Module 5

Templates

(This page is intentionally left blank.)

Module 5 – Templates

About this Practice Workbook...

- The Module 5 - Templates.zip file will be provided for download.
- Extract the zip file to the root C:\
- All files are then automatically extracted here:
  C:\NCDOT Training\Roadway\Module 5 - Templates
- Module 5 - Templates PDF will also be located here.
- This PDF file includes bookmarks providing an overview of the document. Click on the bookmark to quickly jump to any section in the file. You may have to turn on the bookmark function in your PDF viewer, such as Adobe Reader.
- The dataset used throughout this module uses English units and US Survey Feet.
- Each module in this series is self-contained. You can jump to any section and begin the exercises.
- The ***NCDOT_WorkSets.inp*** on your desktop should be set to the following variables:
  - **NCDOT_USE_LOCAL_WORKSETS = L2**
  - **NCDOT_UNIT_TRAINING_WORKSETS = Roadway**
- This training module uses the **DOT-US North Carolina** WorkSpace, **Training-RD_R-2635C** WorkSet and **NCDOT_Roadway** Role. It is very important that you select the correct WorkSpace, WorkSet and Role.
- This workbook and dataset were written with the release of **OpenRoads Designer 23.00.00.129.**
- This workbook and dataset have been upgraded to the release of **OpenRoads Designer 23.00.00.129.**

Module 5 – Templates

Overview

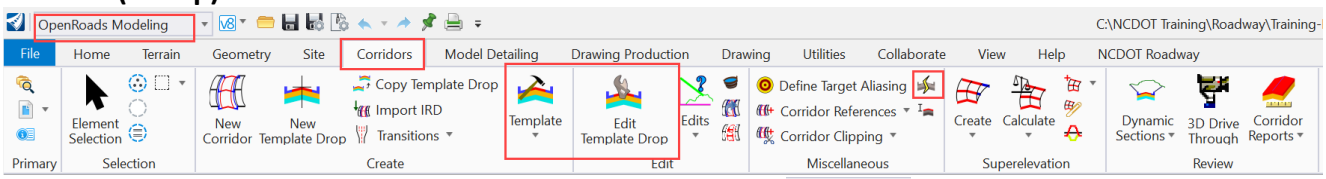
Based on previous training in creating and customizing templates, this workbook is the continuation of corridor templates for OpenRoads Designer (ORD). This module teaches the fundamentals of templates creation and expand the customization techniques used for specific project needs. This training should be used as a guide and aid users in creating and customizing their own templates.

Template Definition

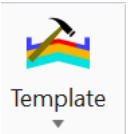
Templates are used to create the 3D model of the road. Similar to typical sections, templates are comprised of **Points** and **Components**. Template points in corridor model produce the **3D linear geometry** or linework while the template components produce the **3D surfaces** or meshes.

Templates are stored in an **InRoads Template Library (ITL),** an XML format file type. Once the template is used in a corridor, templates are saved in the DGN file.

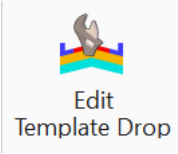
Editing of templates tools can be found under **OpenRoads Modeling (Workflow) >>> Corridor (Tab) >>> Edit (Group).**



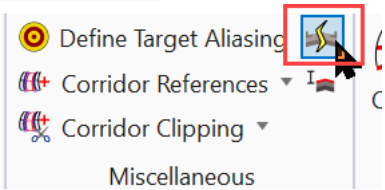
Edit (Create Template) Templates Saved in the **Library** (ITL)



Edit Template Drop Templates Saved in the **Corridor** (DGN file)



Synchronize Changes to Template Library with Corridor Template Drop



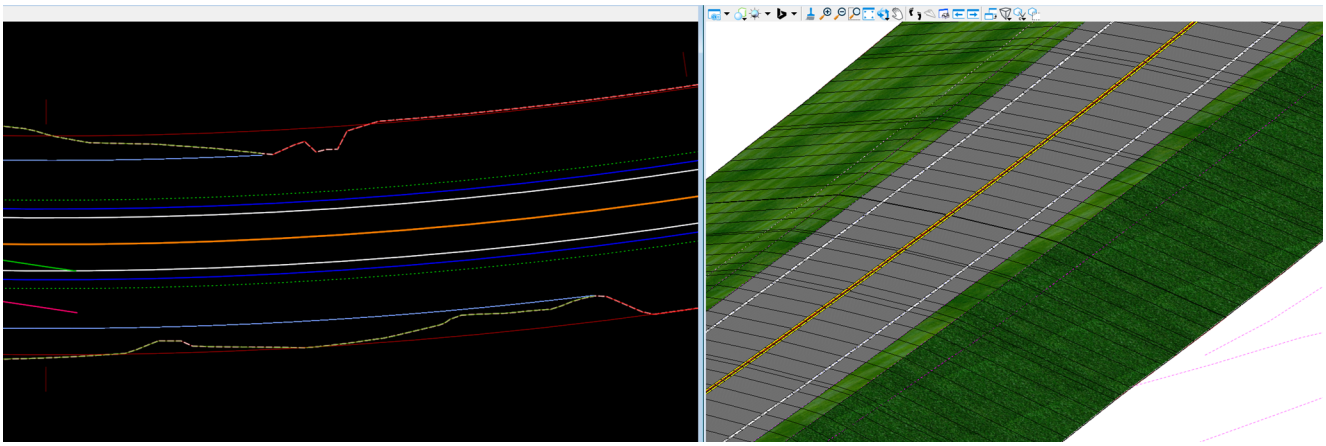
Standard NCDOT Corridor template libraries (ITL) are shown below:

- The **WorkSpace NCDOT Roadway Corridor Template Libraries** are contained in the following folder:
  …\WorkSpaces\DOT-US North Carolina\Roles\NCDOT_Roadway\Standards\Template Library
- The **WorkSet Template Library** should be stored with each WorkSet or Project:
  {TIP No.}\WorkSet\Standards\Template Library
- The **WorkSet Template Library** should be named accordingly:
  NCDOT_RDY_{WorkSet Name}.itl, EX. NCDOT_RDY_R-2635C.itl

Template Points

Template points generate the 2D and 3D linework of the corridor model.

2D Plan and 3D Model Linear Elements Produced by Points



Points are usually connected by a component (member of) to form a segment line or a shape in a template. A single point not connected to a component is called a "Null" point. Even if null points are not connected by a component, they can be constrained to any other points in the template.

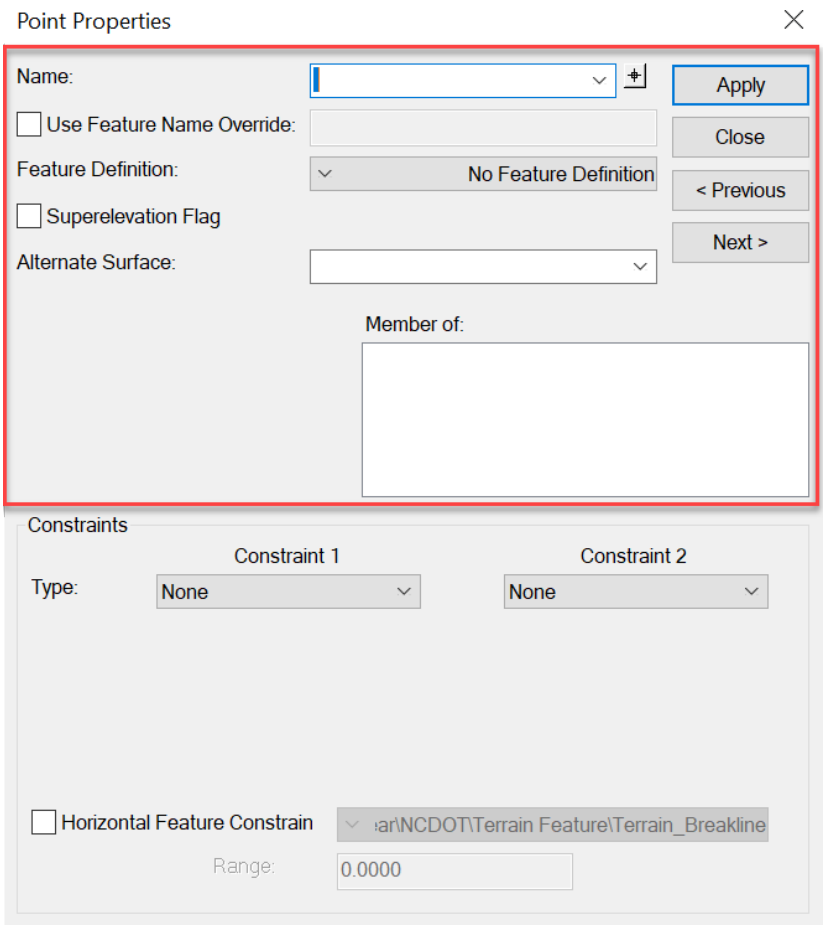
Point name is very important when it comes to **XS Annotation Groups**, specifically when annotating and labeling cross sections. Point names must match with the Annotation Groups or it will not label.

Point Name Convention in Roadway Design

- Capital Letters
- No spaces, use underscore (_) for space.
- Use abbreviations often.
- For mirror points, use (I) inside or (O) outside.
- For points not mirrored, use (LT) left or (RT) right.
- Classify the first term of point names, such as PV (Pavement), SHO (Shoulder Outside), SS (Standard Slope), etc.
- Try not to use special characters such as !,/,(,*, etc. Although +, ~, - and : are used sparingly.
- Limit the use of number as the last character in a point name. The software automatically adds a "1" to the point name or increment it by one (1) if two (2) points have the same name, e.g. 2:1 turns into 2:2 or 2:11.

For a list of standard point names used on a typical road template, see **Appendix A**.

Point Properties



Name:
Name of the point on the template. It appears as a white text. If the name already exists, then a number '1' is added to end of it. If the resource has not been deleted, then an increment of one can be added to the point name which already exist. Each point name on the template must be unique. "Applied Affixes" has an effect on Point Names.

If the point is to be mirrored (left and right side of Centerline), then use the Inside (I) or Outside (O) designation. If the point is not mirrored then the point name may contain an exclusive Left (LT) or Right (RT) designation

Use Feature Name Override:
Each point name must be unique. However, multiple points can share a single name by using this option. When cross sections and the 3D surfaces are created, this feature name override is used instead of the origin name. For example the point name **2:1, 3:1 and 4:1** (because only one successful end condition is display at a time) can have a single override name called "FILL _SLOPE". Points which use the feature name override are shown as a red text in template instead of white.

Feature Definition (Point Properties):
When the models are created, each point in template produces a 3D line work. This line work is referred to as a linear feature. It can be the centerline, edge of travel, paved shoulder lines produced from the template points. The symbology of these line work comes from the feature definition and element templates (level, color, weight, etc.)

Superelevation Flag:
When checked, this point is available to be chosen as the superelevation pivot point or superelevated point. When not checked, the point will not be available for superelevation controls.

Alternate Surface:
By default, the top most surface is usually triangulated as the proposed surface. An alternate surface such as the different pavement layers or a grading surface can also be generated at the same time. To create an alternative surface, give each point a common alternative surface name, e.g. "S_grading".

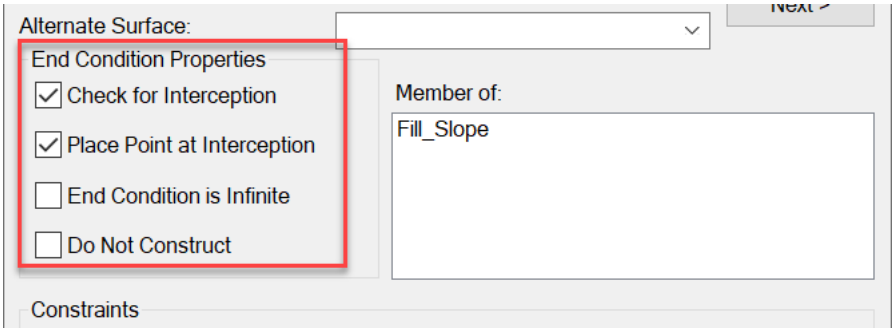
Along with the proposed corridor surface, the alternate surfaces "L-S_grading" can also be created as an option.

Alternative Surfaces can also be used to generate the different pavement layer surfaces, e.g. S_PVMT 1, S_PVMT 2, S_PVMT 3, etc.

Alternate Surface Caveat

Unlike a component property, points cannot trace the existing ground surface (except for two Project to Surface points). When an existing pavement surface is encountered, the alternate surface can only be triangulated (blue line) between points (red dots) and not follow the contour of the existing terrain exactly (black dash).

End Condition Properties:
If the component is an end condition, then the point will have these options.



Check for Interception

Check if the end condition formed from the point intersects the existing ground surface. This is useful when determining the front and back slope of the ditch. The front slope is not checked on since the need to intersect the existing surface is not needed. The back slope of the ditch should be checked on because it is necessary to know if the end condition intersects with the existing surface. Otherwise, the end condition is like any component not considering the existing surface.

Place Point at Interception

If "Check for Interception" is checked on, the option to move the point to the intersection with the existing surface. Otherwise, it stays in its original location.

End Condition is Infinite

If both options above are checked on, the option to intersect the existing surface at any depth or height (infinite) is accomplished with this toggle. Otherwise (when unchecked), the intersection occurs only if the existing surface is within the limits of the points as drawn.

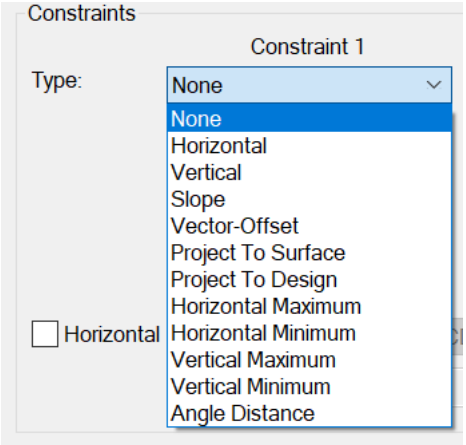
Do Not Construct

When checked on, the end condition segment is NOT drawn to or from the point. For example if **Point 1** and **3** at the beginning and at the end have it unchecked and **Point 2** in the middle is connected have it checked, then the end condition will be drawn from **Point 1** to **3**, even though the middle point is connected as an end condition, it is used only as a reference point. This is commonly used in variable slopes end conditions.

Member of:
The point is part of the component(s) in this list.

Point Constraint Types

Points are ***constrained*** or ruled to each other. There are currently eleven (11) point constraint types available to choose from. The list of point constraint types includes **Horizontal, Vertical, Slope, Vector-Offset, Project to Surface, Project to Design, Horizontal Maximum, Horizontal Minimum, Vertical Maximum, Vertical Minimum,** and **Angle Distance**.



Common Usage for Each Point Constraint Type:

Horizontal – Pavement Width
Vertical – Pavement Depth
Slope – Pavement Slope
Vector-Offset – Continuing Same Subgrade Slope
Project To Surface – Tie to Existing Ground
Project To Design – Subgrade Tie to Side Slopes
Horizontal Maximum –Limits Wedging to Inside EOT or Centerline
Horizontal Minimum – Limits Wedging to Outside EOT
Vertical Maximum – C&G Slope Maximum/Ceiling Elevation
Vertical Minimum – C&G Slope Minimum/Floor Elevation
Angle Distance – Circular (Multi-Sided) Pipe Template

The location (horizontal, vertical, or both/slope) of points is processed in the following order (first on the list is the lowest and last is the highest in priority):

1. Point Constraint (drawn by default at the template level)
2. Parametric Constraint (if different from the default value)
3. Horizontal Feature (Definition) Constraint* (if plan graphics exist)
4. Point Controls**

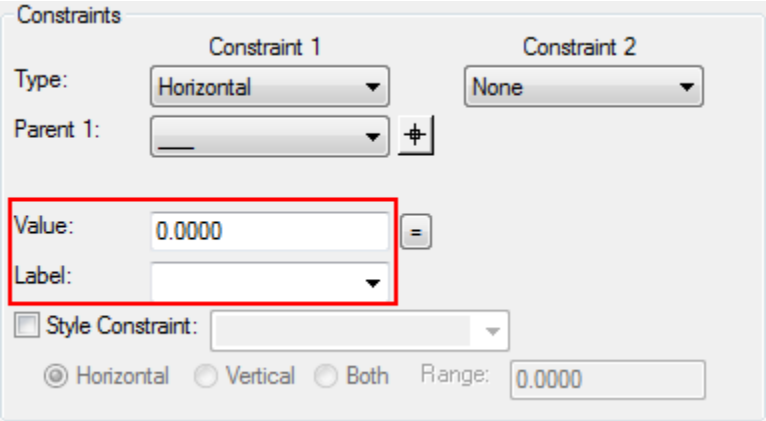
For example, a default 12’ lane width is set at the template level. Change it to 11’ with **Parametric Constraint**. A **Horizontal Feature Constraint** can change it to 14’. Lastly **Point Controls** can be used to override the **Style Constraint**, move EOT point 16’ from the centerline.

* Has a directional seek range left (negative) or right (right) of point.
** Can be prioritized if the same point is being controlled.

Parametric Constraint

Parametric Constraint

Parametric Constraints offer added flexibility in the ability to change the default value for any point constraint type. The **Parametric Constraint Label** (name) and its corresponding default Value are created at the template level under Point Properties and fully both editable and customizable by the user.



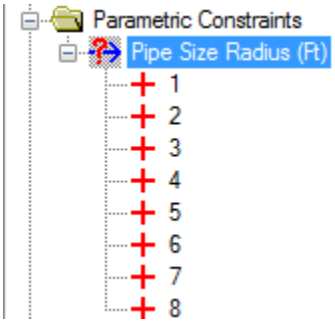
The default parametric constraint value can be changed either at the template level (**Active Template tab** - Parametric Constraints folder) or as a **Corridor Object** (Parametric Constraint dialog box).

For a list of standard parametric constraint labels used on a typical road template, see **Appendix B**.

Mass Editing and Changing Default Value at Template Level vs. in Roadway Designer

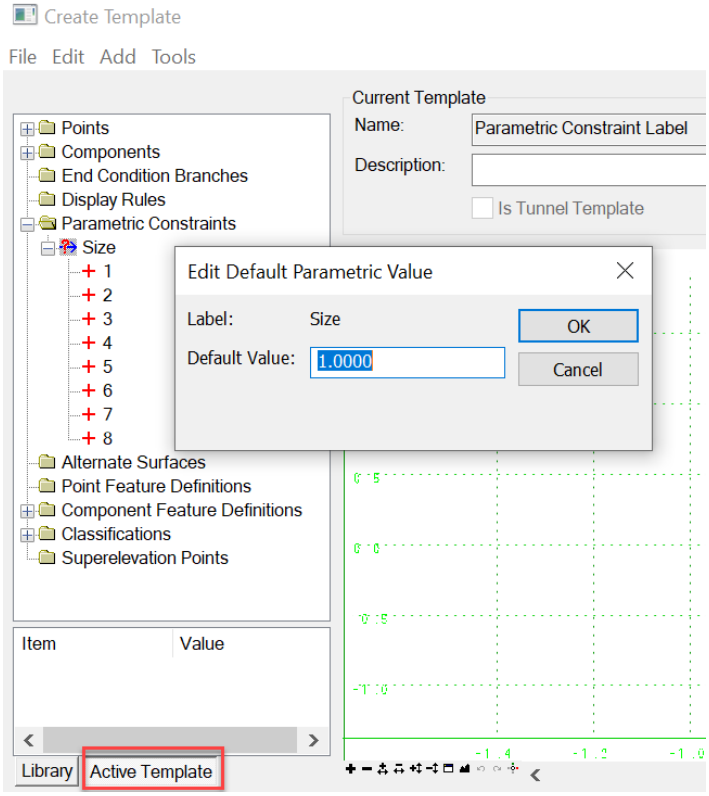
In practice it is common to have one parametric constraint for multiple points.

One Parametric Constraint for Eight Points

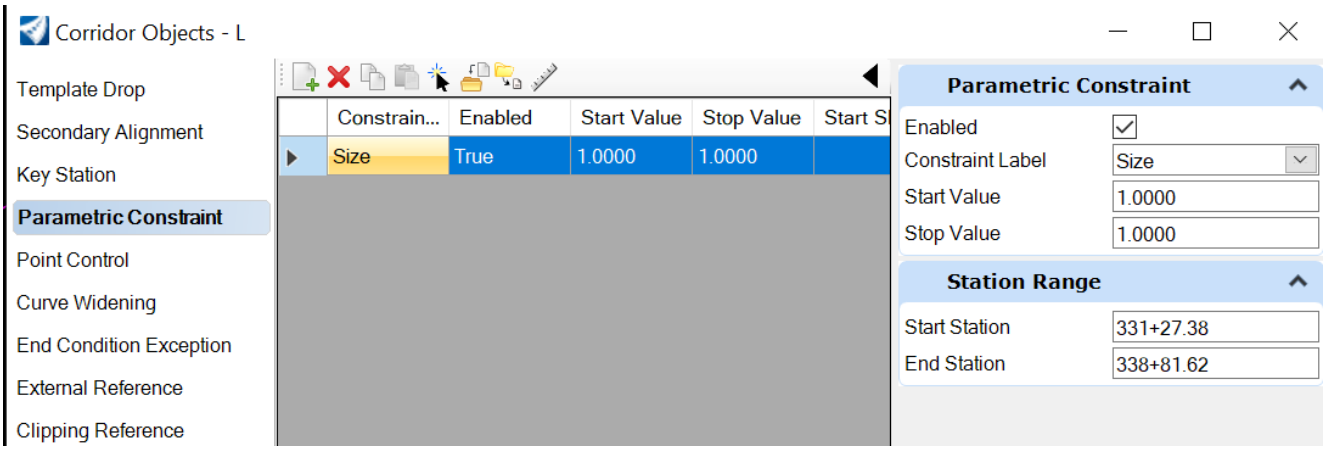


Mass editing of multiple points for a single parametric constraint label should be done at the template level and/or as a Corridor Object, but not at each point (properties) level.

Template Level



Corridor Object



There are advantages and disadvantages with either method and most of the time a combination of the two works best. Below is a table outlining some of the capabilities and limitations of using either method.

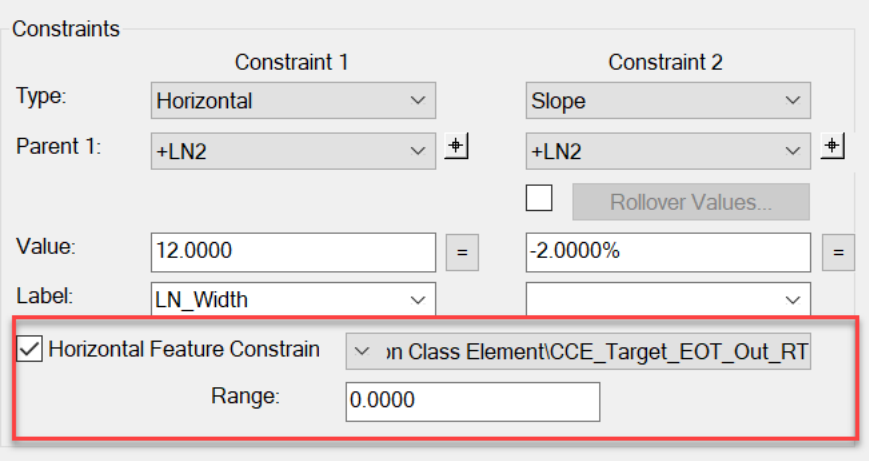
By default any parametric constraints with a horizontal or slope component will have the opposite value when mirrored, e.g. right to left. It is indicated by negative sign (-) on the left side and not displayed as a selectable label. Parametric constraint labels containing only vertical component do not have a negative value when mirrored.

Users may edit or add any parametric constraint labels in their own WorkSet template library.

Horizontal Feature Constrain

The **Horizontal Feature Constrain** field is optional. This is where the user can draw a line or line string in a 2D plan view of a design file using a feature definition. The same feature definition is set here that will move the point horizontally to the location in the design file. Users must add this linear element (feature definition) as a **[Add] Corridor Reference** to enable this function.

The **Range** field is used to specify how far left or right horizontally in reference to the point to search for the linear element on a particular feature definition in the design file. A positive value indicates a search range to the right of the point and a negative value indicates a search range to the left of the point. A search range of 0 (zero) indicates an infinite distance left or right of the point. This is why some features have an exclusive “LT” or “RT” at end with zero as the search range (infinite distance).



A common mistake is drawing something in the design file not in the same direction as the search range. For example, if the search range is set to a positive (right) value, but the linear element is drawn to left of the point, then the search function will not find the linear element.

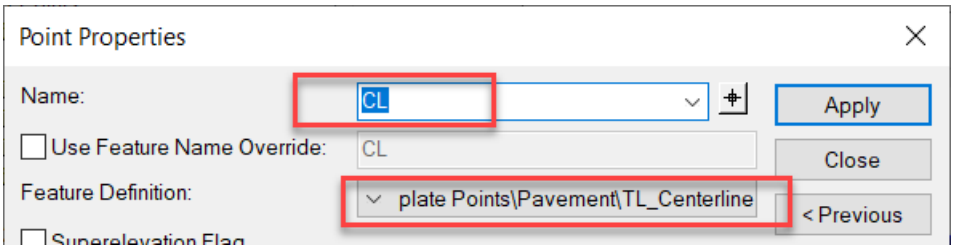
Exercise P1: Basic Point Property Settings

In this exercise we will demonstrate the proper point names used on a template and the feature definition associated with each point.
In the Template Library **Open** the **01 Points\01 Point Basics** folder and select the template **Pavement Layers**.

1. Starting on the top left corner, rename the point and assign a feature definition. Use the table below as a guide.

The feature definition path is “Linear\Roadway\Template Points\Pavement”.

Tips: If the Point Name List has been created in the ITL, when editing the point, select the point name from the drop-down list and the feature definition is automatically assigned.



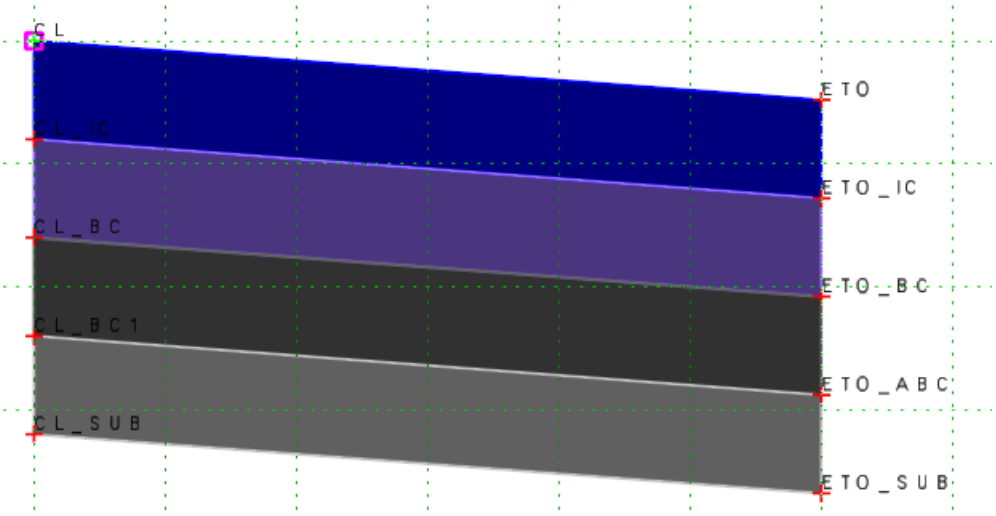
Name	Feature Definition	Description
CL	TL_Centerline	Centerline
CL_IC	TL_Centerline Top Intermediate Course	Centerline Top of Intermediate Course
CL_BC	TL_Centerline Top Base Course	Centerline Top of Base Course
CL_ABC	TL_Centerline Top Aggregate Base Course	Centerline Top of ABC
CL_SUB	TL_Centerline Subgrade	Centerline Subgrade

2. Starting on the top right corner, rename the point and assign a feature definition. Use the table below as a guide.

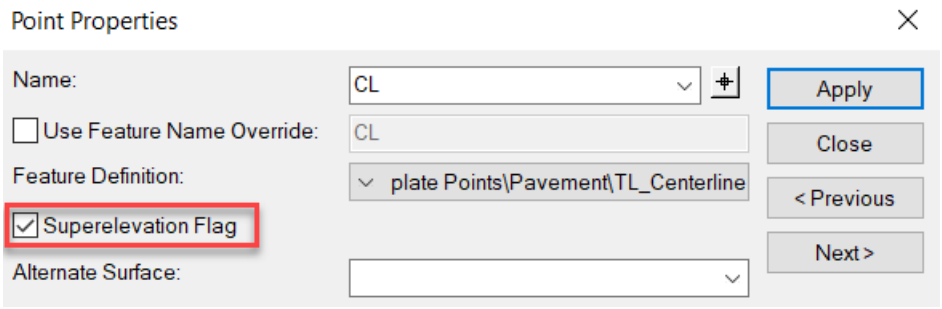
Name	Feature Definition	Description
ETO	TL_Edge of Travel Outside	Centerline
ETO_IC	TL_Edge of Travel Outside Top Intermediate Course	Edge of Travel Outside Top of Intermediate Course
ETO_BC	TL_Edge of Travel Outside Top Base Course	Edge of Travel Outside Top of Base Course
ETO_ABC	TL_Edge of Travel Outside Top Aggregate Base Course	Edge of Travel Outside Top of ABC
ETO_SUB	TL_Edge of Travel Outside Subgrade	Edge of Travel Outside Subgrade

Note the "TL_" prefix in the feature definition name. It stands for **Template Linear**. It signifies the linear element drawn in 3D is created from the template. There is also a "TC_" prefix for **Template Component**.

The finished product should look like the below.

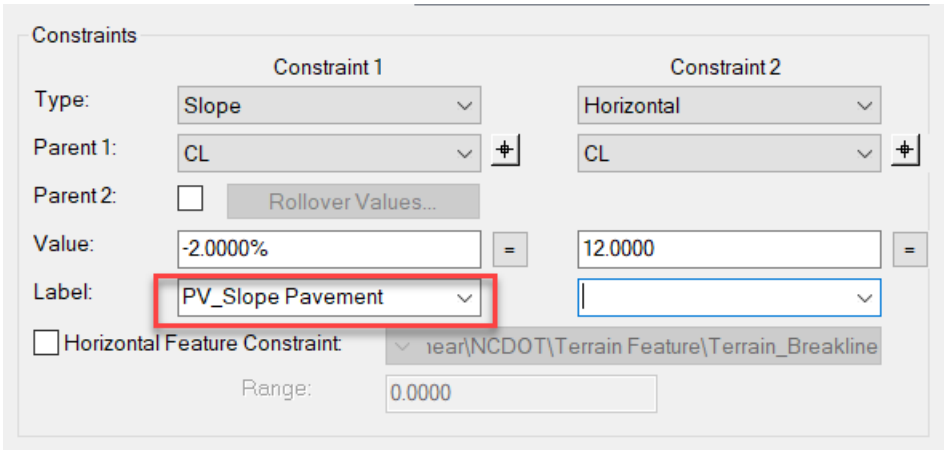


3. Edit the **CL** point and check the **Superelevation Flag** box. **Apply** then **Close**.

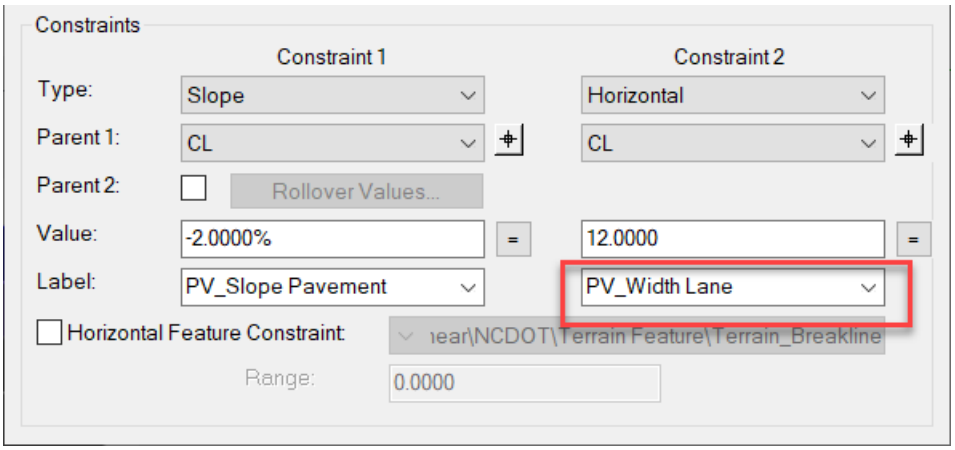


4. Edit the **ETO** point and check the **Superelevation Flag** box.

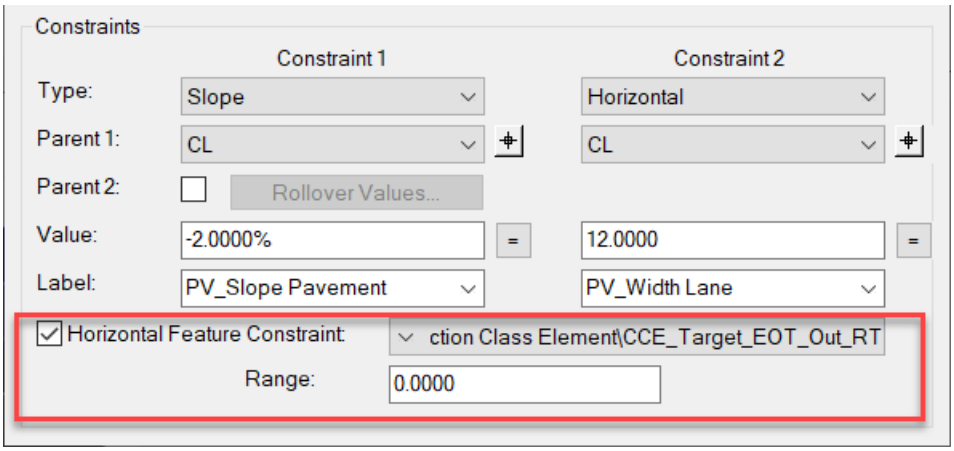
While editing the **ETO** point, key-in the parametric constraint label **PV_Slope Pavement** for the **Slope** constraint type.



5. key-in the parametric constraint label **PV_Width Lane** for the **Horizontal** constraint type.

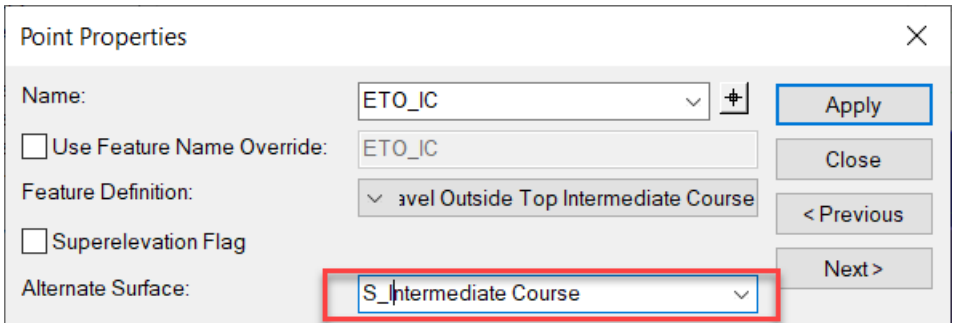


6. Check on **Horizontal Feature Constraint** and select feature definition **Linear\Roadway\ Construction Class Elements\CCE_Target_EOT_Out_RT**. **Range** is set **0**.

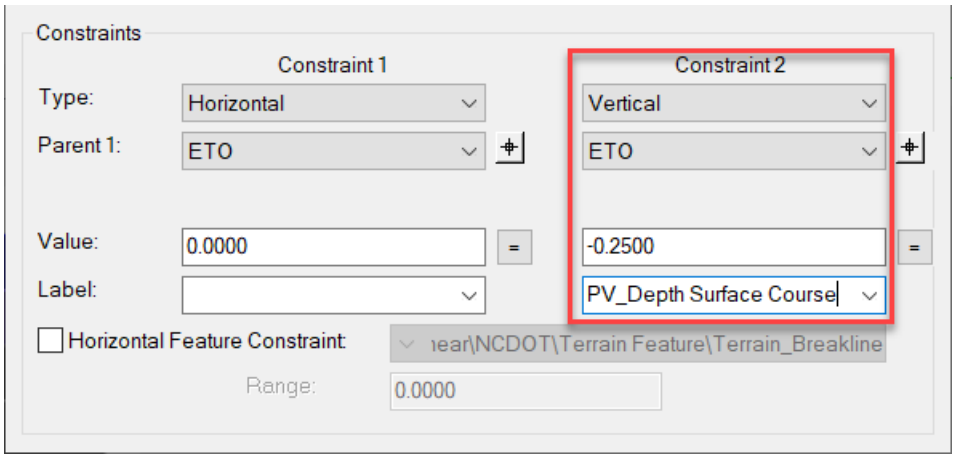


7. **Apply. Close** the Point Properties dialog box and **Save** template library.

8. Edit the **ETO_IC** point. Key-in the Alternative Surface **S_Intermediate Course**

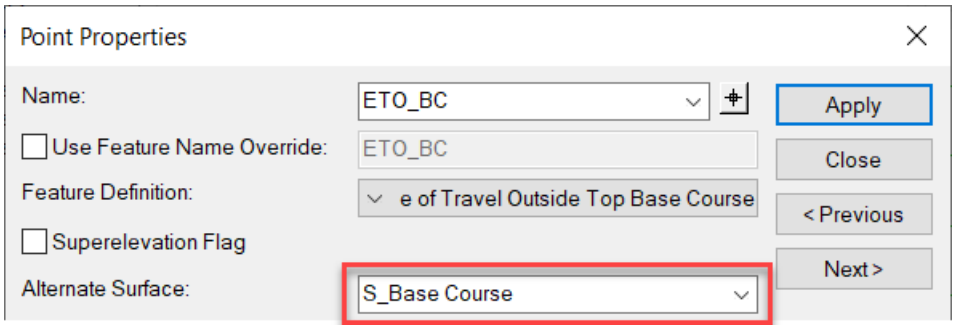


9. Under the **Vertical** constraint type, change the **Value** to **-0.2500** and key-in the **Label PV_Depth Surface Course**.

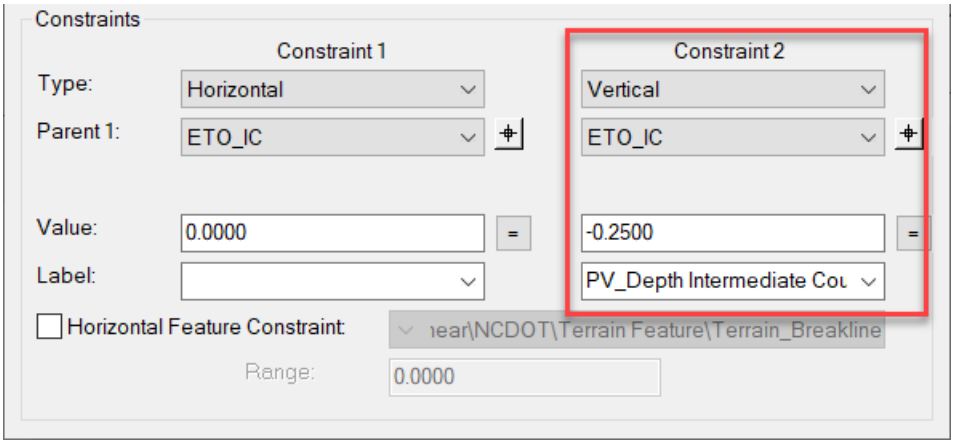


10. **Apply. Close** the Point Properties dialog box and **Save** template library.

11. Edit the **ETO_BC** point. Key-in the Alternative Surface **S_Base Course.**

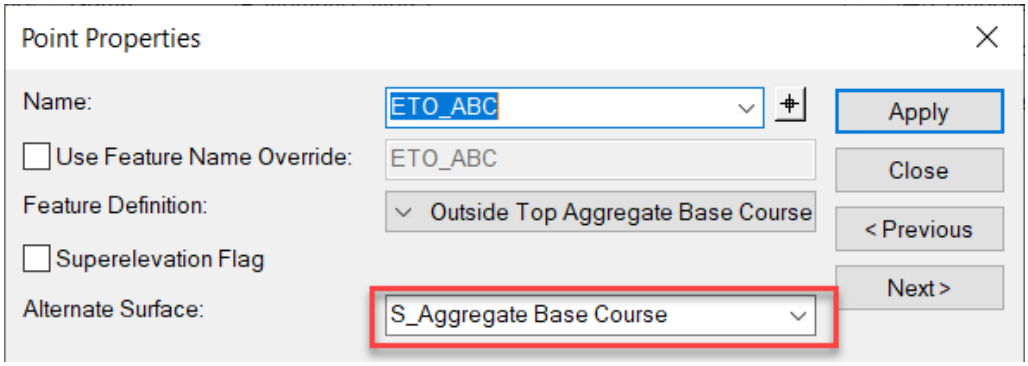


12. Under the **Vertical** constraint type, change the **Value** to **-0.2500** and key-in the **Label PV_Depth Intermediate Course**.

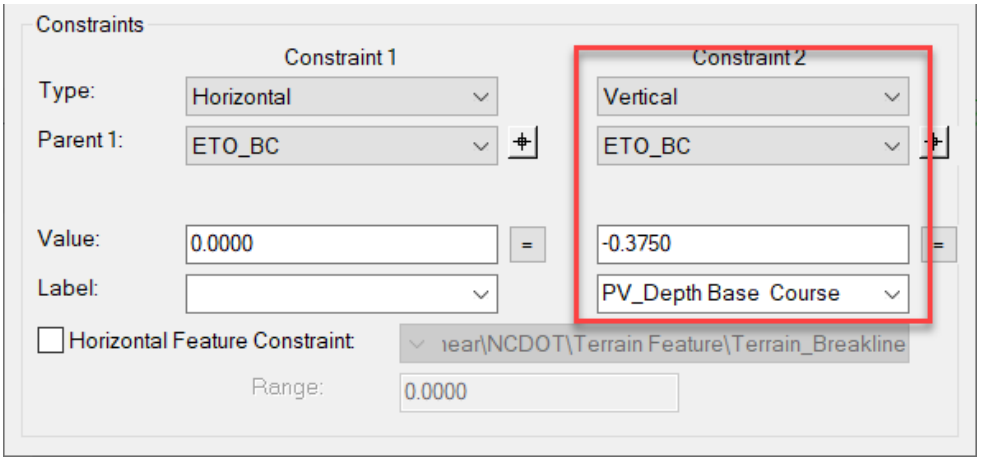


13. **Apply, Close** the Point Properties dialog box and **Save** template library.

14. Edit the **ETO_ABC** point. Key-in the Alternative Surface **S_Aggregate Base Course.**

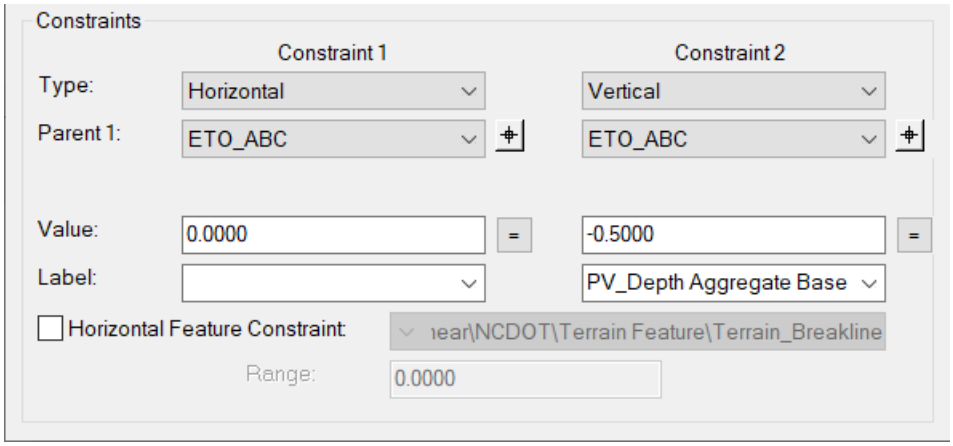


15. Under the **Vertical** constraint type, change the **Value** to **-0.3750** and key-in the **Label PV_Depth Base Course**.



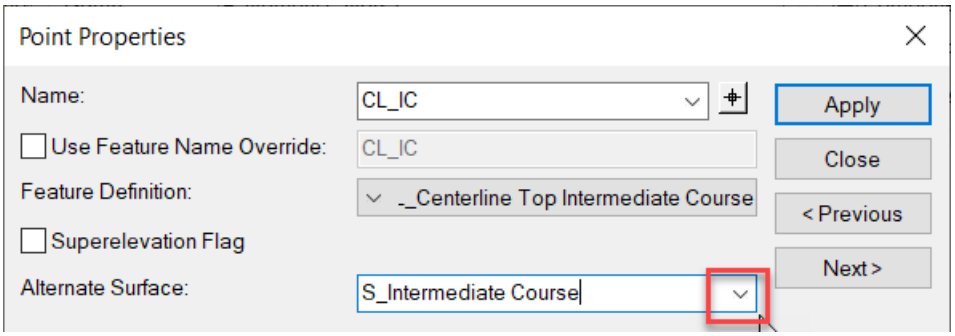
16. **Apply, Close** the Point Properties dialog box and **Save** template library.

17. Edit the **EOT_SUB** point. Under the **Vertical** constraint type, change the **Value** to **-0.5000** and key-in the **Label PV_Depth Aggregate Base Course**.

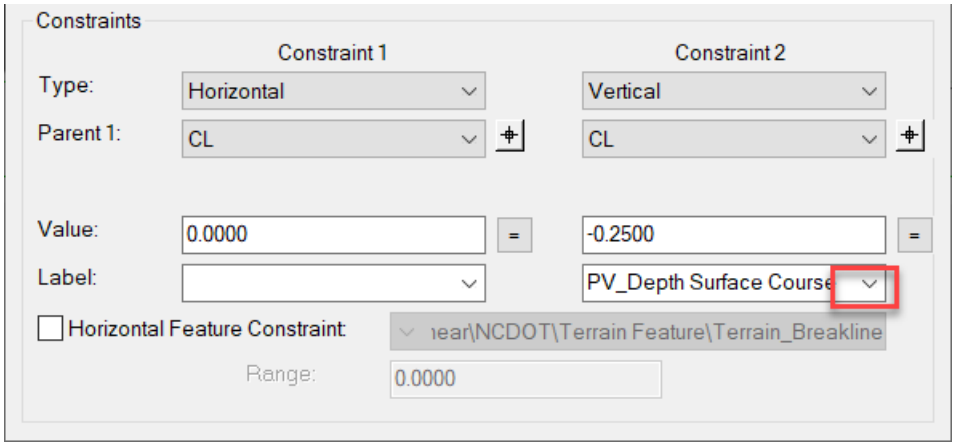


18. **Apply, Close** the Point Properties dialog box and **Save** template library.

19. Edit the **CL_IC** point. Since the label has already been created, select the Alternative Surface **S_Intermediate Course** from the drop-down list.

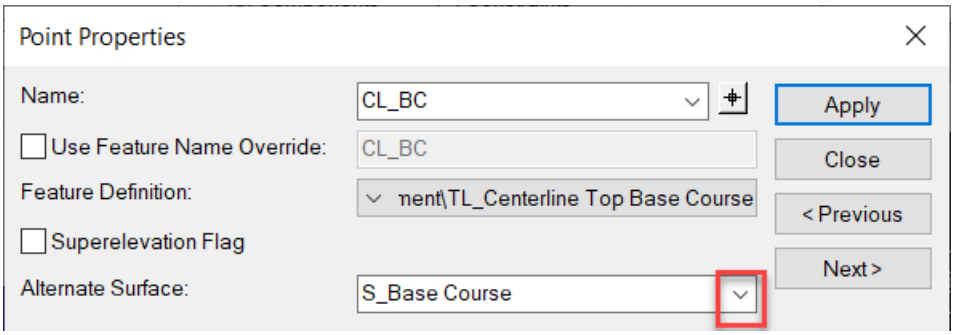


20. Under the **Vertical** constraint type, change the **Value** to **-0.2500** and select the **Label PV_Depth Surface Course**.

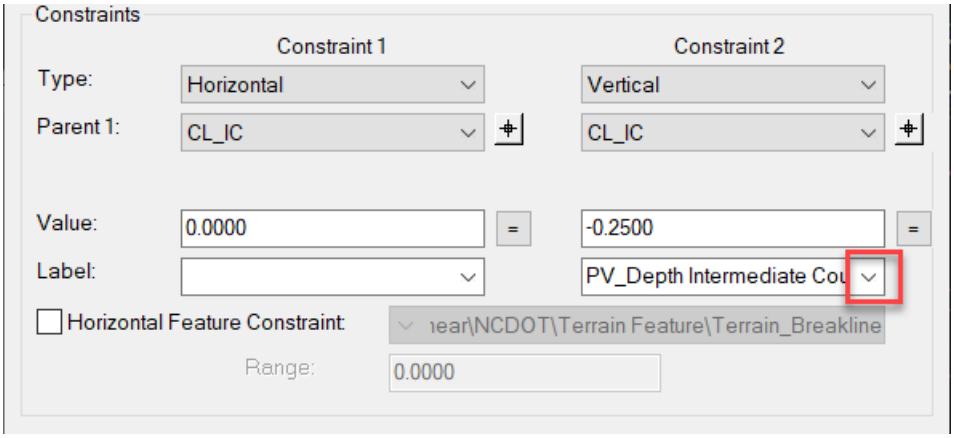


21. **Apply, Close** the Point Properties dialog box and **Save** template library.

22. Edit the **CL_BC** point and select the Alternative Surface **S_Base Course** from the drop-down list.

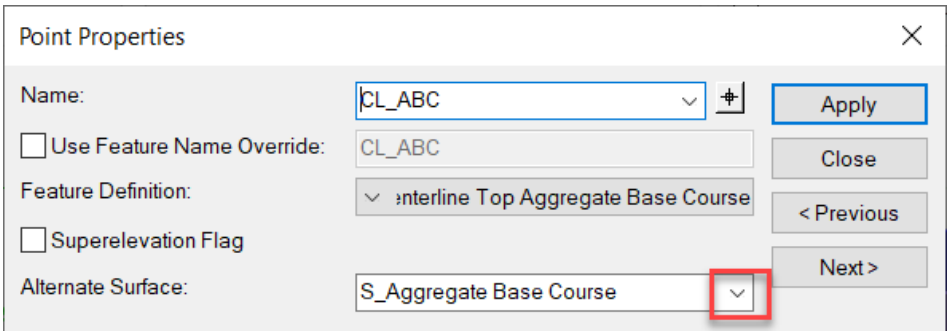


23. Under the **Vertical** constraint type, change the **Value** to **-0.2500** and select the **Label PV_Depth Intermediate Course**.

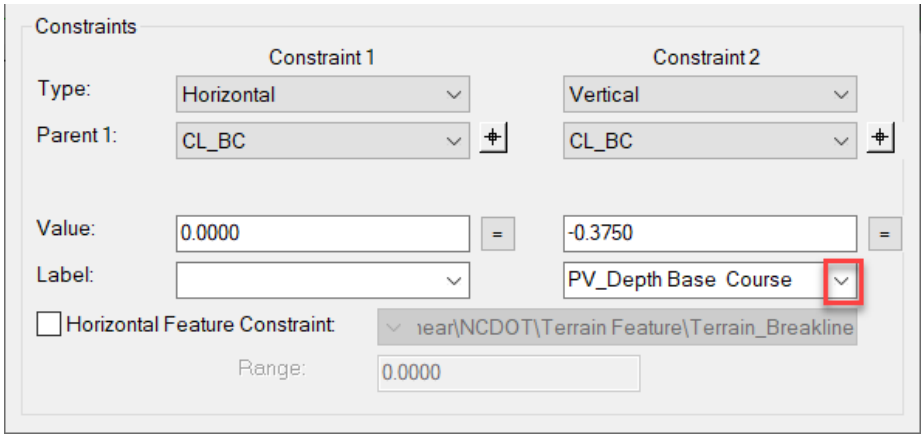


24. **Apply, Close** the Point Properties dialog box and **Save** template library.

25. Edit the **CL_ABC** point and select the Alternative Surface **S_Aggregate Base Course** from the drop-down list.

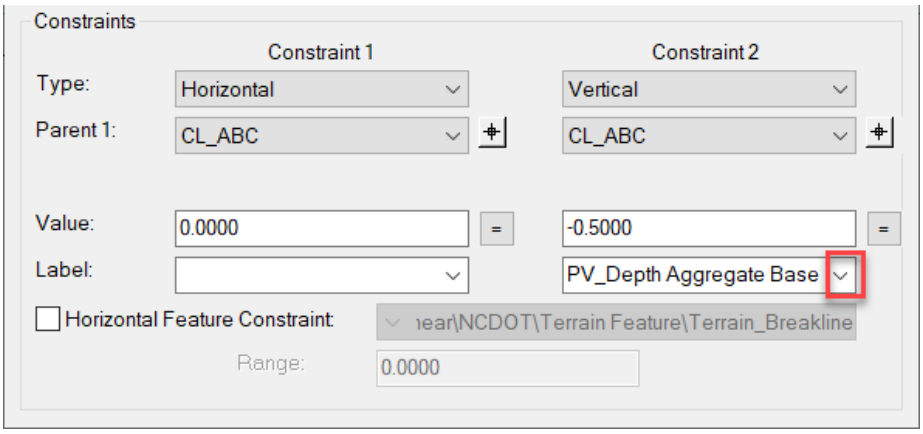


26. Under the **Vertical** constraint type, change the **Value** to **-0.3750** and select the **Label PV_Depth Base Course**.



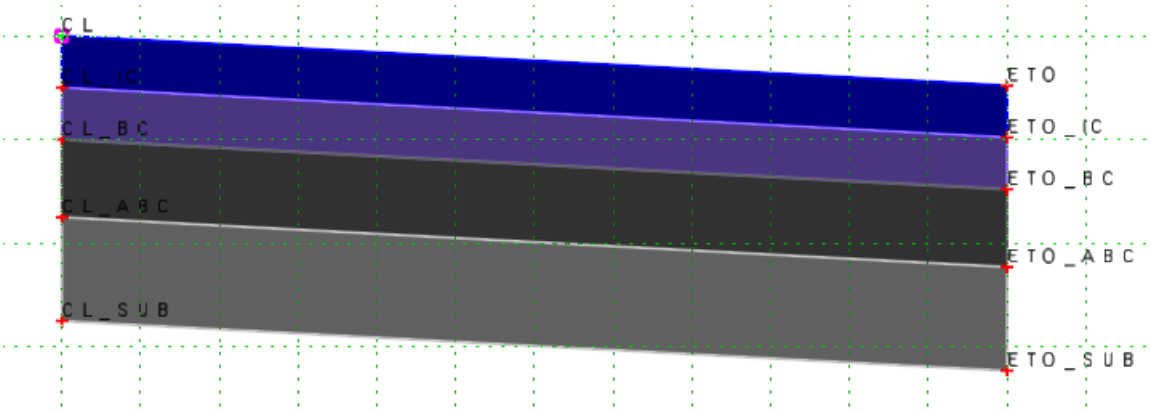
27. **Apply, Close** the Point Properties dialog box and **Save** template library.

28. Edit the **CL_SUB** point. Under the **Vertical** constraint type, change the **Value** to **-0.5000** and select the Label **PV_Depth Aggregate Base Course**.



29. **Apply, Close** the Point Properties dialog box and **Save** template library.

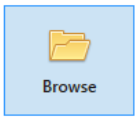
The finished product should look like the below picture.



Exercise P2: Point Constraint Types

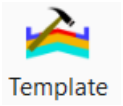
In this exercise we demonstrate the various point constraint types.

1. Open the **Y11 Corridor (CMD) dgn** file.
   Click the browse button and path to the C:\NCDOT Training\Roadway\Training-RD_R-2635C\Module 5 - Templates\Roadway\Design folder and open the R-2635C_RDY_CMD_Y11.dgn file.



2. Open **WorkSet** template library.

Click on the **Template** toolbox. Open WorkSet template library by selecting File> Open… and navigate to and selecting C:\NCDOT Training\Roadway\Training-RD_R-2635C\Module 5 - Templates\WorkSet\Standards\Template Livrary\NCDOT_RDY_Training-RD_R-2635C.itl.

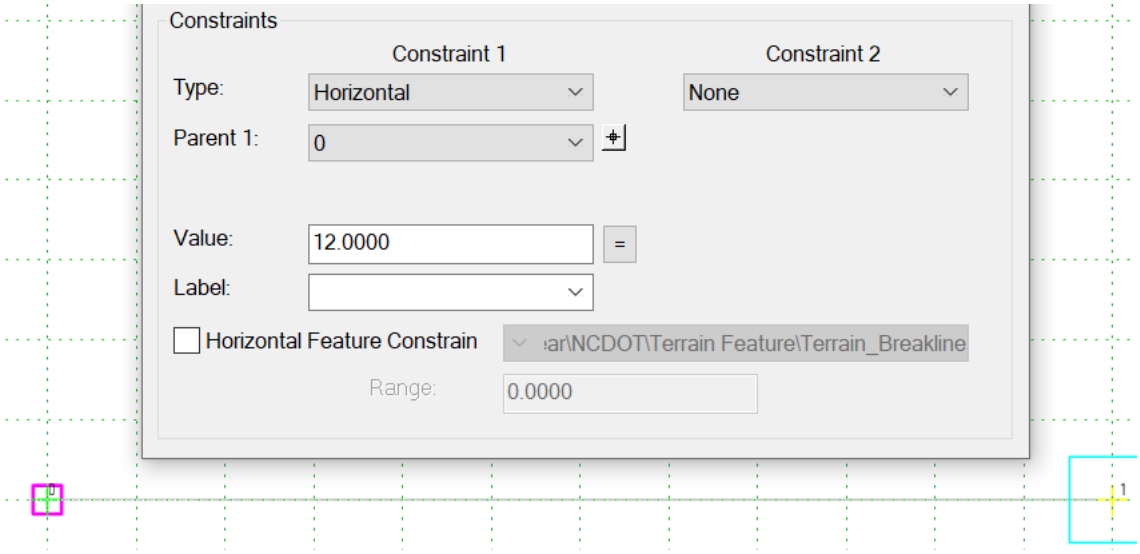


3. Open the **01 Points\01 Point Constraint Types** folder.

A. Horizontal

Constrained a point to a parent point horizontally. A positive value indicates the horizontal distance to the right of the parent point. A negative value indicates the horizontal distance to the left of the parent point.

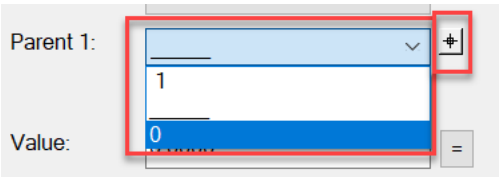
In template 01 Horizontal, constrain Point 1 horizontally 12’ to the right of Point 0 (Parent 1).



Selecting Parent Point

The selection of the parent point to constrain the point to/from can be made in two (2) ways:

- Drop Down List
- Point Picker (graphically)



Zoom with Mouse Wheel

Use the mouse wheel to zoom in or zoom out of the template window.

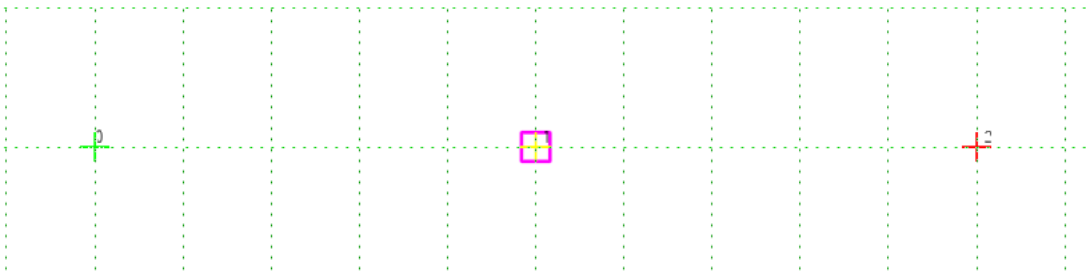
- Use the mouse wheel and Shift key to adjust the vertical exaggeration.
- Use the mouse wheel and Ctrl key to adjust the horizontal exaggeration.

Point Colors

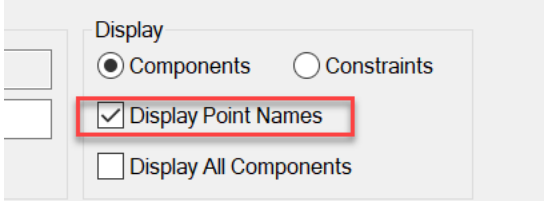
Each point in a template is represented by a plus sign. The point color indicates the state of how it is being constrained.

- Green is an unconstrained point (no constraint).
- Yellow is a partially constrained point (1 out of 2 constrained).
- Red is a fully constrained point (2 out of 2 constrained).

Note that each point can have a maximum of 2 constraints, excluding the Angle Distance point constraint type.



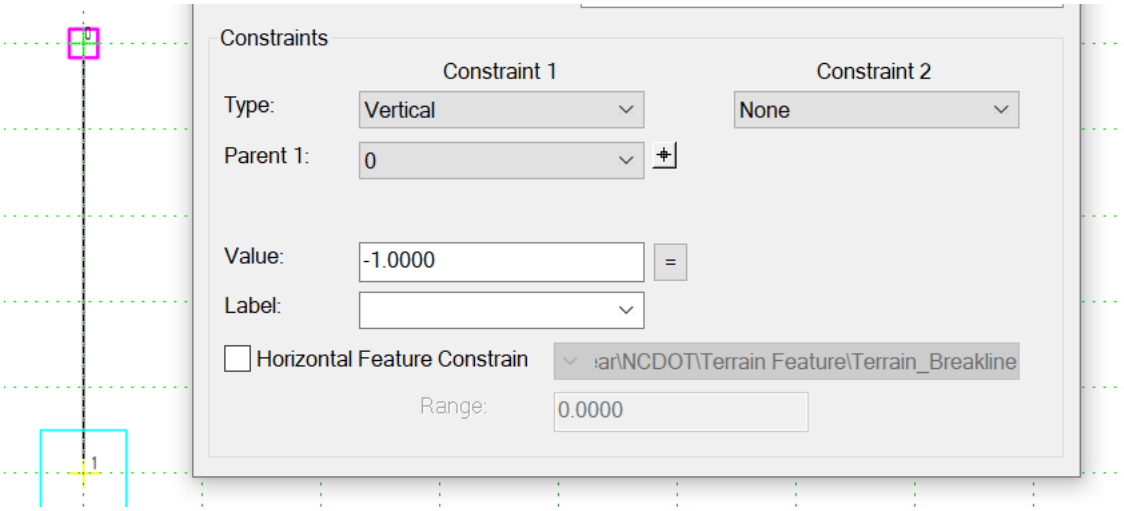
Displaying Point Names (through checkbox option)



B. Vertical

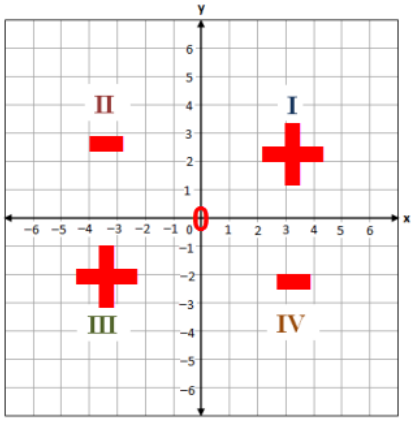
Constrained a point to a parent point vertically. A positive value indicates the vertical distance above the parent point. A negative value indicates the vertical distance below the parent point.

In template **02 Vertical**, constrain **Point 1** vertically 1’ below **Point 0**.



C. Slope

Constrained a point to a parent point with a slope in percent (%). A positive slope value indicates the constrained point is located in quadrant I or III in relationship to the parent point (0). A negative slope value indicates the constrained point is located in quadrant II or IV in relationship to the parent point (0).



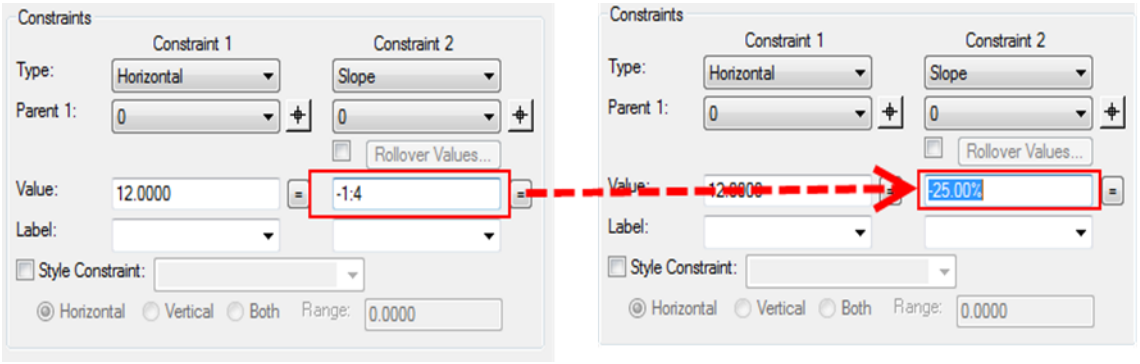
Slope Percentage Format Conversion

Since slope is entered as a percentage value, 0.02 ft/ft (V:H) = 2%. Also side slope 3:1 (H:V) must be entered as 1:3 (V:H) or 33.33% (100/3). Below are common side slopes and their equivalent percentage values.

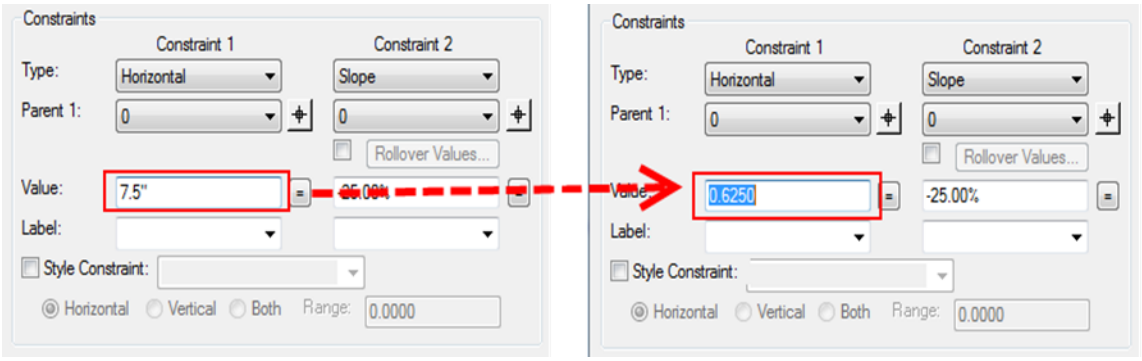
H:V	%		H:V	%
1:1	100		5:1	20
1.5:1	66.67		6:1	16.67
2:1	50		8:1	12.5
3:1	33.33		10:1	10
4:1	25		12:1	8.33

Auto Conversion

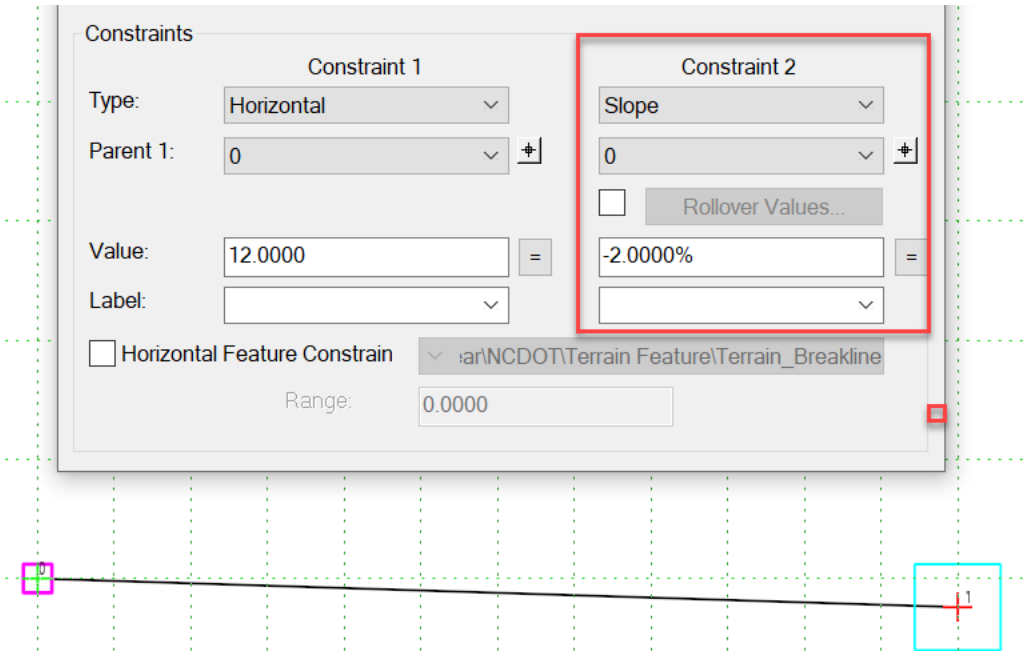
-25% slope could have been entered as **-1:4** (V:H) in the key-in field and hitting **Tab** on the keyboard, the value will automatically change to -25%. Also note that you can key in 0.25 and it can be converted to 25%.



Also values in inches (“) are automatically converted to feet.

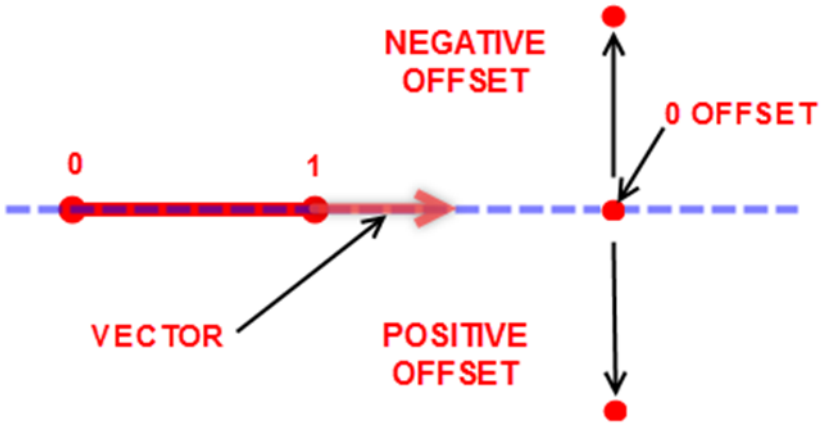


In template **03 Slope**, constrain **Point 1** with a downward slope of -2% to the right of **Point 0**.

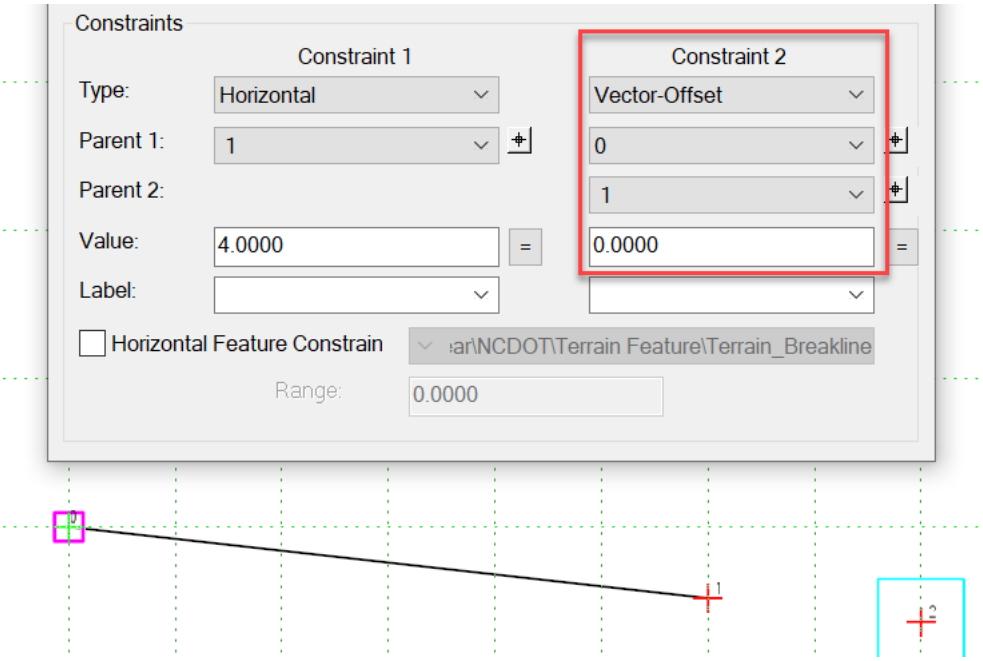


D. Vector-Offset

Constrained a point to the vector (slope) formed by two parent points. A **positive** offset value indicates the distance **right** of the vector. A **negative** offset value indicates the distance **left** of the vector. A value of **zero** indicates the **same vector (slope)** is maintained.



In template **04 Vector-Offset**, constrain **Point 2** to have the same slope as the vector formed by **Point 0** and **Point 1**.



Additional Vector-Offset Notes

The constrained point does not have to exist between the two parent points. Most of the time a Horizontal Constraint (with a vector-offset value of zero) is used with a Vector-Offset Constraint to place it in a confined horizontal location.

Also it does not matter if the vector is derived from the first parent point to the second parent point or from second parent point to first parent point if the value is zero. If the offset value is non-zero, then the order of the parent points determines the value signage, positive or negative offset.

Because Vector-Offset has a vertical component, it should not be used for points with a required constant positive or constant positive or negative value, such as pavement depth. If a positive vector-offset value is set for the right side, when mirrored the left side would have a negative value.

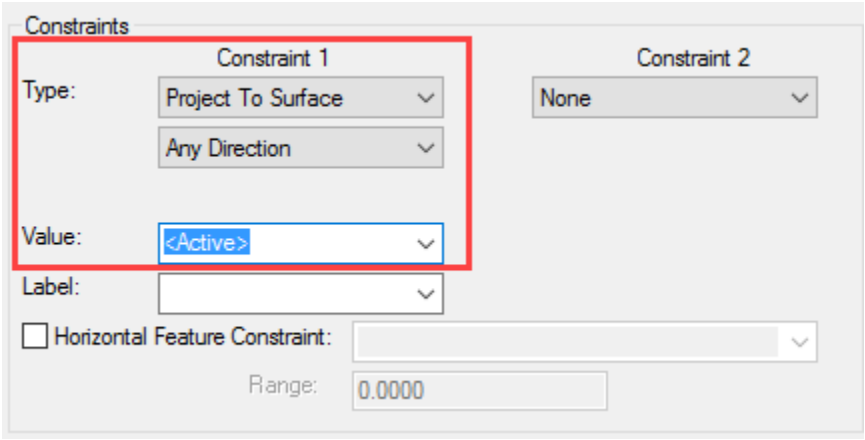
E. Project To Surface

Constrained a point to intersect to a surface (such as the existing ground surface). It is required that another Point Constraint Type such as Horizontal or Slope be used in combination to determine the direction/angle of intersection.

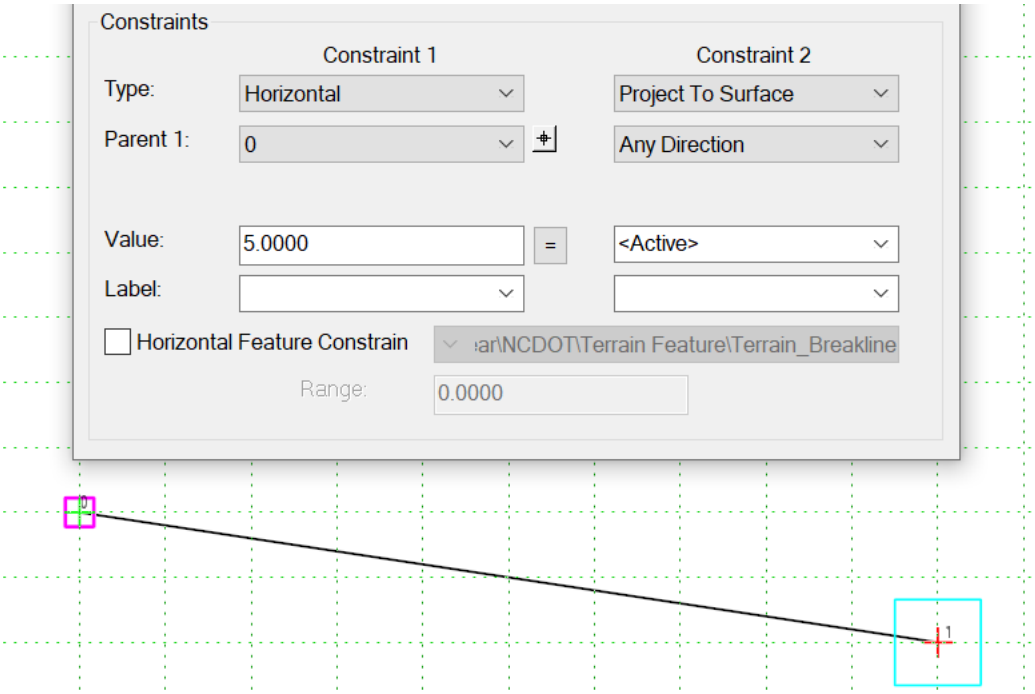
Active Surface Tie and in "Any Direction"

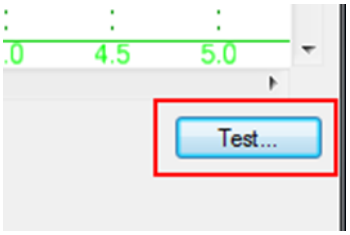
Although there are options to project to the Active, Default, or to a specific surface (existing ground TIN, DTM, Terrain Model, etc.), the Active surface as defined by the active corridor should be used in most cases.

"Any Direction" is mostly used in case more than one intersection point is found.

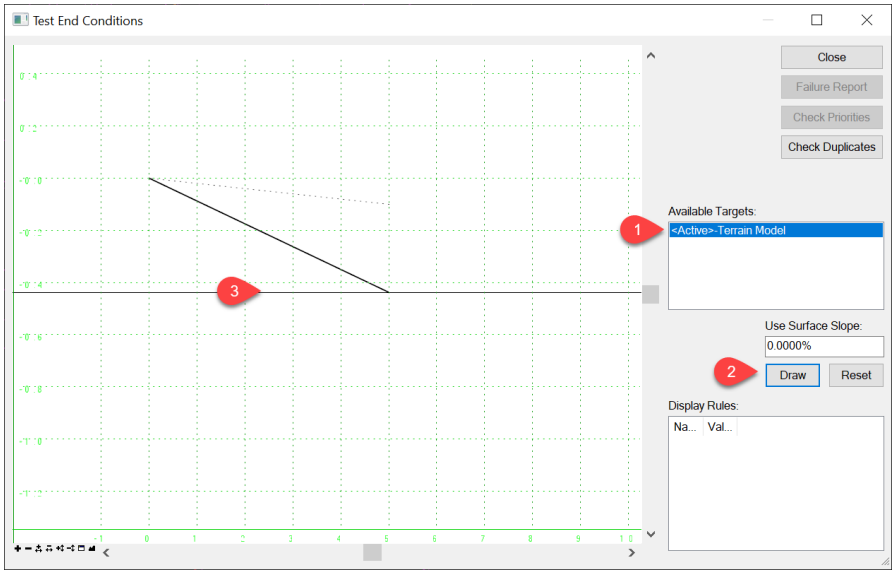


In template **05 Project To Surface**, constrain **Point 1** five feet horizontally to the right of **Point 0**. Also constrain **Point 1** to project to the active surface at any direction. Use the **Test** button to see how the template behaves while moving the existing ground up and down the screen.





(Lower Right Corner)

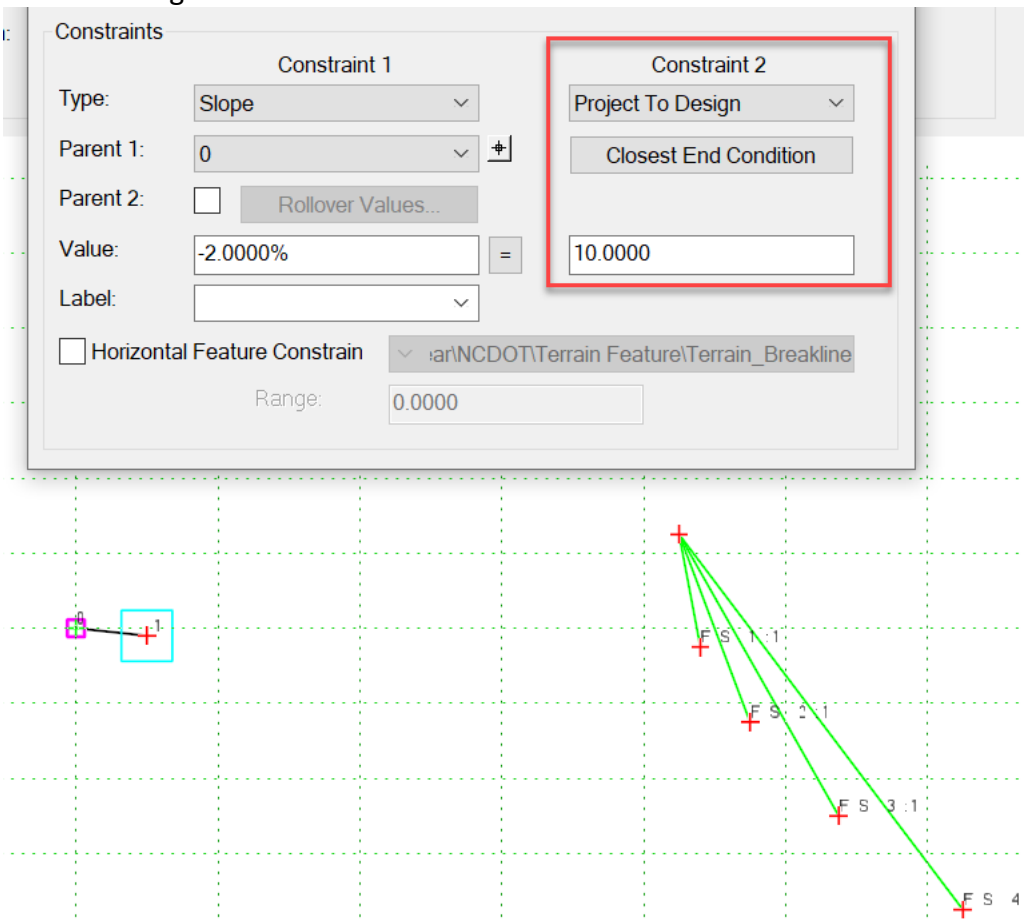


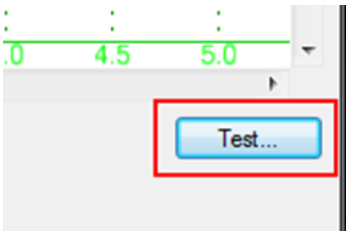
F. Project To Design (Closest End Condition or Component)

Constrained a point to intersect to the nearest active end condition branch or component. A positive value indicates a search range distance to right of the constrained point. A negative value indicates a search range distance to left of the constrained point. A zero value indicates an infinite search range distance to the left or right of the constrained point (not recommended in most cases).

It is required that another Point Constraint Type such as **Vector-Offset** or **Slope** be used in combination to determine the direction of the intersection projection.

In template **06A Project To Design-EC**, constrain **Point 1** with a slope of -2% from **Point 0**. Also constrain **Point 1** to intersect the nearest active end condition with a search range of 10’ to the right of it. Use the **Test** button to see how the template behaves while moving the existing ground up and down the screen. Note the failure when the intersection point is out of the search range distance.





(Lower Right Corner)

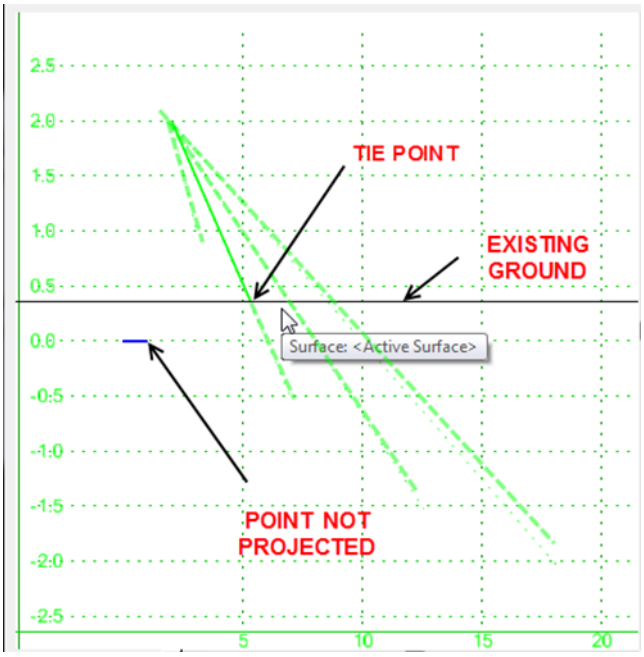
Failure to Project

The Fill Slope 1:1 failed to project because the end condition ended before the intersection point.

The Fill Slope 4:1 failed to project because it exceeded the 10’ search range.

Warping the Subgrade

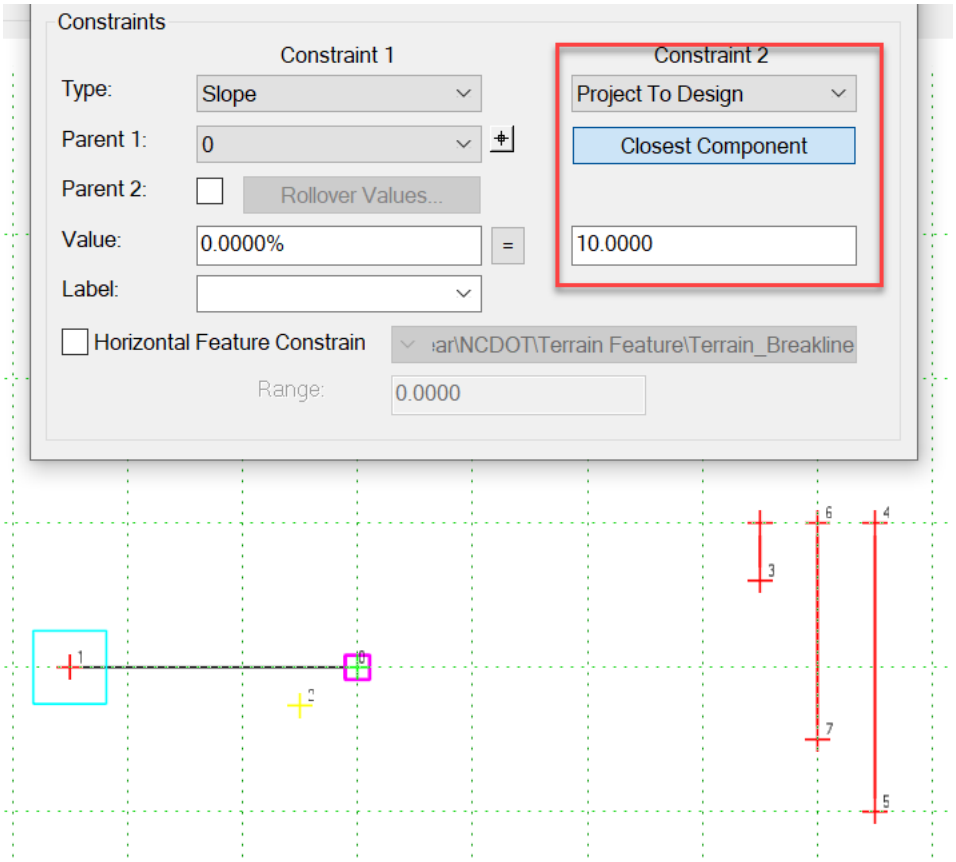
In the above exercise, note that Fill Slope 1:1 and part of Fill Slope 2:1 does not intersect with Point 1. This is due to the end condition branch tie to existing ground above the projected intersection location. The location of Point 1 remains as drawn in the template (does not move).

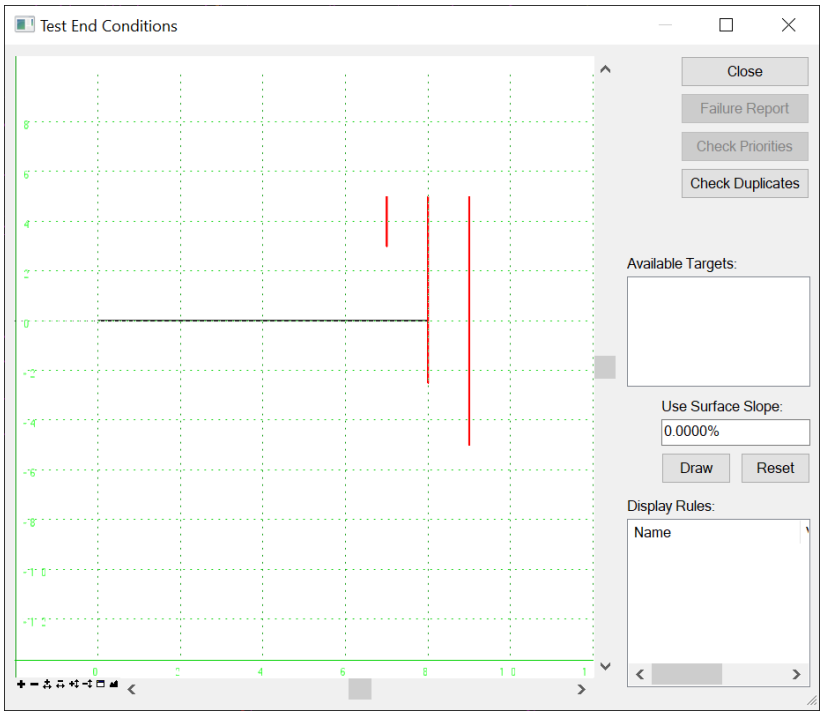


The subgrade daylight points in the template library are constrained with a Project to Design constraint. If a physical projected intersection point cannot be determined, the subgrade line

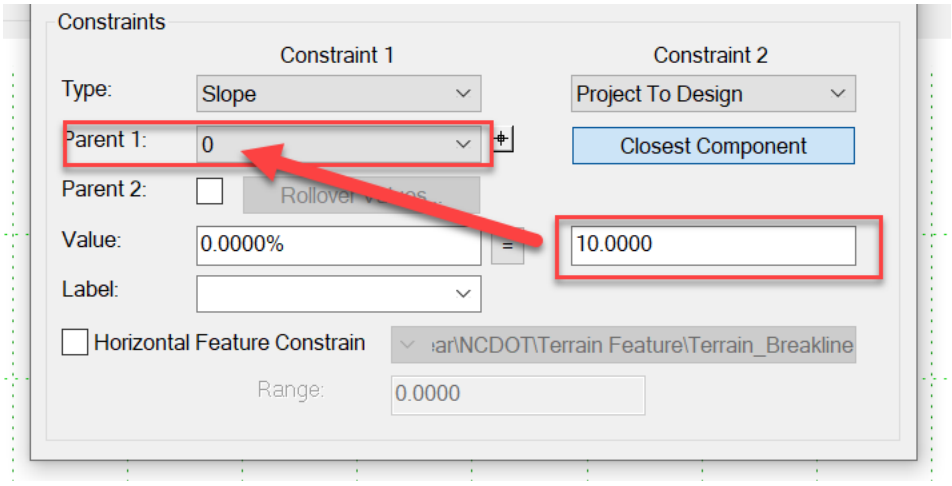
appears “bent” or skewed (as drawn by default in the template). The recommendation is to warp the subgrade manually in the cross section XSC for earthwork. This can be accomplished with **Edit Station** in a Corridor function. If most cross sections do not have the subgrade intersection with the side slopes, then consider permanently merging the subgrade daylight point with the end condition tie point. See the “Parent-to-Child Relationship” and “End Condition Components” sections for further details.

In template **06B Project to Design-C**, constrain **Point 1** to intersect the closest component with a search range of 10’ to the right. Use the **Test** button to see how the template behaves.





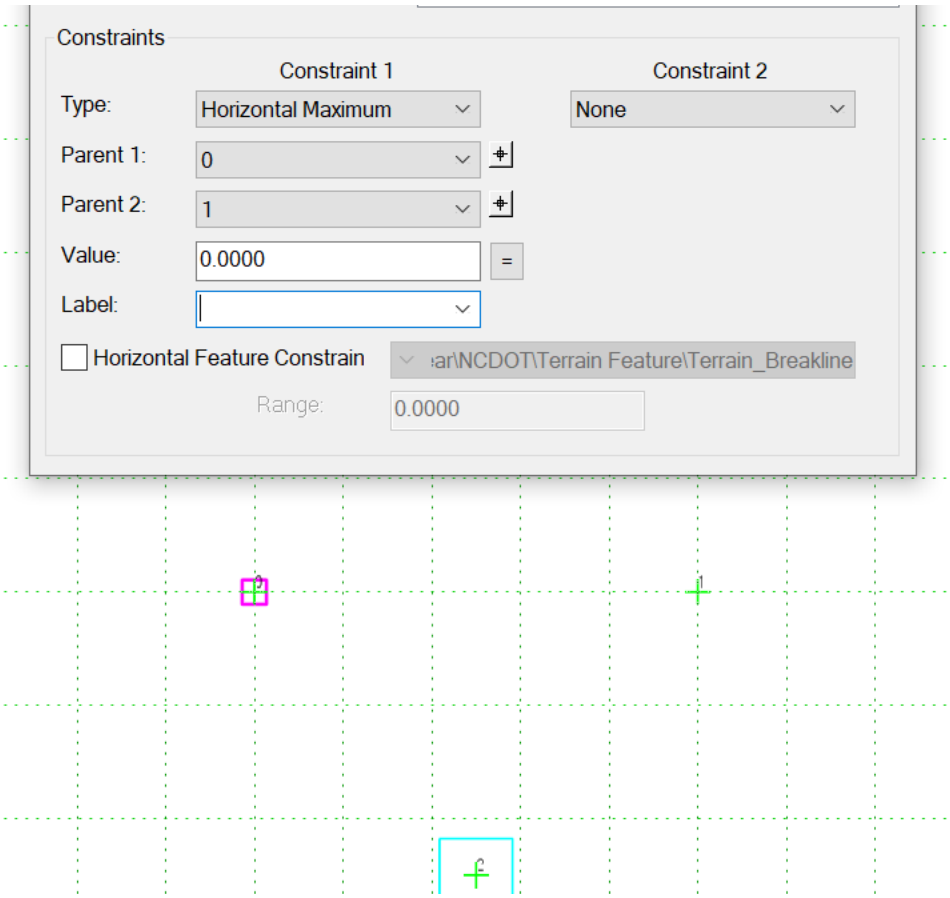
Based on the test results, the **search range** value entered as "10" is not from **Point 1**. It is taken from **Point 0** (Constraint 1).

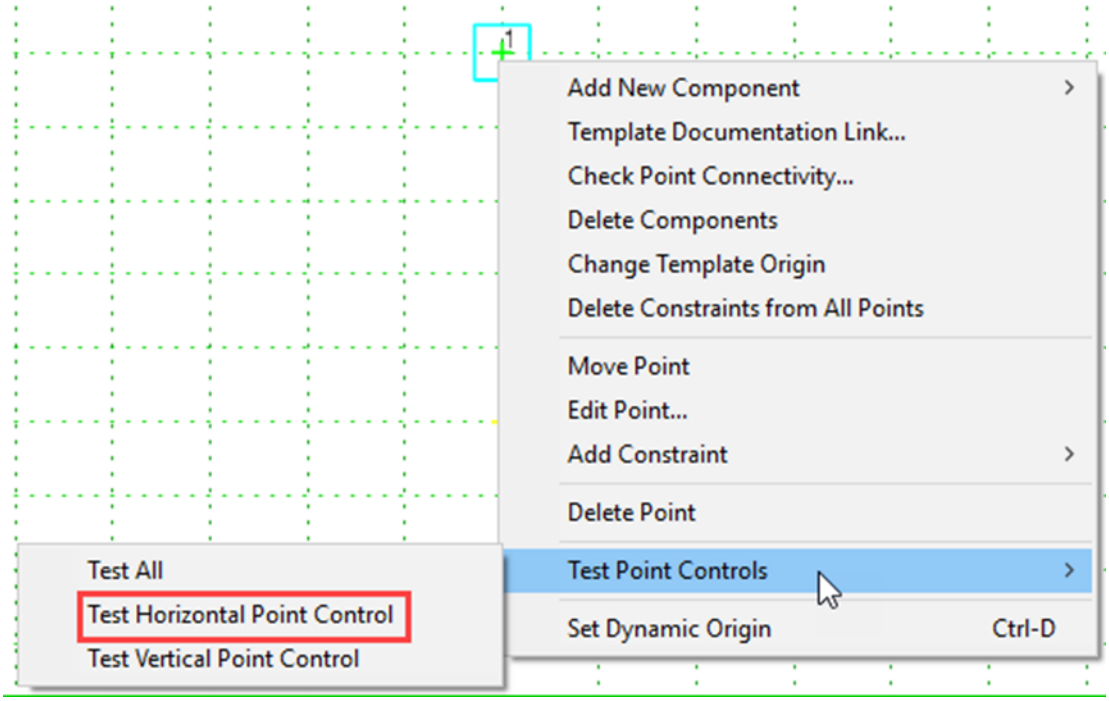


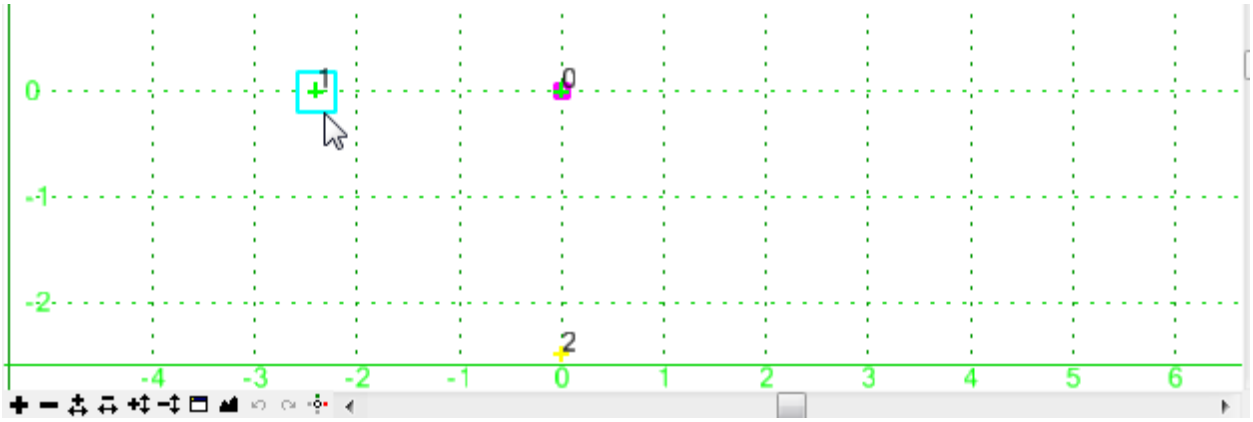
G. Horizontal Maximum

Compare two (2) points and constrain a point horizontally to the right-most (X maximum) parent point. A horizontal offset distance value can be applied to both parent points.

