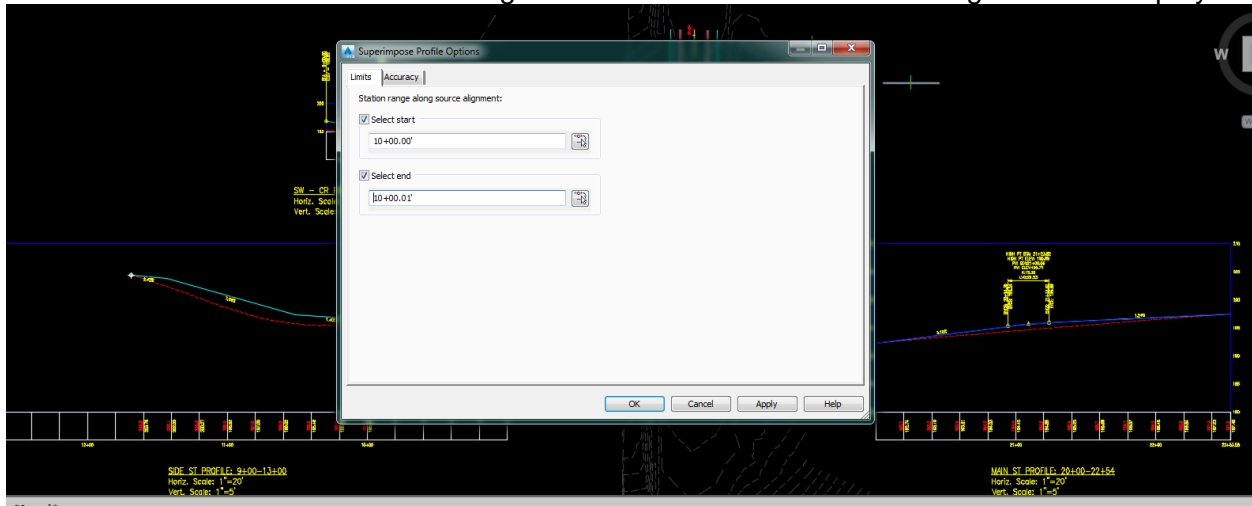
Now the marker shows on the destination view:
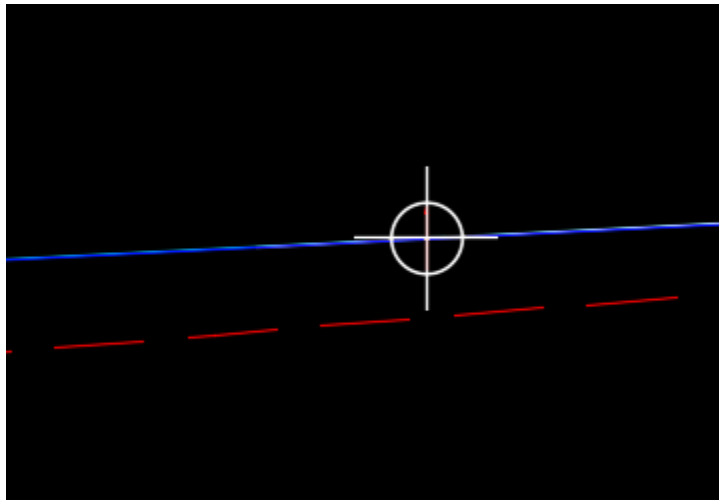
Although you cannot snap to this profile, you can see the elevation by showing the destination profile view properties, on the profiles tab you will see the profile listed there with a start and end elevation.


Once this is complete, you can adjust the profile geometry of the source profile and watch as the destination profile view updates.
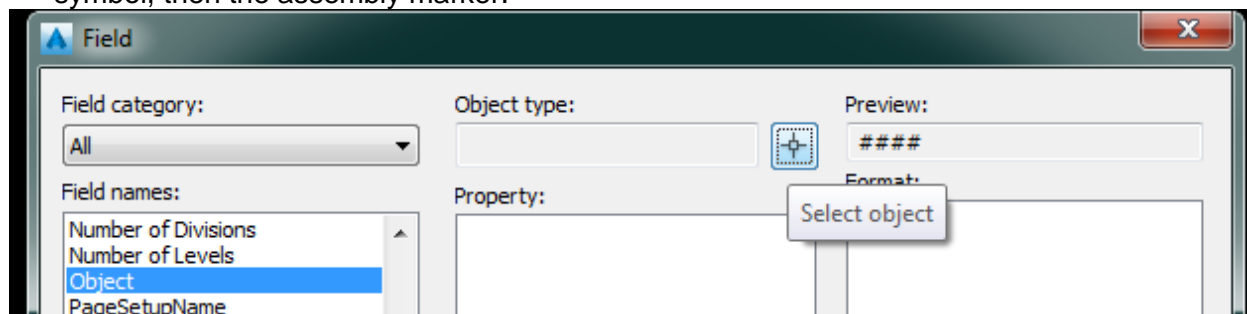
Assembly Name

It is sometimes difficult to find assemblies in your model when you have a lot. This exercise will demonstrate how to use Fields to annotate assemblies.
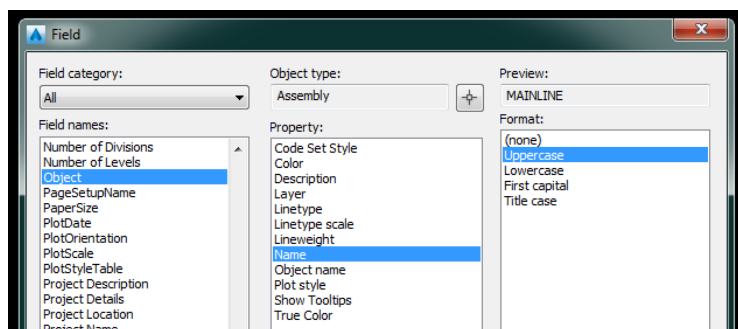
Steps:

1. Locate your assembly and place a piece of text above it that says "NAME".
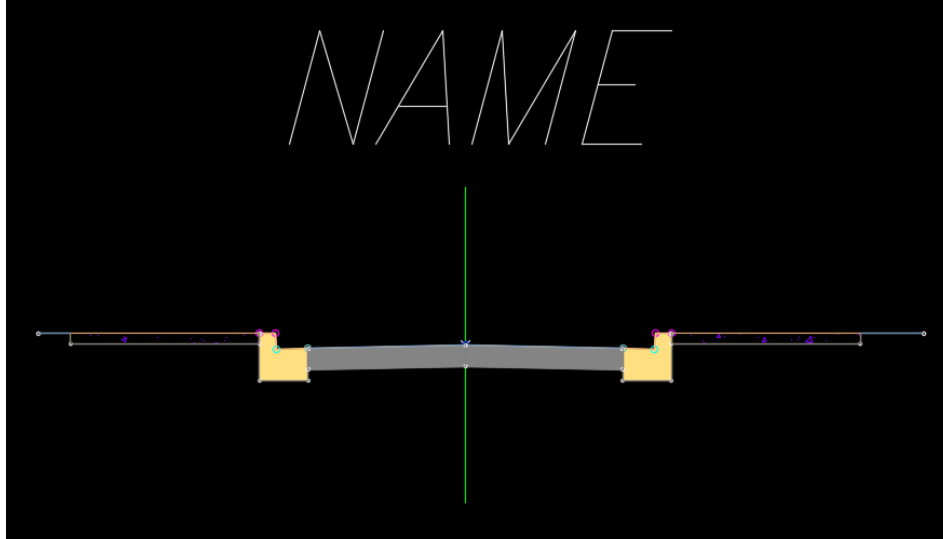
2. Highlight the text as if you are going to replace the text, and right click Select "*Insert Field*".
3. Select *object* in the field name column. In the property column, select the "*select objects*" symbol, then the assembly marker.

4. Select "name" and your format preferences

The assembly name will appear above your marker

****
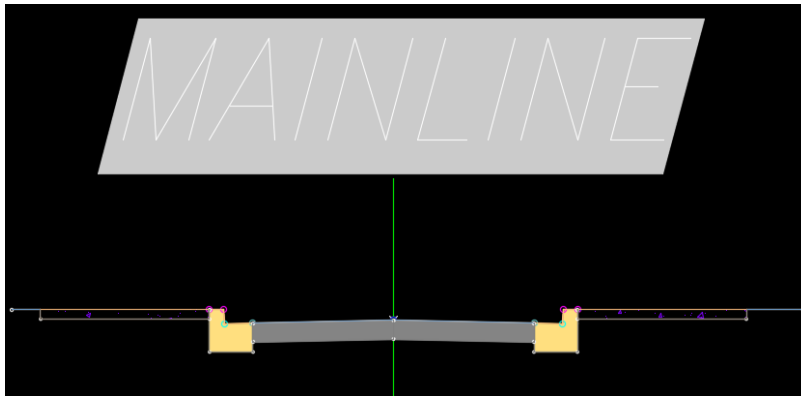
Fields can be used on many other objects in your model:
Polylines can display; length, area, and elevation.

Controlling sub-assembly parts using other sub-assemblies!

To save on clicks, you can edit all of the sub-assembly parts by right clicking on the assembly marker and editing in the assembly editor.
AND you can save even more clicks by using the Parameter Reference column in the assembly properties dialog box!!

Exercise: Create a new assembly, add two "ShapeTrapezoidal" shapes to the right side.
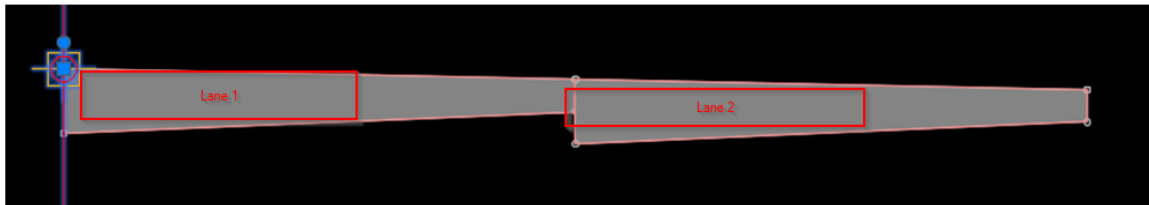
In this exercise we want the outside lane to mimic the some of the properties of the inside lane. But we do not want to have to remember to set them.
We will use the Parameter Reference column to "USE" the values from the inside lane (Lane 1) and apply them to the outside lane (Lane 2).
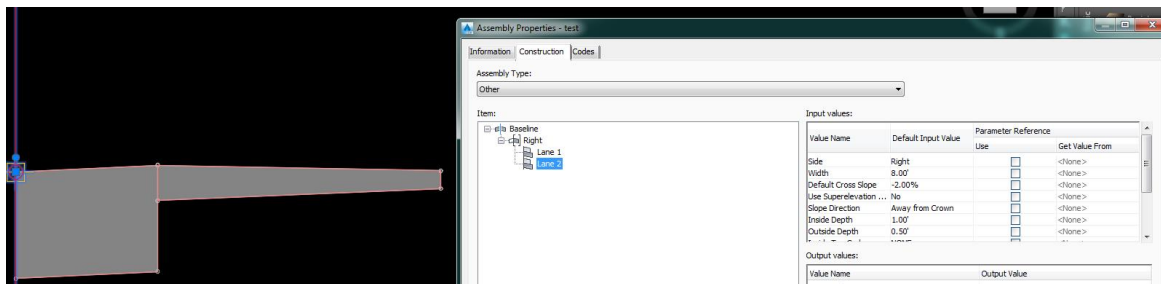
NOTE: the lane 2 values and different than Lane 1.

Using the Parameter Reference column you now select "Use" and "get value from" to select the property of Lane 1 you want to apply. We will set the width of Lane 2 to the "lane 1 width". Hit apply and notice the Lane 2 width is now the same as Lane 1.
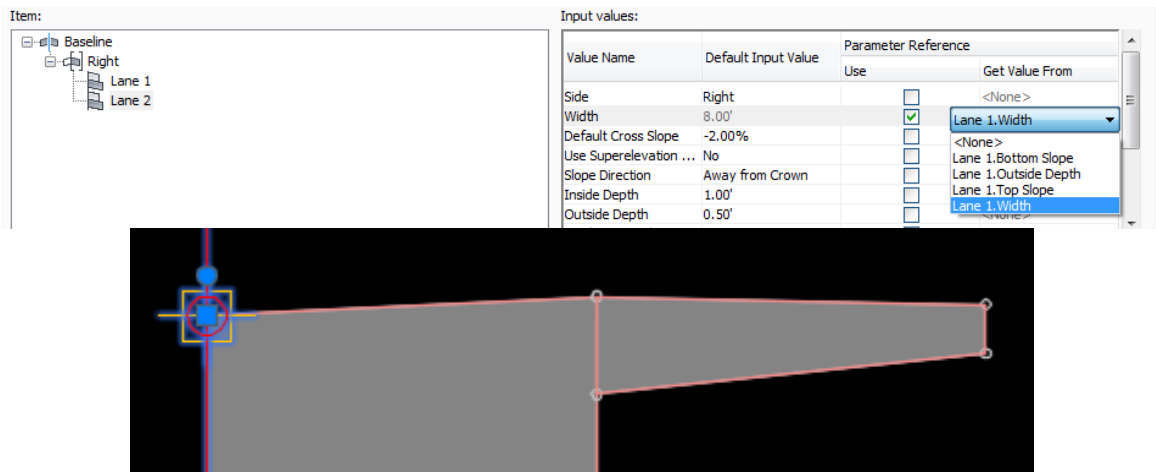
You can get values of objects that are closer to the assembly marker only. Ie; you cannot get values from Lane 2 and set them to Lane 1
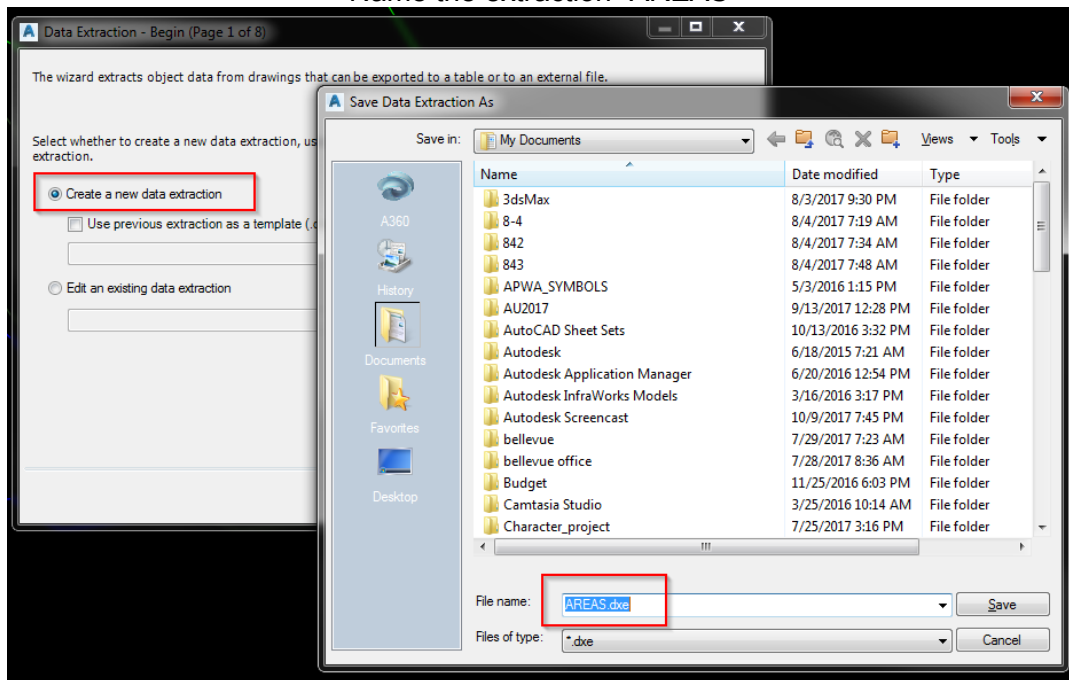
Data Extraction

In this exercise we will extract area attributes from basic 2D polylines and display the data in a dynamic table.
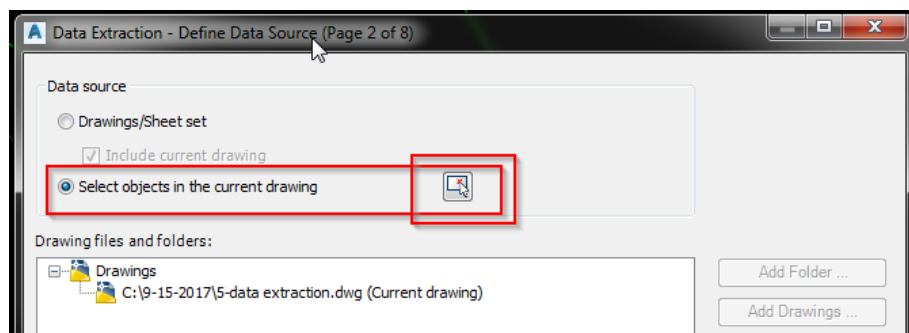
2D closed polylines

1. Command: DATAEXTRACTION

Name the extraction "AREAS"

Choose "select objects in current drawing and select the closed polylines
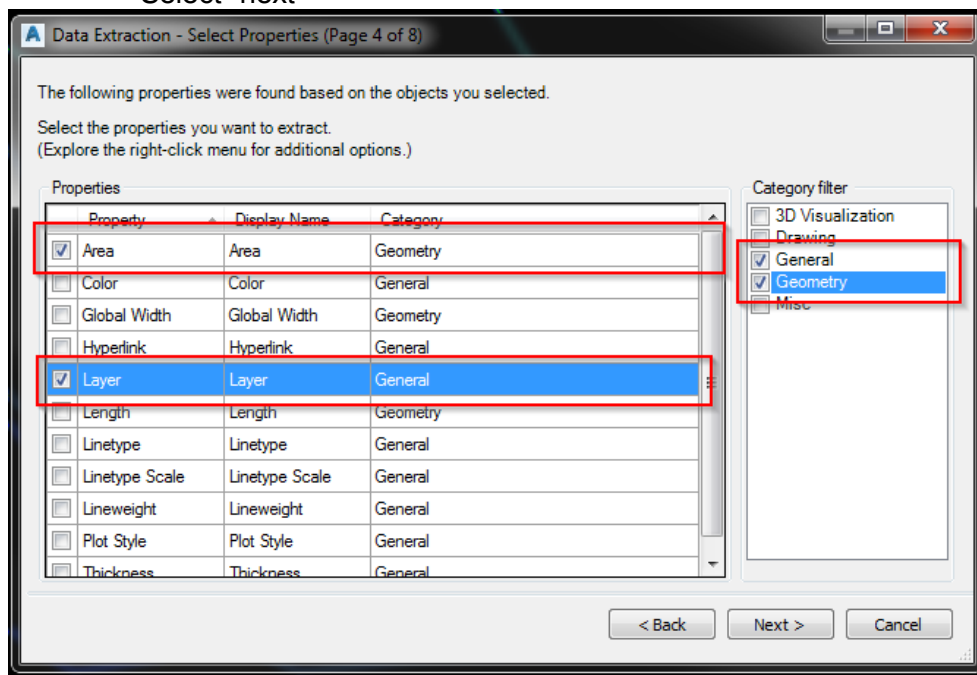