In template **07 Horizontal Maximum**, constrain **Point 2** horizontally to follow the right-most location of Point 0 or **Point 1**. Afterwards, right click on **Point 1** and select **Test Point Controls >>> Test Horizontal Point Control** and move **Point 1** across and over to the left of **Point 0**. Note the horizontal location of **Point 2** before and after the crossover occurs.



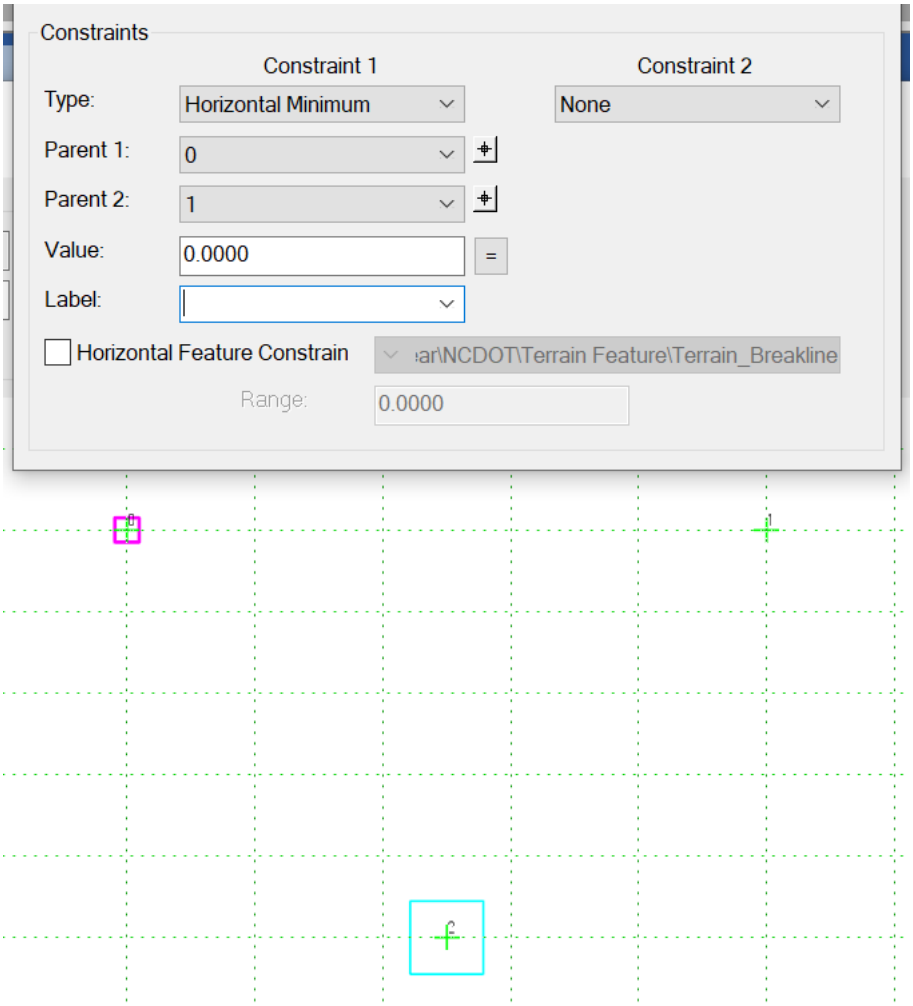

1
Add New Component
Template Documentation Link...
Check Point Connectivity...
Delete Components
Change Template Origin
Delete Constraints from All Points
Move Point
Edit Point...
Add Constraint
Delete Point
Test Point Controls
Set Dynamic Origin
Ctrl-D
Test All
Test Horizontal Point Control
Test Vertical Point Control

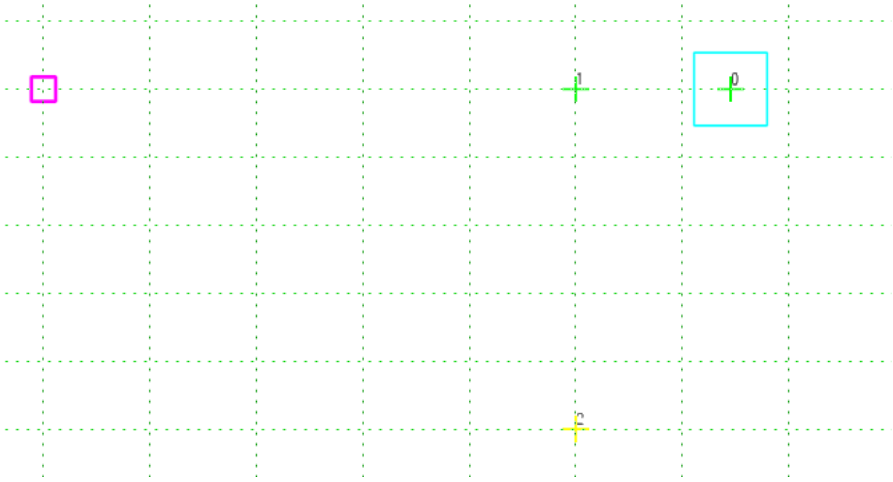

0
-1
-2
1
0
2
-4
-3
-2
-1
0
1
2
3
4
5
6

H. Horizontal Minimum

Compare two (2) points and constrain a point to the left-most (X minimum) parent point. A horizontal offset distance value can be applied to both parent points.

In template **08 Horizontal Minimum**, constrain **Point 2** horizontally to follow the left-most location of **Point 0** or **Point 1**. Afterwards, right click on **Point 0** and select **Test Point Controls >>> Test Horizontal Point Control** and move **Point 0** across and over to the right of **Point 1**. Note the horizontal location of **Point 2** before and after the crossover occurs.

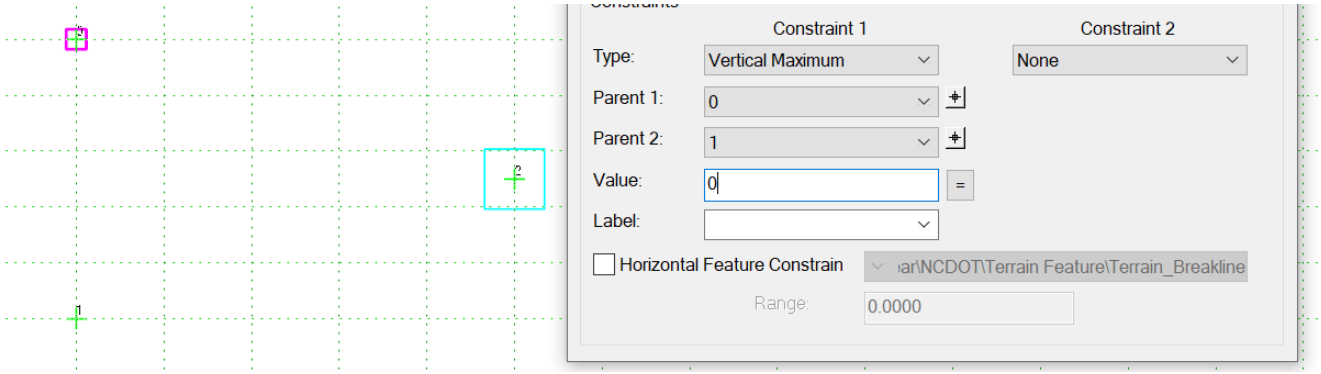


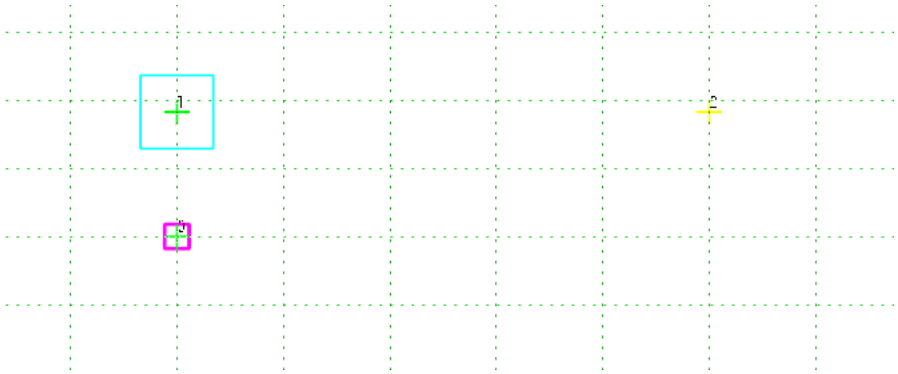

1
0
2

I. Vertical Maximum

Constrained a point vertically between two parent points. The highest (Y maximum) parent point is used as the reference point. A vertical offset distance value can be applied to both parent points.

In template **09 Vertical Maximum**, constrain **Point 2** vertically to follow the higher location of either **Point 0** or **Point 1**. Afterwards, right click on **Point 1** and select **Test Point Controls >>> Test Vertical Point Control** and move **Point 1** up and above **Point 0**. Note the vertical location of **Point 2** before and after the switch occurs.

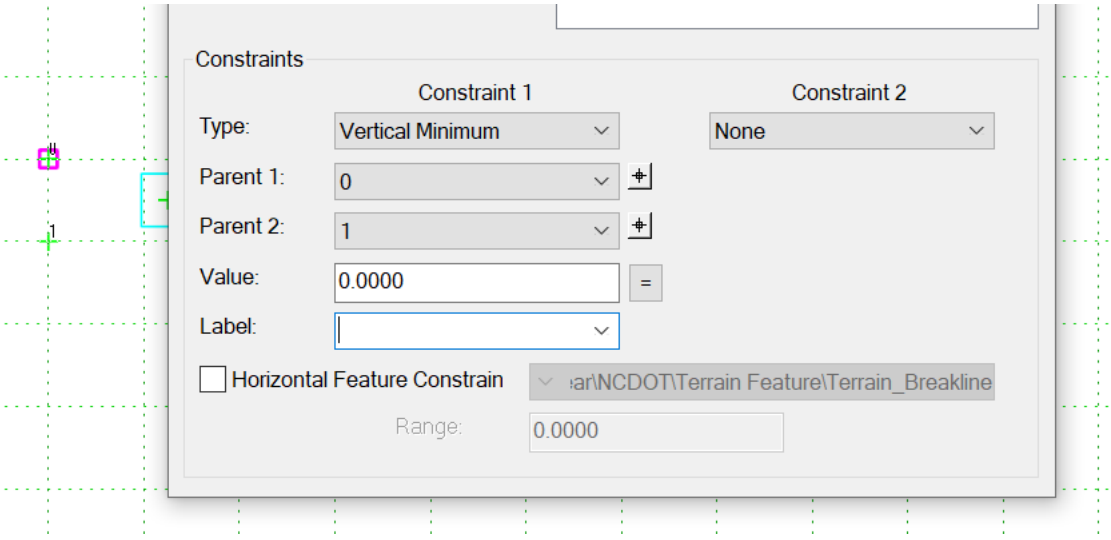




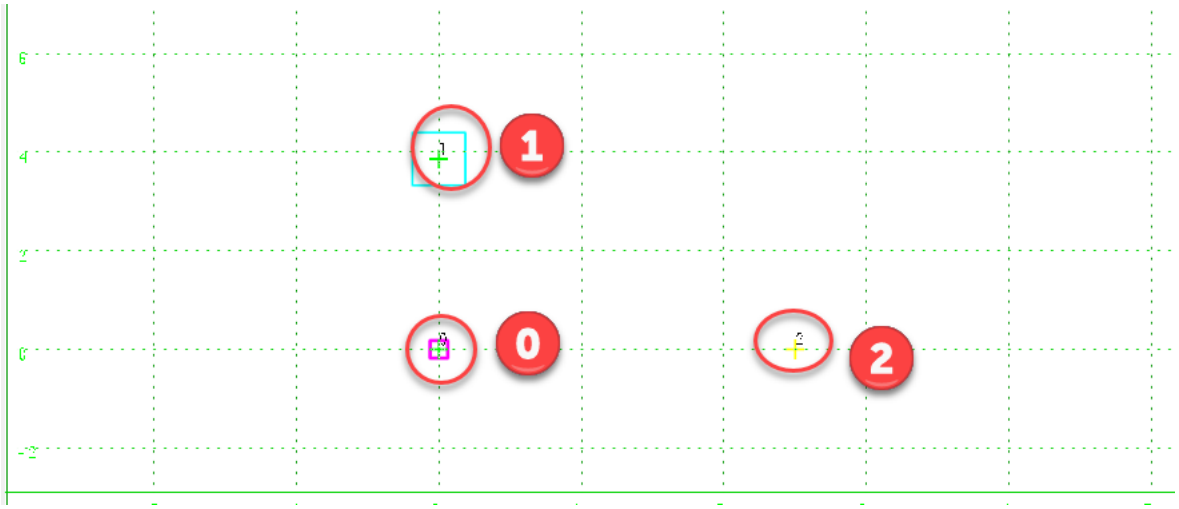
J. Vertical Minimum

Constrained a point vertically between two parent points. The lowest (Y minimum) parent point is used as the reference point. A vertical offset distance value can be applied to both parent points.

In template **10 Vertical Minimum**, constrain **Point 2** vertically to follow the lower location of either **Point 0** or **Point 1**. Afterwards, right click on **Point 1** and select **Test Point Controls >>> Test Vertical Point Control** and move **Point 1** up and above **Point 0**. Note the vertical location of **Point 2** before and after the switch occurs.

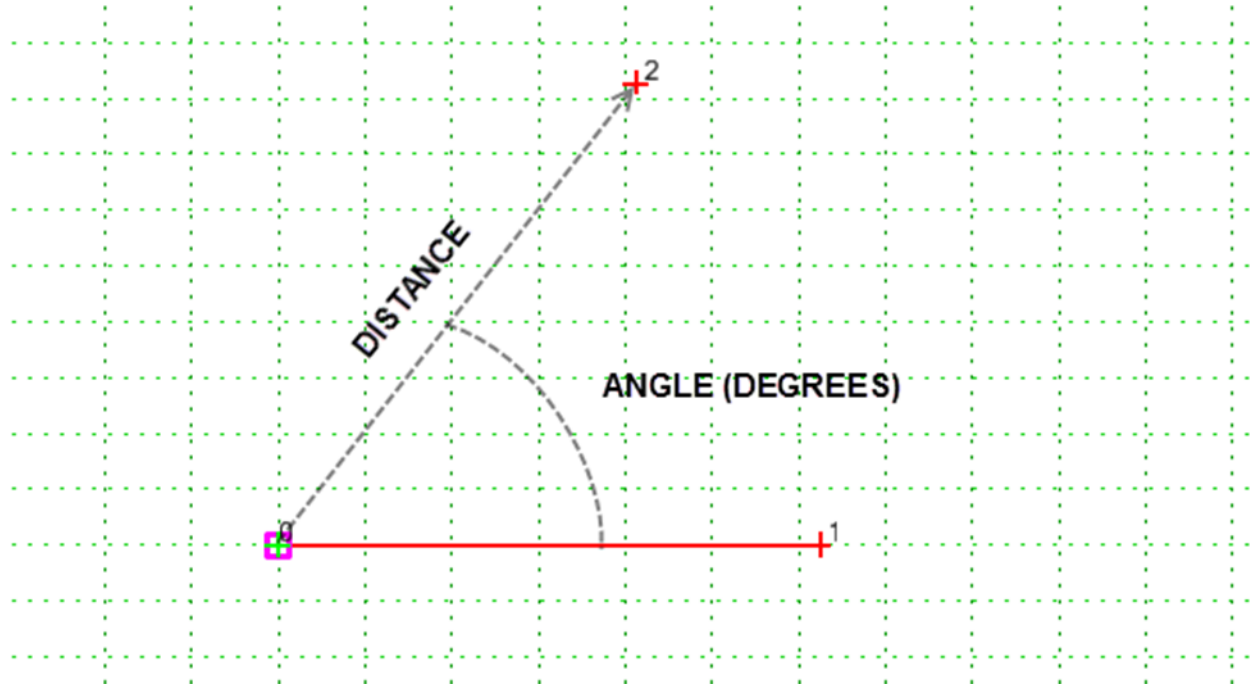


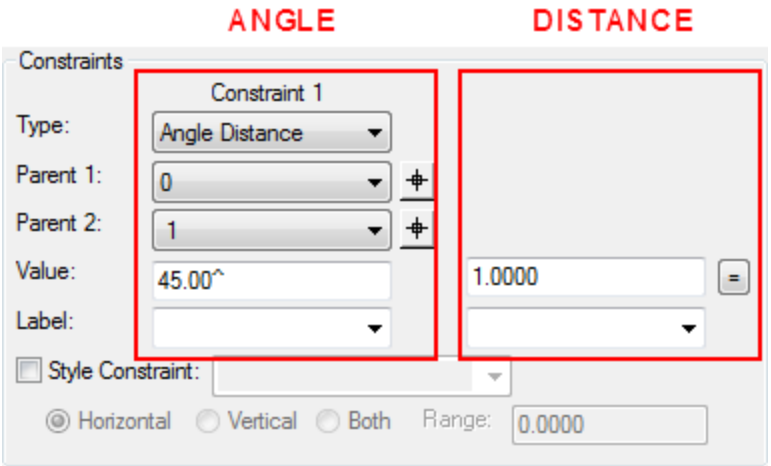
Vertical Minimum Check



K. Angle Distance

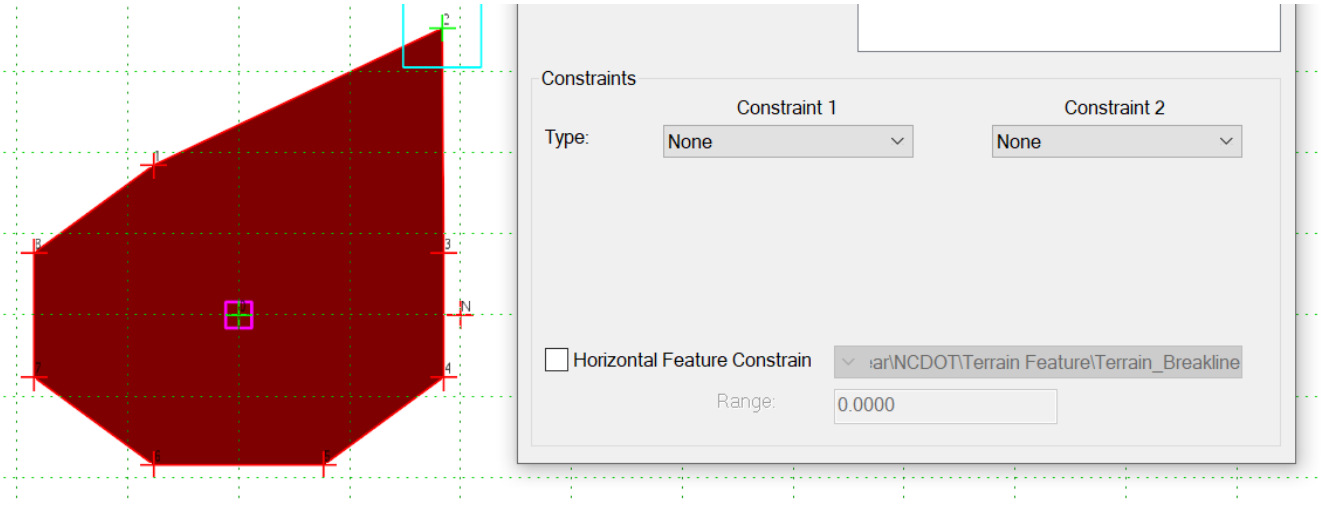
Constrained a point from an angle formed from a vector between two parent points. The distance value is taken from the first parent point. A positive angle value indicates a counter-clockwise angle in degrees. A negative angle value indicates a clockwise angle in degrees. A positive distance value indicates a point location to the right of the first parent point. A negative distance value indicates a point location to the left of the first parent point. Note that this is two full constraint types (occupies both angle and distance value fields) and disabled if another constraint type, such as Horizontal or Vertical, has already been set.

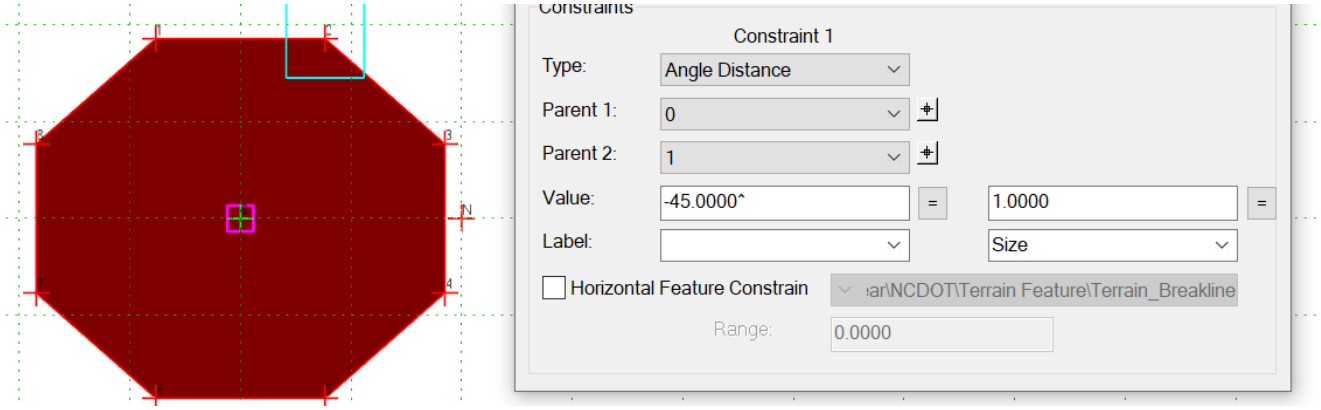




In template **11 Angle Distance**, constrain **Point 2** at an angle of **-45°** (negative) formed by the vector between **Point 0** (center) and **Point 1** (left of **Point 2**). Set the distance of **1'** from **Point 0**.

From the drop-down arrow icon under the distance value, choose "**Size**" as the **Parametric Constraint Label**.



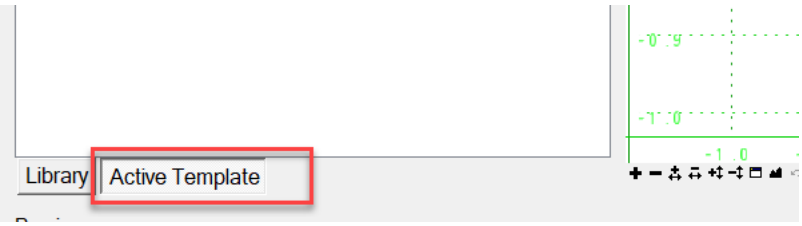


Exercise P3: Parametric Constraint-Basic

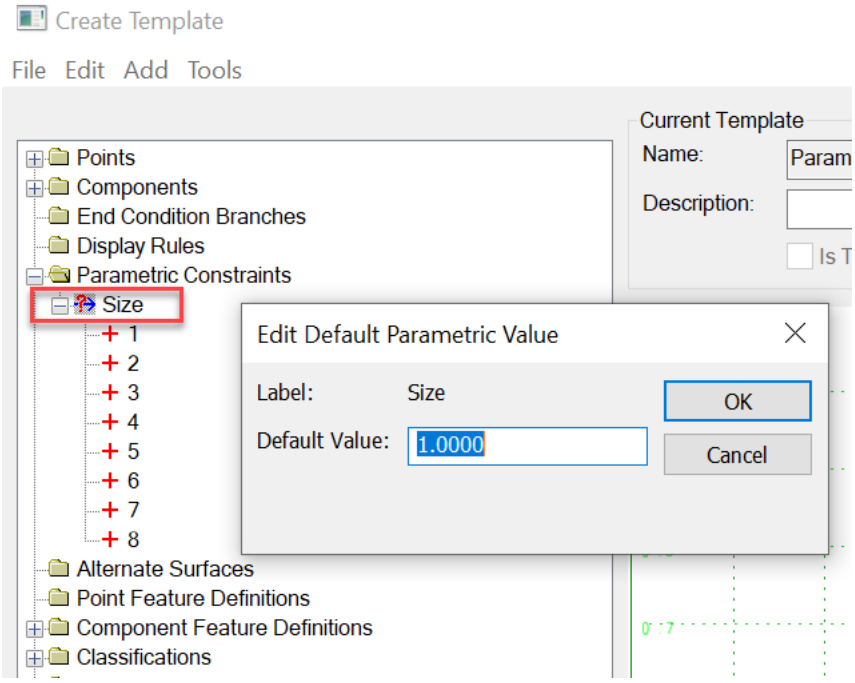
In this exercise, we will demonstrate the basics of parametric constraints.

In the Template Library **Open** the **01 Pomts\03 Parametric Constraint** folder and select the template **Parametric Constraint Basics**.

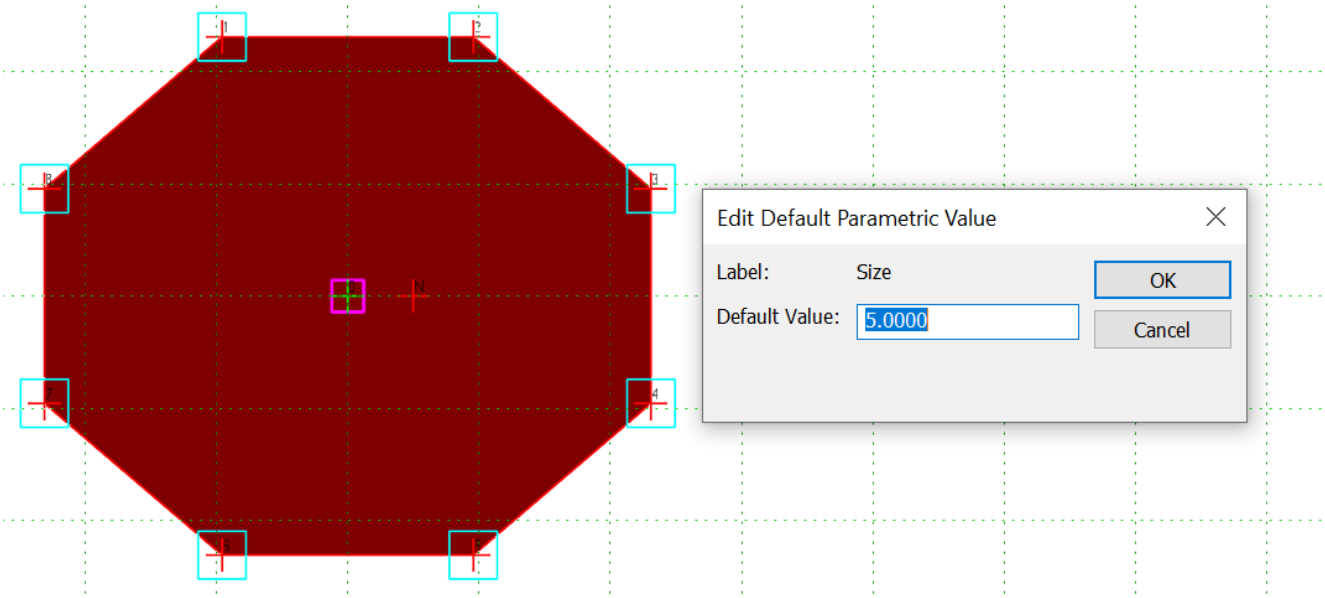
1. Select the **Active Template** tab (lower left).



2. Open the **Parametric Constraints** folder and double click on the label **Size** to edit the default value.



3. Change the default value from 1 to 5 and click OK.



Note the highlighted points (blue box) containing this parametric constraint label.

Parametric constraint labels can also be added and edit to the corridor as a Corridor Object (bucket) when a corridor modeling file has been created. This procedure will be demonstrated in other training modules.

For a list of standard parametric constraint labels used on a typical road template, see **Appendix B**.

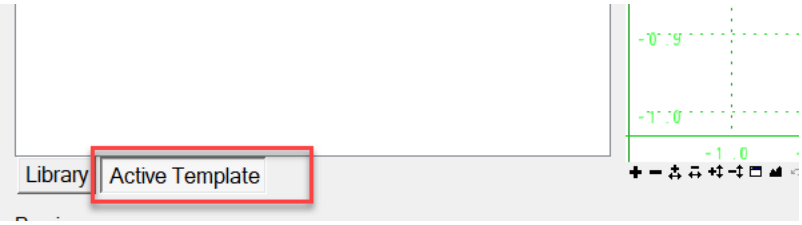
Exercise P4: Parametric Constraint–Zero Out Pavement Layers

In this exercise, we will demonstrate the proper technique to zero out the pavement layers not needed for the project. In this exercise we will zero out the ABC and Intermediate Layers.

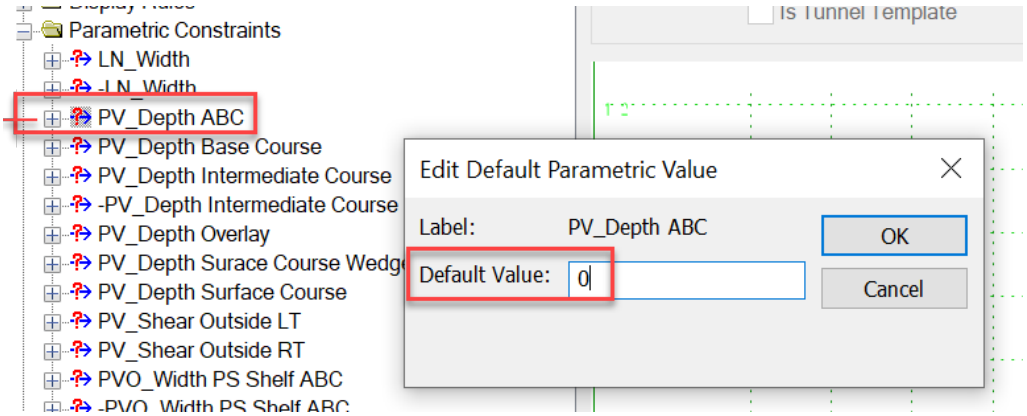
In the Template Library **Open** the **01 Points\04 Parametric Constraint** folder and select the template **1+1 Lanes - LDSS**.

Note the standard naming convention used on our templates. “1+1” means 1 lane on the left and 1 lane on the right of the centerline. LDSS is the end condition for Local Design Standard Slopes.

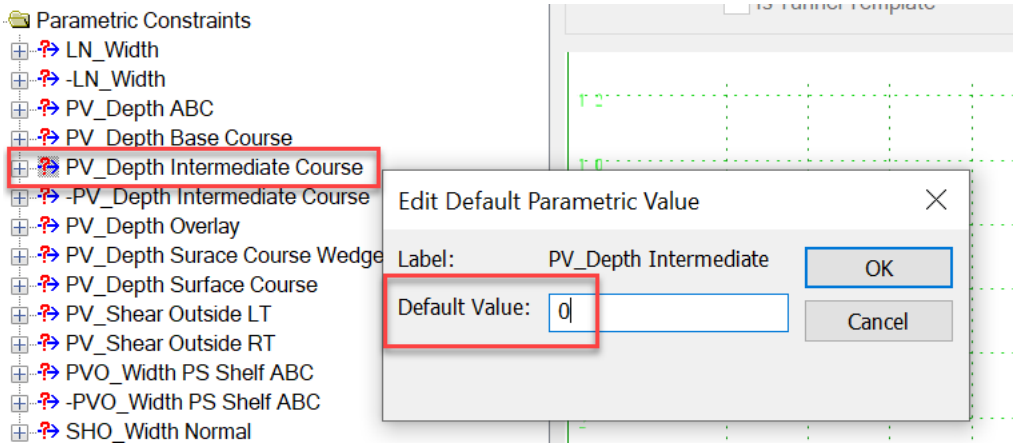
1. Select the **Active Template** tab (lower left).



2. Open the **Parametric Constraints** folder and double click on the label **PV_Depth ABC** to edit the default value. Change the **Default Value** to **0** (zero). Then click on **OK**.



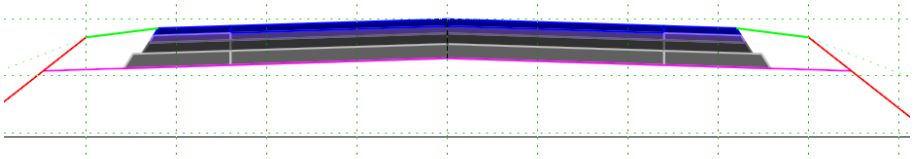
3. Double click on the label **PV_Depth intermediate Cours** to edit the default value. Change the **Default Value** to **0** (zero). Then click on **OK**.



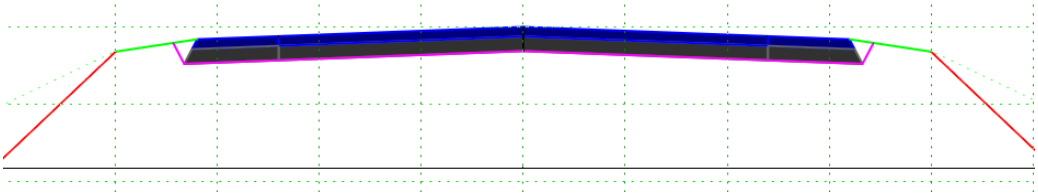
Note that some parametric constraint labels have a negative sign in front of them. This means the parametric constraint label on the right side (without the negative sign) has a “mirror” exact same constraint on the left side. Usually, the value of the left side is the opposite if it is a horizontal or slope constraint. Since the pavement depth is purely vertical, the value is the same for left and right side.

The negative sign also functions as to hide the label from the list the user can pick from.

Mote the pink subgrade daylight component. When the pavement depth is greater than 10” it is considered a “Graded Shoulder”. The subgrade line intersects the side slopes.



When the pavement depth is less than 10”, it is considered a “Trenched Section”. The subgrade line intersects the grass shoulder.



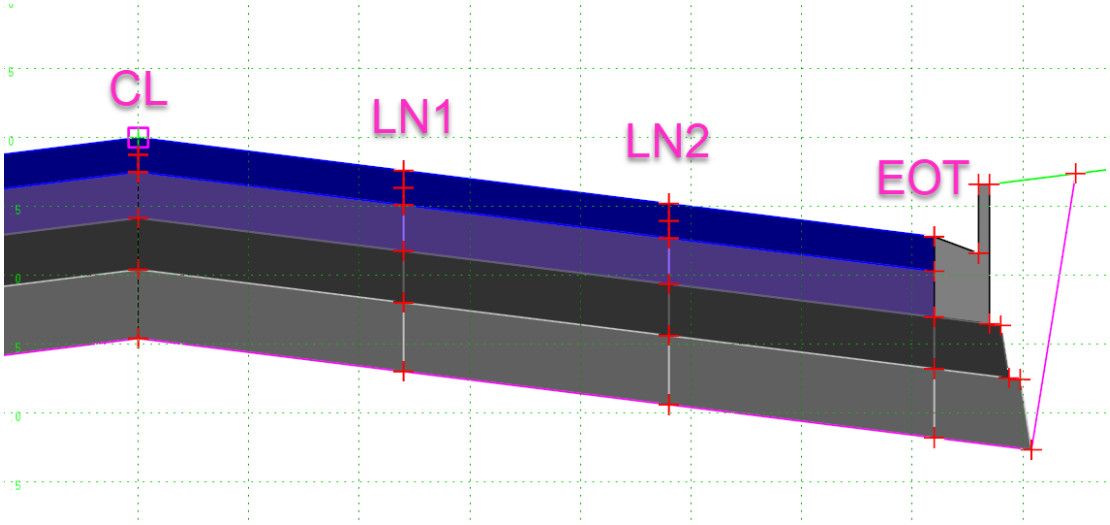
Exercise P5: Parametric Constraints for Middle Turn Lane

In this exercise we will demonstrate the proper technique to modify the standard template for symmetrical and asymmetrical control. Although we do not provide a midde turn lan templae, users can take a standard template and modify it to meet their project needs.

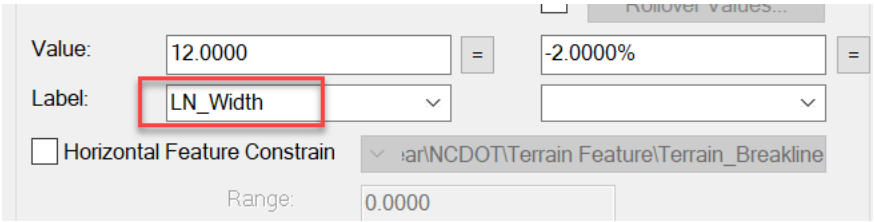
In the Template Library **Open** the **01 Points\05 Parametric Constraints** folder and select the template **UF - 3+3 Lanes - CSLP**.

Note the standard naming convention used on our templates. “3+3” means 3 lanes on the left and 3 lanes on the right of the centerline. CSLP is the end condition for Catch Slopes (Curb and Gutter).

Also note how each lane is numbered from the centerline.



1. Edit the point **+LN1**. This the Lane 1 point on the right side of the centerline. The standard parametric constraint label **LN_Width** is applied to all three (3) lanes on the right side of the centerline.



Note the Affixes applied to the point names when we mirror the right side to create the left side. The “~” (tilde) and “+” (plus sign) prefixes are used as the affixes for the left and right-side point names, respectively.

Any parametric constraint label can be edited by the user, Please edit the templates in the WorkSet/Project folder not the standard template in the WorkSpace.

When editing the parametric constraint labels, there are two (2) options depending on the desired result.

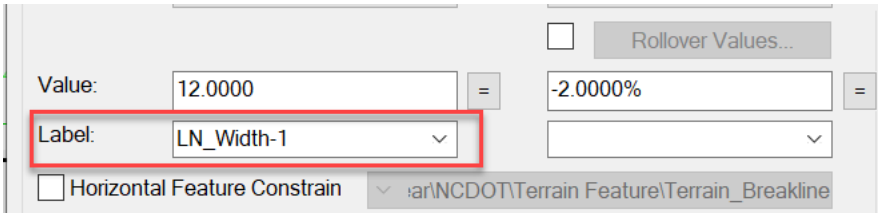
Symmetrical Control

If the desired result is to keep the value the same but opposite direction (symmetrical) then rename the label with “-1” at the end. When editing the left point ~LN1 and add the same suffix “-1”.

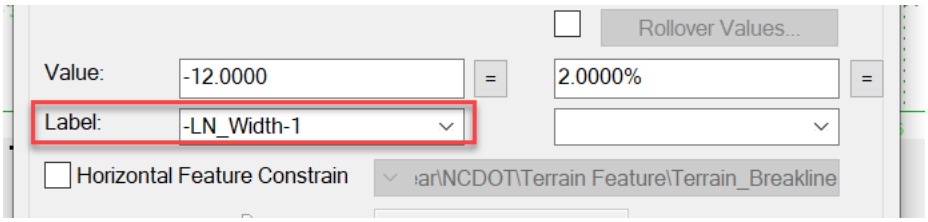
Asymmetrical Control

If the desired result is to have independent left and right values (asymmetrical) then rename the label with “-RT” at the end. When editing the left point ~LN1 and add a different suffix “-LT”. Do not forget to remove the negative sign from the left label to be able display it.

In this case the middle turn lane is symmetrical 8’ on both sides of the centerline. Edit the label to “LN_Width-1”. Then click **on Apply** and **Close**. Note that the label could have been named “LN1_Width” or “1LN_Width”. Any name that is unique to the original is valid.

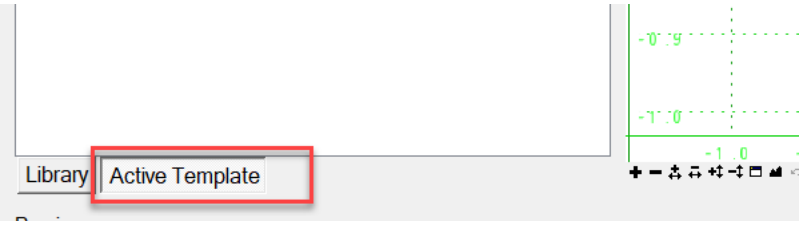


2. For the left point **~LN1**, edit the label to “-LN_Width-1”. Then click **Apply** and **OK**.

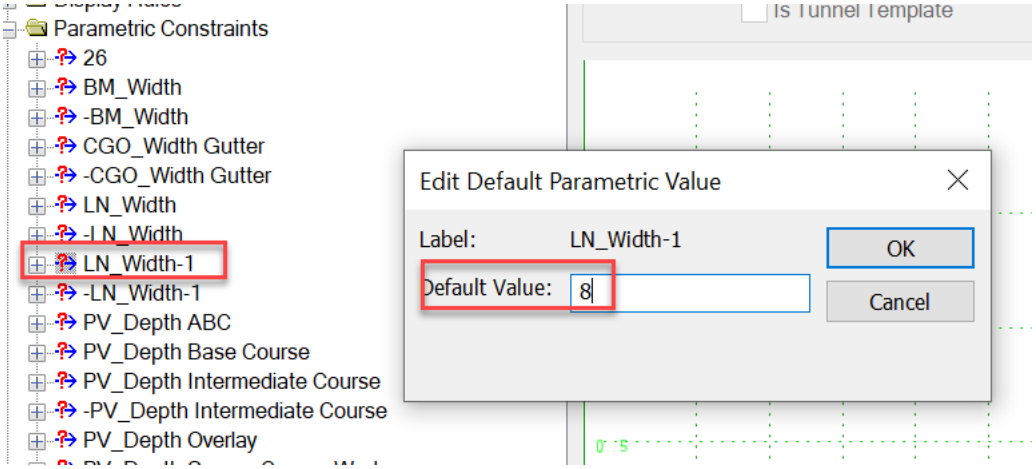


3. From the menu click on **File > Save** to save the changes to the ITL.

4. Select the **Active Template** tab (lower left).

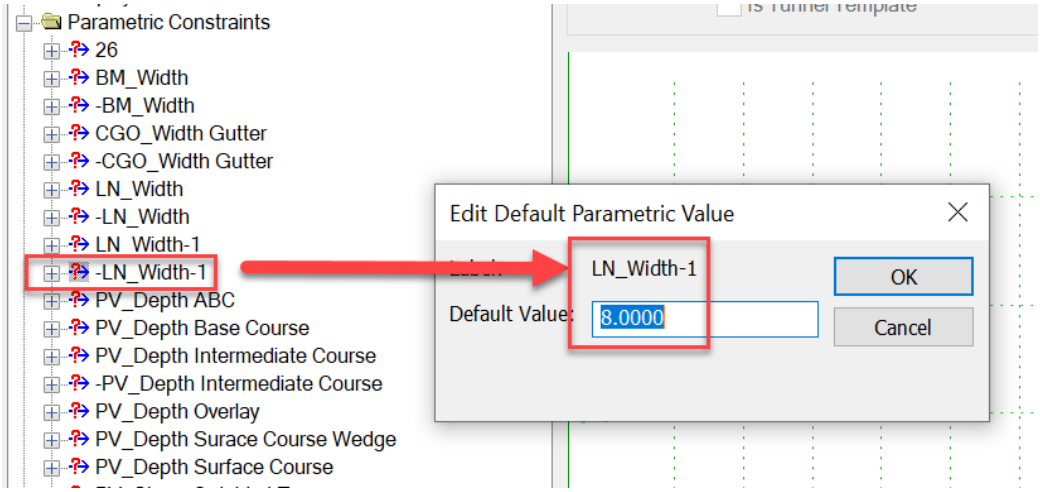


5. Open the **Parametric Constraints** folder and double click on the label **LN_Width-1** to edit the default value. Change the **Default Value** to **8** (eight). Then click on **OK**.

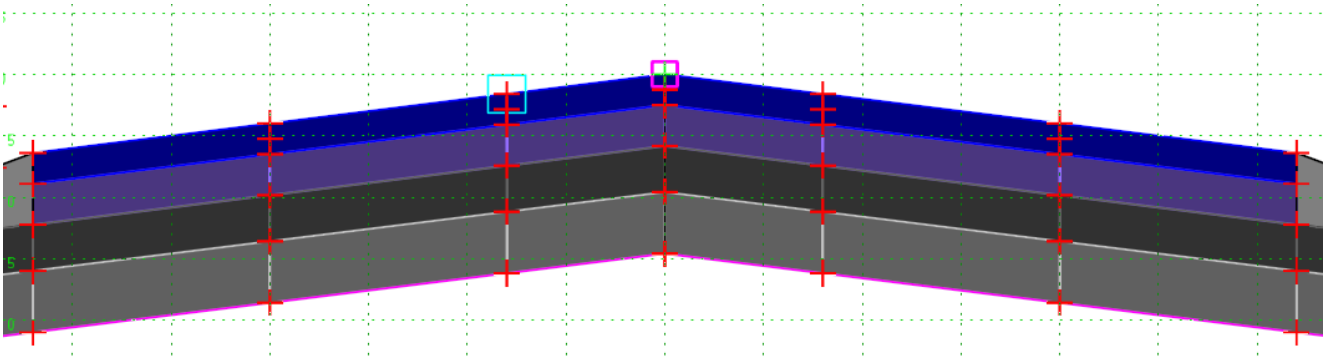


6. Since it is symmetrical, the left side label "-LN_Width-1" is automatically set by reading the right side value and applying the negative to it (-8).

.



The finished product should have a 16’ middle turn lane (8’+8’) while the other lanes are 12’ wide.

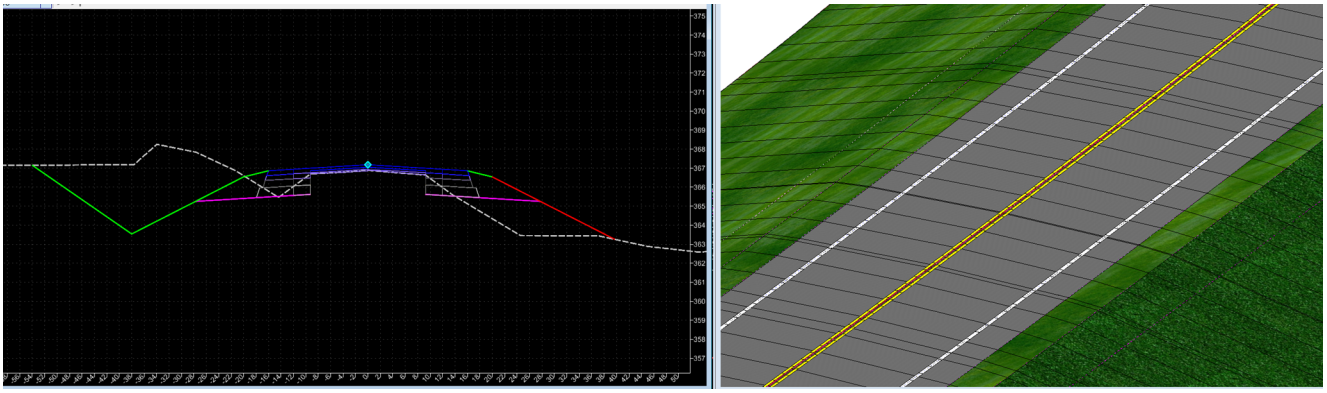


Template Components

Components are the basic building blocks or assemblies of a template. It can be simple as a line formed by two points or complex as multiple shapes joined together by a parent to child relationship. Complex compound components can be attached to each other to create advanced components and templates.

The symbology drawn in the cross section comes from the component feature definition. The 3D surface feature definition and symbology also come directly from the component feature definition. Unlike point properties, components are not automatically merged together as one when place on top of each other.

2D XS and 3D Meshes (Surfaces) Produced by Components



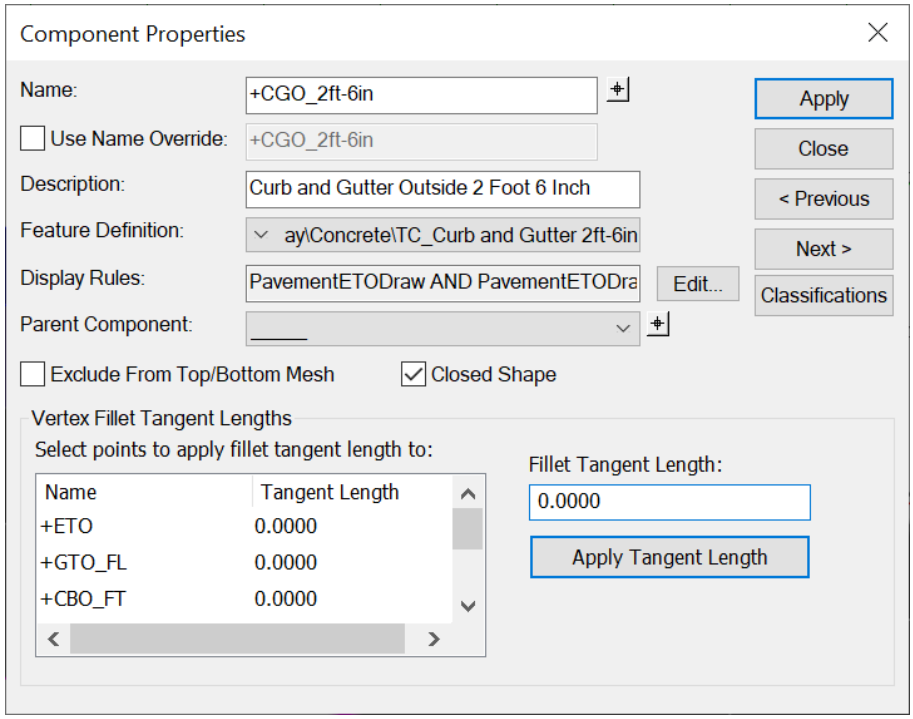
There are four (4) types of components:

I. Component (Regular)
   - Simple*
   - Unconstrained
   - Constrained
II. End Condition (Side Slopes)
III. Overlay/Stripping (Pavement Wedging/Milling)
IV. Circle

*Simple component is perfectly suited for pavement layers (12’ wide with -2% slope).

Component Properties

Regular components, excluding null points, should have these common properties.



Name:
Name of the component on the template. It appears as a white text. If the name already exists, then a number '1' is added to end of it. If the resource has not been deleted, then an increment of one can be added to the component name which already exists. Each component name on the template must be unique. "Applied Affixes" has an effect on Component Names.

Use Name Override:
Similar to **Use Feature Name Override** for point properties, each component name must be unique. However, multiple components can share a single name by using this option. When cross sections and the 3D surfaces are created, this component name override is used instead of the original name.

Feature Definition:
Used to determine the component symbology in the template, cross sections, and 3D surfaces (DGN). May also be used in component quantities.

Parent Component:

Select the controlling parent component. See Parent-to-Child Component Relationship section for further detail.

Display Rules:

Select the rule(s) to turn component off or on. See Display Rules Overview section for further detail.

Exclude From Top/Bottom Mesh:

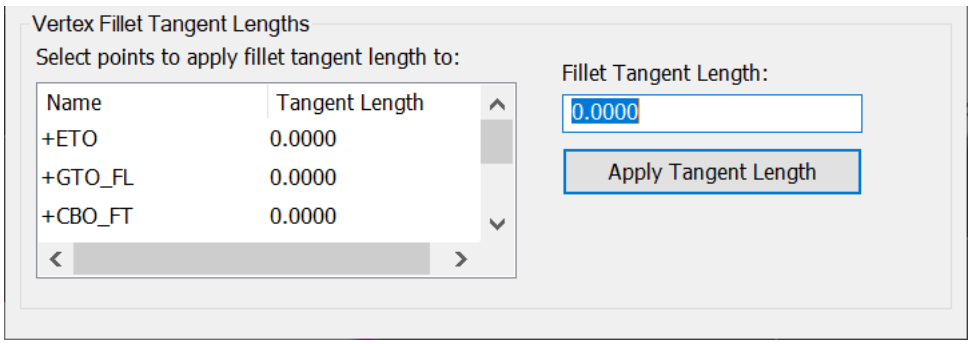
Option to not triangulate the component when the surface is created. A couple of good examples are guardrail and retaining wall components. Even though these are drawn in cross sections and 3D DGN model, the surface should ignore them. Examples of components excluded from top and bottom meshes include, bridge and guardrail.

Closed Shape:

Option to fill in the component shape. It is important to have this checked on if a volume quantity is to be computed from the corridor.

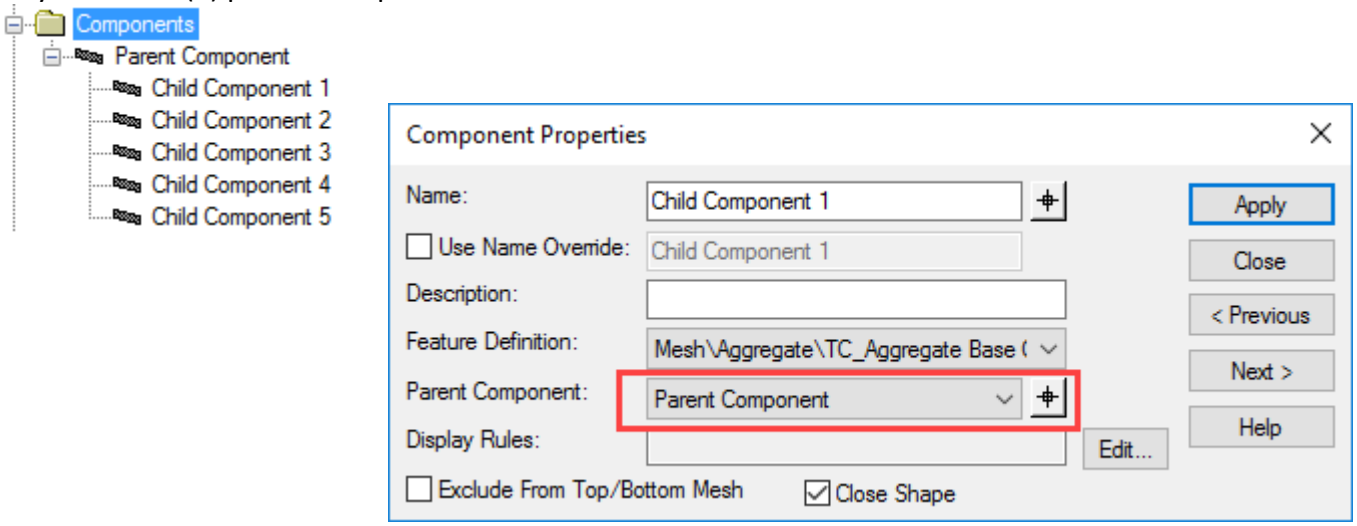
Fillet Options:

Points can be selected on the left panel and a fillet with a desired length can be used instead of a square corner.



Parent Component

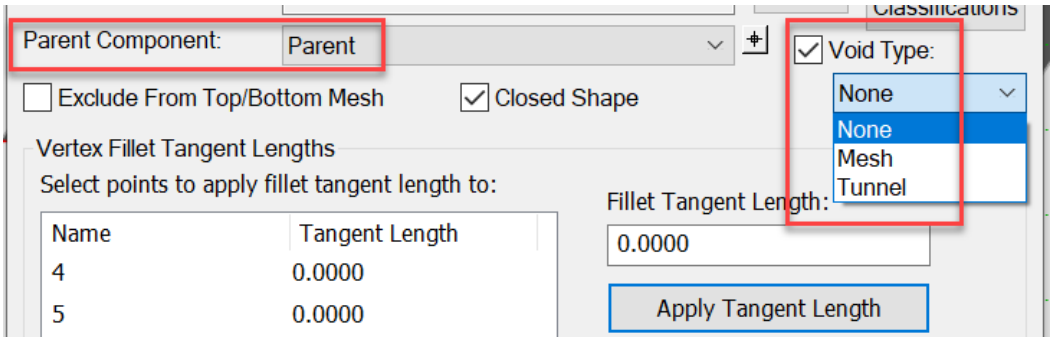
A Parent-to-Child relationship is mainly used to control the display a group of components. Each individual "child" component in the group can be displayed or un-displayed by a single parent component. In addition, if the parent component is deleted, then all child components are also deleted. A parent compoment can have multiple child(ren) components, but a child component may only have one (1) parent component.



Purpose of Parent-to-Child Relationship:

- Group Components
- Dependent Display/Un-display
- Mass Display/Un-display
- Mass Delete

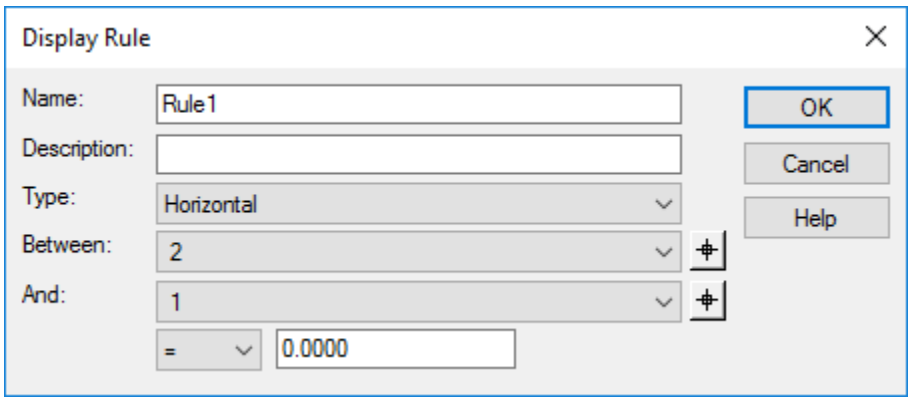
Furthermore. if the child component is completely enclosed by the parent component, it will enable an option to define it (child component) as a **Void Type**.



Display Rules

Display Rules mainly use two (2) points and compare their distance or slope to each other. Equations written as **Display Rules** can be used to turn a component display on (true) or off (false). It is mostly by a parent component, but a child component or any independent component not in a parent-to-child relationship can be assigned a display rule. When mirroring a component, the left side component will have number "1" added to end of the name.

Display Rule Properties



Name:

Name of display rule (cannot contain spaces).

Description:
Full description of what the display rule does.

Type:

There are four (4) main classifications of mathematical comparison operators which can be used to determine if the rule is true or false. The list of available expression evaluation type includes:

- Horizontal/Absolute Horizontal
- Vertical/Absolute Vertical
- Slope/Absolute Slope
- Component is Displayed (Boolean – choose a component)

Between:

First point in the equation.

And:

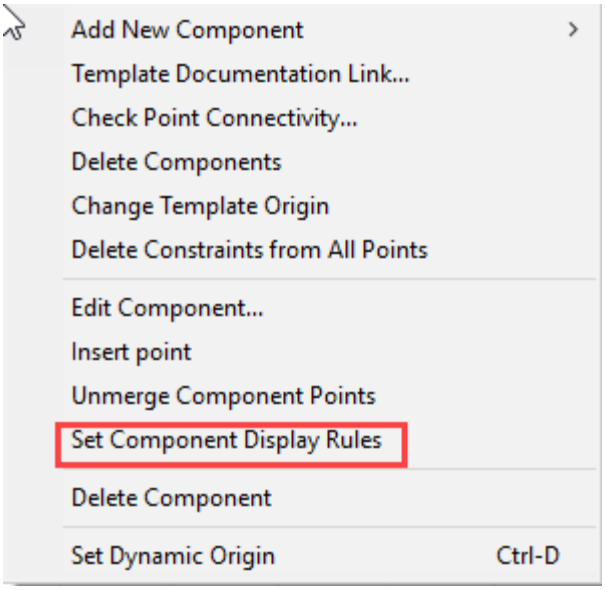
Second point in the equation.

Operator:
< (less than), <= (less than or equal to), = (equal to), >= (greater than or equal to), > (greater than)

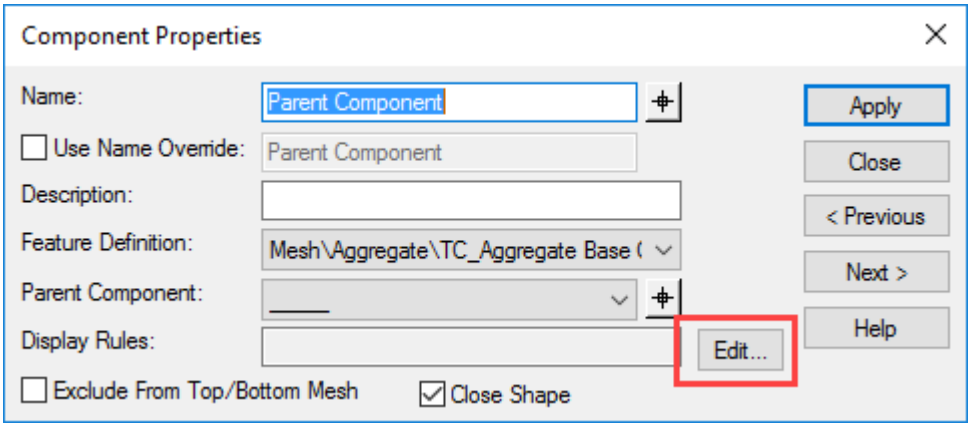
Value:

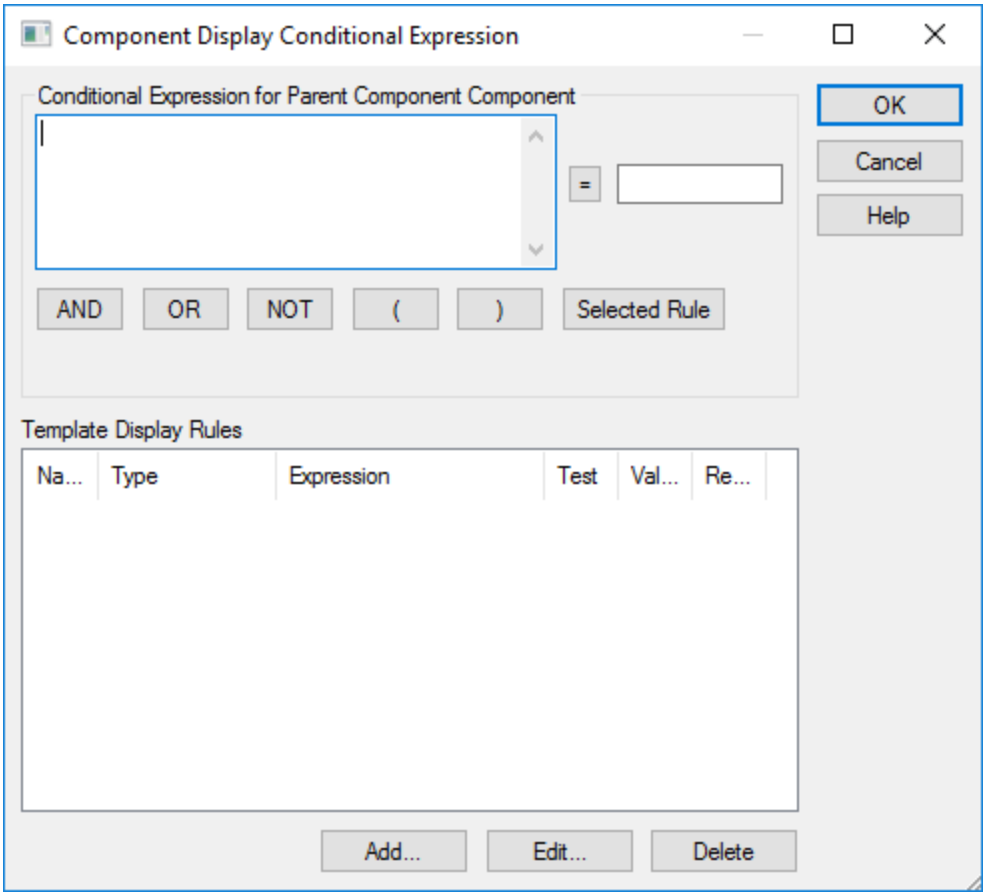
Set a numeric value to evaluate the equation to determine if it is true (On) or false (Off).

Display rules can be assigned to a component either by right mouse click on the component and choose “Set Component Display Rules”.



Or while editing the component properties, click on **Edit** to the right of the Display Rules field.





A list of available display rules can be found under the Template Display Rules field.

Add:

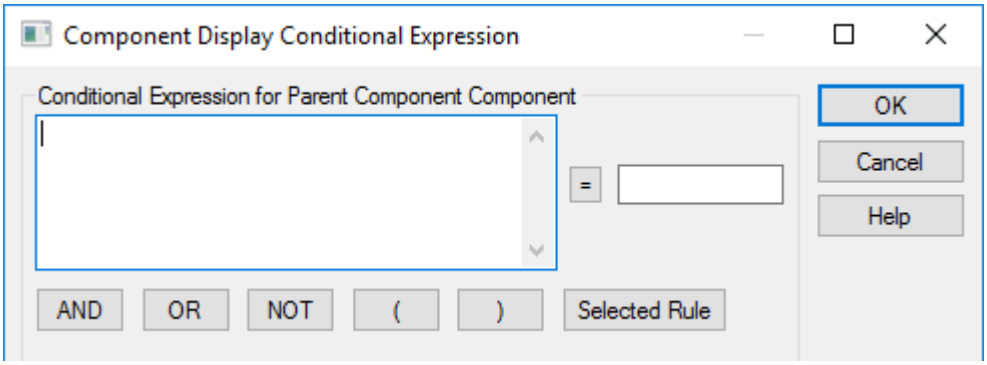
Create new display rule and add it to the list.

Edit:

Modify existing display rule. The display rule must be selected first before it can be edited.

Delete:

Remove and delete the display rule from the list.

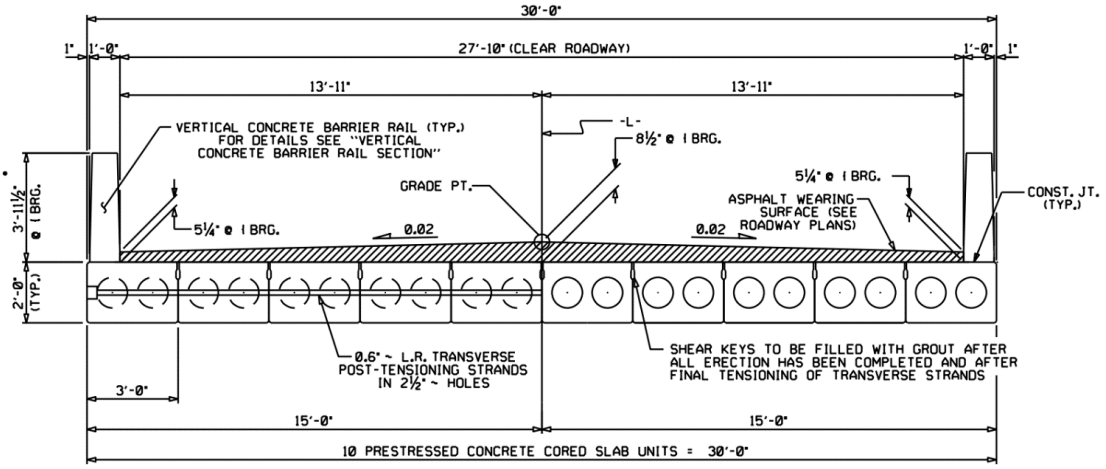


Once the display rules are available in the **Template Display Rules field**, highlight the desired rule and click on the **Select Rules** button to add it to the **Conditional Expression for Parent Component** field. Any combination of the **AND, OR, and NOT** operators can be used for a single display rule or multiple display rules. What is contained in the parentheses "( )" is processed first. Use the equal sign icon "=" to evaluate the expression (true or false) in its default state.

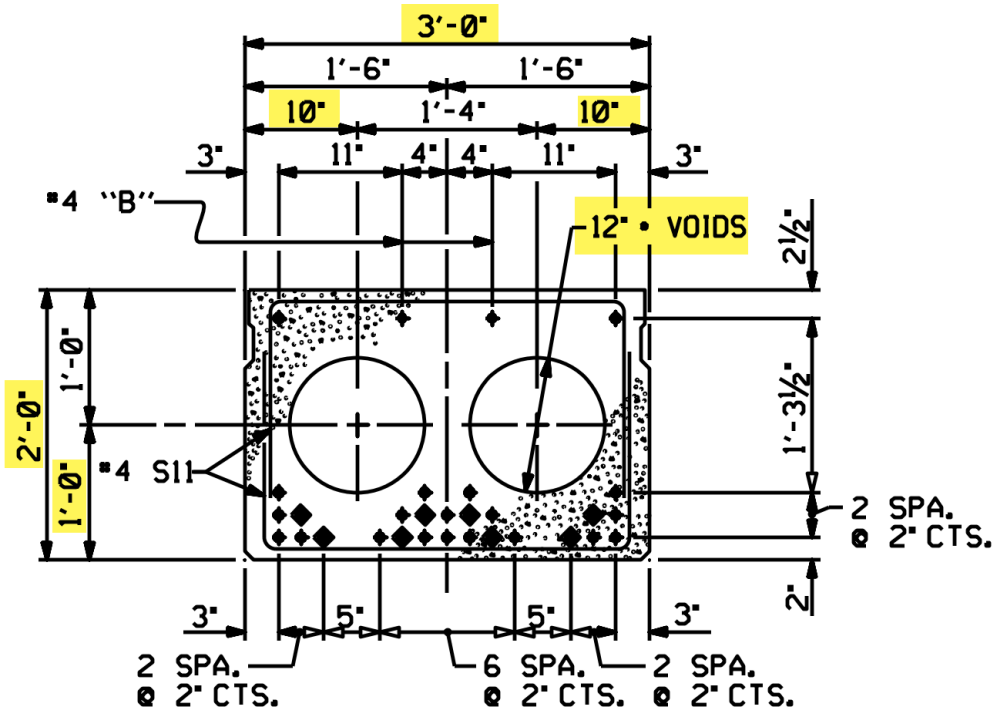
Exercise C1: Cored Slab and Box Beam Bridge Components

In this exercise we will demonstrate how each of the prestressed concrete cored slab and box beam unit is created. Lessons learned through this exercise include creating a **Circle** component, **Fillet Tangent Lengths** and using the **Void** option.

Cored Slab Bridge

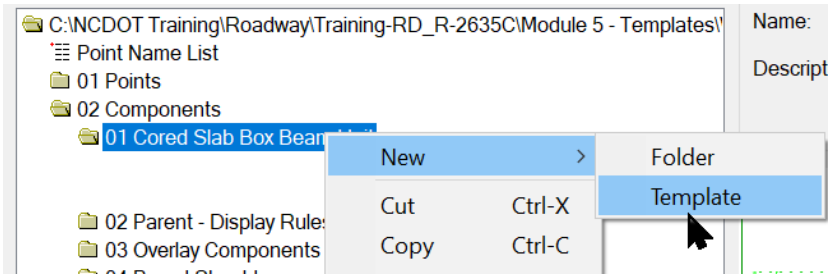


Cored Slab Unit (CSU)

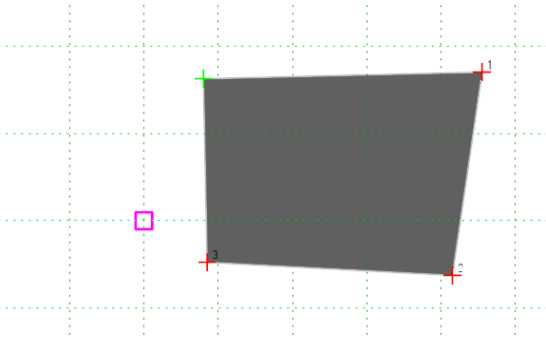


1. In the Template Library open the **02 Components\01 Core Slab Box Beam Unit** folder.

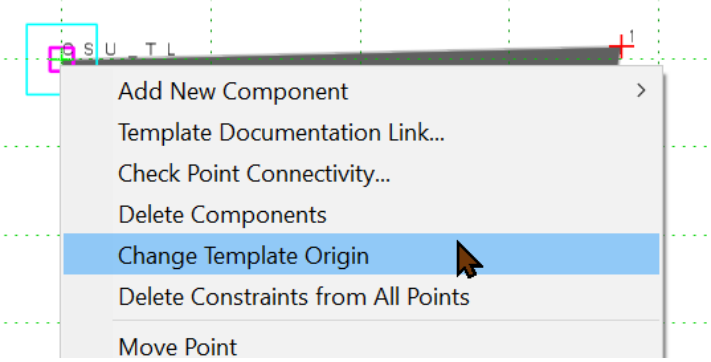
2. Create a new template and name it **CSU**.



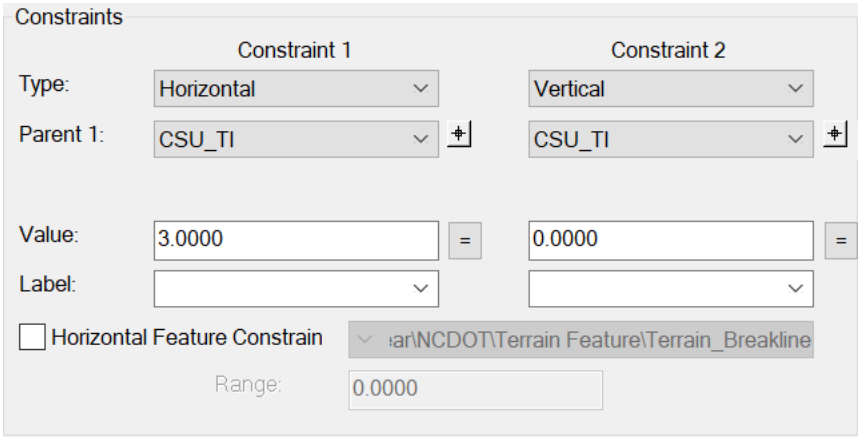
3. Starting from the top left corner **Add New Component >>> Constrain**, create a rectangle that will be the cored slab unit.



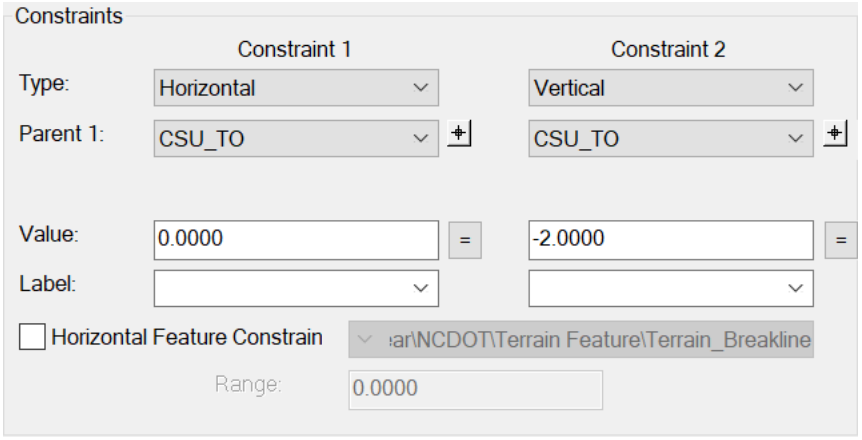
4. Rename the blank point (green) **CSU_TI** and make it the template origin.



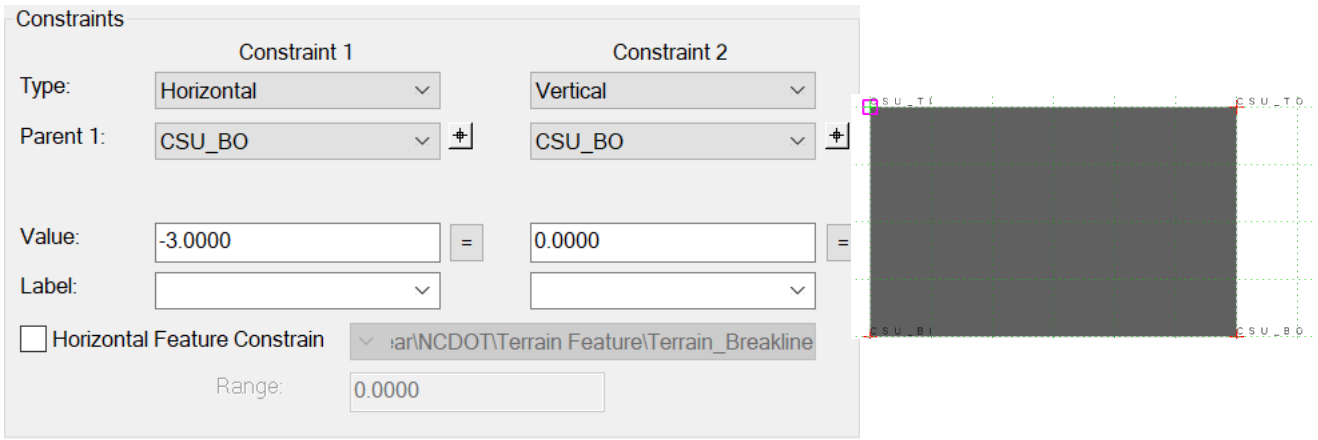
5. To make the **CSU** a 2'x3' box as drawn, edit point **1** and rename it **CSU_TO**. Constrain it as shown below.



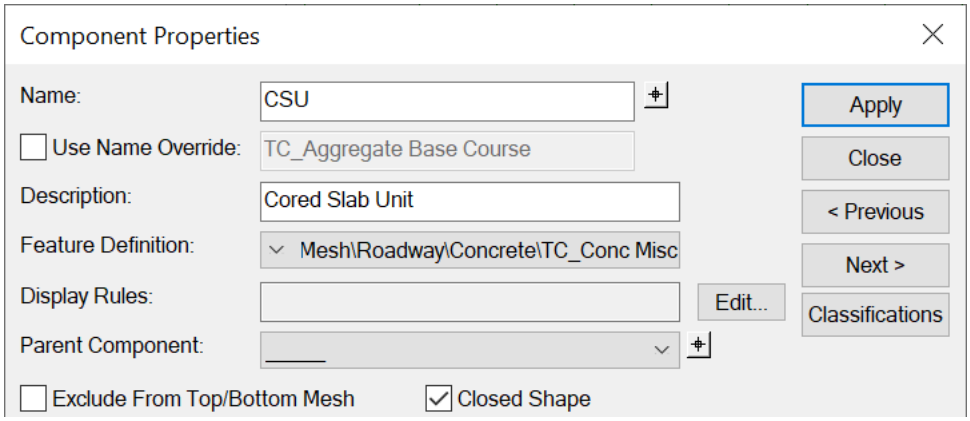
6. Edit point **2** and rename it **CSU_BO.** Constrain it as shown below.



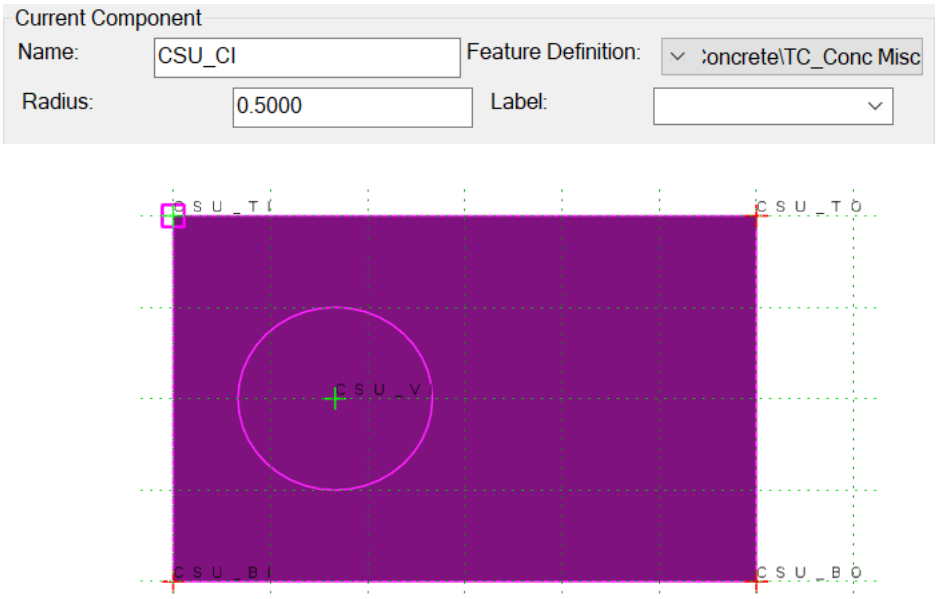
7. Edit point **3** and rename it **CSU_BI**. Constrain it as shown below.



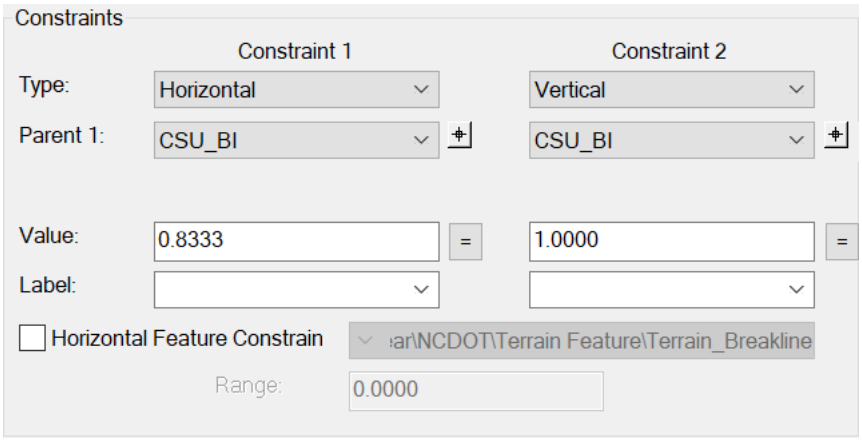
8. Edit the CSU component and rename it **CSU**. The feature definition is set to "**Mesh\Roadway\Concrete\TC_Conc Misc**".



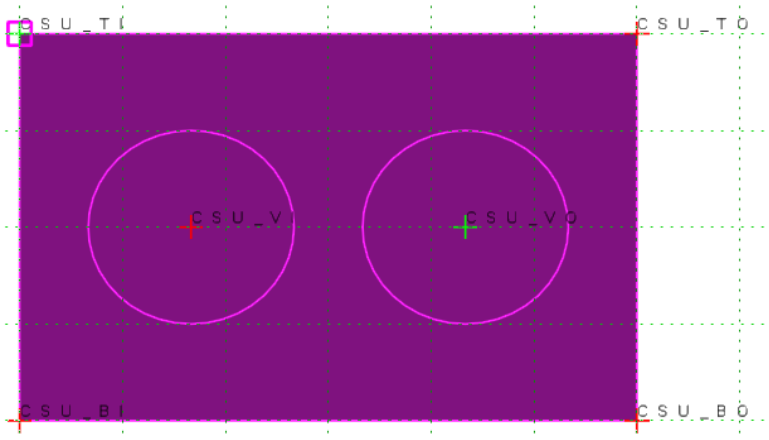
9. Create the left core **Add New Component >>> Circle**. Enter the following values in the component properties prior to placement. The Feature Definition is "**Mesh\Roadway\Concrete\TC_Concrete Misc**".



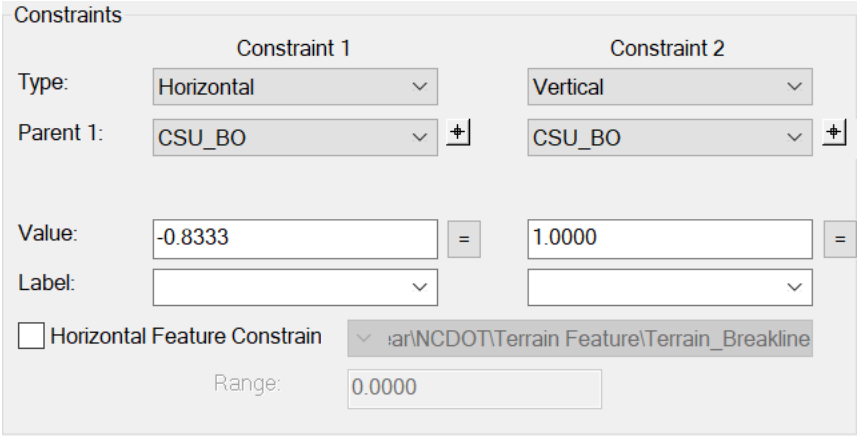
10. Edit the blank point (green) and rename it **CSU_VI**. Constrain it as shown below.



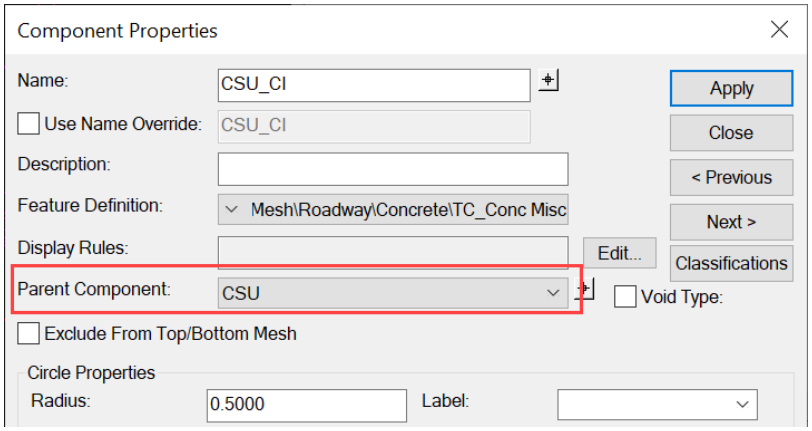
11. Create the right core **Add New Component >>> Circle**. Enter the following values in the component properties prior to placement. The Feature Definition is "**Mesh\Roadway\Concrete\TC_Concrete Misc**".



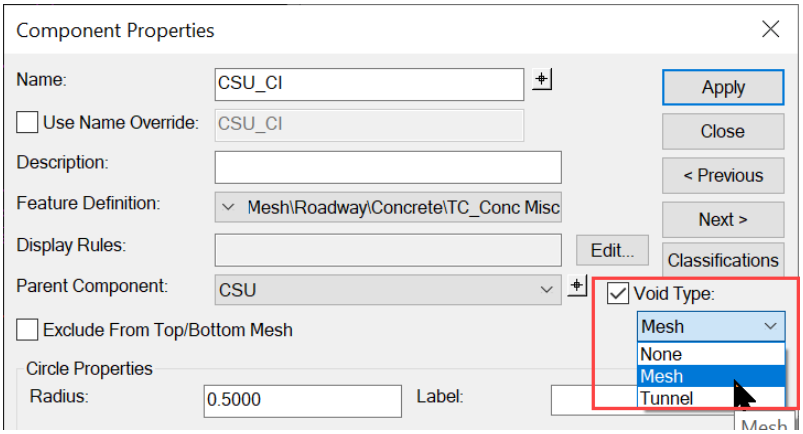
12. Edit the blank point (green) and rename it **CSU_VO**. Constrain it as shown below.



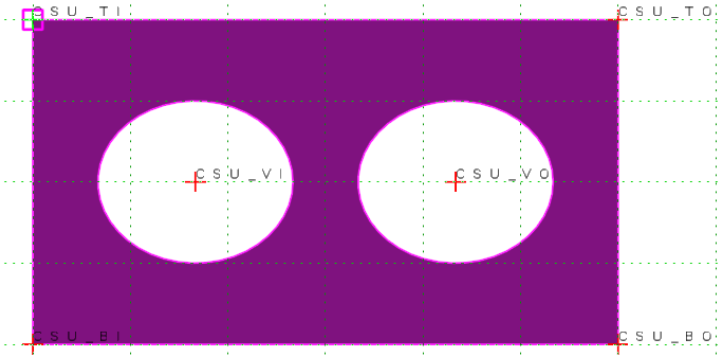
13. Edit the two (2) core components and make **CSU** the **Parent Component**.



14. Note **Void Type** is now enabled. Check this box and select **Mesh**.

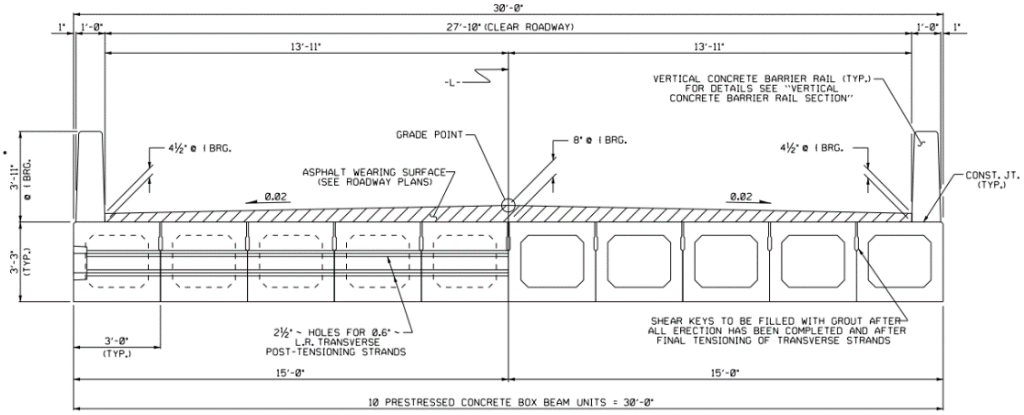


The final CSU component should look like the below picture.

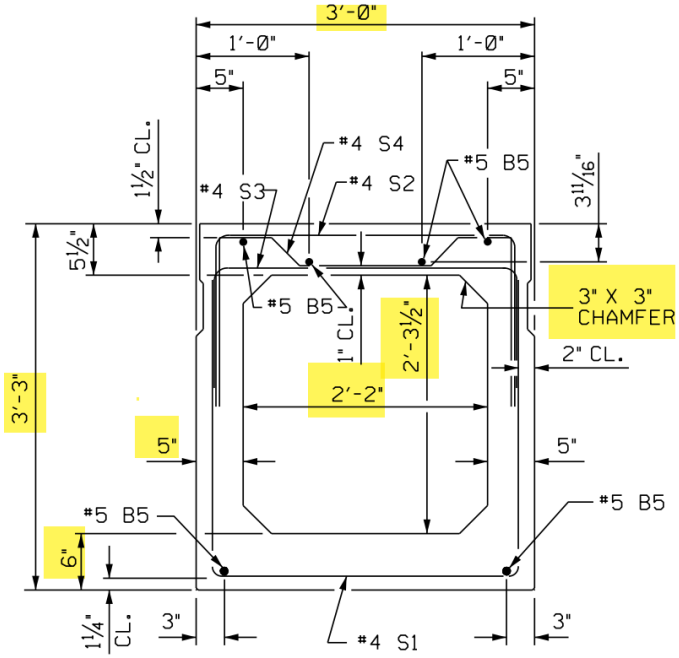


Next, create the Box Beam Unit (BBU).

Box Beam Bridge

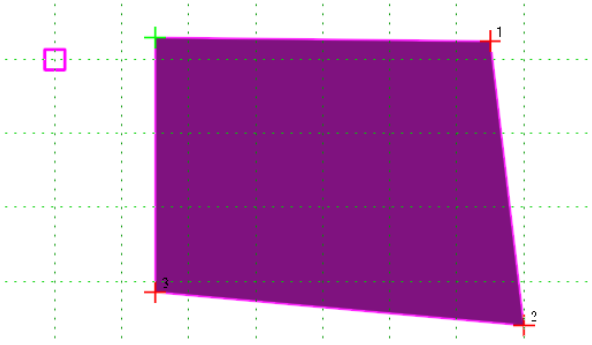


Box Beam Unit (BBU)

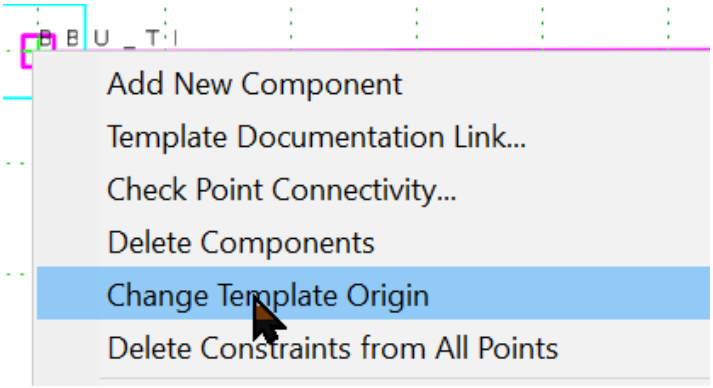


15. Create a new template and name it **BBU**.

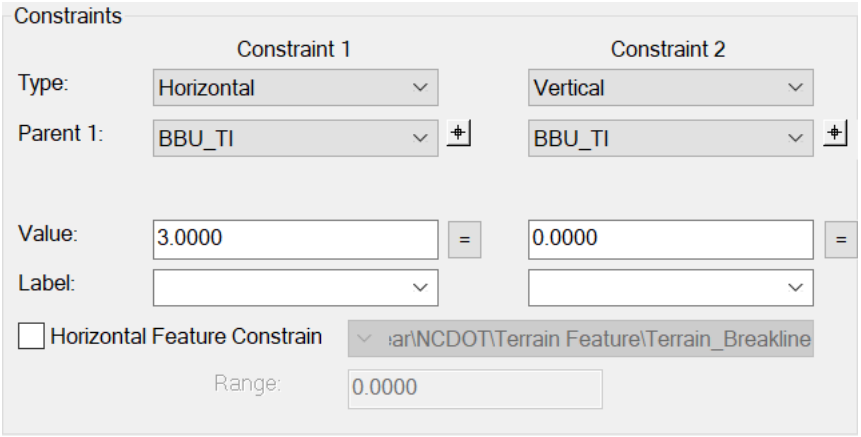
16. Starting from the top left corner **Add New Component >>> Constrain**, create a rectangle that will be the box beam unit.



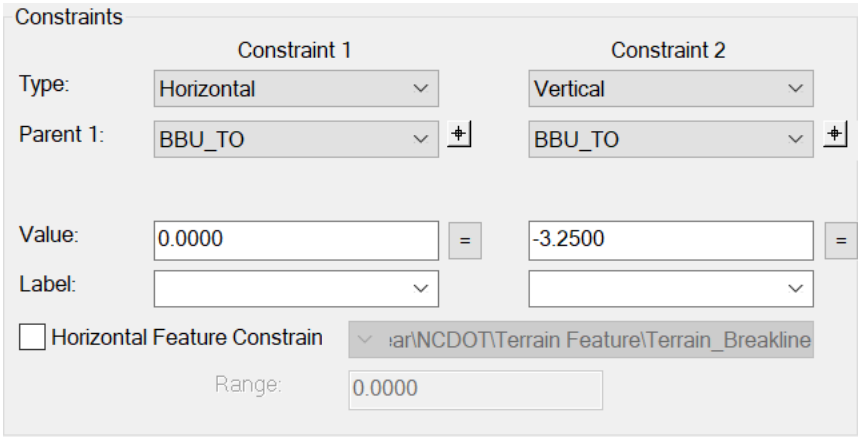
17. Edit the blank point (green) and rename it **BBU_TI**. Make this point the template origin.



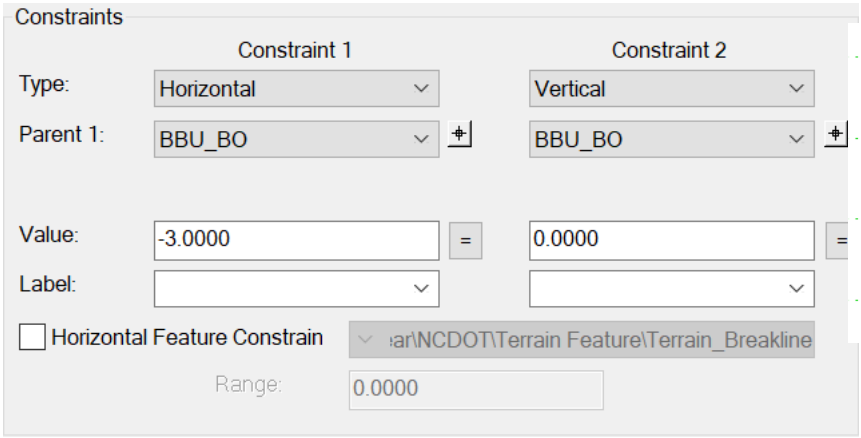
18. To make the **BBU** a 3’-3”x3’ box as drawn, edit point **1** and rename it **BBU_TO**. Constrain it as shown below.