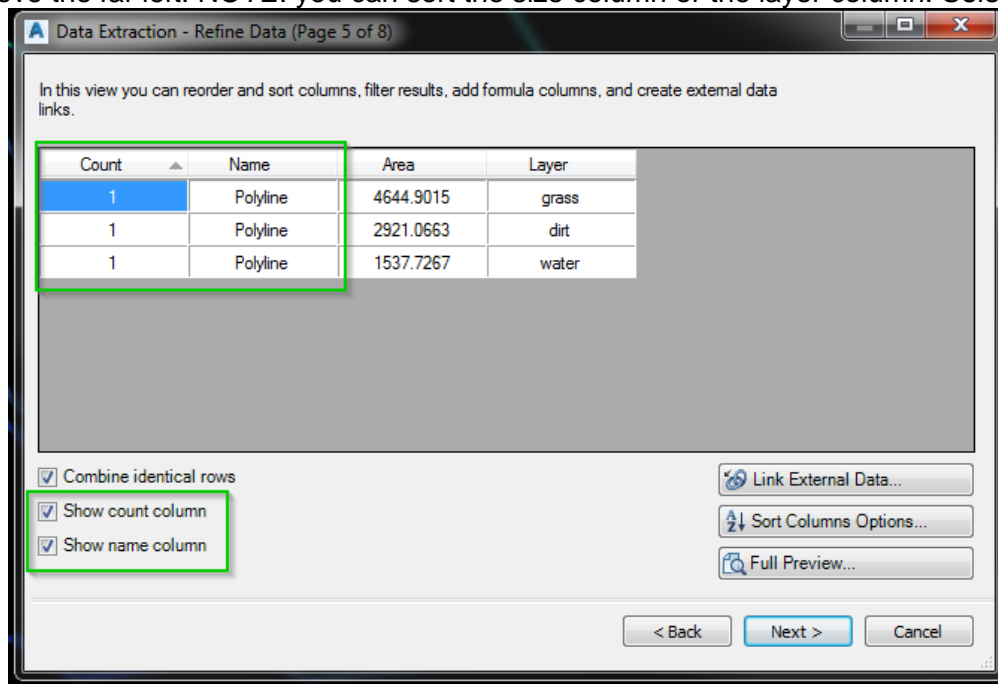Select Next and notice the objects, they should be polylines only

Select next and select the following property check boxes and category filters:
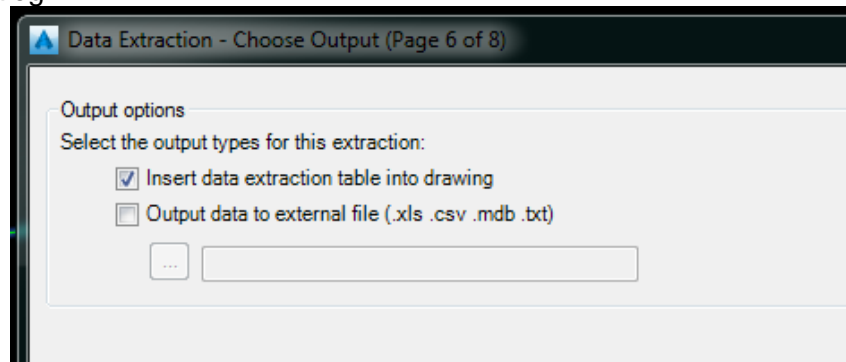
  AREA and LAYER
  Select "next"

In this dialog, uncheck "show count column" and "show name column". Grab "layer" column and move the far left. NOTE: you can sort the size column or the layer column. Select "next"

Select "Insert data extraction table into drawing" select "next" to display the table dialog

For this exercise select the standard table style, and title the table "AREAS". Leave the check mark on "Use property names as additional column headers" This will put "LAYER" above the proper column
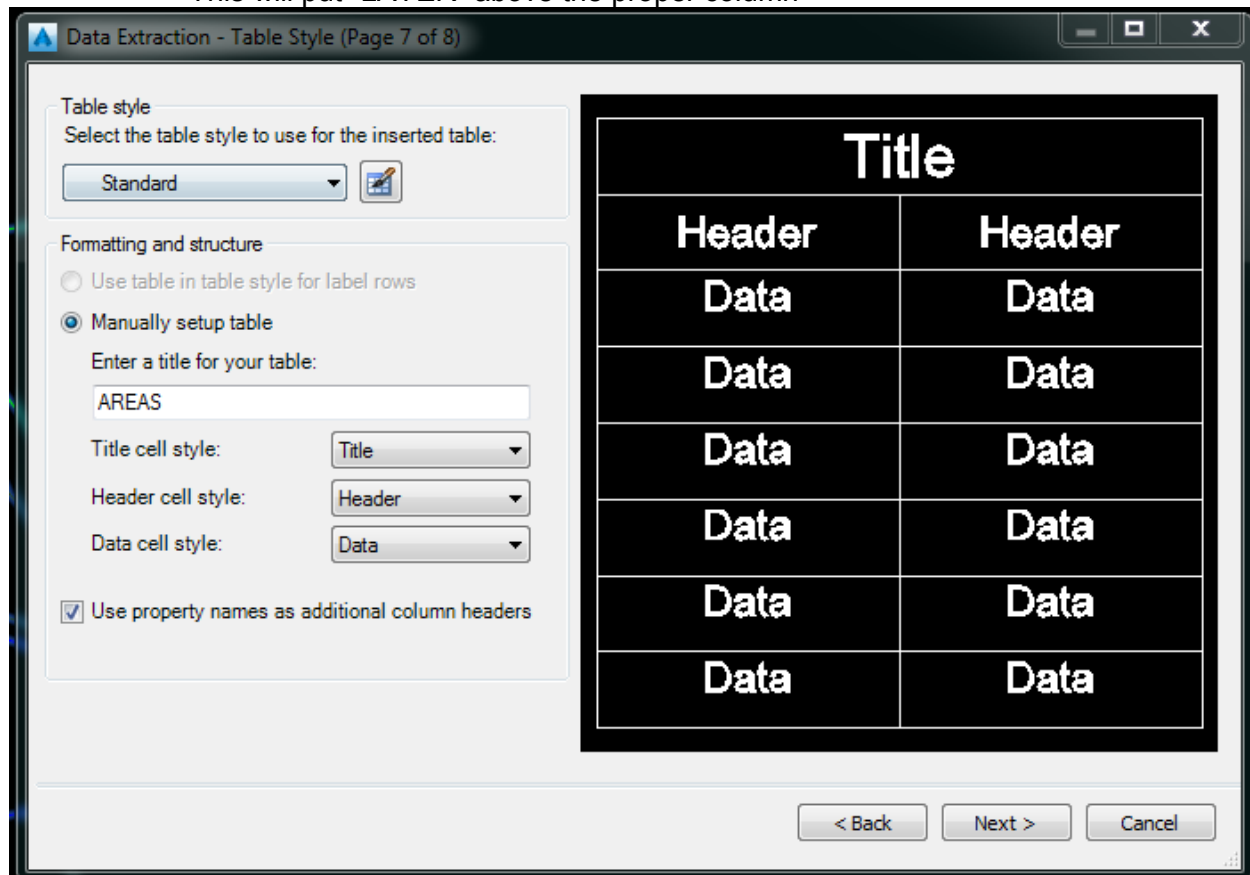
AREAS	
Layer	Area
grass	4644.9015
dirt	2921.0663
water	1537.7267

You can now update the boundaries of the polylines and regen the file and you will see your new values in the table.

Useful Links:
Video showing basic attribute data extraction:
https://www.youtube.com/watch?v=V6CAM4Ich08

Autodesk Knowledge site:
https://knowledge.autodesk.com/support/autocad/learn-explore/caas/CloudHelp/cloudhelp/2016/ENU/AutoCAD-Core/files/GUID-D1433C93-0E7F-46D0-8D05-78DF3626B146-htm.html

Data Extraction

In this exercise we will use an excel file with numerous construction notes to display only those notes that we want to see.

Sometimes a discipline sheet will have several construction notes. Often times some do not apply to that sheet, or they do not all fit on the sheet. Some clients are moving toward showing only the notes that apply to that sheet. In any of these cases we can use data extraction to link a master excel file with a full set of notes to attributed blocks on a per-sheet basis.

Select Construction notes

      Required data:

1. Excel file with two columns "number, and construction note"

2. A block to be used as a note bubble with at least one attribute

   You can insert these blocks thru Mleader styles or standalone blocks.

3. Start the **DATAEXTRACTION** command, name the DXE file "*construction notes*", and save to a location of your choice.

4. Choose "*Select Objects in the current drawing*" and select the blocks that contain the attributes.

5. Hit next

You should have a dialog that looks similar to this
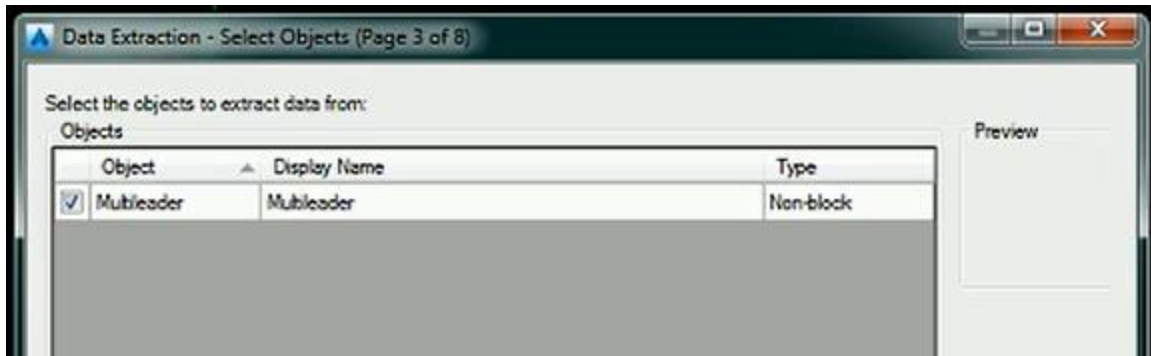
6. Hit next

7. Uncheck all but "Attribute" in the category filter column to show only the attributes
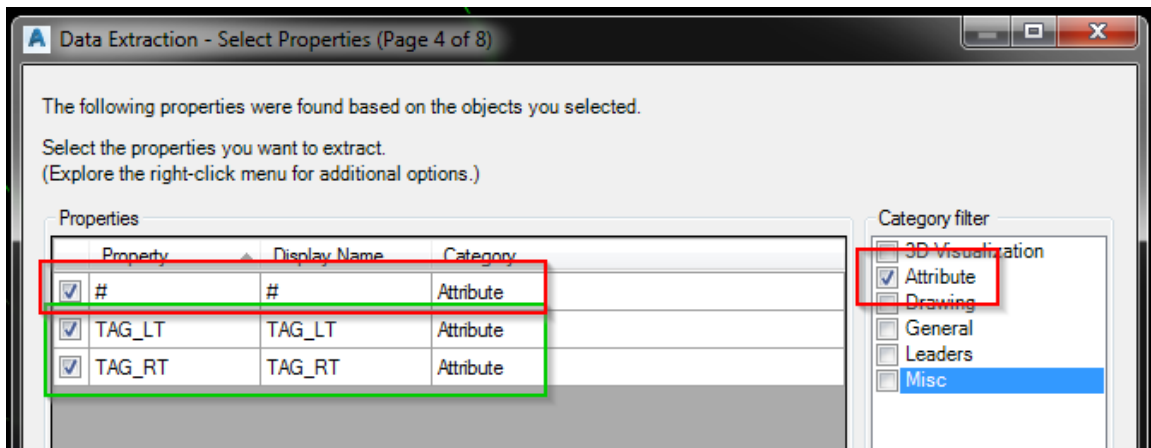
8. Uncheck TAG_LT and TAG_RT, we will use those in a later exercise.

9. Hit next

10. Uncheck "*show count*, and *show name*". This will leave only the numbers. Next we will attach the numbers to the spreadsheet we created

11. Select "link external Data" to link the data

12. Create a new data link and name it "Construction *Notes*"
13. Select "*create a new excel data link*" and browse to our excel file that contains the construction notes.
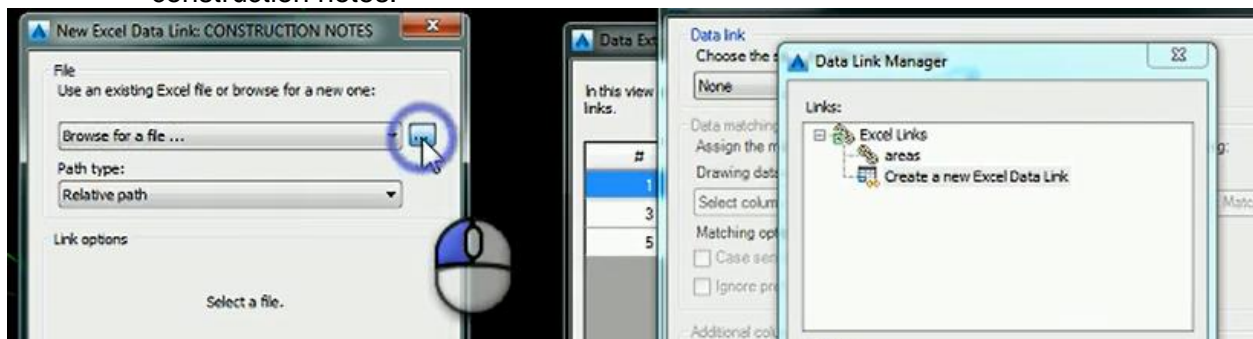
14. Link to the tab you have your notes on
15. This next step is the most important, we need to validate the data in the file against the spreadsheet. In the "drawing data column" select "#" and in the "External data column" select "Number" then select "Check Match" to let C3D verify the data.

16. Hit OK
17. Hit Next
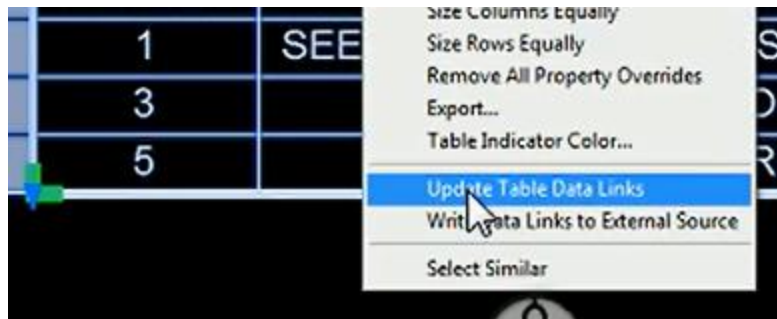18. Now select insert data extraction table into drawing and follow the prompts.
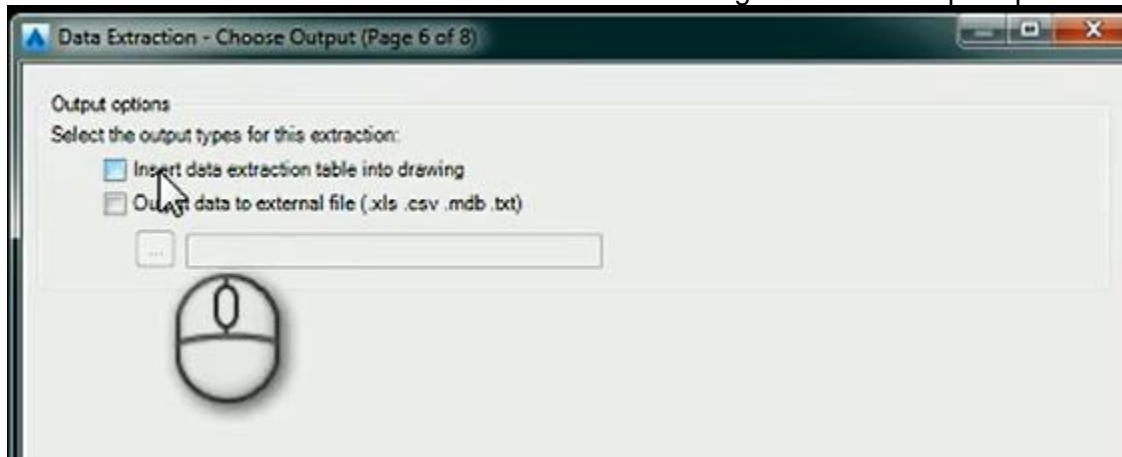
Your table should look similar to this:

CONSTRUCTION NOTES	
NUMBER	CONSTRUCTION NOTE
1	SEED PER LANDSCAPE SPECIFICATIONS
3	PROTECT BODY OF WATER
5	PROTECT CONCRETE PAD

You can format the table like any other table.

19. Now change the bubble 5 to 2
20. Right click on the table and select "update table data links" to display the new list of construction notes
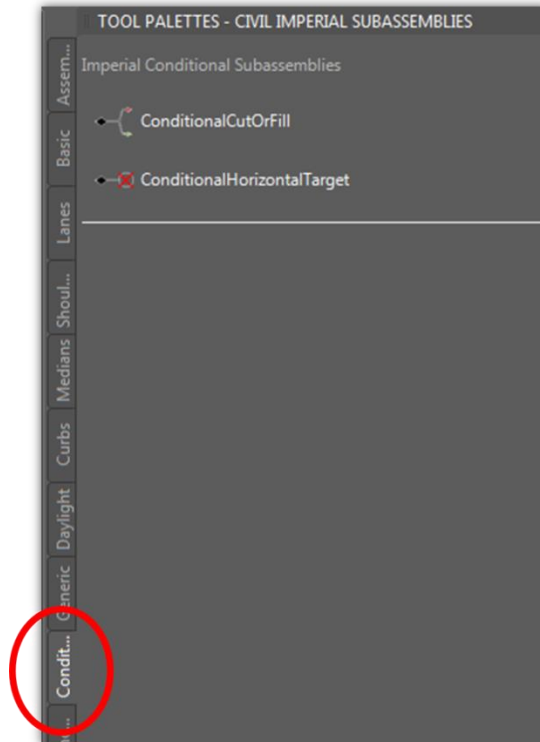
And you can continue to format the table as needed.
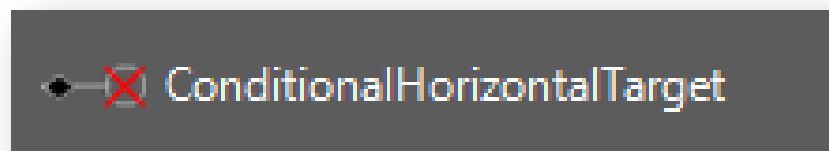
Conditional Subassemblies

What are they?

Conditional Subassemblies are subassemblies used in conjunction with other subassemblies that will model the connected subassembly when a condition is met.

- **ConditionalCutOrFill** – Used if a Corridor Model requires that different subassemblies are to be built in a "Cut" situation vs. a "Fill" situation. Will build a connected subassembly if in a Cut situation and will build a different subassembly if in a Fill situation.
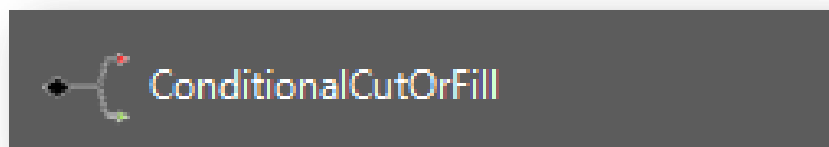

- **ConditionalHorizontalTarget** - Will build a connected subassembly if it finds a Horizontal Target within its specified distance. Horizontal targets consist of: polylines, feature lines, and alignments.

How Do They Operate?

Conditional subassemblies can be compared easily to an "If-Then" statement is used in an Excel formula. If the subassembly meets the specified condition then perform this action (i.e. model this assembly). If the subassembly does not meet the specified condition then perform a different action (i.e. model a different subassembly).

IF
"CONDITION" IS MET → THEN "ACTION"

IF
"CONDITION" IS NOT MET → THEN DIFFERENT "ACTION"
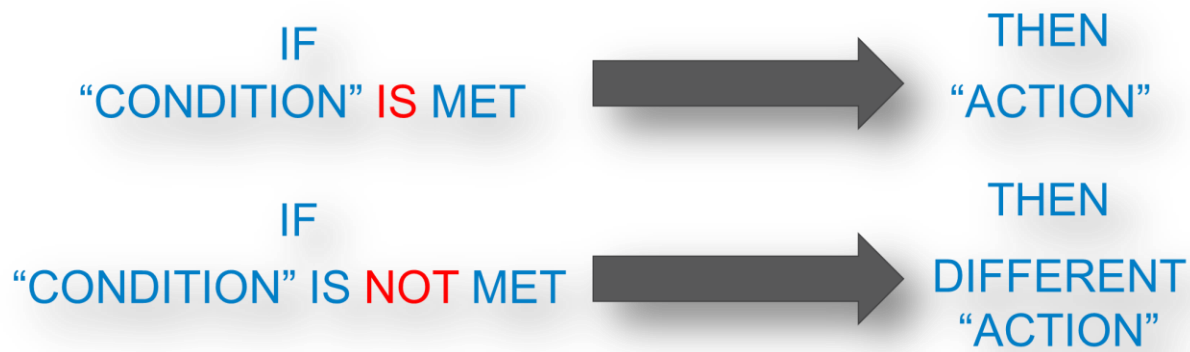
The Four "If-Then" Statements Used by the Conditional Subassemblies