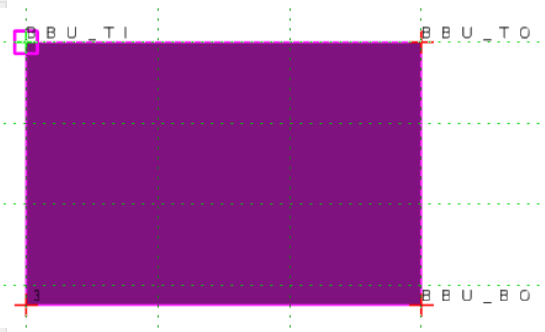


19. Edit point **2** and rename it **BBU_BO**. Constrain it as shown below.

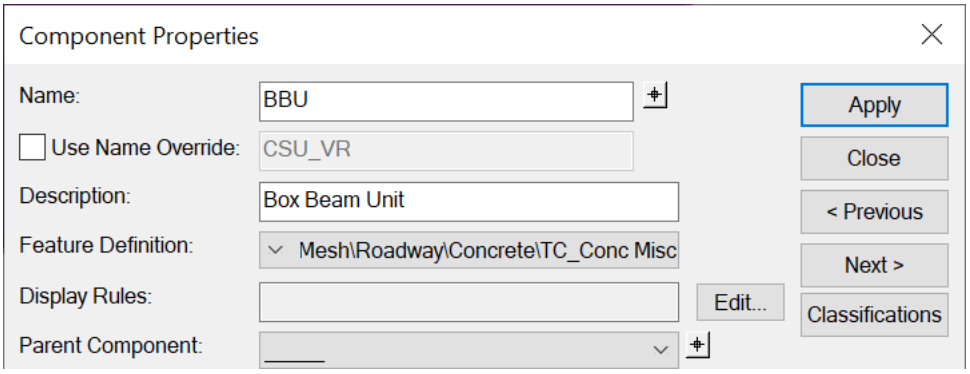


20. Edit point **3** and rename it **BBU_BI**. Constrain it as shown below.

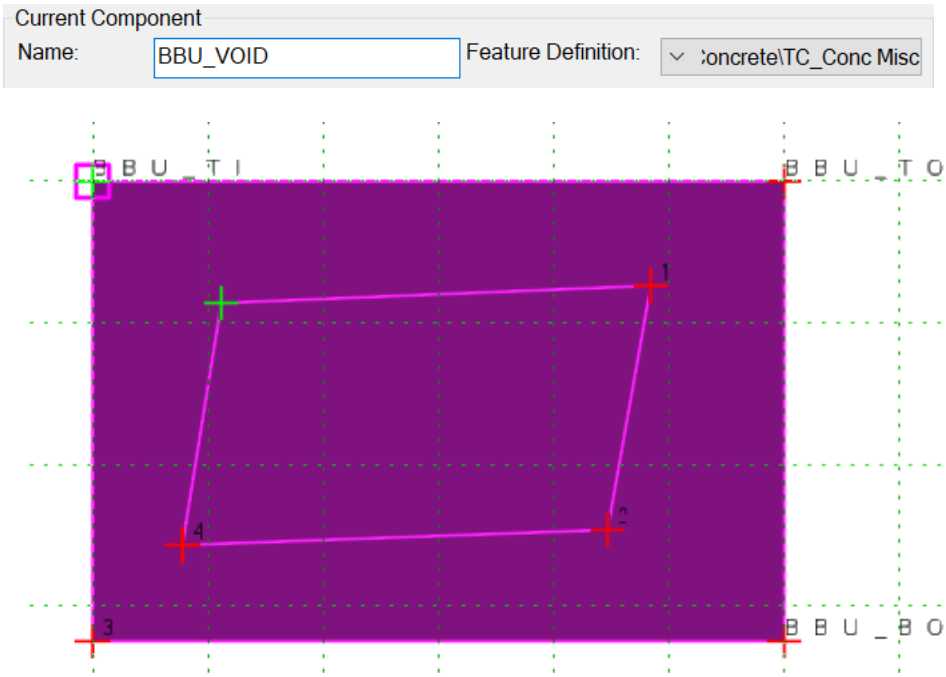




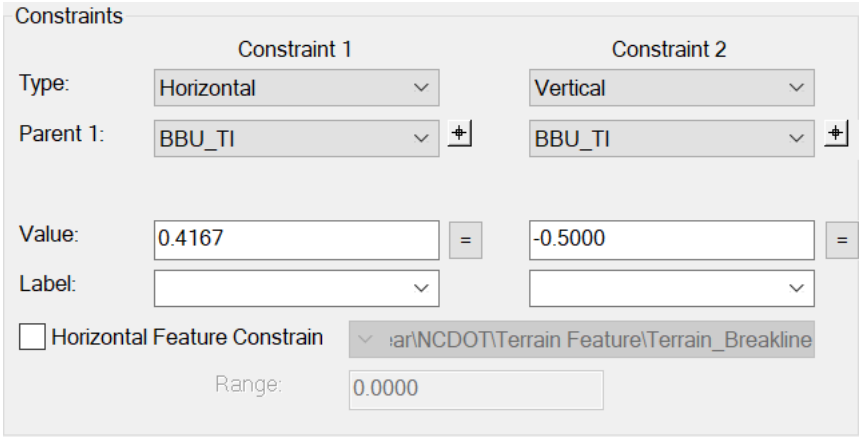
21. Edit the **BBU** component and rename it **BBU.** The feature definition is set to "**Mesh\Roadway\Concrete\TC_Conc Misc**".



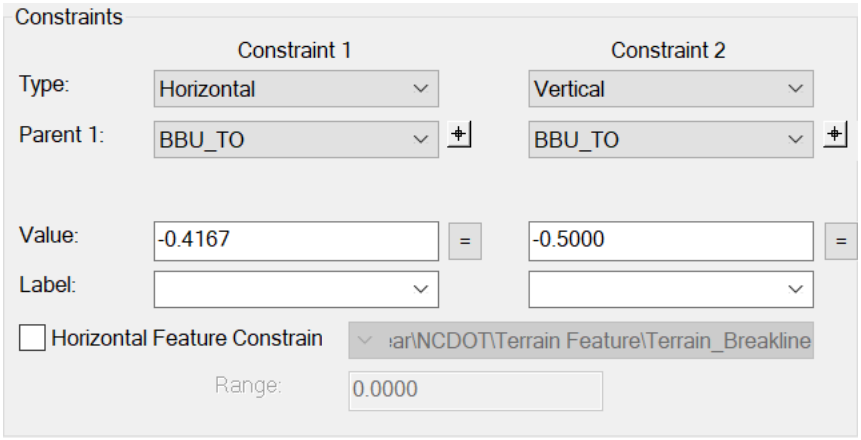
22. Create the inner box (void) **Add New Component >>> Constrained**. Enter the following values in the component properties prior to placement. The Feature Definition is "**Mesh\Roadway\Concrete\TC_Concrete Misc**".



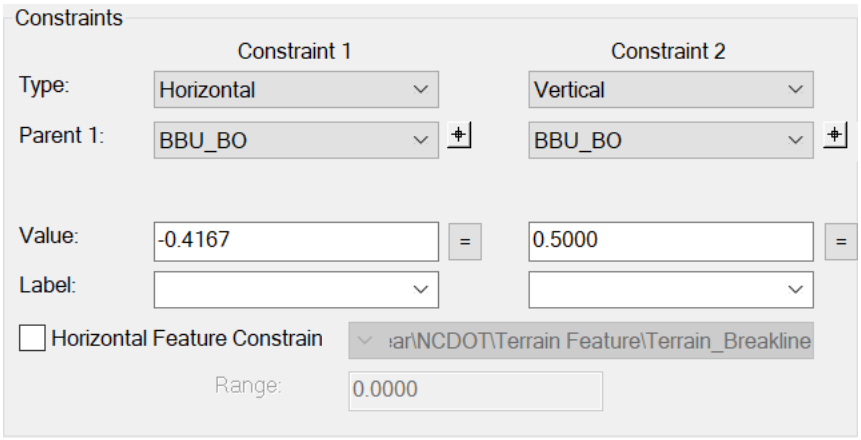
23. Edit the blank point (green) and rename it **BBU_VD_TI**. Constrain it as shown below.



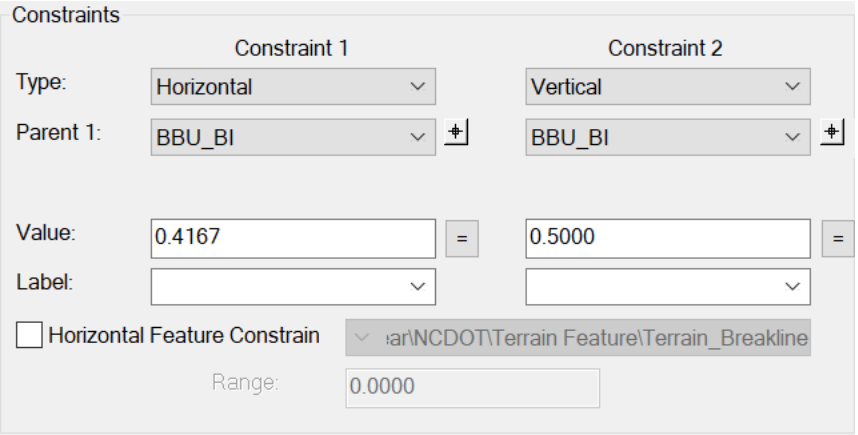
24. Edit point **1** and rename it **BBU_VD_TO**. Constrain it as shown below.

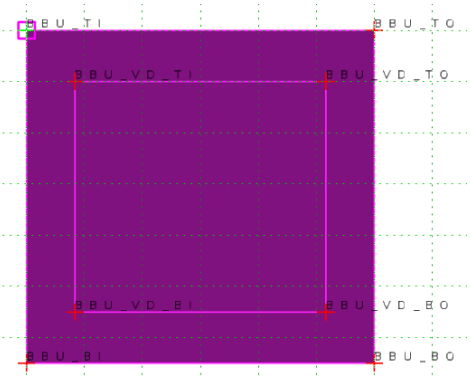


25. Edit point **2** and rename it **BBU_VD_BO**. Constrain it as shown below.

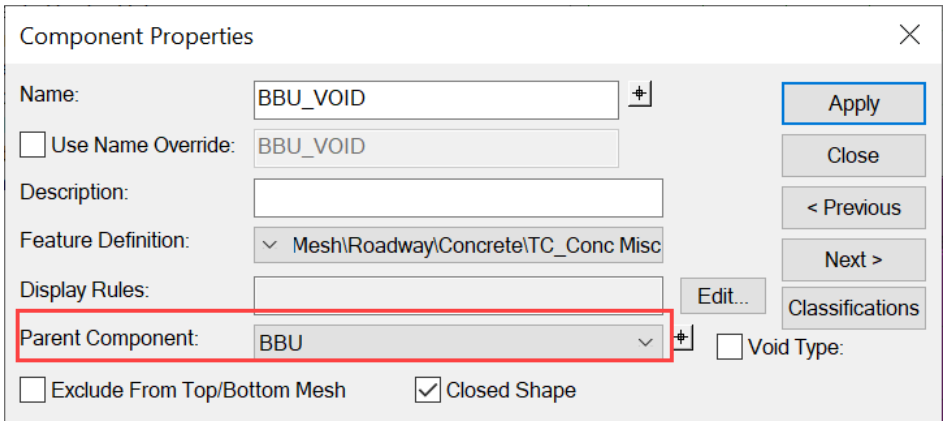


26. Edit point **3** and rename it **BBU_VD_BL**. Constrain it as shown below.

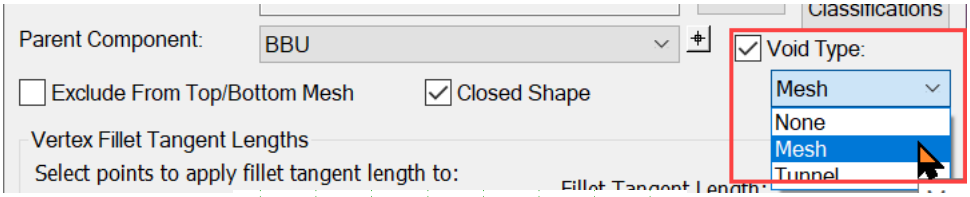


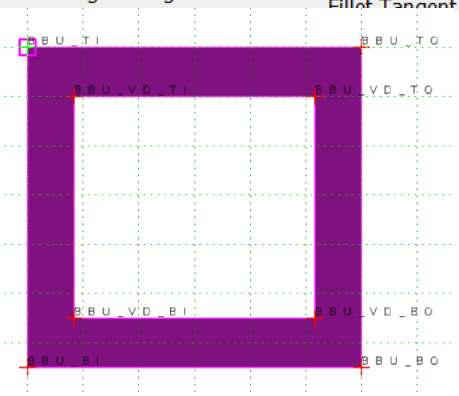


27. Edit the **BBU_VOID** component and make **BBU** the **Parent Component**.



28. Note **Void Type** is enabled. Check this box and select **Mesh**.





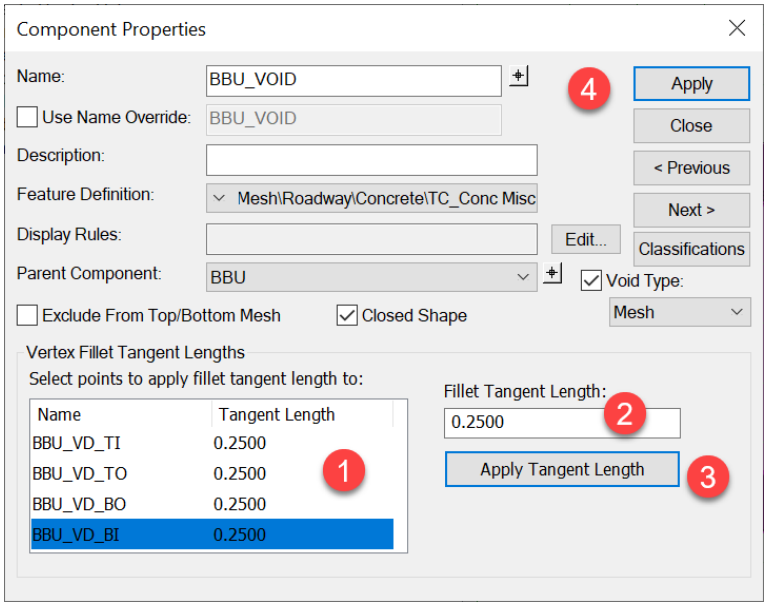
Chamfer vs. Fillet

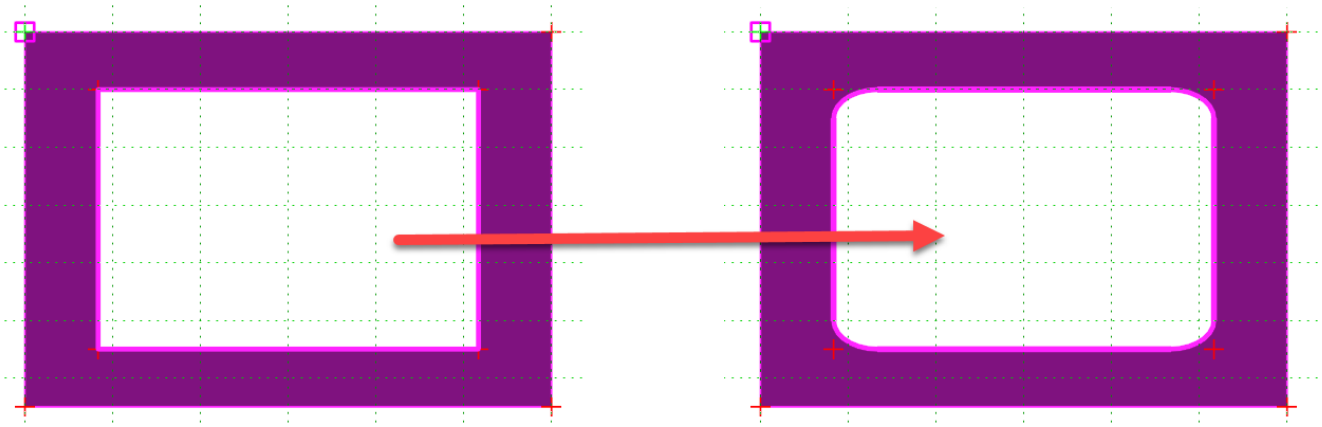
Chamber – bevel (tangent distance from) at the corner.

Fillet – Arc (curve radius) at the corner.

Currently chamfer is not capable with ORD, only fillet.

29. Optional: to treat the corners of the void with a fillet, edit the **BBU_VOID** component and **Apply Tangent Length** (3”) to the corner points.



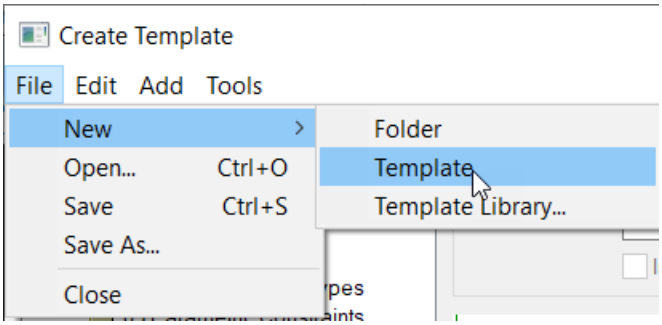


Exercise C2: Pavement Layer Components

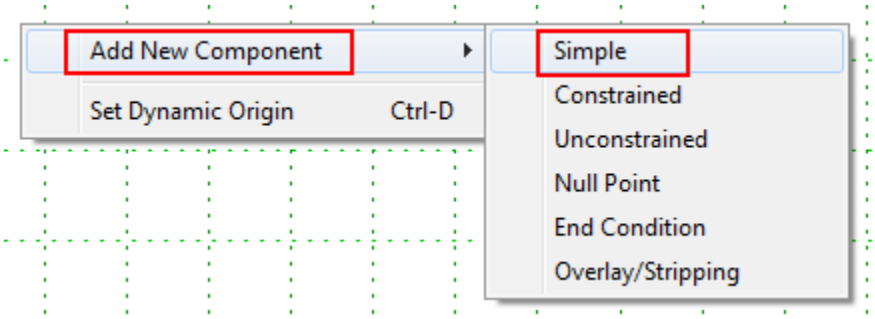
In this exercise we will create four (4) pavement layers (on top of each other). Assign each layer a name and a feature definition. Establish a parent-to-child relationship by making the first layer the parent component. Write a display to turn off all of the layers when the top layer has a zero width (less than or equal 0). Test the template after you are finished. The topics covered in this exercise include **Parent Component** and **Expressions** for **Display Rules**.

In the Template Library open the **02 Components\02 Parent – Display Rules** folder.

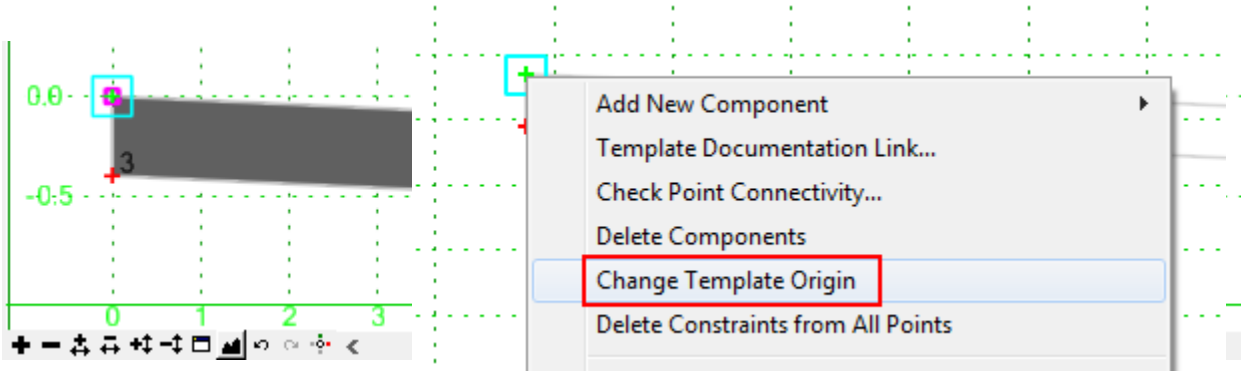
1. Create a new template and name it **CDR.**



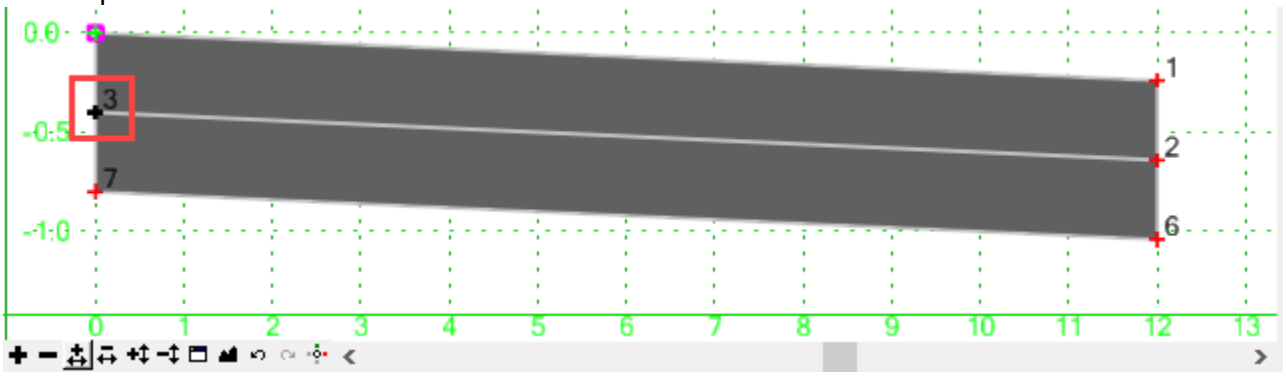
2. In the template **1 Components**, right mouse click on the **Current Template** screen and choose **Add New Component >>> Simple**.



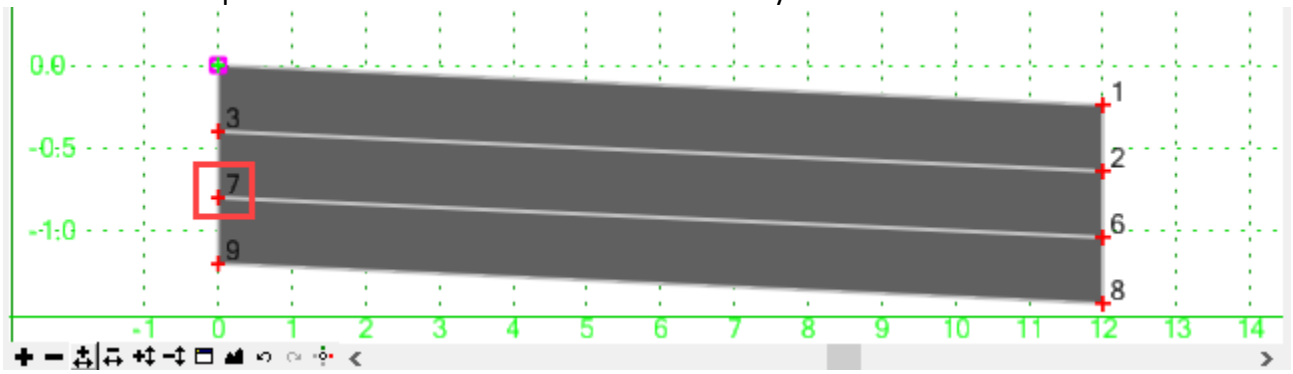
3. Make the unconstrained top left point (not named) as the template origin.



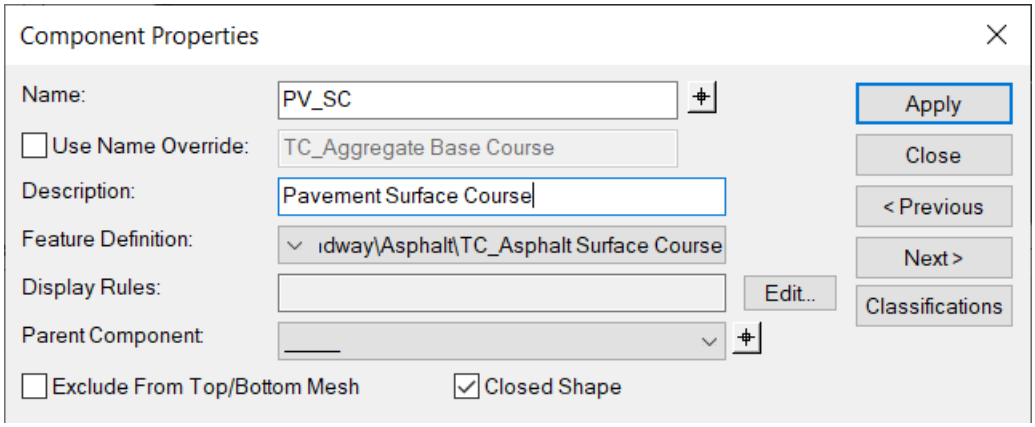
30. Add the second layer by creating another simple component (step 1). Merge the insertion point at Point 3.



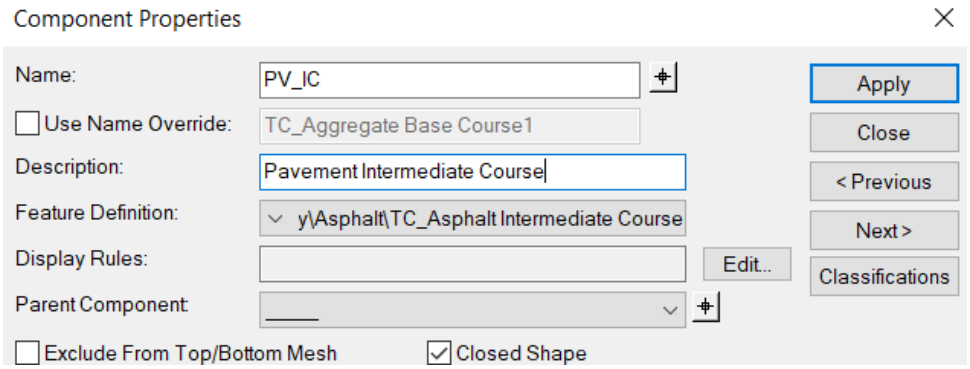
31. Add the third and fourth layer by creating another simple component (step 1). Merge the insertion point at bottom left corner of the above layer.



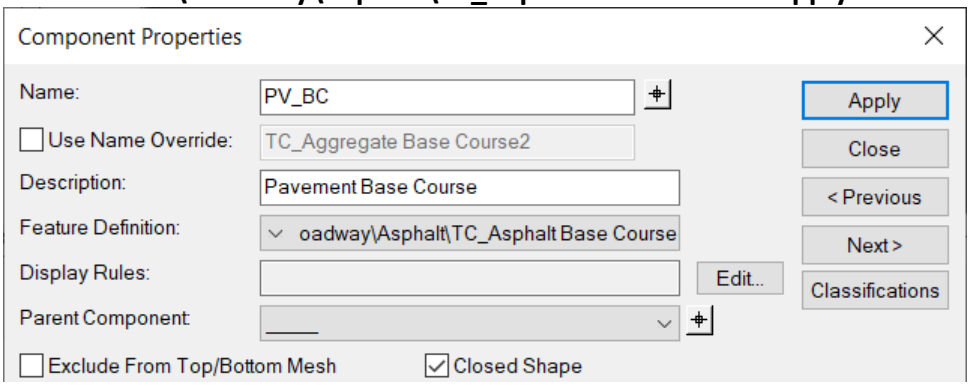
32. Edit the first layer component and name it **PV_SC** with a Feature definition of **Mesh\Roadway\Asphalt\TC_Asphalt Surface Course**. **Apply** and then **Close**.



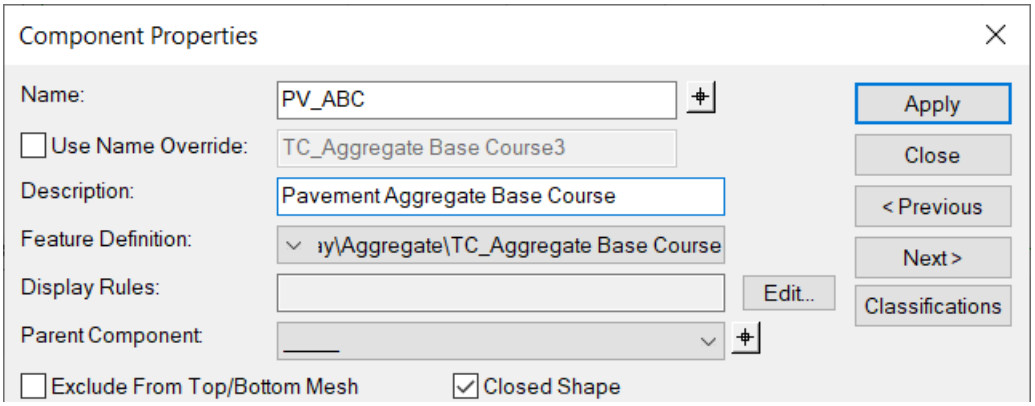
33. Edit the second layer component and name it **PV_IC** with a Feature definition of **Mesh\Roadway\Asphalt\TC_Asphalt Intermediate Course**. **Apply** and then **Close**.



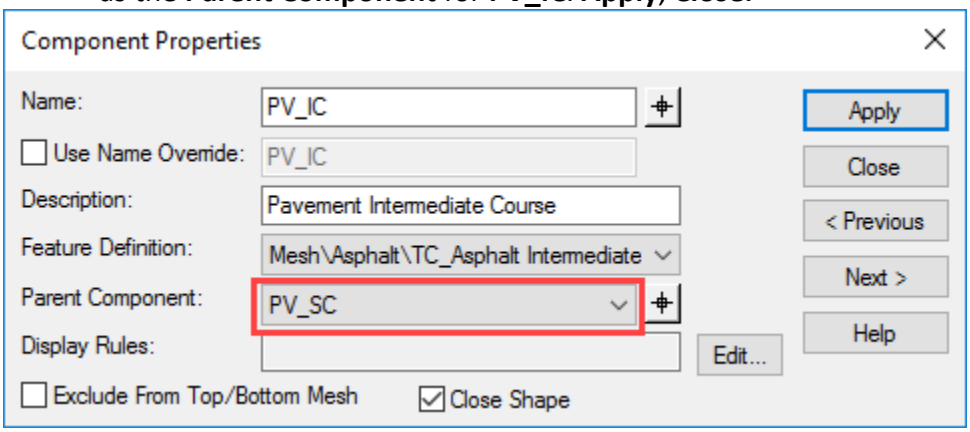
34. Edit the third layer component and name it **PV_BC** with a Feature definition of **Mesh\Roadway\Asphalt\TC_Asphalt Base Course**. **Apply** and then **Close**.



35. Edit the fourth layer component and name it **PV_ABC** with a Feature definition of **Mesh\Roadway\Aggregate\TC_Asphalt Base Course**. **Apply** and then **Close**.

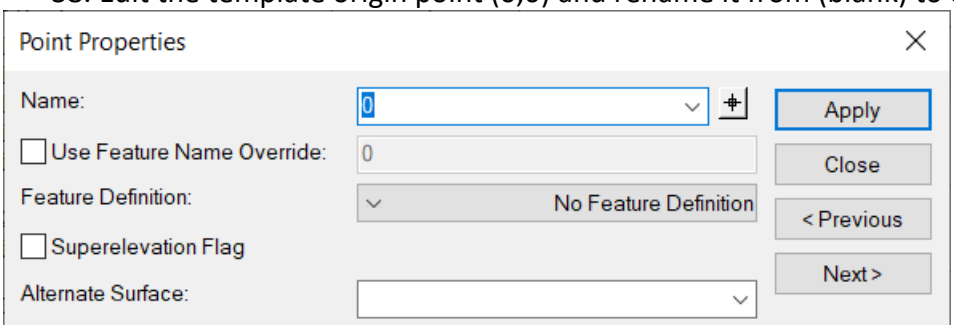


36. To establish a parent-to-child component relationship, edit the second layer and assign **PV_SC** as the **Parent Component** for **PV_IC**. **Apply**, **Close.**



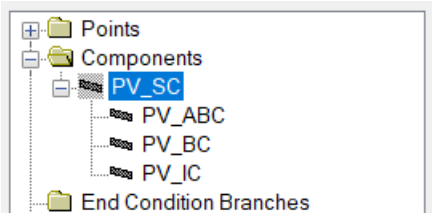
37. Repeat the same procedure and make the surface course **PV_SC** the parent component for the third and fourth pavement layer.

38. Edit the template origin point (0,0) and rename it from (blank) to **0** (zero).



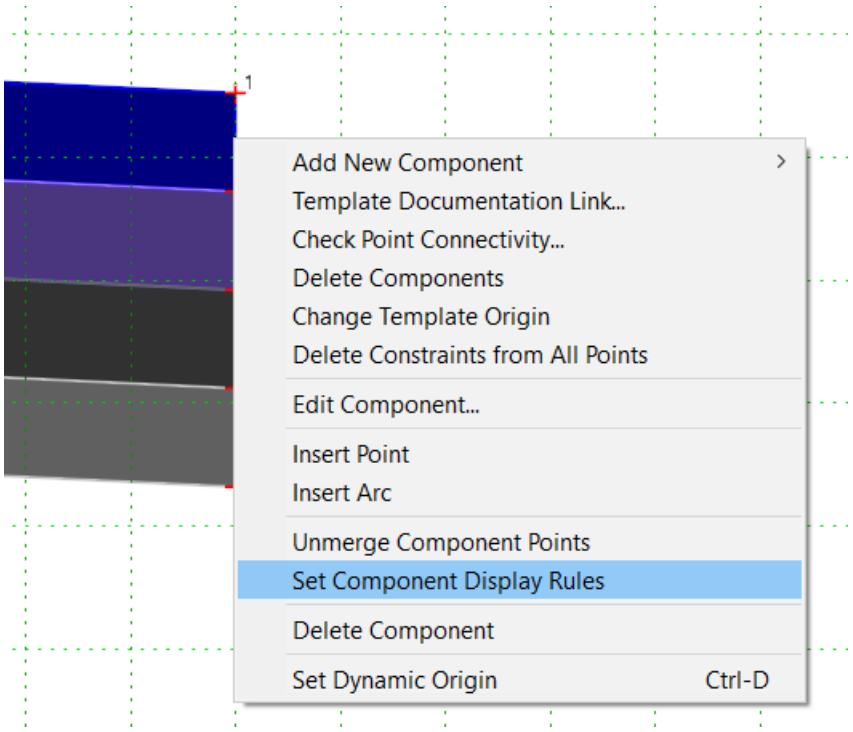
39. Save the template library (**ITL**).

40. Click the **Active Template** tab and open the **Components** folder. Verify the parent-to-child structure hierarchy.

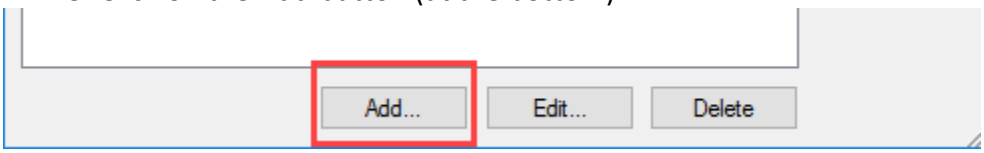


41. Since **PV_SC** is the parent component for the other layers, a display rule can be written to turn it **ON** or **OFF** and the rest of the children pavement layers.

42. Right-mouse click on the parent component **PV_SC** and select **Set Component Display Rules**.

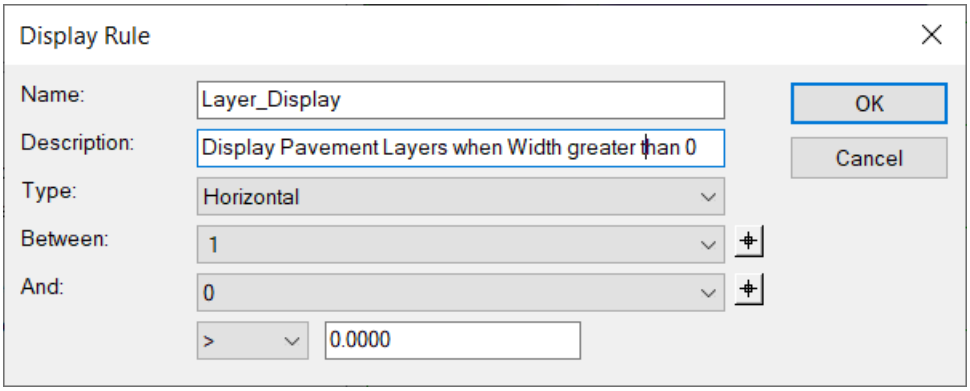


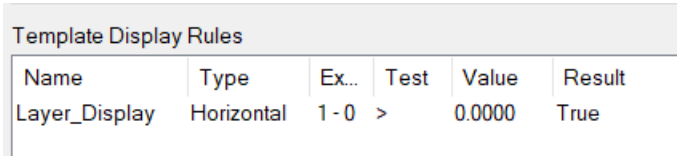
43. Click on the **Add** button (at the bottom).



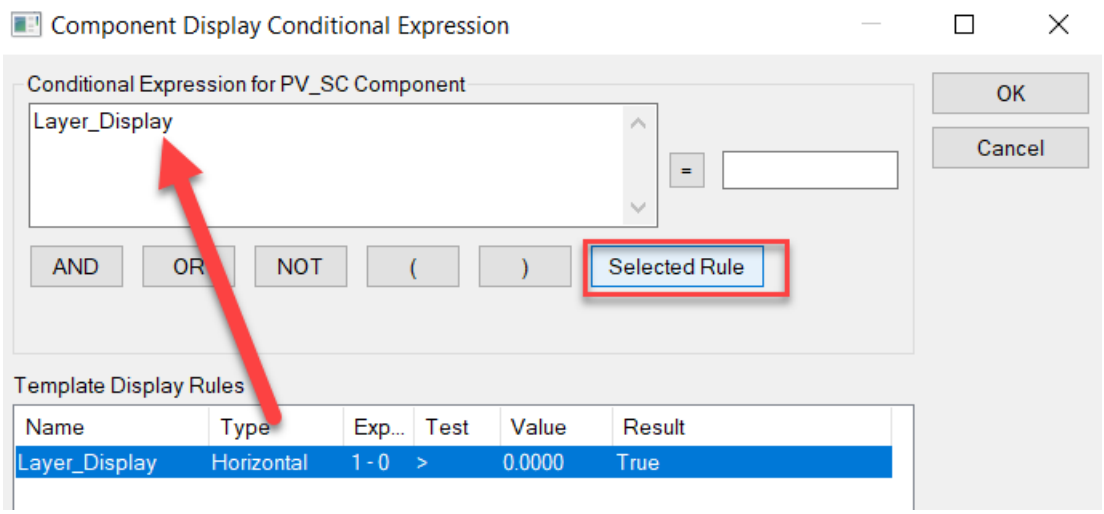
44. An equation can be written to turn **OFF** the first layer component (PV_SC) when the pavement width is zero.

Name: **Layer_Display**
Description: Display Pavement Layers when Width greater than 0
Type: **Horizontal**
Between: **1**
And: **0**
Operator: **>**
Value: **0.0000**

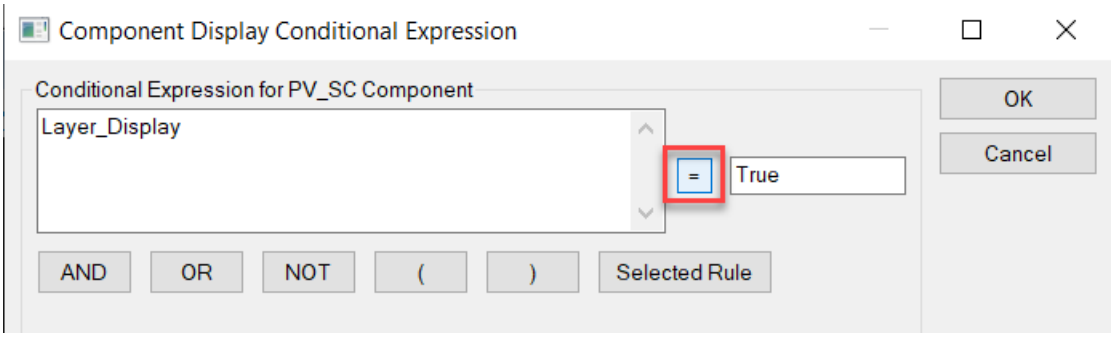




45. After the display rule **Layer_Display** has been created and added to **Template Display Rules** field, select it and click on the **Selected Rule** button to add it to the **Conditional Expression** for **PV_C1** Component field.

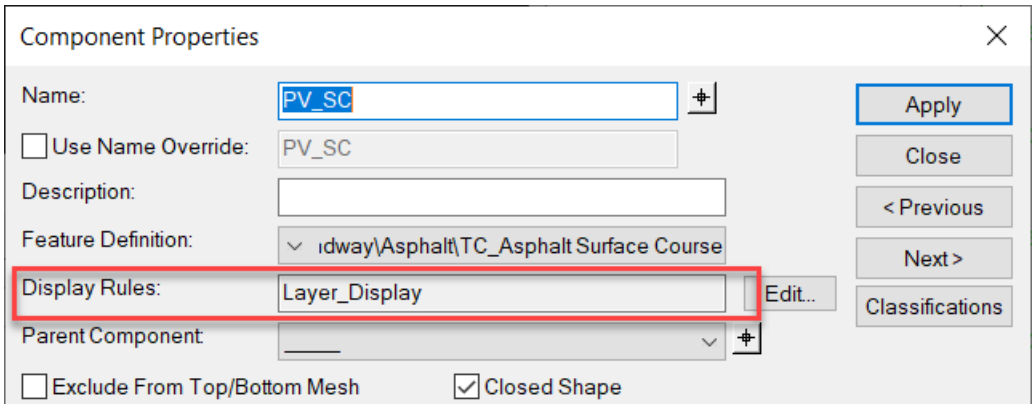


46. The "=" button to the right is to test the Display Rule(s) at its current default condition. If the result is **True**, then the pavement layers are turned on (by default). It is a way to verify the display rule at the default condition.



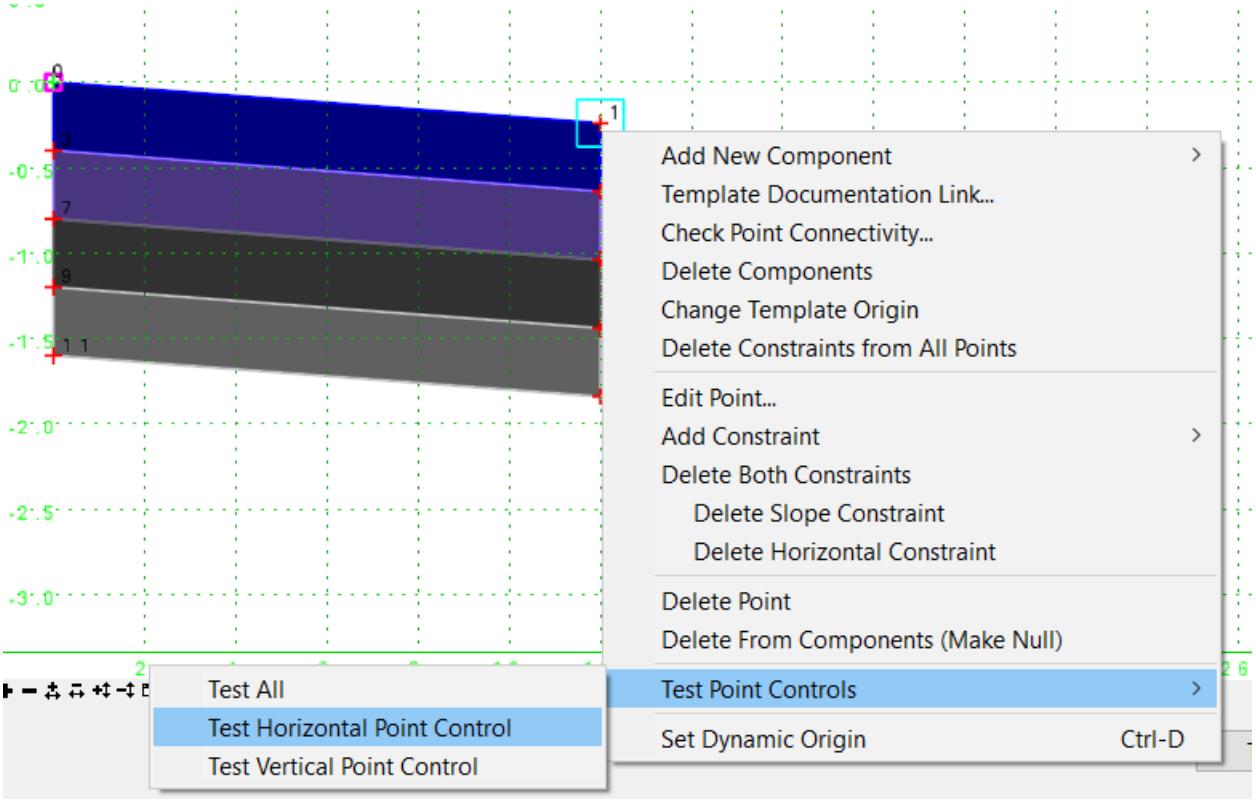
47. Click **OK** to close the dialog box.

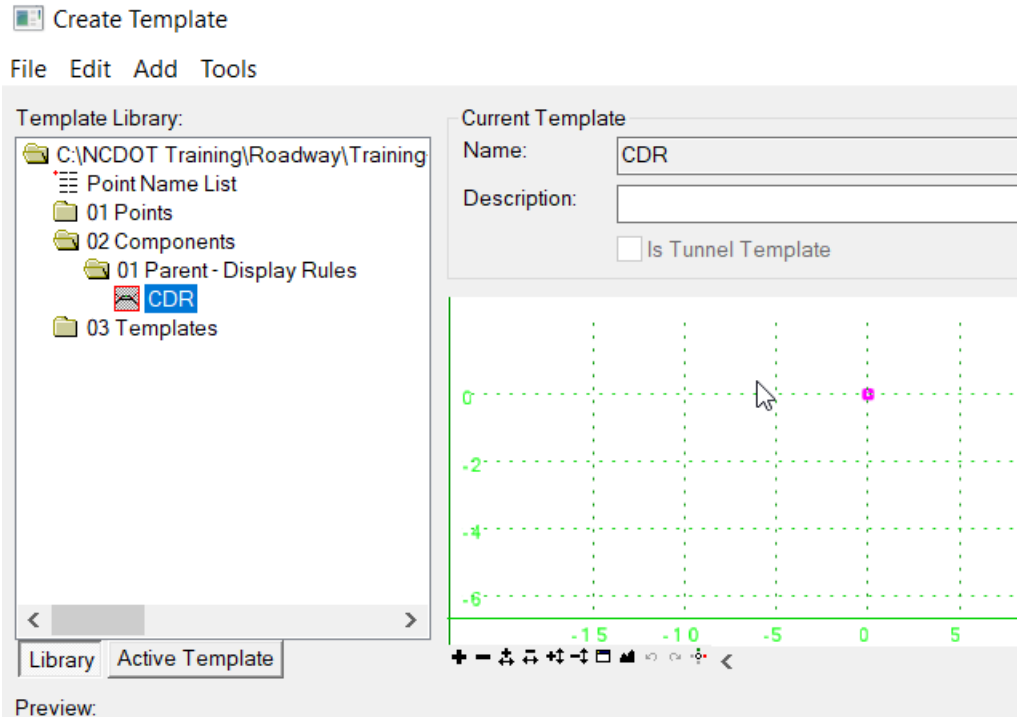
48. Verify the Display Rule is set in the **PV_SC** component properties dialog box.



49. **Save** the template library.

50. To test if the display rule is working as design, right mouse click on point **1** and select Test **Point Controls >>> Test Horizontal Point Controls**. Move point **1** across over and to the left of point **0**. Note the pavement layers are turned off during the crossover.



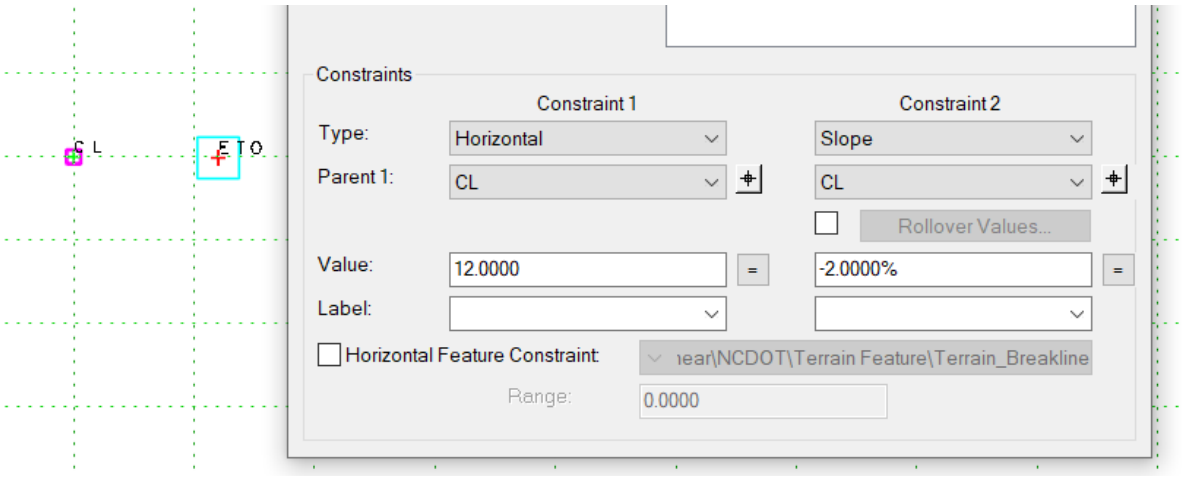

Create Template
File Edit Add Tools
Template Library:
C:\NCDOT Training\Roadway\Training
Point Name List
01 Points
02 Components
01 Parent - Display Rules
CDR
03 Templates
Library
Active Template
Current Template
Name:
CDR
Description:
Is Tunnel Template
0
-2
-4
-6
-15
-10
-5
0
5
Preview:

Exercise C3: Pavement Wedging Overlay Components

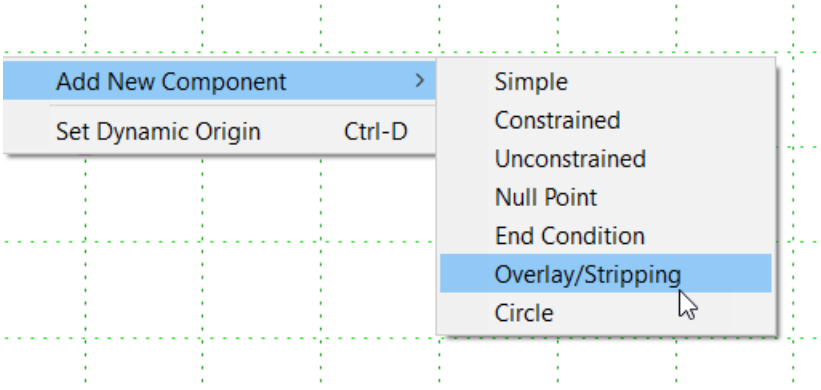
In this exercise we will demonstrate how pavement wedging works. The topics covered in this exercise include **Overlay/Stripping** (milling) components and their proper settings.

In the Template Library **Open** the **02 Components\03 Overlay Components** folder and create a new template named **Wedge Pavement Layers.** Use the overlay/stripping components to create three wedge layers. Test the template to see how each wedge layer is affected by the surface.

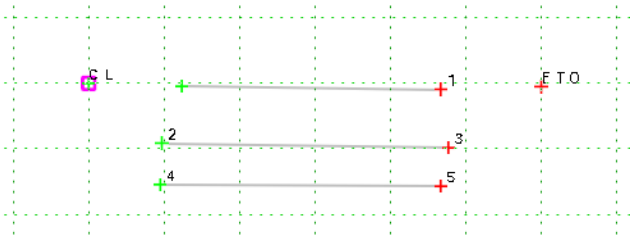
14. Create two (2) **Null Points** representing the **CL** and **ETO**. Make the **CL** point the template origin while the **ETO** point is constrained **12.0000'** horizontally to it and have a slope of **-2.0000%** from the **CL** point.



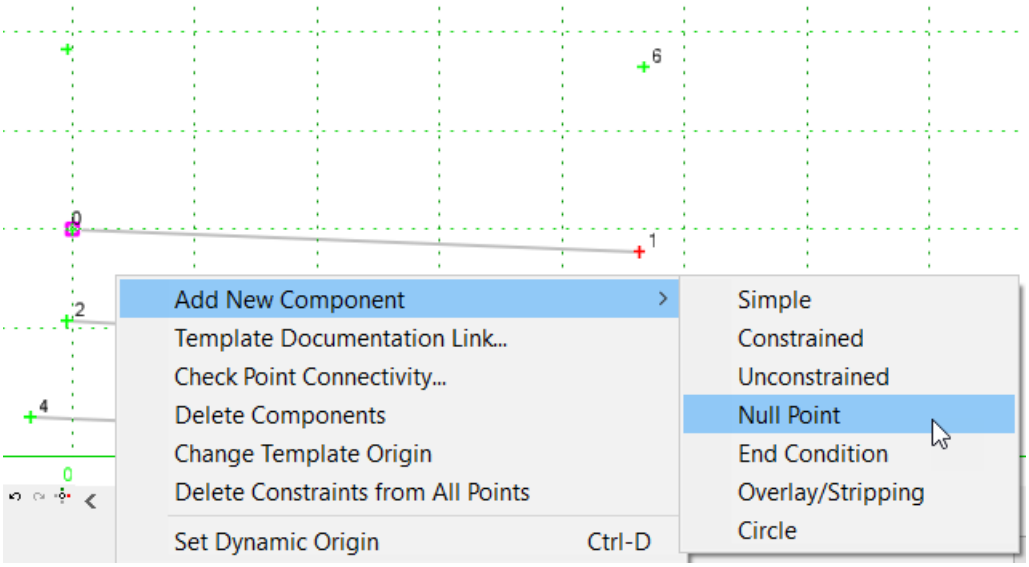
15. In the template **Pavement Wedge Layers** between the CL and **ETO** points, right-mouse click on the **Current Template** screen and choose **Add New Component** >>> **Overlay/Stripping**.



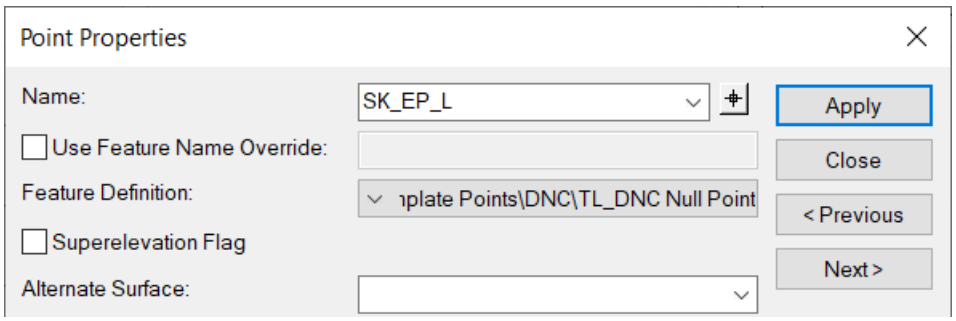
16. Draw three overlay component lines like the picture below. The horizontal and vertical distance between points does not matter at this time. We will constrain them later on in this exercise.



17. Create two (2) **Null Points** to control the width of wedging layers. Right-mouse click on the **Current Template** screen and choose **Add New Component** >>> **Null Point**. Place a null point near the top of **0** and another null point near the top of point **1**.

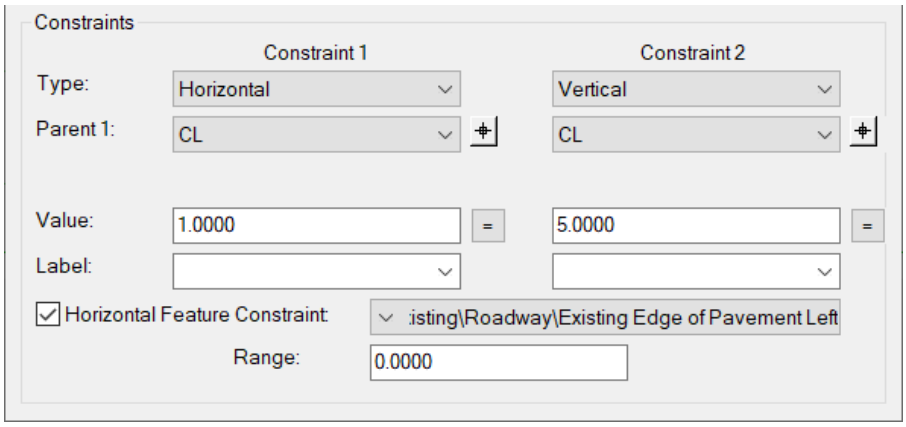


18. Rename the null point above point **0** to **SK_EP_L** and set the feature definition to **Linear\Roadway\Template Pints\DNC\TL_DNC Null Point**.



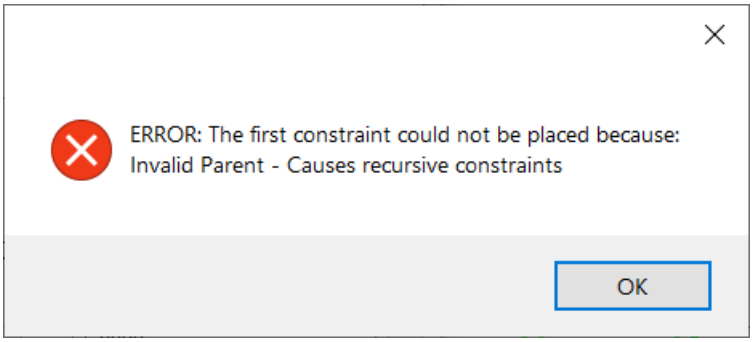
Note the **DNC** stands for "Do Not Construct". It is used mainly for linear elements drawn on Construction Class.

19. Constrain the **SK_EP_L** point to the CL point like the picture below. Check on the **Horizontal Feature Constraint** and set the feature definition **"Linear\Roadway\Existing\Roadway\Existing Edge of pavement Left"** to with a **Range** of **0** (zero).

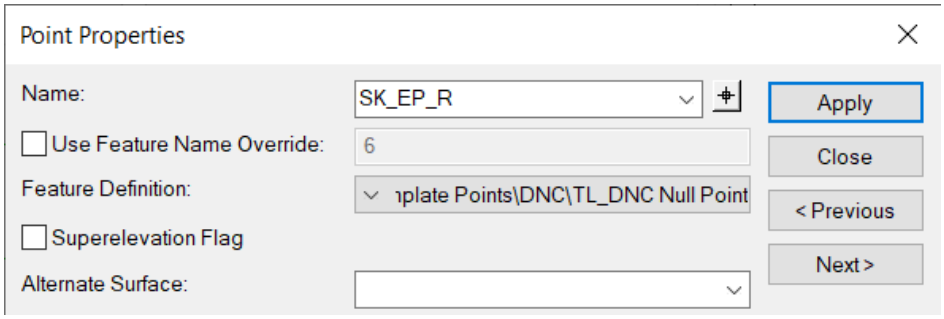


20. **Apply, Close** the Point Properties dialog box and **Save** template library.

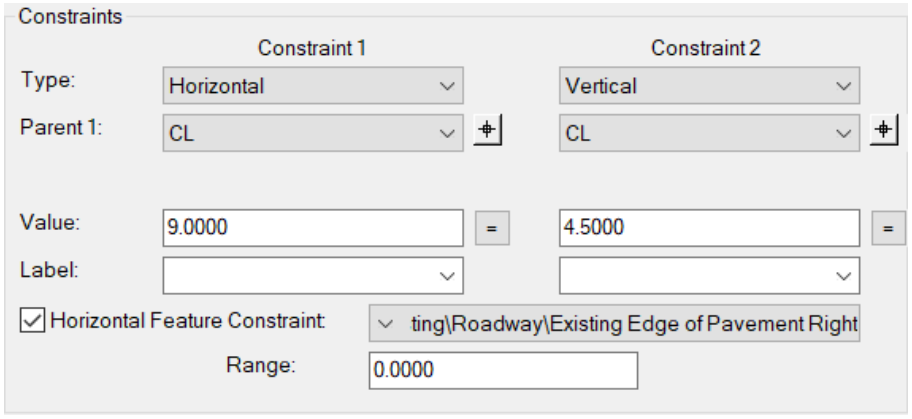
When constrain a point that is already constraint indirectly to another constrained point a **Recursive Error** can occur.



21. Rename the null point above point **1** to **SK_EP_R** and set the feature definition to **Linear\Roadway\Template Pints\DNC\TL_DNC Null Point**.

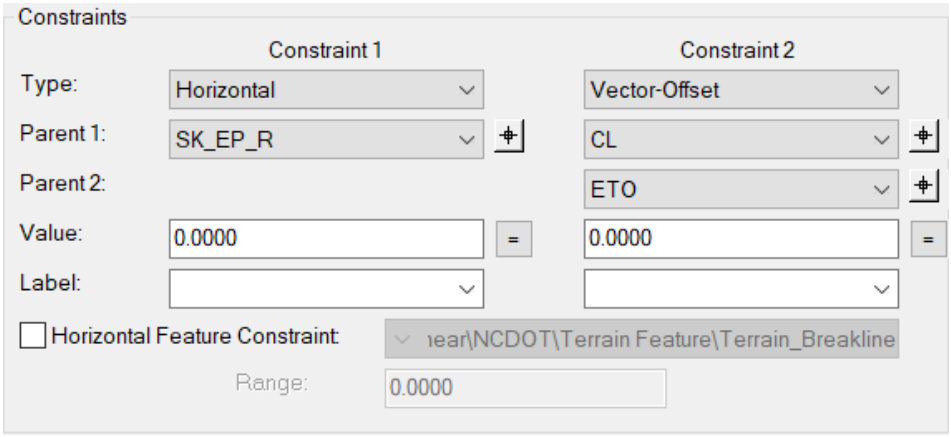


22. Constrain the **SK_EP_R** point to the **CL** point like the picture below. Check on the **Horizontal Feature Constraint** and set the feature definition **"Linear\Roadway\Existing\Roadway\Existing Edge of pavement Right"** to with a **Range** of **0** (zero).

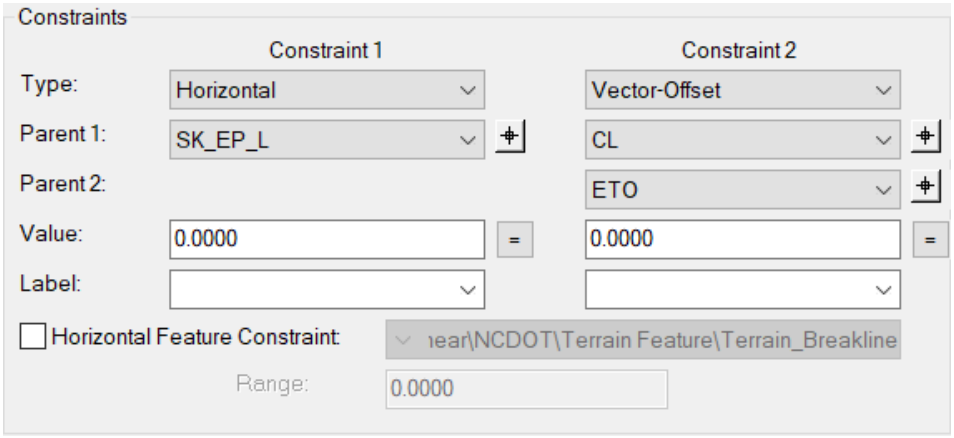


23. **Apply, Close** the Point Properties dialog box and **Save** template library.

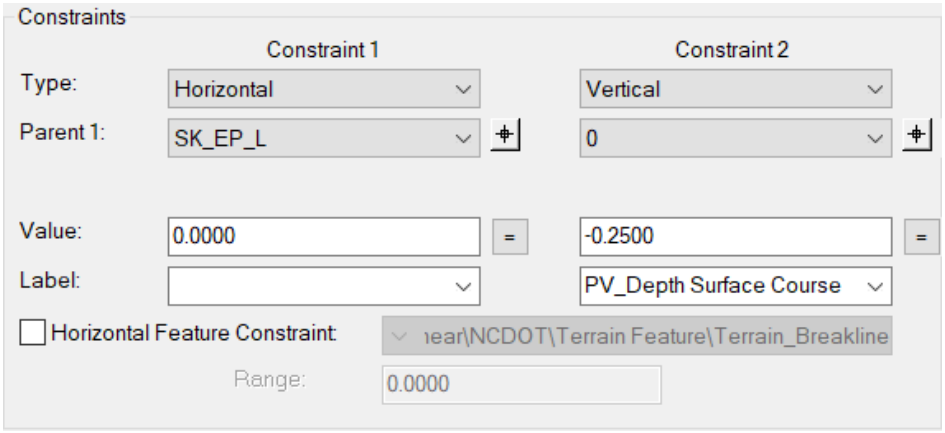
24. Constrain point **1** horizontally **0.0000'** from the point **SK_EP_R**. With the second constraint, create a Vector-Offset of **0.0000'** from the **CL** to the **ETO** point.



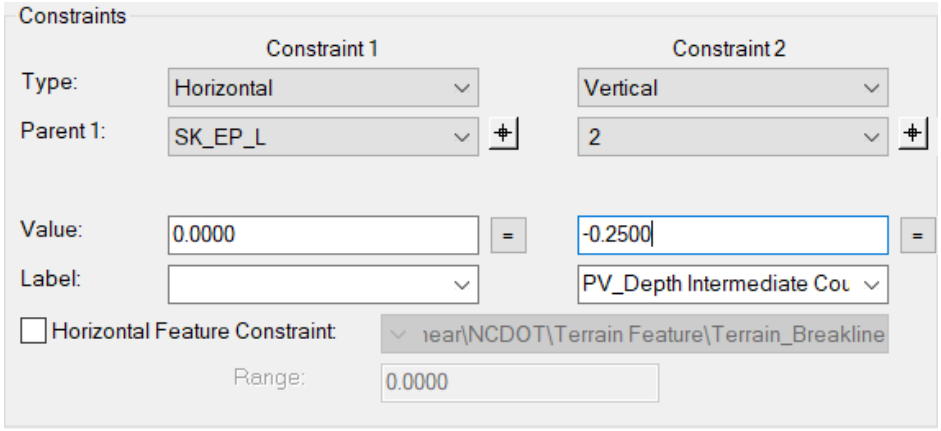
25. Constrain point **0** horizontally **0.0000'** to the **SK_EP_L** point. With the second constraint, create a Vector-Offset of **0.0000'** from the **CL** to the **ETO** point.



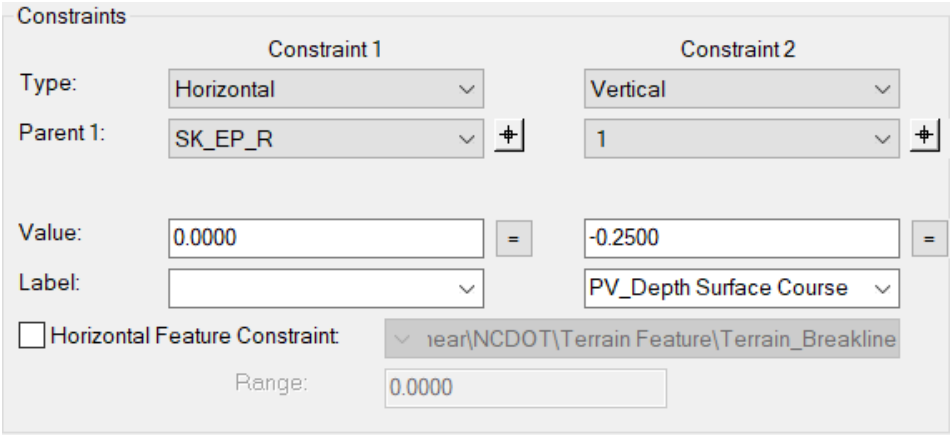
26. Constrain point **2** horizontally 0' to the **SK_EP_L** point. With the second constraint, constrain it vertically **-0.2500'** (surface course depth) below point 0. Key-in **PV_Depth Surface Course** as the parametric constraint label for the **Vertical** constraint.



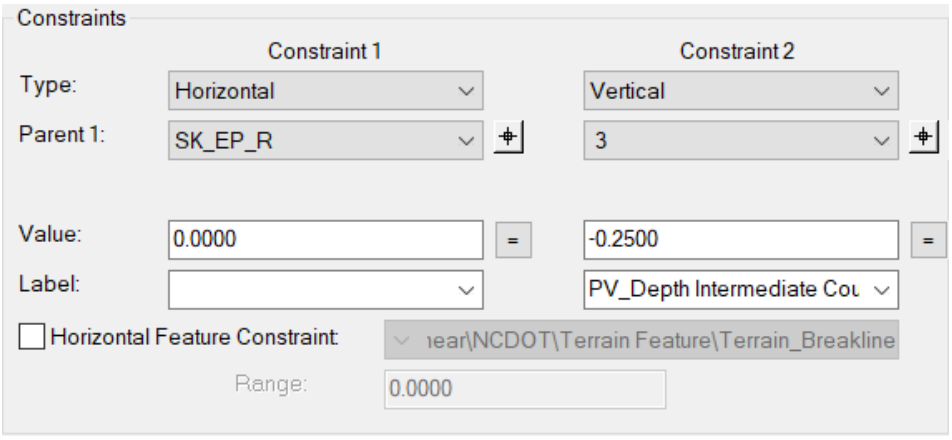
27. Constrain point **4** horizontally 0’ to the **SK_EP_L** point. With the second constraint, constrain it vertically **-0.2500’** (Intermediate course depth) below point **2**. Key-in **PV_Depth Intermediate Course** as the parametric constraint for the V**ertical** constraint.



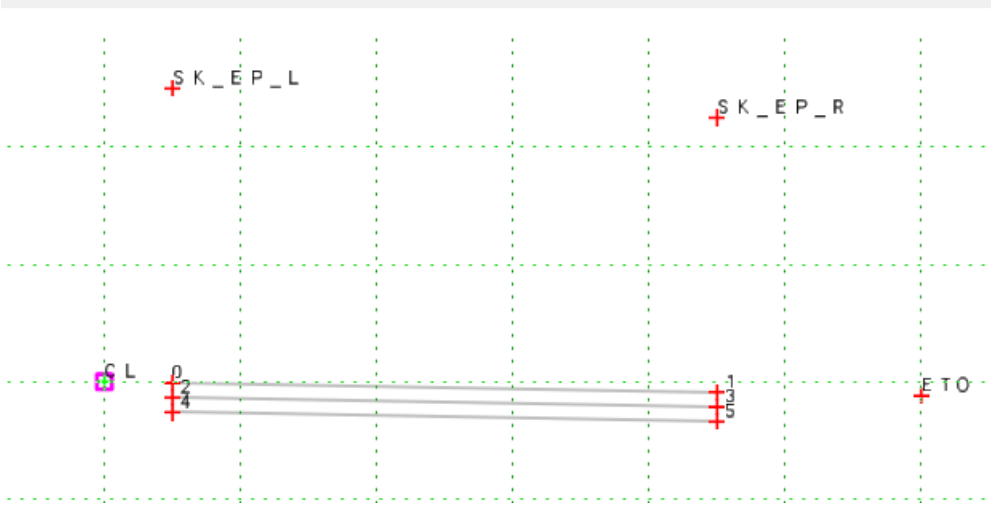
28. Constrain point **3** horizontally **0.000’** to the **SK_EP_R point**. With the second constraint, constrain it vertically **-0.2500’** below point **1**. Select **PV_Depth Surface Course** as the parametric constraint label for the **Vertical** constraint.



29. Constrain point **5** horizontally 0’ to the **SK_EP_R point**. With the second constraint, constrain it vertically -0.2500’ below point **3**. Select **PV_Depth Intermediate Course** as the parametric constraint for the V**ertical** constraint.



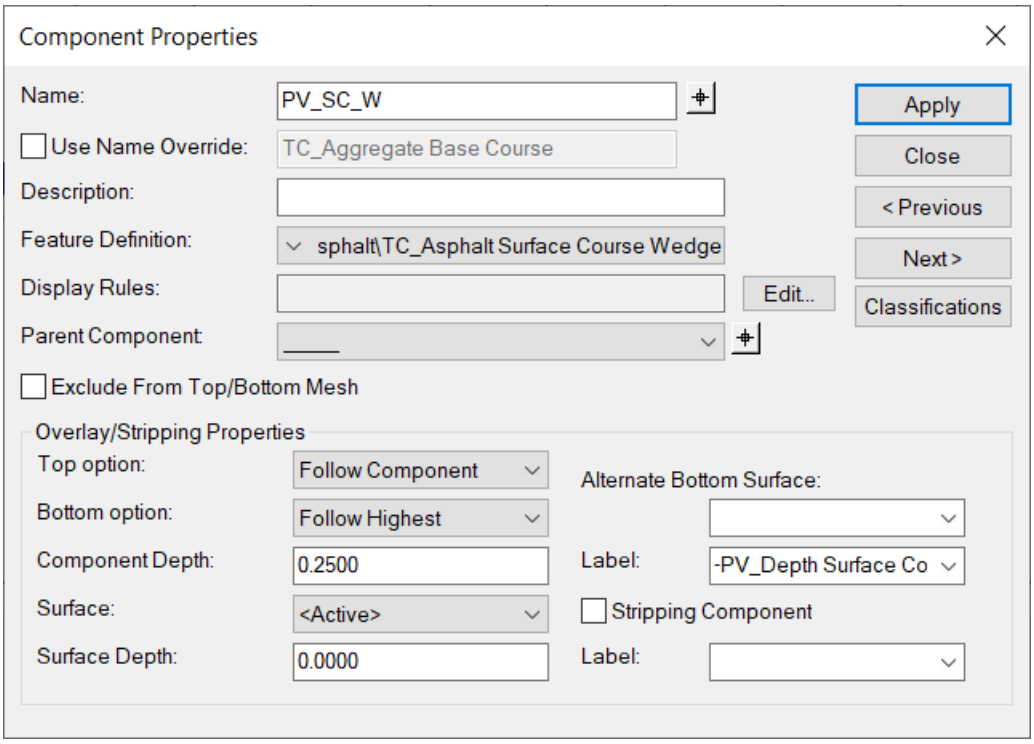
The overlay components should look like the picture below.



30. Edit the top overlay component. Name it **PV_SC W** with a feature definition of "**Mesh\Roadway\Asphalt\TC_Asphalt Surface Course Wedge**".

Set the following **Overlay/Stripping Properties**:

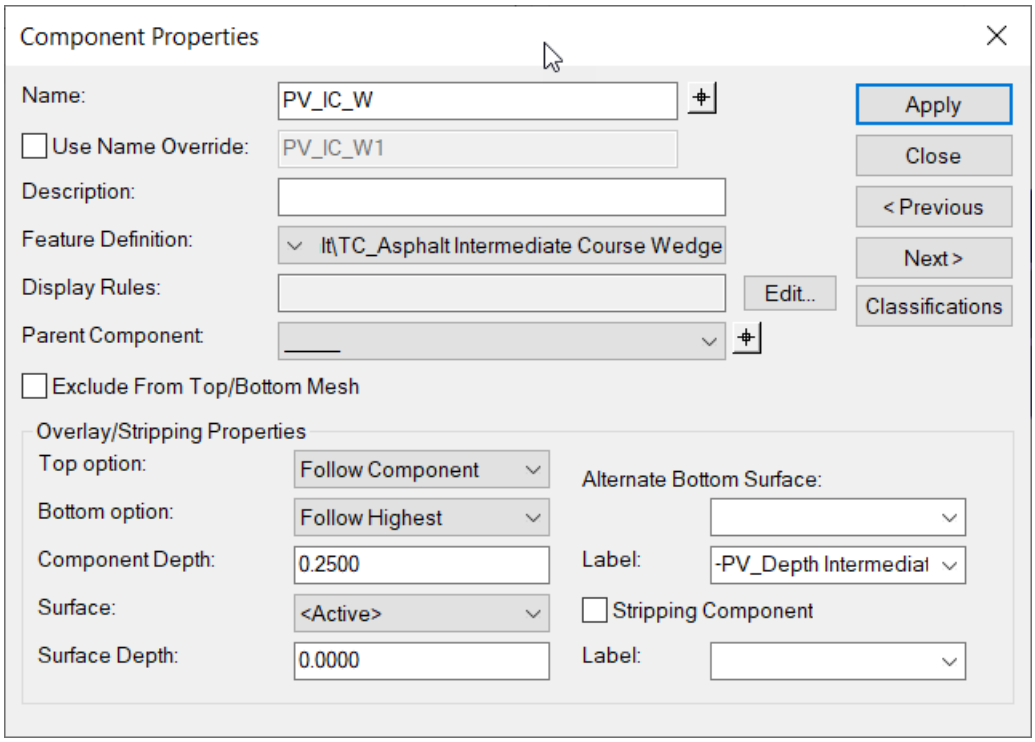
Top Option: **Follow Component**
Bottom Option: **Follow Highest**
Component Depth: **0.2500 (note this is a positive value)**
Label (Component Depth): -**PV_Depth Surface Course**



31. Edit the second overlay component. Name it **PV_IC W** with a feature definition of "**Mesh\Roadway\Asphalt\TC_Asphalt Intermediate Course Wedge**".

set the following **Overlay/Stripping** Properties:

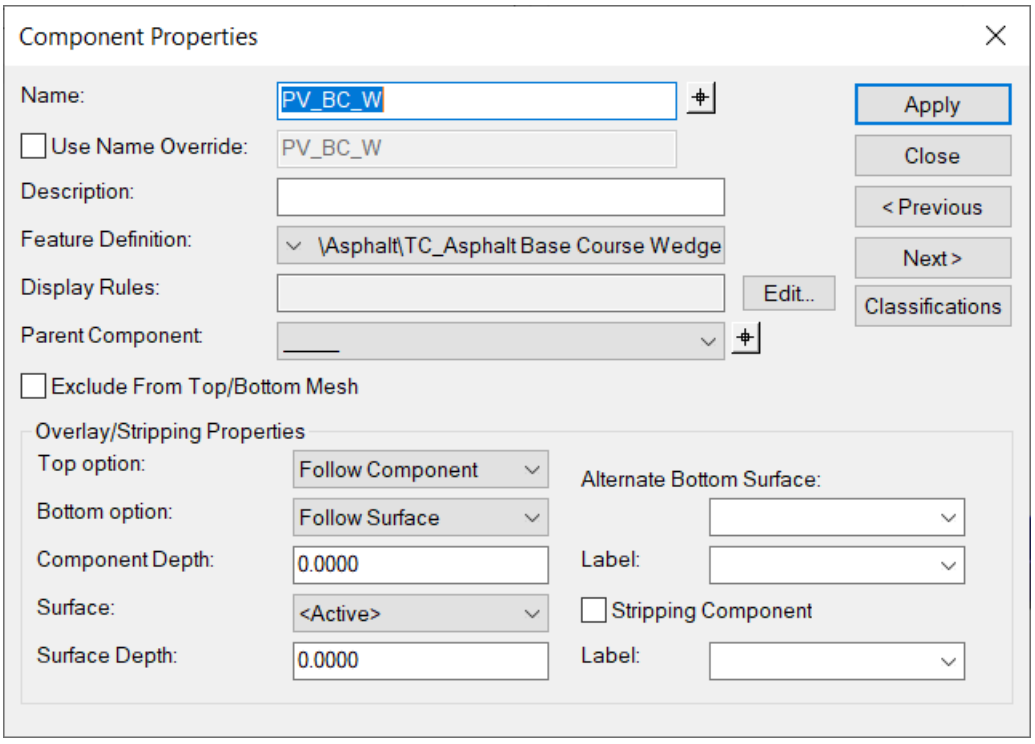
Top Option: **Follow Component**
Bottom Option: **Follow Highest**
Component Depth: **0.2500**
Label: **-PV_Depth Intermediate Course**



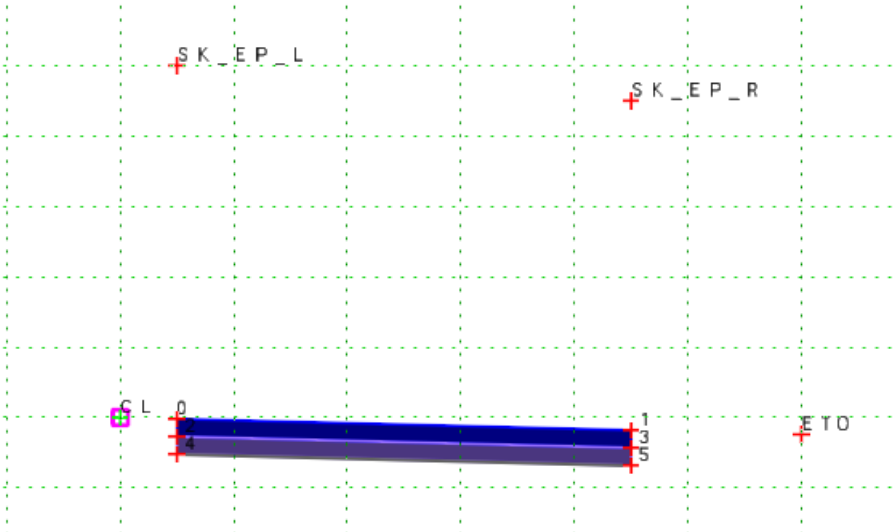
Edit the third overlay component. Name it **PV_BC W** with a feature definition of "**Mesh\Roadway\Asphalt\TC_Asphalt Base Course Wedge"**.

While editing the overlay component **PV_BC W**, set the following **Overlay/Stripping Properites**:

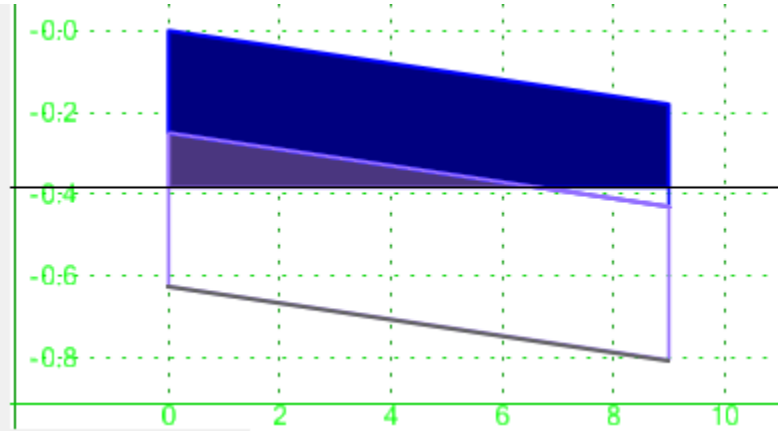
Top Option: **Follow Component**
Bottom Option: **Follow Surface** (note that this has unlimited depth for wedging)
Component Depth: **0.0000**



The current template should look like the below picture. Save the template.



Test how each wedge layer behaves as the existing ground is moving up and down on the screen.

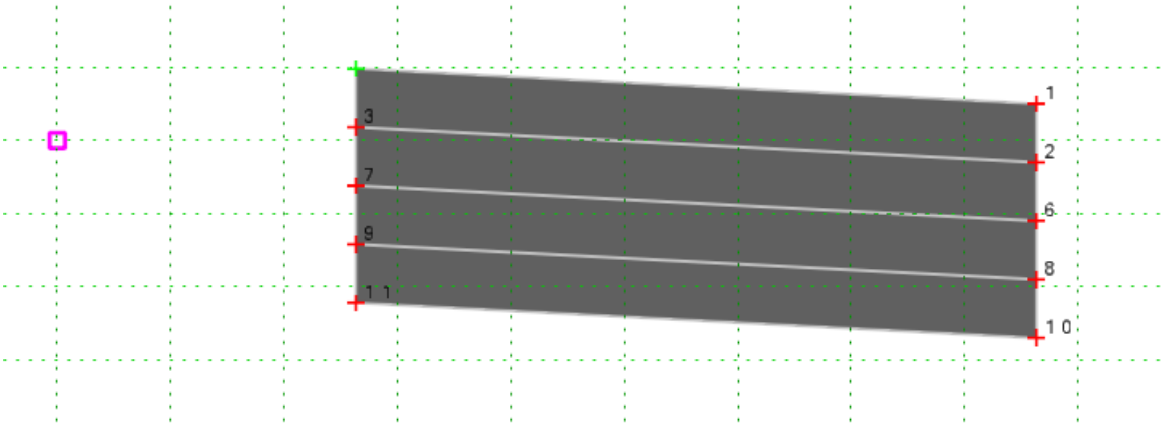


Exercise C4: Paved Shoulder Components

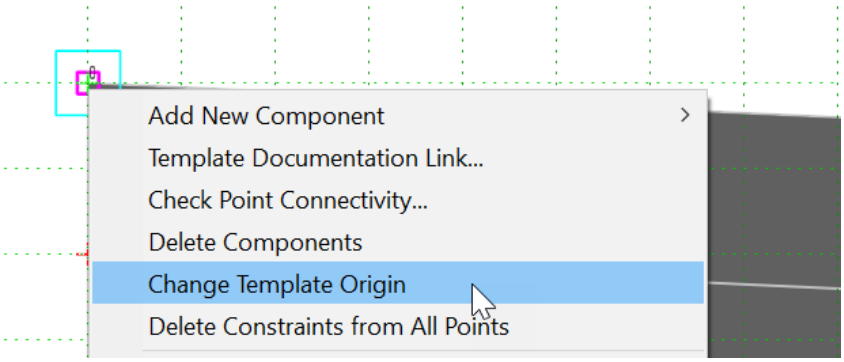
In this exercise we will go over how paved shoulder components are created and the difference between a 4’ and 10’ paved shoulder. The topics covered in this exercise include creating Null Points as a reference, assigning a **Parent Component,** writing **Equation**s for constraint values and applying **Rollover Locks** per standards.

In the Template Library **Open** the **02 Components\04 Paved Shoulders** folder and create a new template named **Paved Shoulder Rollovers**.

1. Create four (4) full depth paved shoulder (FDPS) components, on top of each other, using the **Add New Component >>> Simple** tool.

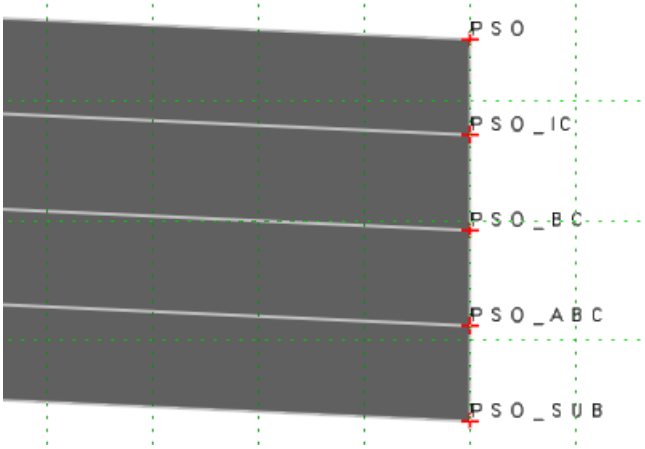


2. Rename the green blank point **0** (zero) and make it the template origin.

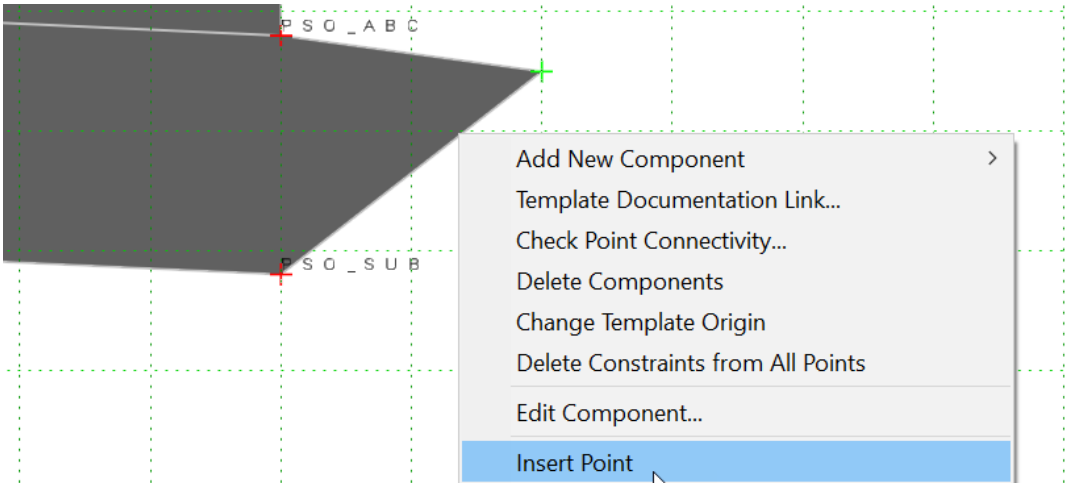


3. Starting at the top on the right side points, rename the following points by selecting it from the point name list.

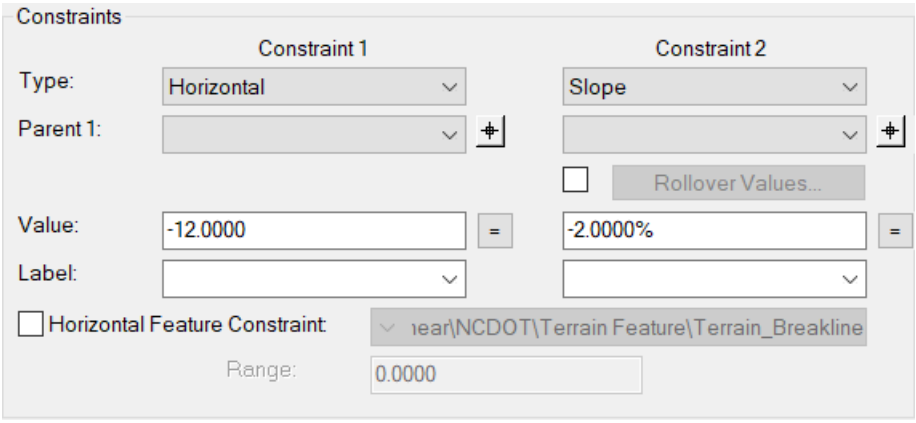
1	PSO
2	PSO_IC
6	PSO_BC
8	PSO_ABC
10	PSO_SUB



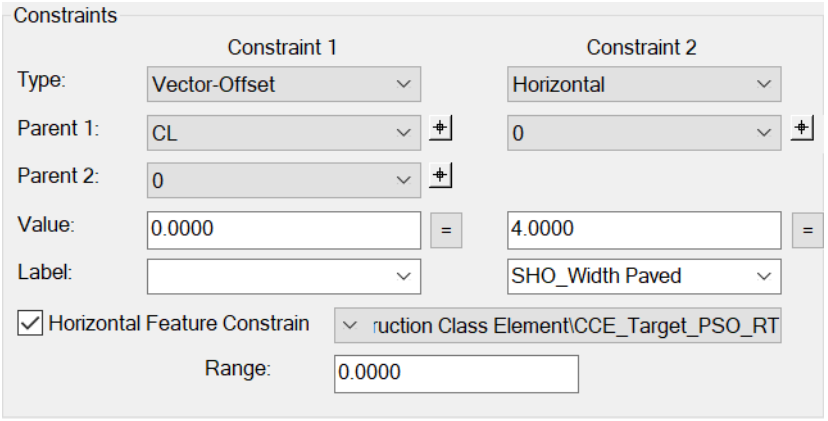
4. Insert a point between PSO_ABC and PSO_SUB an place it to the right of the PSO_ABC point.



5. Rename (select from point name list) the inserted point to **PSO_ABC-SH**.
6. Add a null point to the left of the template point and rename this point **CL**. Constrain this point as shown.

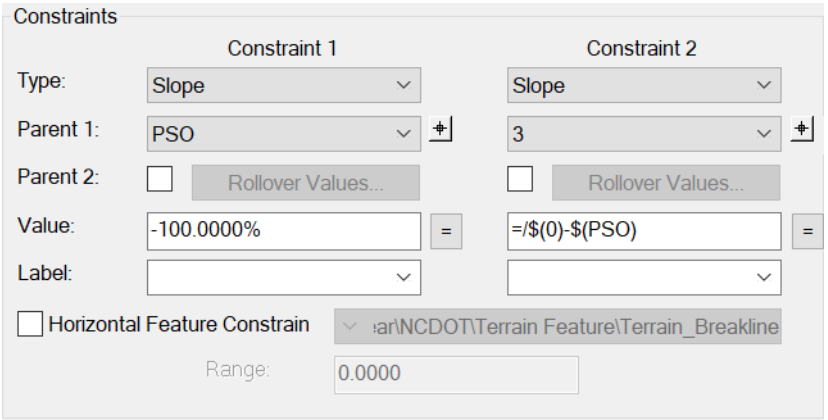


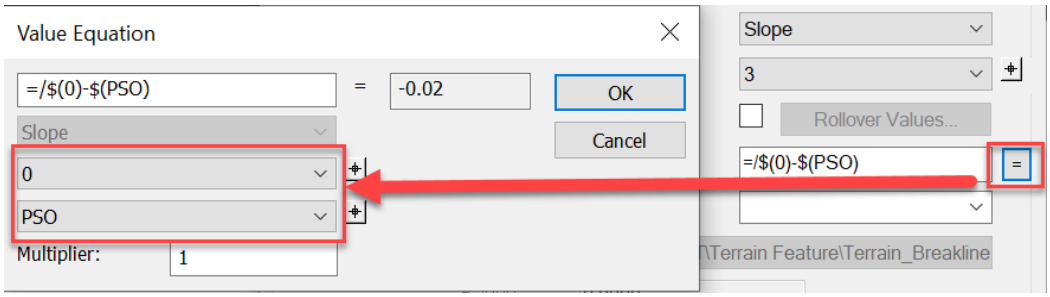
7. Edit the **PSO** point and constrain it as shown below. The **Horizontal Feature Constraint** is set to "**Linear\Roadway\Construction Class Element\CCE_Target_PSO_RT**".



Note point **0** (zero) is the template origin point.

8. Edit the **PSO_IC** Point and constrain it as shown below. Write an **equation** for the second **Slope** constraint using point **0** and **PSO** (slope of surface course).



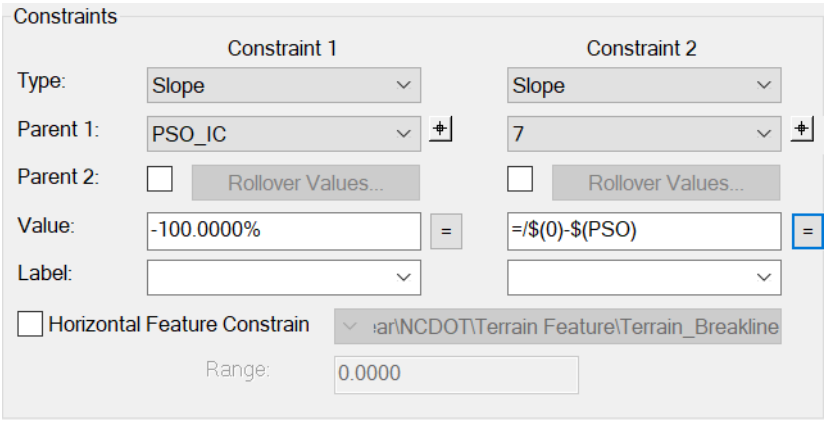


Equations

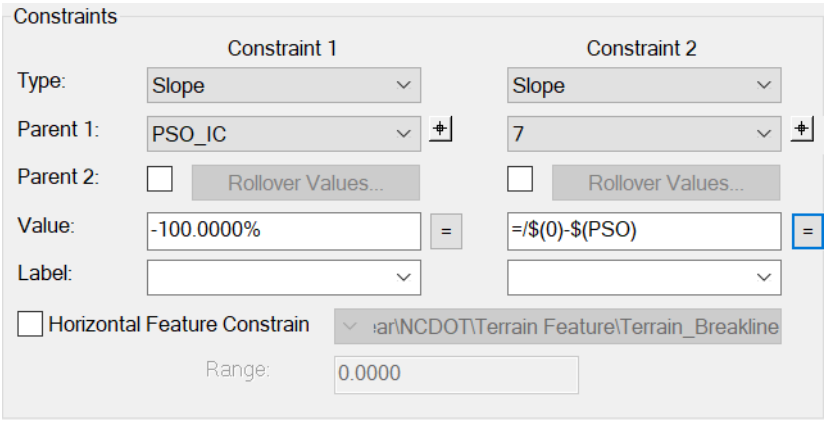
When creating an equation for a constrain value, first click on the "=" button on the right of the Value field. Then select the point to be included as part of the equation (slope between the blank template origin point and the **PSO** point). Equations are a form of a variable. Instead of fixed value, use the slope that is formed between two (2) points. Equations can be written for horizontal, vertical or slope value.

The **Multiplier** field is used to switch the desired signage of a value, such as (+)2% to -2% (-1). It can also be used to take half of a horizontal or vertical distance between two (2) points (0.5).

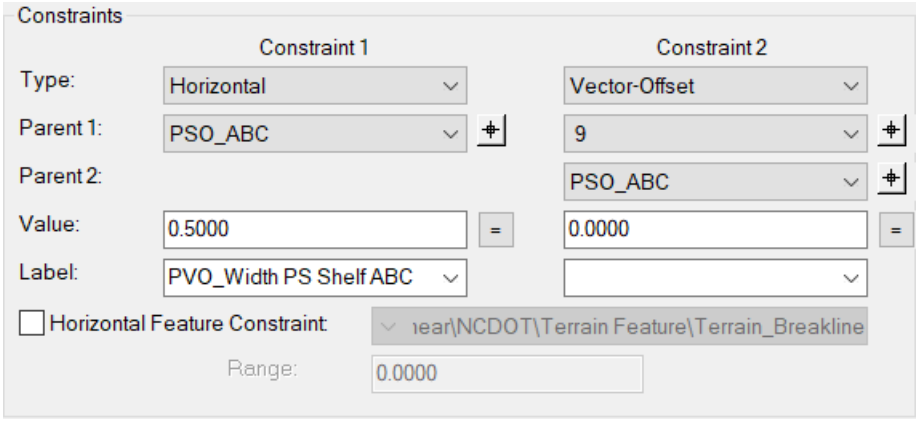
9. Edit the **PSO_BC** point and constrain it as shown below.