- **ConditionalCutOrFill**

1. If the subassembly is in a Cut condition, build the subassembly that is connected to the "Cut" Conditional.

2. If the subassembly is in a Fill condition, build the subassembly that is connected to the "Fill" Conditional.
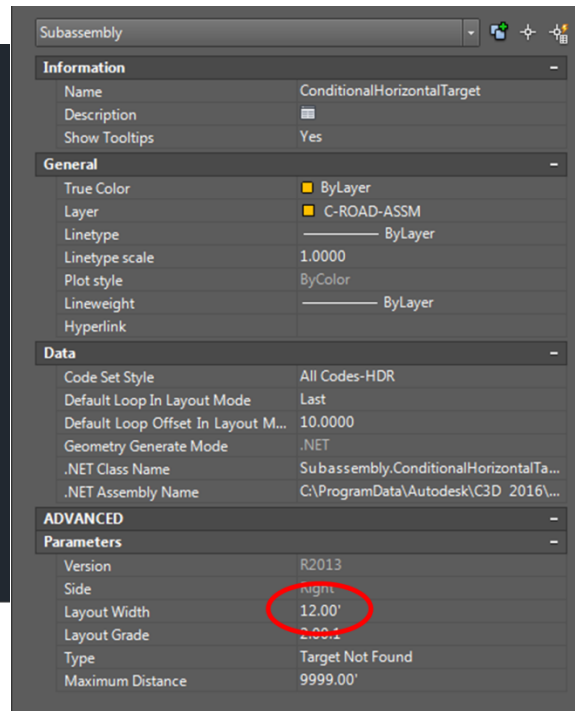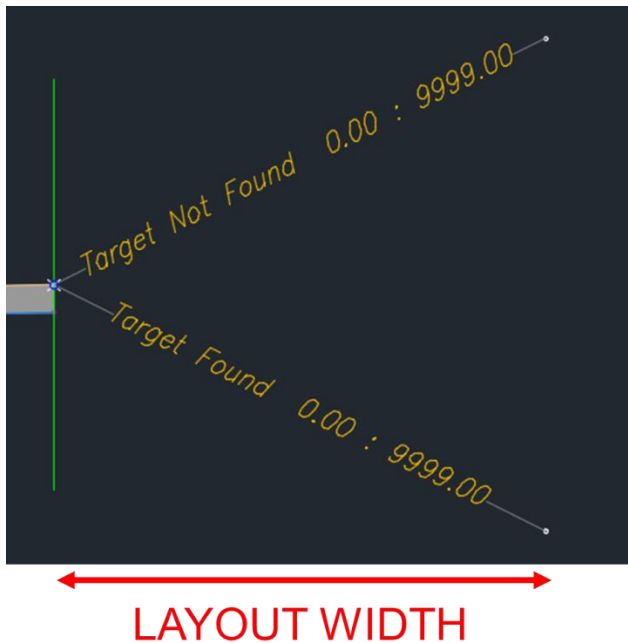
- **ConditionalHorizontalTarget**

3. If the subassembly finds the horizontal target within the specified distance, build the subassembly connected to the "Found" Conditional.

4. If the subassembly does not find the horizontal target within the specified distance, build the subassembly connected to the "Not Found" Conditional.

Organizing the Conditionals

   As your model becomes more complex and you add additional Conditional Subassemblies, the organization can become cluttered very quickly with each new addition. Here are the four settings available in the subassembly properties that can be adjusted to help organize a complex assembly.

- ▪ **Layout Width** – This setting adjusts the horizontal width of the subassembly from its connection point outward.


LAYOUT WIDTH

- ▪ **Type** – This can be changed in the properties palette and adjusts the **ConditonalCutOrFill** between "Cut" or "Fill" or the **ConditionalHorizontalTarget** between "Target Not Found" and "Target Found." If you imagine a horizontal line coming out of the connection point, the "Target Not Found" and the "Cut" conditionals will always be above horizontal and the "Target Found" and "Fill" conditionals will always be below horizontal. So when the type is changed in the properties, the conditional will change to be above or below horizontal to reflect the change.

23
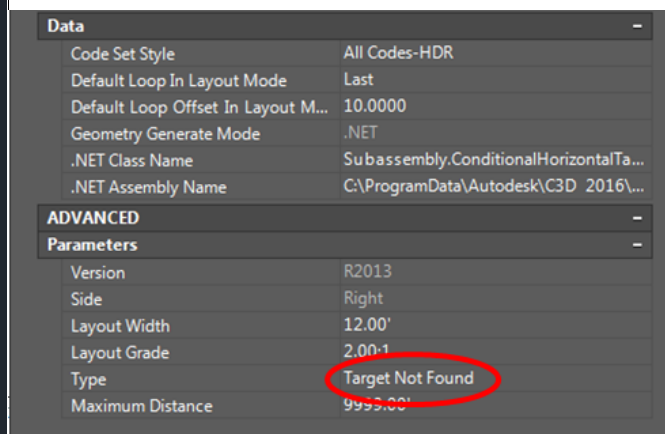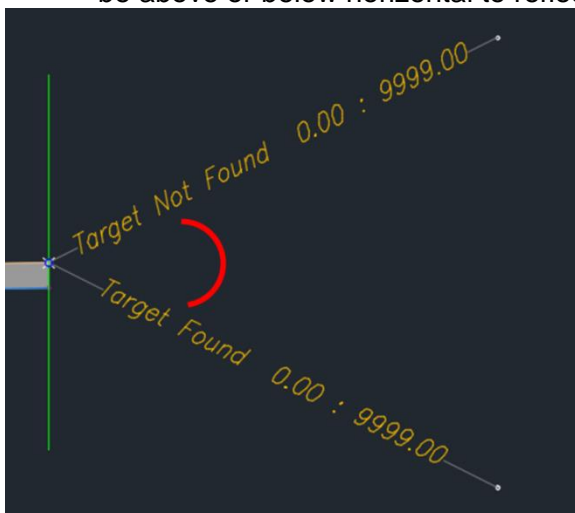
▪   **Layout Grade** – The layout grade is the grade (or slope or angle) of the conditional from its main connection point. This number is always positive and not negative because it is the type of conditional that determines if it is above or below horizontal.
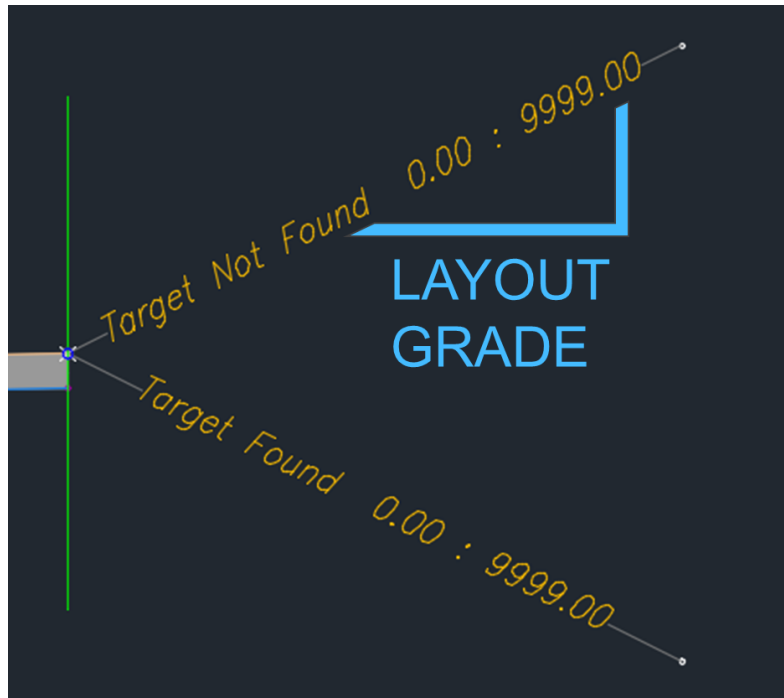
●   **Maximum and Minimum Distance** – These properties have no affect on the layout of the subassembly in model space, but reflects the distance that the subassembly searches for the condition. The default maximum distance is set at 9999' while the default minimum distance is set at zero. These properties can be adjusted to a specified range.

o   In the **ConditionalCutOrFill** subassembly, the max. and min. distance is measured vertically from the innermost connection point of the subassembly. The distance between the connection point and the target surface is then measured and if this distance is within the specified range, the connecting subassembly is built.

o   In the **ConditionHorizontalTarget** subassembly, there is only a maximum distance and that distance is measured horizontally from the innermost connection point out to the specified horizontal target. If this distance is within the specified range, the "Found" conditional will build its connected subassembly. If the distance is not within the specified distance, the "Not Found" conditional will build its connected subassembly.
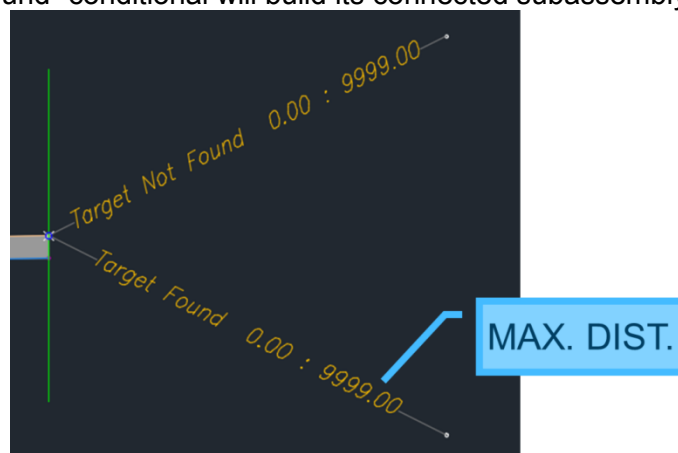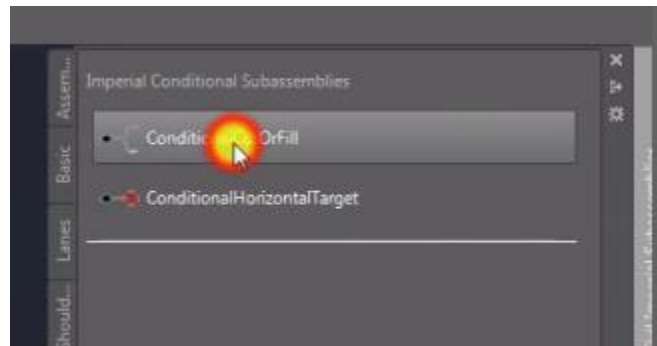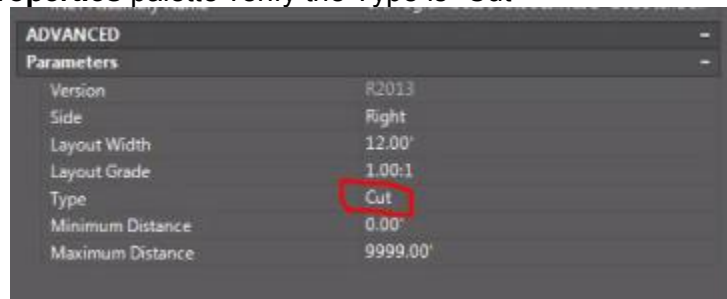
Creating a ConditionalCutOrFill Subassembly
1. Open the file, "Creating the Ditch Assembly.dwg"
2. Start with the main portion of your assembly. In this case a ditch with side slopes and a bench on each side.