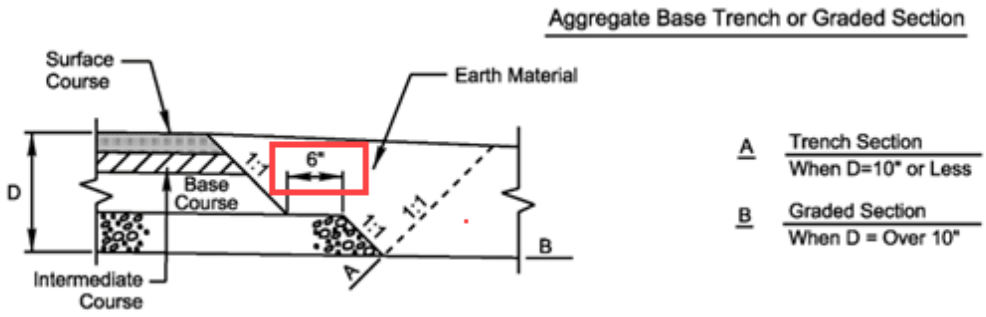
10. Edit the **PSO_ABC** point and constrain it as shown below.



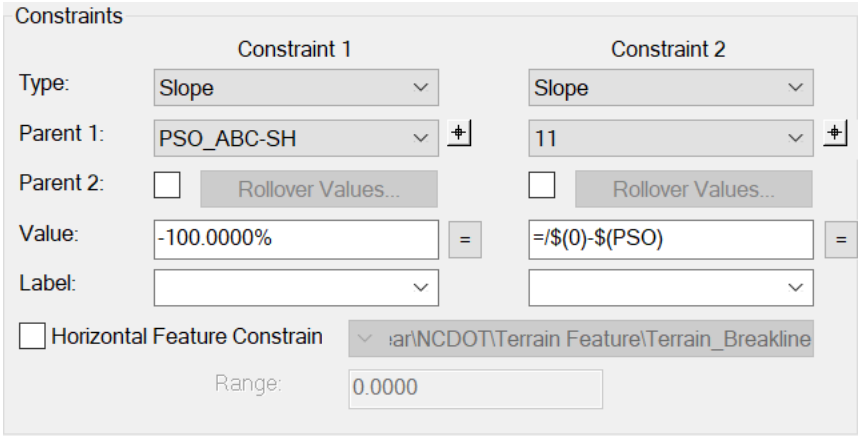
11. Edit the **PSO_ABC-SH** point and constrain it as shown below.



The **ABC** course requires a **6”** shelf distance (Roadway Design Manual)



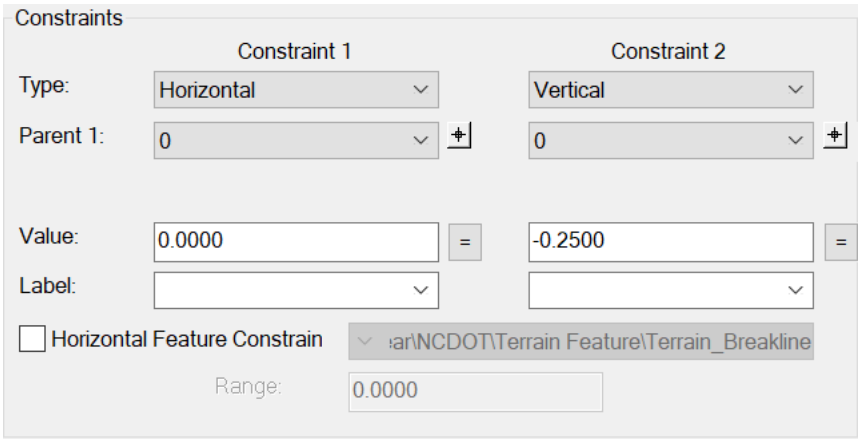
12. Edit the **PSO_SUB** point and constrain it as shown below.



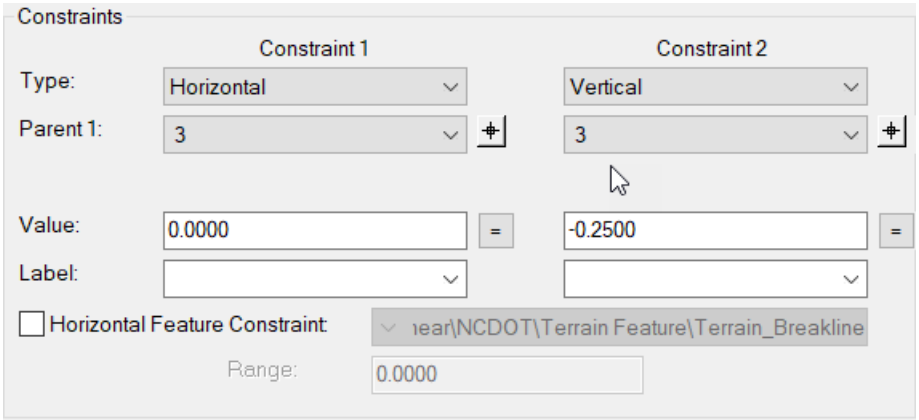
Merged Points

Note that it is not necessary to edit the point names on the left side. When the paved shoulder components are connected with the pavement components, the points will merge together and the properties of the pavement, such as name, feature definition and parametric constraint label, from the pavement is transferred to the pave shoulder. The important thing is making sure the vertical (pavement depth) and horizontal values match between the pavement and paved shoulder components for them to merge properly.

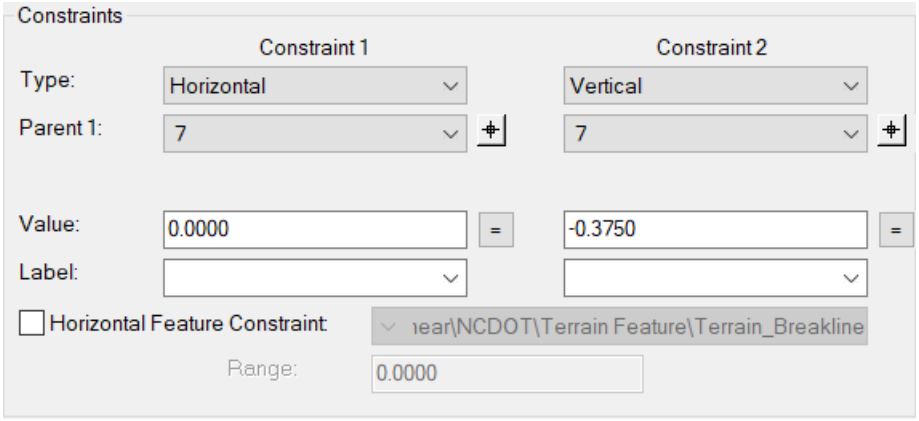
13. To ensure these paved points match with the pavement points vertically, starting with the second point from the top on the left side. Edit point **3** and constrain it as shown below.



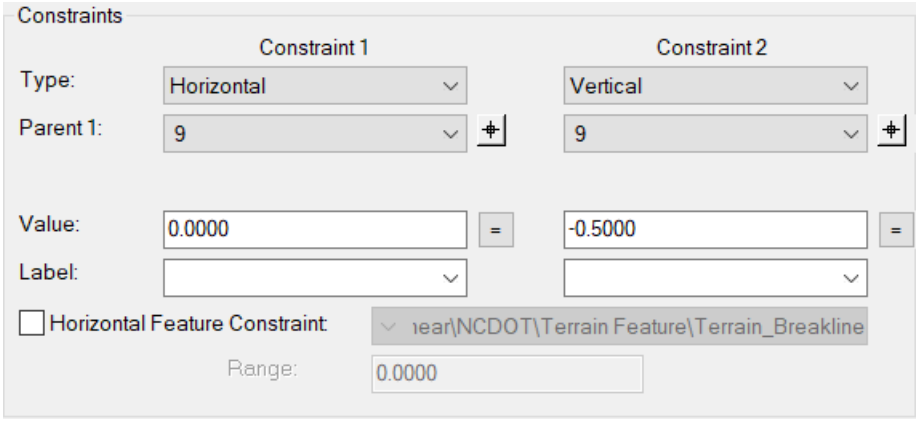
14. Edit point **7** and constrain it as shown below.



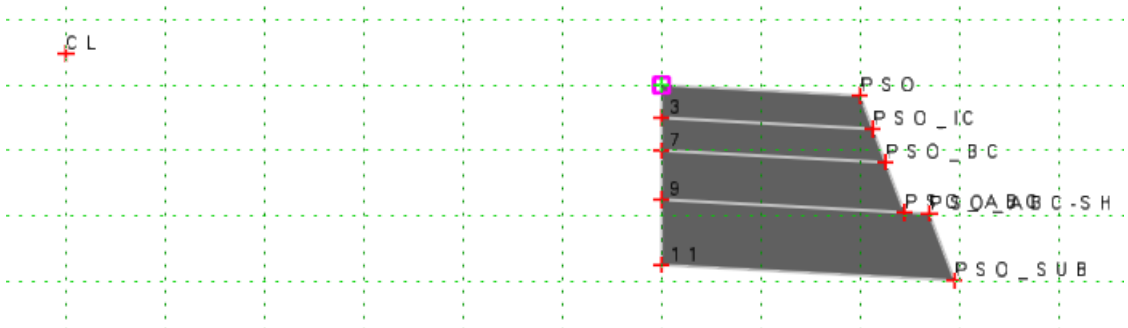
15. Edit point **9** and constrain it as shown below.



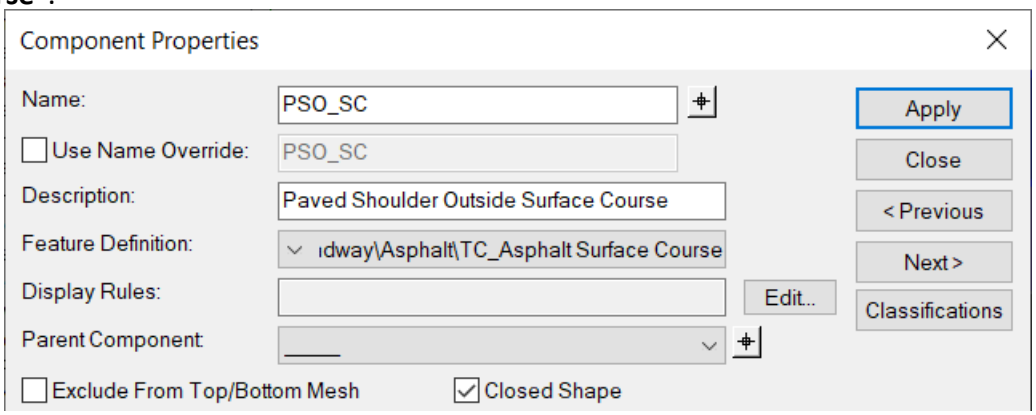
16. Edit point **11** and constrain it as shown below.



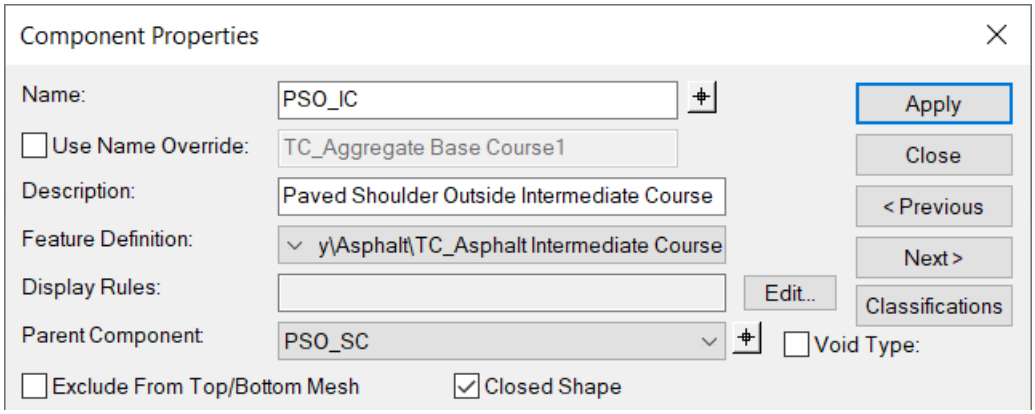
The paved shoulder components should look like the below picture.



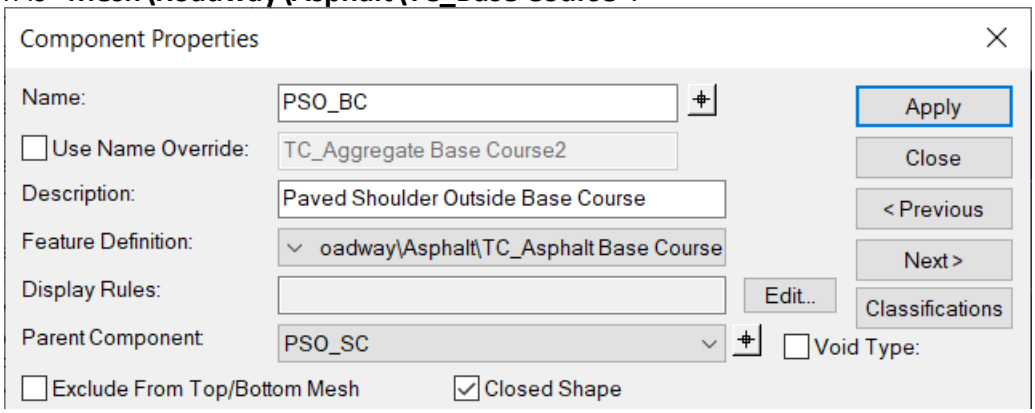
17. Starting with the top layer, edit the component properties, specifically name and feature definition, as shown below. Feature definition is "**Mesh\Roadway\Asphalt\TC_Surface Course**".



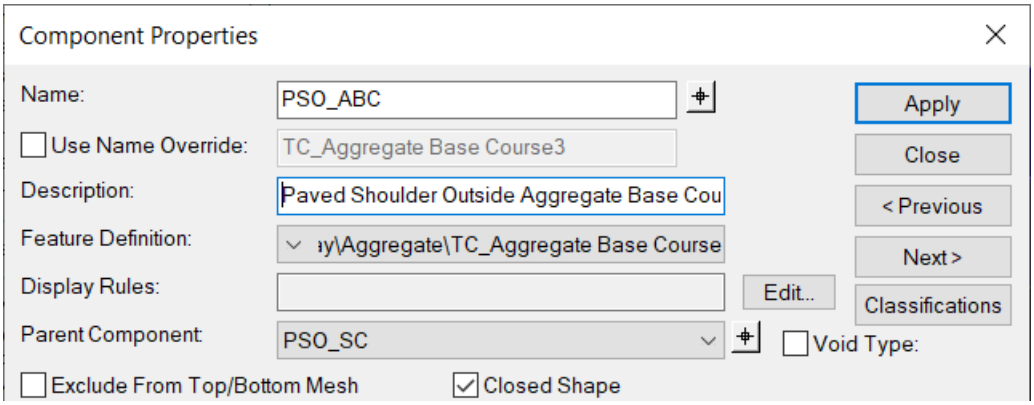
18. Edit the second layer from the top with the component properties as shown below. Feature definition is "**Mesh\Roadway\Asphalt\TC_Intermediate Course**". Be aware of the **Parent Component**.



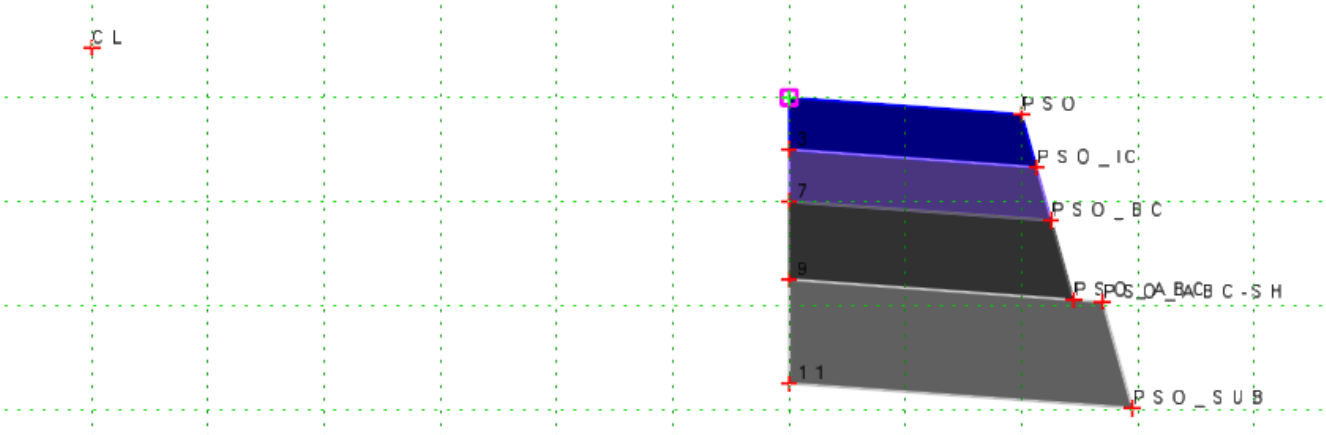
19. Edit the third layer from the top with the component properties as shown below. Feature definition is "**Mesh\Roadway\Asphalt\TC_Base Course**".



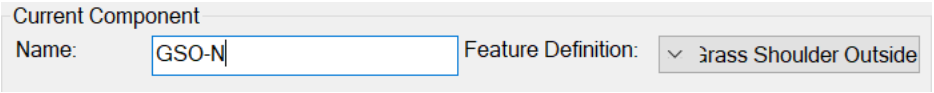
20. Edit the fourth layer from the top with the component properties as shown below. Feature definition is "**Mesh\Roadway\Aggregate\TC_Aggregate Base Course**".



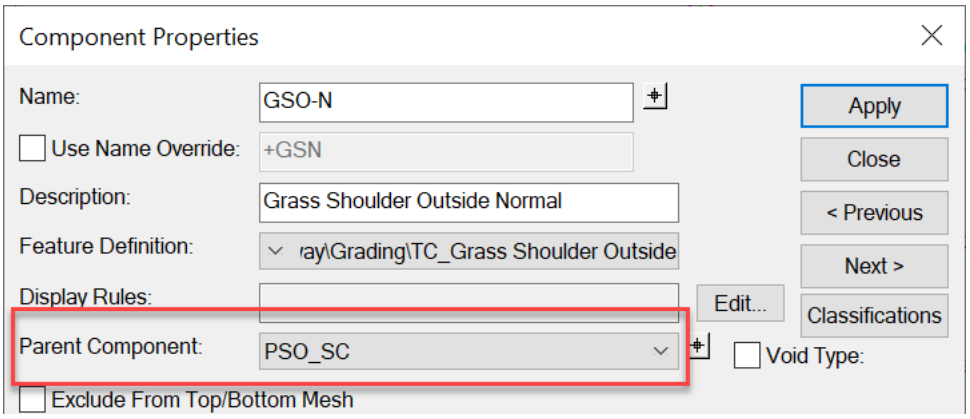
The paved shoulder components should look like the below picture.



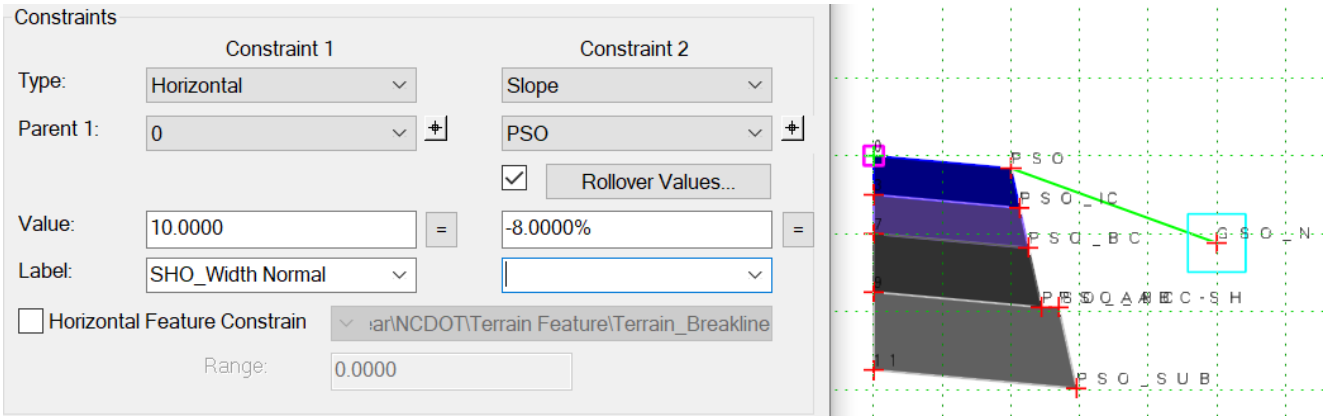
21. To complete paved shoulder component, create the turf/grass shoulder. **Add New Component >>> Constrained** and starting from point PSo draw a line to the right. Enter the following values in the component properties prior to placement. The Feature Definition is "**Mesh\Roadway\Grading\TC_Grass Shoulder Outside**".



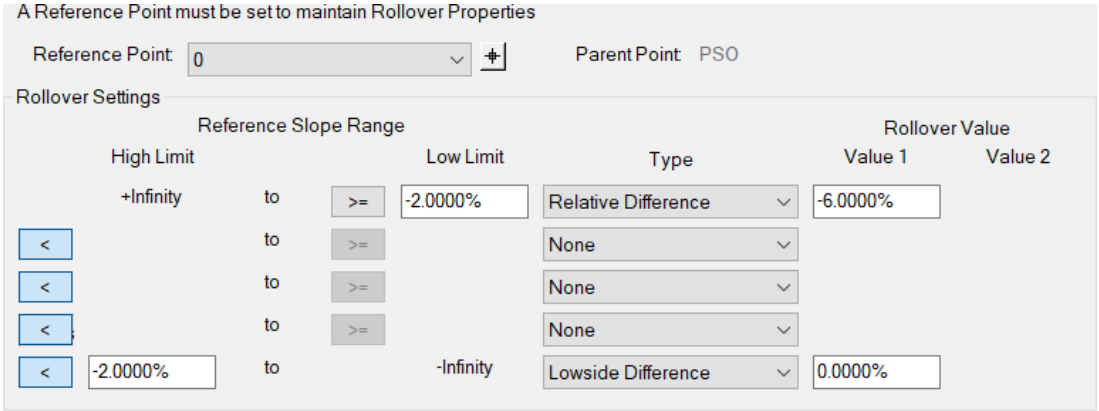
22. Make **PSO_SC** the **Parent Component**.



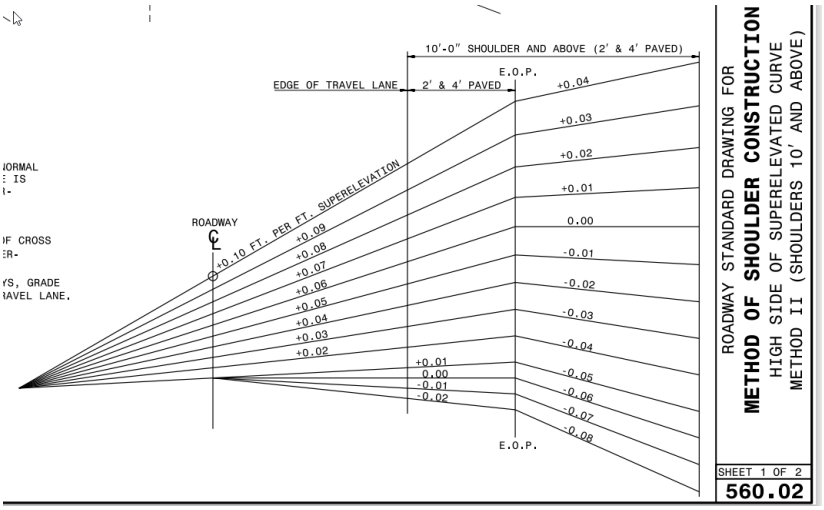
23. Edit the blank point and rename it **GSO_N**. Constrain it as shown below.



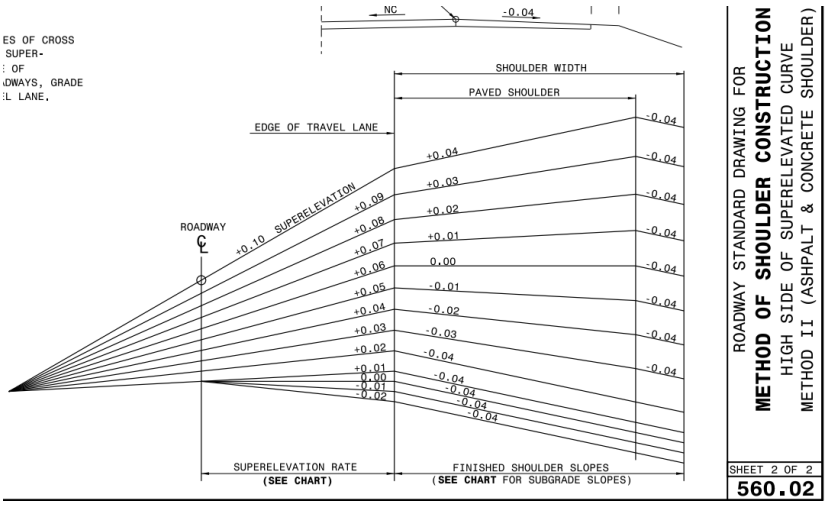
24. To account for the **6% rollover slope** on the high side, click the **Rollover Value** check box and set the following values in the **Rollover Point Properties**. The **Rollover Value Properties** will be explained later in this manual.



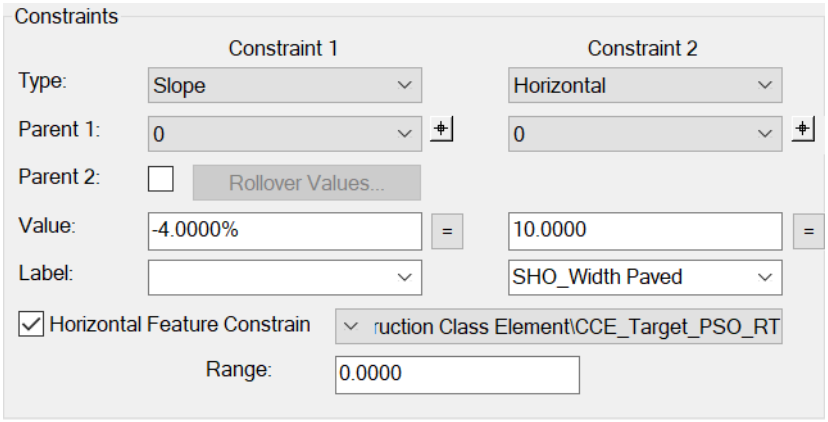
This component follows the standard for 2’ or 4’ paved shoulder where the paved shoulder slope is the same as pavement slope.



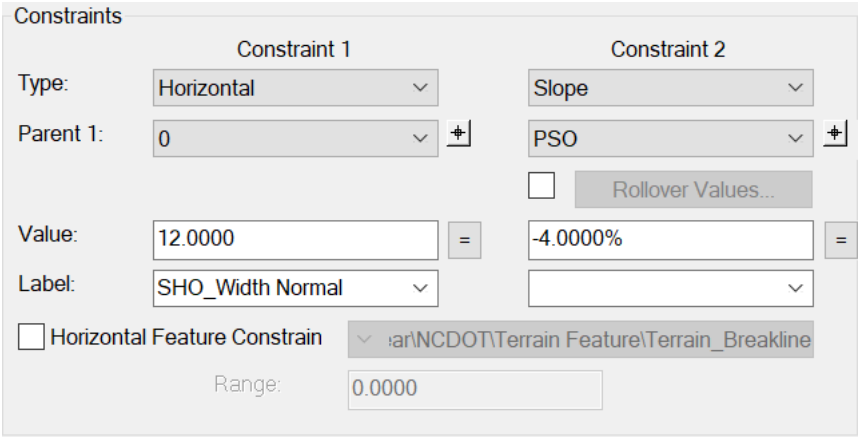
10’ or wider paved shoulders are created in a similar fashion. Editing the rollover slope at the **ETO** point is required.



25. Edit the **PSO** point and constrain it as shown below.

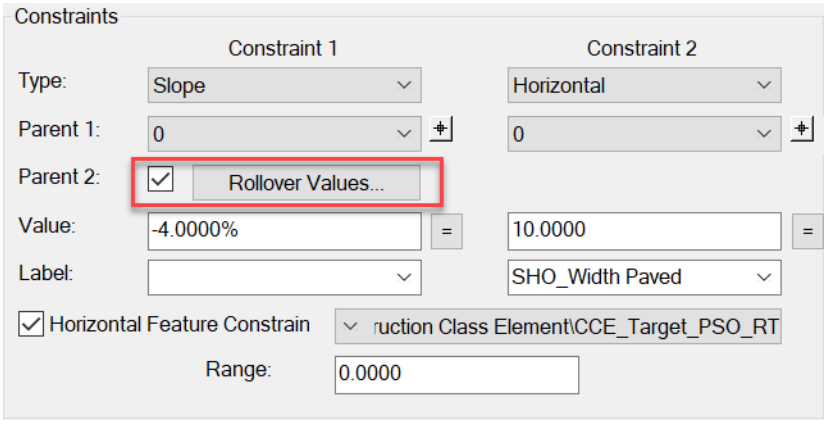


26. Edit the **GSO_N** point and constrain it as shown below.

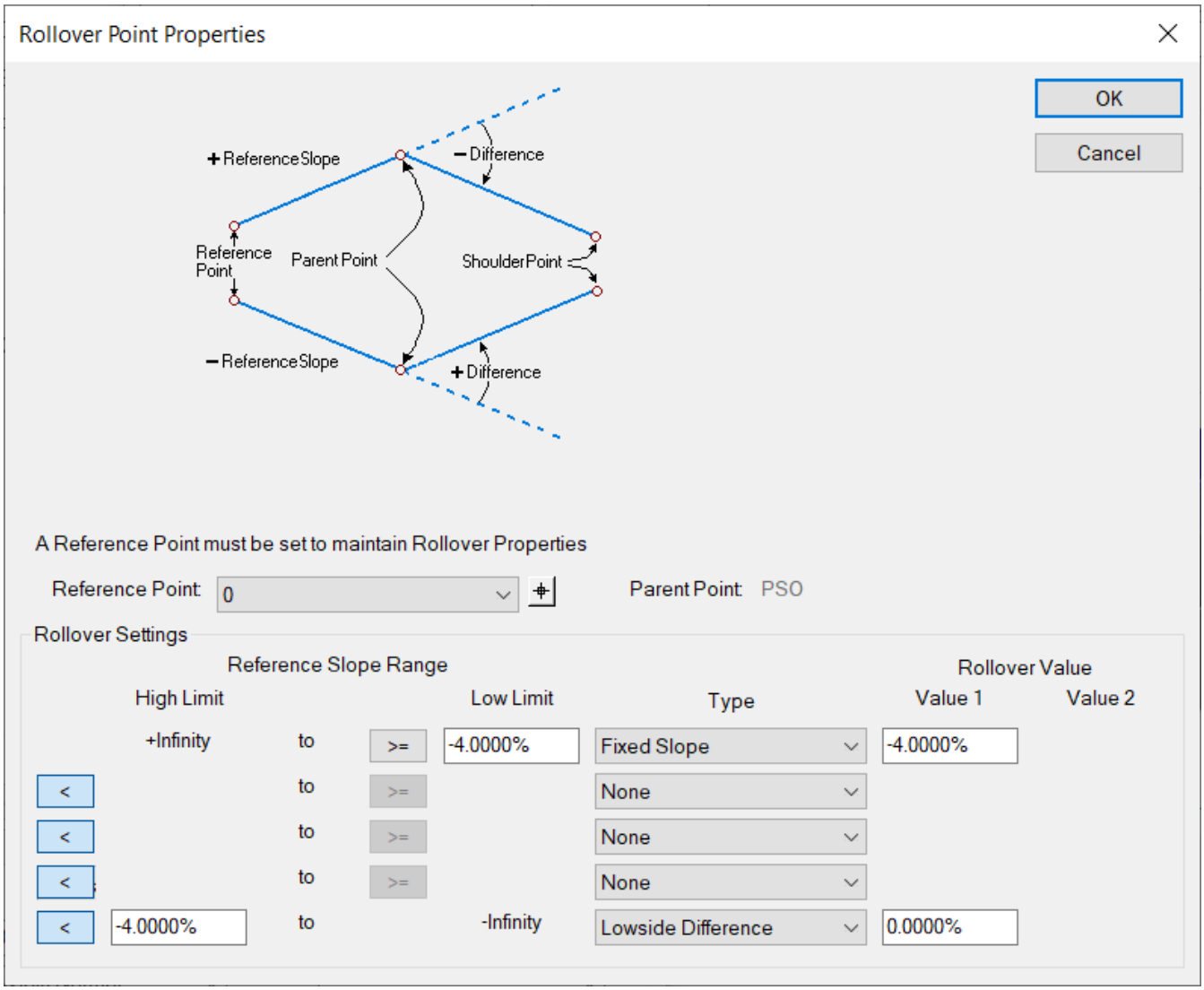


Note this is the default state of the component. When the pavement slope is -2%, the paved shoulder slope is -4%. To account for the **6%** rollover lock at the **ETO** point, edit the **PSO** point and set the **Rollover Values**.

27. Edit the **PSO** point. Click on the check box and select the **Rollover Values** button.



Rollover Point Properties



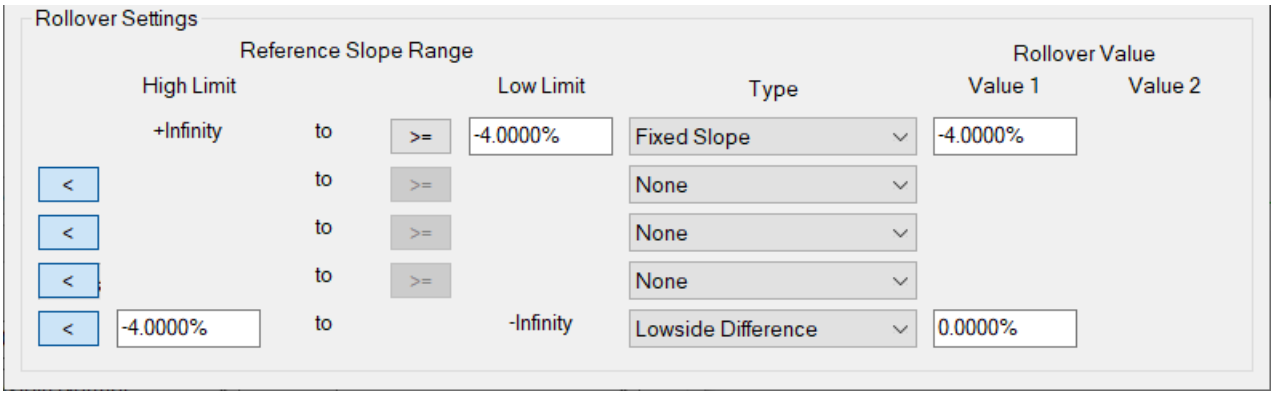
Diagram

Shoulder Point – Point being programmed with the **Rollover Slope Value**.

Parent Point – Point adjacent to the **Shoulder Point** forming the resulting +/- **Rollover Slope Value**.

Reference Point – Point adjacent to **Parent Point** forming the +/- **Refence Slope** to evaluate the resulting **Rollover Slope Value**.

Difference - +/- difference between the **Reference Slope** and the resulting **Rollover Slope Value**.



Rollover Settings

Reference Range

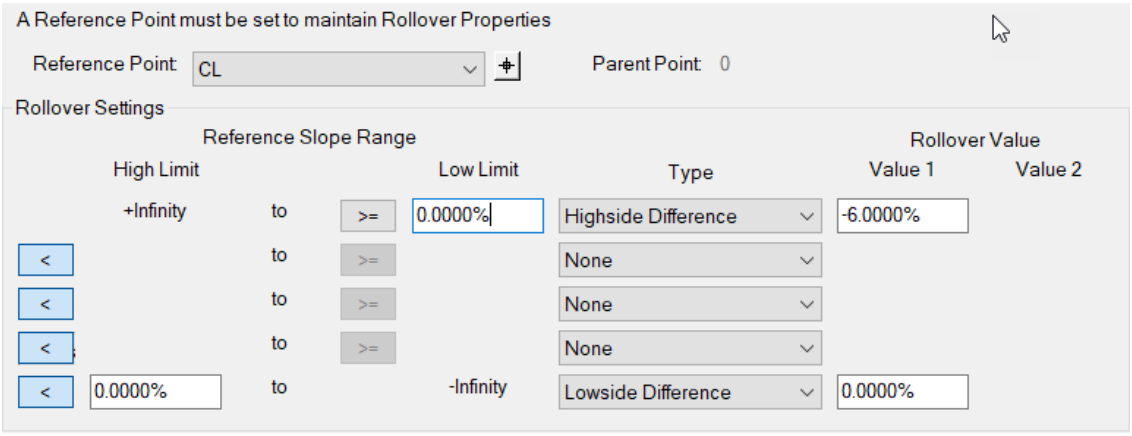
High Limit – Low Limit – Define the Slope Range formed between the Reference and Parent Points. It is essentially the Reference Slope in the diagram. Note the >, < and = sign in front of the value.

Type

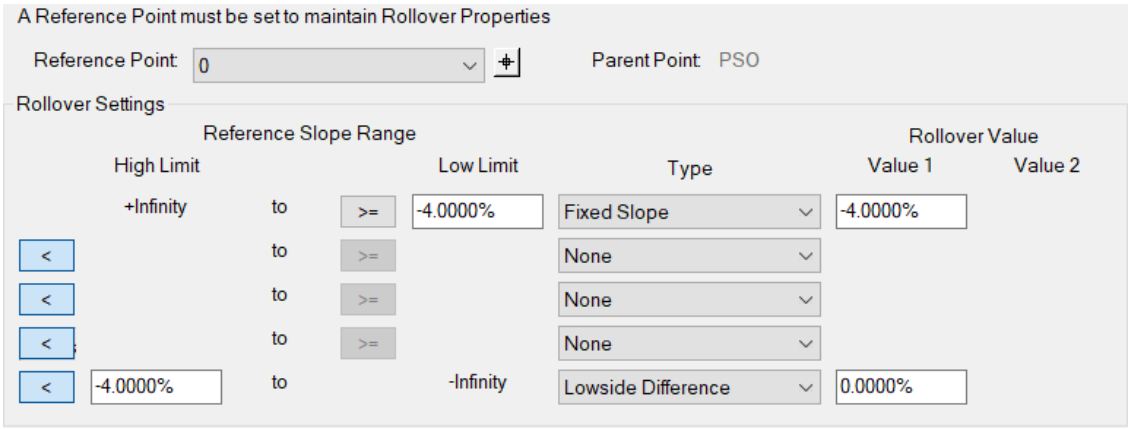
- **None** –no **Rollover Slope Value** applied (use slope drawn in the template).
- **Relative Difference** – +/- difference in slope formed between the **Reference Slope** and the resulting **Rollover Slope Value** (independent of highside or lowside).
- **Variable Slope** – variable slops from the **High Limit** (Value 1) to the **Low Limit** (Value 2).
- **Fixed Slope** – fixed slope for the **High Limit** and the **Low Limit**.
- **Highside Difference** – +/- difference between the **Reference Slope** and the resulting **Rollover Slope Value**. **Highside** is usually determined by a positive (+) **Reference Slope** greater than or equal to **0%**.
- **Lowside Difference** - +/- difference between the **Reference Slope** and the resulting **Rollover Slope Value**. **Lowside** is usually determined by a negative (-) **Reference Slope** less than **0%**.

Value – desired slope(s) or difference (Rollover Lock) **Value** to determine the resulting **Rollover Slope**.

28. Set the following values for the **PSO** point in the **Rollover Point Properties**.

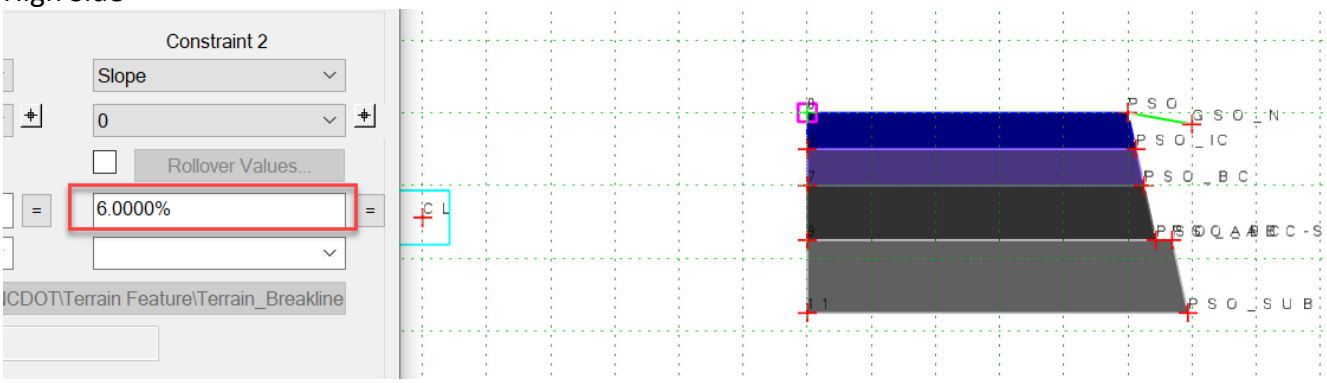


29. To account for the 4% high side and matching the paved shoulder slopes of the low side on the turf shoulder, edit the **GSO_N point** and set the following values in the **Rollover Point Properties**.

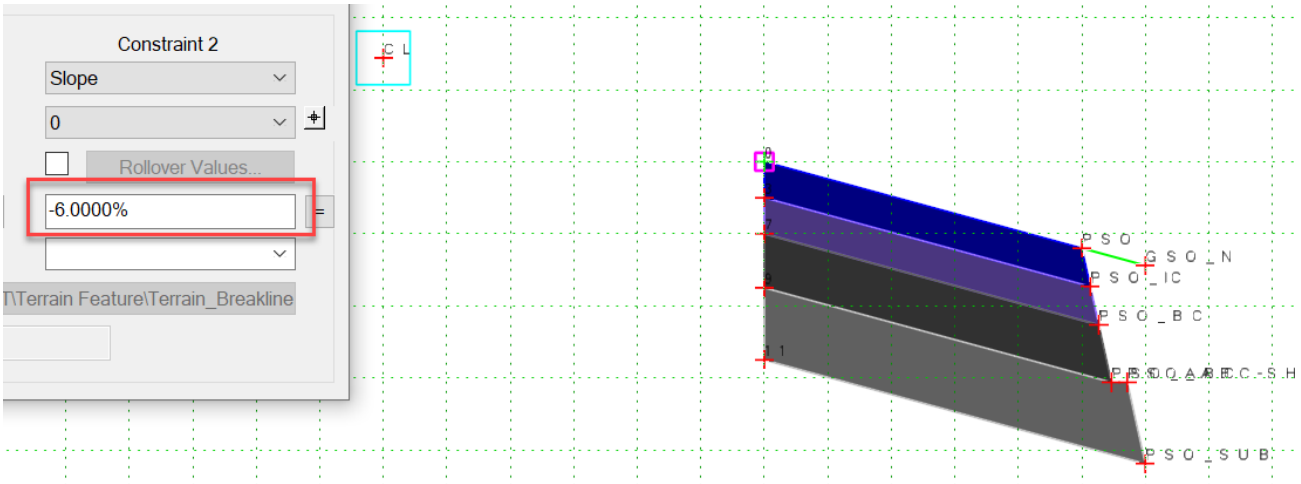


30. Test to see if the component is behaving as expected by editing the **CL** point and entering the various pavement slope values (superelevation).

High Side



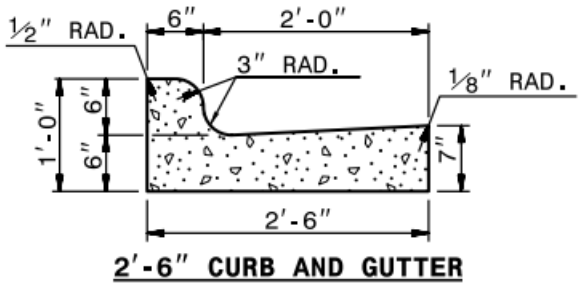
Low Side



Exercise C5: Curb and Gutter Component

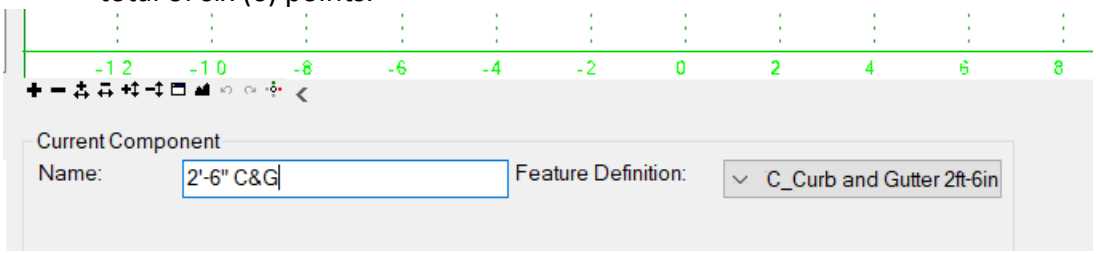
In this exercise we will demonstrate the proper technique to create a curb and gutter component per drawing specifications. Then slope the gutter per standard. The use of **Null Points** as a reference to determine the slope of the curb and gutter components, gutter **Rollover** slopes and **Equation** for gutter pan slope are taught in this exercise.

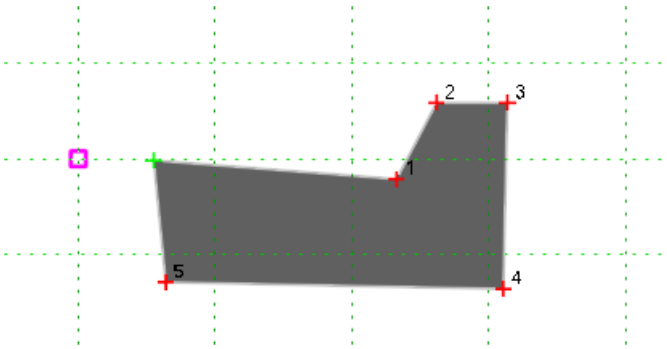
In the Template Library **Open** the **02 Components\05 Curb and Gutter** folder and create a new template named **2'-6" C&G**.



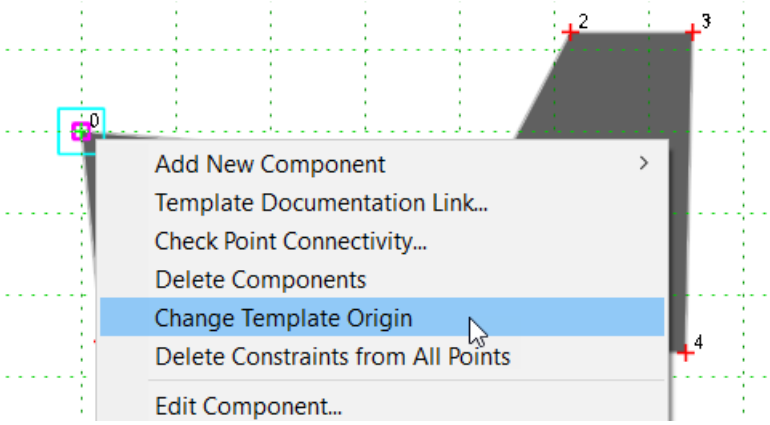
2'-6" CURB AND GUTTER

1. Create the shape of the curb and gutter as on the right side of the road. Use **Add New Component >>> Constrained** method and work clockwise. The slope, horizontal, and vertical distances between points do not matter at this time. We will constrain them later. Specify in the lower left part of the screen the component Name as **2'-6" C&G** and the Feature Definition of "**Mesh\Roadway\Concrete\TC_Curb and Gutter 2ft-5in**". There should be a total of six (6) points.

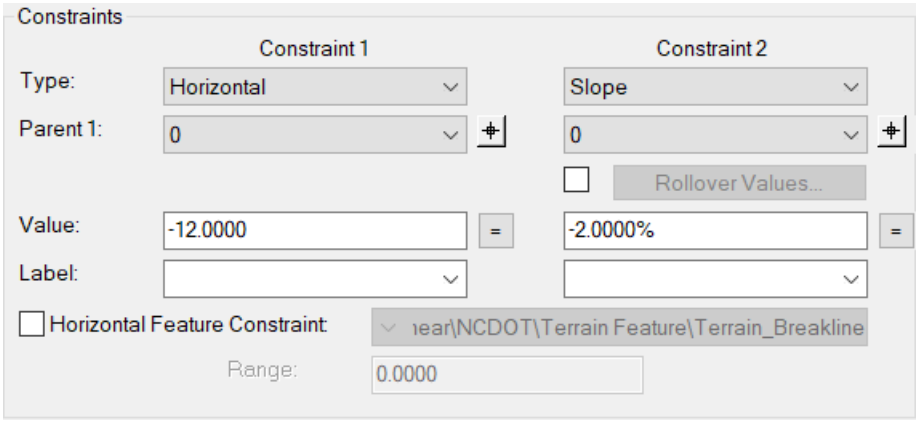




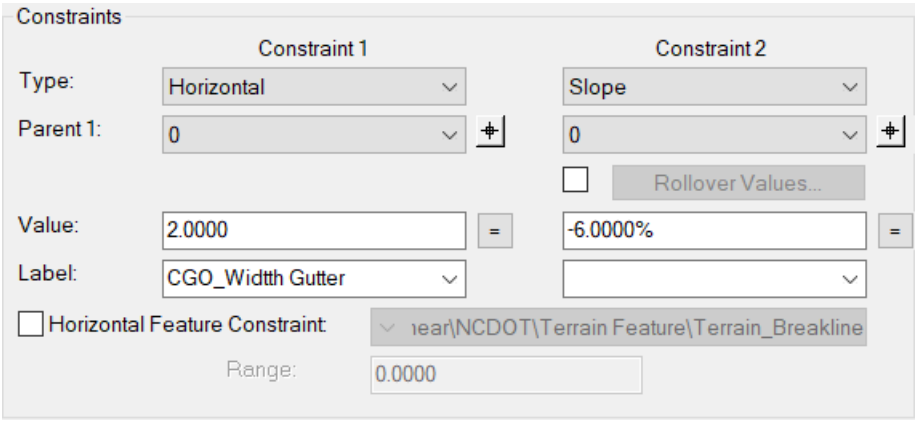
2. Rename the green blank point **0** (zero) and make it the template origin.



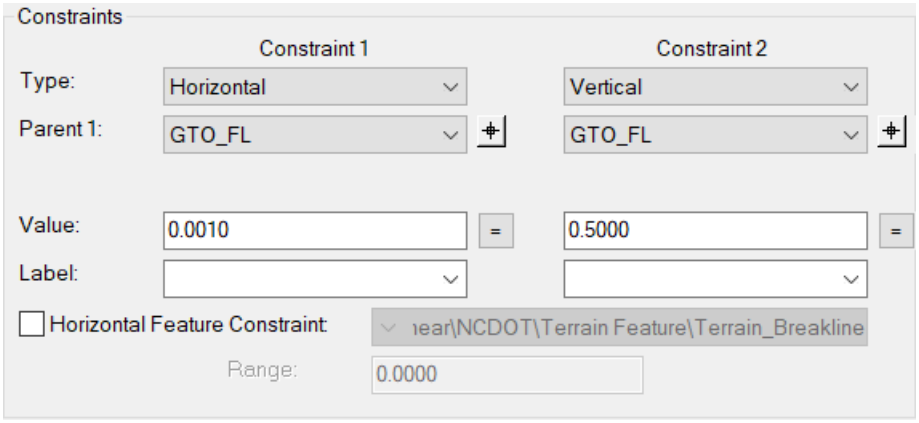
3. Create a new **Null Point** name **CL** to the left of point **0** and constrain it as shown below.



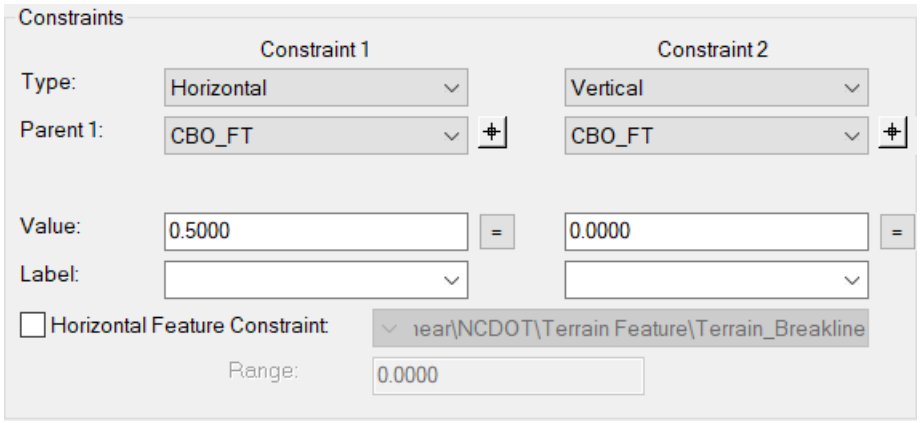
4. Edit point **1** and rename it **GTO_FL**. Constrain it as shown below.



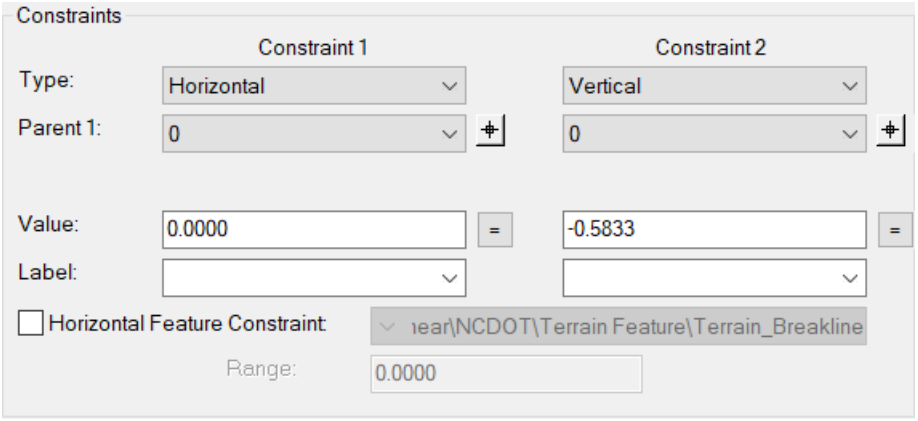
5. Edit point **2** and rename it **CBO_FT**. Constrain it as shown below.



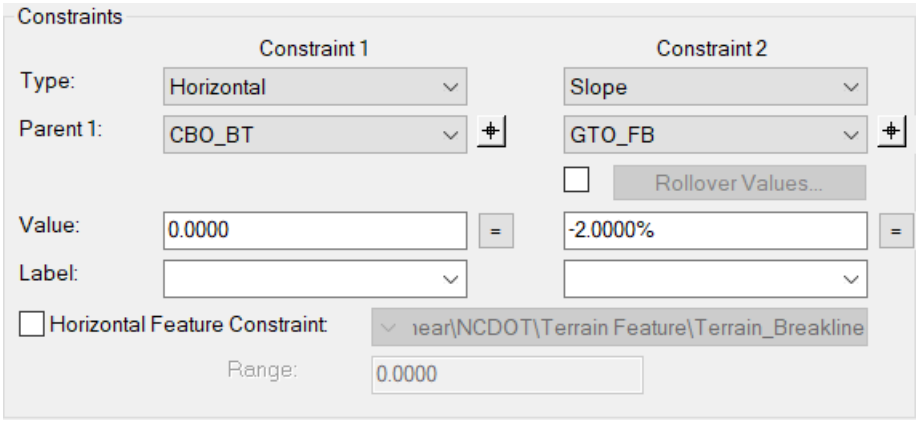
6. Edit point **3** and rename it **CBO_BT**. Constrain it as shown below.



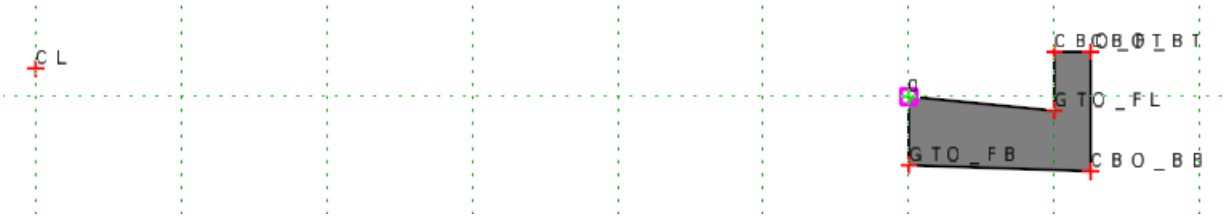
7. Edit point **5** and rename it **GTO_FB**. Constrain it as shown below. Note Point **5** will be edited before point **4** because it will cause a recursive error if point **4** is constrained to point **5**.



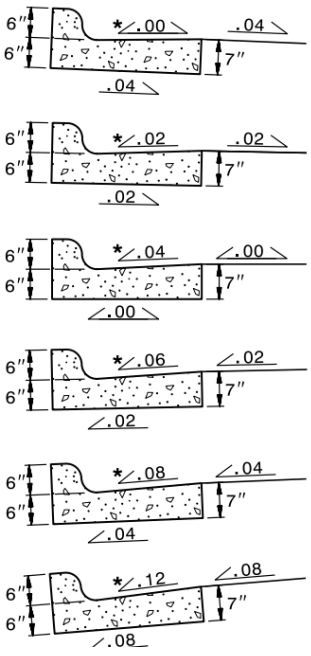
8. Edit point **4** and rename it **CBO_BB**. Constrain it as shown below.



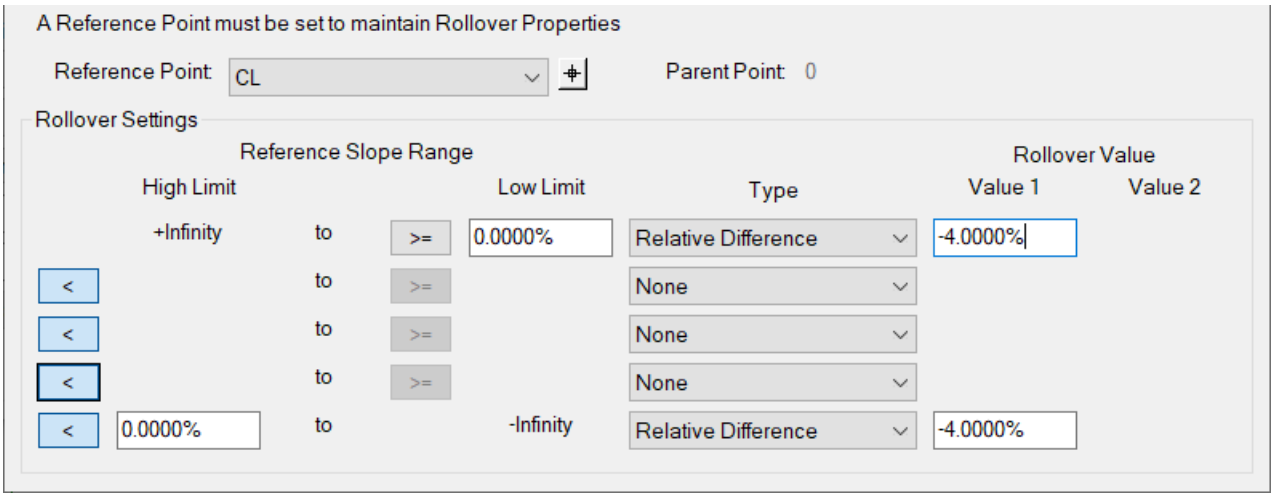
The component should look like the below be picture.



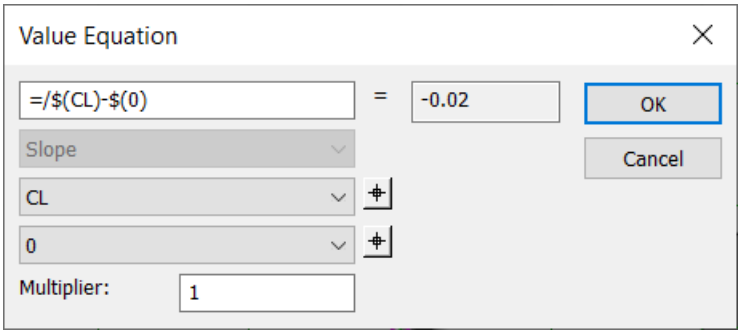
To follow the slope standards, two points will need to be edited, **GTO_FL** and **CBO_BB**.



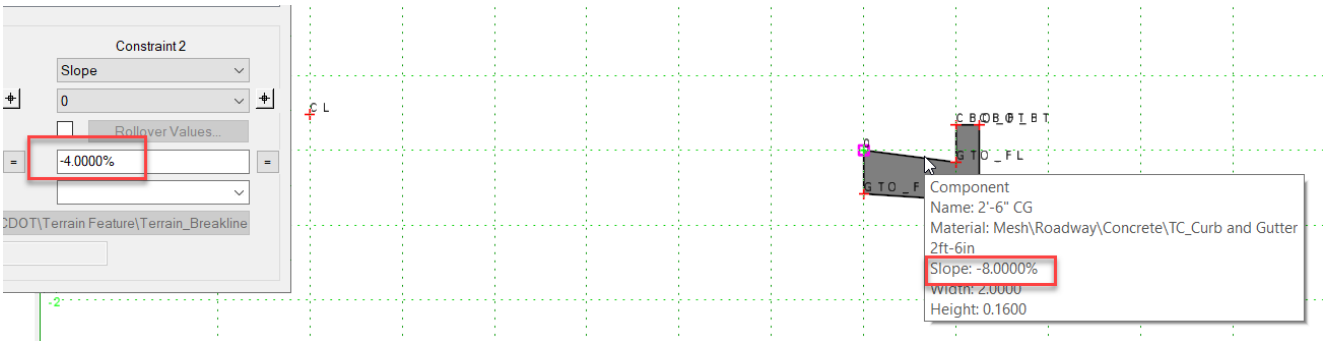
9. To account for the 4% rollover lock at the top of the gutter, edit the **GTO_FL** point. Enable the **Rollover Values** button and set the following rollover values.

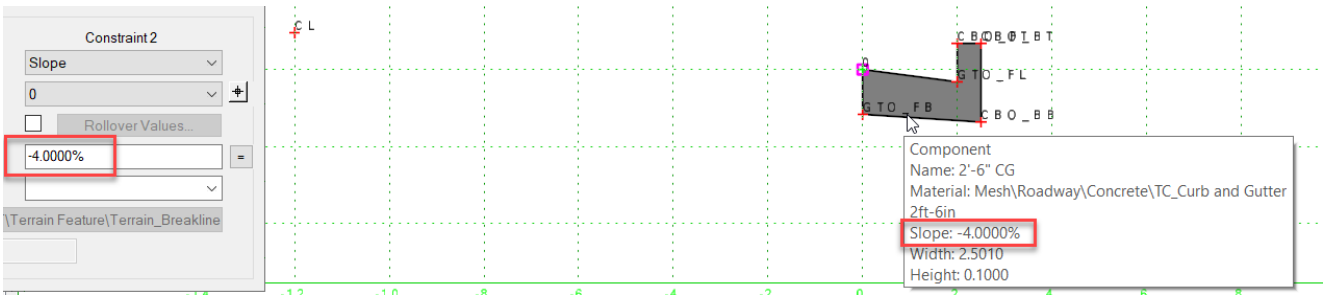


10. To match the pavement slope for the bottom of the gutter, an **Equation** can be written. Edit the **CBO_BB** point and enable the equation dialog box ("=") under the **Slope** constraint value. Set the following equation expression.



Test to see if the curb and gutter component is behaving as designed by editing the **CL** point and changing the various **Slope** values.



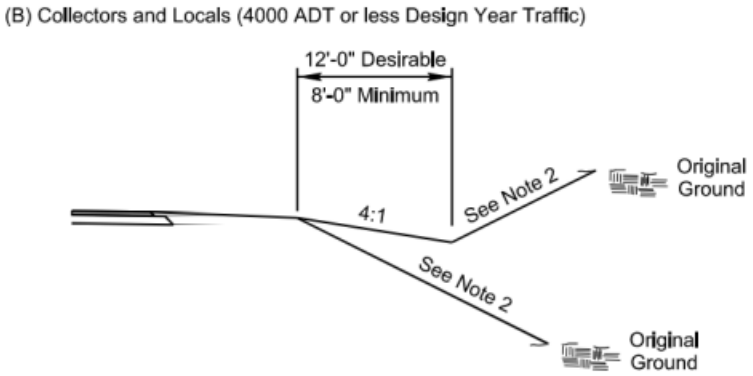


Exercise C6: End Conditions

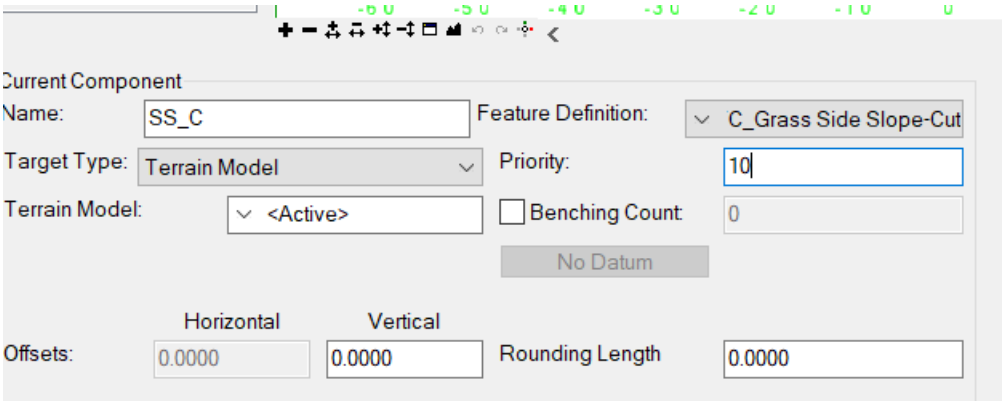
In this exercise we will teach how the creation of end conditions (side slopes) are created and what properties control their behavior. The Roadway Design Manual is used as a guide. **End Condition Properties, End Condition Priority**, **Equations, Parent Component** and **Variable Slopes** are taught in this exercise.

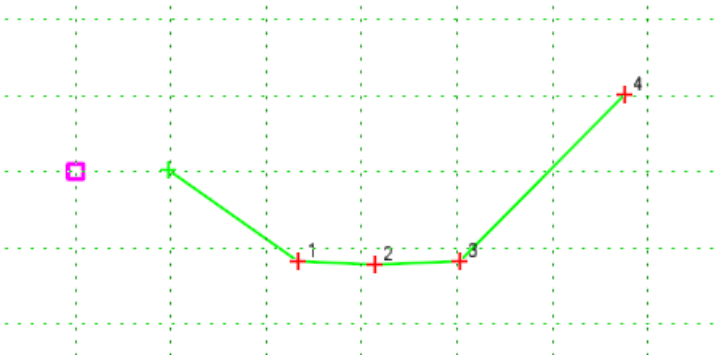
There are three (3) types of end conditions used at the NCDOT, Local Design Standard Slopes (**LDSS**), Freeway/Arterial Design Standard Slopes (**FD/ADSS**) and Catchslopes (**CSLP**) for C&G sections.

In the Template Library **Open** the **02 Components\06 End Conditions** folder and create a new template called **LDSS**.

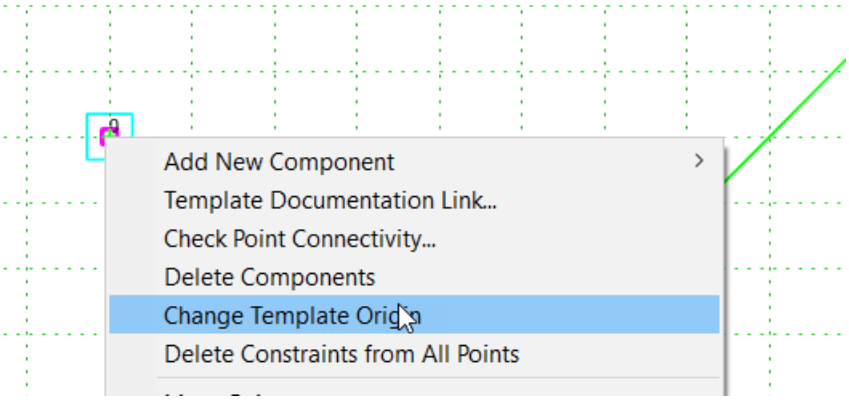


31. Starting with the local design ditch, **Add New Component >>> End Condition** and create the shape of the ditch (5 points). Enter the following values in the component end condition properties prior to placement. The Feature Definition is "**Mesh\Roadway\Grading\TC_Grass Side Slope-Cut**".

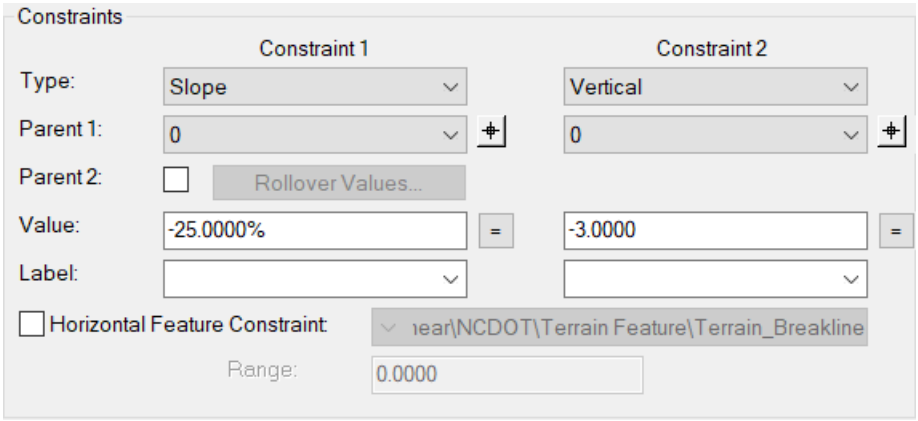




32. Edit the green blank point and rename it **0** (zero). Make it the template origin.

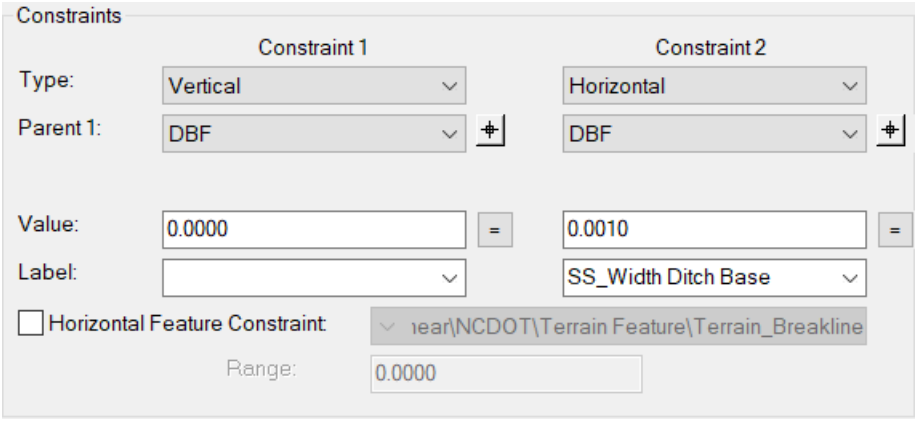


33. Edit point **1** and rename it DBF. Constrain it as shown below.



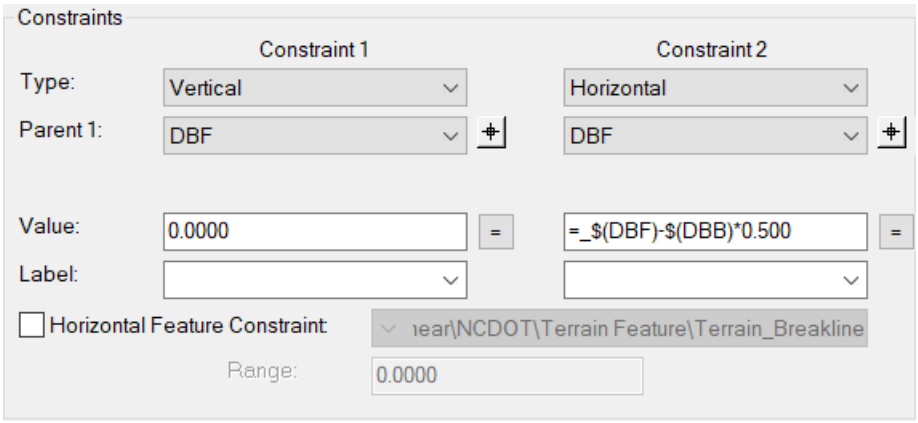
Note that the second constraint can be set to **Horizontal** for the ditch width, but when a special ditch grade is applied, the constraint must be set to **Vertical**.

34. Edit point **3** and rename it **DBB**. Constrain it as shown below.



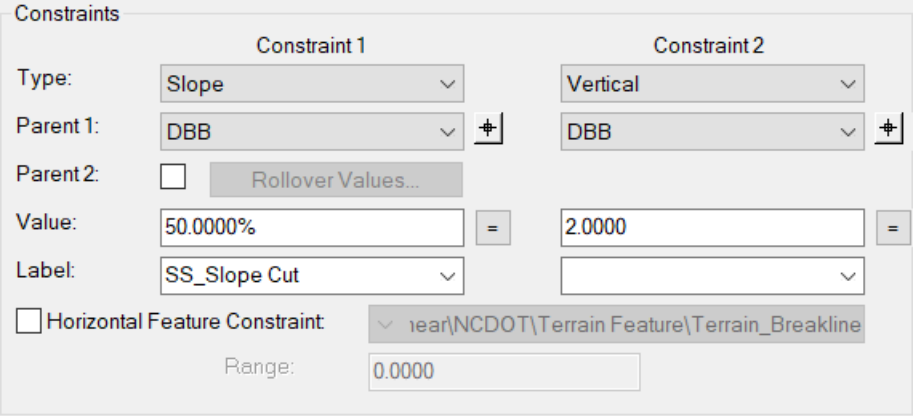
Note point **3** is edited before point **2** because it will cause a recursive error if we edit point 2 prior of editing point **3**.

35. Edit point **2** and rename it **DBM**. Constrain it as shown below.

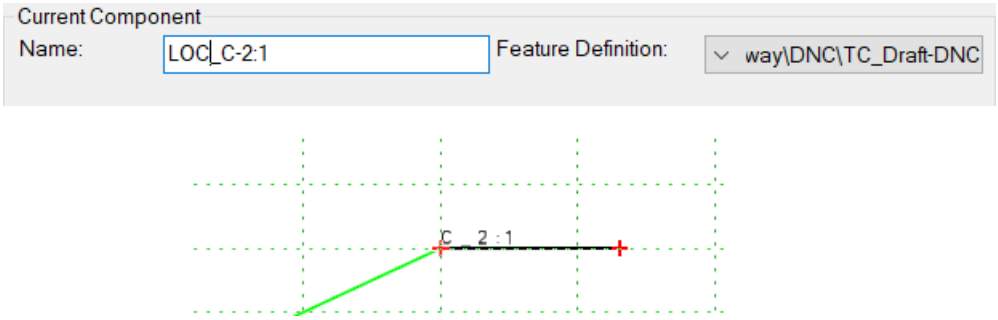


Note the equation is half (**0.5**) the distance between the **DBF** and **DBB** points.

36. Edit point **4** and rename it **C_2:1**. Constrain it as shown below.



37. Create the LOC for this end condition, **Add New Component >>> Constrained** and starting at the **C_2:1** point create a line component to the right. Enter the following values in the component end condition properties prior to placement. The Feature Definition is "**Mesh\Roadway\DNC\TC_Draft-DNC**".



The limits of construction **(LOC)** is a dashed slope stake line that is printed on the plans. The cut and fill lines are solid and they are not plotted out. They are displayed in the color green for cut and red for fill to show engineers where to place the Cs and Fs on the plans. Regular cut and fill lines are not used for printing because in the transition area between cut and fill or fill and cut, there exists a "gap" between the two (2) lines. The LOC dashed lines are always connected regardless of cut or fill.

38. Make SS_C the Parent Component

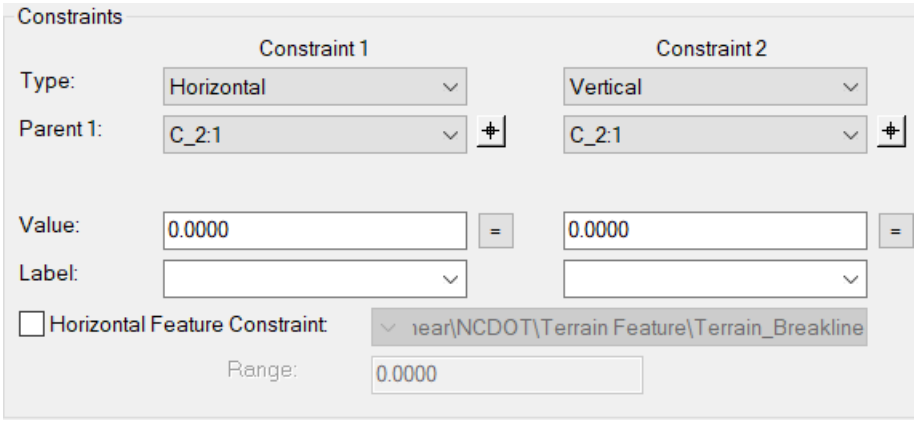
Also note points can be inserted or added to a component. In this case, the inserted LOC point will cause the end conditions to not work properly. That is the reason a regular component has to be created connected to the end condition.

If the point is to become a member of the component (not a null point) there are two options, **Insert Point** and **Add Point**.

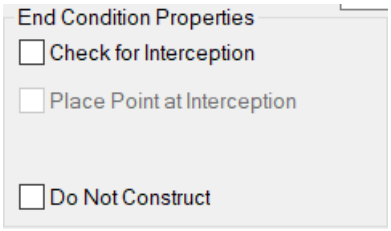
Insert vs. Add Point

Insert Point is a new point inserted between two points. Add point is adding a new point at the end of after the last point of the component.

39. Edit the blank point and rename it **LOC_C-2:1**. Constrain it as shown below.



40. Check each point **End Condition Properties** for correct settings. Starting with the point **0** and all other points except for **C_2:1**, they all should have the same **End Condition Properties** as shown below.



End Condition Properties

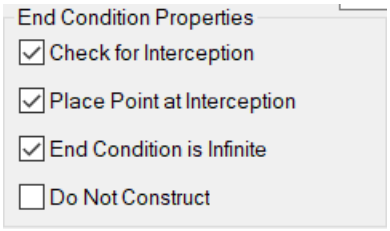
Check for Interception – If the existing ground (active terrain model) intersects the end condition component, then display it (turn it on).

Place Point at Interception – If Check for Interception is true, then place the point at the intersection.

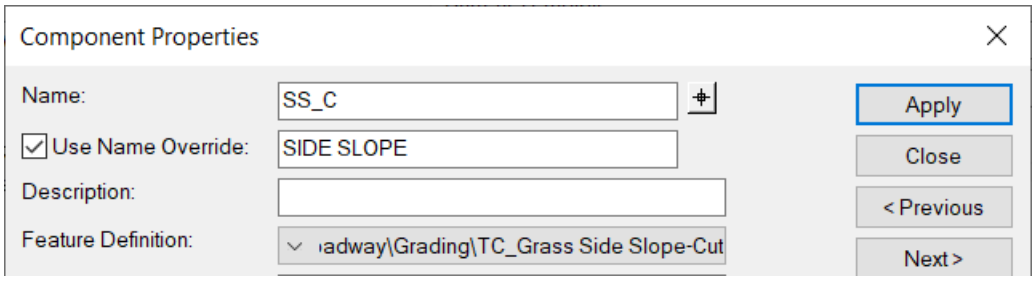
End Condition is Infinite (last point only) – Determine if the end condition can intersect the existing ground at an infinite distance or limited to as drawn (constrained).

Do Not Construct – For Intermediate points, connect the point to the before and after points or DNC and connect the end condition from the before point to the after point, skipping the intermediate. This option is used to make the slope variable with a fixed width.

41. Check point **C_2:1** for the following **End Condition Properties**.

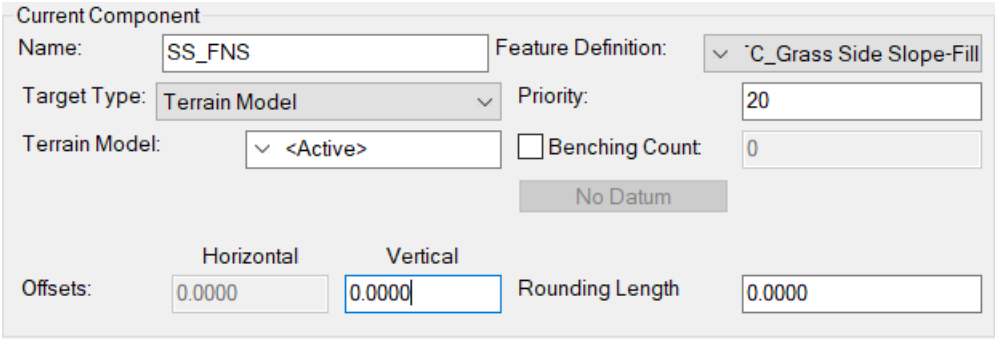


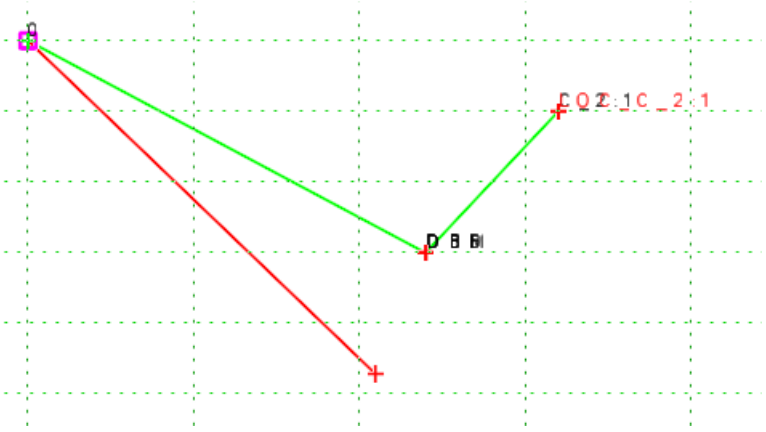
42. The last step for the ditch component is to **Use Name Override** and key-in **SIDE SLOPE**. The fill slope will have the same component name override to ensure the surface does not contain gaps transitioning from cut to fill or fill to cut.



The cut ditch component is complete. The fill slope can now be constructed.

43. **Add New Component >>> End Condition** starting from point **0** (zero) a draw a shape of the fill slope below the ditch. Enter the following values in the component end condition properties prior to placement. The Feature Definition is "**Mesh\Roadway\Grading\TC_Grass Side Slope-Fill**".

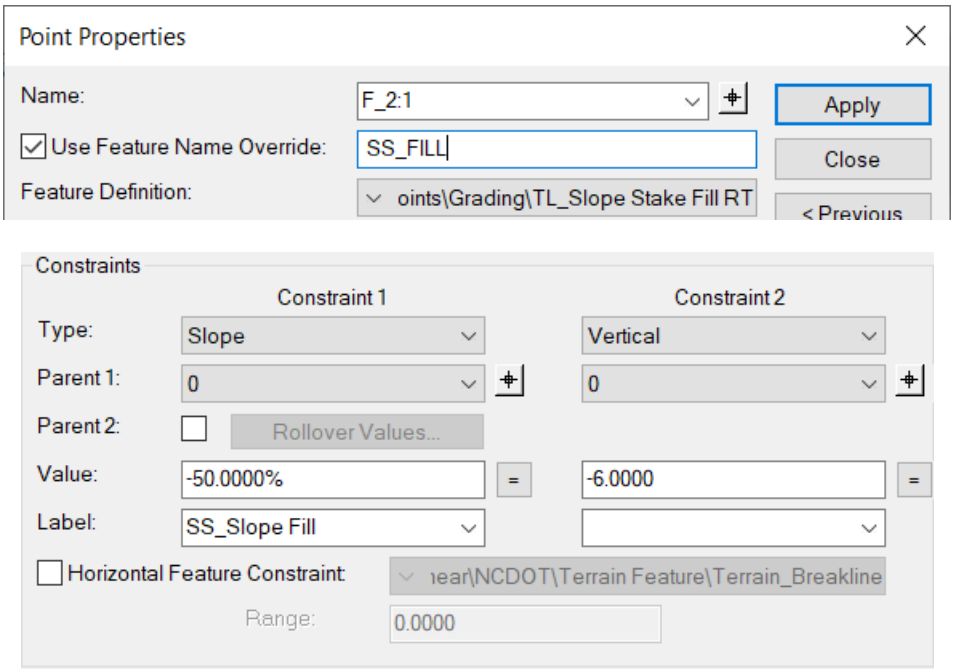




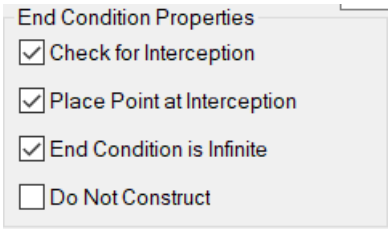
End Condition Priority

Note the fill slope **Priority** is set to **20**. The priority for the cut slope component was set to **10**. This means the cut component will be evaluated first before the fill component. Users may modify the end condition Priority in the template to evaluate fill before cut. End condition branches from the same origin point should not have the same Priority number.

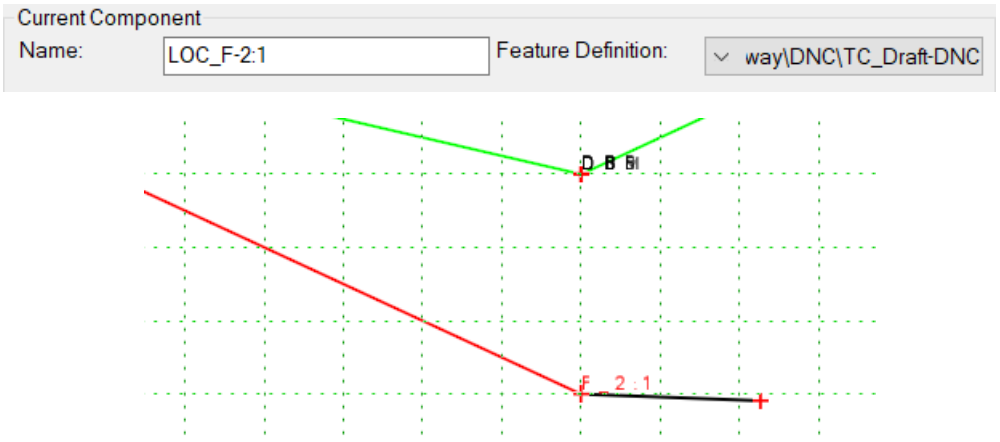
44. Edit point **1** and rename it **F_2:1**. **Use Feature Name Override** and key-in **SS_FILL**. Constrain it as shown below.



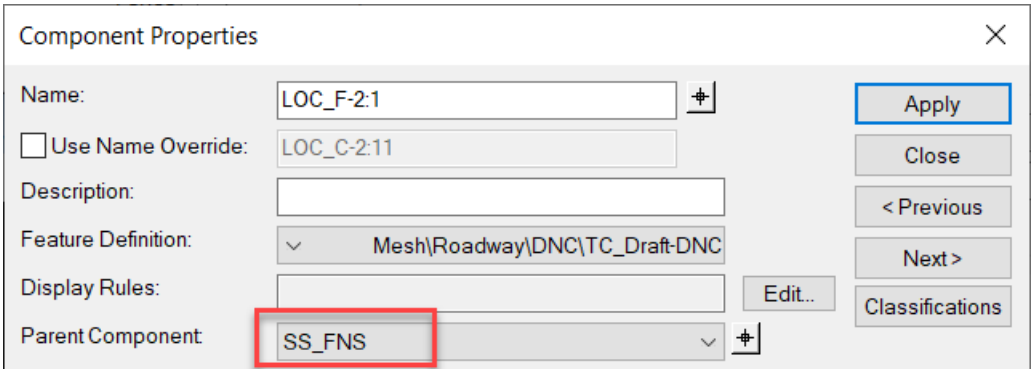
45. Point **F_2:1** should have the following End Condition Properties.



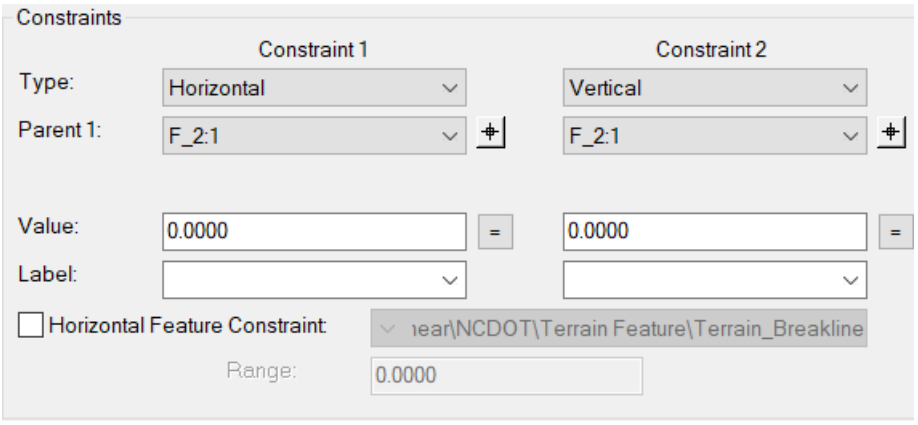
46. Create the LOC for this end condition, **Add New Component >>> Constrained** and starting at the **F_2:1** point create a line component to the right. Enter the following values in the component end condition properties prior to placement. The Feature Definition is "**Mesh\Roadway\DNC\TC_Draft-DNC**".



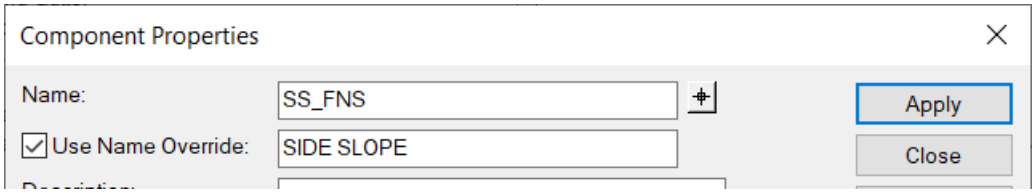
47. Make **SS_FNS** the **Parent Component**.



48. Edit the blank point and rename it **LOC_F_2:1**. Constrain it as shown below.



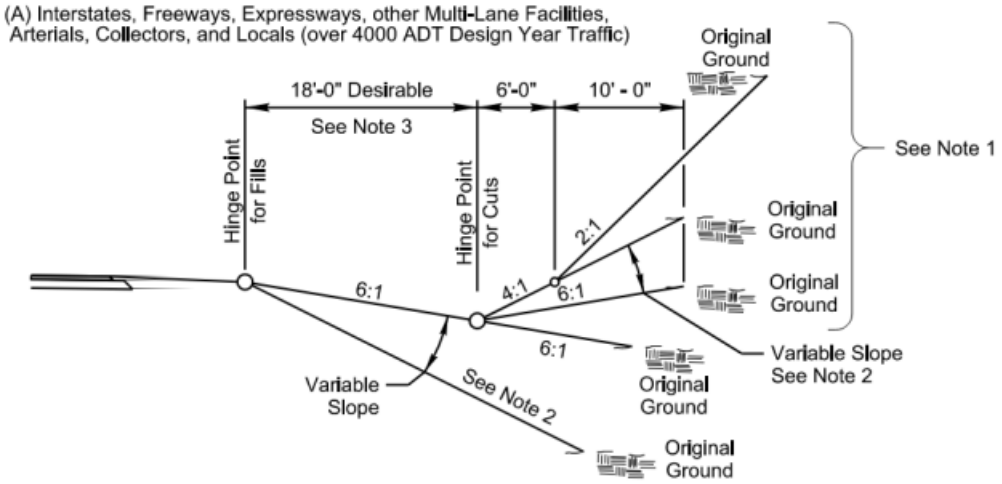
49. last step for the ditch component is to **Use Name Override** and key-in **SIDE SLOPE**.



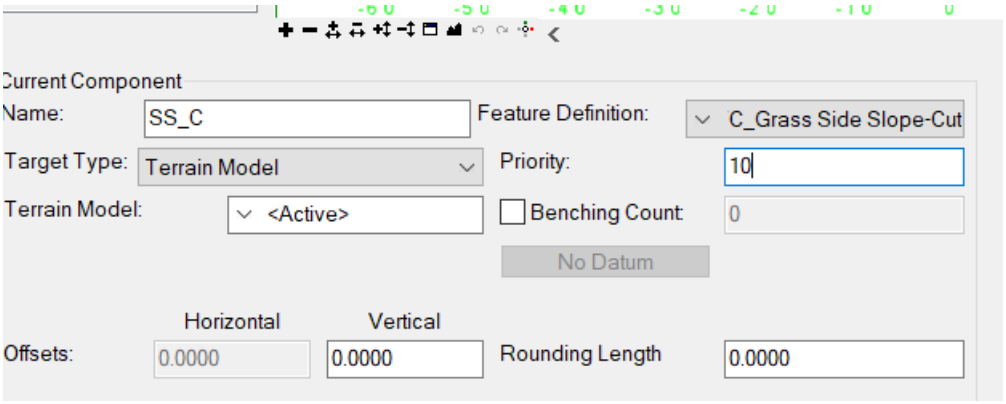
Test (bottom right button) the end condition.

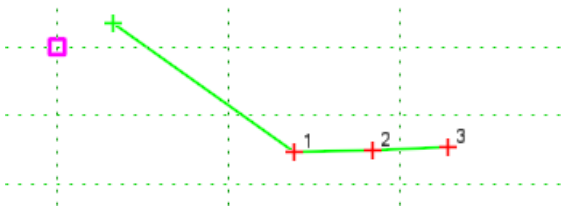
Create the **FD/ADSS** end conditions.

In the Template Library **Open** the **02 Components\05 End Conditions** folder and create a new template called **ADSS**.

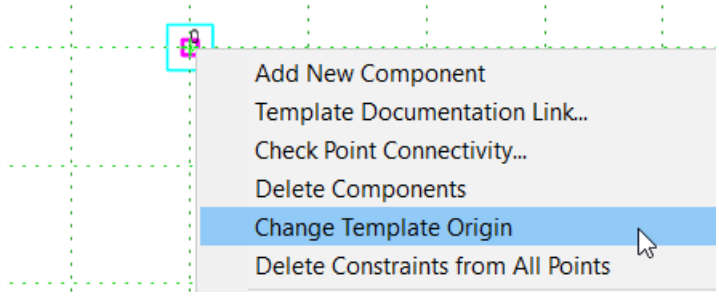


20. **Add New Component >>> End Condition** and create the shape of the ditch (front of ditch slope and the three base points, 4 points total). Enter the following values in the component end condition properties prior to placement. The Feature Definition is "**Mesh\Roadway\Grading\TC_Grass Side Slope-Cut**".

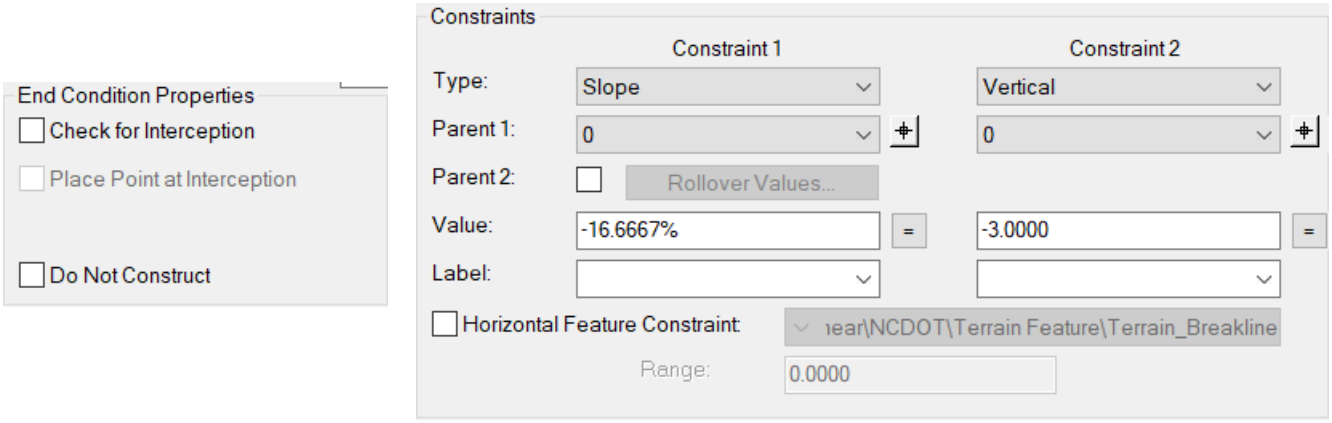




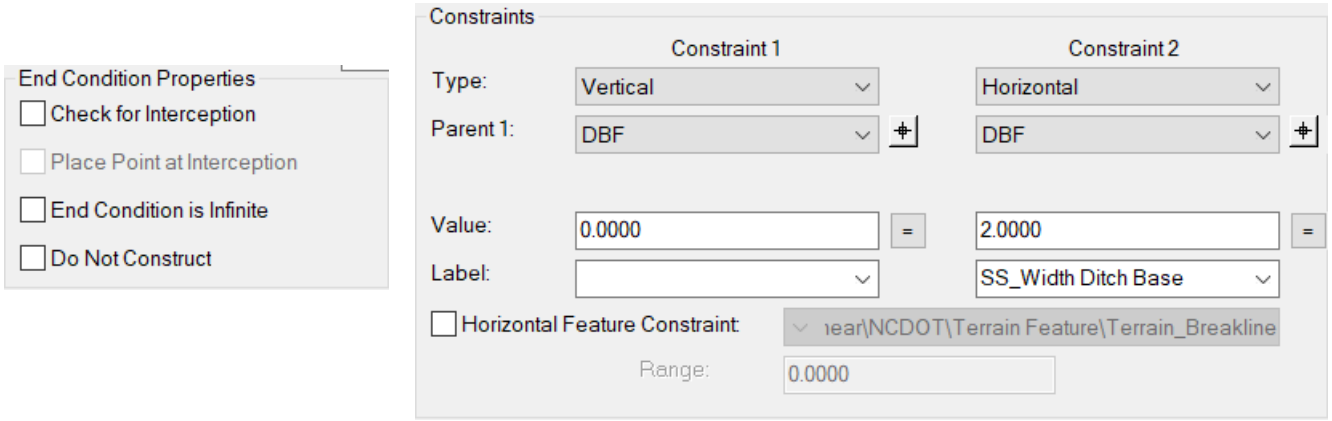
21. Edit the green blank point **0** (zero) and make it the template origin.