3. Go to the **Tool Palettes – Civil Imperial Subassemblies ► Conditional Tab ► ConditionalCutOrFill**
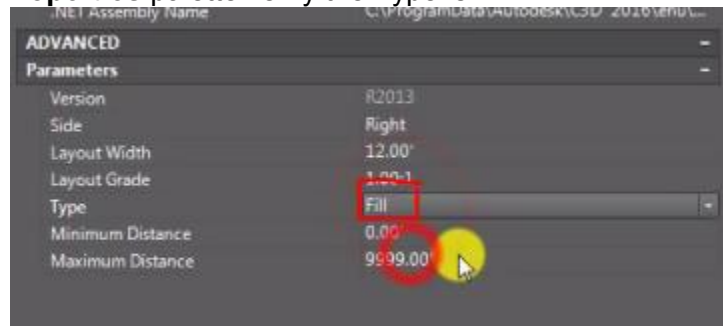

4. Go to the **Properties** palette verify the Type is "Cut"

5. Select the end node of the bench on the right side.

6. Go back to the **Properties** palette verify the Type is "Fill"

7. Select the same node again. This will have created a Cut and Fill conditional subassembly.
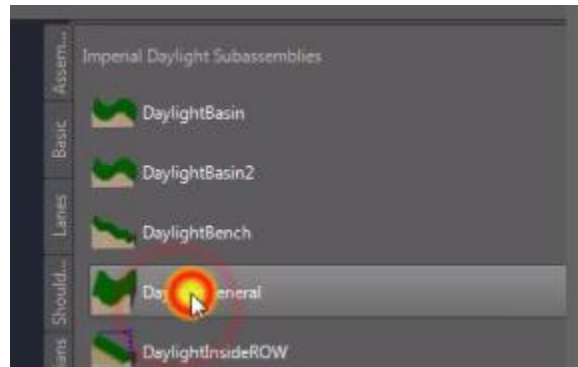
8. Select the bench and slope that make up a portion of the ditch. Right click ► Copy To

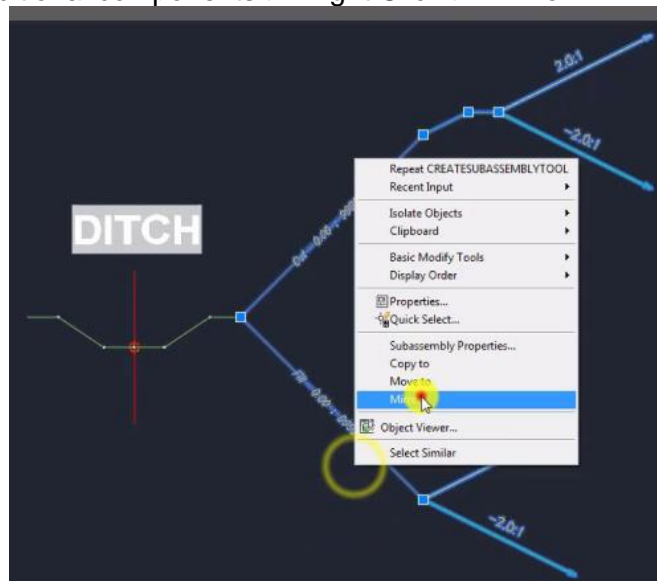9. Select the end node of the Cut subassembly.

10. Go to the **Tool Palettes – Civil Imperial Subassemblies ► Daylight Tab ► DaylightGeneral ► Properties Palette**. Set the slope parameters to all be at a 2:1 slope.

11. Pick the end node on the bench for our Cut conditional and pick the end point of our Fill conditional.
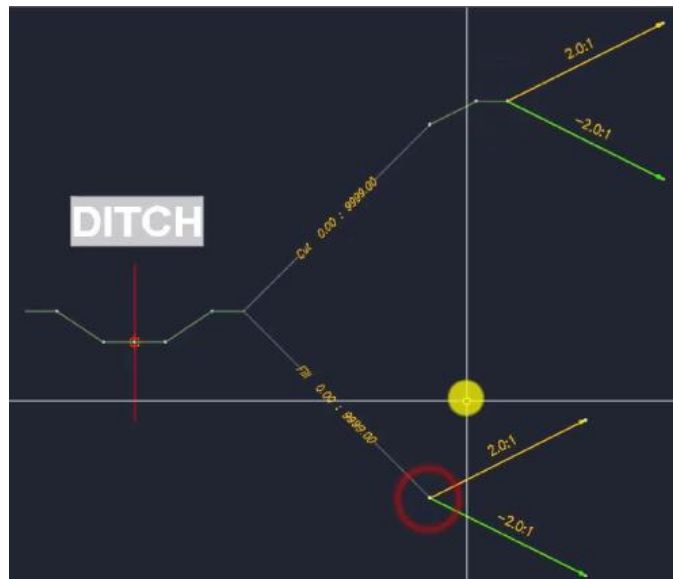
12. Select all the conditional components ► Right Click ► **Mirror**.

13. Then select the node on the opposite side of the assembly to mirror everything to the other side.

14. Repeat the above process to add one additional Cut and one additional Fill conditional to each side. Go to the **Tool Palettes – Civil Imperial Subassemblies ► Retaining Walls Tab ► RetainingWallVertical.** In the **Properties Palette,** update the **Maximum Distance** on the original conditionals to be 8.00'. Then update the **Minimum Distance** and **Maximum Distance** on the new conditionals to be 8.00' and 9999.00', respectively. These additional conditionals will model a retaining wall when we are in Cut or Fill that is deeper than 8.00'.
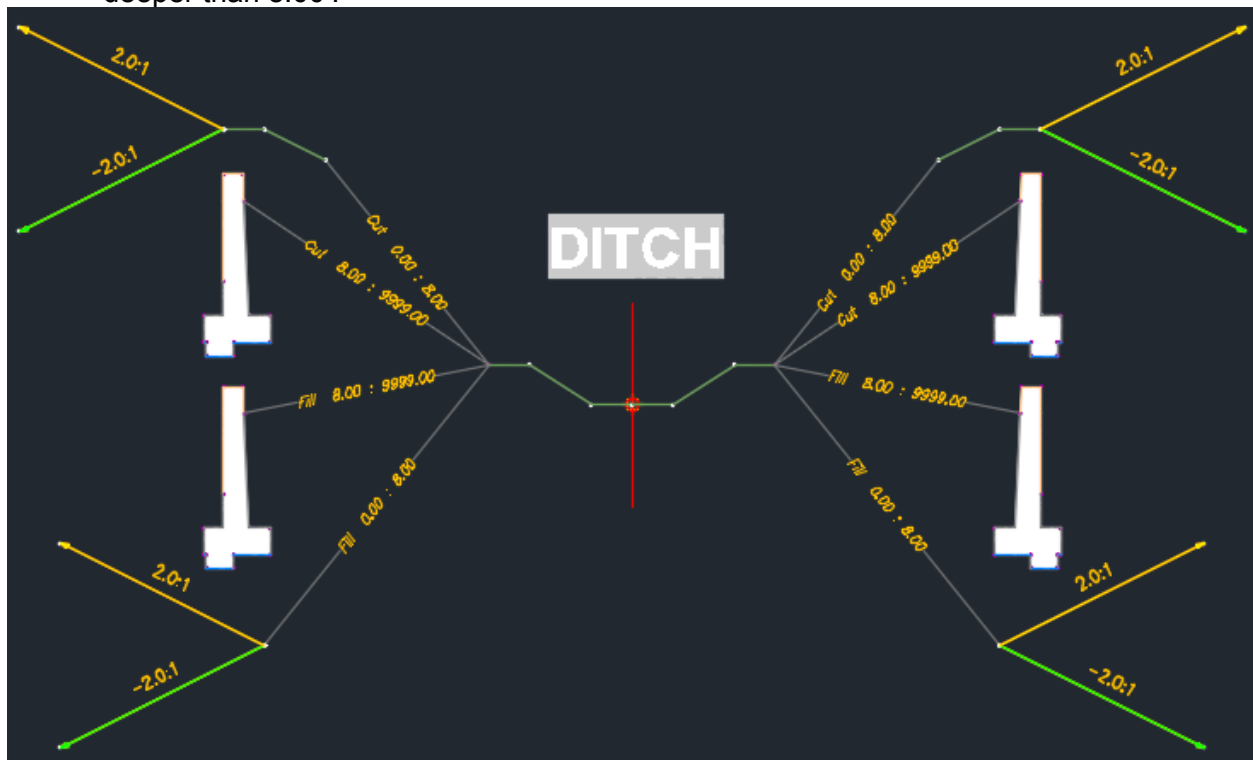
Using the ConditionalCutOrFill Subassembly

1. Open the file, "Using the Ditch Assembly.dwg."
2. Type **CREATECORRIDOR** and fill in all properties using the screen capture below as a guide. Do not set a target surface for now. ► **OK**

3. The Baseline and Region Parameters window will appear. Click the top ellipsis under the Target column.
4. The Target Mapping window will appear. The ConditionalCutOrFill subassemblies will appear in the top portion under the surfaces. These conditional subassemblies use a surface as their target to determine in the connecting node is in a Cut or Fill condition. For this case, we are building the ditch assembly in relation to the existing surface so all of our conditionals and the daylights will target existing. Select the top Object Name, <**Click here to set all**> ► **Landfill Lidar** ► **OK** ► **OK** ► **OK** ► **Rebuild the corridor**
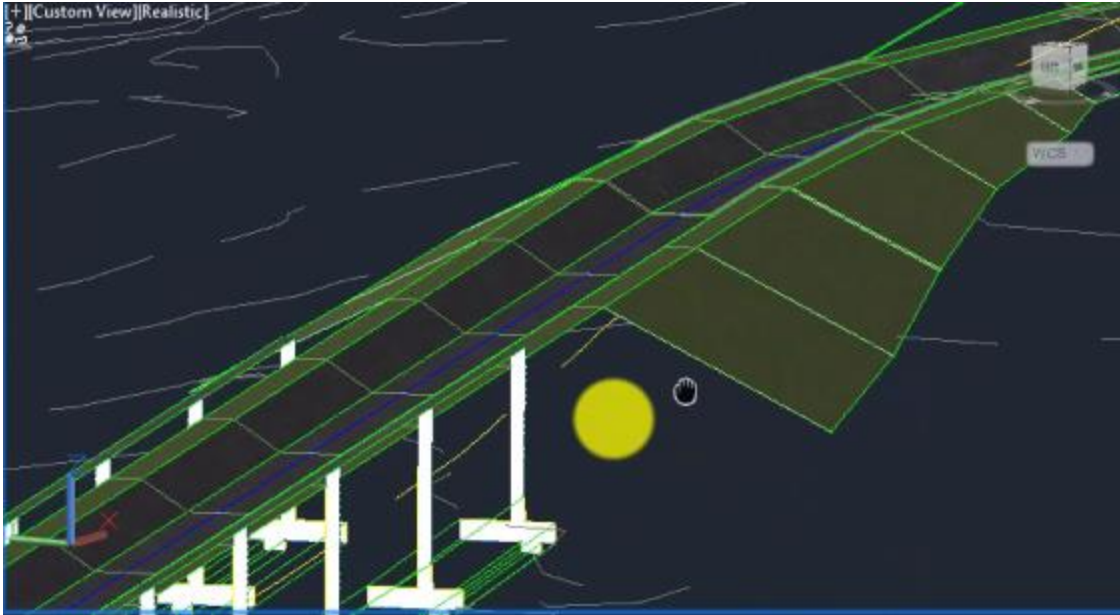
Creating a ConditionalHorizontalTarget Subassembly

1. Open the drawing, "Creating the Driveway Assembly.dwg"
2. Go to the **Tool Palettes – Civil Imperial Subassemblies ► Conditional Tab ► ConditionalHorizontalTarget.** Click the node on the end of the roadway lane. Then go to the **Properties Palette ► Type ► Target Not Found.** Then select the same node at the end of the roadway lane.


3. Add the following subassemblies to the Target Not Found conditional with the following properties:
   - **UrbanCurbAndGutterGeneral** – Keep all properties default except change the subbase dimension to 1.50'.
   - **LinkWidthAndSlope** – Change slope to 1.5% and width to 5.00'. In the link codes, add a link called "Bench" (e.g. Top,Datum,Bench).
   - **LaneSuperelevationAOR**
     - o Width: 5.00'
     - o Default Slope: 1.50%
     - o Pave1 Depth: 0.33'

o Pave2 Depth: 0.00'
o Base Depth: 0.00'
o Sub-base Depth 0.50'

4. Add the following subassemblies to the Target \ Found conditional with the following properties:

*   **UrbanCurbAndGutterGeneral** – Copy the curb and gutter from the Target Not Found condition and change "Dimension D" to 0.20".
*   Copy the **LaneSuperelevationAOR** from the Target Not Found conditional and connect it to the Curb and Gutter above. Adjust the following settings:
o   Default Slope: 5.33%
o   Pave1 Depth: 0.78'
o   Sub-base Depth: 0.60'
*   Select the above **LaneSuperelevationAOR ► Right Click ► Copy To** and click the end node of the **LaneSuperelevationAOR**. Adjust the following settings:
o   Default Slope: 1.50%
*   The end result should like the screenshot below:
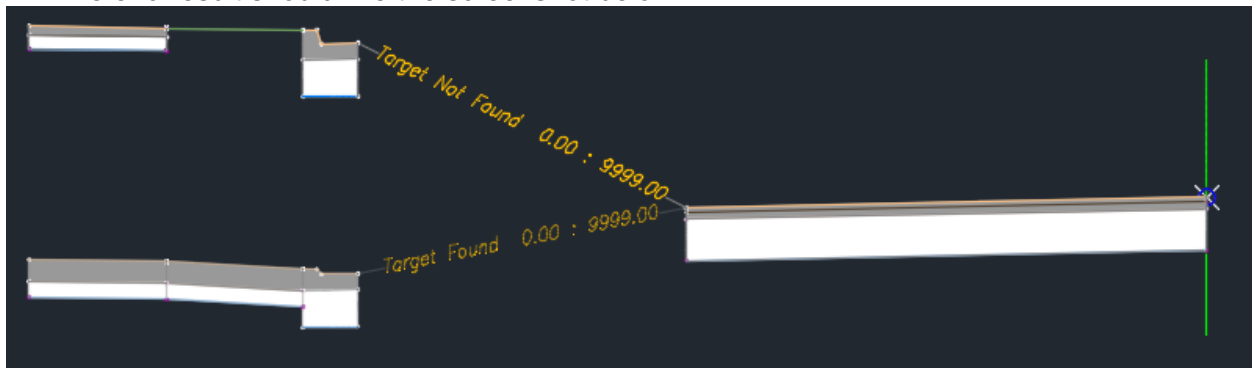
Using the ConditionalHorizontalTarget Subassembly
1. Open the drawing, "Using the Driveway Assembly.dwg."
2. Draw a polyline at the back of the driveway as a target for the Target Found condition of the driveway assembly.
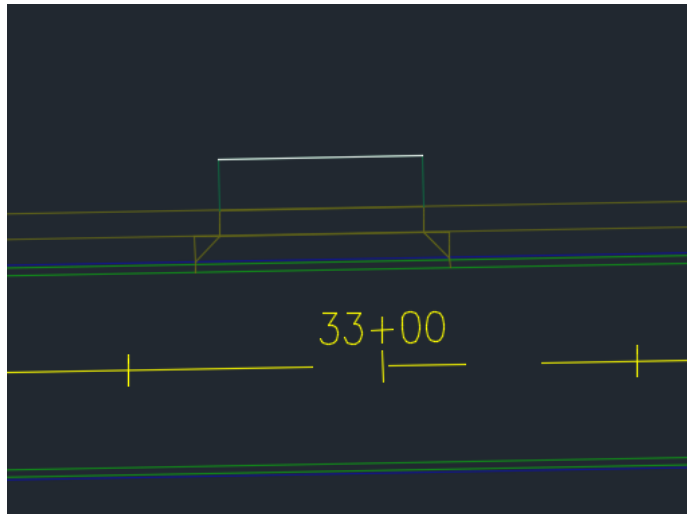
3. Type **CREATECORRIDOR** and fill in all properties using the screen capture below as a guide. Do not set a target surface for now. ► **OK**

4. The Baseline and Region Parameter window will appear. Click **OK** for now.

5. Not setting any targets will build the Target Not Found subassembly only.


6. Select the Corridor ► Right Click ► Corridor Properties.
7. In the Target Mapping window, in the Width or Offset Targets category, click the <None> Object Name next to the Not Found subassembly ► in the pulldown, select Feature lines, survey figures and polylines ► Select from Drawing ► Select the polyline that was drawn at the back of the driveway. ► Enter ► OK
8. Repeat Step 8 in the Target Mapping window for the Found subassembly.
9. Click <None> next to "Driveway Tie In ► In pulldown, select, "Feature lines, survey figures and polylines" ► Select from drawing ► Select the same polyline in steps 7 and 8 ► Enter ► OK. This step will now draw the driveway out to the polyline. The Target Mapping window should look similar to below:
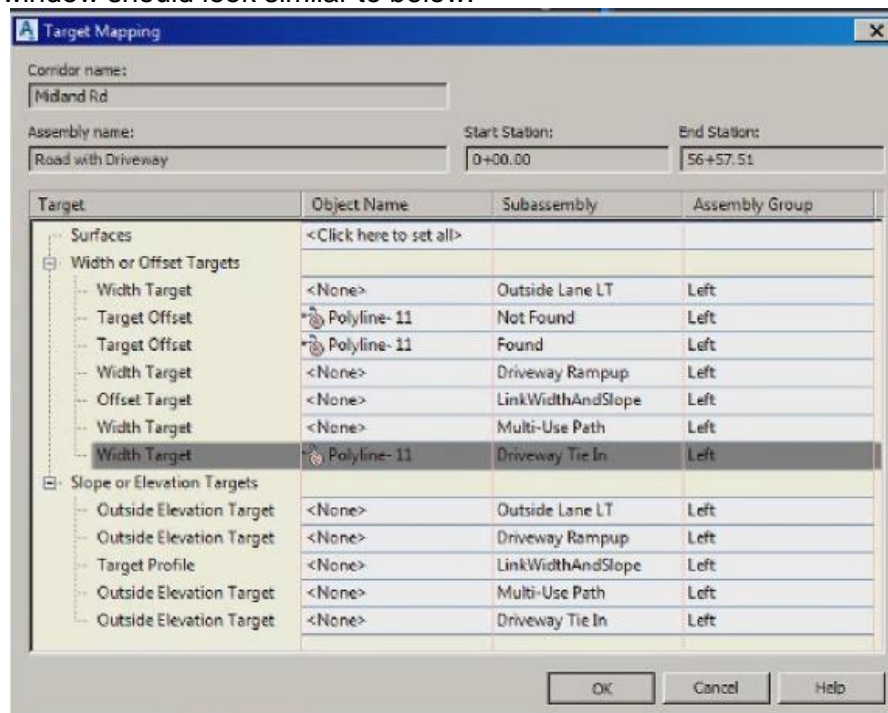