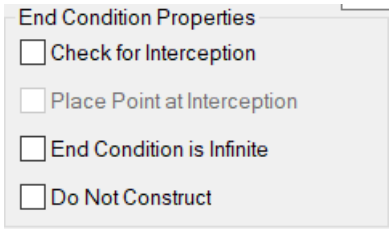
22. Edit point **1** and rename it **DBF**. Constrain as shown below. It should have the following **End Condition Properties**.

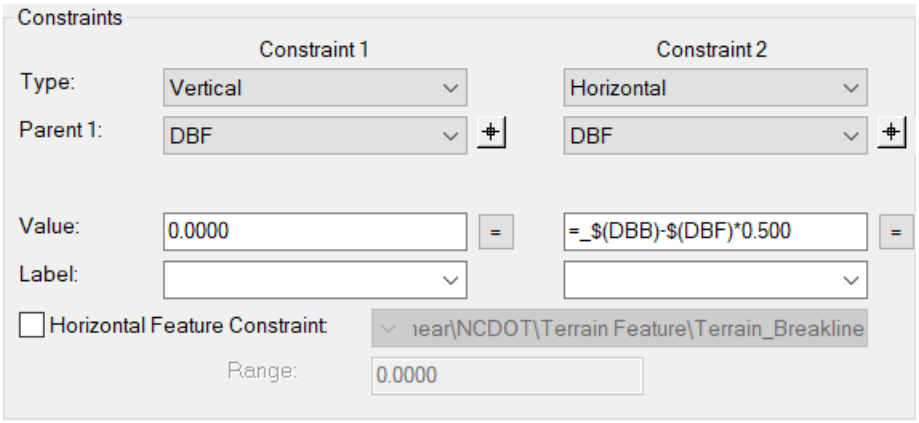


23. Edit point **3** and rename it **DBB**. Constrain as shown below. It should have the following **End Condition Properties**.

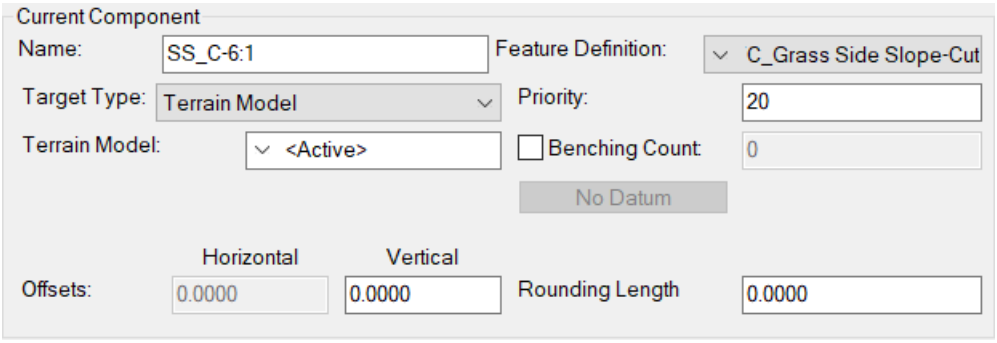


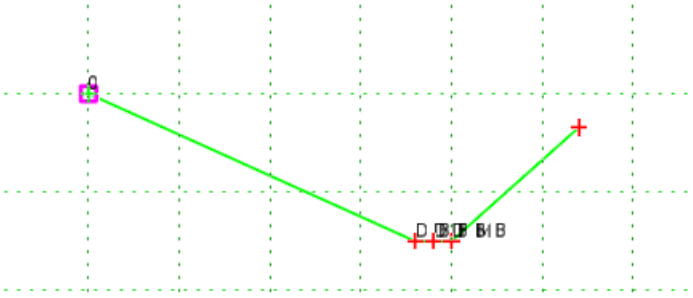
24. Edit point **2** and rename it **DBM**. Constrain as shown below. It should have the following **End Condition Properties**. Note the equation is written to be half the horizontal distance between point **DBF** and **DBB**.



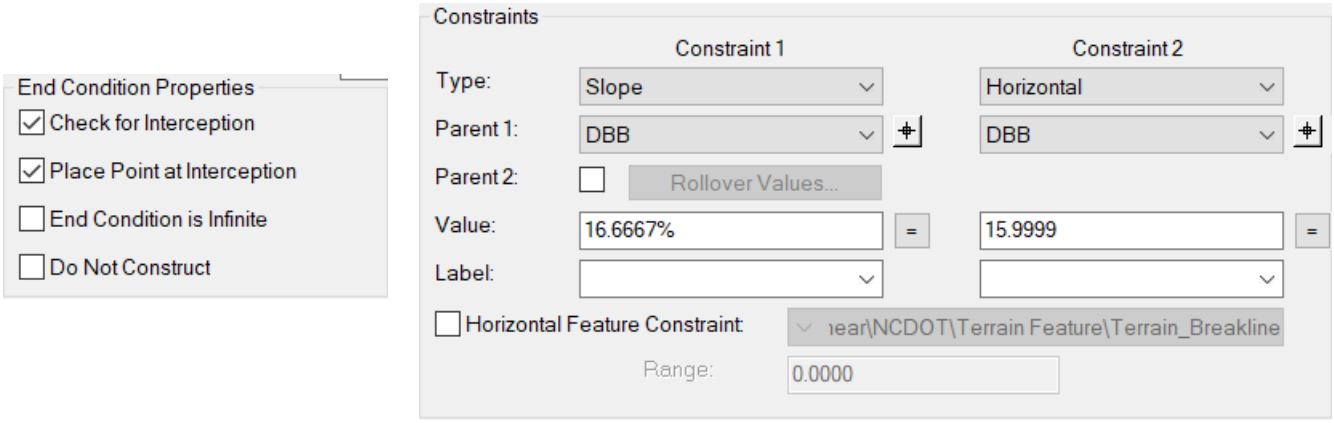


25. **Add New Component >>> End Condition** and starting at the **DBB** point create the 6:1 back slope of the ditch. Enter the following values in the component end condition properties prior to placement. The Feature Definition is "**Mesh\Roadway\Grading\TC_Grass Side Slope-Cut**".





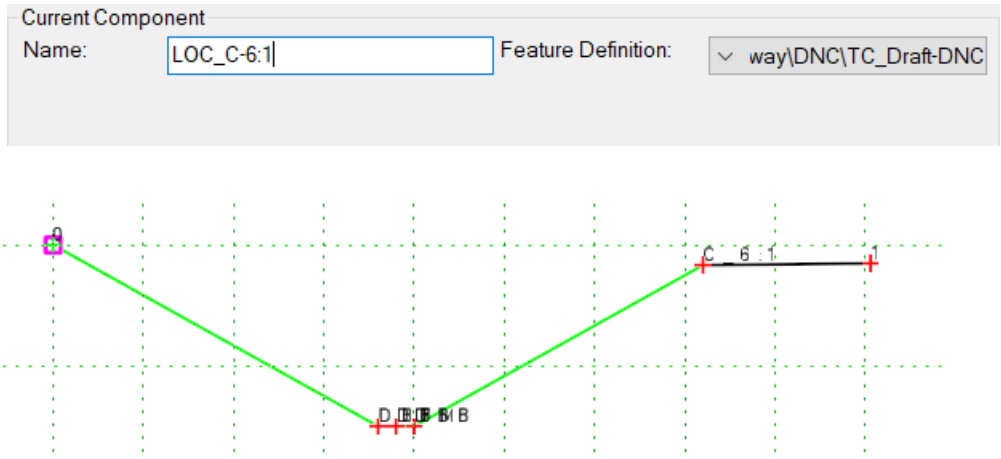
26. Edit the blank last point and rename it C_6:1. Con Constrain as shown below. It should have the following **End Condition Properties**.



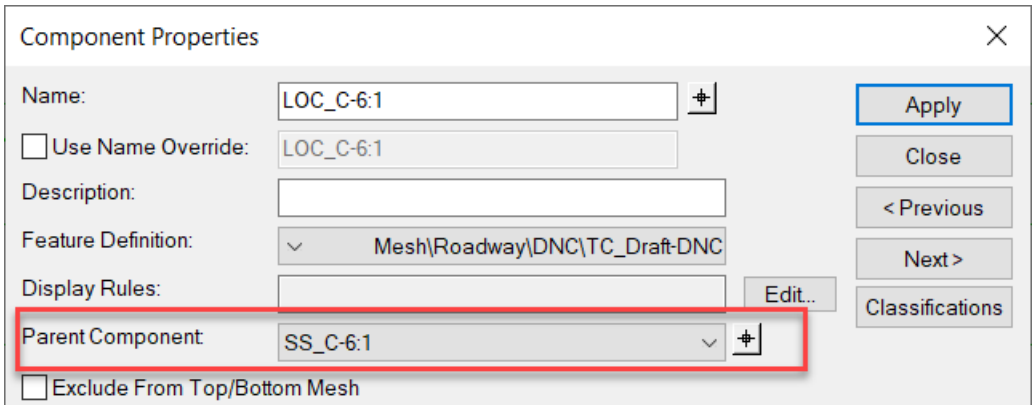
Auto-Conversion of Values

When entering a slope, horizonal or vertical value, the auto-conversion feature can be used. Enter **1:6** in the **Value** field and hit Enter on the keyboard. The value is then automatically converted to **16.6666%**. **7"** in the Horizontal or Vertical **Value** field is automatically converted to **0.5833**.

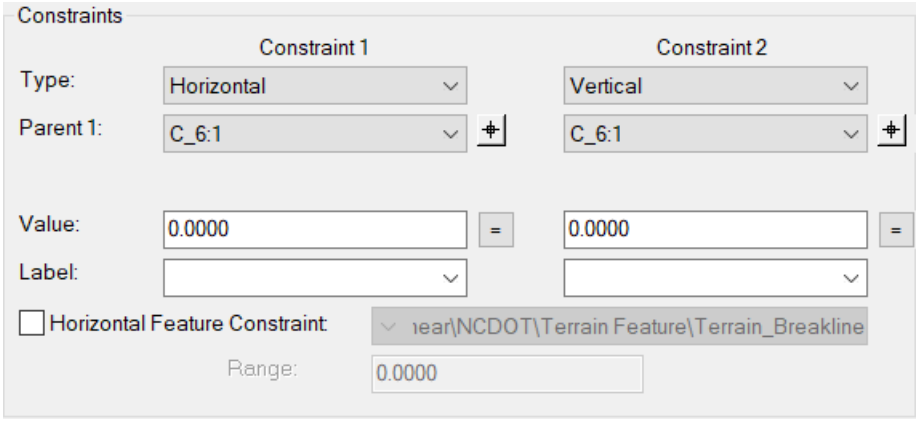
8. To create the LOC for this end condition, **Add New Component >>> Constrained** and starting at the **C_6:1** point create a line component to the right. Enter the following values in the component end condition properties prior to placement. The Feature Definition is "**Mesh\Roadway\DNC\TC_Draft-DNC**".



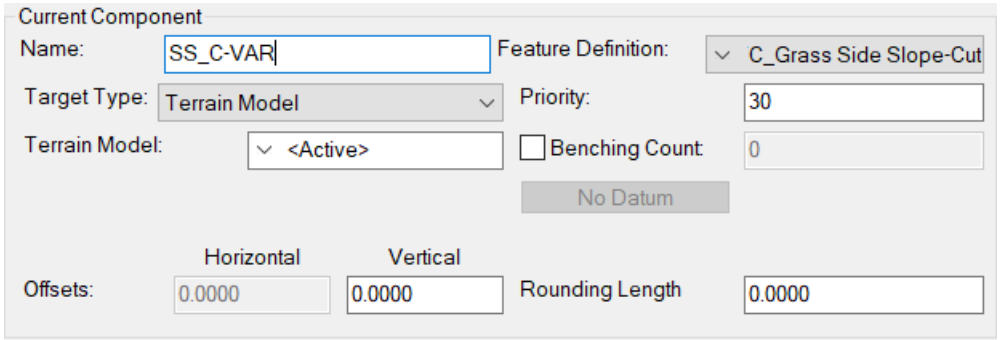
9. Make the 6:1 back slope the **Parent Component** of this LOC component line.



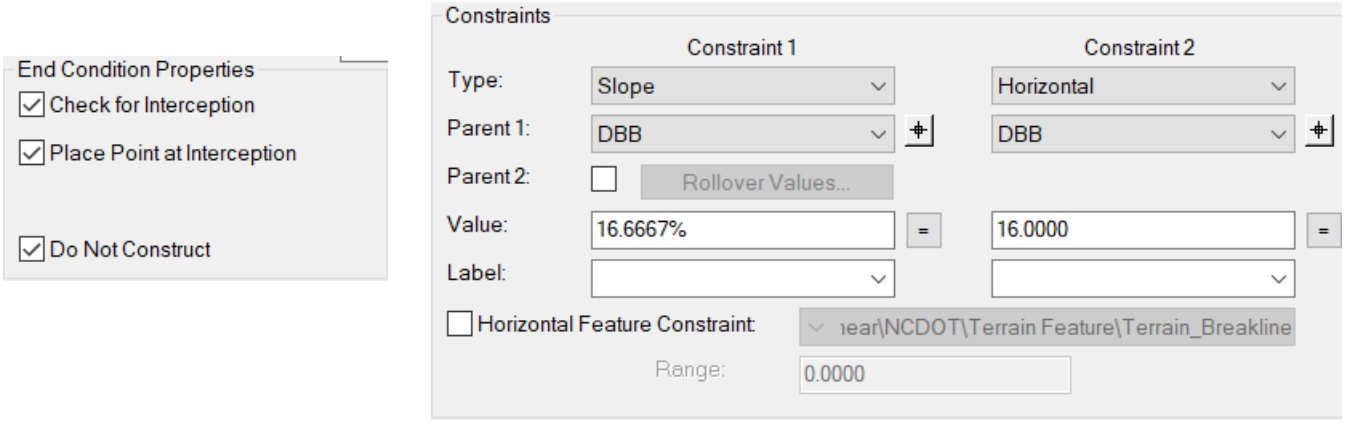
10. Edit the last point of this regular component and rename it **LOC_C_6:1.** Constrain as shown below.



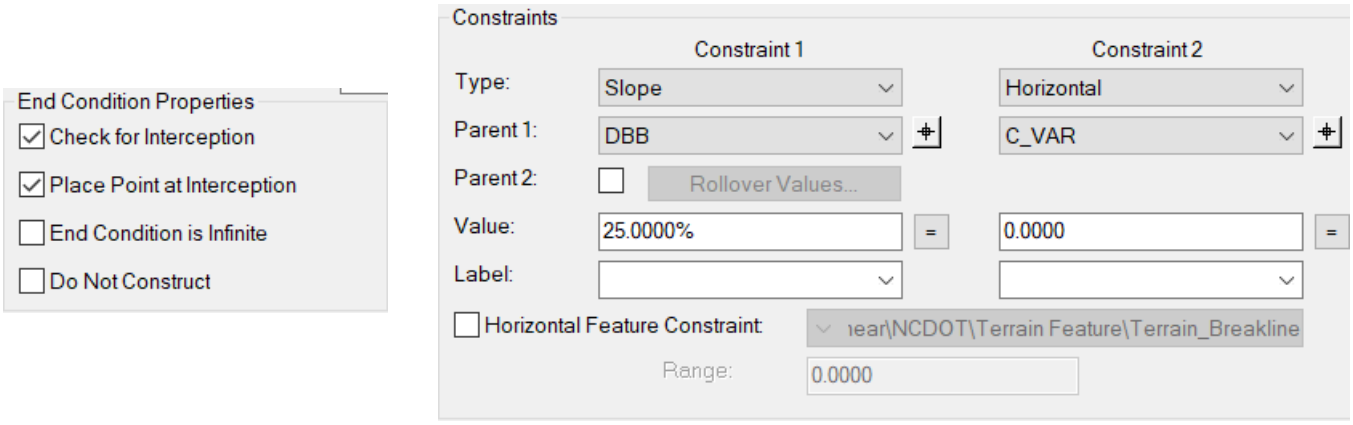
11. **Add New Component >>> End Condition** and starting at the **DBB** point create the variable **6:1** to **4:1** back slope of the ditch (3 points total). Enter the following values in the component end condition properties prior to placement. The Feature Definition is "**Mesh\Roadway\Grading\TC_Grass Side Slope-Cut**".



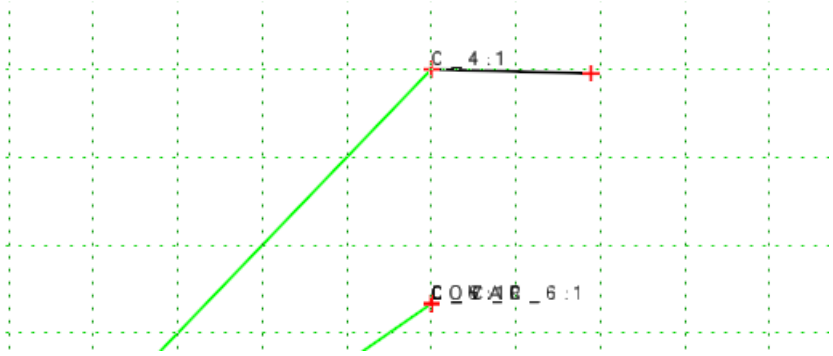
12. Edit the first blank point and rename it **C_VAR**. Constrain it as shown below. It should have the following **End Condition Properties**.



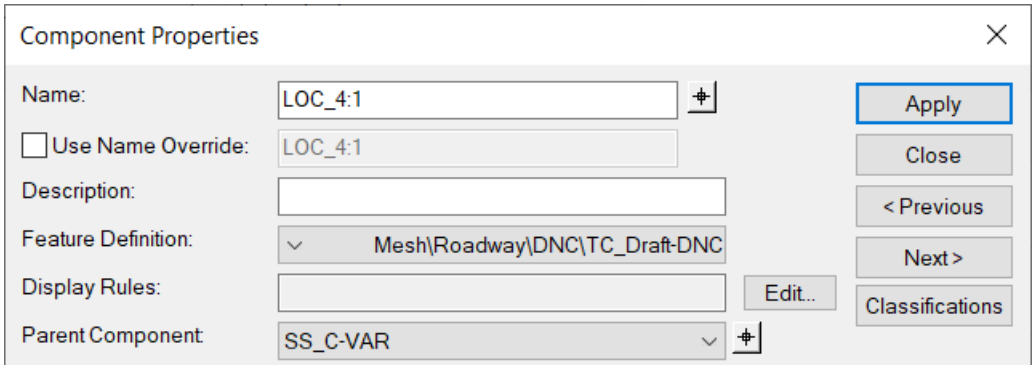
13. Edit point **1** and rename it **C_4:1**. Constrain it as shown below. It should have the following **End Condition Properties**.



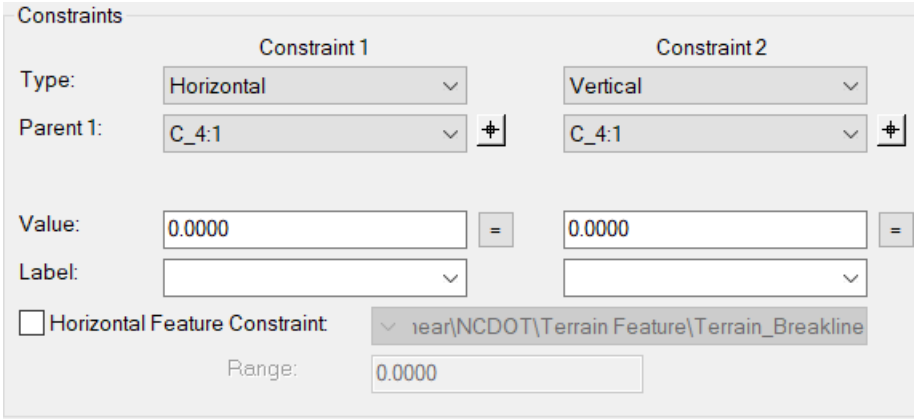
14. Create the LOC for this end condition. **Add New Component >>> Constrained** and starting at the **C_4:1** point create a line component to the right. Enter the following values in the component end condition properties prior to placement. The Feature Definition is "**Mesh\Roadway\DNC\TC_Draft-DNC**".



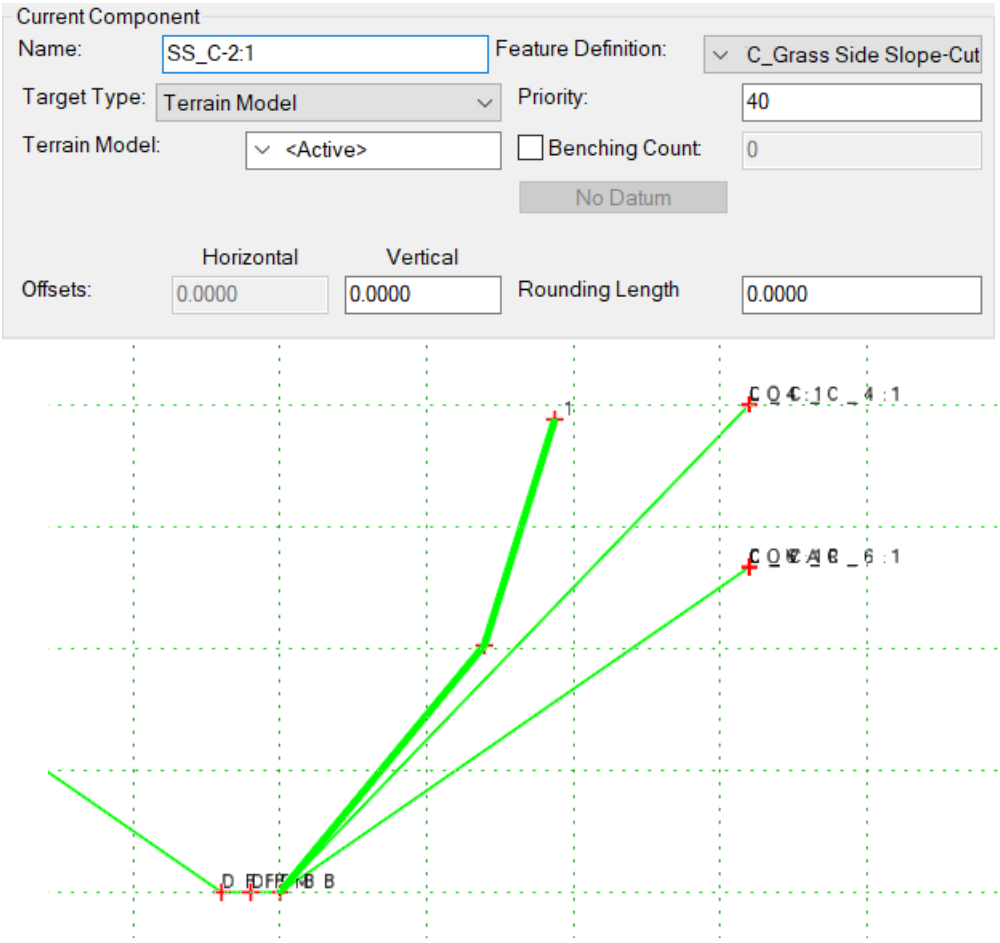
15. Make **SS_C-VAR** the **Parent Component**.



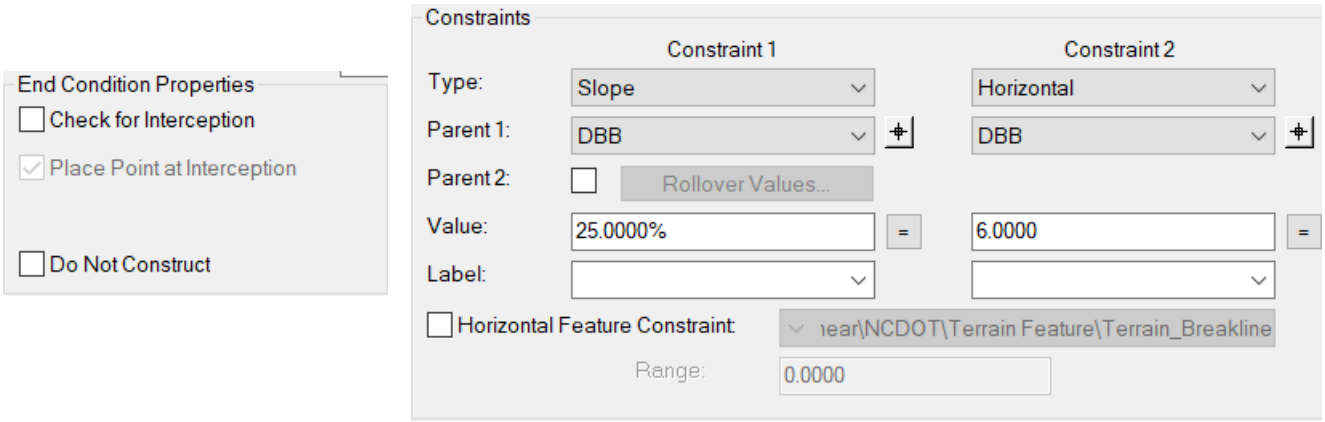
16. Edit the blank point and rename it **LOC_C-4:1**. Constrain it as shown below.



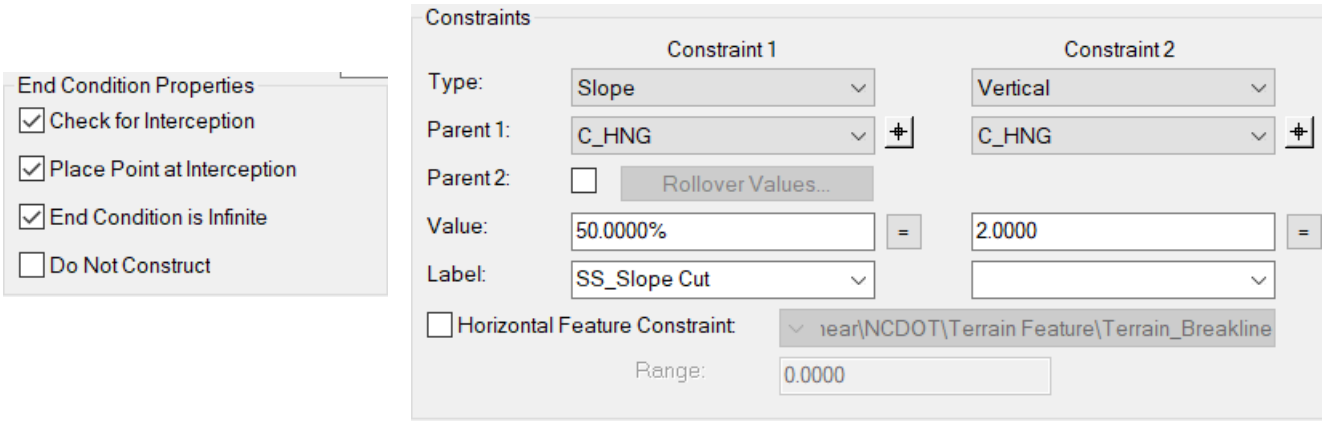
17. Create the last branch of the ditch. **Add New Component >>> End Condition** and starting at the **DBB** point create a 4:1 and 2:1 slope component. Enter the following values in the component end condition properties prior to placement. The Feature Definition is "**Mesh\Roadway\Grading\TC_Grass Side Slope-Cut**".



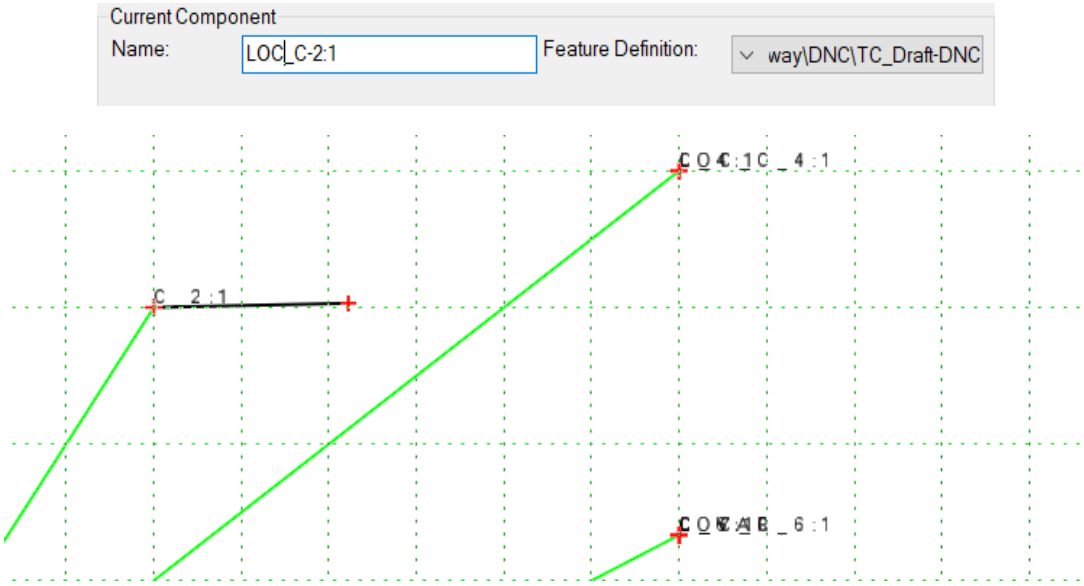
18. Edit the blank point and rename it **C_HNG**. Constrain it as shown below. It should have the following **End Condition Properties**.



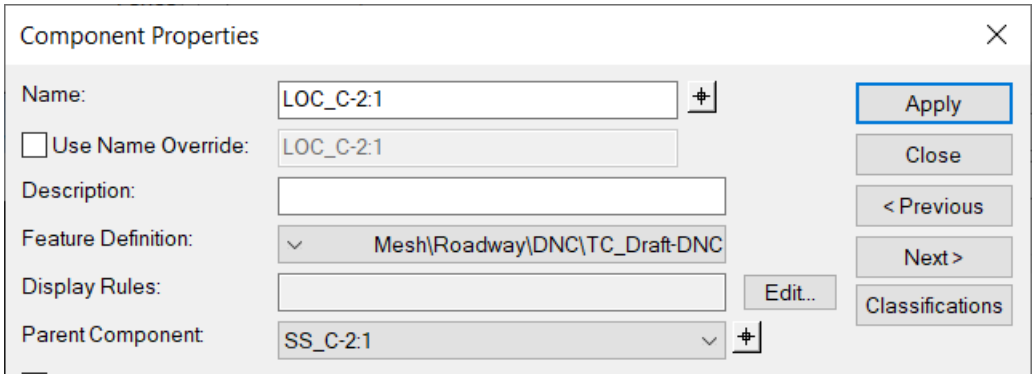
19. Edit point **1** and rename it **C_2:1**. Constrain it as shown below. It should have the following **End Condition Properties**.



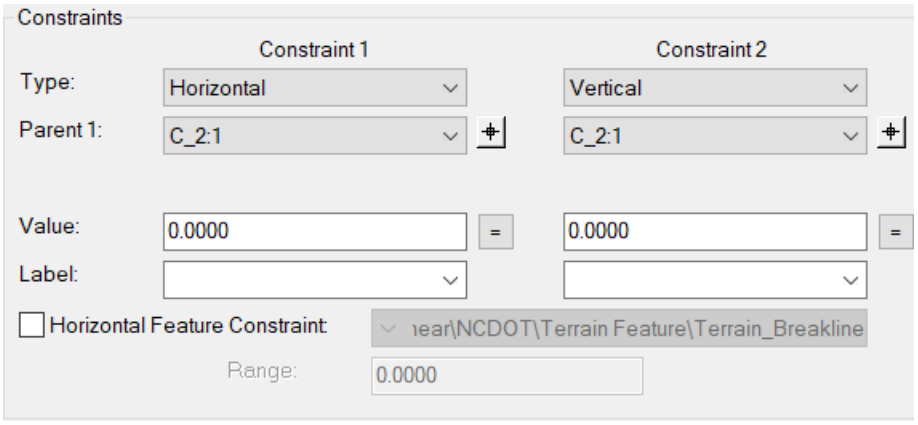
20. Create the LOC for this end condition. **Add New Component >>> Constrained** and starting at the **C_2:1** point create a line component to the right. Enter the following values in the component end condition properties prior to placement. The Feature Definition is "**Mesh\Roadway\DNC\TC_Draft-DNC**".



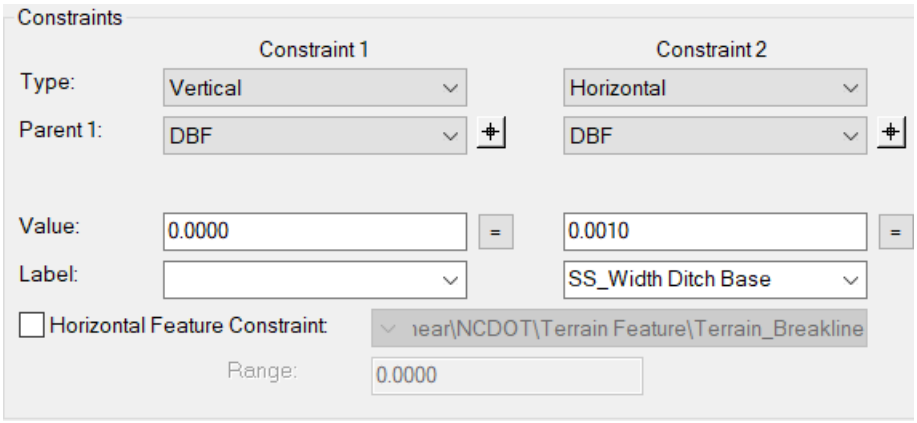
27. Make **SS_C-2:1** the **Parent Component**.



28. Edit the blank point and rename it **LOC_C_2:1**. Constrain it as shown below.

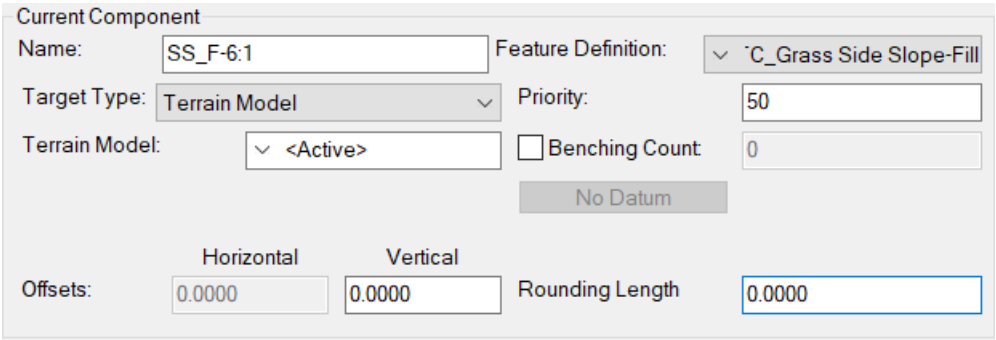


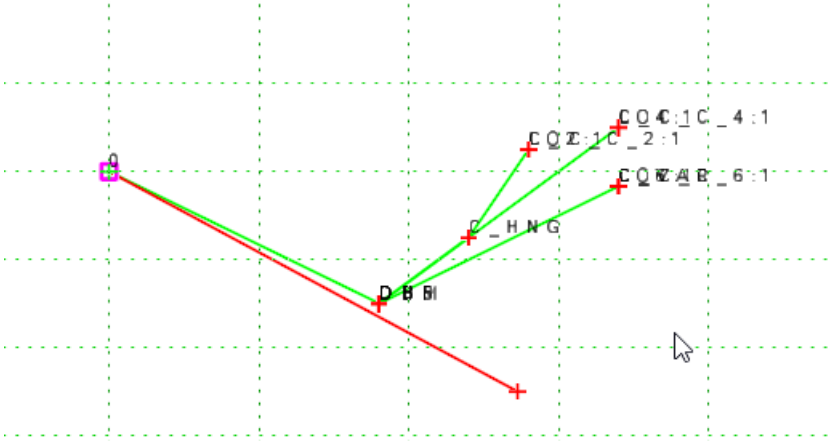
29. The last step for the cut ditch component is to change the default ditch base width to virtually zero. Edit the **DBB** point and constrain it as shown below.



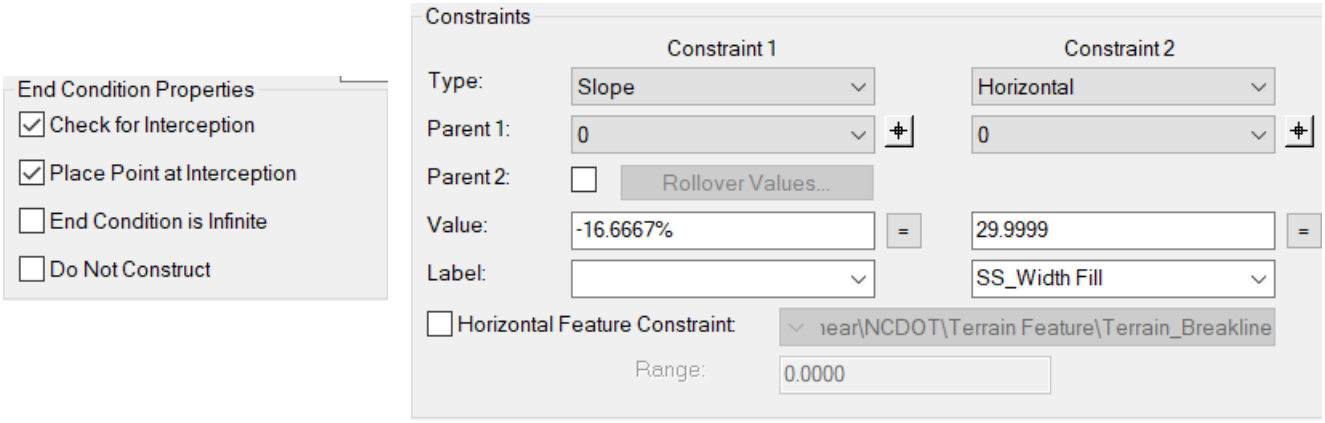
Construct the first fill slope component

30. **Add New Component >>> End Condition** starting from point **0** (zero) a draw a shape of the 6:1 fill slope below the ditch. Enter the following values in the component end condition properties prior to placement. The Feature Definition is "**Mesh\Roadway\Grading\TC_Grass Side Slope-Fill**".

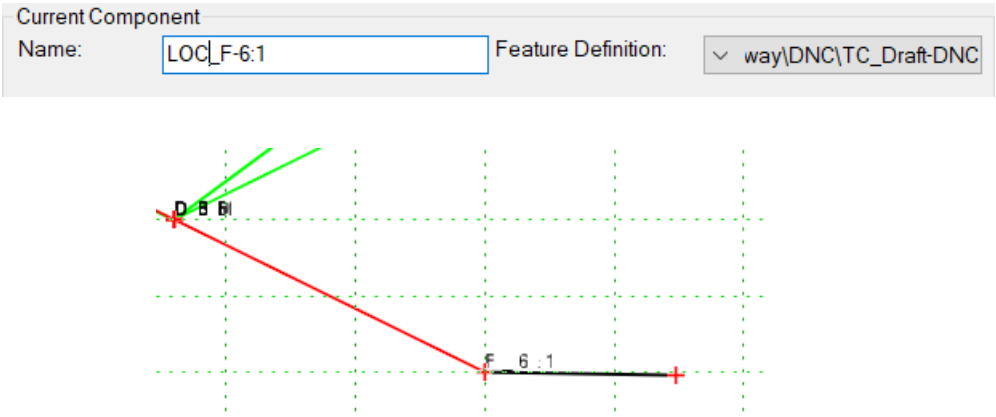




31. Edit the blank point and rename it **F_6:1**. Constrain it as shown below. It should have the following **End Condition Properties**.

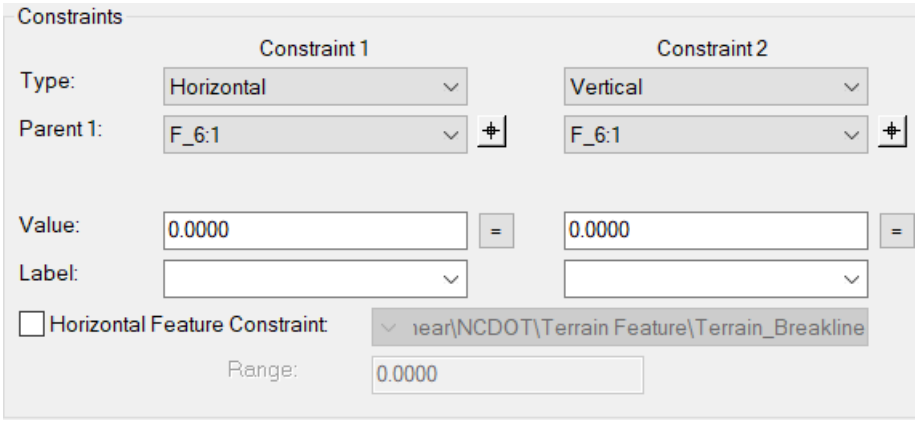


32. Create the LOC for this end condition. **Add New Component >>> Constrained** and starting at the **F_6:1** point create a line component to the right. Enter the following values in the component end condition properties prior to placement. The Feature Definition is "**Mesh\Roadway\DNC\TC_Draft-DNC**".

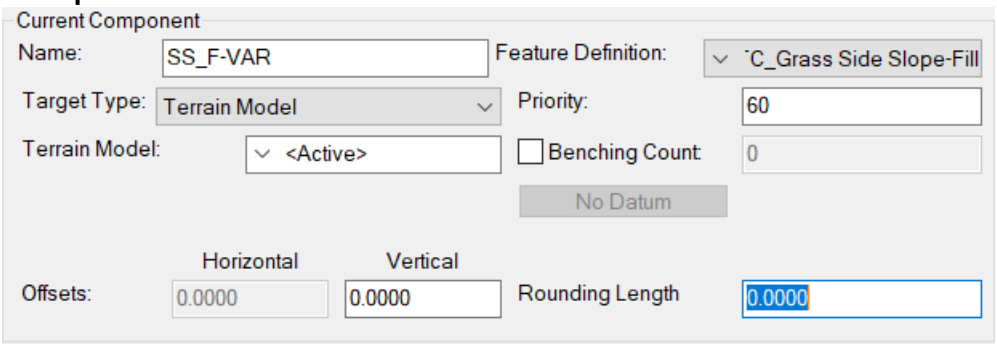


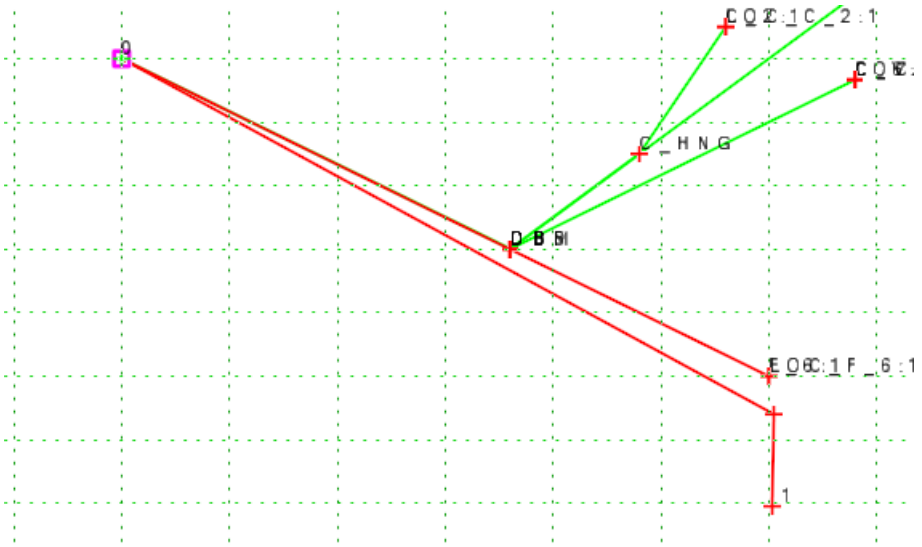
33. Make **SS_F-6:1** the Parent component.

34. Edit the blank point and rename it LOC_F_6:1. Constrain it as shown below.

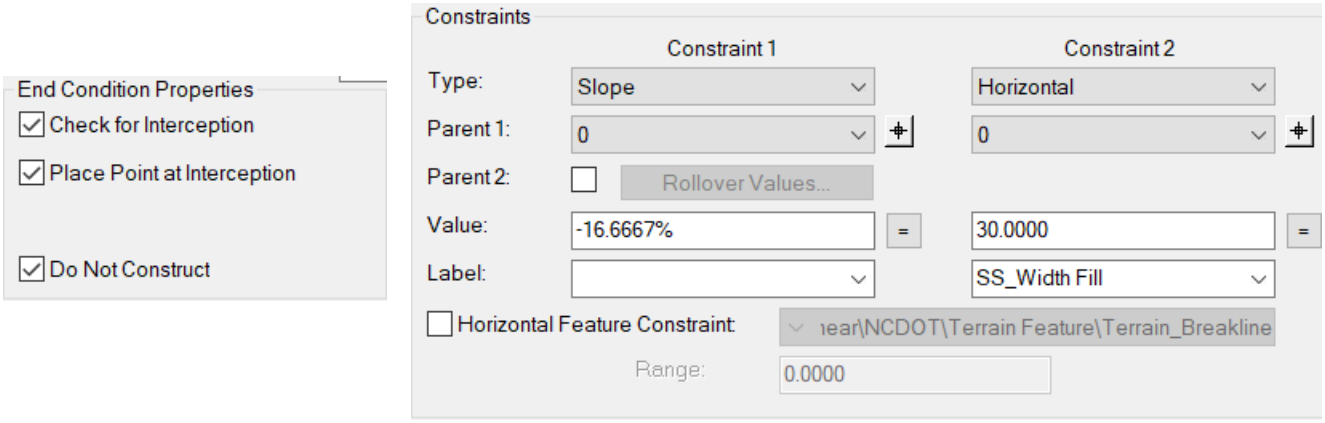


35. **Add New Component >>> End Condition** starting from point **0** (zero) a draw a shape of the variable 6:1 to 2:1 fill slope. Enter the following values in the component end condition properties prior to placement. The Feature Definition is "**Mesh\Roadway\Grading\TC_Grass Side Slope-Fill**".

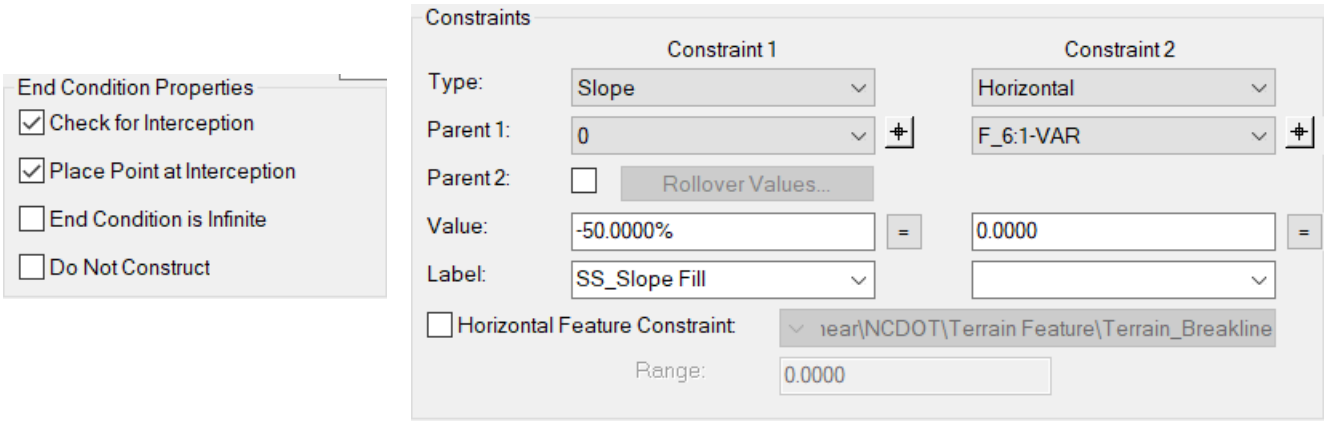




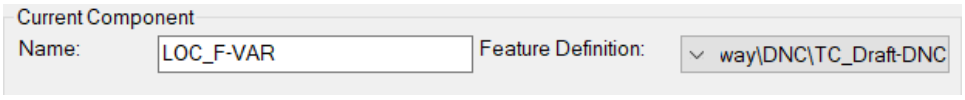
36. Edit the blank point and rename it **F_6:1-VAR**. Constrain it as shown below. It should have the following **End Condition Properties**.



37. Edit point **1** and rename it **F_2:1-VAR**. Constrain it as shown below. It should have the following **End Condition Properties**.

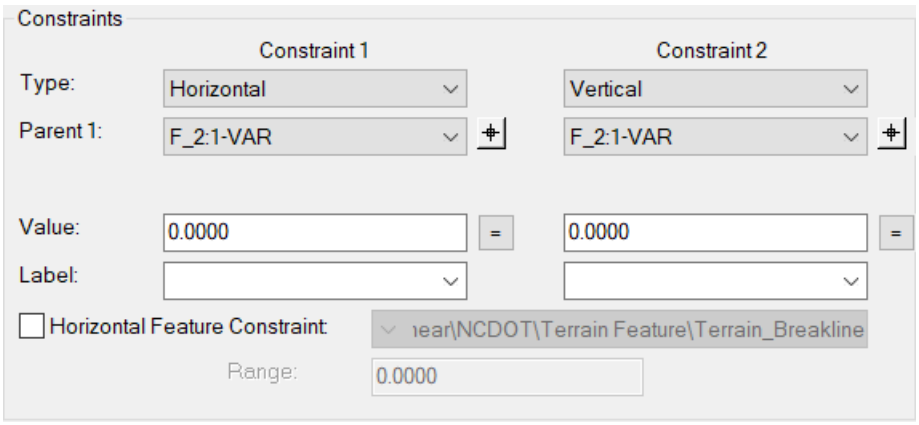


38. Create the LOC for this end condition. **Add New Component >>> Constrained** and starting at the **F_2:1-VAR** point create a line component to the right. Enter the following values in the component end condition properties prior to placement. The Feature Definition is "**Mesh\Roadway\DNC\TC_Draft-DNC**".

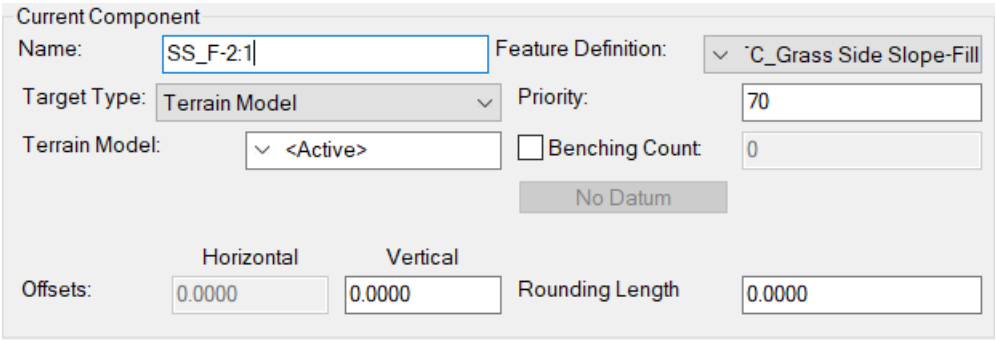


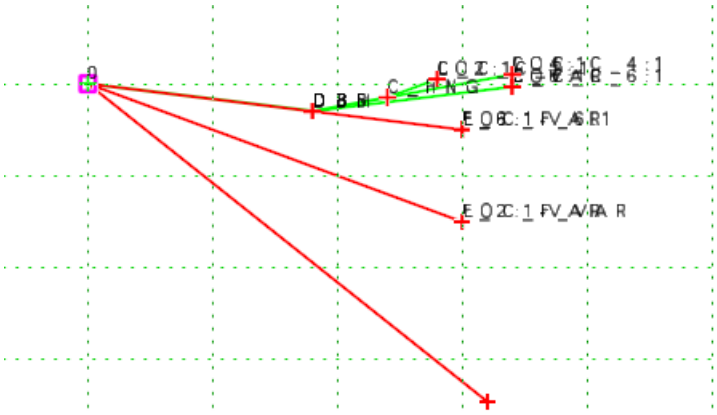
39. Make **SS_F-VAR** the **Parent Component**.

40. Edit the blank point and rename it **LOC_F_VAR**. Constrain it as shown below.

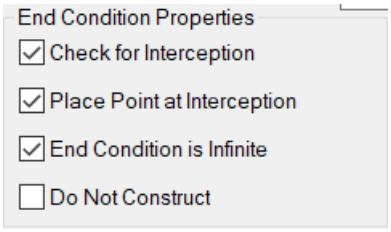


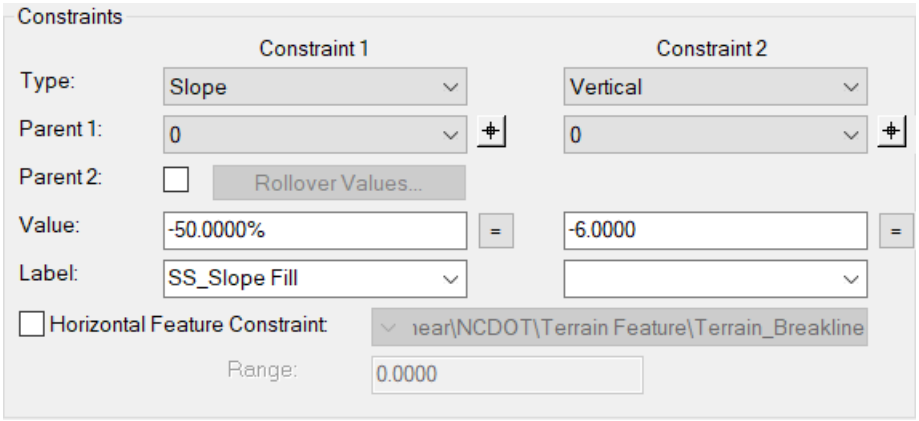
41. Construct the last 2:1 fill slope at an infinite depth. **Add New Component >>> End Condition** starting from point **0** (zero) a draw a shape of the 2:1 fill slope. Enter the following values in the component end condition properties prior to placement. The Feature Definition is "**Mesh\Roadway\Grading\TC_Grass Side Slope-Fill**".



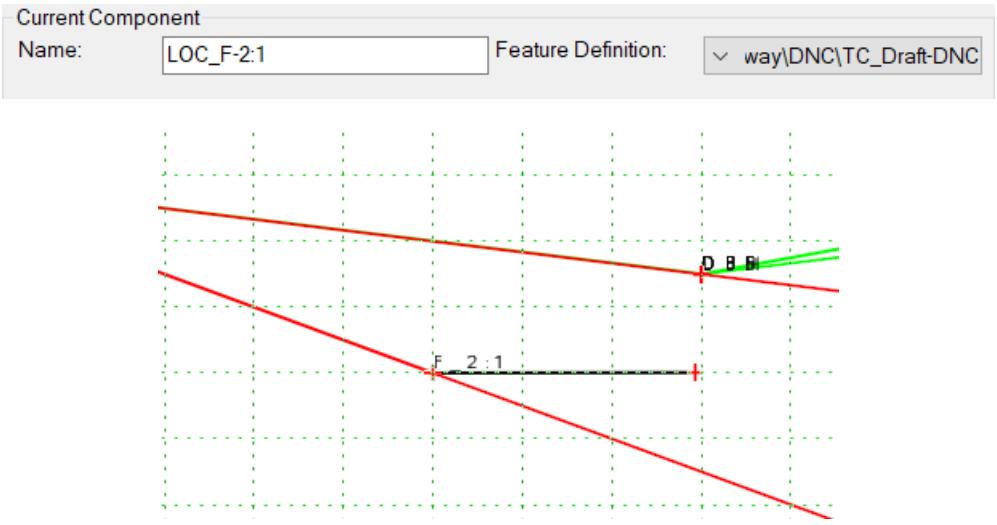


42. Edit the blank point and rename it **F_2:1**. Constrain it as shown below. It should have the following **End Condition Properties**.



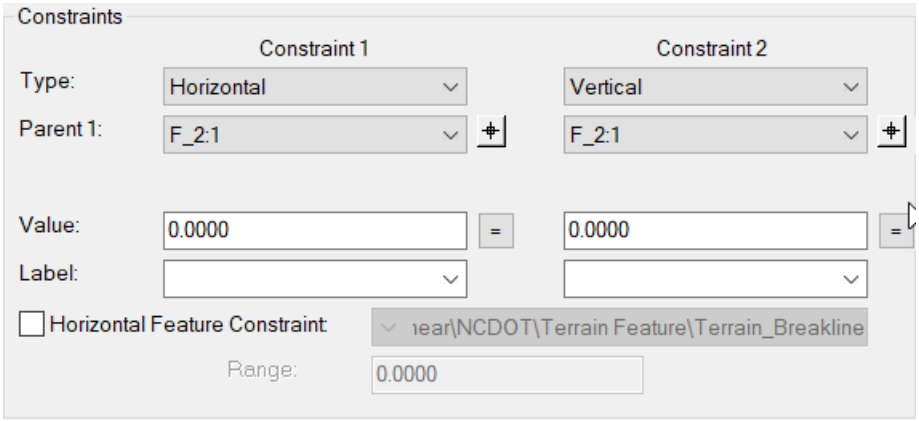


43. Create the LOC for this end condition. **Add New Component >>> Constrained** and starting at the **F_2:1** point create a line component to the right. Enter the following values in the component end condition properties prior to placement. The Feature Definition is "**Mesh\Roadway\DNC\TC_Draft-DNC**".

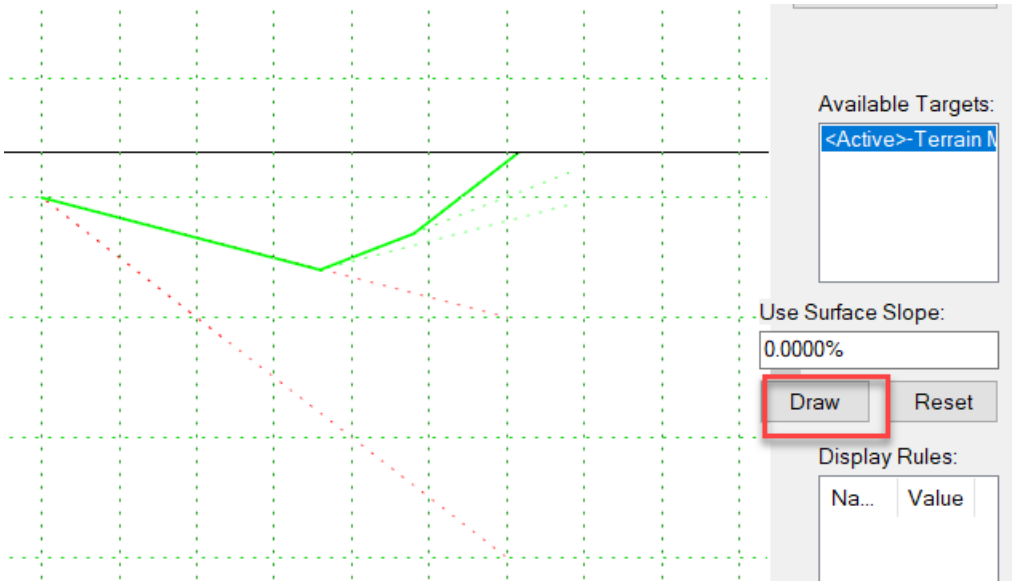


44. Make **SS_F_-2:1** the **Parent Component**.

45. Edit the blank point and rename it **LOC_F_2:1**. Constrain it as shown below.

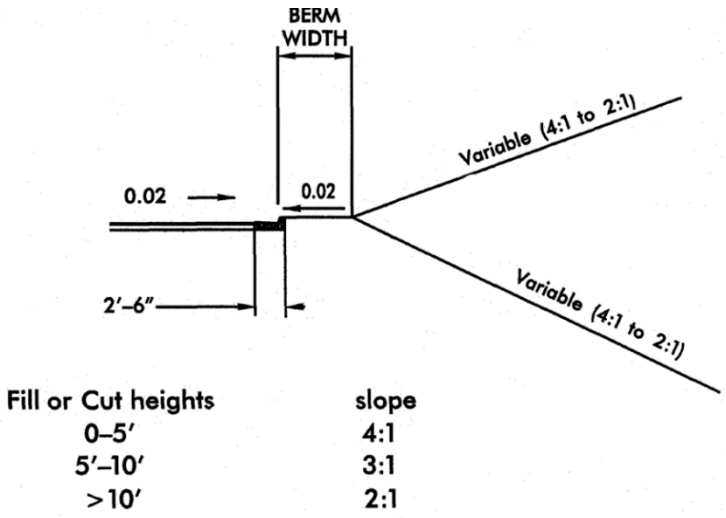


Test (bottom right button) the end condition.

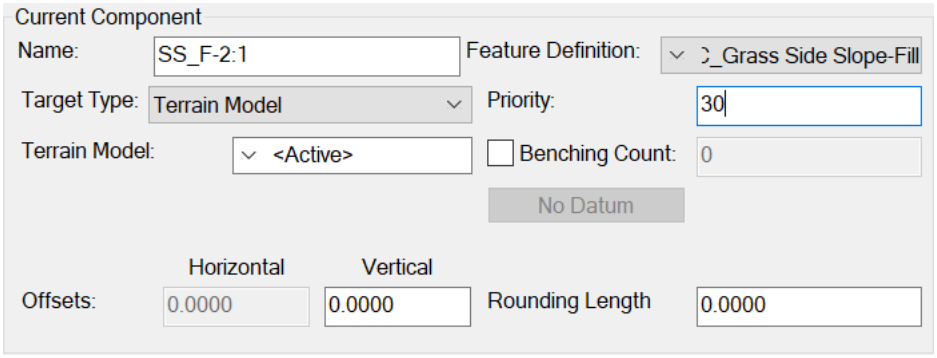


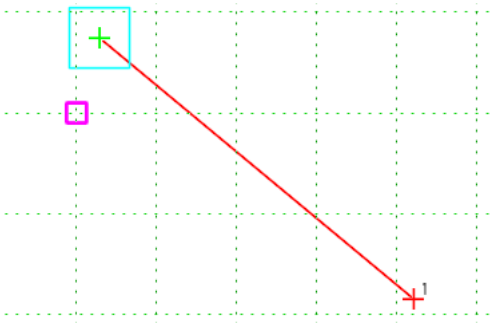
Create the **CSLP** end conditions.

In the Template Library **Open** the **02 Components\05 End Conditions** folder and create a new template called **CSLP**.

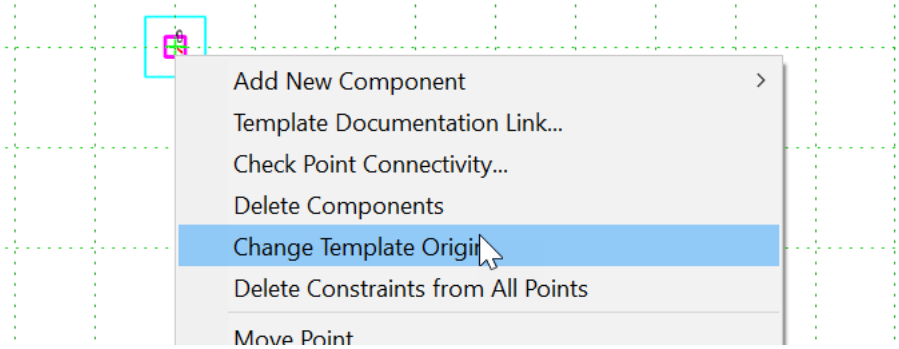


46. **Add New Component >>> End Condition** starting from point **0** (zero) a draw a shape of the 2:1 fill slope. Enter the following component end condition properties prior to placement.

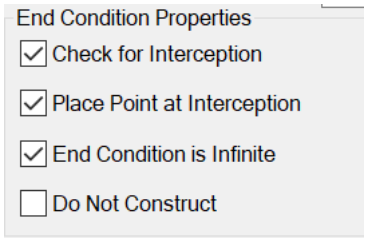


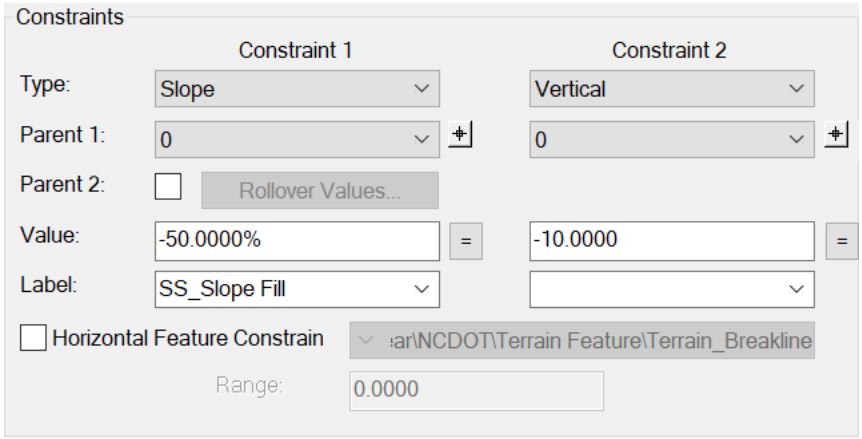


47. Rename the green blank point to **0** (zero) and make it the template origin.

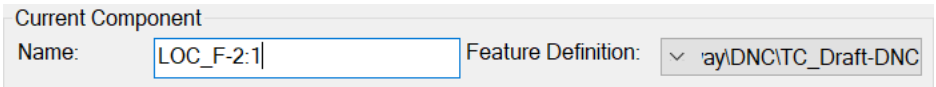


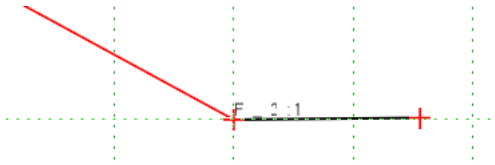
48. Edit point 1 and rename it **F_2:1**. Constrain it as shown below. It should have the following **End Condition Properties**.





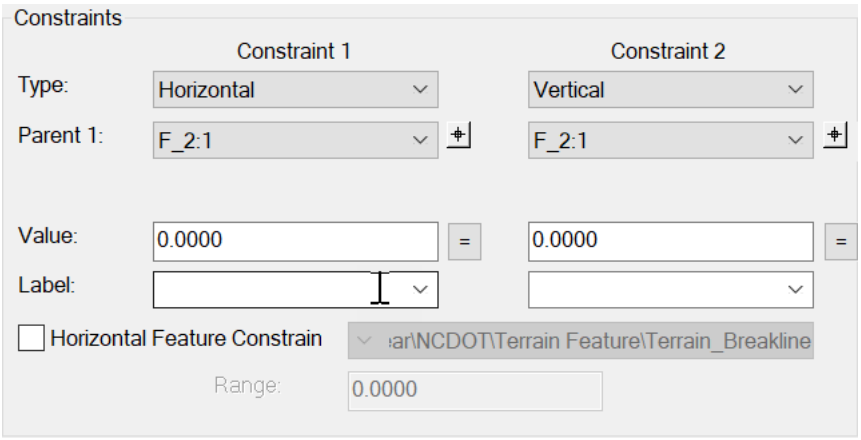
49. Create the LOC for this end condition. **Add New Component >>> Constrained** and starting at the **F_2:1** point create a line component to the right. Enter the following values in the component end condition properties prior to placement. The Feature Definition is "**Mesh\Roadway\DNC\TC_Draft-DNC**".



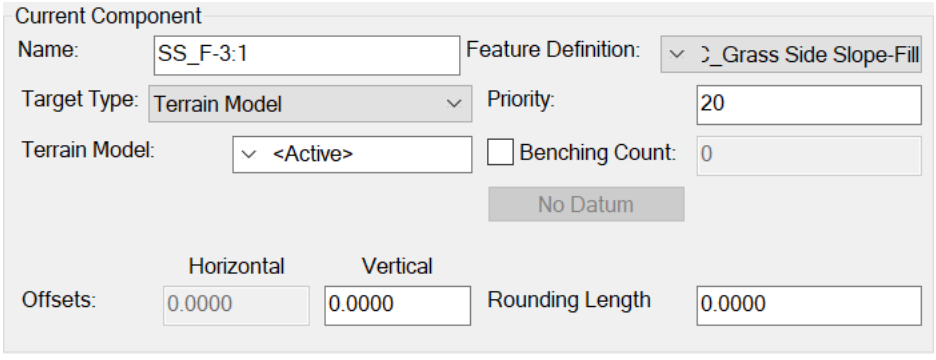


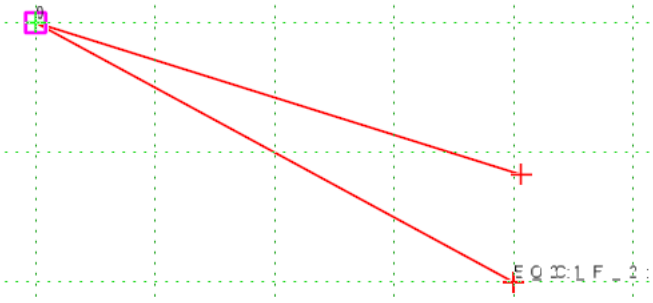
50. Make **SS_F-2:1** the **Parent Component**.

50. Edit the blank point and rename it **LOC_F_2:1**. Constrain it as shown below.

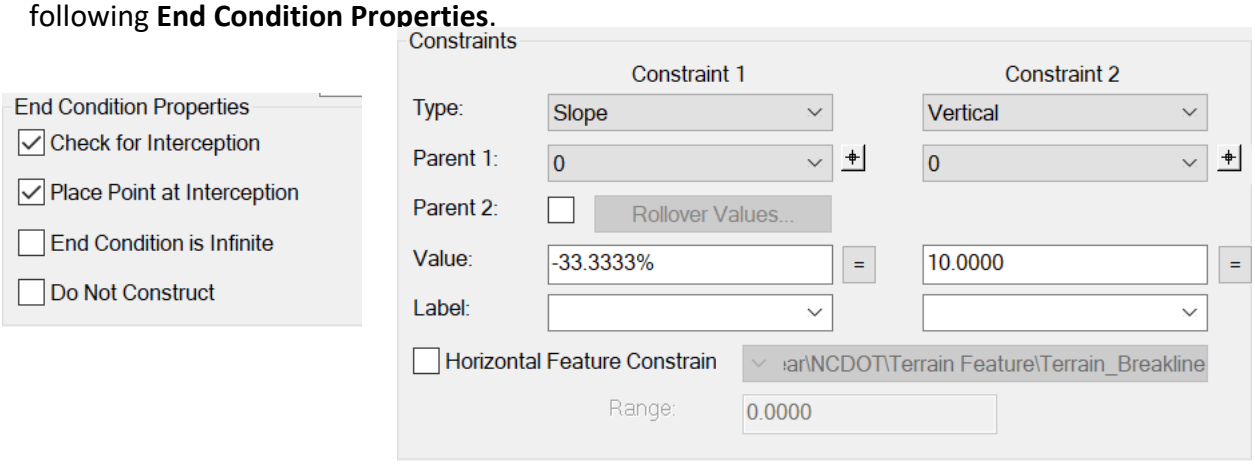


51. **Add New Component >>> End Condition** starting from point **0** (zero) a draw a shape of the 3:1 fill slope. Enter the following component end condition properties prior to placement.





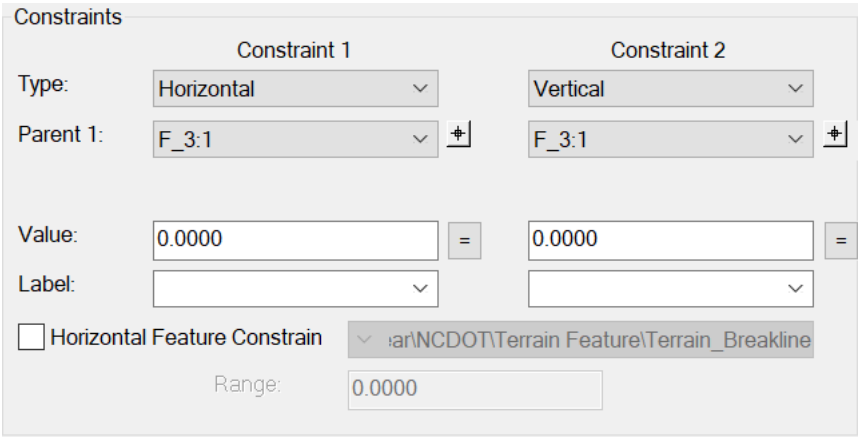
52. Edit the blank point and rename it **F_3:1**. Constrain it as shown below. It should have the following **End Condition Properties**.



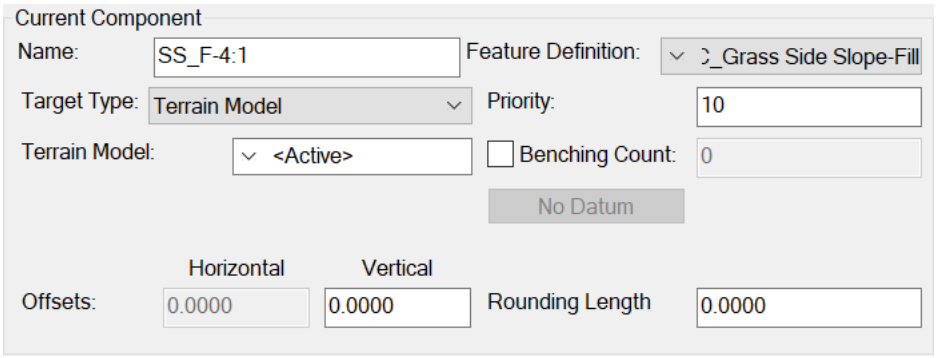
53. Create the LOC for this end condition. **Add New Component >>> Constrained** and starting at the **F_3:1** point create a line component to the right. Enter the following values in the component end condition properties prior to placement. The Feature Definition is "**Mesh\Roadway\DNC\TC_Draft-DNC**".

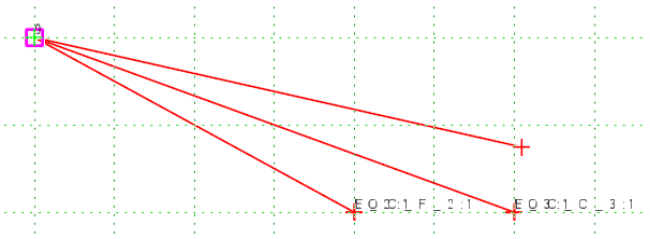
54. Make **SS_F-3:1** the **Parent Component**.

55. Edit the blank point and rename it **LOC_F_3:1**. Constrain it as shown below.

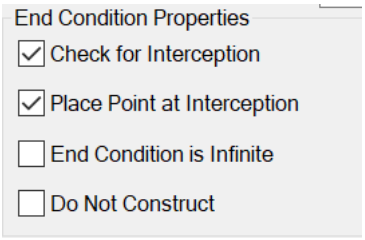


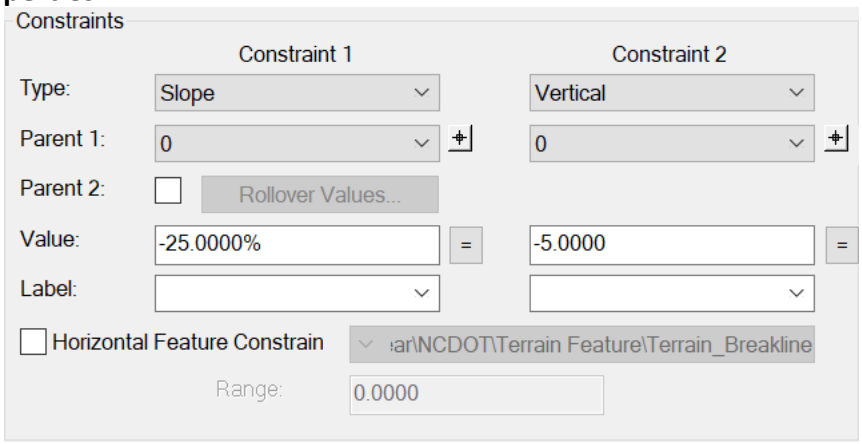
56. **Add New Component >>> End Condition** starting from point **0** (zero) a draw a shape of the 4:1 fill slope. Enter the following component end condition properties prior to placement.



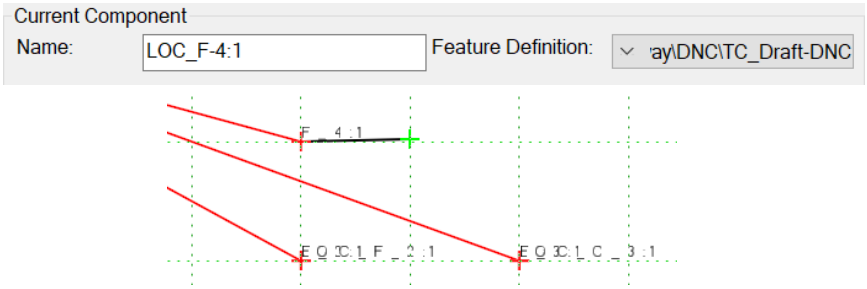


57. Edit the blank point and rename it **F_4:1**. Constrain it as shown below. It should have the following **End Condition Properties**.



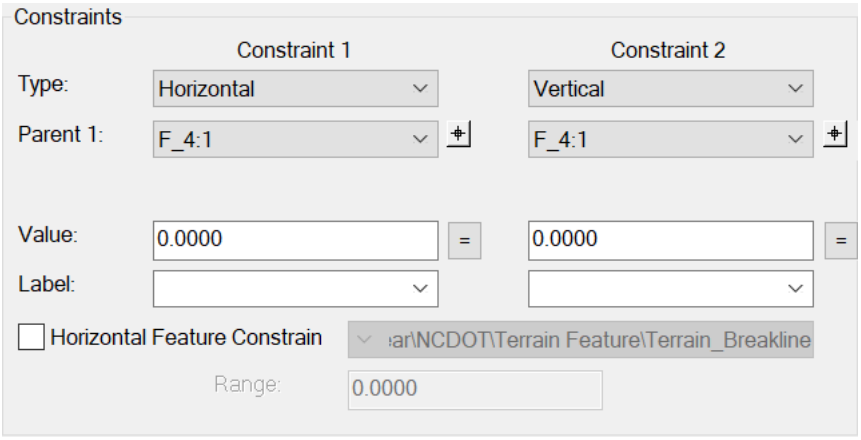


58. Create the LOC for this end condition. **Add New Component >>> Constrained** and starting at the **F_4:1** point create a line component to the right. Enter the following values in the component end condition properties prior to placement. The Feature Definition is "**Mesh\Roadway\DNC\TC_Draft-DNC**".

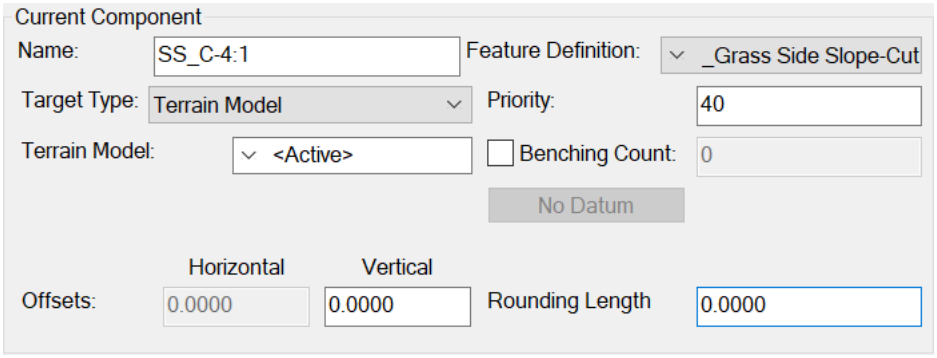


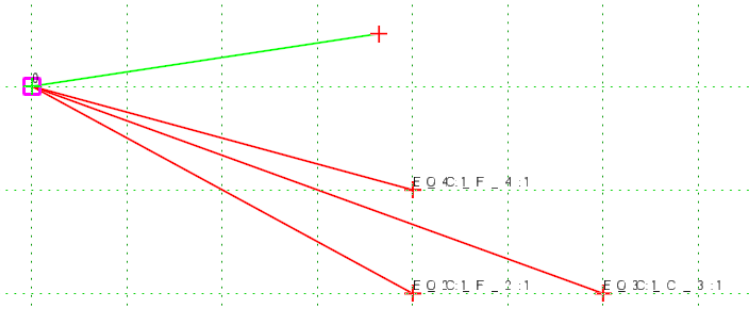
59. Make **SS_F-4:1** the **Parent Component**.

60. Edit the blank point and rename it **LOC_F_4:1**. Constrain it as shown below.

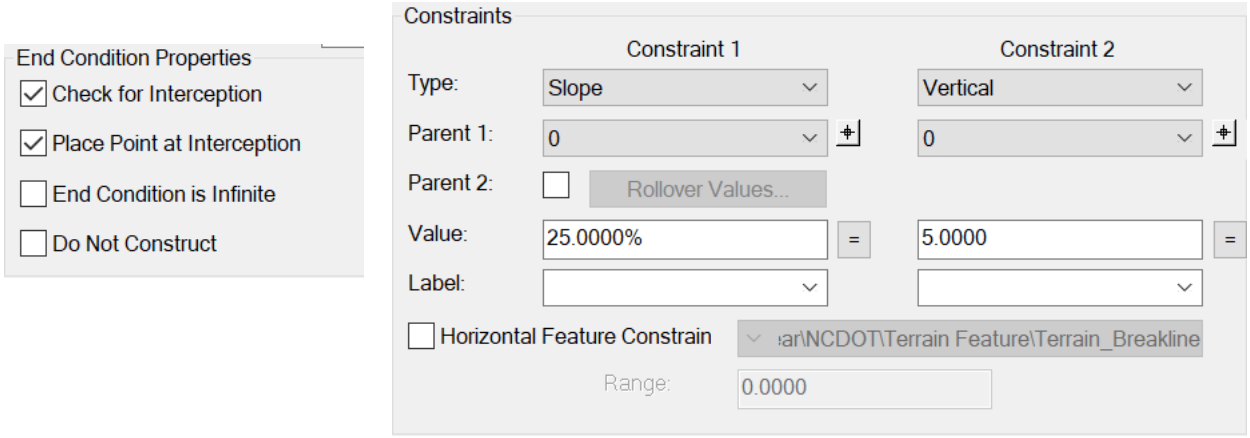


61. Create the cut slopes. **Add New Component >>> End Condition** starting from point **0** (zero) a draw a shape of the 4:1 cut slope. Enter the following component end condition properties prior to placement.

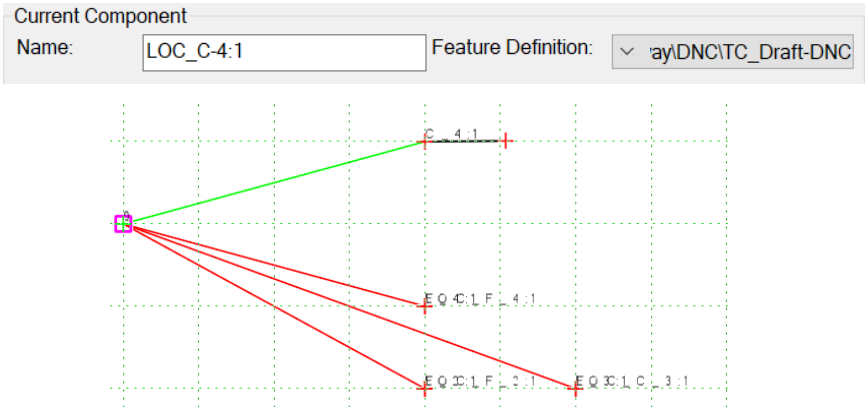




62. Edit the blank point and rename it **C_4:1**. Constrain it as shown below. It should have the following **End Condition Properties**.

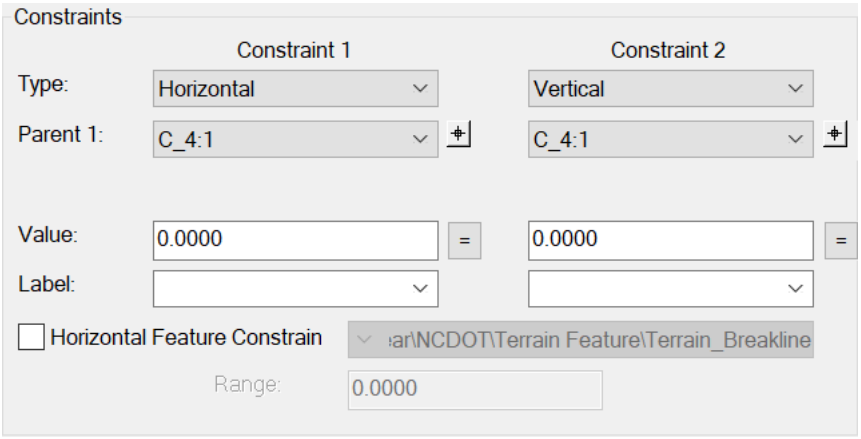


63. Create the LOC for this end condition. **Add New Component >>> Constrained** and starting at the **F_4:1** point create a line component to the right. Enter the following values in the component end condition properties prior to placement. The Feature Definition is "**Mesh\Roadway\DNC\TC_Draft-DNC**".

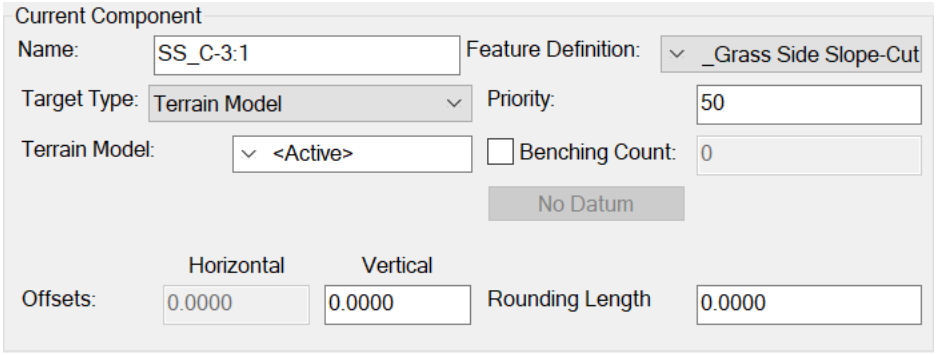


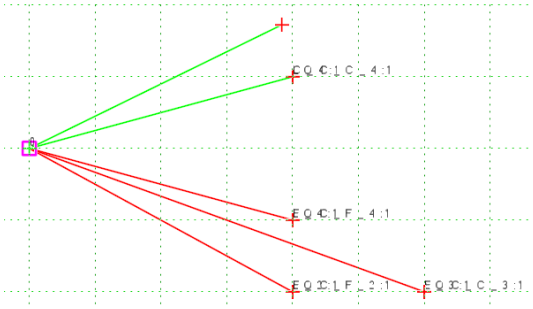
64. Make **SS_C-4:1** the **Parent Component**.

65. Edit the blank point and rename it **LOC_C_4:1**. Constrain it as shown below.

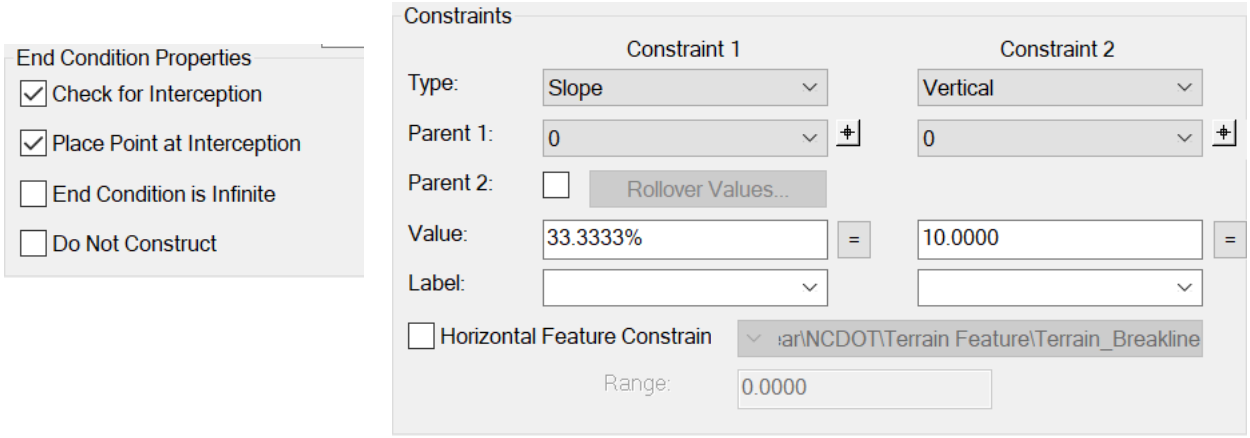


66. **Add New Component >>> End Condition** starting from point **0** (zero) a draw a shape of the 3:1 cut slope. Enter the following component end condition properties prior to placement.

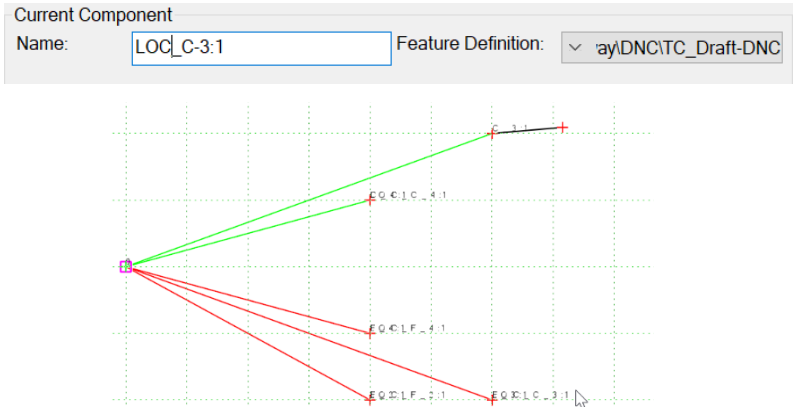




67. Edit the blank point and rename it **C_3:1**. Constrain it as shown below. It should have the following **End Condition Properties**.

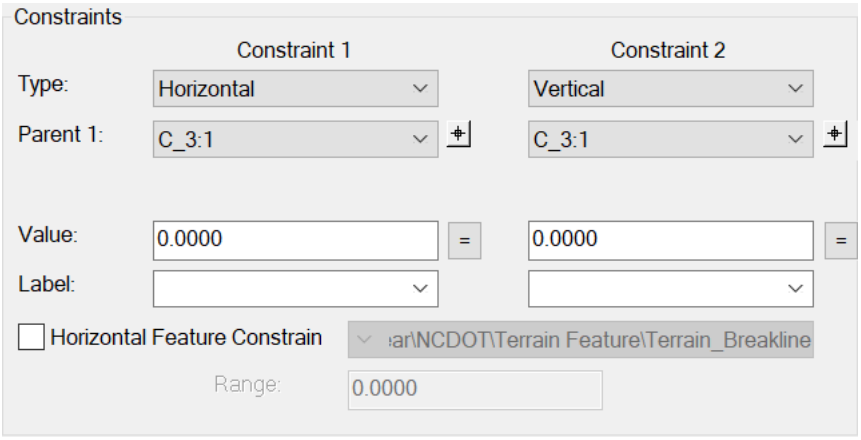


68. Create the LOC for this end condition. **Add New Component >>> Constrained** and starting at the **F_3:1** point create a line component to the right. Enter the following values in the component end condition properties prior to placement. The Feature Definition is "**Mesh\Roadway\DNC\TC_Draft-DNC**".

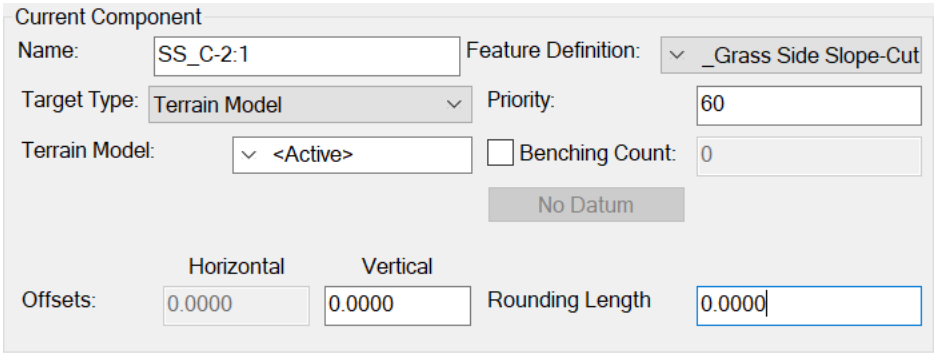


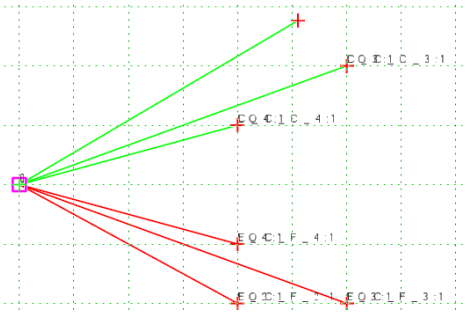
69. Make **SS_C-3:1** the **Parent Component**.

70. Edit the blank point and rename it **LOC_C_3:1**. Constrain it as shown below.

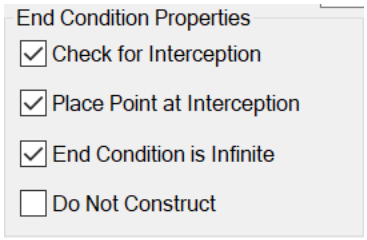


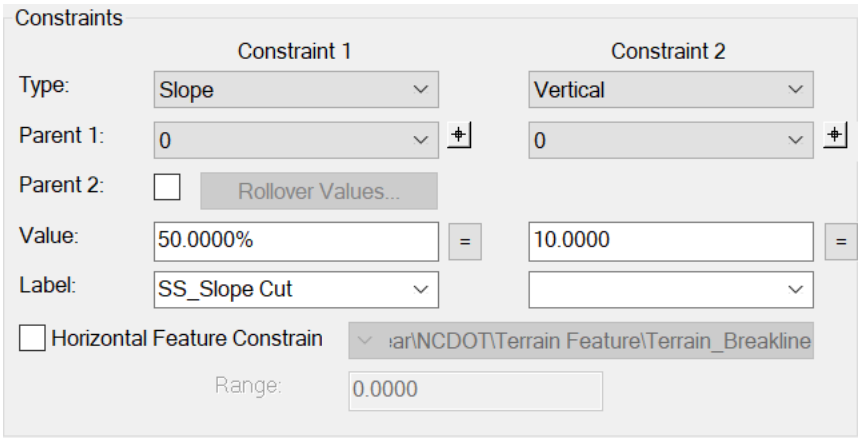
71. **Add New Component >>> End Condition** starting from point **0** (zero) a draw a shape of the 2:1 cut slope. Enter the following component end condition properties prior to placement.



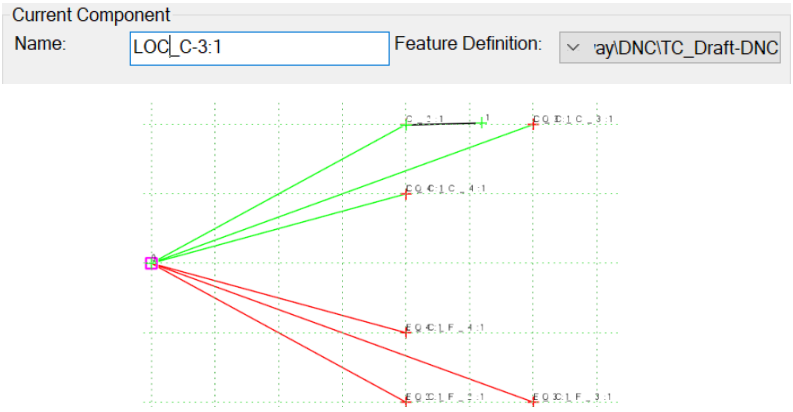


72. Edit the blank point and rename it **C_2:1**. Constrain it as shown below. It should have the following **End Condition Properties**.



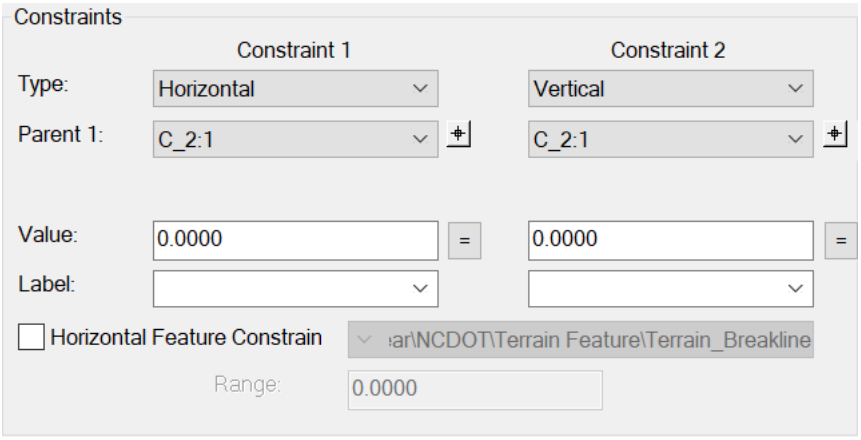


73. Create the LOC for this end condition. **Add New Component >>> Constrained** and starting at the **F_2:1** point create a line component to the right. Enter the following values in the component end condition properties prior to placement. The Feature Definition is "**Mesh\Roadway\DNC\TC_Draft-DNC**".

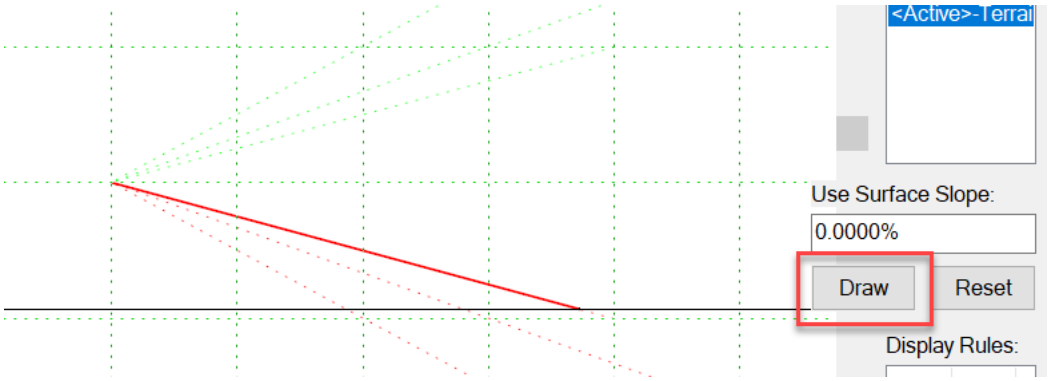


74. Make **SS_C-2:1** the **Parent Component**.

75. Edit the blank point and rename it **LOC_C_2:1**. Constrain it as shown below.



Test (bottom right button) the end condition.



Exercise C7: Median Ditch Components

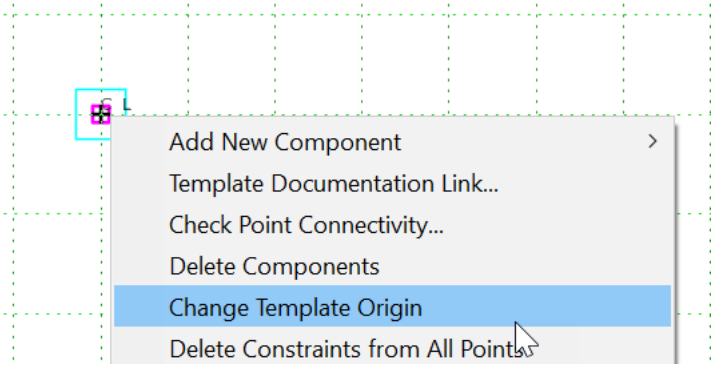
In this exercise we will go over how to create the two (2) types of median ditches. Topics covered in this training include **Null Points**, **Vertical Maximum** and **Vertical Minimum**.

A **Type 1** Median Ditch has variable slopes. The ditch point remains at the Centerline.

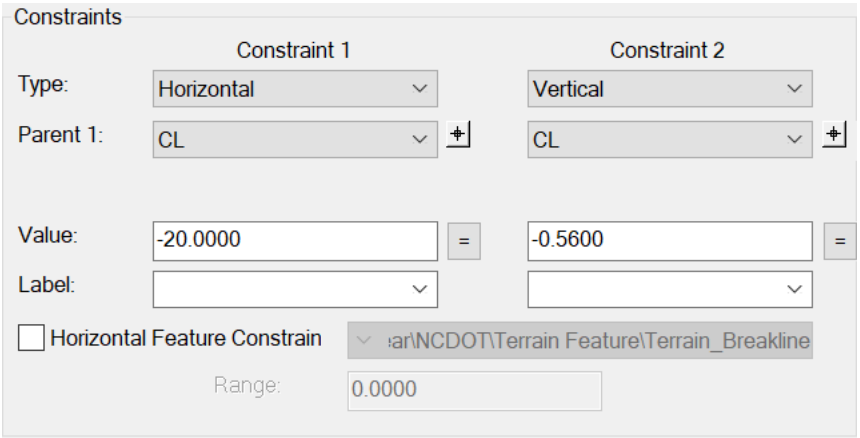
A **Type 2** Median Ditch has fixed slopes. The ditch point varies horizontally depending on the lowside or highside of superelevation.

In the Template Library **Open** the **02 Components\07 Median Ditch** folder and create a new template named **Median Ditch - Type 1**.

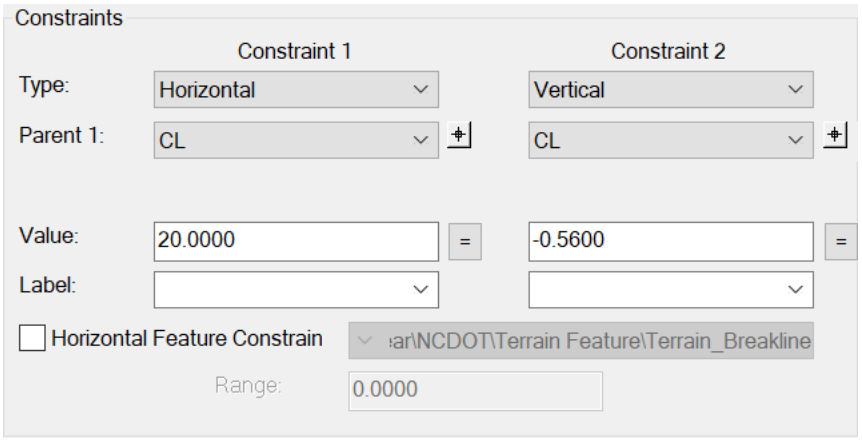
1. **Add New Component >>> Null Point** and name it **CL**. Make this the template origin.



2. **Add New Component >>> Null Point** to the left of the **CL** point and name it **GSI_NL**. Constrain it as shown below.

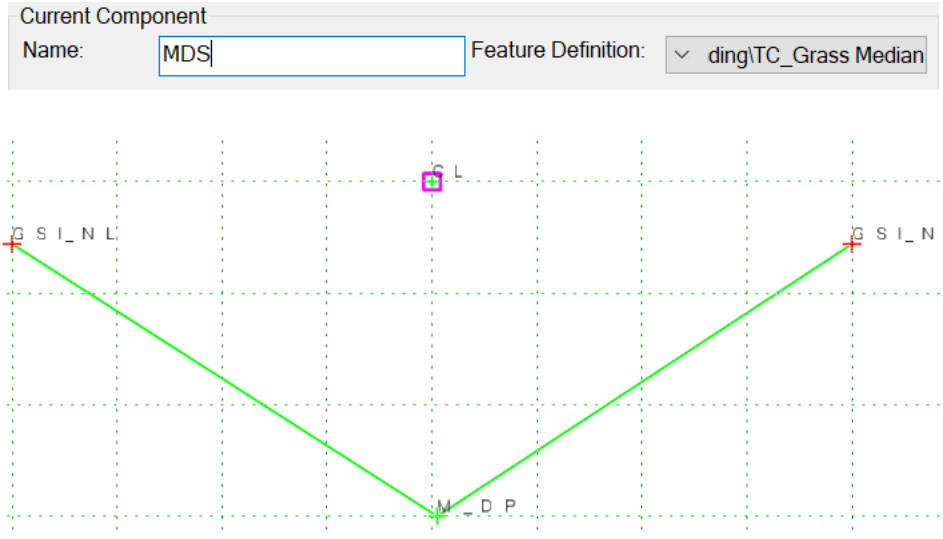


3. **Add New Component >>> Null Point** to the right of the **CL** point and name it **GSI_NR**. Constrain it as shown below.

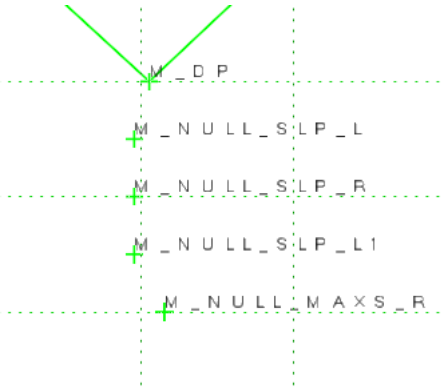


4. **Add New Component >>> Null Point** below the **CL** point and name it **M_DP**.

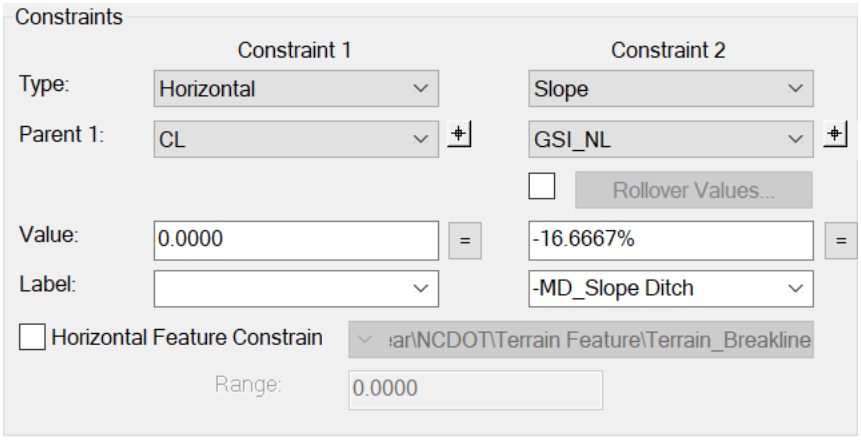
5. Draw the median ditch component. **Add New Component >>> Constrained** starting with the **GSI_NL** point connect to **M_DP** and end with the **GSI_NR** point to form the median ditch. Enter the following values in the component end condition properties prior to placement. The Feature Definition is "**Mesh\Roadway\Grading\TC_Grass Median**". Uncheck **Closed Shape**.



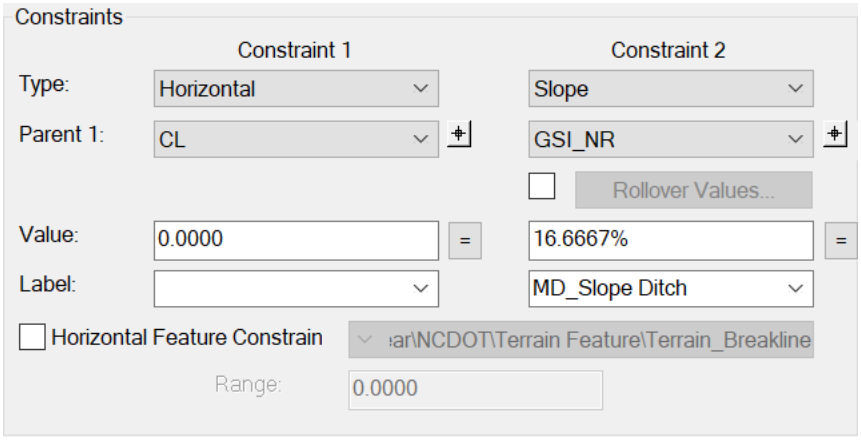
6. Add four (4) null points underneath the **M_DP** point. Their names should be **M_NULL SLP_L**, **M_NULL SLP_R**, **M_NULL MAXS_L** and **M_NULL MAXS_R**.



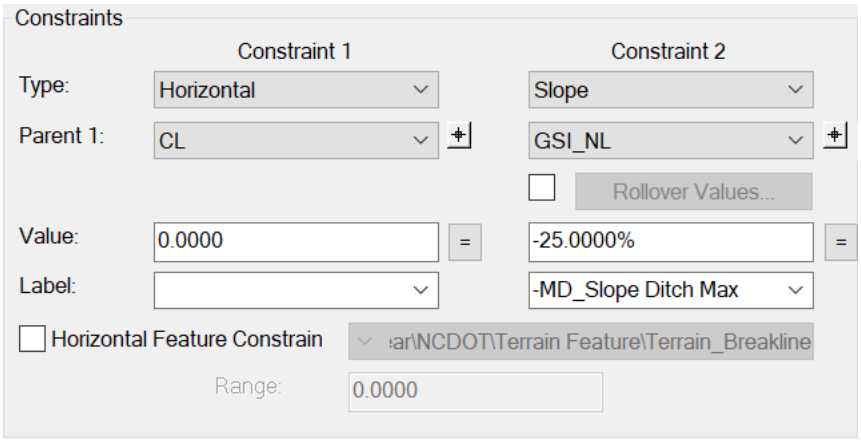
7. Constrain the **M_NULL_SLP_L** as shown below. This is the normal **6:1** fixed ditch slope.



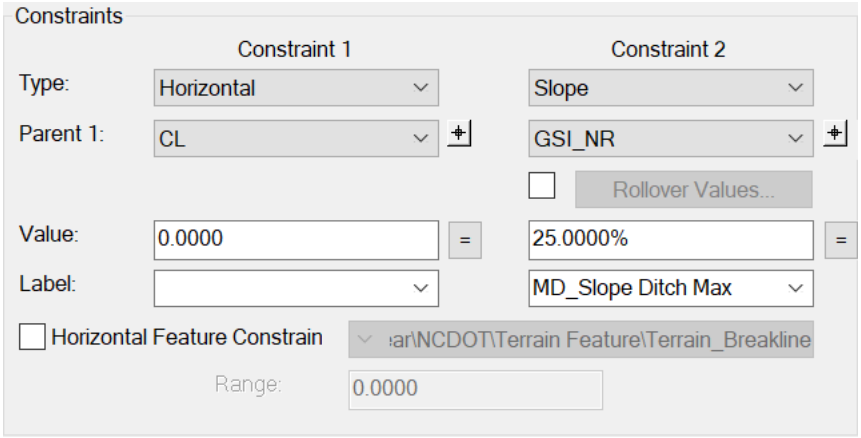
8. Constrain the **M_NULL_SLP_R** as shown below. This is the normal **6:1** fixed ditch slope.



9. Constrain the **M_NULL_MAXS_L** as shown below. This is the maximum (steepest) **4:1** ditch slope.

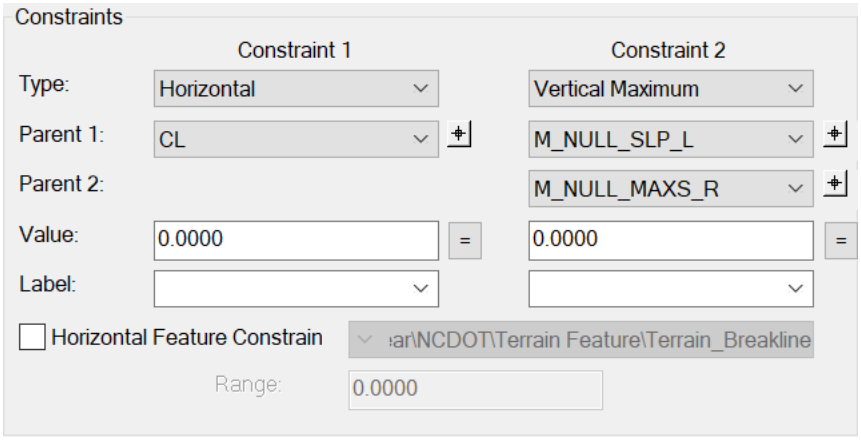


10. Constrain the **M_NULL_MAXS_R** as shown below. This is the maximum (steepest) **4:1** ditch slope.

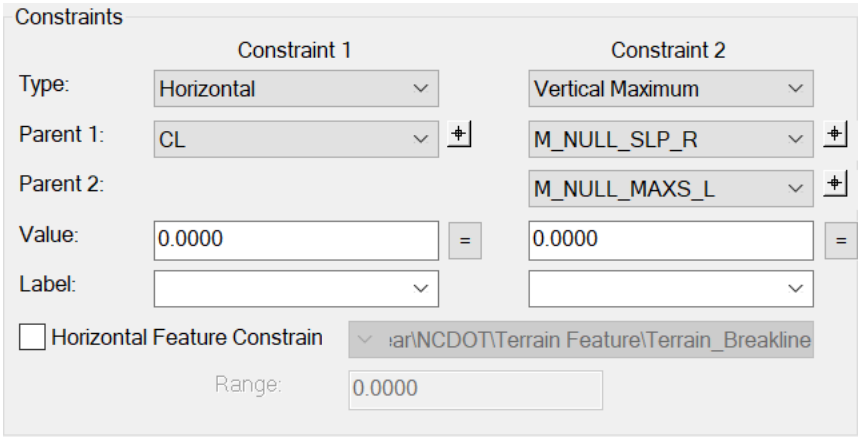


11. Create two (2) null points underneath the M_DP point and name them M_NULL_VMAX_L and M_NULL_VMAX_R. The purpose of these points is to compare the slope of the ditch to the maximum (steepest) slope on the opposite side and taking the higher (vertical maximum) of the two (2). On the low side of superelevation the maximum slope (4:1) is used resulting in the low side slope being variable (no longer fixed at 6:1). The high side slope is maintained at 4:1 (maximum) while the low side becomes variable (flatter than 6:1).

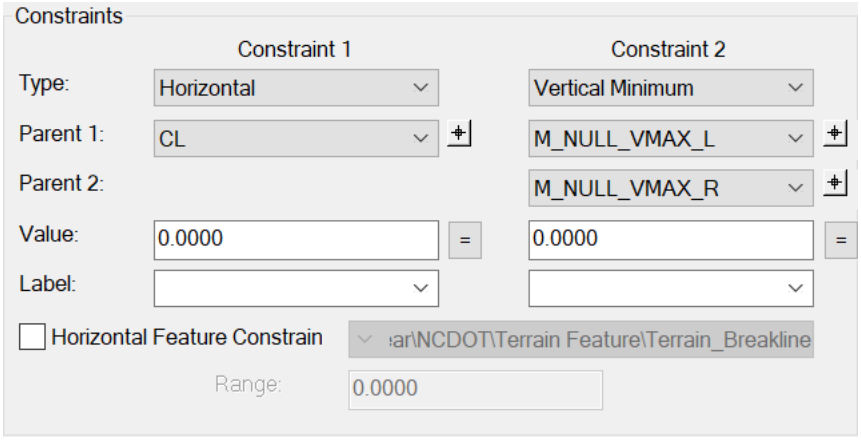
12. Constrain the **M_NULL_VMAX_L** as shown below. This is taking the higher (vertical maximum) between the **6:1** normal slope on the left and the maximum (steepest) **4:1** ditch slope on the right for low side condition.



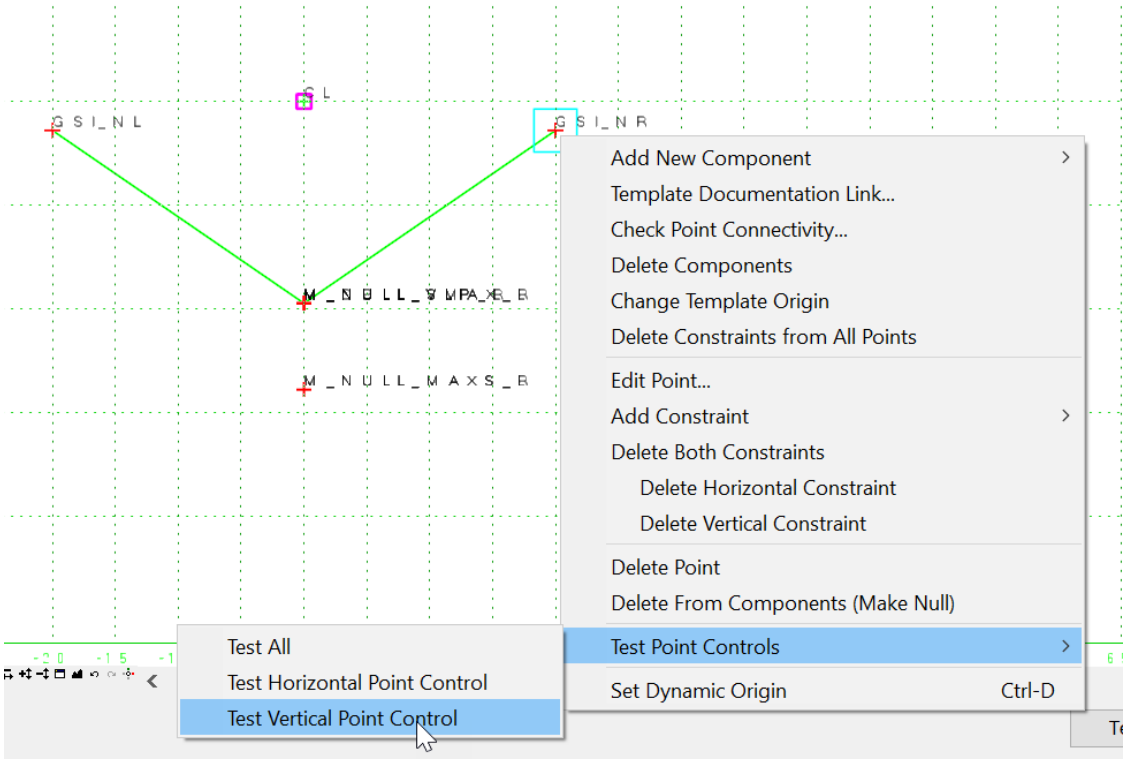
13. Constrain the **M_NULL_VMAX_R** as shown below. This is taking the higher (vertical maximum) between the **6:1** normal slope on the right and the maximum (steepest) **4:1** ditch slope on the left for low side condition.



14. Lastly, constrain the **M_DP** point as shown below. The final ditch point elevation is dependent on the lower (vertical minimum) between the **M_NULL_VMAX_L** and **M_NULL_VMAX_R** points.



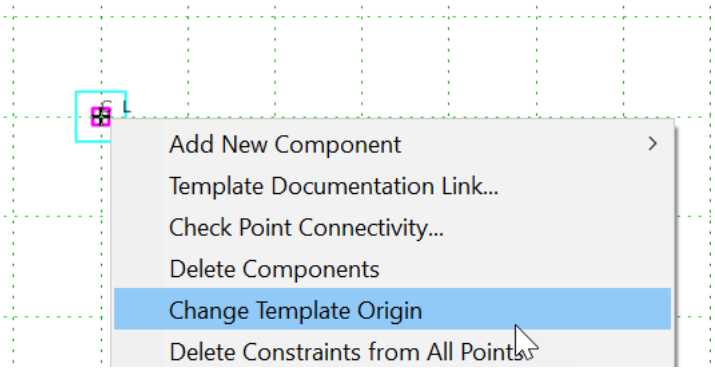
15. Test the component is behaving as designed. To simulate the high and low side of superelevation, right mouse click on the point and **Test Point Controls >>> Test Vertical Control**. Use either the **GSI_NL** or **GSI_NR** point.



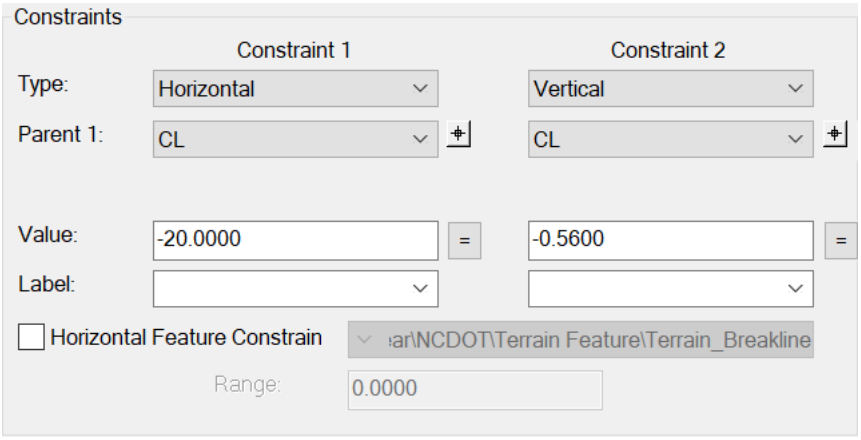
A **Type 2** Median Ditch has fixed slopes. The ditch point varies horizontally depending on the lowside or highside of superelevation.

In the Template Library **Open** the **02 Components\06 Median Ditch** folder and create a new template named **Median Ditch - Type 2**.

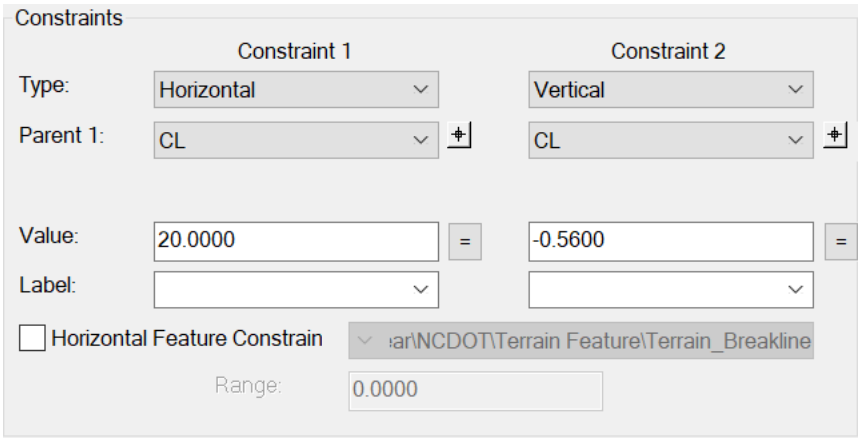
1. **Add New Component >>> Null Point** and name it **CL**. Make this the template origin.



2. **Add New Component >>> Null Point** to the left of the **CL** point and name it **GSI_NL**. Constrain it as shown below.

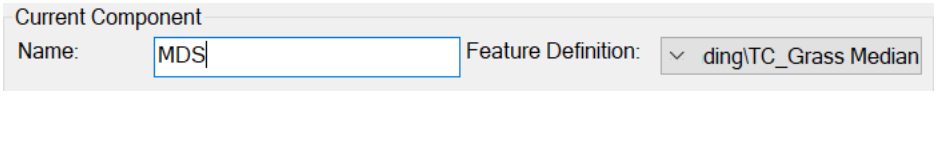


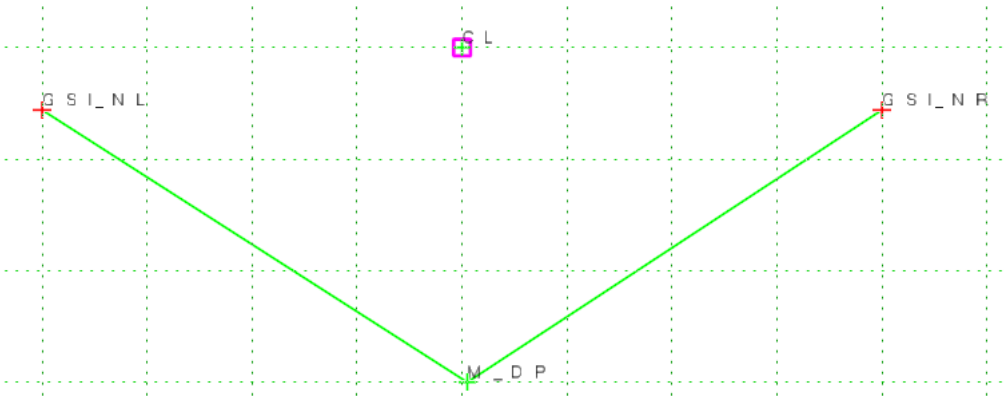
3. **Add New Component >>> Null Point** to the right of the **CL** point and name it **GSI_NR**. Constrain it as shown below.



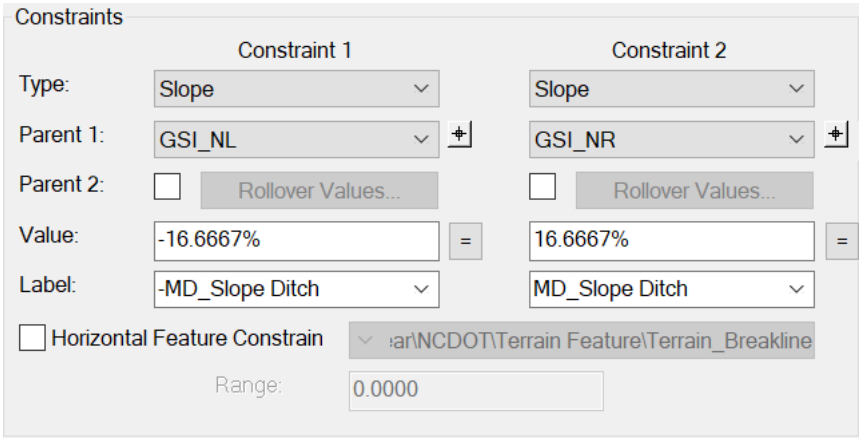
4. Add New Component >>> Null Point below the CL point and name it **M_DP**.

5. Draw the Median ditch component. **Add New Component >>> Constrained** starting with the **GSI_NL** point connect to **M_DP** and end with the **GSI_NR** point to form the median ditch. Enter the following values in the component end condition properties prior to placement. The Feature Definition is "**Mesh\Roadway\Grading\TC_Grass Median**". Uncheck **Closed Shape**.

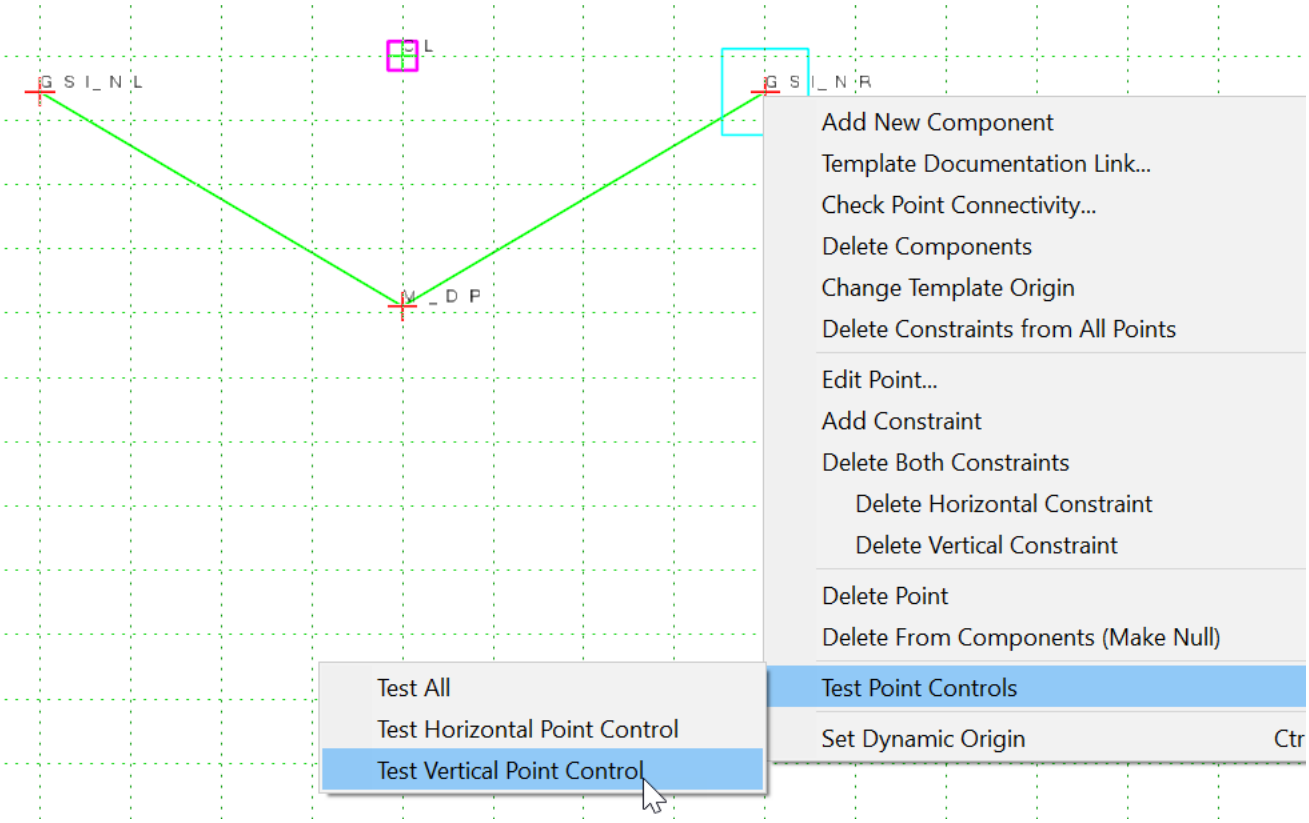




6. Constrain the M_DP point as shown below. Note only one parametric constraint is required for both sides of the median ditch.



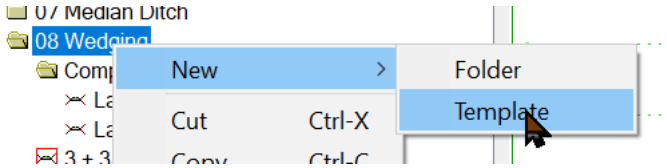
7. Test the component is behaving as designed. To simulate the high and low side of superelevation, right mouse click on the point and **Test Point Controls >>> Test Vertical Control**. Use either the **GSI_NL** or **GSI_NR** point**.**



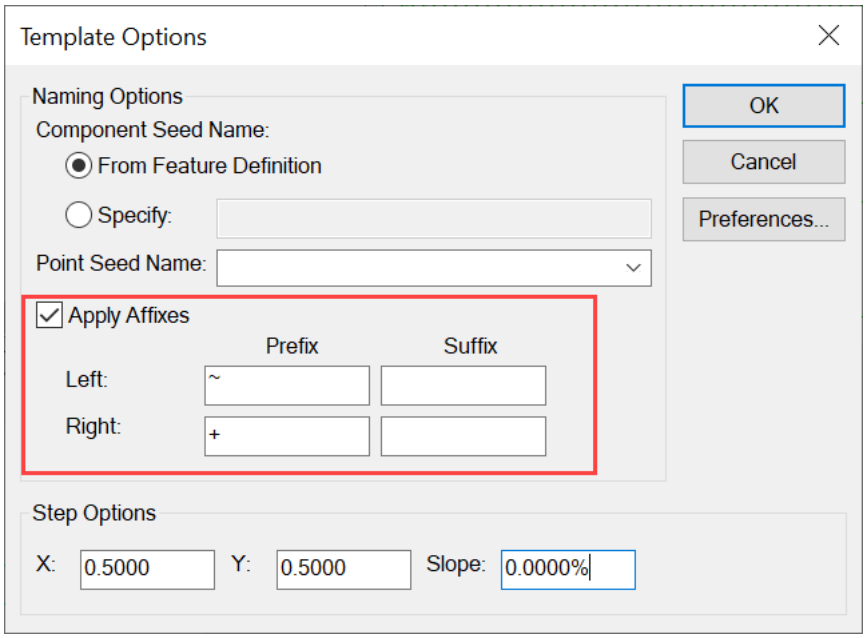
Exercise C8: Pavement Compound Components

In this exercise we will cover the creation of a pavement compound component and the mechanics of how pavement wedging works. Lessons learned in this exercise include the Components ITL, the between Simple and Compound components (both types exist in the ITL), Apply Affixes, Horizontal Feature Constraints and Moving and Merging of points.

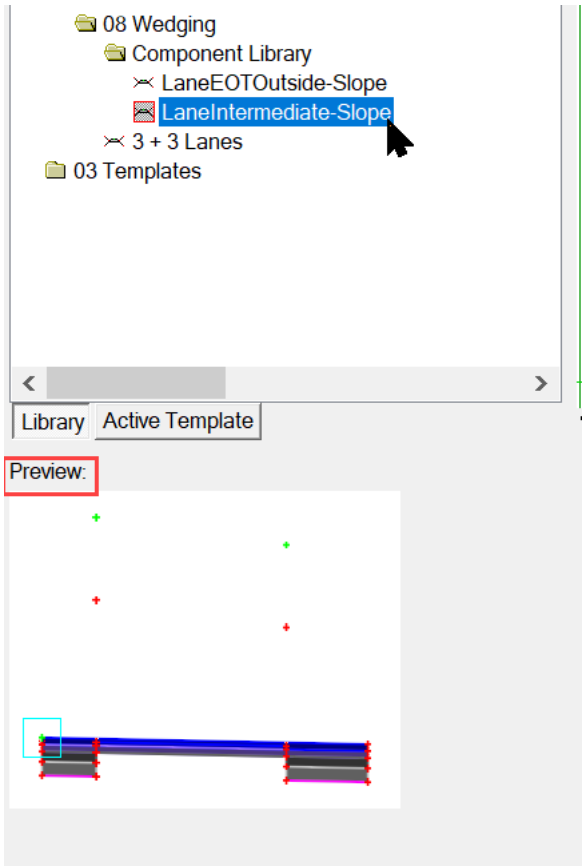
1. In the **Template Library** Open the **02 Components\08 Wedging** folder and create a new template named **3 + 3 Lanes**.



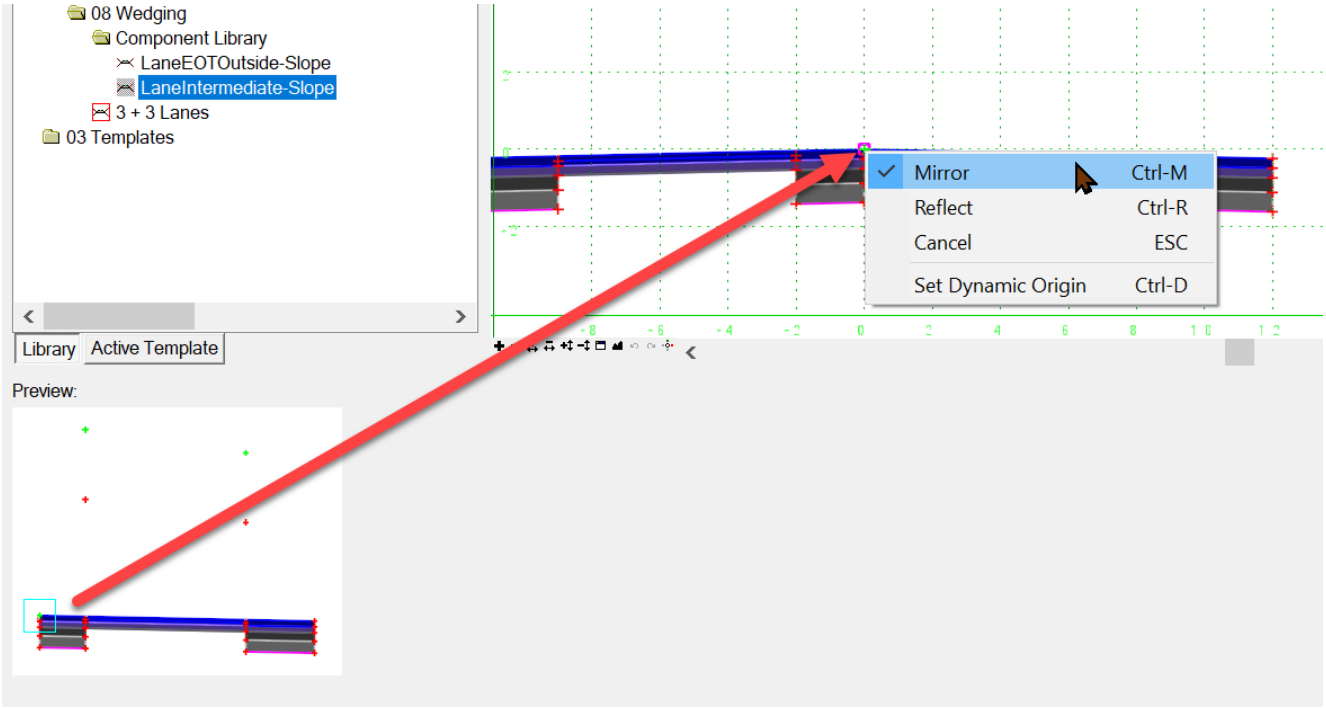
2. Under the toolbox menu **Tools >>> Options** turn on **Apply Affixes**.



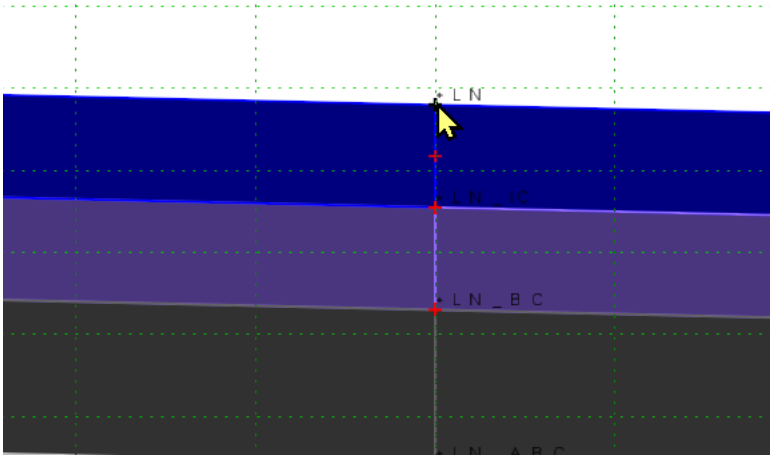
3. Open the **Component Library** folder and select (single left mouse click) the **LaneIntermediate-Slope** simple component. This will display the selected component in the **Preview** screen in the lower left corner.



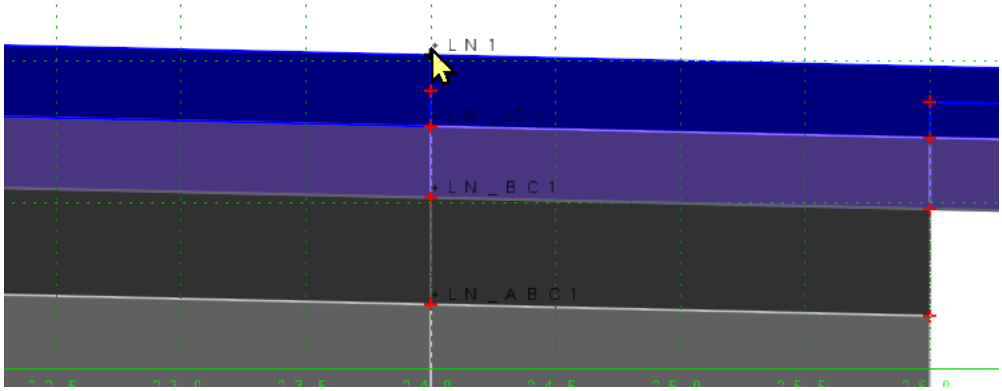
4. Drag the component from the **Preview** screen to the active template screen by selecting the light blue box (anchor point) in the **Preview** screen and while holding down the left mouse button, right mouse click to choose **Mirror.** Place the component at the template origin (purple box).



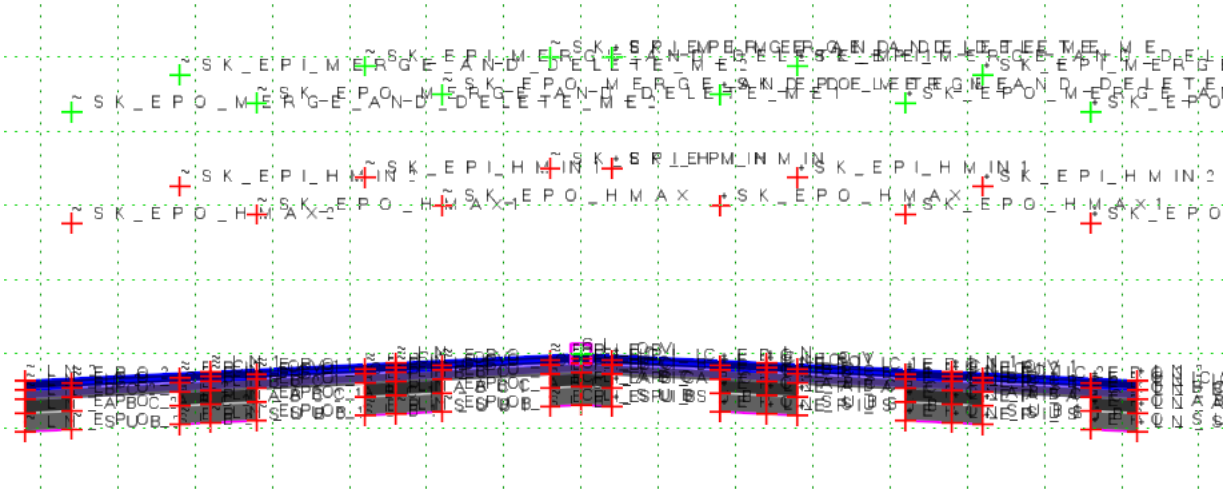
5. Select the **LaneIntermediate-Slope** simple component again in the **Component Library** folder and drag it to the active template screen. **Mirror** should still be checked on. Place it at the **+LN** point.



6. Select the **LaneEOTOutside-Slope** simple component in the **Component Library** folder and drag it to the active template screen. **Mirror** should still be checked on. Place it at the **+LN1** point.

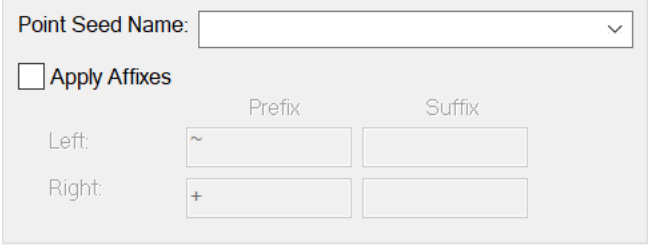


The **3 + 3 Lanes** compound component should look like the picture below.

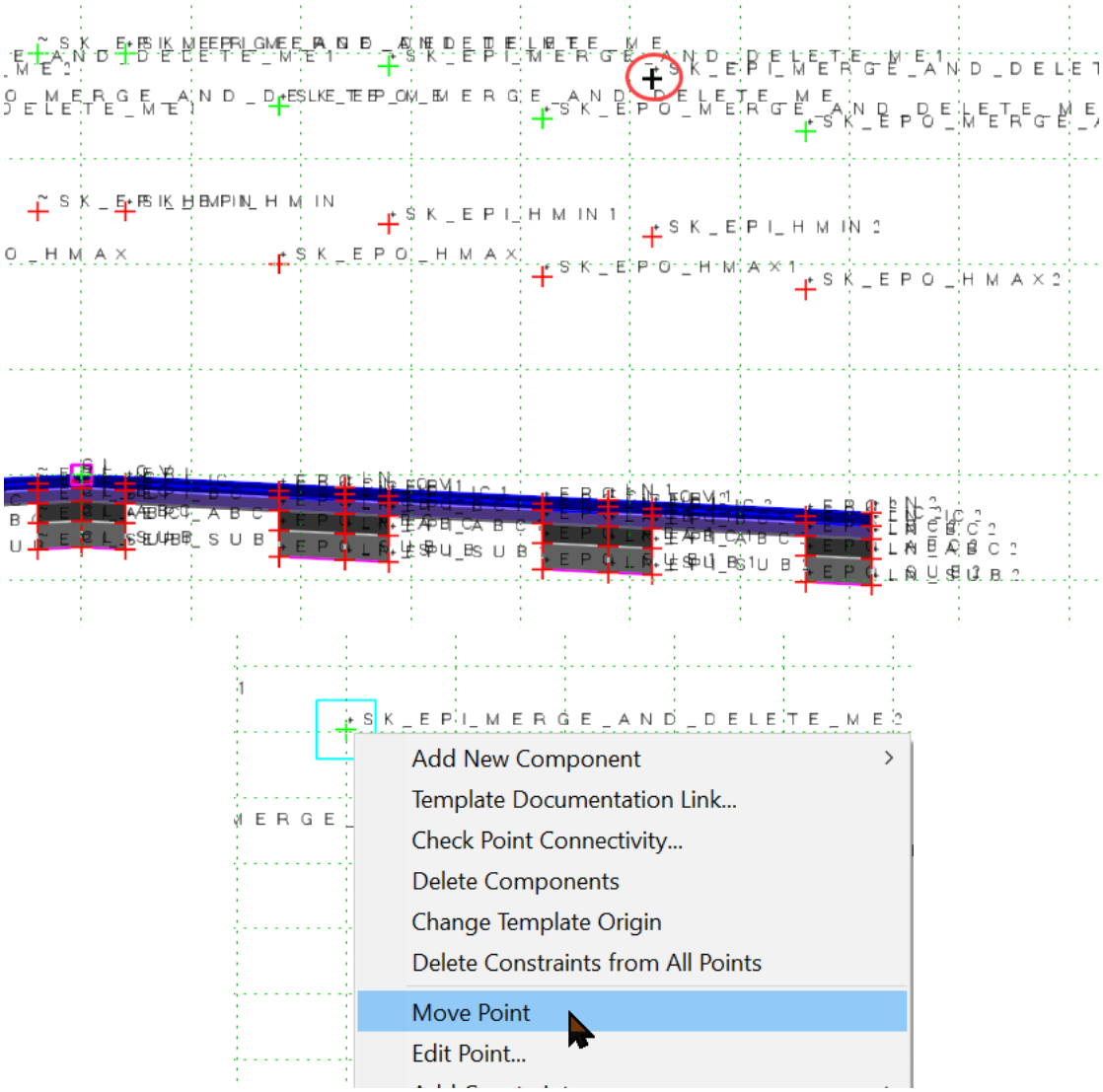


Next, we will need to program how the pavement wedging mechanics should work.

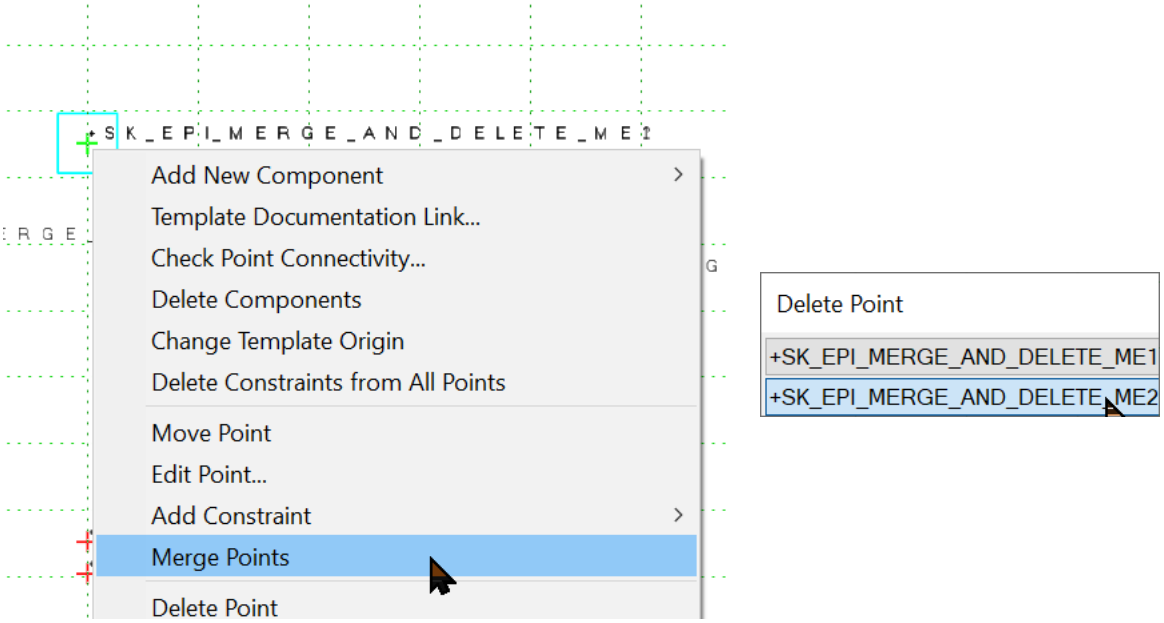
7. Under the toolbox menu **Tools >>> Options** turn off **Apply Affixes.**



8. Starting the right **+SK_EPI_MERGE_AND_DELETE_ME2** point, **Move** (right mouse click) it on top of the **+SK_EPI_MERGE_AND_DELETE_ME1** to the left of it.

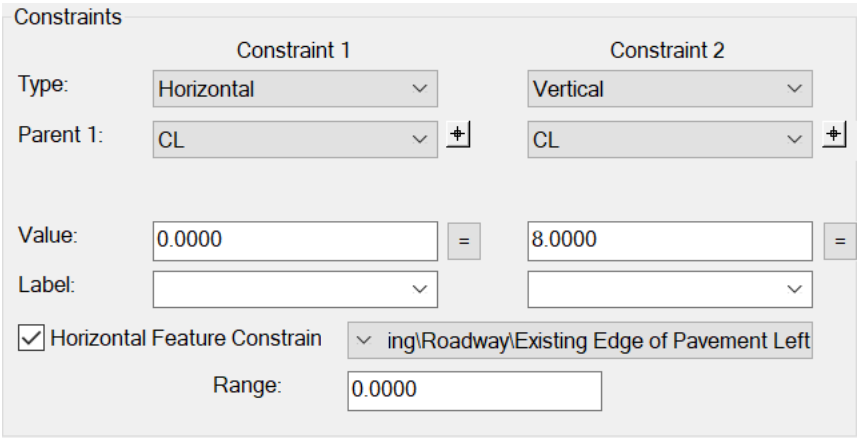


9. **Merge** (right mouse click) **+SK_EPI_MERGE_AND_DELETE_ME2** and **+SK_EPI_MERGE_AND_DELETE_ME1**. When prompted to delete which point when they are merged, select the **+SK_EPI_MERGE_AND_DELETE_ME2** point.



10. Repeat the above steps to Move and **Merge** the **+SK_EPI_MERGE_AND_DELETE_ME1** point with the **+SK_EPI_MERGE_AND_DELETE_ME** point. Select **+SK_EPI_MERGE_AND_DELETE_ME1** as the point to be deleted during the merging process.

11. Edit the **+SK_EPI_MERGE_AND_DELETE_ME** point and rename it **SK_EP_L**. Constrain it as shown below. The feature definition for the Horizontal Feature Constraint is "**Linear\Roadway\Existing\Roadway\Existing Edge of Pavement Left**".



12. On the left side, Move and **Merge ~SK_EPO_MERGE_AND_DELETE_ME2** point with the **~SK_EPO_MERGE_AND_DELETE_ME1** point. The **~SK_EPO_MERGE_AND_DELETE_ME2** point is deleted during the merging process.

13. Repeat the the step to **Move** and **Merge** the **~SK_EPO_MERGE_AND_DELETE_ME1** point with the **~SK_EPO_MERGE_AND_DELETE_ME** point. The **~SK_EPO_MERGE_AND_DELETE_ME1** point is deleted during the merging process.

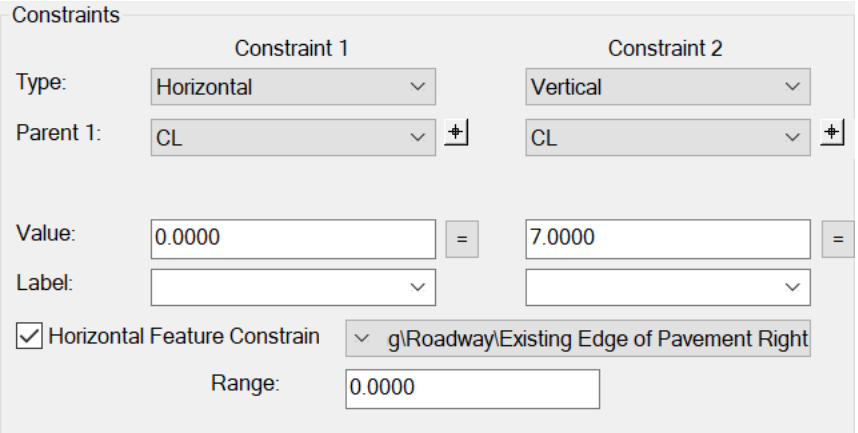
14. **Move** and **Merge** the **~SK_EPO_MERGE_AND_DELETE_ME2** point with the **SK_EP_L** point. The **~SK_EPO_MERGE_AND_DELETE_ME** point is deleted during the merging process.

This completes the existing left edge of pavement seek points (horizontal feature constrained). Next work on the existing right edge of pavement seek points.

15. On the right side, **Move** and **Merge** the **+SK_EPO_MERGE_AND_DELETE_ME2** point with the **+SK_EPO_MERGE_AND_DELETE_ME1** point. The **+SK_EPO_MERGE_AND_DELETE_ME2** point is deleted during the merging process.

16. **Move** and **Merge** the **+SK_EPO_MERGE_AND_DELETE_ME1** point with the **+SK_EPO_MERGE_AND_DELETE_ME** point. The **+SK_EPO_MERGE_AND_DELETE_ME1** point is deleted during the merging process.

17. Edit the **+SK_EPO_MERGE_AND_DELETE_ME** point and rename it **SK_EP_R**. Constrain it as shown below. The feature definition for the Horizontal Feature Constraint is **“Linear\Roadway\Existing\Roadway\Existing Edge of Pavement Right”.**

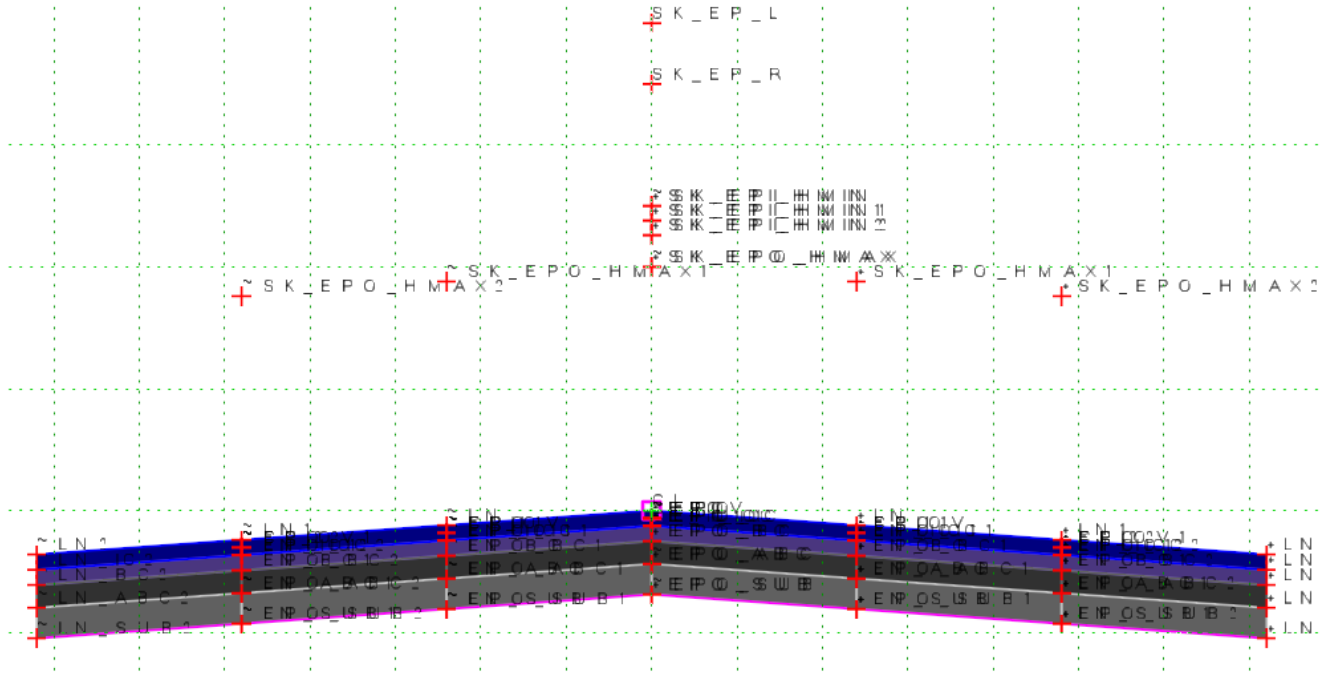


18. On the left side, **Move** and **Merge** the **~SK_EPI_MERGE_AND_DELETE_ME2** point with the **~SK_EPI_MERGE_AND_DELETE_ME1** point. The **~SK_EPI_MERGE_AND_DELETE_ME2** point is deleted during the merging process.

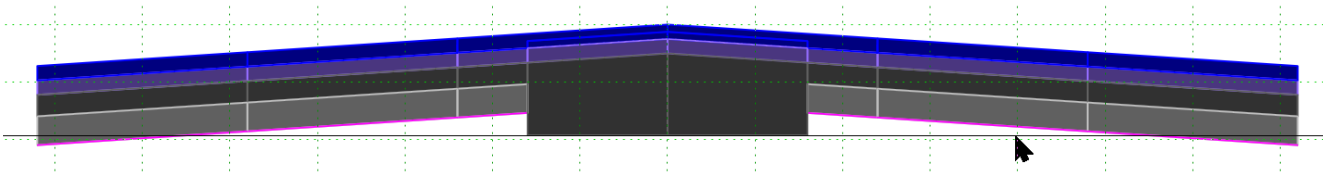
19. **Move** and **Merge** the **~SK_EPI_MERGE_AND_DELETE_ME1** point with the **~SK_EPI_MERGE_AND_DELETE_ME1** point. The **~SK_EPI_MERGE_AND_DELETE_ME1** point is deleted during the merging process.

20. **Move** and **Merge** the **~SK_EPI_MERGE_AND_DELETE_ME** point with the **SK_EP_R** point. The **~SK_EPI_MERGE_AND_DELETE_ME** point is deleted during the merging process.

The completed 3 + 3 Lanes compound component should look like the picture below.



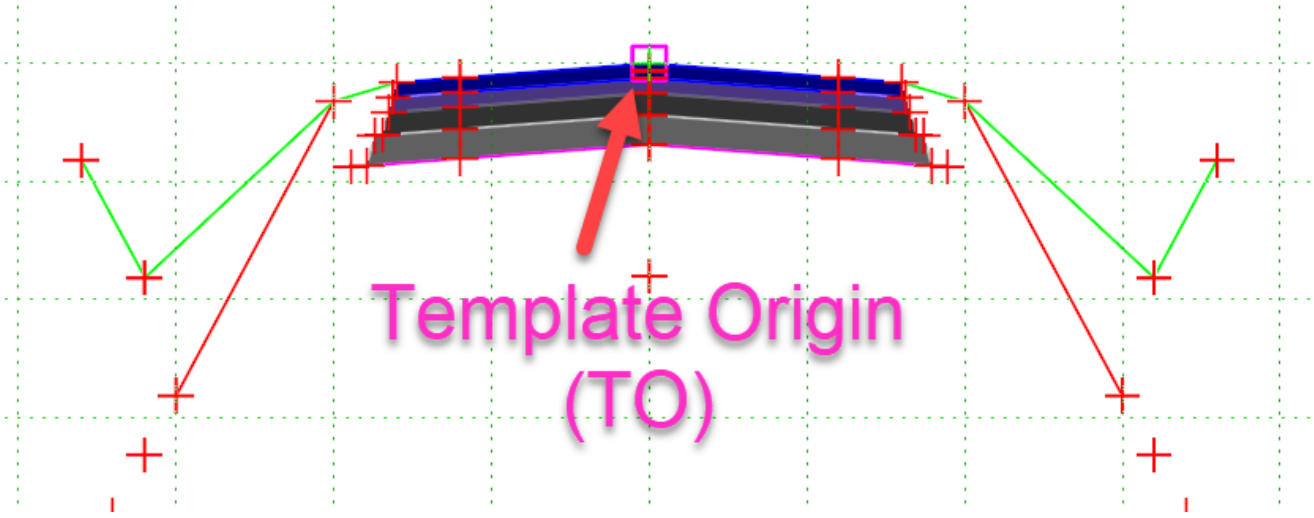
21. **Test** the template by changing the horizontal constraint values for the **SK_EP_L** and **SK_EP_R** points.



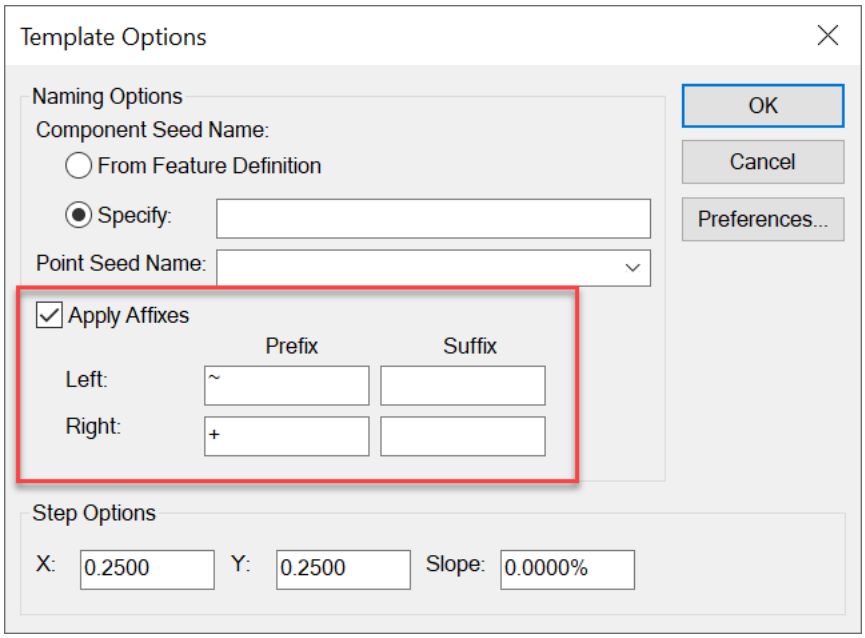
Templates

Templates are usually created by assembling the components together. Components are stored in the **NCDOT_RDY_Components.itl** in the WorkSpace. Importance is stressed in the order t**he** assembly of the components. They were designed with the **POMM** method in mind. Start with the **Pavement. T**hen work toward the **Outside** (curb and gutter/berm or shoulder and the end conditions). Then **Mirror** the Right Side Only **(RSO)** template to get both sides**.** Lastly finish it off with the **Median** components if required.

The **Template Origin** (TO) at the (0,0) mark indicated by a purple box is very important and often overlooked. When used in Corridor Modeling, the TO is where the horizontal (X,Y) and vertical (Z) alignments of the corridor are located. The point at the TO is usually unconstrained (green). This eliminates recursive errors when other points are constrained to it. All other points in the template are directly or indirectly constrained (red) to the TO point. Although a point does not have to be at the TO, most of the time it is occupied by a point such as the **CL** point.



When mirroring the **RSO** template, it is critical to turn on **Apply Affixes**. This is what makes the left side point and component names different from the right side. **Applied Affixes** can be accessed through the menu **Tools >>> Options**.



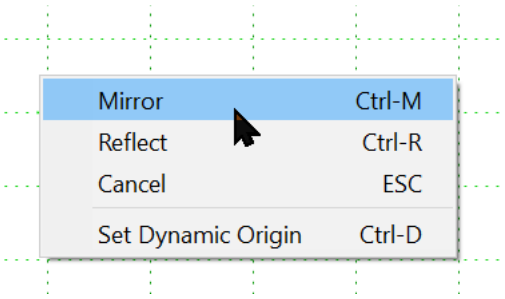
The standard affixes for Roadway Design templates are a tilde (**~**) for the left side **Prefix** and a plus sign (**+**) for the right side **Prefix**. When instructed to "Turn on Apply Affixes" in the manual, the check box must be enabled and Prefixes filled in.

Apply Affixes should be turned off most of the time, especially when building components, RSOs and when adding the median components to the templates.

The **Step Options** is the snap distance of cursor to the active template grid. This is useful when placing a point of a component at template origin. A Step options value of 0.25 or 0.5 for both X and Y is recommended.

When dragging a component over to the active screen, the option to **Mirror** or **Reflect** is available. User may right mouse click or in the bottom right corner check on or off **Mirror** or **Reflect** before placement.

Right Mouse Click

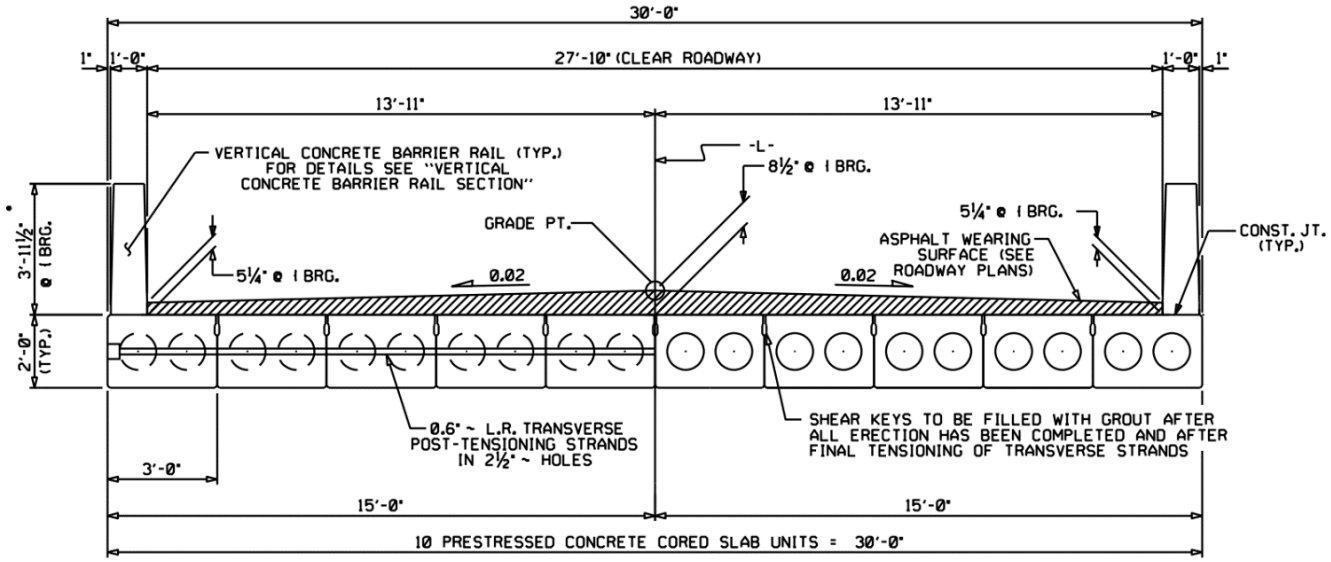


Lower Bottom Right Corner

Each exercise in this training contains a **Component Library** folder in the training ITL simulating the components in the **NCDOT_RDY_Components.itl** found in the WorkSpace.

Exercise T1: Cored Slab Bridge Template

In this exercise we will go over the basics of assembling components to create a **Cored Slab Bridge** template. Using the components created earlier in this manual and from the Components ITL, assemble them together to create a complete template. Lessons learn in this exercise include **Null Points, Components ITL, Move** and **Merge** points and the **Order** components are designed to be assembled for a complete template.

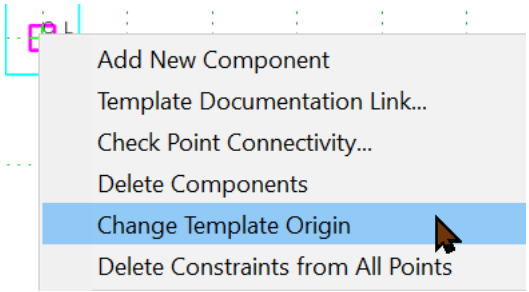


1. In the Template Library **Open** the **03 Templates\01 Cored Slab Bridge** folder and create a new template named **Cored Slab - RSO.**

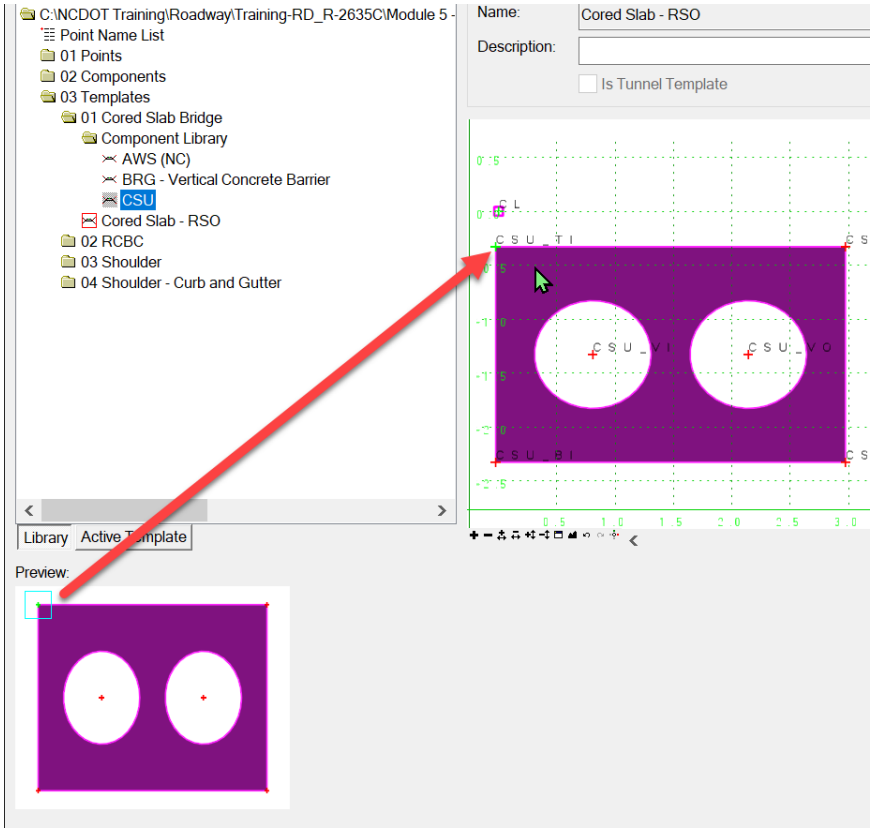
   RSO stands for Right Side Only. Templates are designed to be built with the right side only (RSO) first, then Mirror to complete both sides for a complete template.

2. Turn off **Apply Affixes**.

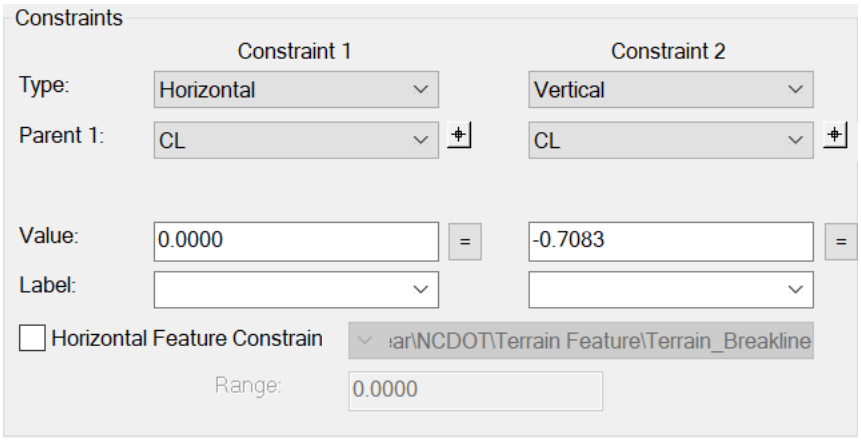
3. **Add New Component >>> Null Point** and rename it **CL**. Make this point the template origin.



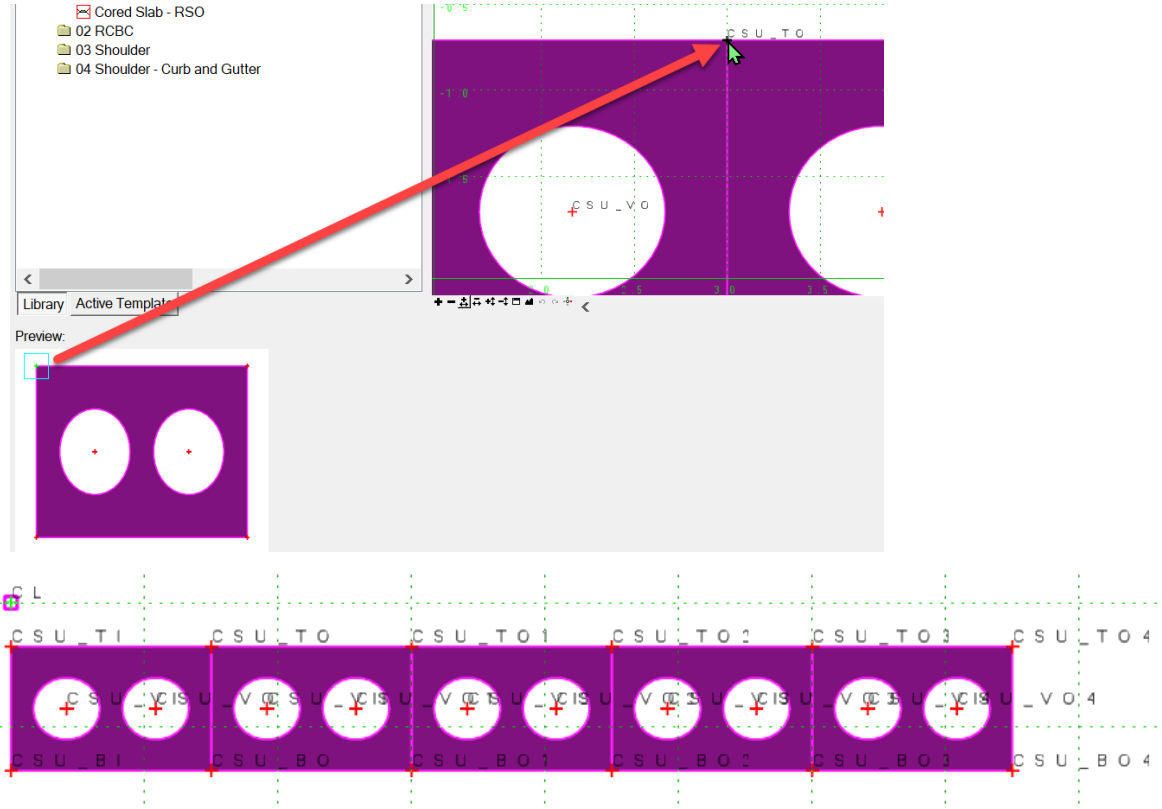
4. Open the **Component Library** folder, select (single left mouse click) and drag (hold down left mouse button) from the **Preview** screen the component **CSU** to the active template screen. Place the **CSU** component below the **CL** point.



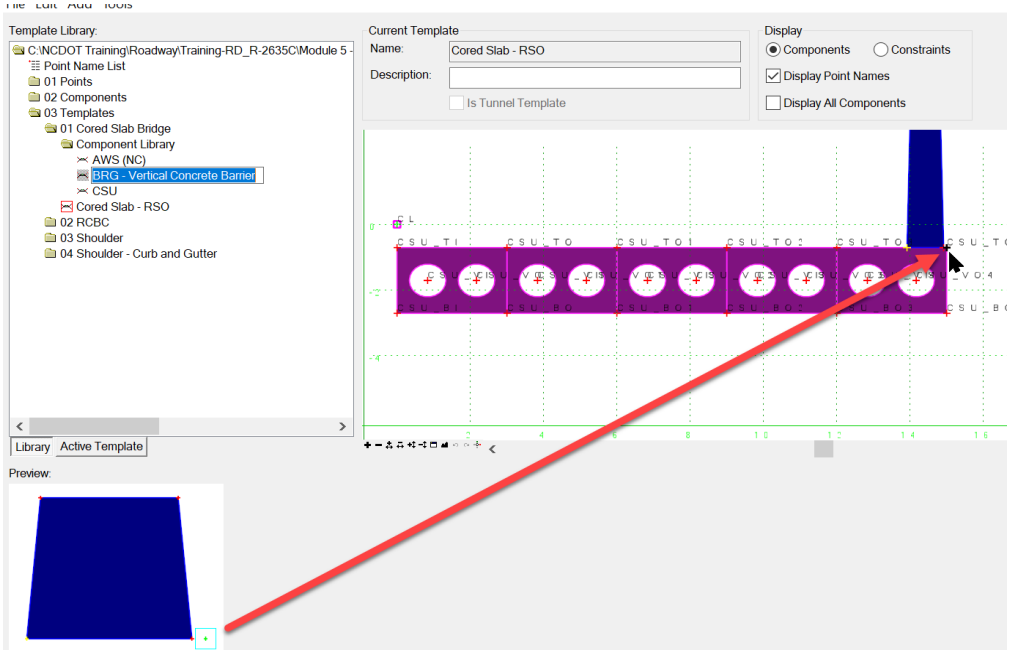
5. Constrain the **CSU_TI** point underneath the **CL** point as shown below.



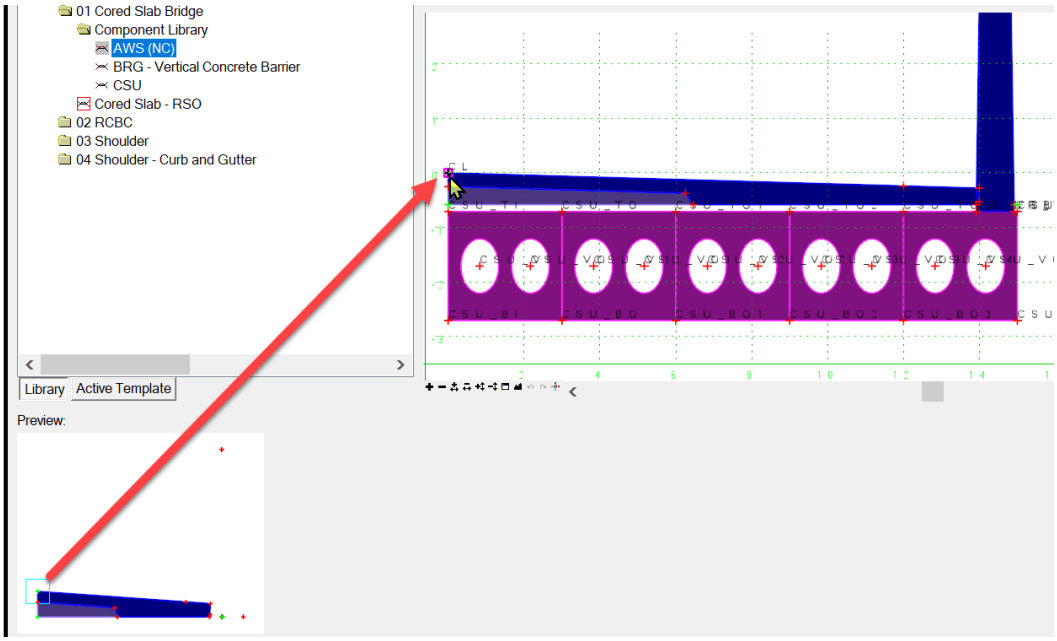
6. Drag four (4) more **CSU** components to the active template screen. Merge the **CSU** components at each of the **CSU_TO** point forming the superstructure.



7. Select the **BRG - Vertical Concrete Barrier** component in the **Components Library** folder and drag it over to the active template screen. Merge it at the **CSU_TO4** point.

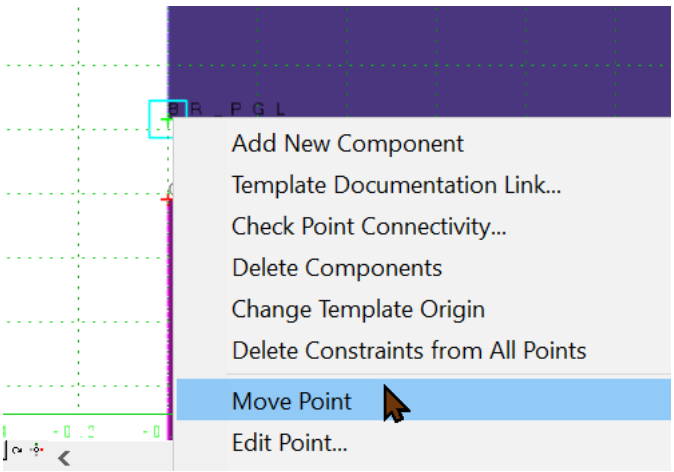


8. Select the **AWS (NC)** component in the **Components Library** folder and drag it over to the active template screen. Merge it at the **CL** point.

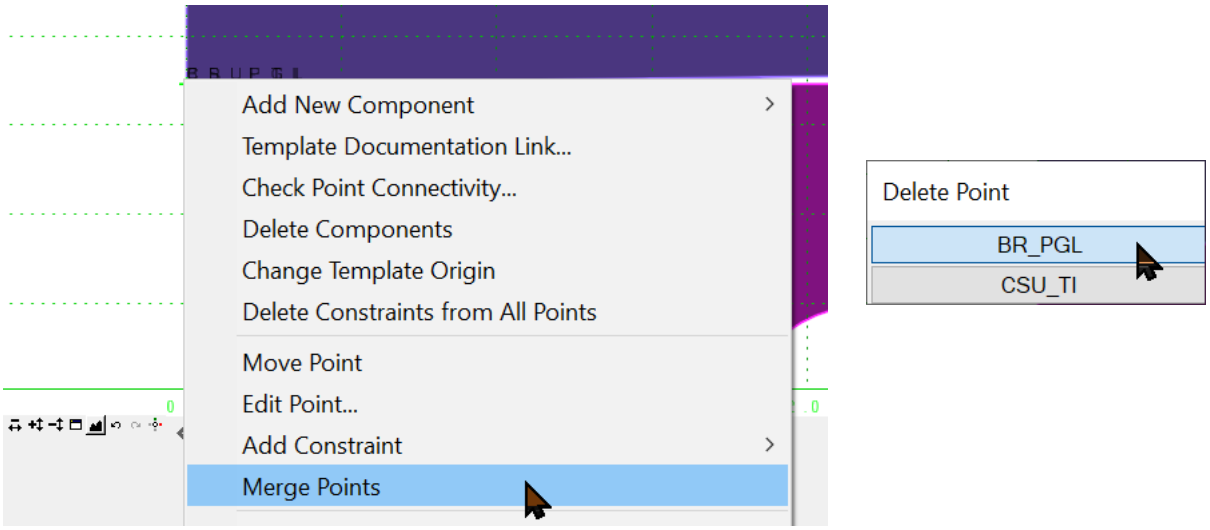


AWS (NC) stands for Asphalt Wearing Surface (Normal Crown).

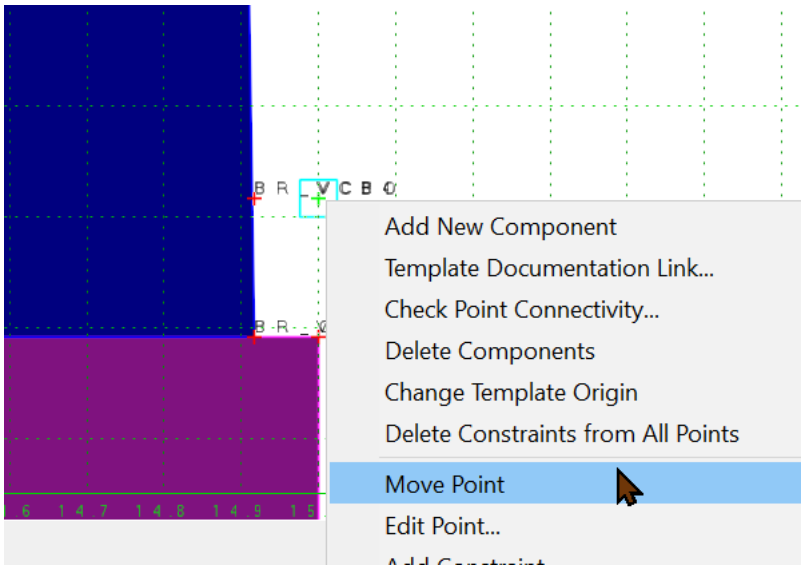
9. Move the **BR_PGL** point and place it on top of the **CSU_TI** point (below).



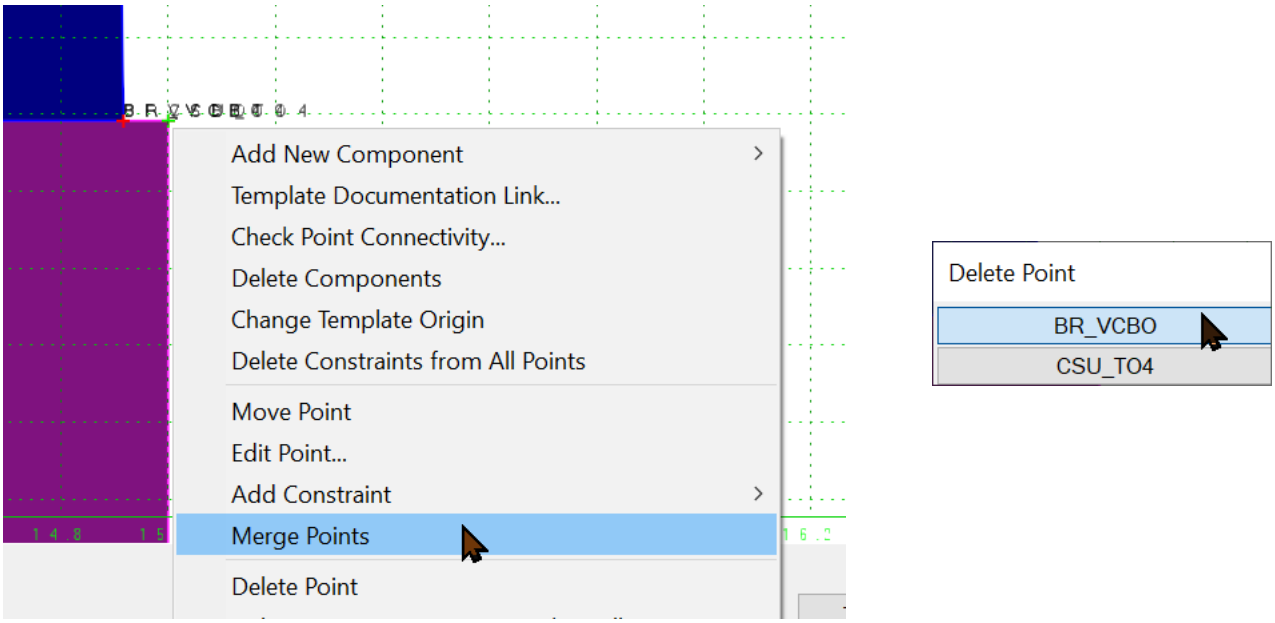
10. Merge the **BR_PGL** point with the **CSU_TI** point. When prompted to delete which point when they are merged, select the **BR_PGL** point to be deleted.



11. Move the **BR_VCBO** point and place it on top of the **CSU_TO4** point (below).



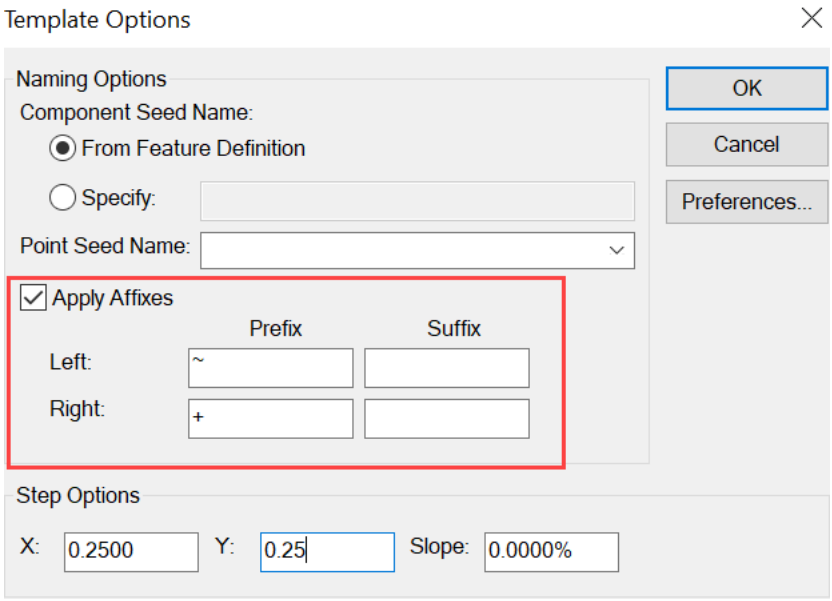
12. Merge the **BR_VCBO** point with the **CSU_TO4** point. When prompted to delete which point when they are merged, select the **BR_VCBO** point to be deleted.



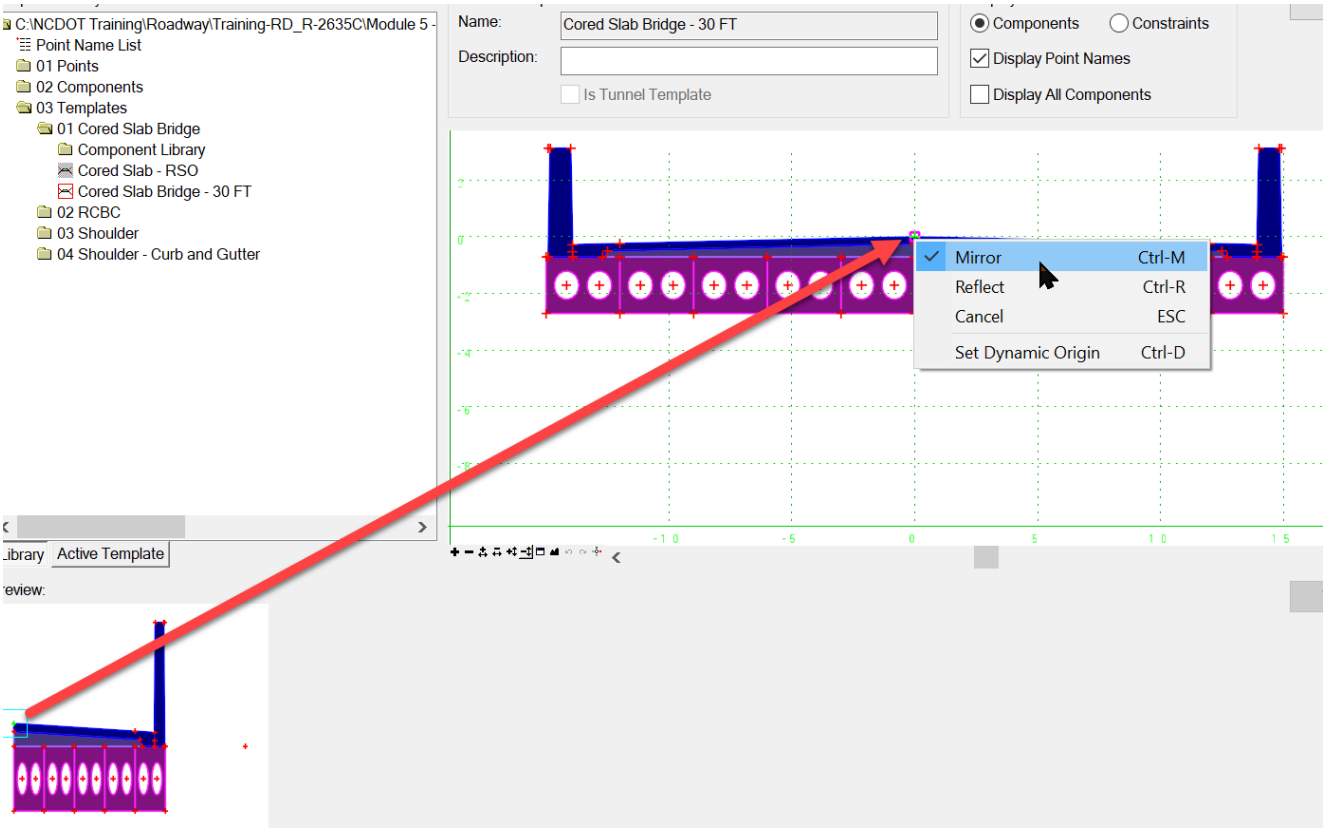
The **RSO** template is complete. The final step is to mirror the **RSO** and complete the template creation.

13. Create a new template and name it **Cored Slab Bridge – 30 FT**.

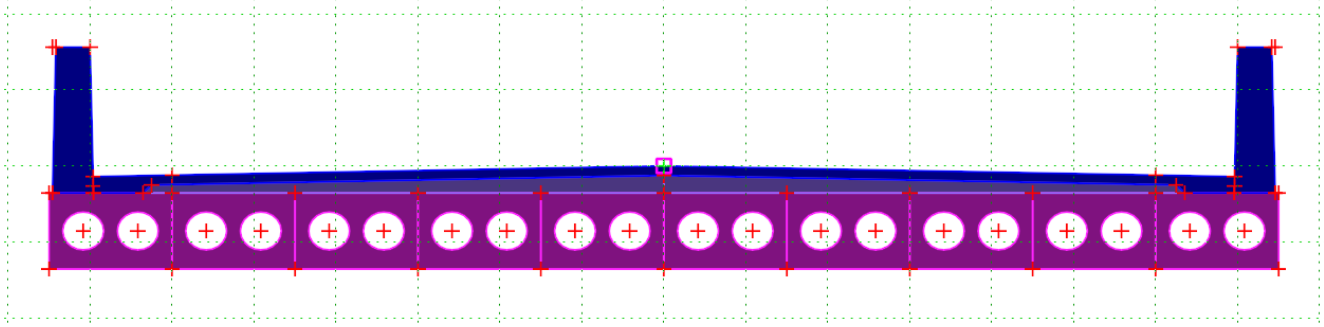
14. Turn on **Apply Affixes**



15. Select the **Cored Slab – RSO** template and drag it over to the active template screen. Check **Mirror** (right mouse click) and place it at the template origin.