10. The model will build the driveway as shown below. The model will now seek out that polyline and build the depressed curb with the driveway. Since there is not
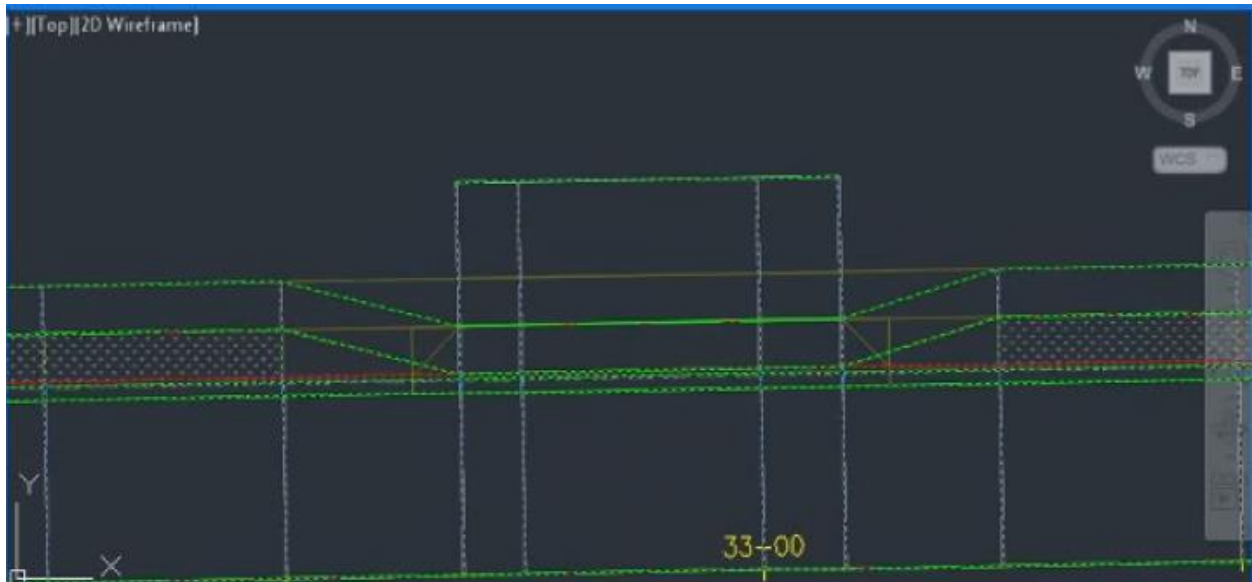
There are a few ways to adjust the model so that the sidewalk gets built across the driveway in the right location. See below. The sidewalk is currently being built along the RED line, but we want the sidewalk to be built across the BLUE line. A quick and easy way to fix this is to edit the frequency by adding in frequency points.
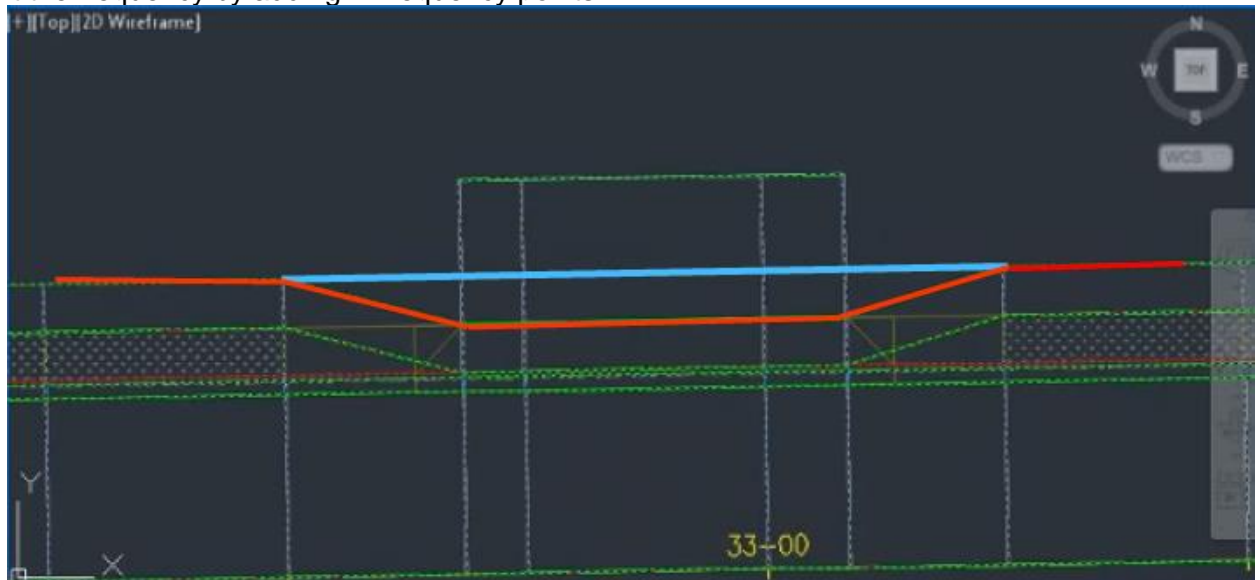
11. Select the corridor ► In the Ribbon, select "Edit Frequency" ► Select the region with the driveway ► Click the plus sign near the bottom ► Pick the two endpoints shown below ► Enter
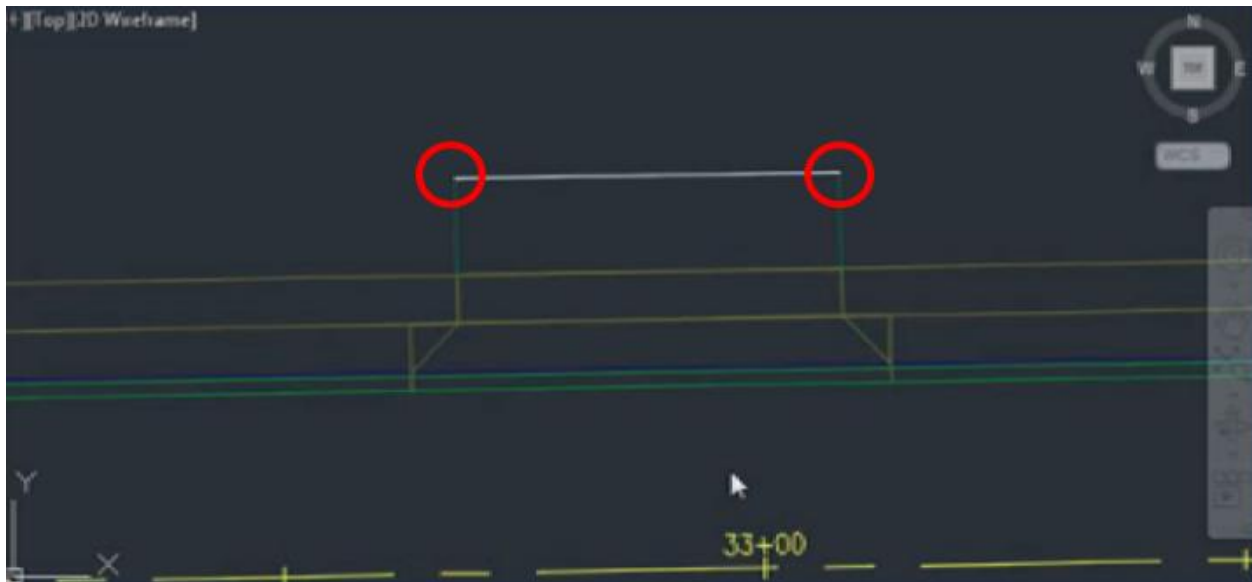
12. Back in the "Frequency to Apply Assemblies" window, there are two stations that appear at the bottom. Edit the smaller station by reducing it by 0.10' and edit the larger station by increasing it by 0.10' ► OK. The model will rebuild and look like the below.
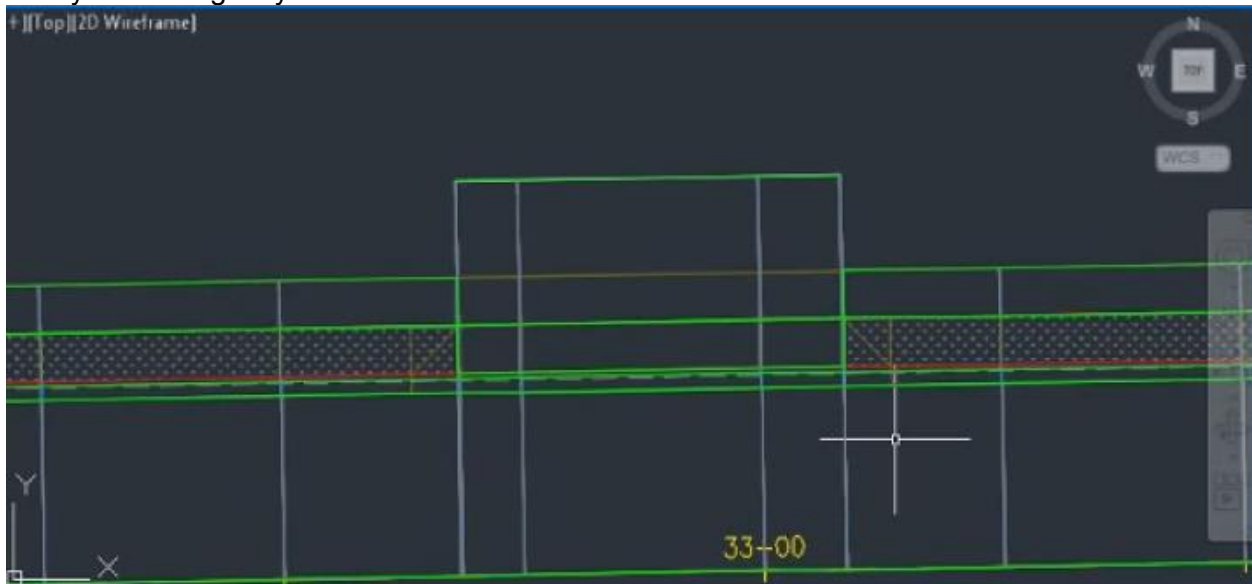
This is what the model will look like as a finished result.

Case Study – Federal Way Link Extension Project in Federal Way, WA
- During our presentation, we will be reviewing a case study and discussing an advanced method for Conditional Subassemblies. This case study will be made available in the recording of the class presentation. Please be sure to watch the recording to pick up on all advanced methods.