The final Cored Slab Bridge template should look like the below.



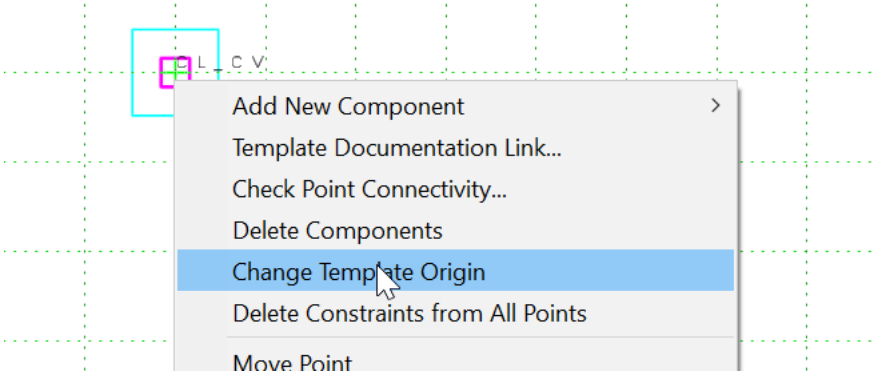
Exercise T2: Reinforced Concrete Box Culvert (RCBC) Template

In this exercise we will cover the basics of assembling components to create a Reinforced Concrete Box Culvert (**RCBC**) template. An odd number of barrels RCBC template is created differently from the even number of barrel RCBC template. Lessons learned in this exercise include **Null Points, Parent Component, Voids, Parametric Constraint Labels and Equations**.

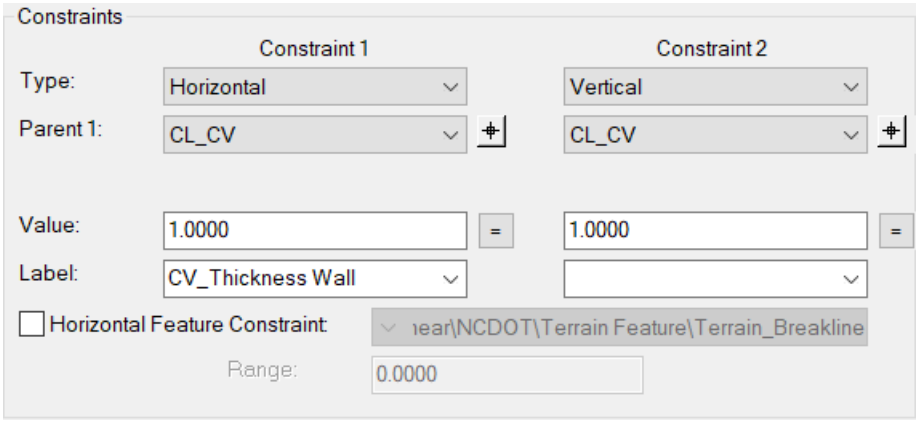
1. In the Template Library **Open** the **03 Templates\02 RCBC** folder and create a new template named **RCBC – Double Barrel**.

2. Turn off **Apply Affixes**.

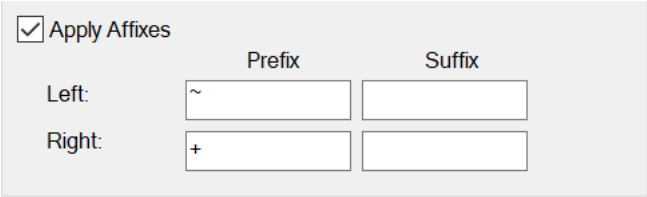
3. **Add New Component >>> Null Point** and name it **CL_CV**. Make this point the template origin.



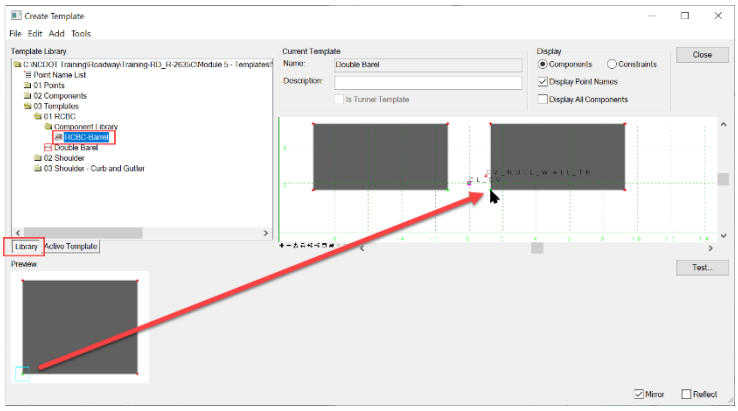
4. **Add New Component >>> Null Point** to the right of the CL_**CV** point and name it **CV_NULL_WALL_TH**. Constrain it as shown below.



5. Turn on **Apply Affixes.**

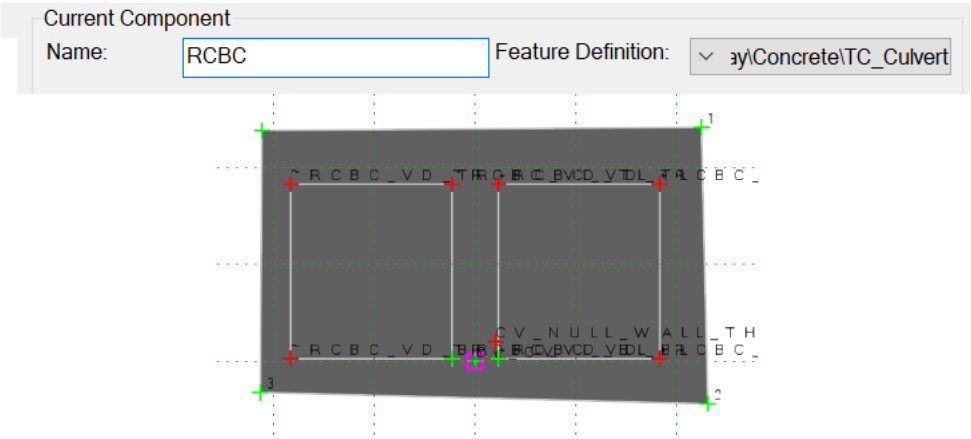


6. Open the **Component Library** folder, select (single left mouse click) and drag (hold down left mouse button) from the **Preview** screen the component **RCBC - Barrel** to the active template screen. Check on **Mirror** and place the barrel component to the right of the **CL_CV** point.

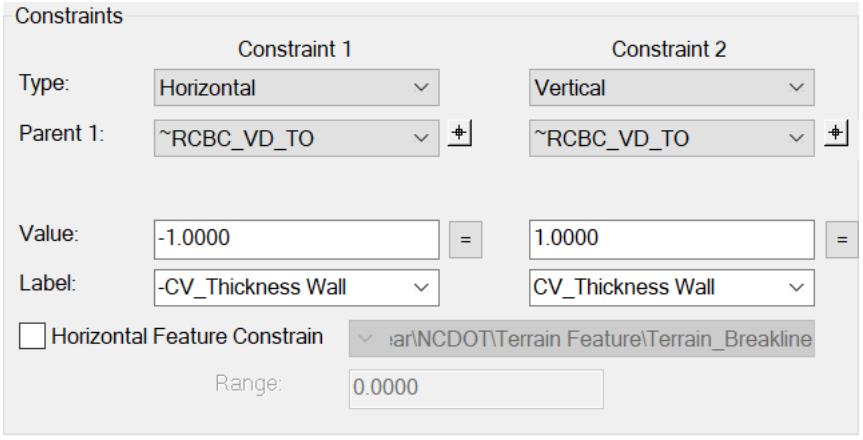


7. Turn off **Apply Affixes**.

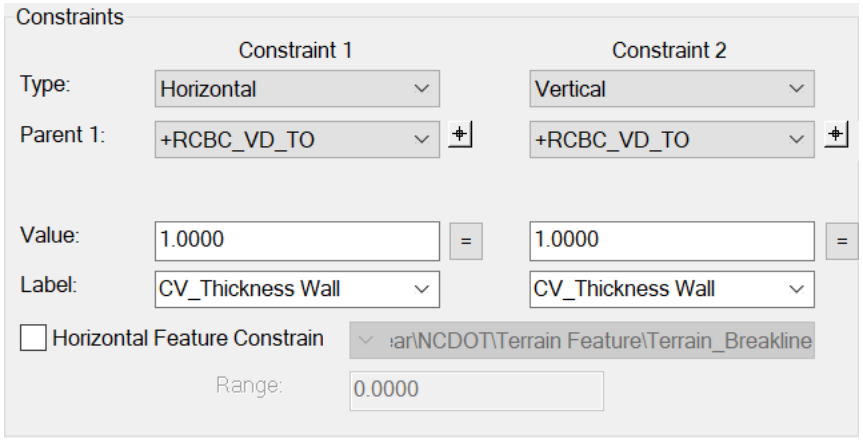
8. Create the outer shell of the culvert to encompass the barrels. Starting above and left of the **~RCBC_TO** point and working clockwise, **Add New Component >>> Unconstrained** to create a rectangle around the barrels. The culvert component should have these properties as shown below. The feature definition is **"Mesh\Roadway\Concrete\TC_Culvert".** Remember to turn off **Mirror.**



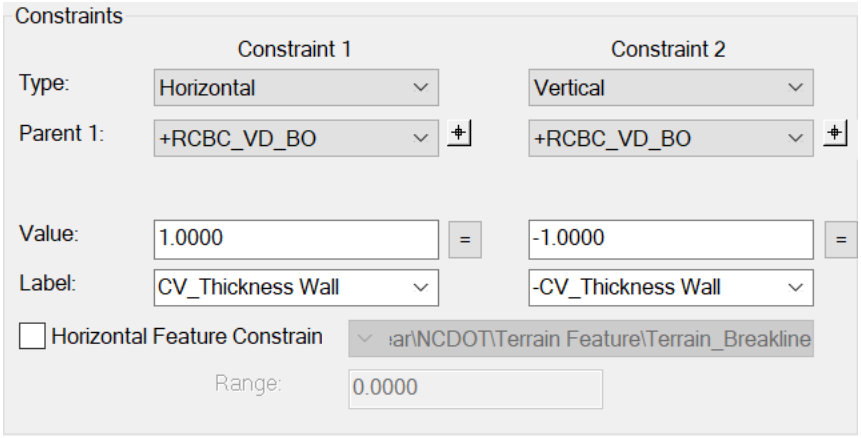
9. Starting from the top left corner, rename this point **RCBC_TL**. Constrain it as shown below.



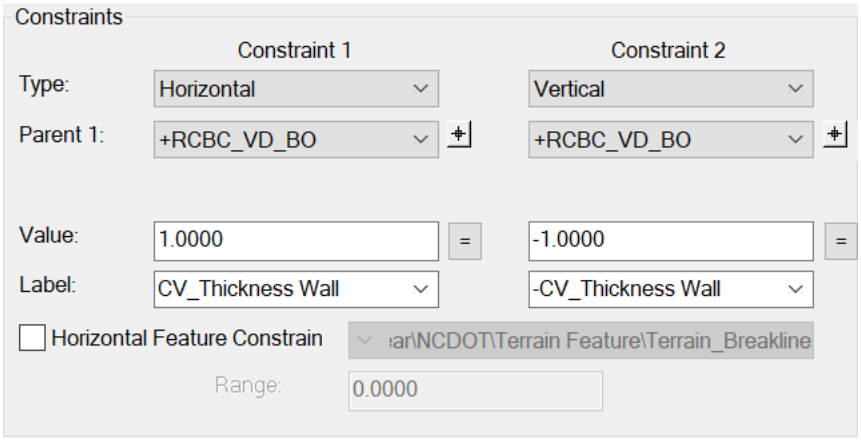
10. Edit the top right corner point and rename it **RCBC_TR**. Constrain it as shown below.



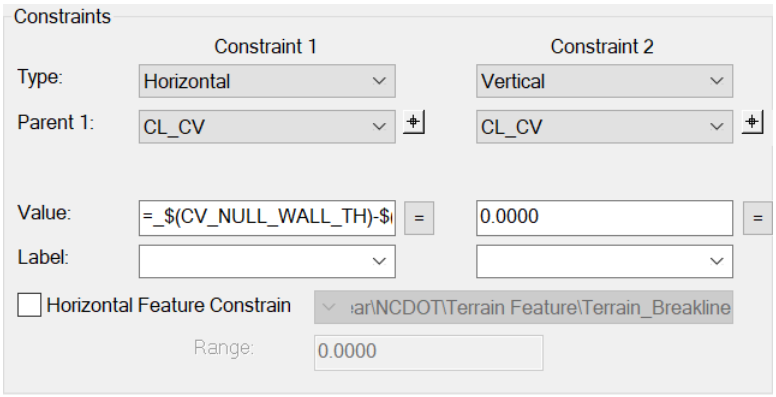
11. Edit the bottom right corner point and rename it **RCBC_BR**. Constrain it as shown below.

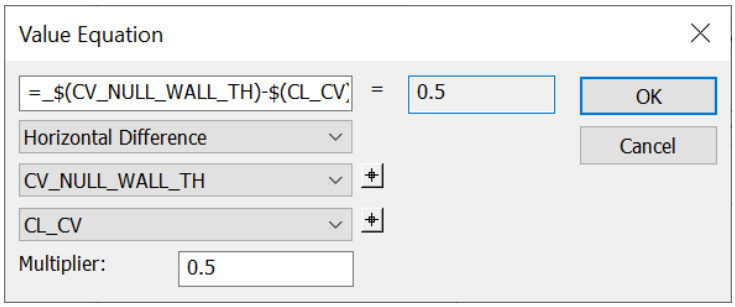


12. Edit the bottom left corner point and rename it **RCBC_BL**. Constrain it as shown below.

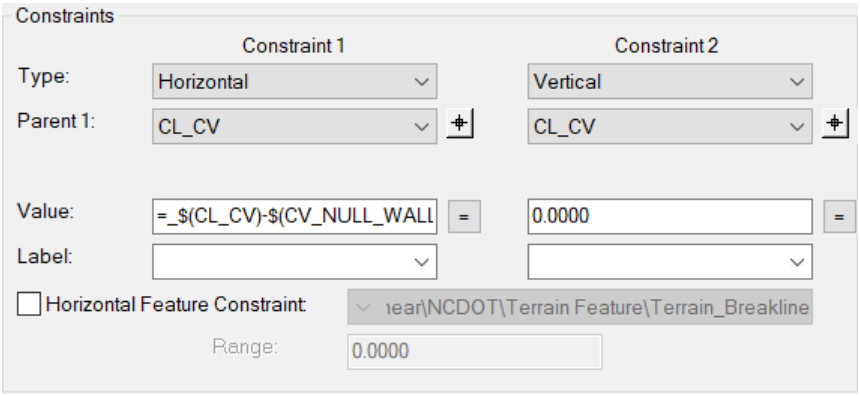


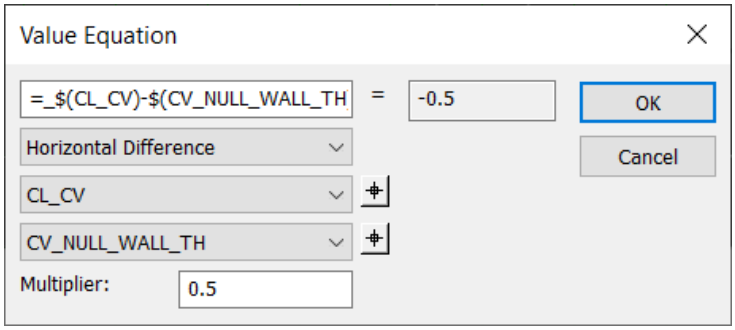
13. Constrain the **+RCBC_TI** point (right green) as shown below. Note how the horizontal equation is written.



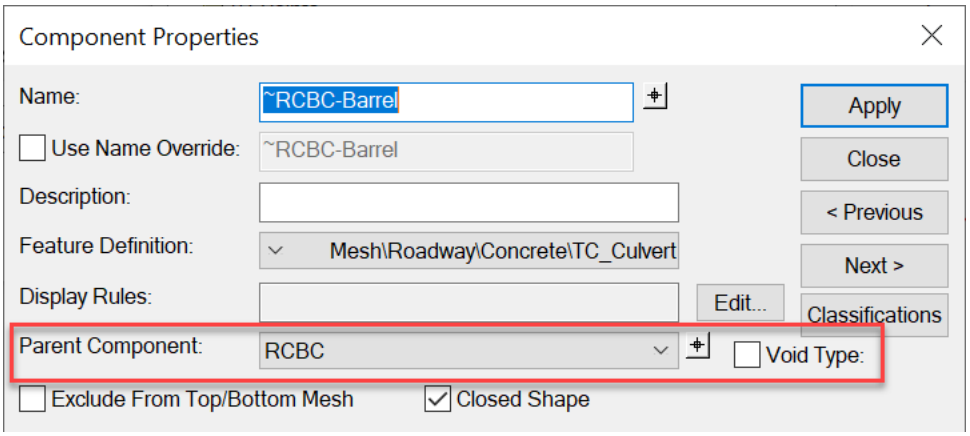


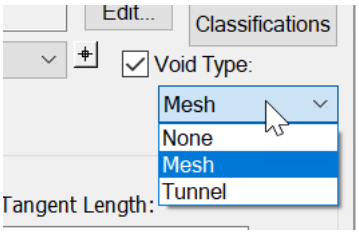
14. Constrain the **~RCBC_TI** point (left green) as shown below. Note how the horizontal equation is written.



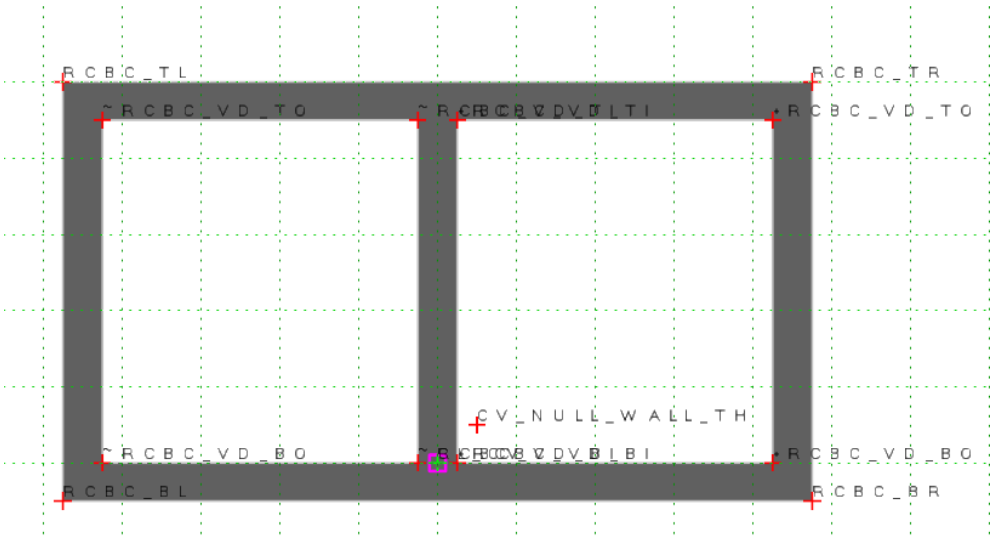


15. Make **RCBC** the **Parent Component** for both barrels. Note the **Void Type** option is enabled after having the encompassing Parent Component. Check this box and select **Mesh** as the Void Type.



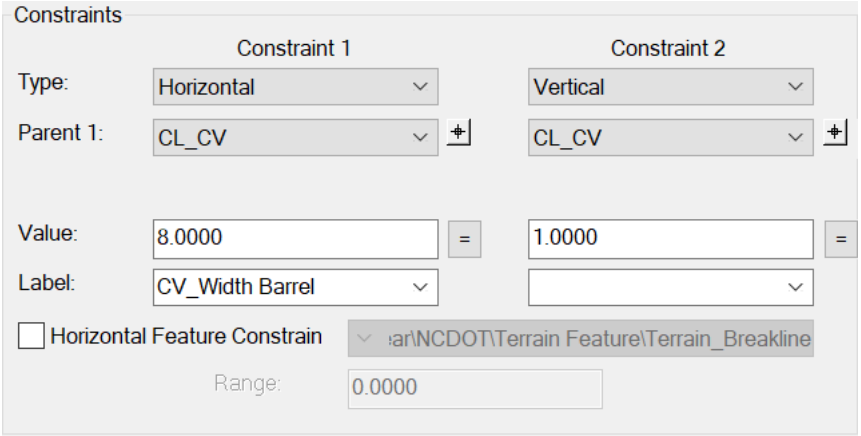


The **RCBC-Double Barrel** template should look like the picture below.

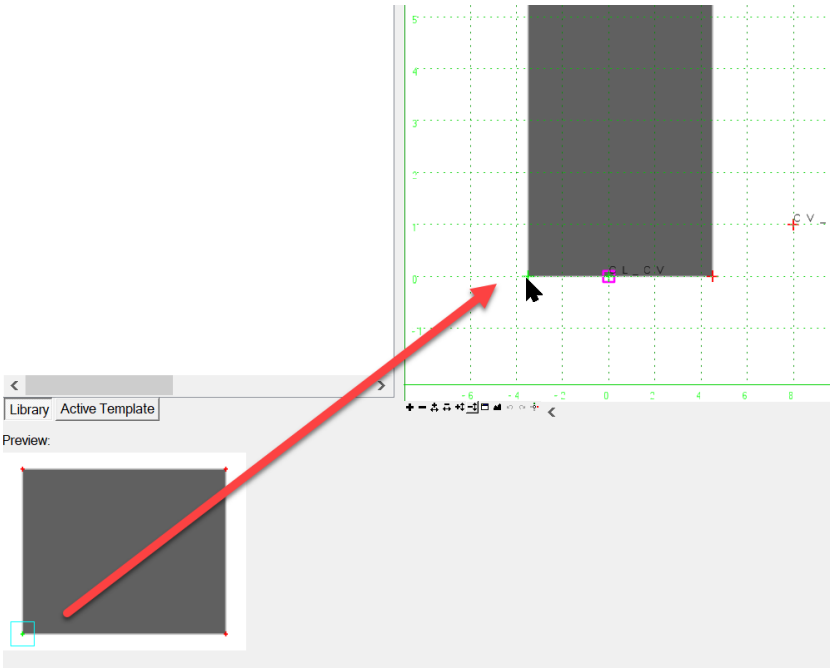


A triple barrel or odd number of barrel template is created slightly differently.

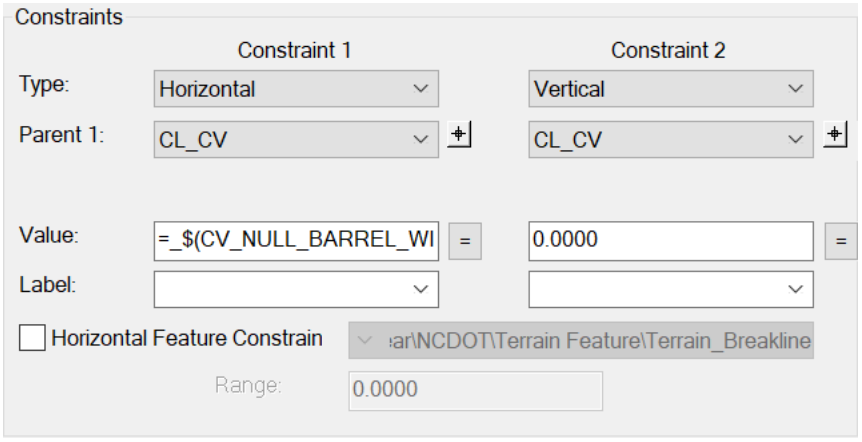
16. Create a new template and name it **RCBC – Triple Barrel.**
17. Turn off **Apply Affixes**.
18. **Add New Component >>> Null Point** and name it **CL_CV**. Make this point the template origin.
19. **Add New Component >>> Null Point** to the right of the CL_**CV** point and name it **CV_NULL_BARREL_WIDTH**. Constrain it as shown below.

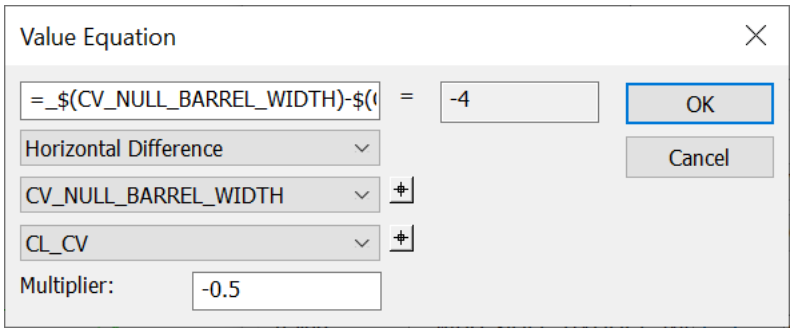


20. Select the component **RCBC - Barrel** from the **Components Library** folder and drag it to active template screen. Place it to the left of the **CL_CV** point.

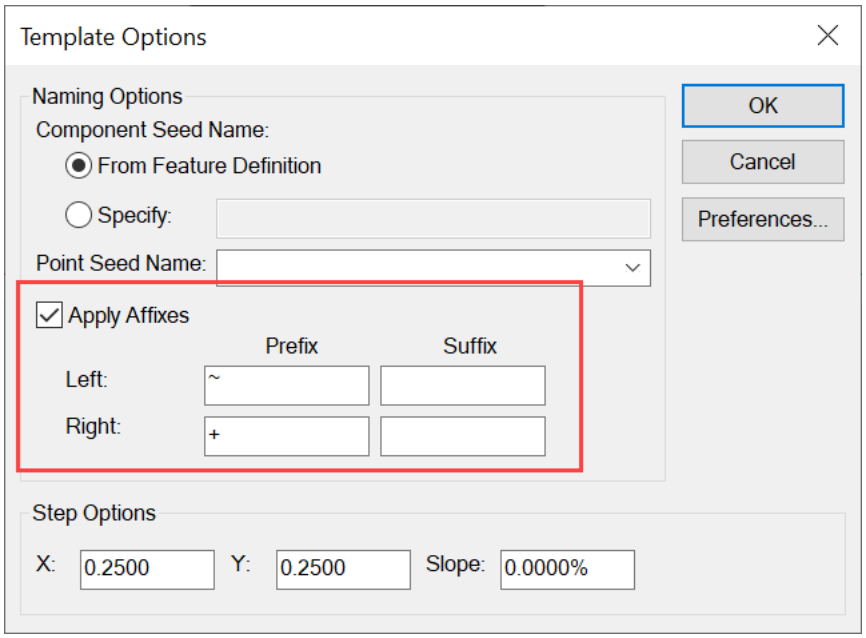


21. Constrain the green **RCBC_VD_BI** point as shown below. Note how the horizontal equation is written.

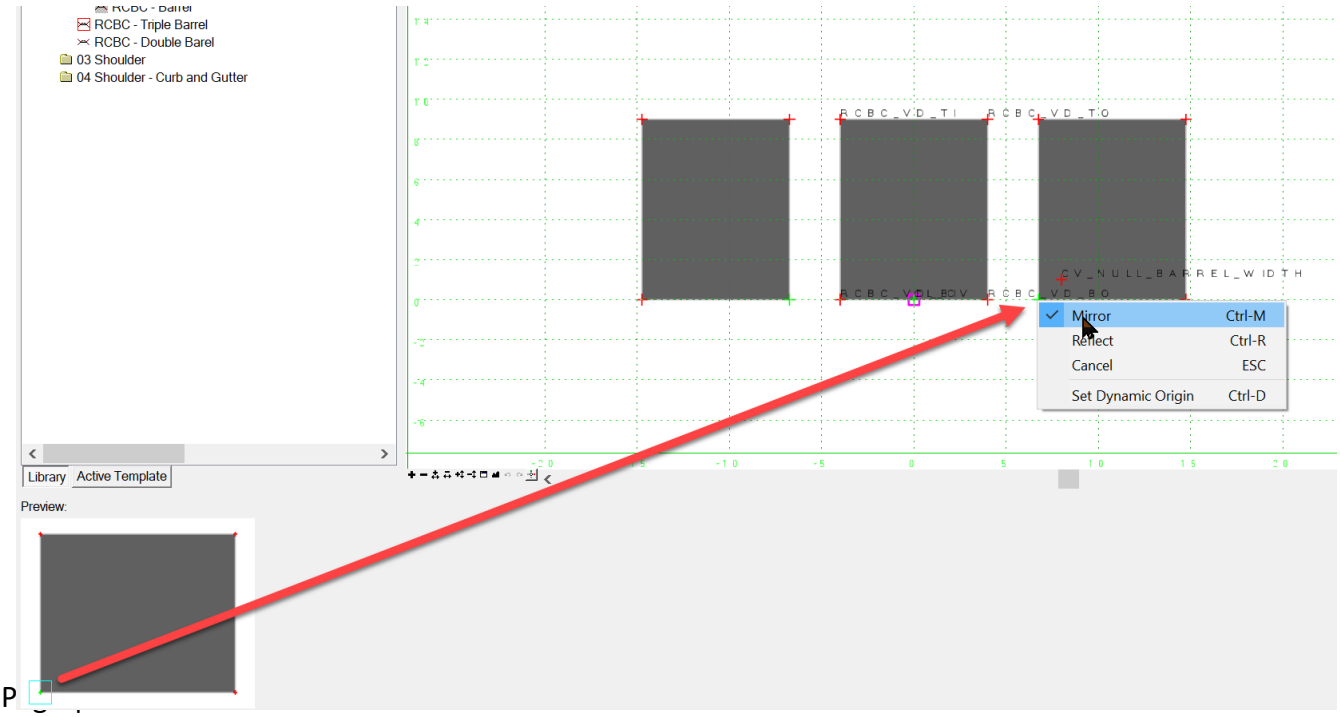




22. Under the toolbox menu **Tools >> Options** turn on **Apply Affixes**.

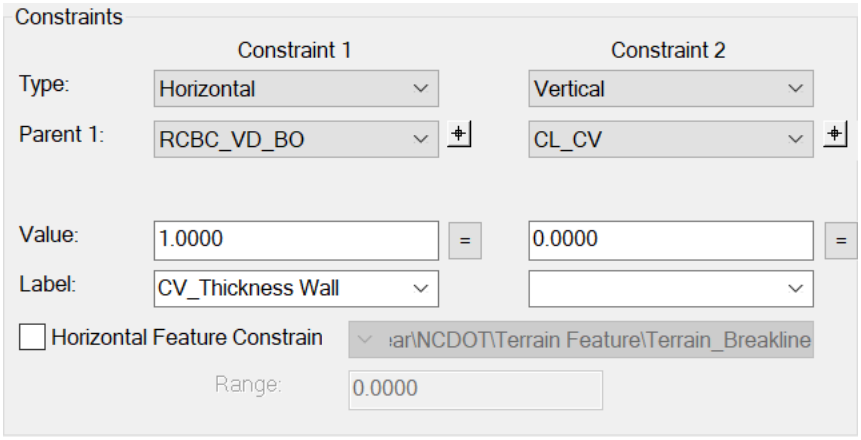


23. Select the component **RCBC - Barrel** from the **Components Library** folder and drag it to active template screen. Place it to the right of the center barrel. **Mirror** should be checked on (right mouse click).

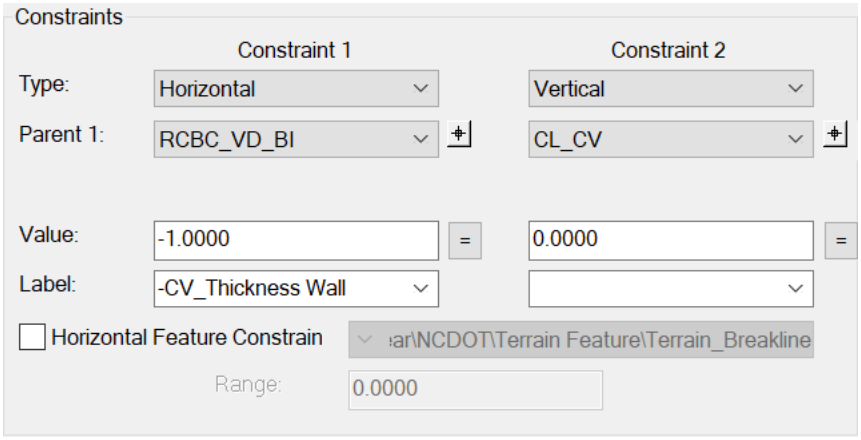


P

24. Constrain the **+RCBC_VD_BI** (right green) point as shown below.



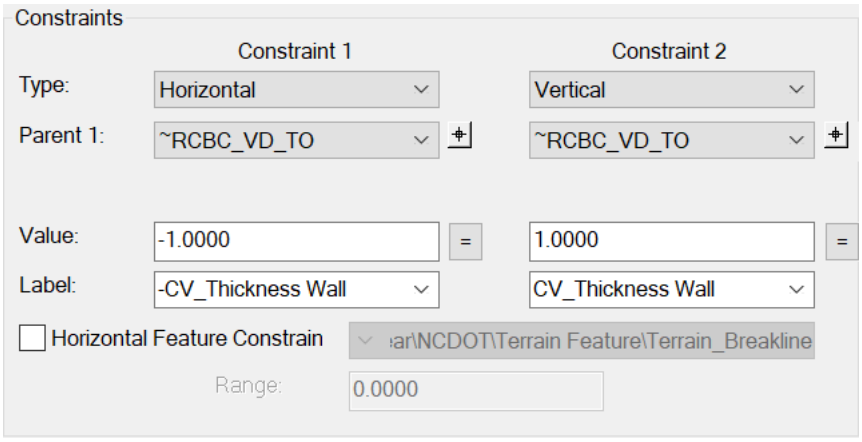
25. Constrain the **~RCBC_VD_BI** (left green) point as shown below.



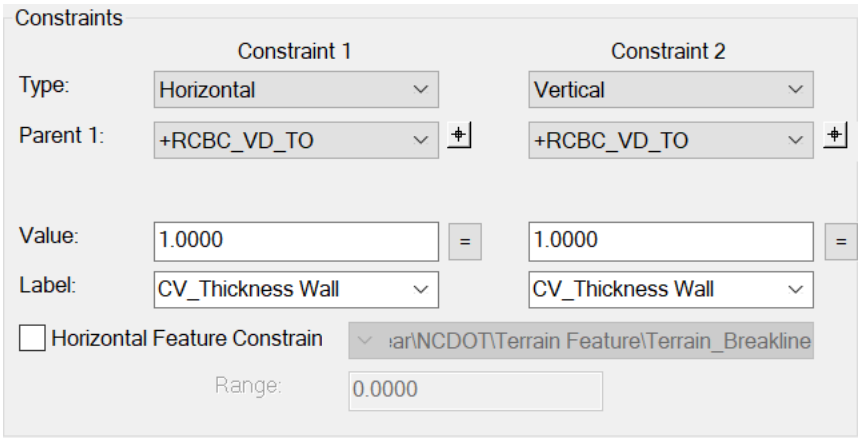
26. Turn off **Apply Affixes**.

27. Create the outer shell of the culvert to encompass the barrels. Starting above and left of the **~RCBC_TO** point and working clockwise, **Add New Component >>> Unconstrained** to create a rectangle around the barrels. The culvert component should have these properties as shown below. The feature definition is **"Mesh\Roadway\Concrete\TC_Culvert".** Remember to turn off **Mirror.**

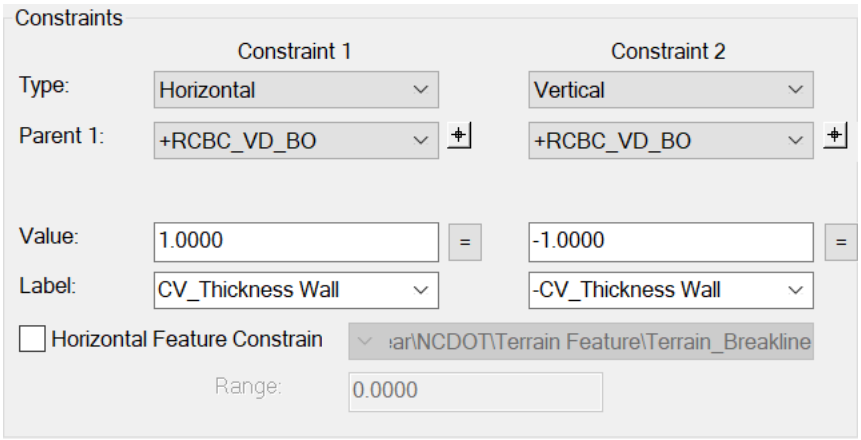
28. Starting from the top left corner, rename this point **RCBC_TL**. Constrain it as shown below.



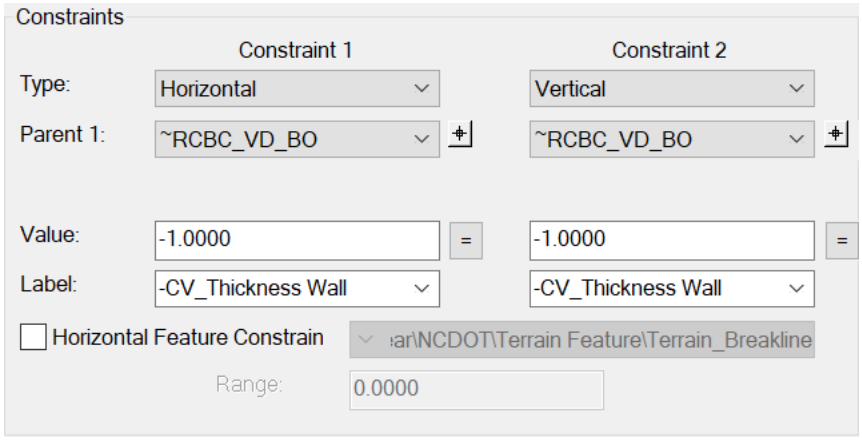
29. Edit point **1** and rename it **RCBC_TR**. Constrain it as shown below.

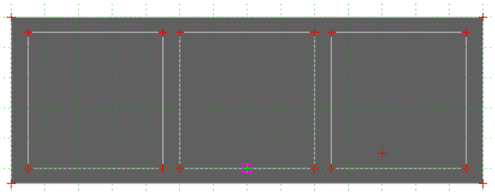


30. Edit point **2** and rename it **RCBC_BR**. Constrain it as shown below.

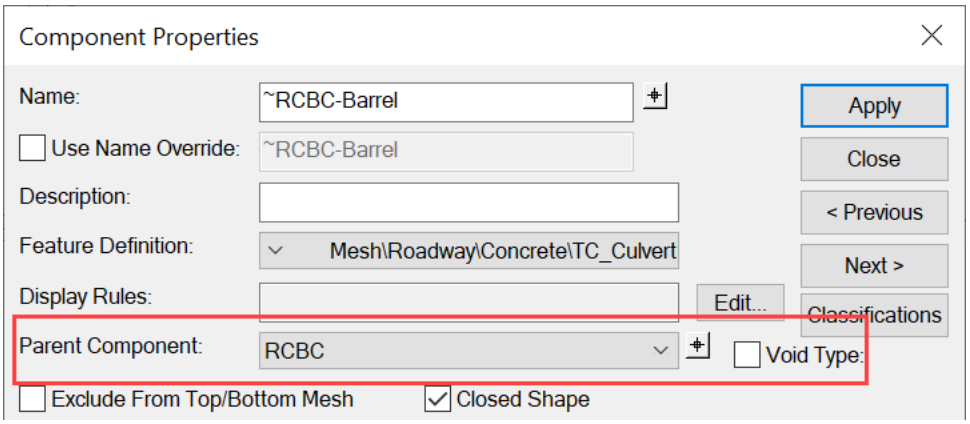


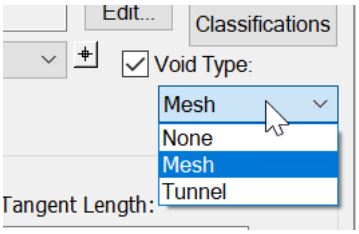
31. Edit point **1** and rename it **RCBC_BL**. Constrain it as shown below.



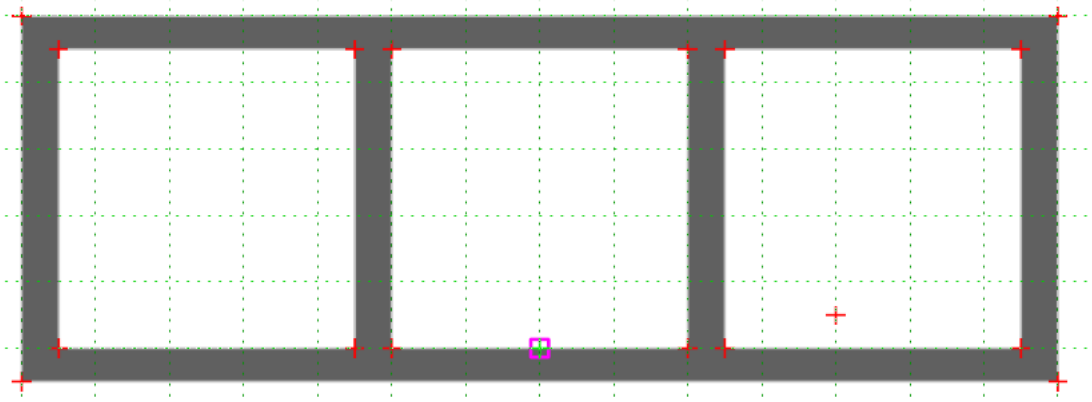


32. Make **RCBC** the **Parent Component** for all three (3) barrels. Note the **Void Type** option is enabled after having the encompassing **Parent Component**. Check this box and select **Mesh** as the Void Type.

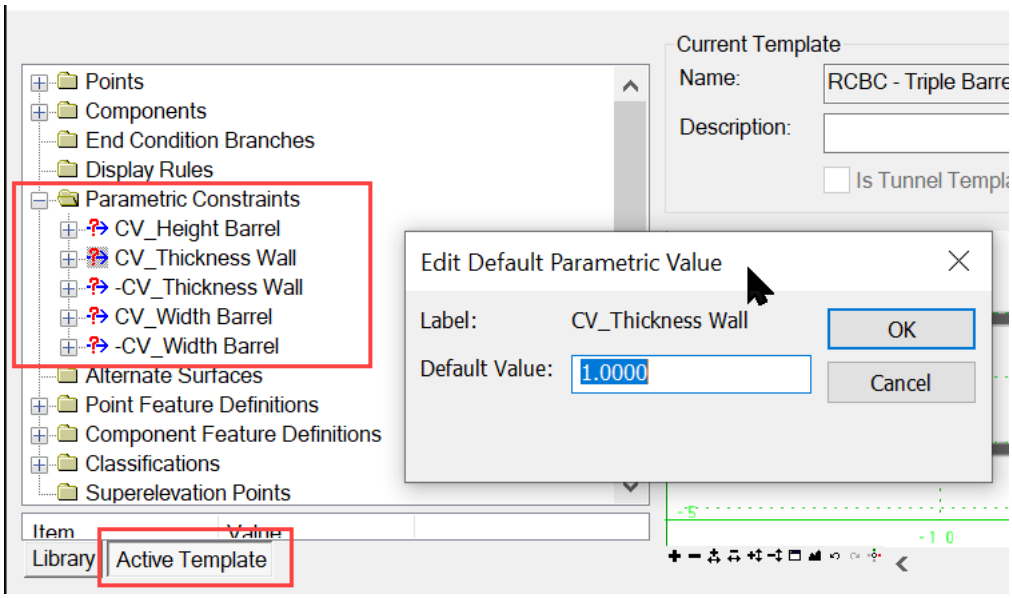




33. The RCBC-Triple Barrel template should look like the picture below.



34. Test the template by changing the various parametric constraint default values.



Exercise T3: Basic Dual Lane Road Shoulder Template

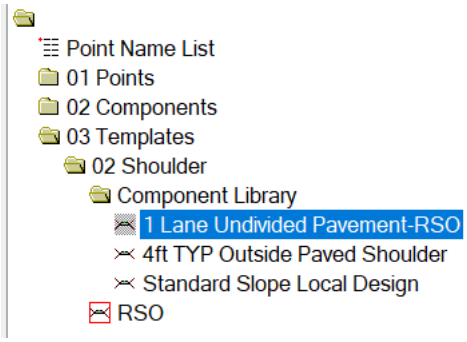
In this exercise we will teach how to assemble the components together to build a dual-lane shoulder template. The topics of **RSO, POMM**, **Apply Affixes**, **Mirroring** and **Poins Merging** are taught in this exercise.

In the Template Library **Open** the **03 Templates\03 Shoulder** folder and create a new template **RSO**.

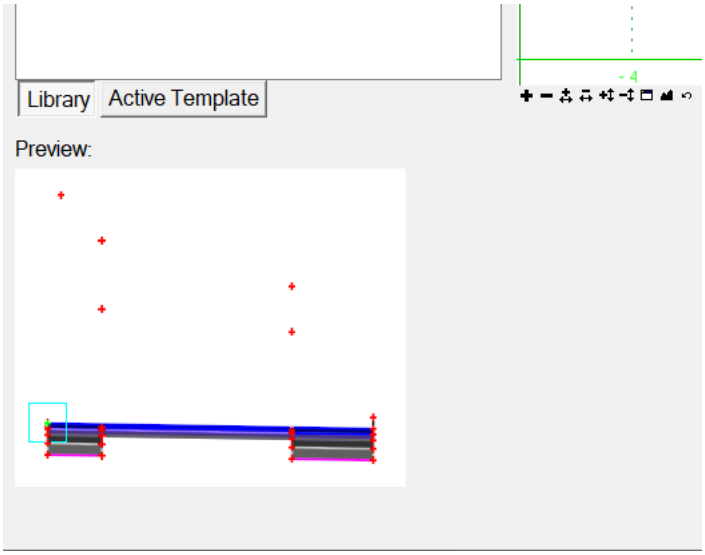
Note that **RSO** stands for **Right Side Only**. The objective is to build the right side of the road first then mirror it for a complete template.

1. Make **RSO** the active template.

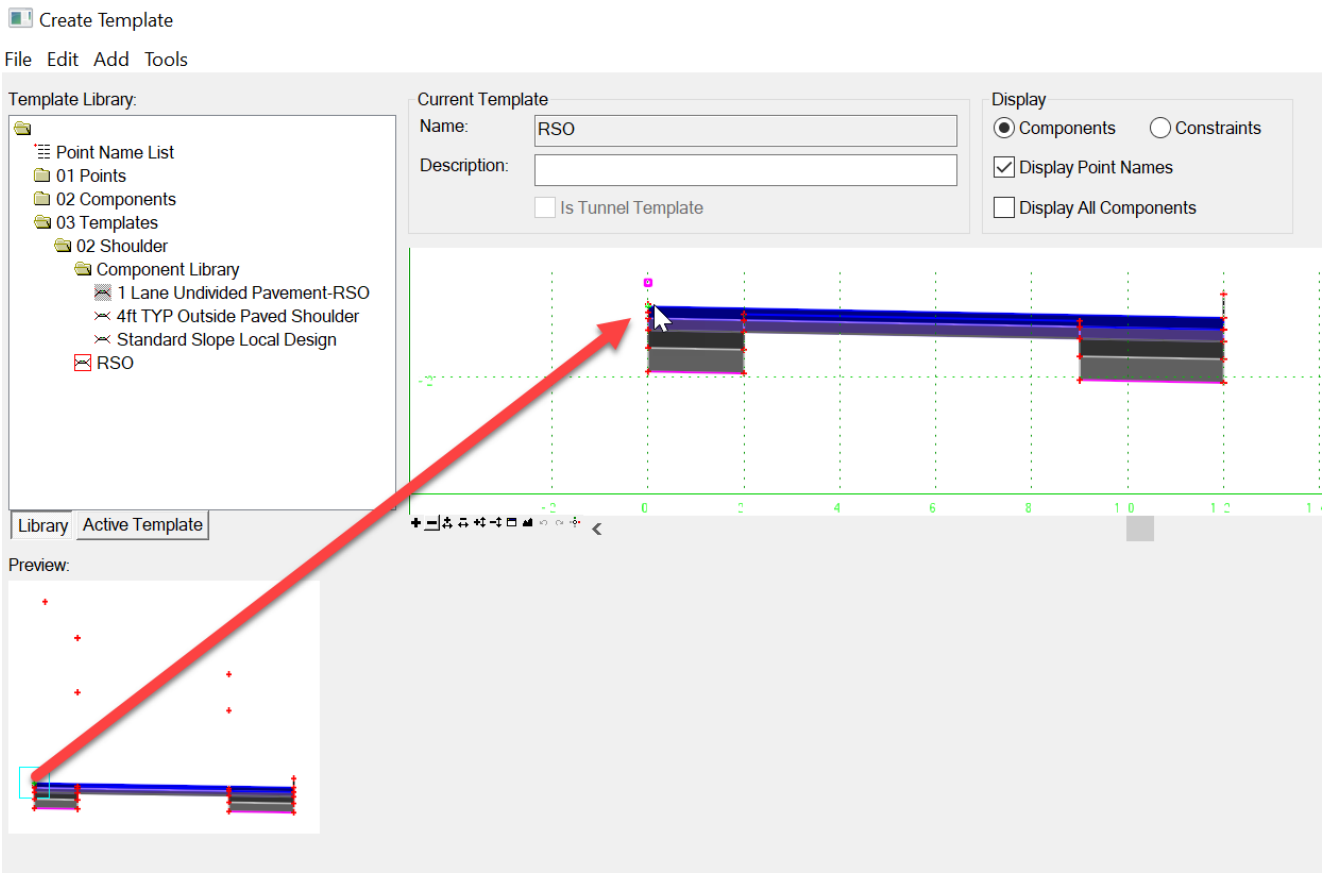
2. Open the **Component Library** folder and select (click once with the left mouse button) the component **1 Lane Undivided Pavement-RSO**.



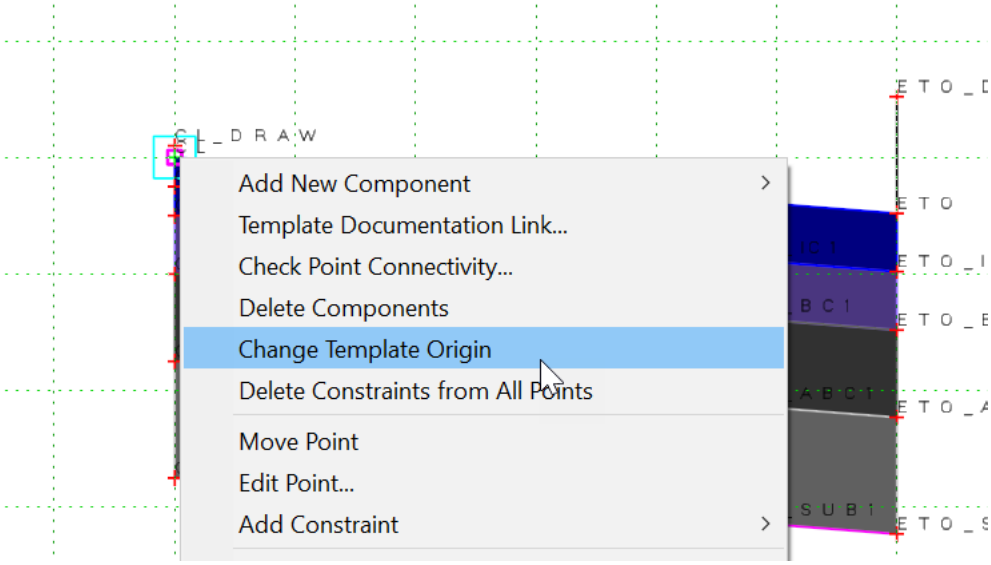
3. When the component or template is selected with a single click, it will be displayed in the bottom left corner the screen title **Preview**. Select the template origin (blue box) and drag the component over to the active template RSO by holding down the left mouse button.



of

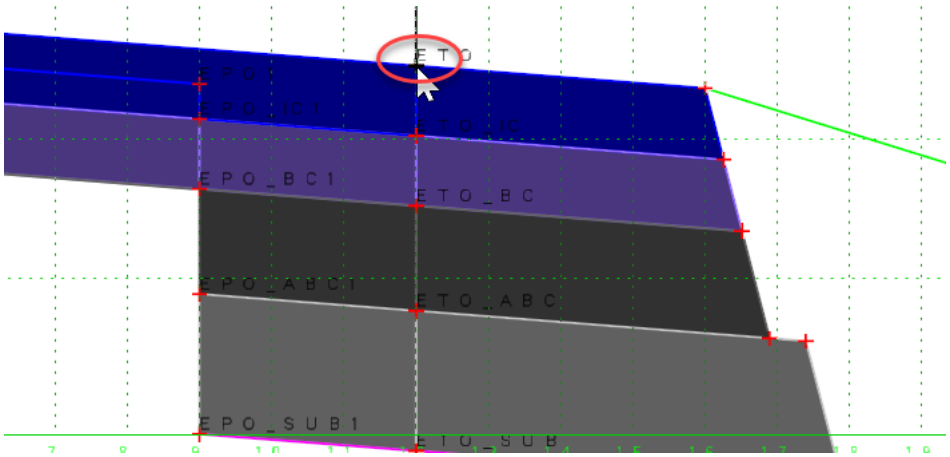


4. Make the **CL** point the template origin.



5. Select the component **4ft TYP Outside Paved Shoulder**.

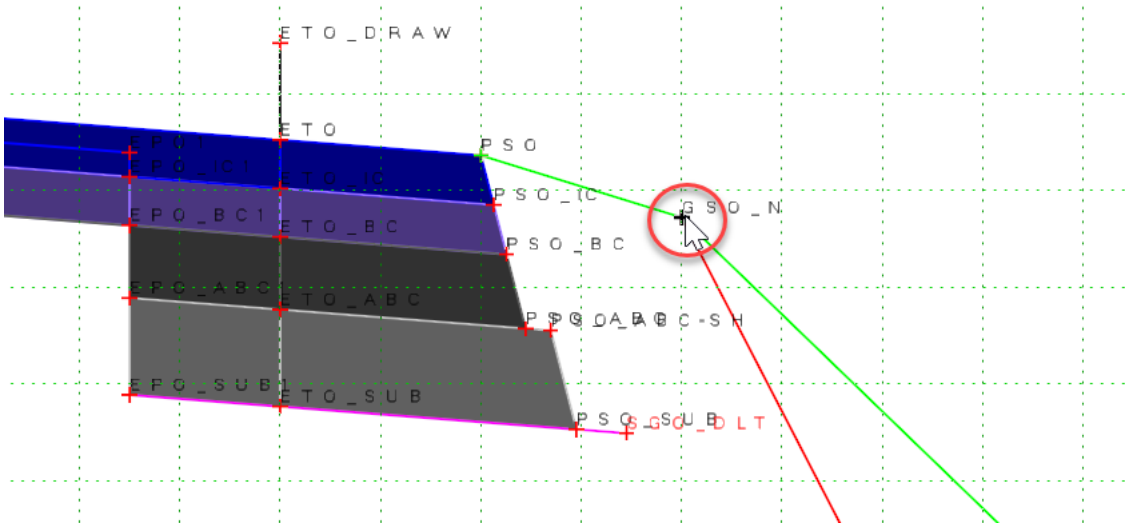
6. Drag it to the active **RSO** template screen, but this time the template origin of the shoulder will need to be placed on top of the **ETO** point for them to merge.



Note that all other pavement points underneath the ETO point are also merged with the paved shoulder points. These components were designed to connect to each other using the **POMM** sequence.

7. Select the component **Standard Slope Local Design**.

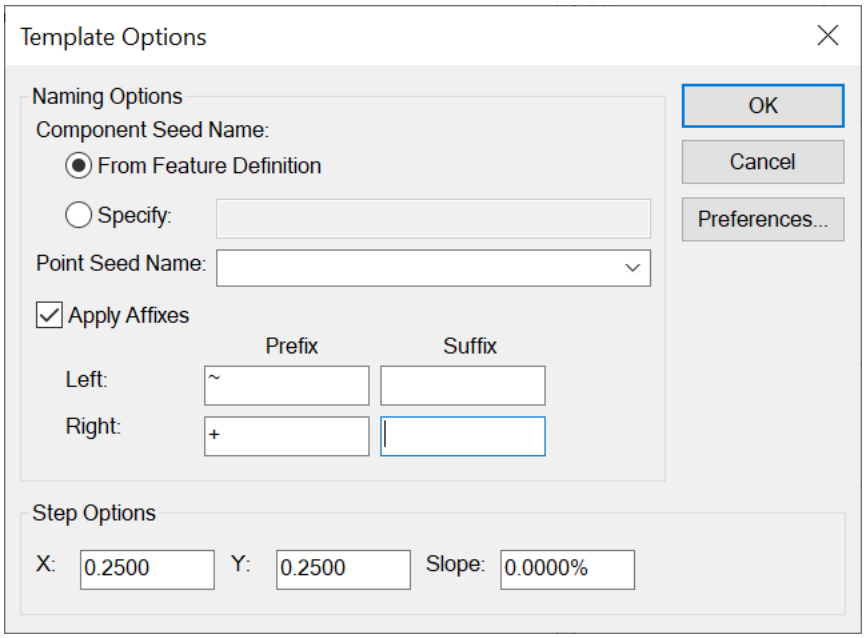
8. Drag to the active **RSO** template screen and merge the template origin of the end condition component with the shoulder point.



9. Create a new template called **1+1-LDSS**.

10. Under the template library menu select **Tools >>> Options**.

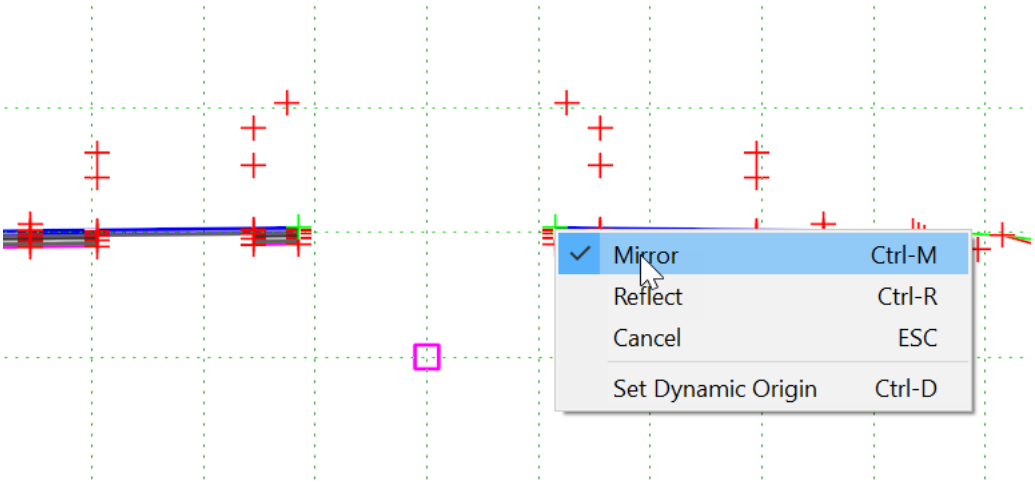
11. Check On **Apply Affixes** and under the Prefix collum, key-in "~" (tilde) for Left and "**+**" for Right prefixes. Set **0.2500** as the **Step Options** (snap increments).



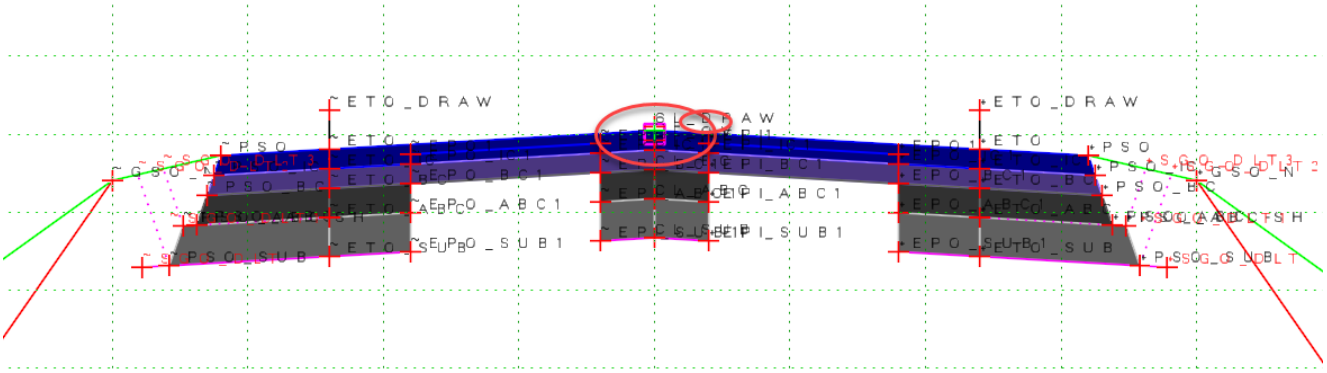
Note that these affixes/prefixes will be prepended to both point and component names when mirroring the **RSO**.

12. Select the template **RSO**.

13. Drag the RSO template over and prior to placement right mouse click and choose Mirror.



14. Place the **RSO** template at the template origin (pink box).



This template is almost finished. We will go over Triggers and Switches in the last exercise to finish all templates.

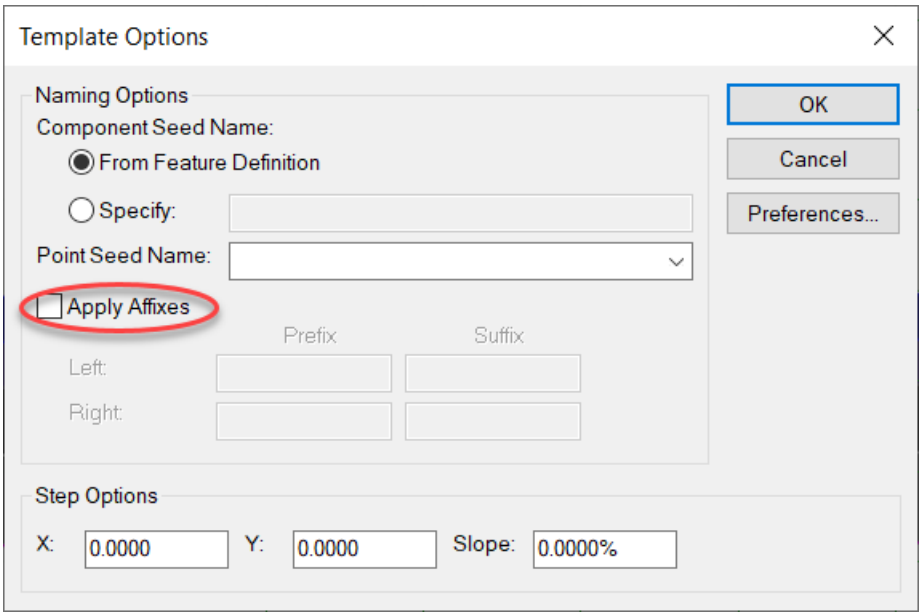
Exercise T4: Road Shoulder Left - Curb and Gutter Right Template

In this exercise we will teach how to assemble the components together using an alternative method to build a dual-lane shoulder template. The topics of **Compound Components**, **Apply Affixes**, **Mirroring, Reflect** and **Poins Merging** are taught in this exercise.

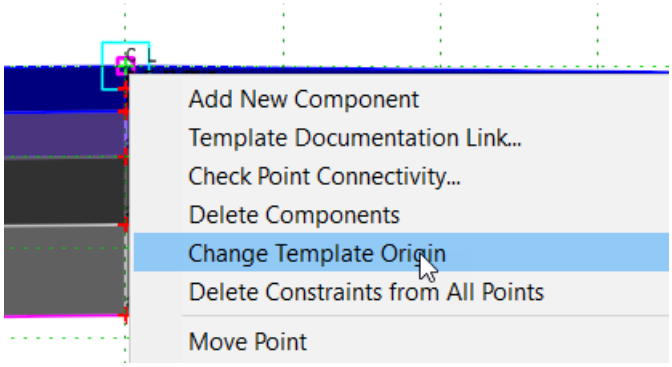
In the Template Library **Open** the **03 Templates\04 Shoulder-Curb and Gutter** folder and create a new template called **SH 2+2 C&G**.

In the Components ITL, there are two (2) types of components, single and compound. Single is a curb and gutter or paved shoulder component. Compound components are paved shoulder combined with the end conditions or pavement with both sides already completed. In this exercise, we will use compound components to quickly build a template.

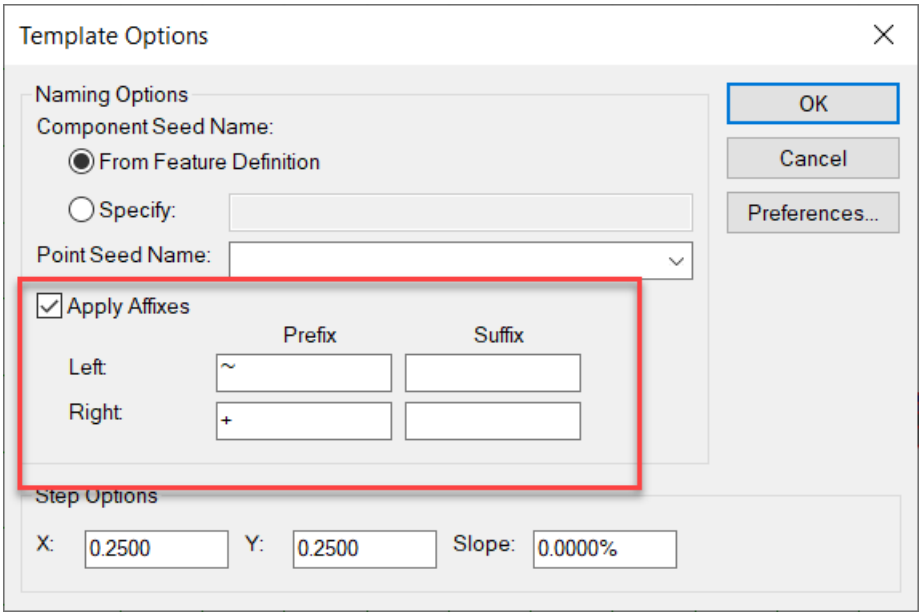
1. Make sure Apply Affixes is turned off. Since the pavement compound components are built with both sides in mind, affixes do not to be applied.



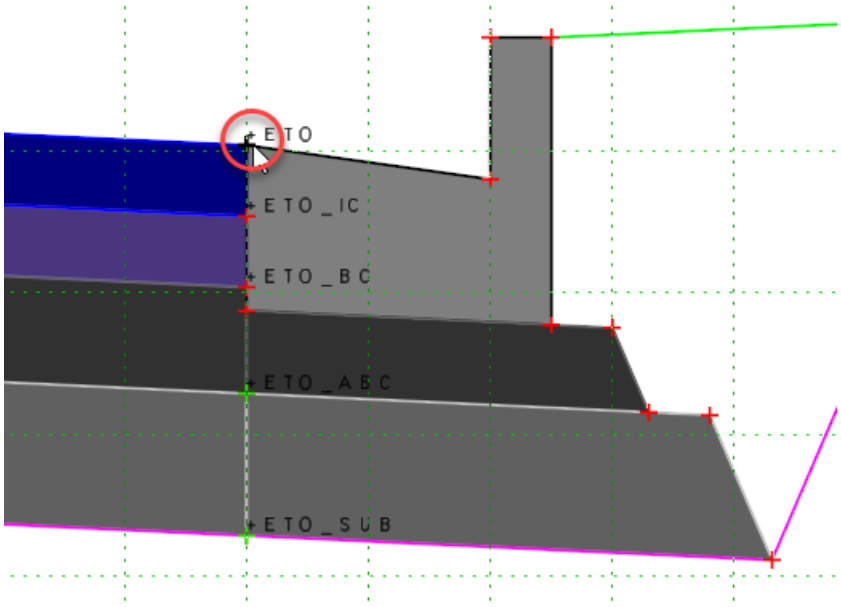
2. While template SH 2+2 C&G is active, Select (single and drag over the compound component 2+2 Lanes Undivided Pavement from the Preview screen (lower left). Place it at the template origin (pink box). If the template point was missed, use the Change Template Origin command to make the CL point the template origin.



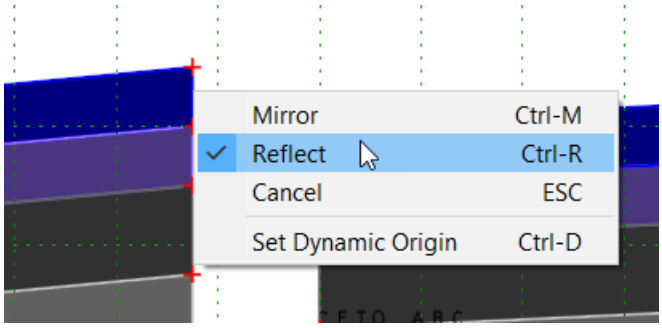
3. Turn on Apply Affixes and key-in the Prefix values as shown below.



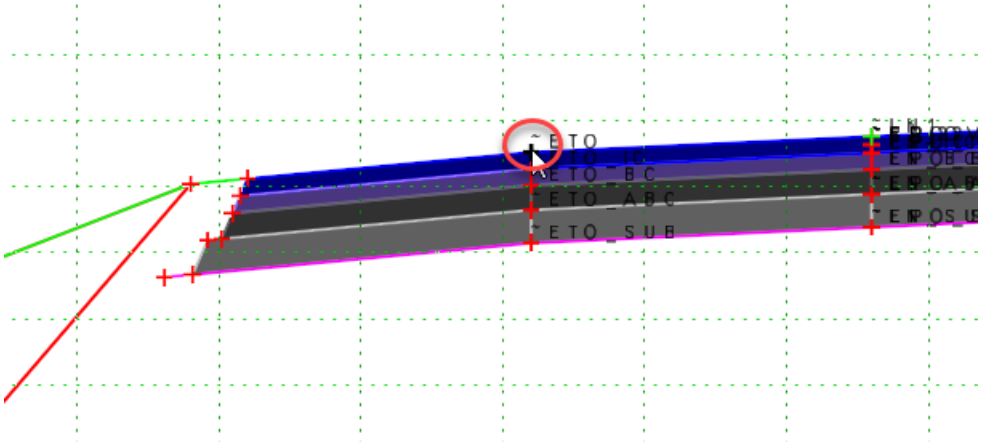
4. Select the compound component C&G Catch Slopes Outside and drag it over to the active template screen. Connect the C&G compound component to the right EOT point of the pavement.



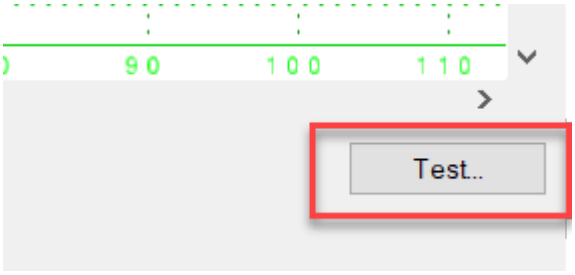
5. Select the compound component 10ft PS FD/ADSS Outside LT GR Flag – RSO and drag it over to the active template. Prior to placement right mouse click and choose Reflect.

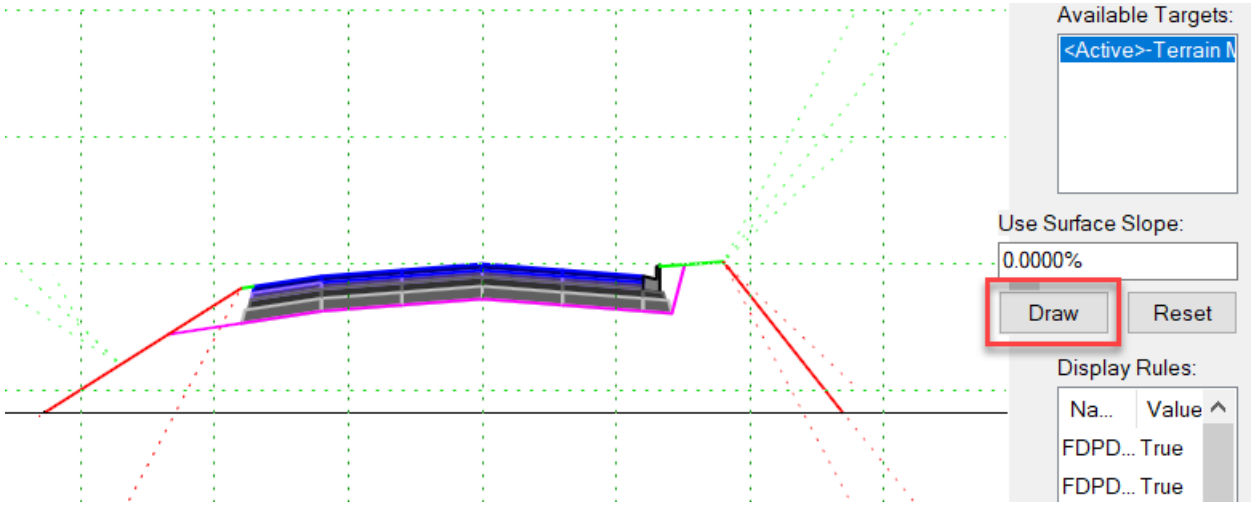


6. Connect the paved shoulder – end condition compound component to the left EOT point.



7. The template creation is complete. Test if it is working as expected.

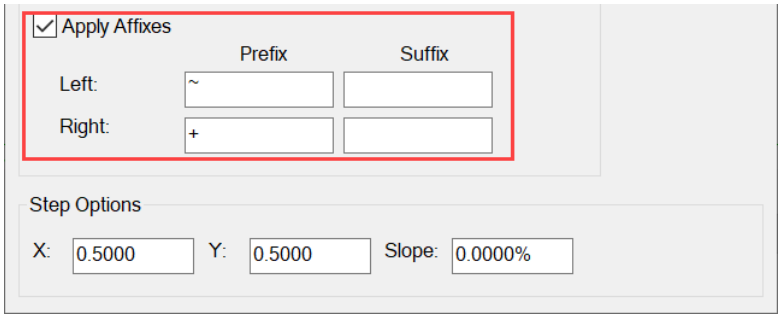




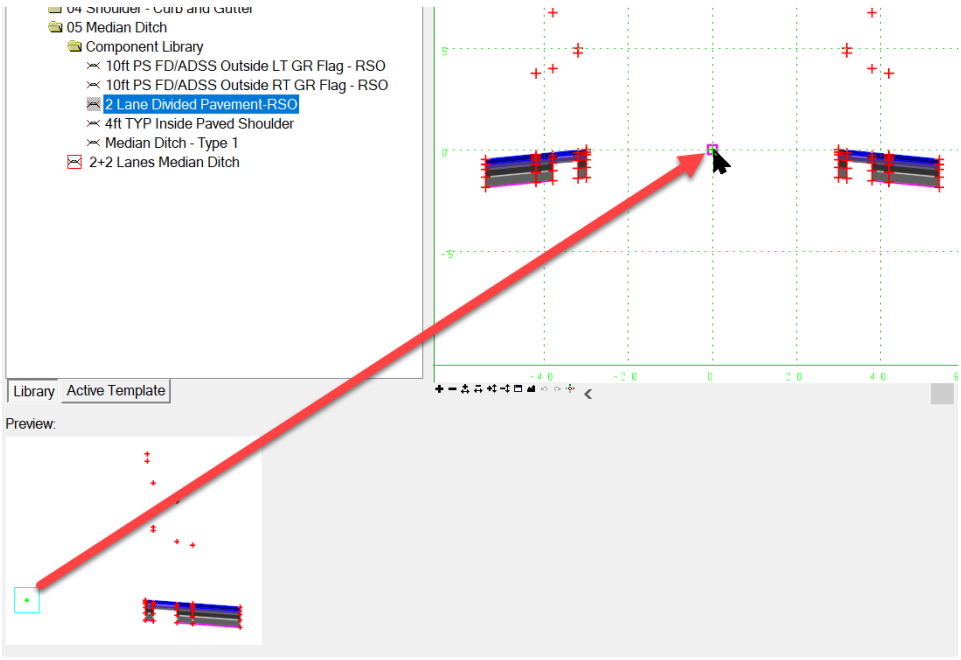
Exercise T5: Divided Facility with Median Ditch Template

In this exercise we will demonstrate the proper technique to assemble components to create a divide facility template with a median ditch. Lessons learned in this exercise include **Mirror** and **Reflect**, the **Order** of assembling the components and the **Merging** of points.

1. In the Template Library **Open** the **03 Templates\05 Median Ditch** folder and create a new template and name it **DF - 2+2 Lanes Median Ditch**.
2. Turn on **Apply Affixes**.

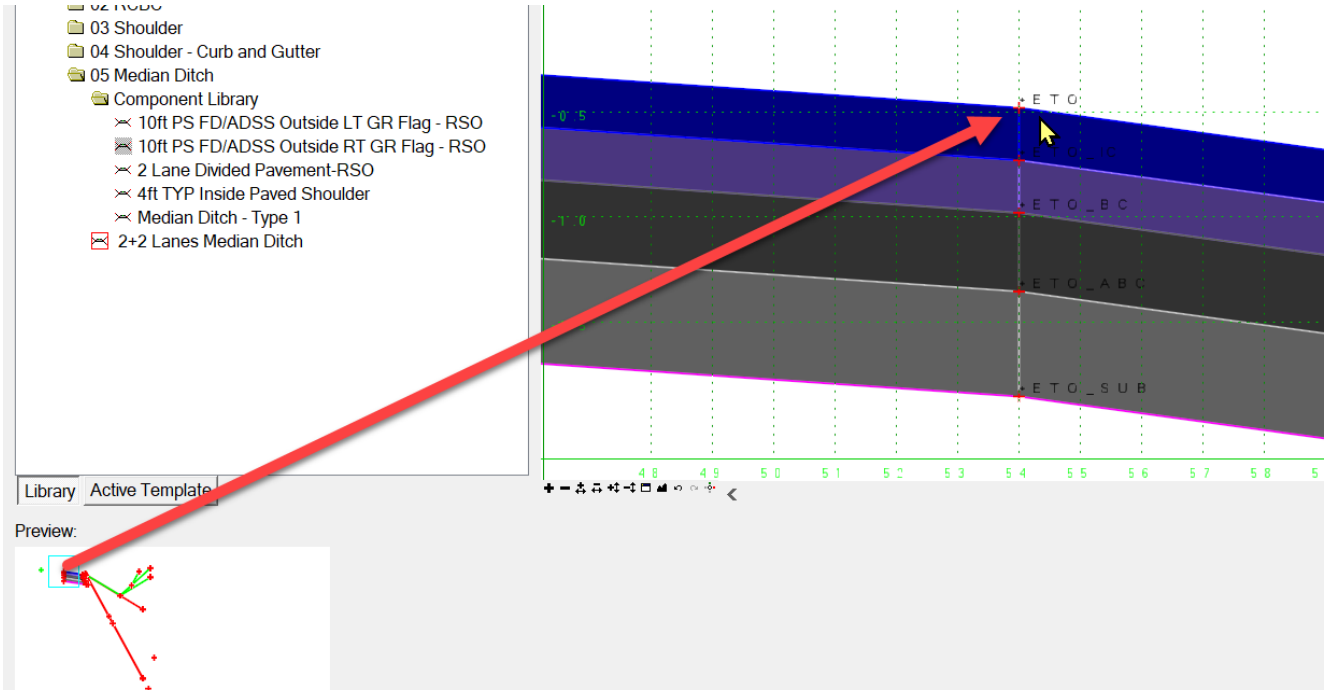


3. In the **Component Library** folder, select the **2 Lane Divided Pavement-RSO** compound component and drag it over to the active template screen. **Mirror** checked on place it at the template origin.

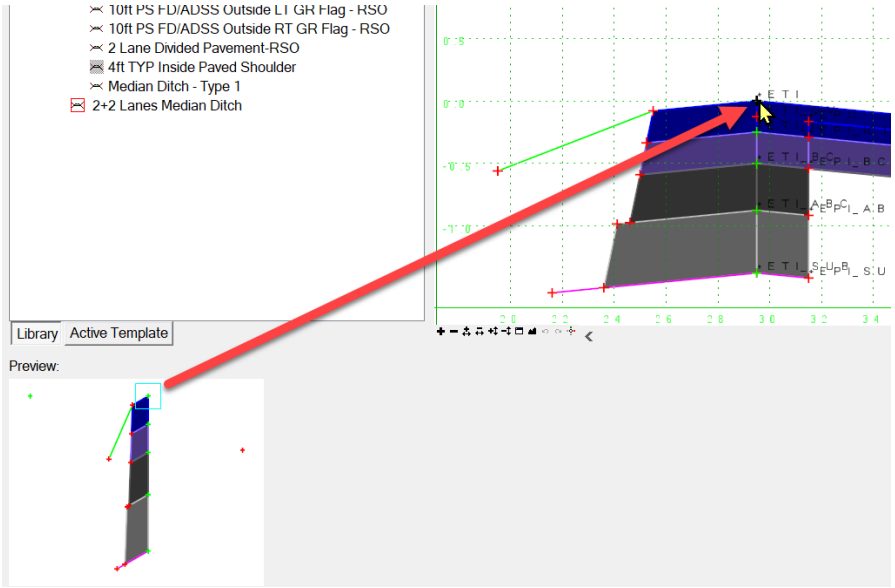


Note the default median width is 60', 30' left and right of Centerline. Also note the pavement wedge width has a gap to prevent the merging of points when the inside shoulders are added.

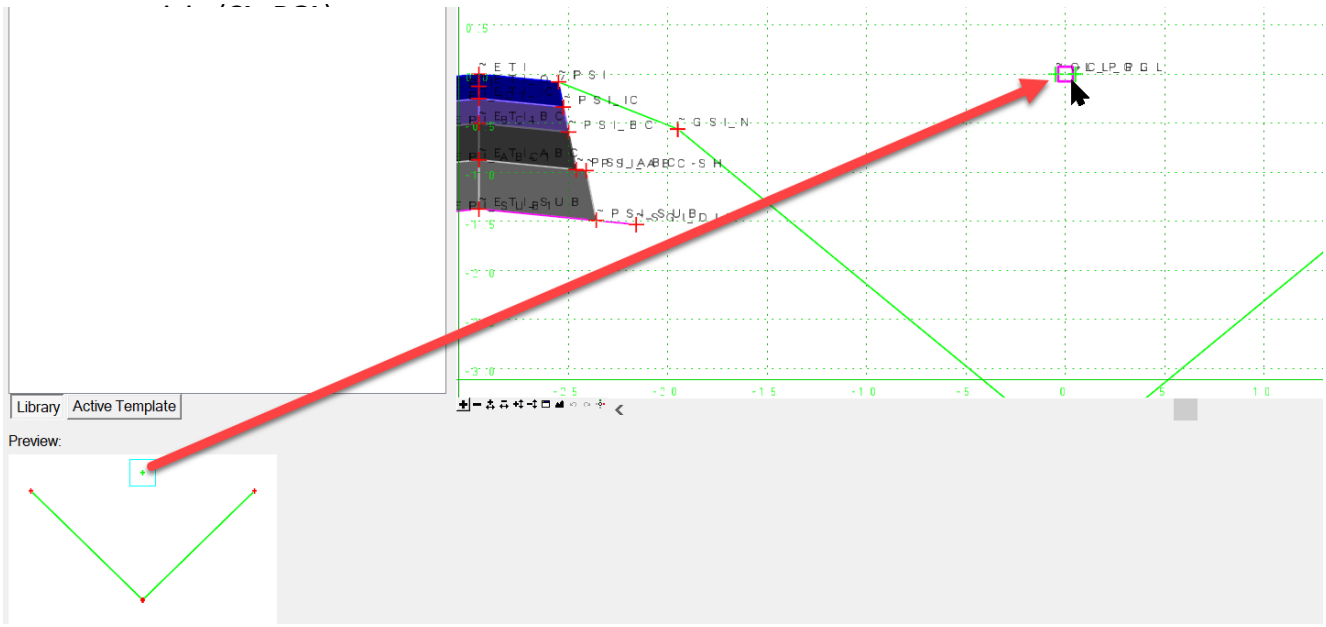
4. In the **Component Library** folder, select the **10ft PS FD/ADSS Outside RT GR Flag – RSO** compound component and drag it over to the active template screen. **Mirror** <u>unchecked</u> (off) place it at the **+ETO** point.



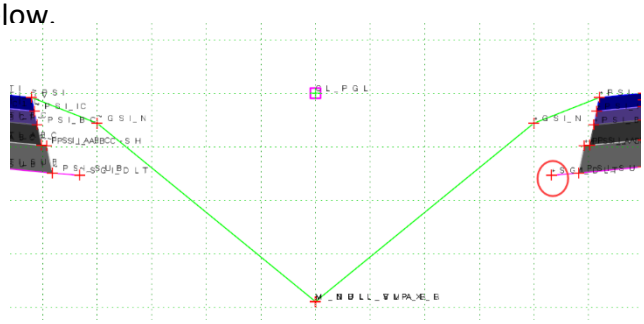
5. In the **Component Library** folder, select the **4ft TYP Inside Paved Shoulder** compound component and drag it over to the active template screen. **Mirror** checked on place it at the **+ETI** point.

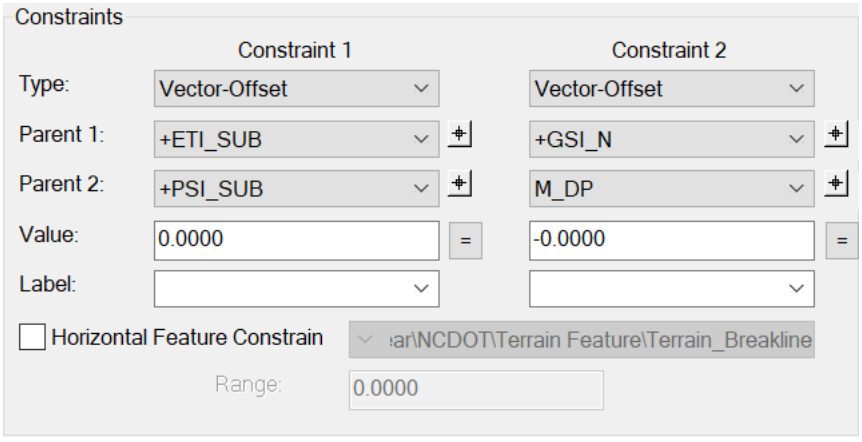


6. Turn off **Apply Affixes**.
7. In the **Component Library** folder, select the **4ft TYP Inside Paved Shoulder** component and drag it over to the active template screen. **Mirror** unchecked (off) place it at the template

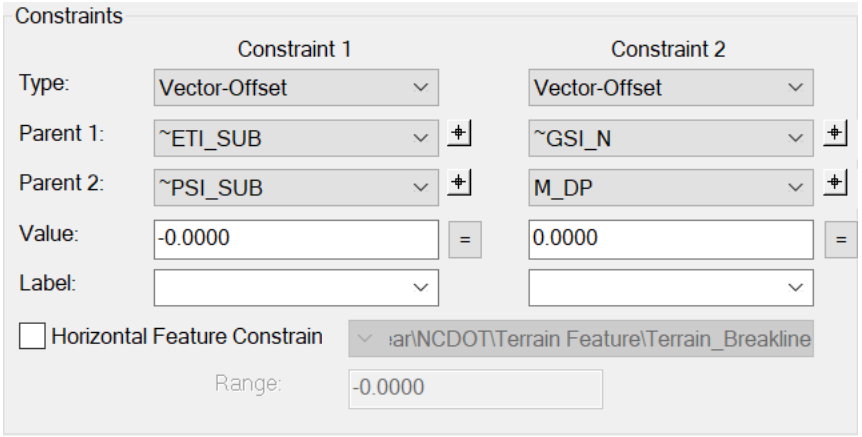


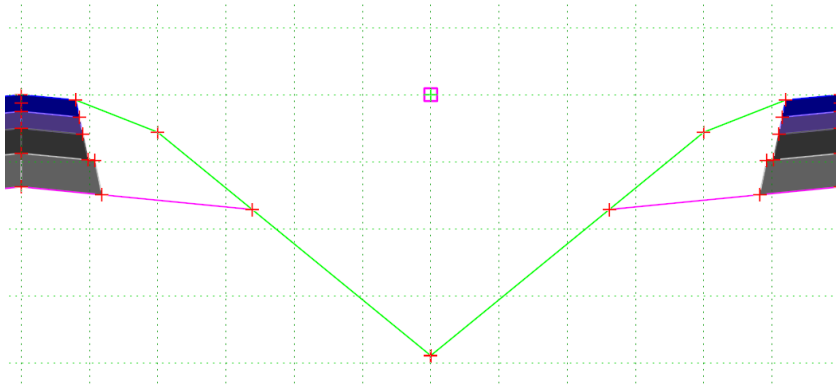
8. To tie the subgrade daylight point to the median ditch slopes, edit the **+SGI_DLT** point and constrain it as shown below.



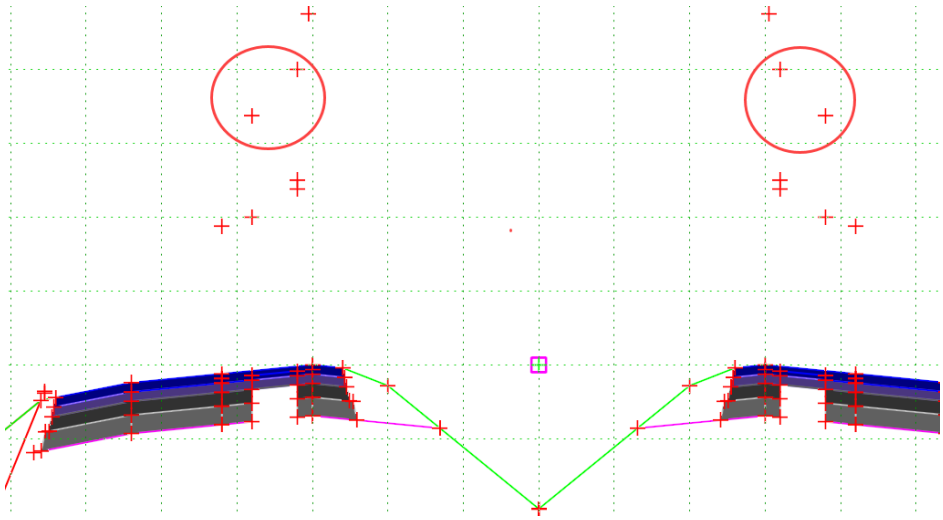


9. Edit the **~SGI_DLT** point and constrain it as shown below.

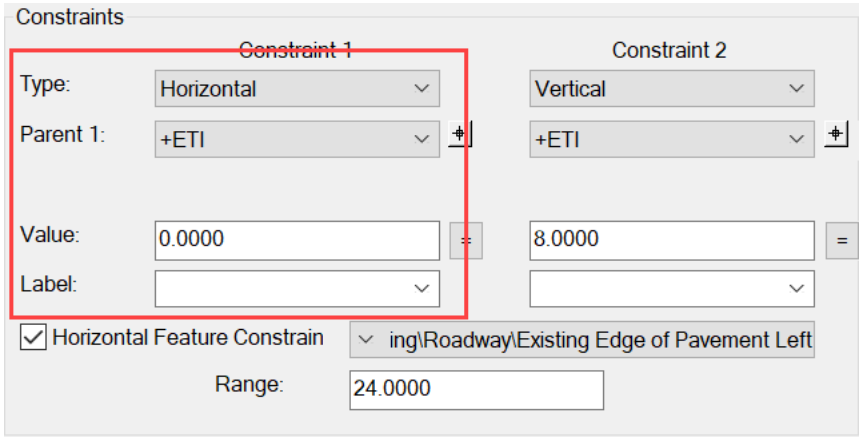




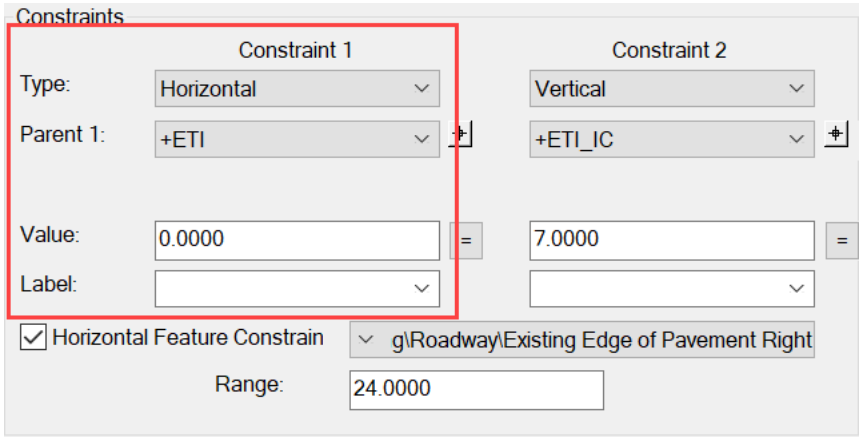
10. Edit the four (4) existing EOP seek points. They should be constrained horizontally 0’ from the **ETI** point on their respective side.



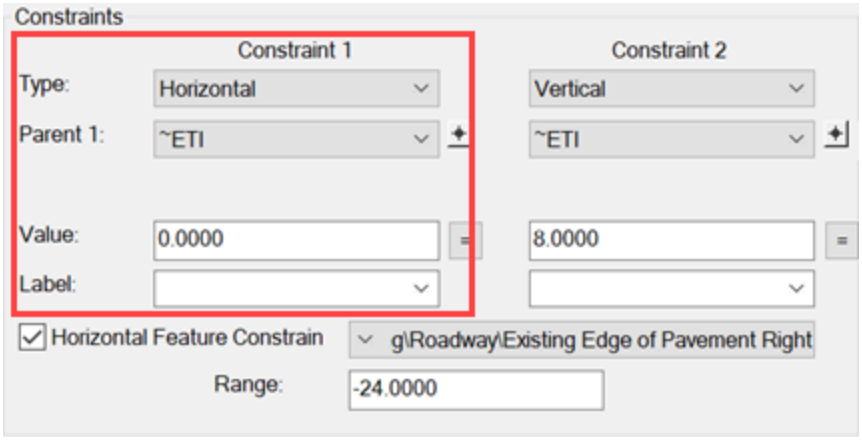
+SK_EPI



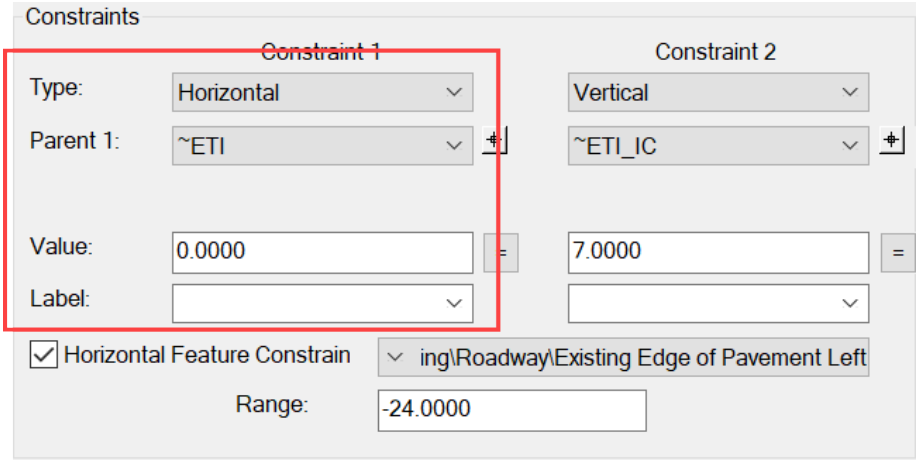
+SK_EPO



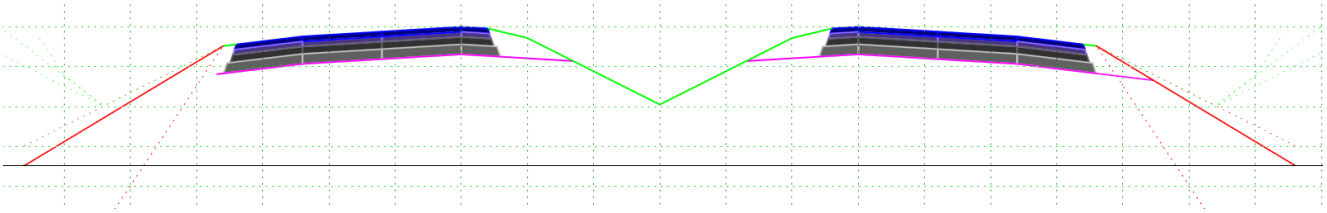
~SK_EPI (Note the Horizontal Feature Constrain is changed to Existing Edge of Pavement Right).



~SK_EPO (Note the Horizontal Feature Constrain is changed to Existing Edge of Pavement Left).



11. **Test** (lower right corner button) the template to see if it is working as intended.

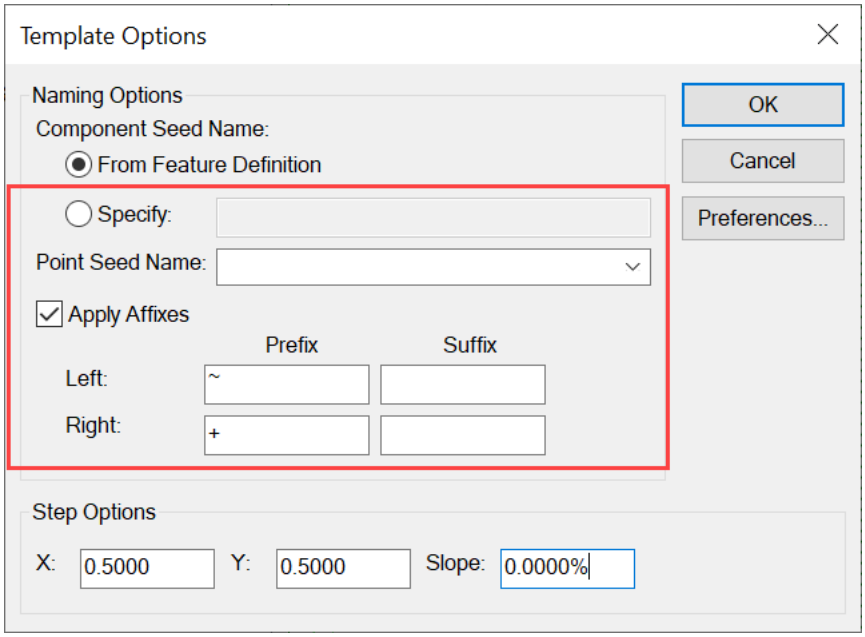


Exercise T6: Divided Facility with Raised Median Template

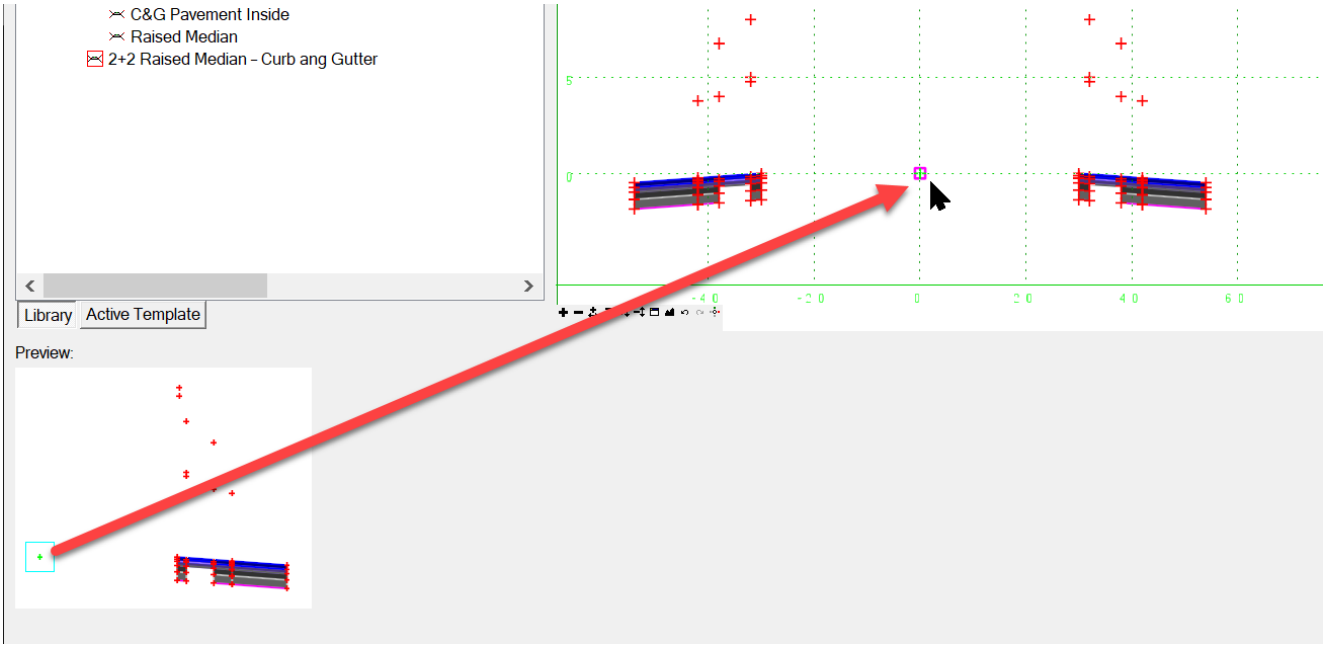
In this exercise we will demonstrate the proper technique to assemble components to create a divide facility template with a raised median. Lessons learned in this exercise include **RSO**, **Mirror**, the **Order** of assembling a template and the Merging of points.

1. In the Template Library Open the **03 Templates\06 Raised** Median folder and create a new template and name it **2+2 Raised Median – Curb ang Gutter**.

2. Turn on **Apply Affixes**.

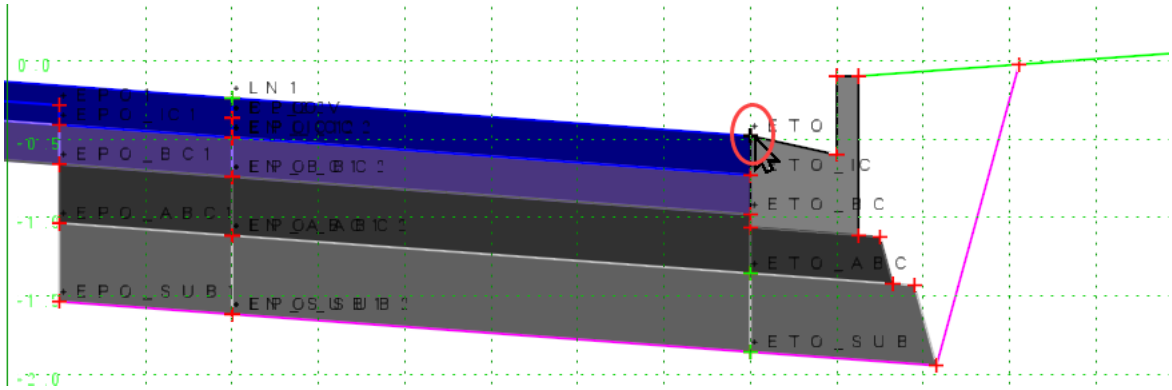


3. In the **Component Library** folder, select the **2 Lane Divided Pavement-RSO** component and drag it over to the active template screen. **Mirror** checked on place it at the template origin.

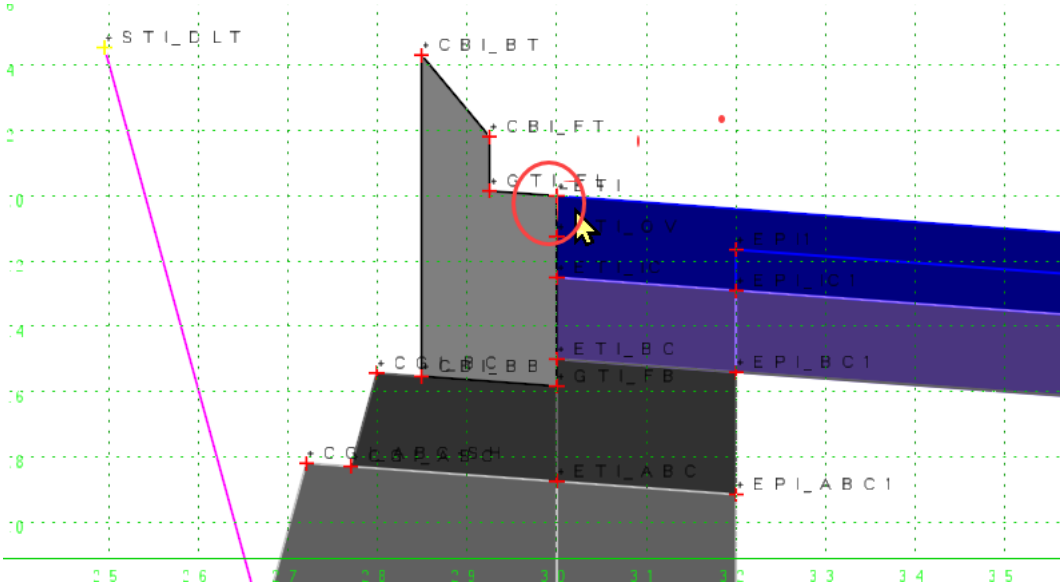


4. Note the default median width is 60', 30' left and right of Centerline. Also note the pavement wedge width has a gap to prevent the merging of points when the median curbs are added.

5. In the **Component Library** folder, select the **C&G Catch Slopes Outside** compound component and drag it over to the active template screen. **Mirror** checked on place it at the **+ETO** point.

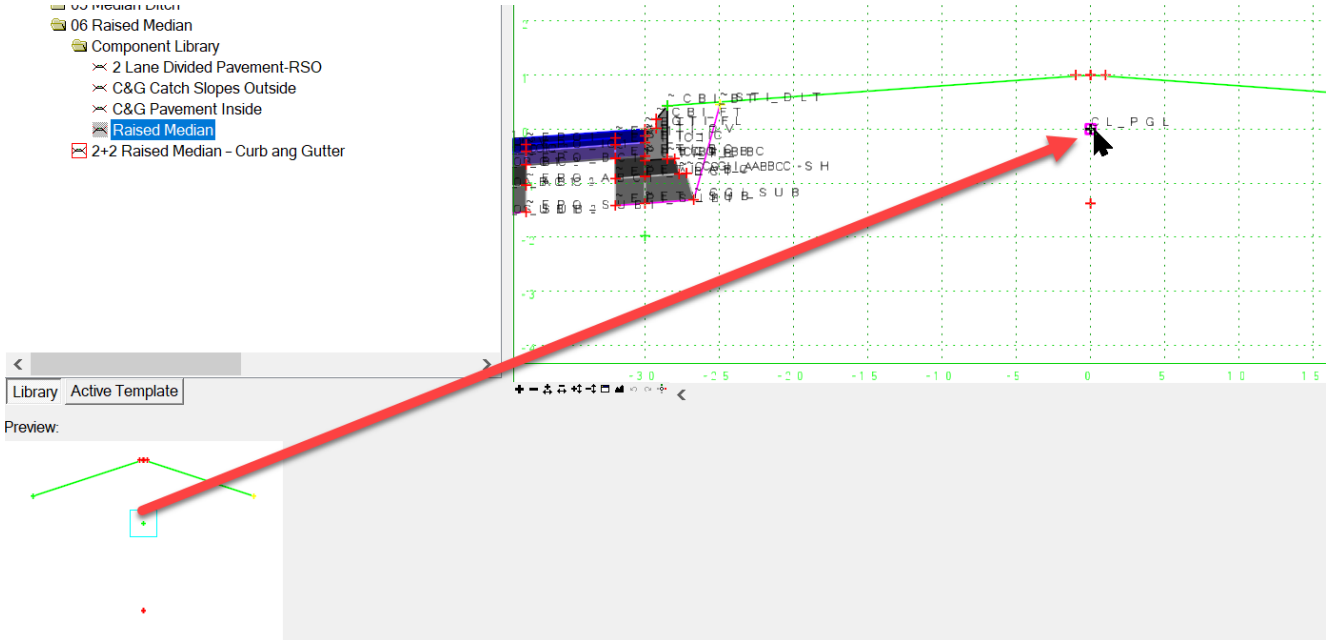


6. In the **Component Library** folder, select the **C&G Pavement Inside** compound component and drag it over to the active template screen. **Mirror** checked on place it at the **+ETI** point.

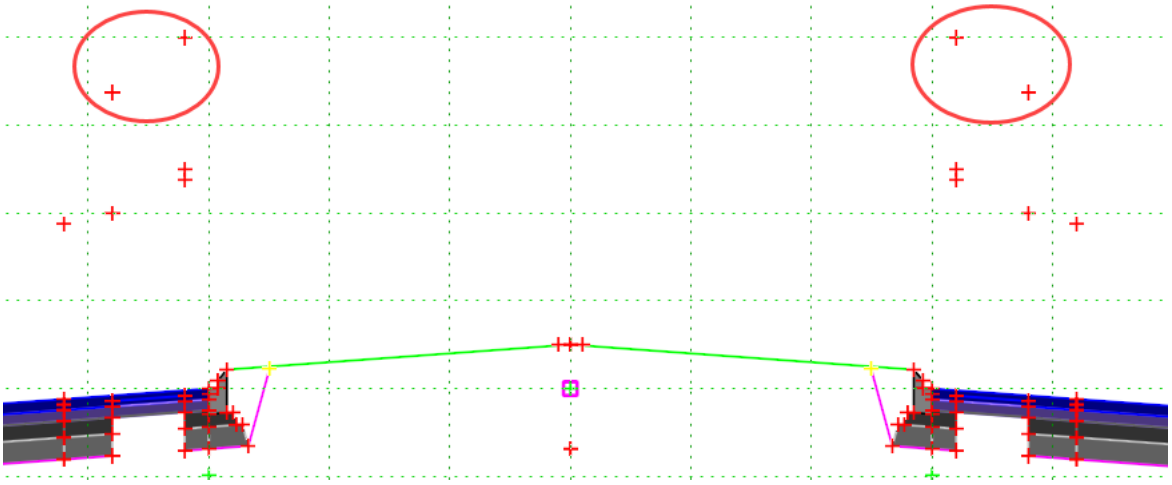


7. Turn off **Apply Affixes**.

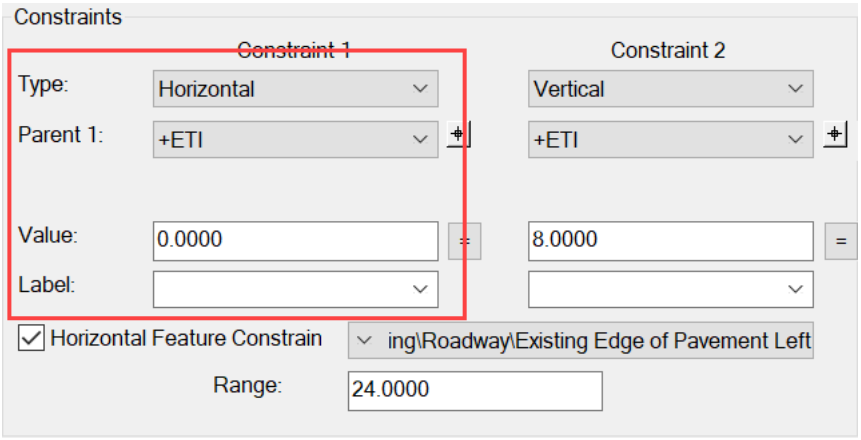
8. In the **Component Library** folder, select the **Raised Median** compound component and drag it over to the active template screen. **Mirror** unchecked (off) on place it at the template origin (**CL_PGL**).



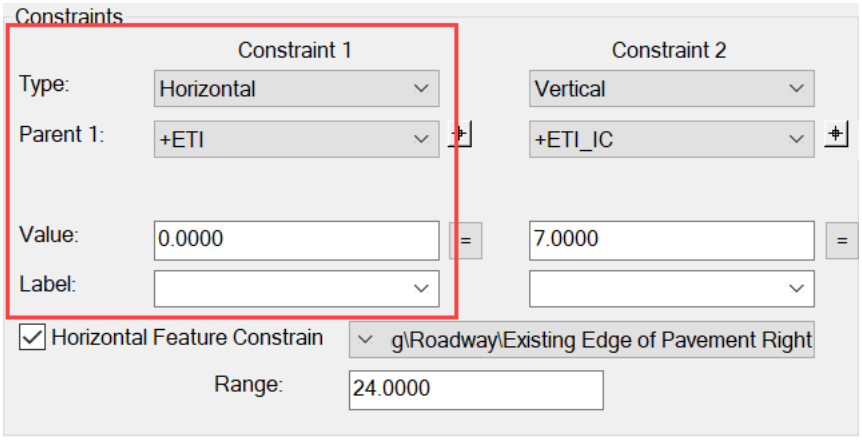
9. Edit the four (4) existing EOP seek points. They should be constrained horizontally 0’ from the ETI point on their respective side.



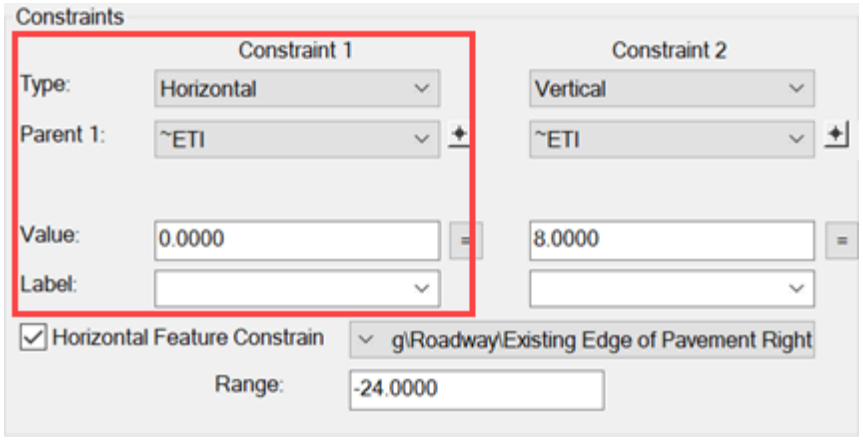
+SK_EPI



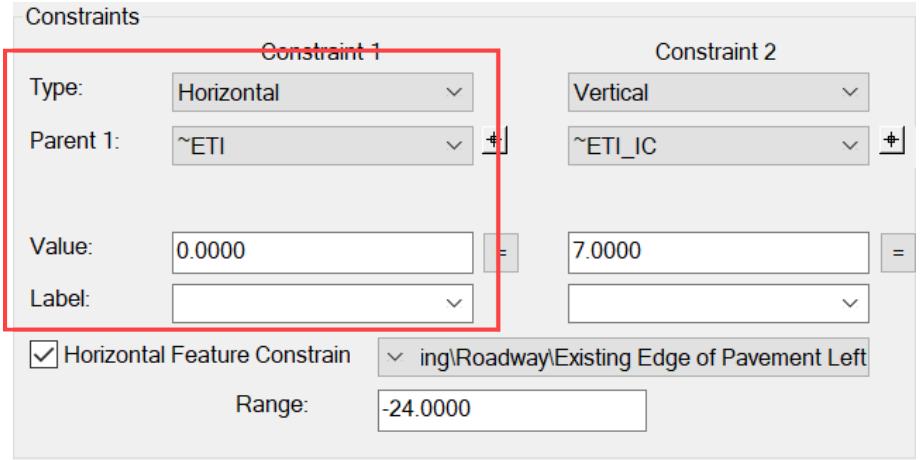
+SK_EPO



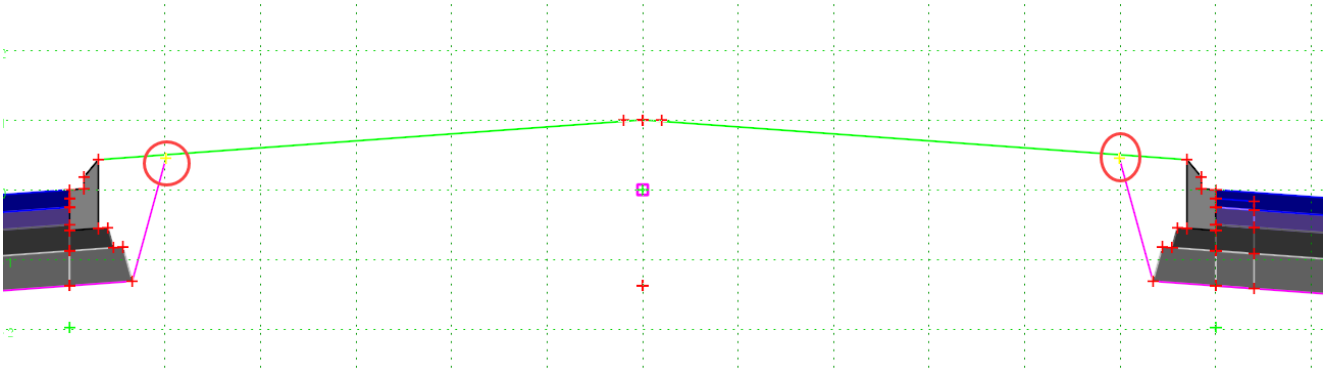
~SK_EPI (Note the Horizontal Feature Constrain is changed to Existing Edge of Pavement Right).



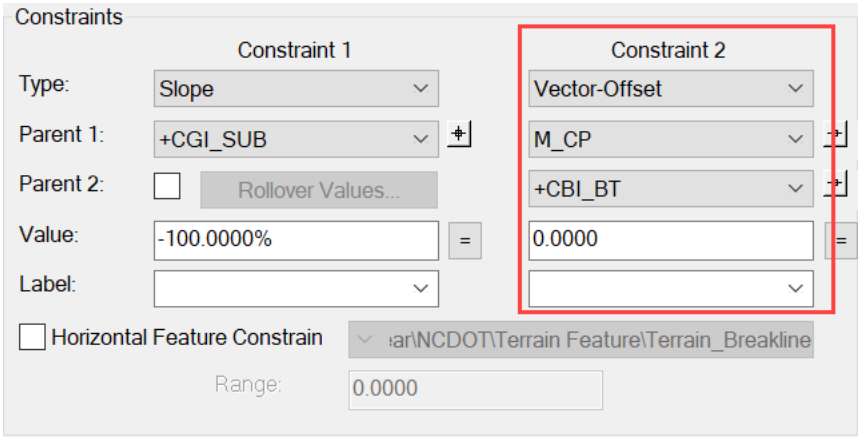
~SK_EPO (Note the Horizontal Feature Constrain is changed to Existing Edge of Pavement Left).



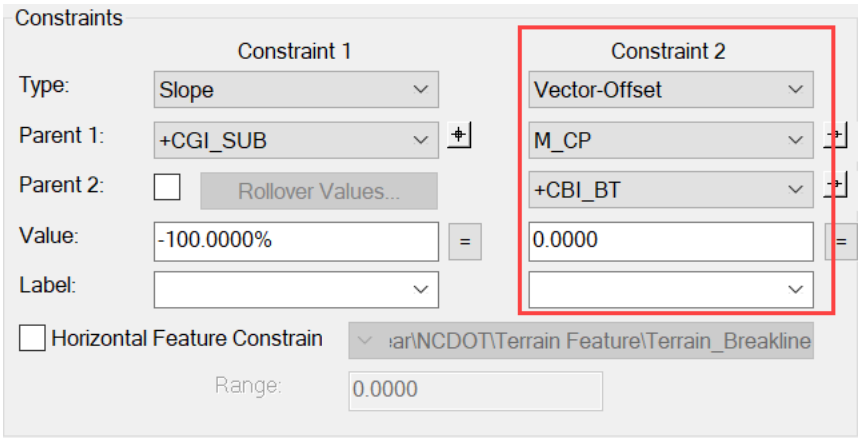
10. Edit the two (2) yellow partially constrained subgrade daylight (DLT) points in the median. Constrain them as shown below.



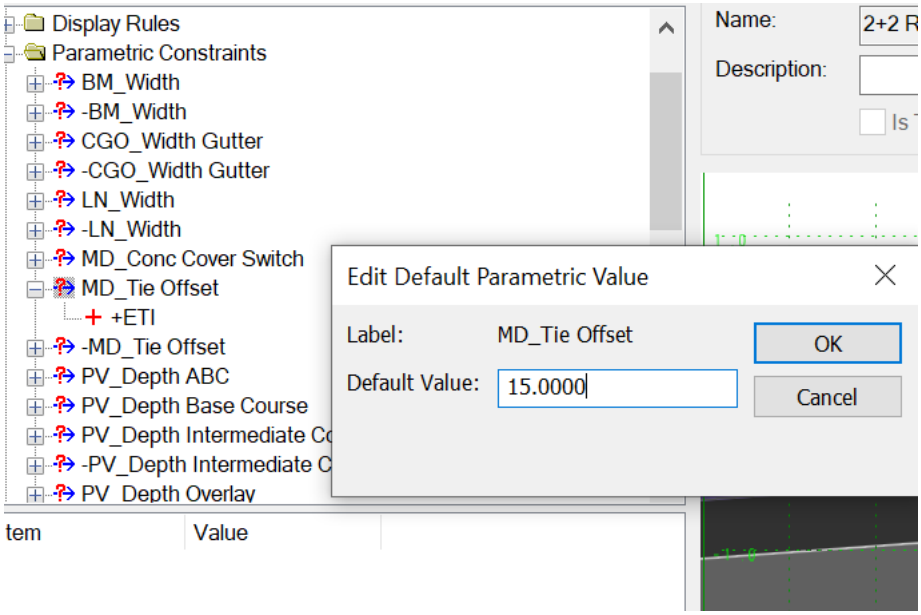
+STI_DLT



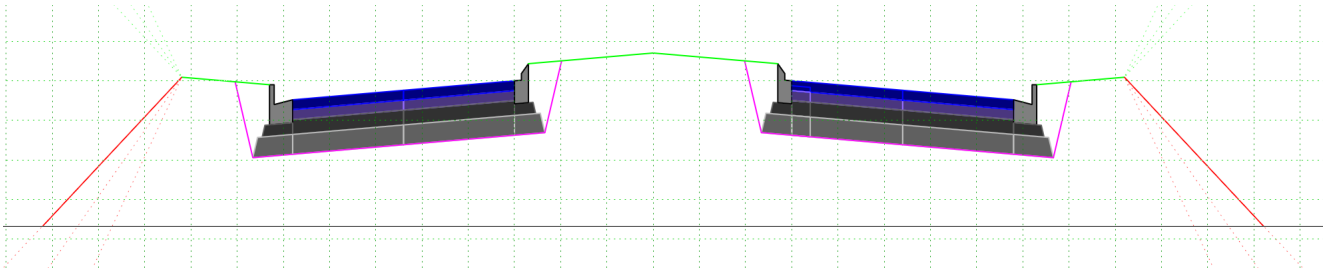
~STI_DLT



11. Before finalizing the template, change the default value of the **Parametric Constraint Label MD_Tie_Offset** from **30** to **15**.



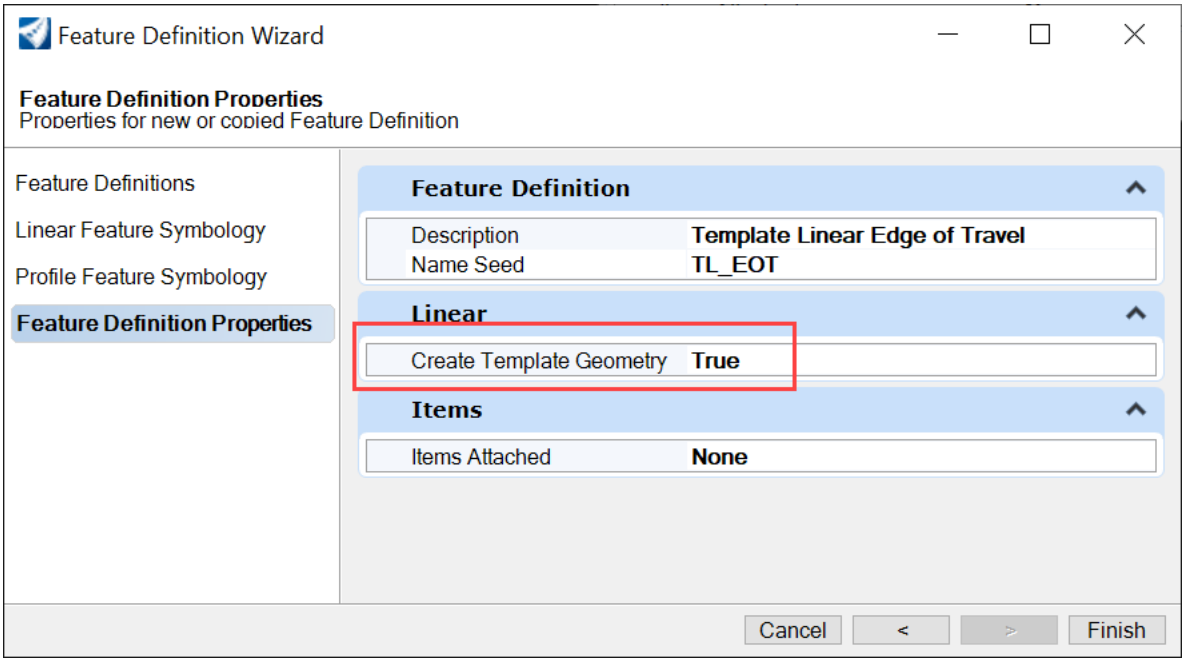
12. **Test** (lower right corner button) the template to see if it is working as intended.



Exercise T7: Triggers and Switches

Up to this point most of the templates created in these exercises are incomplete. They are required to be finalized with triggers and switches. A Trigger is a mechanism to “move” a point from its default location usually by means of a Horizontal Feature Constraint (drawn in a design file). A Switch is a mechanism to turn off or on a component usually by means of a Parametric Constraint. Both triggers and switches use Display Rules to achieve the desired effect.

By default, all linear features generated by the template points are drawn in the 3D model. Only certain linear features, such as the EOT, C&G and paved shoulder lines are also drawn in the 2D plan view. The symbology of how the linear features are displayed in 3D and 2D is controlled through the Element Templates and Feature Symbologies. The setting that enables the drawing of linear features in the 2D plan view is in the Feature Definition. Create Template Geometry must be set to True in order to draw in the 2D plan view.

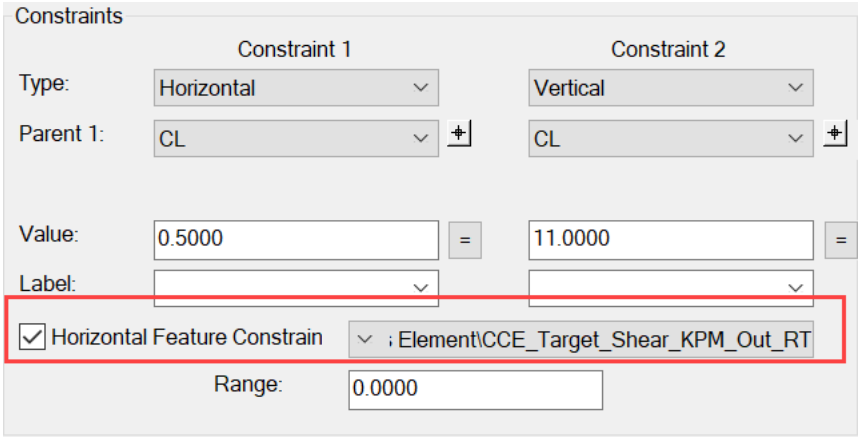


Triggers – Shear at EOT

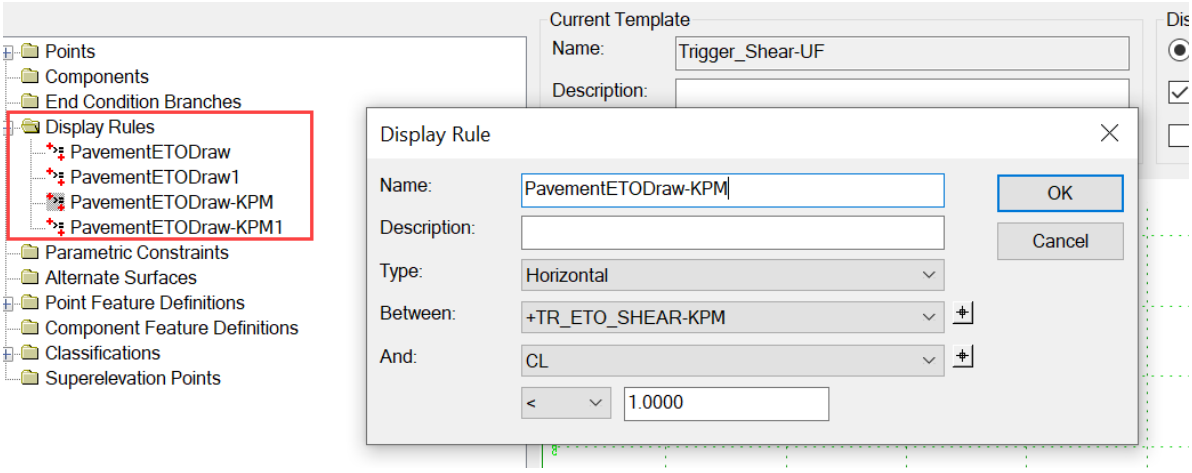
There are two (2) types of shear triggers, at intersections (INT) when the pavement markings (2D Linear Features and 3D pavement markings) are removed and keep pavement markings (**KPM**), such as placing a Hydro ditch.

1. In the Template Library **Open** the **03 Templates\07 Triggers-Switches\Component Library** folder and select the **Trigger_Shear-UF** component.
2. Inspect the four (4) trigger (TR_) points. Note how each has a **Horizontal Feature Constrain** Feature Definition.

+TR_ETO_SHEAR-KPM



3. Also Inspect the four (4) **Display Rules** (**Active Template** tab) associated with these trigger points.

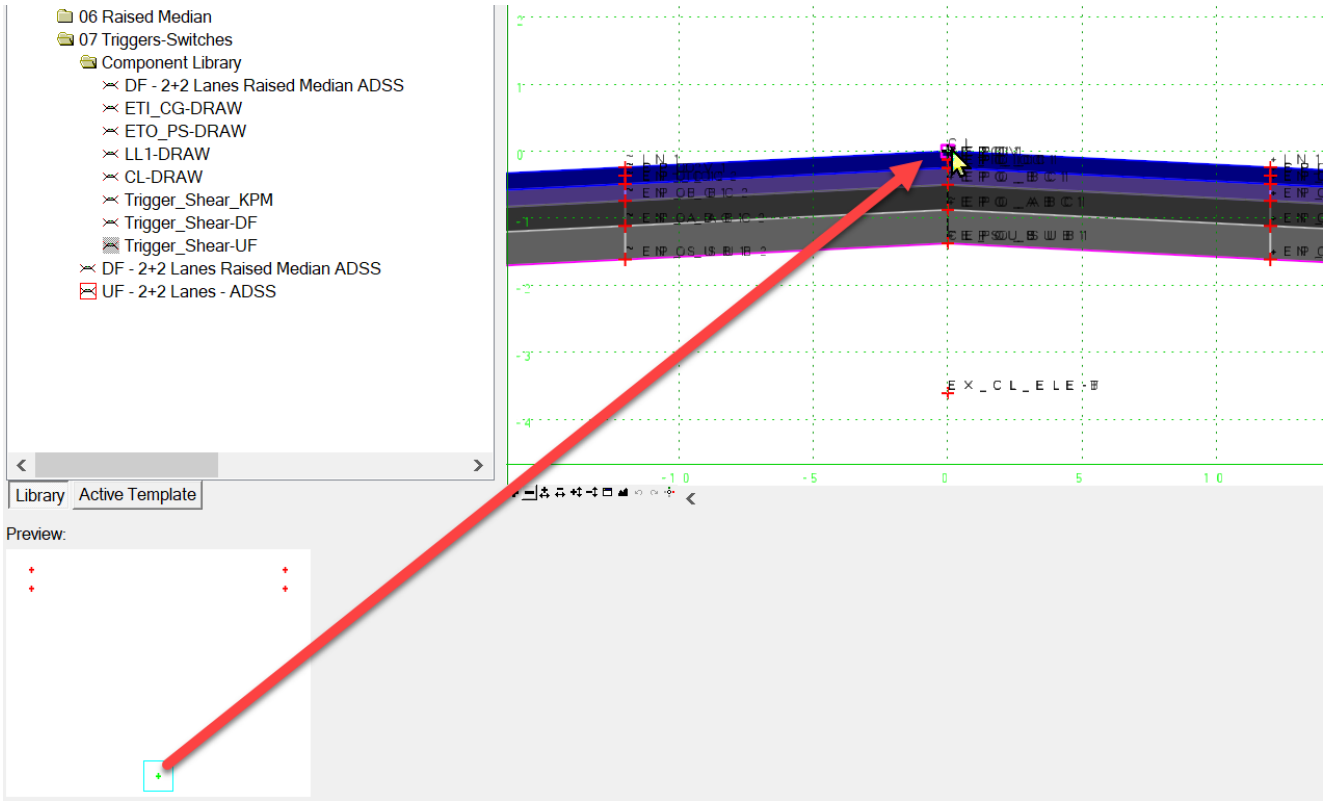


This **Display Rule** reads if the horizontal distance between point **+TR_ETO_SHEAR-KPM** and the **CL** point (the template default state) then the condition is True and everything this drawn. No shearing at the EOT will occur. If the trigger is activated and the **+TR_ETO_SHEAR-KPM** point moves to the right (greater than 1’), then the condition is **False** resulting in the shearing (turning off) of the components outside the EOT point.

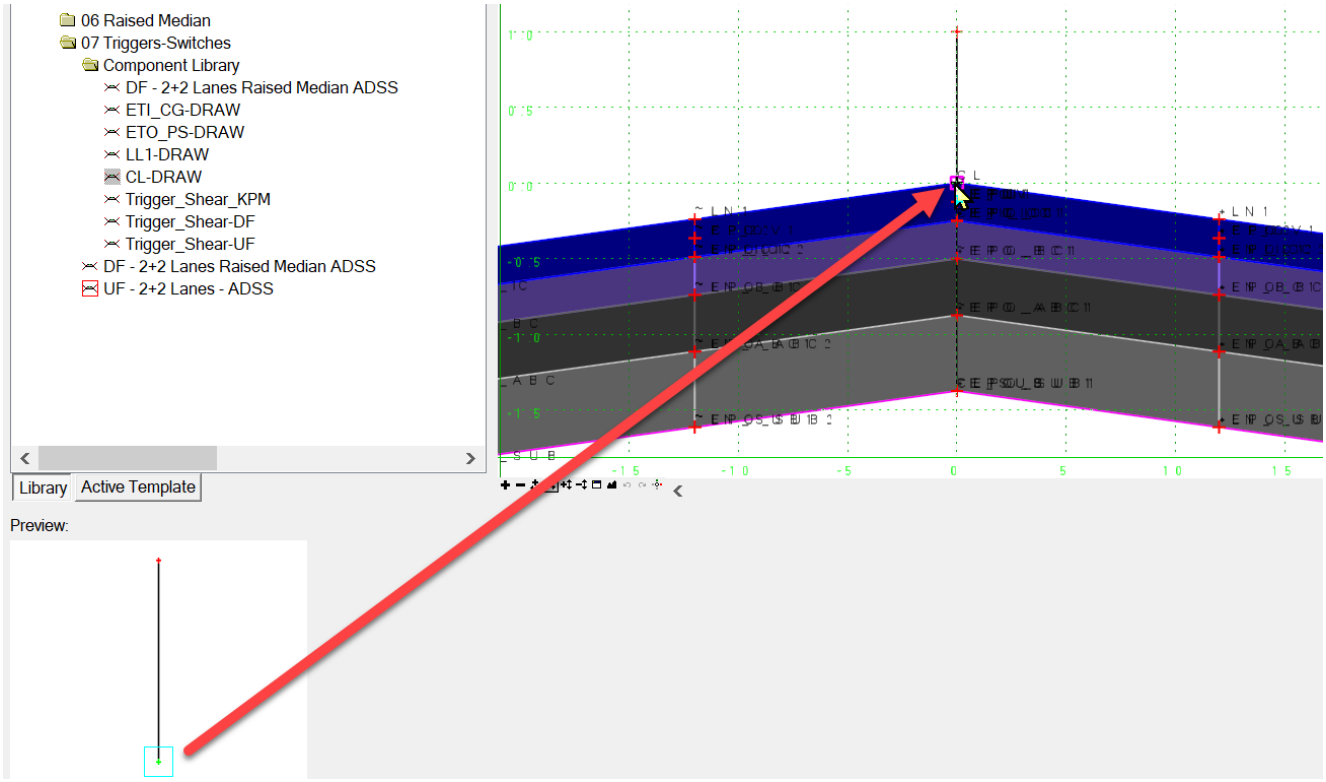
4. In the Template Library **Open** the **03 Templates\07 Triggers-Switches** folder and select the **UF - 2+2 Lanes – ADSS** template.
5. Turn off **Apply Affixes**.

With the trigger points with the display rules in place, add the components to draw the 2D plan view features and the 3D pavement markings.

6. In the **Component Library** folder, select the **Trigger_Shear-UF** component and drag it over to the active template screen. **Mirror** checked off place it at the **CL** point.

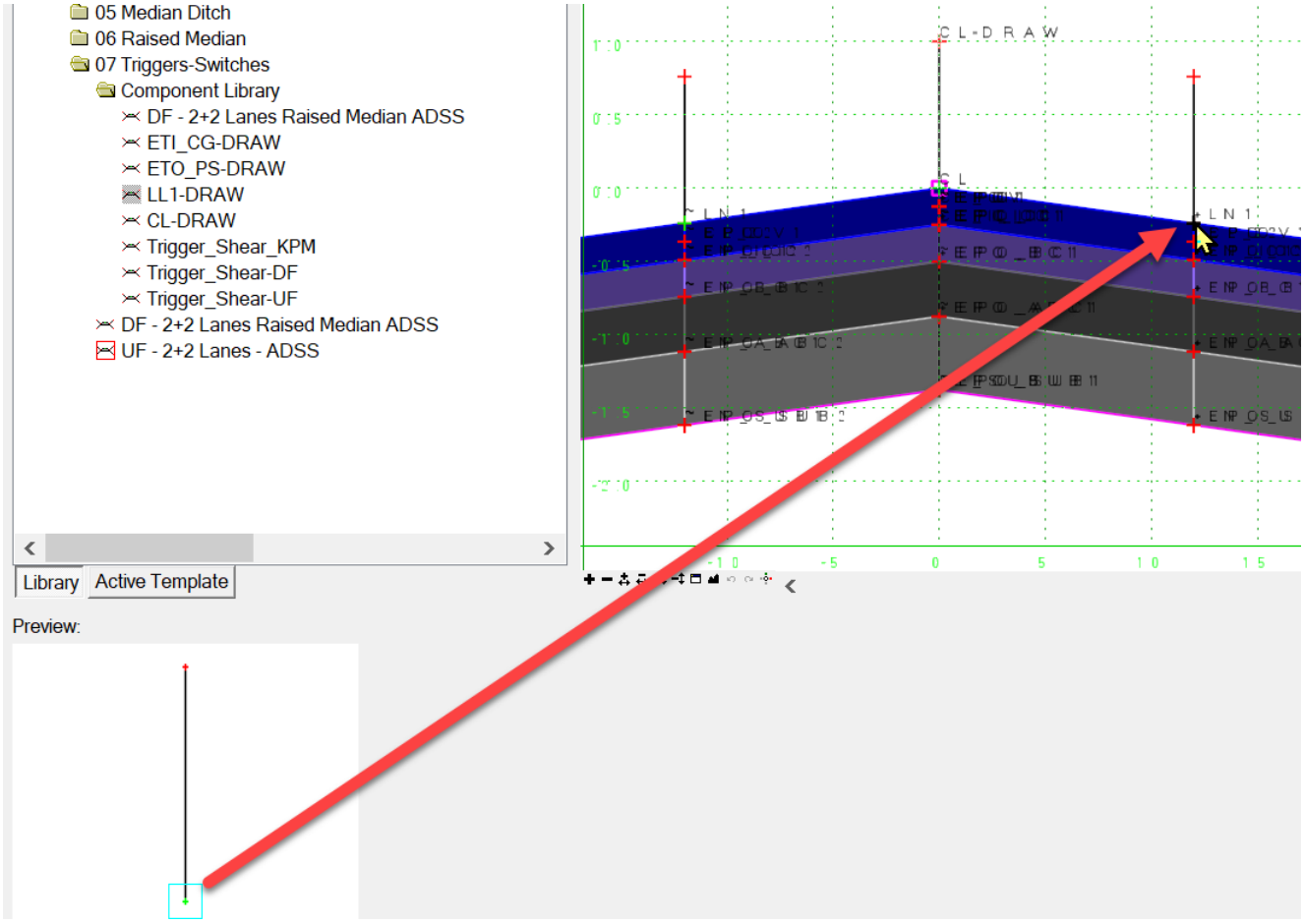


7. In the **Component Library** folder, select the **CL-Draw** component and drag it over to the active template screen. **Mirror** checked off place it at the **CL** point.

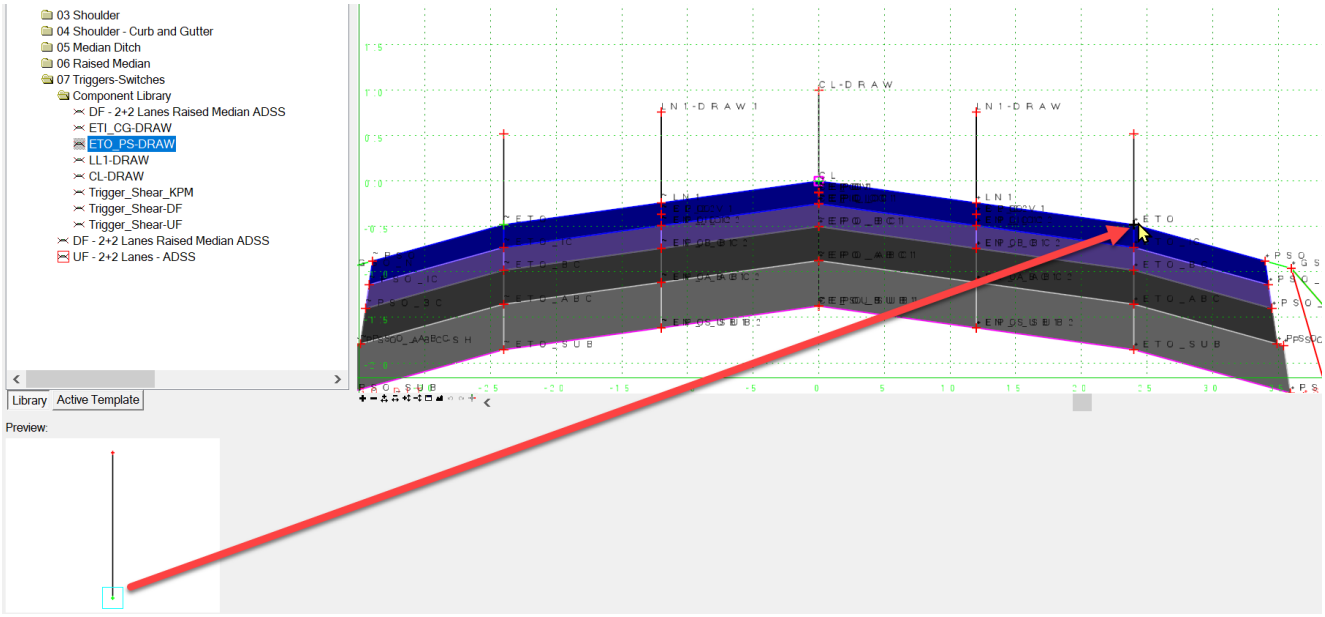


8. Turn on **Apply Affixes**.

9. In the **Component Library** folder, select the **LL1-Draw** component and drag it over to the active template screen. **Mirror** checked on place it at the **+LN1** point.

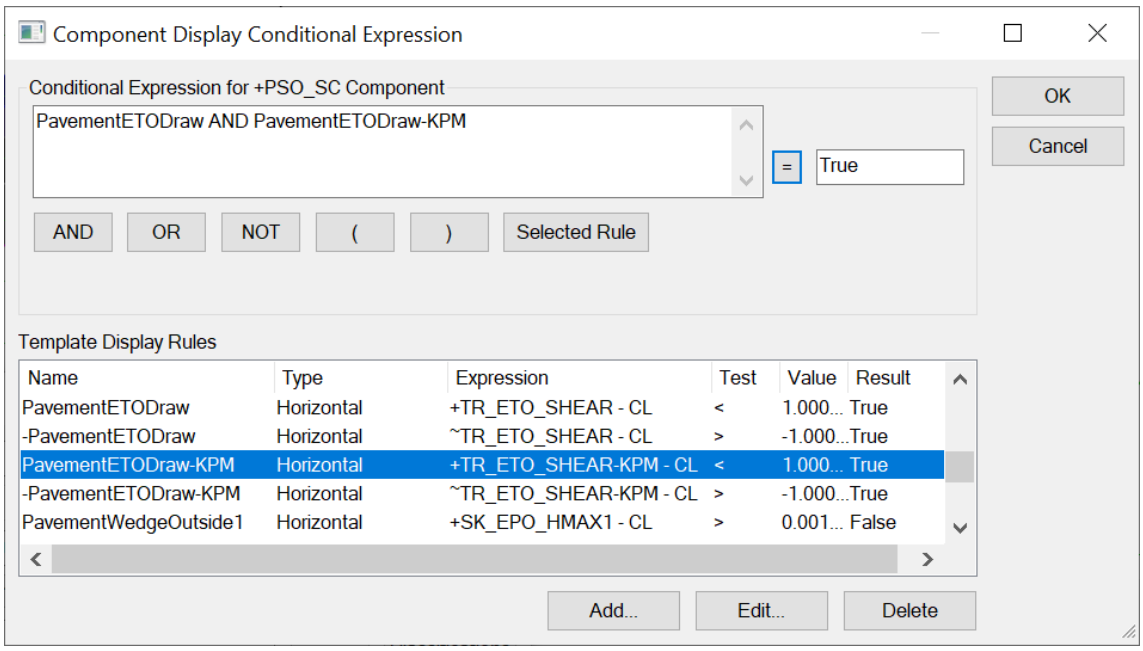


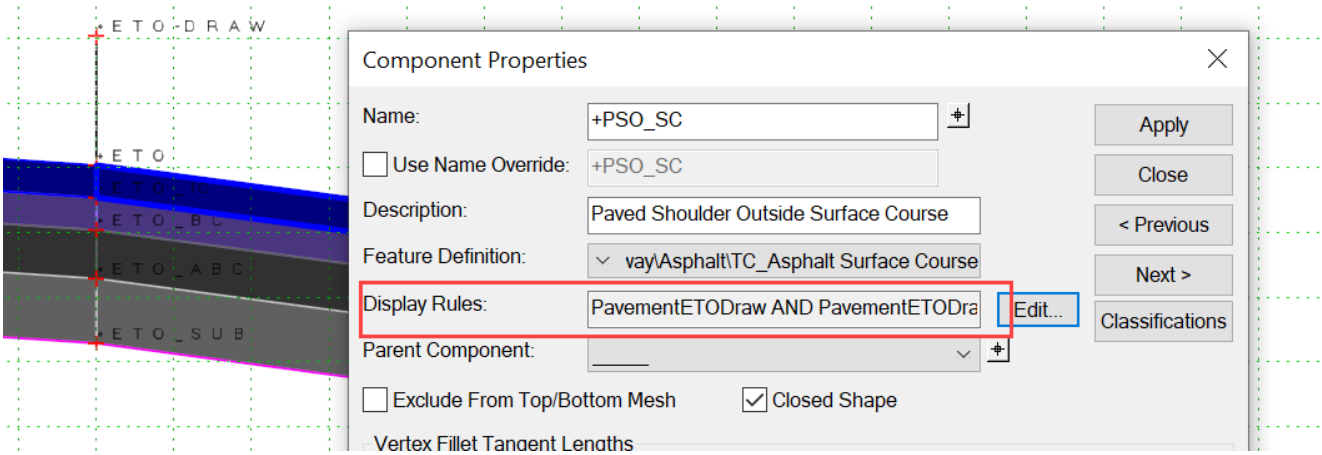
10. In the **Component Library** folder, select the **ETO_PS-Draw** component and drag it over to the active template screen. **Mirror** checked on place it at the **+ETO** point.



With both the triggers and components in place, assign the display rules to them.

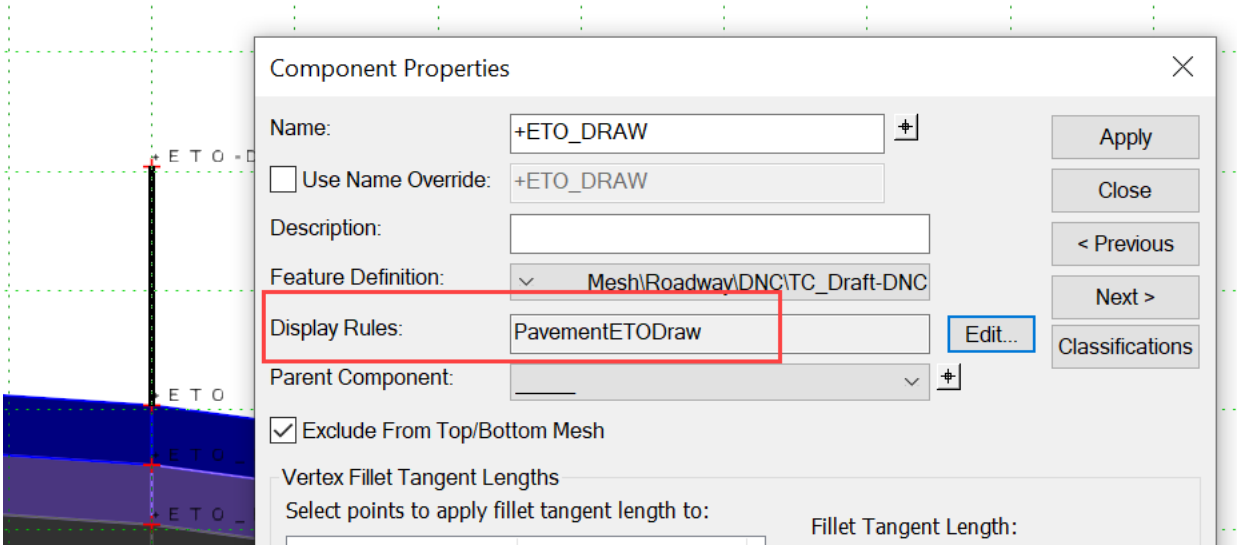
11. Turn off **Apply Affixes**.
12. Starting on the right paved shoulder component (first layer) assign the following display rules.



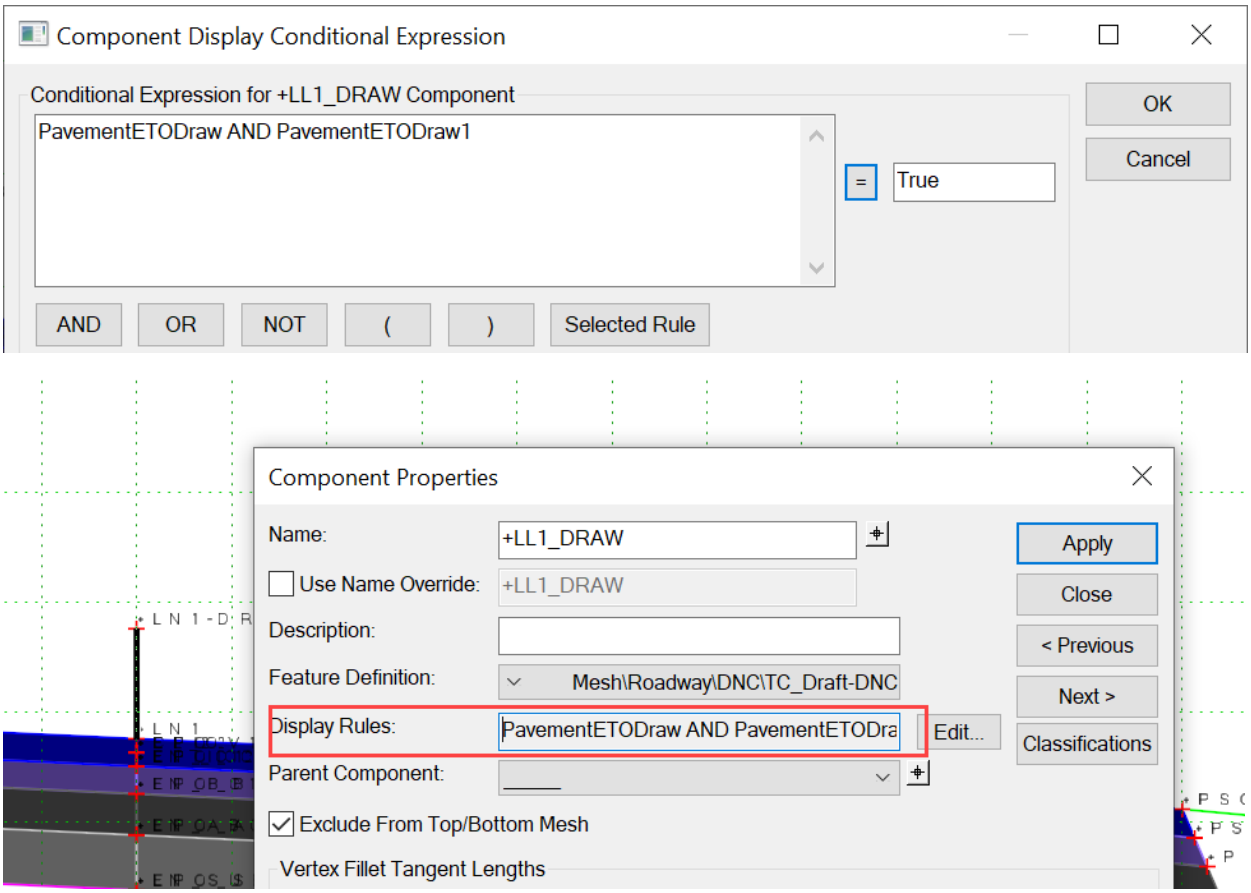


The Display Rules read if either trigger point is activated, the condition is False resulting in the components being sheared off from the EOT point. Since the surface course (first layer) of paved shoulder component is the Parent Component, every Child Components and end conditions are also turned off.

13. Edit the **+ETO_DRAW** component and assign it the following display rule.

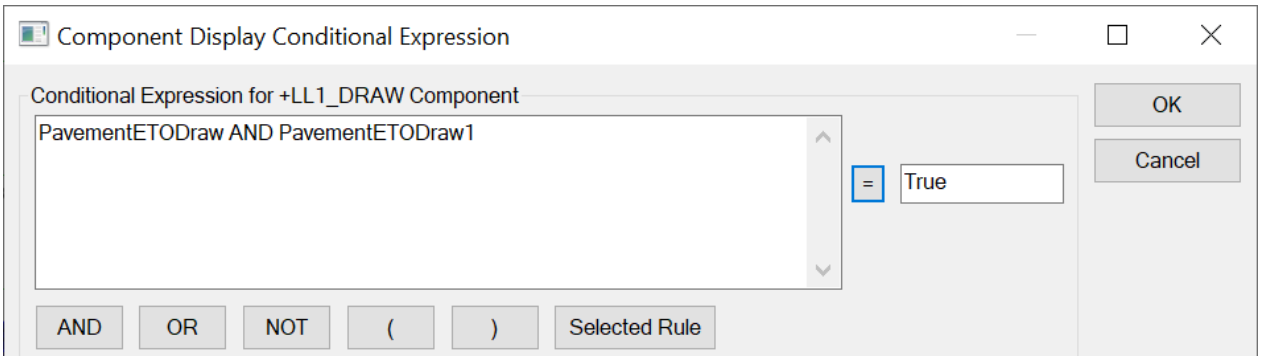


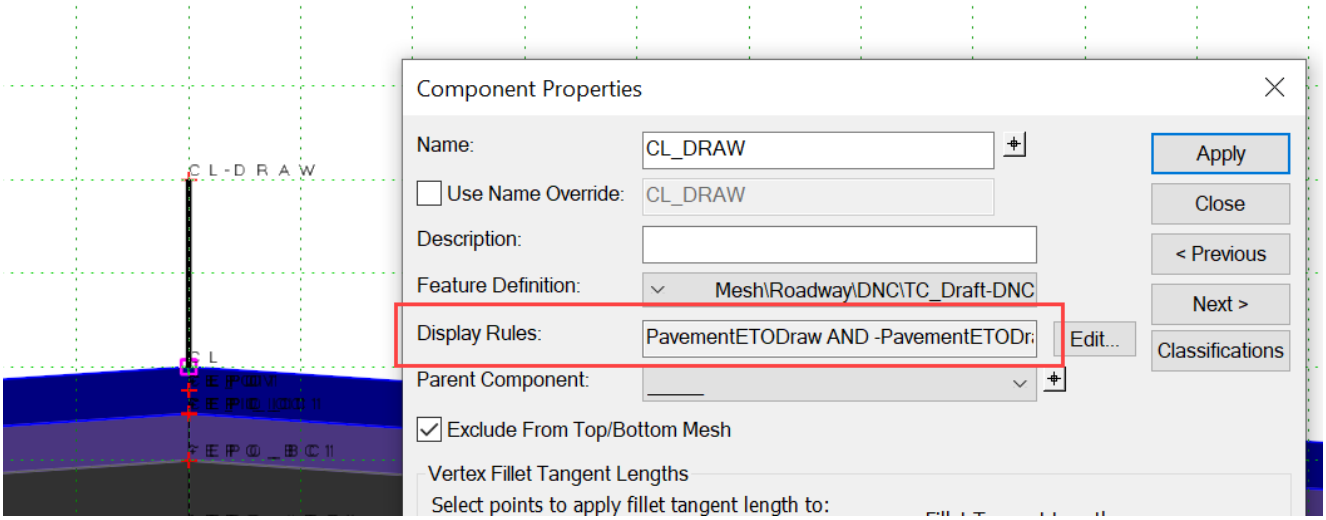
14. Edit the **+LL1_DRAW** component and assign it the following display rules.



The **Display Rules** read if either trigger point on either side is activated (not including KPM), the condition is **False** resulting in the Lane Line being turned off. Both sides of the Centerline are considered in case of a T-intersection.

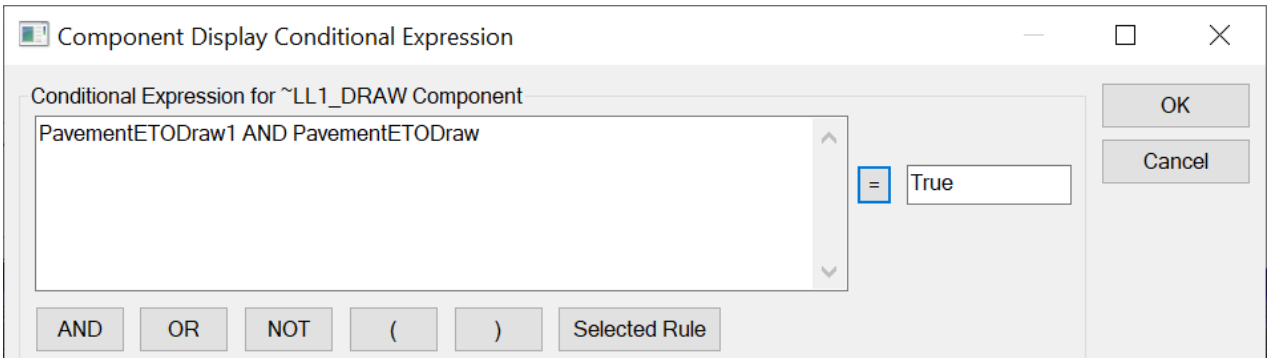
15. Edit the **CL_DRAW** component and assign it the following display rules.

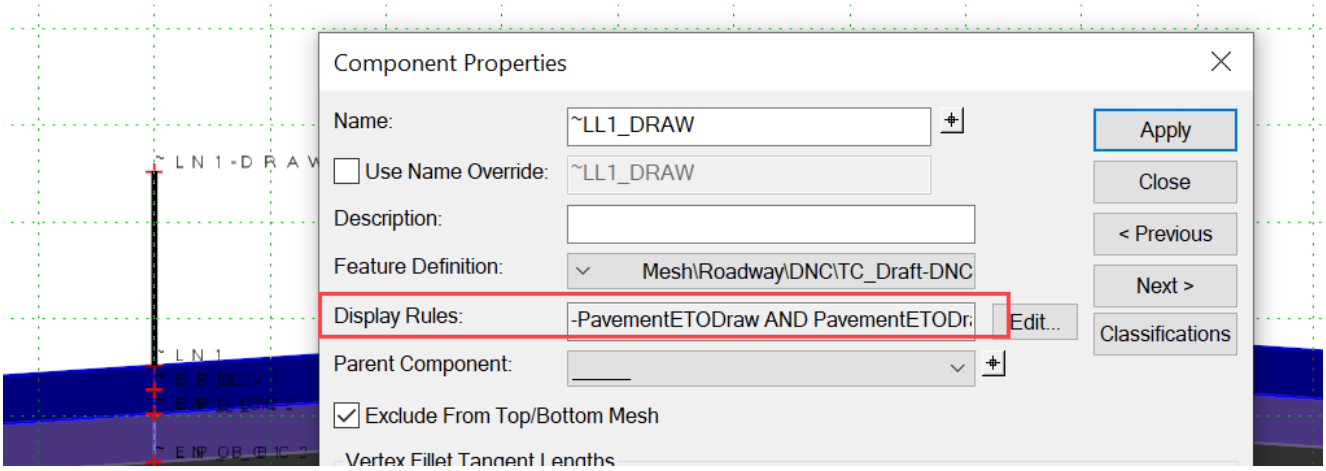




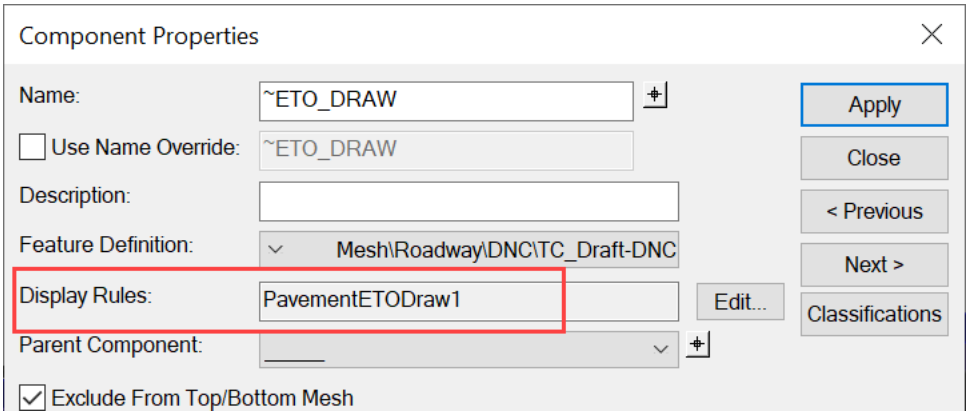
The **Display Rules** read if either trigger point on either side is activated (not including KPM), the condition is False resulting in the 3D Centerline pavement markings being turned off. Both sides of the Centerline are considered in case of a T-intersection.

16. Edit the **~LL1_DRAW** component and assign it the following display rules.

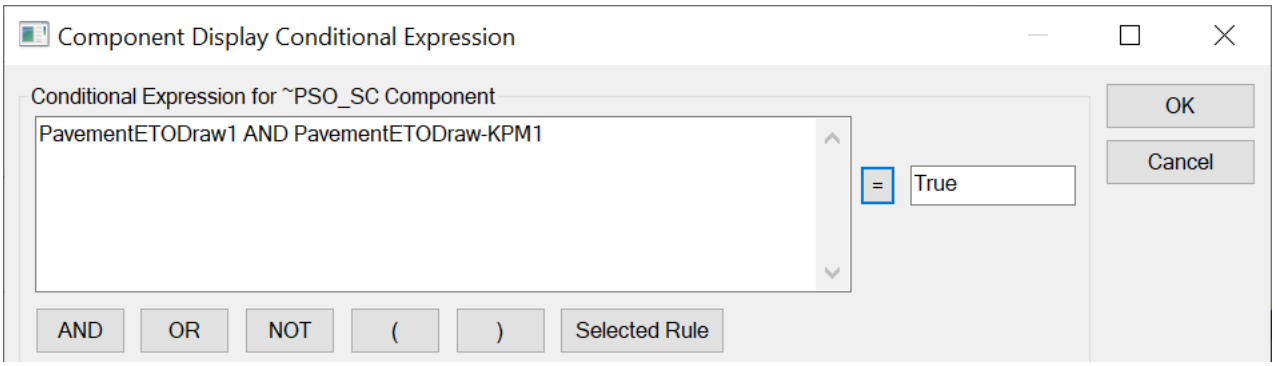


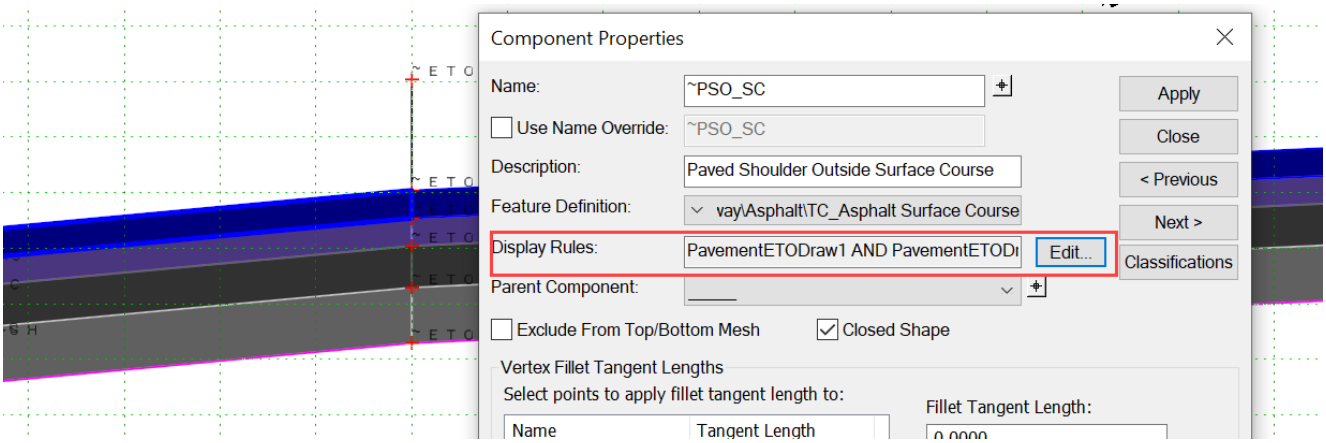


17. Edit the **~ETO_DRAW** component and assign it the following display rule.

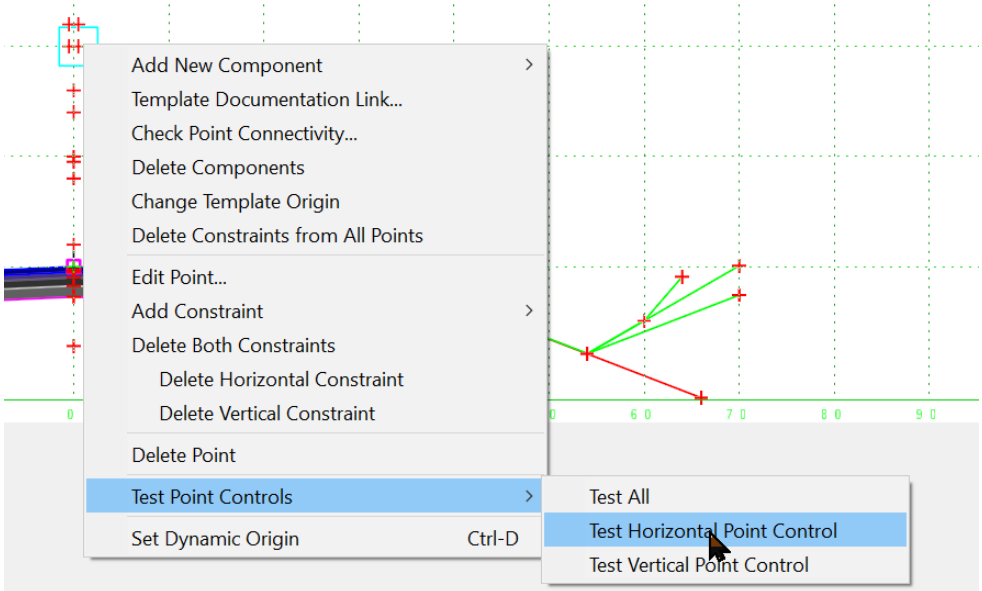


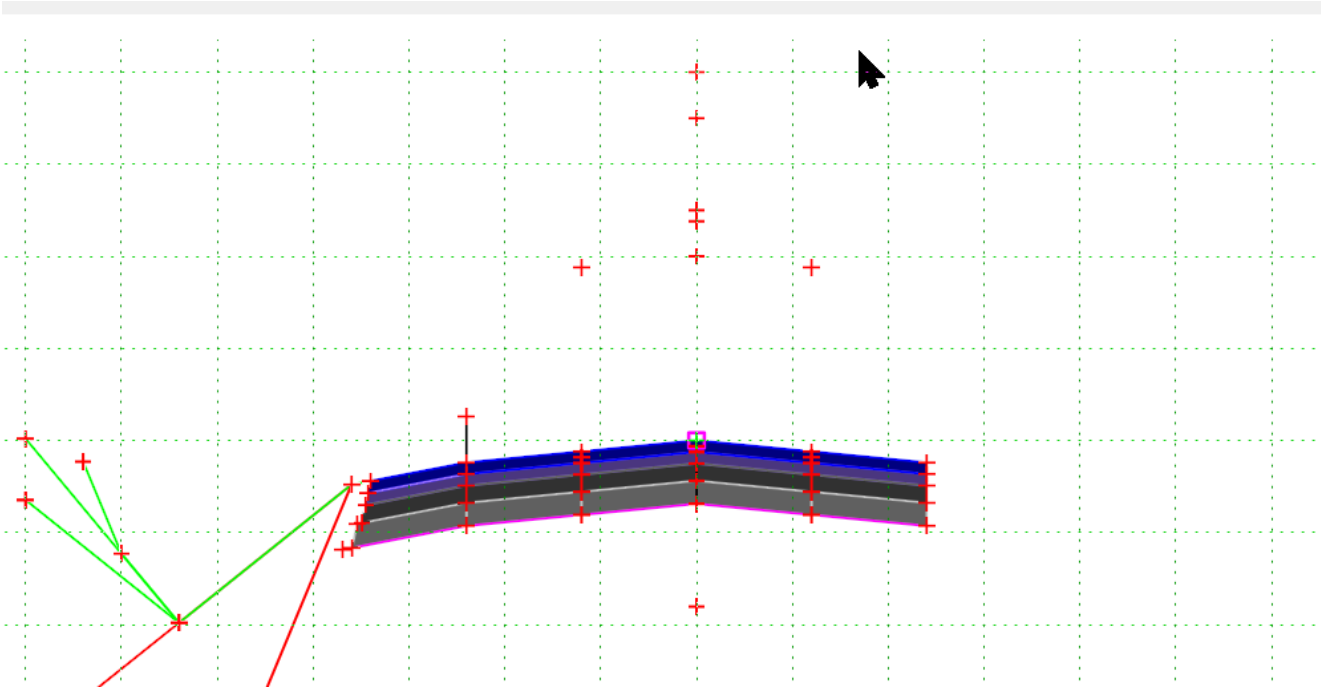
18. Edit the left paved shoulder component (first layer) and assign the following display rules.



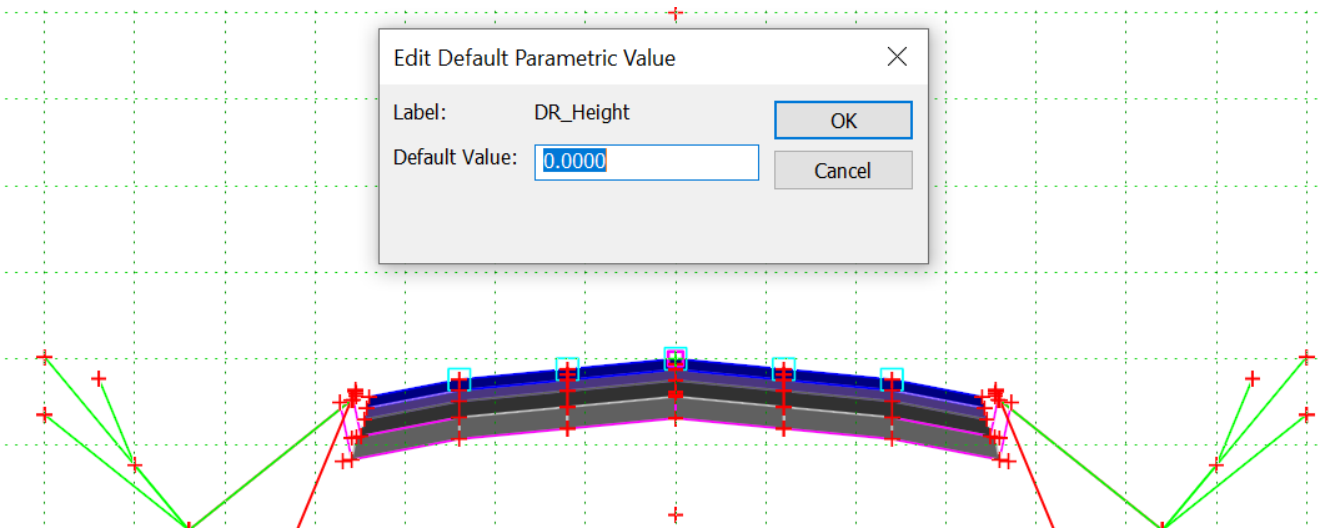


19. Test the template by moving one (1) of the four (4) trigger points horizontally (**Test Point Control >>> Test Horizontal Point Control**).





20. Last step in the trigger exercise is to change the Parametric Constraint Label DR_Height from "1" to "0" (zero).

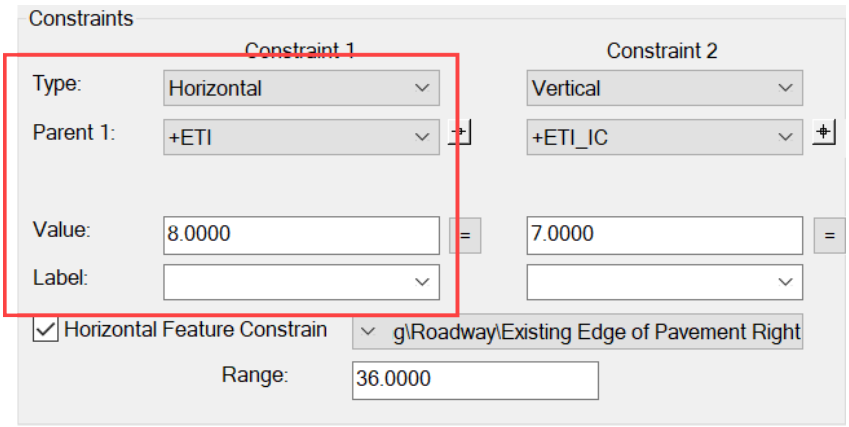


21. **Save** template.

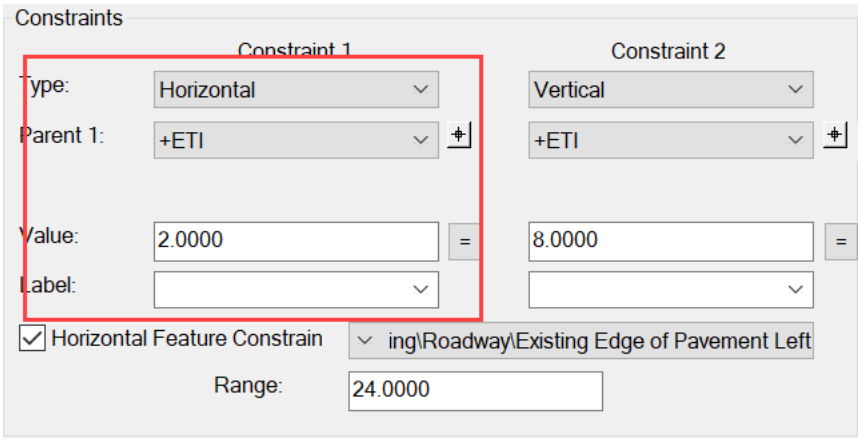
Switches– Median Crossover Pavement

22. Continuing the exercise with switches, in the Template Library **Open** the **03 Templates\07 Triggers-Switches** folder and select the **DF - 2+2 Lanes Raised Median ADSS** template.
23. Prior to dragging over the median crossover pavement components, move the four (4) seek existing EOP points under the ETI point. This is to avoid the prompts when merging multiple points in the same location.

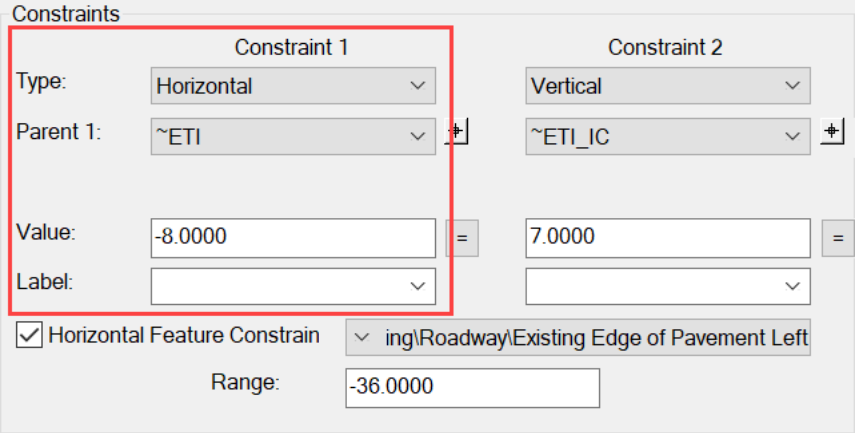
+SK_EPO



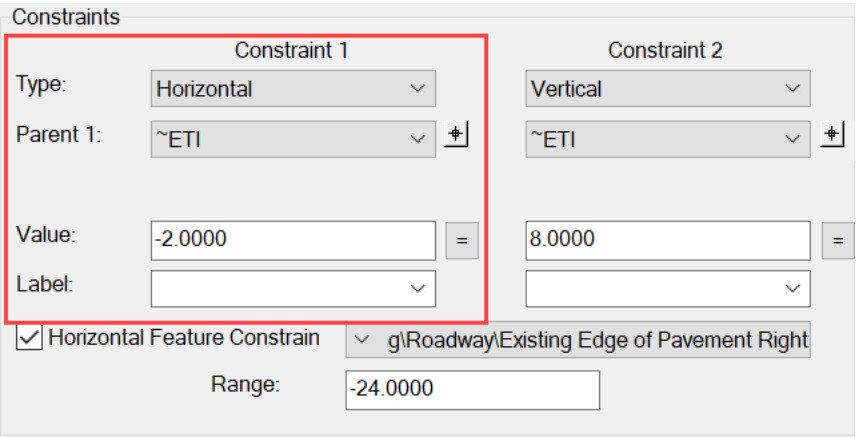
+SK_EPI

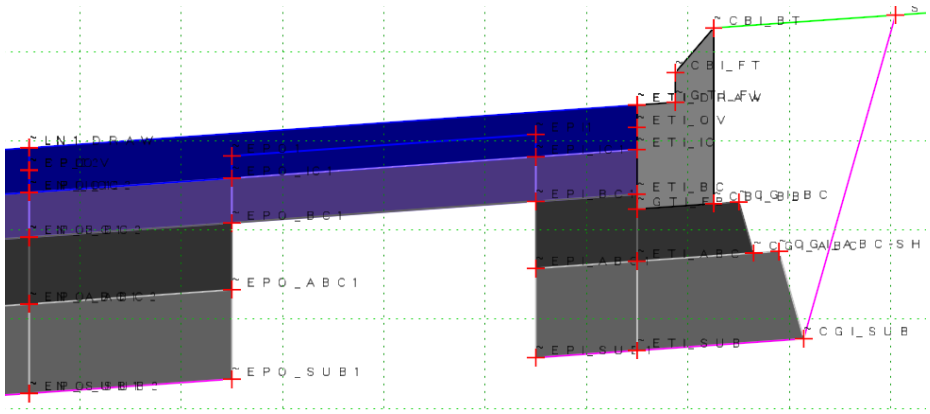


~SK_EPO

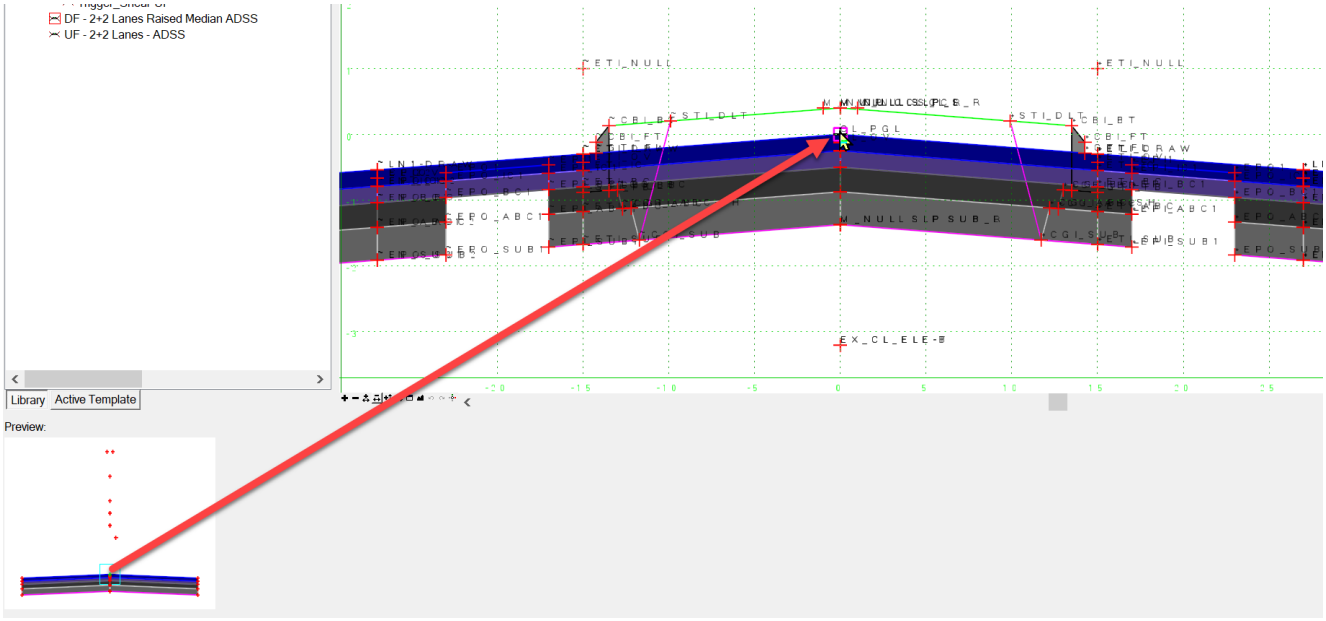

Constraints
Constraint 1
Constraint 2
Type:
Horizontal
Vertical
Parent 1:
~ETI
~ETI_IC
Value:
-8.0000
7.0000
Label:
Horizontal Feature Constrain
ing\Roadway\Existing Edge of Pavement Left
Range:
-36.0000

~SK_EPI

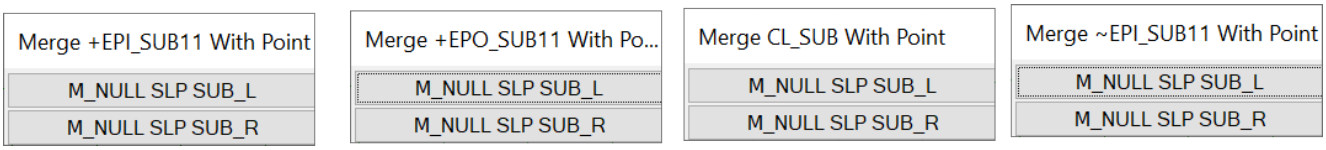

Constraints
Constraint 1
Constraint 2
Type:
Horizontal
Vertical
Parent 1:
~ETI
~ETI
Value:
-2.0000
8.0000
Label:
Horizontal Feature Constrain
g\Roadway\Existing Edge of Pavement Right
Range:
-24.0000


~EPO_ABC1
~EPO_SUB1
~CGI_SUB

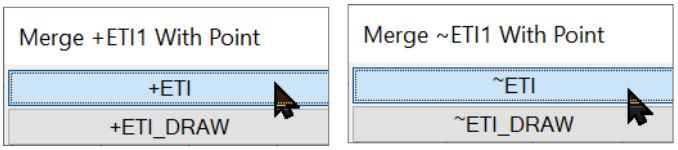
24. In the **Component Library** folder, select the **Median Crossover** component and drag it over to the active template screen. **Mirror** checked off place it at the **CL_PGL** point.



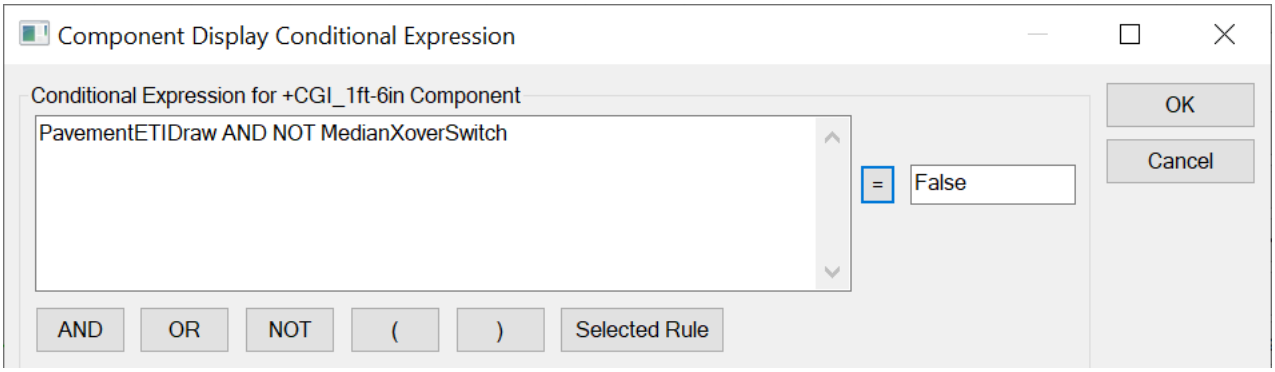
There are going to be several prompts to merge points in the same location. Do not merge any points with “NULL” in the name. By hitting the Esc key on the keyboard during the prompts, it will cancel the merging of these points. These are the point not merged or delete (Esc).

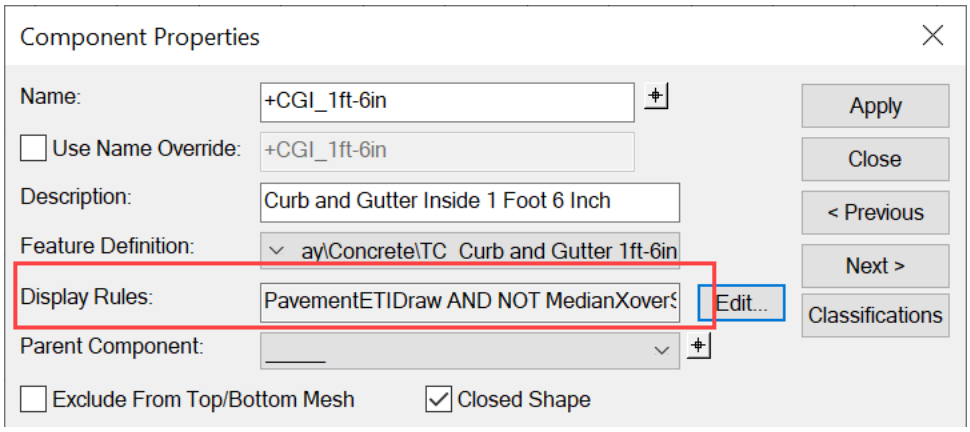


These are the two (2) points that required merging. Select the point without “DRAW” to merge with.

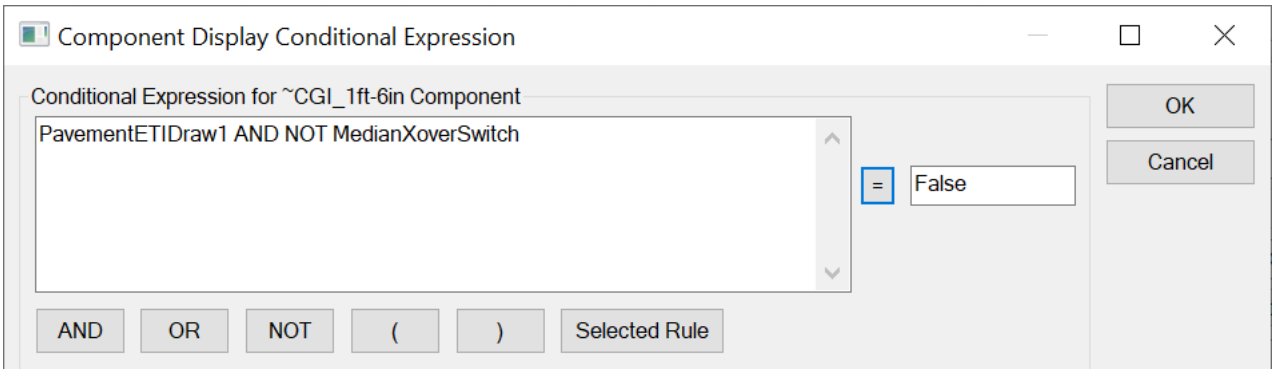


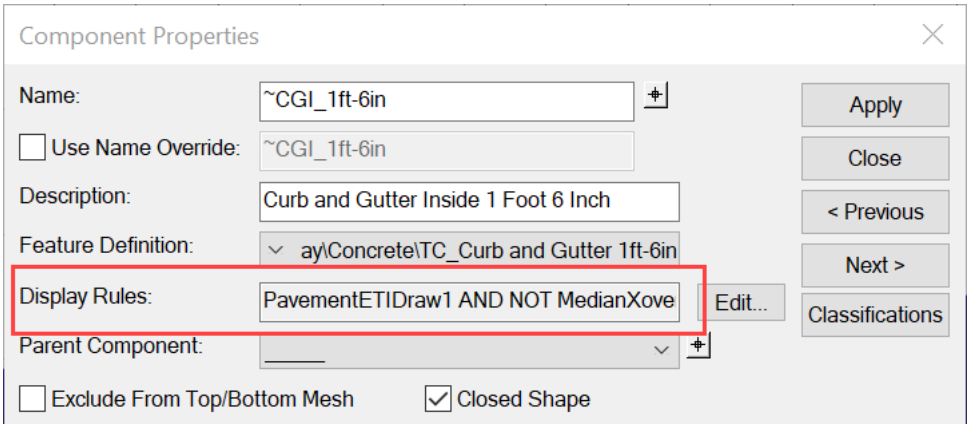
25. To turn off the regular median components when the median switch is turned on, edit the right inside median curb component and assign it the following display rule.



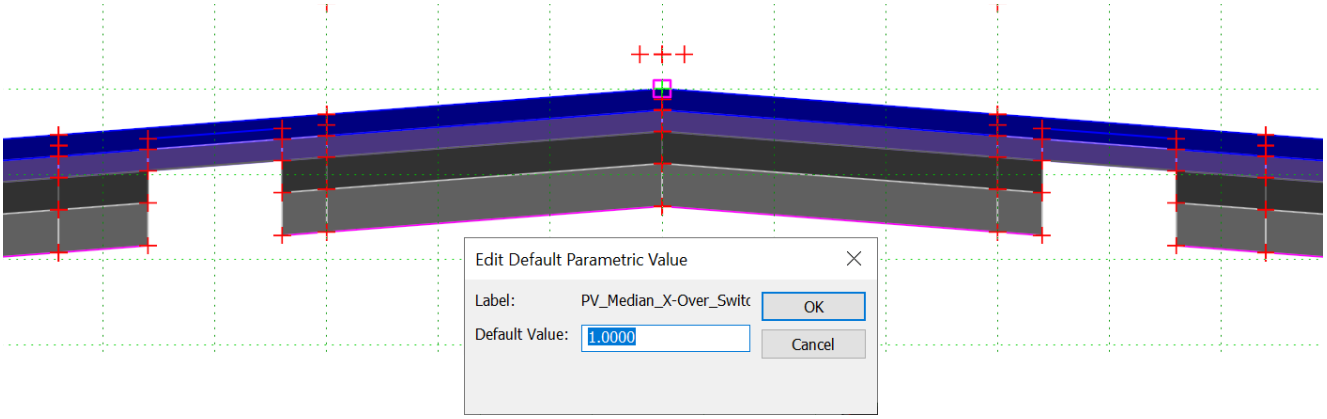


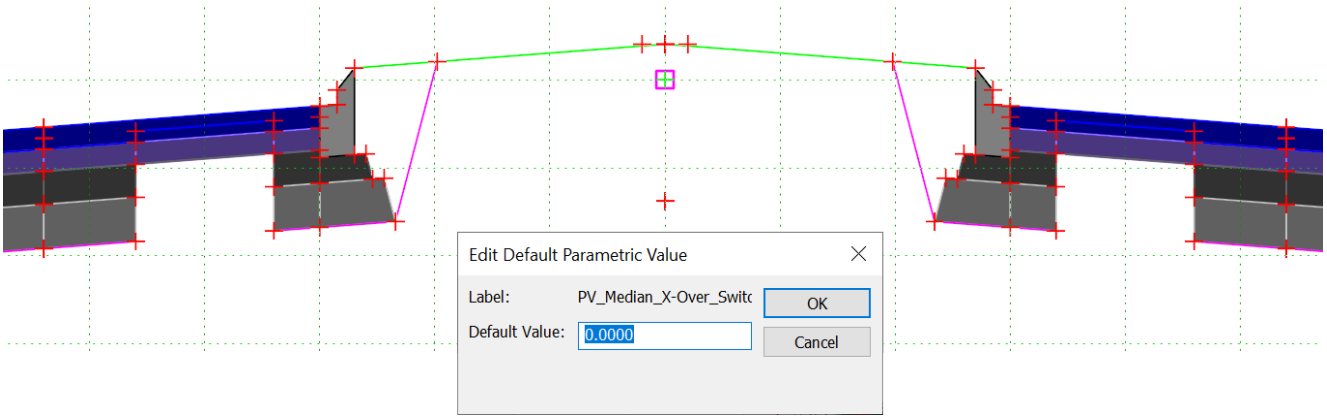
26. Edit the left inside median curb component and assign it the following display rule.



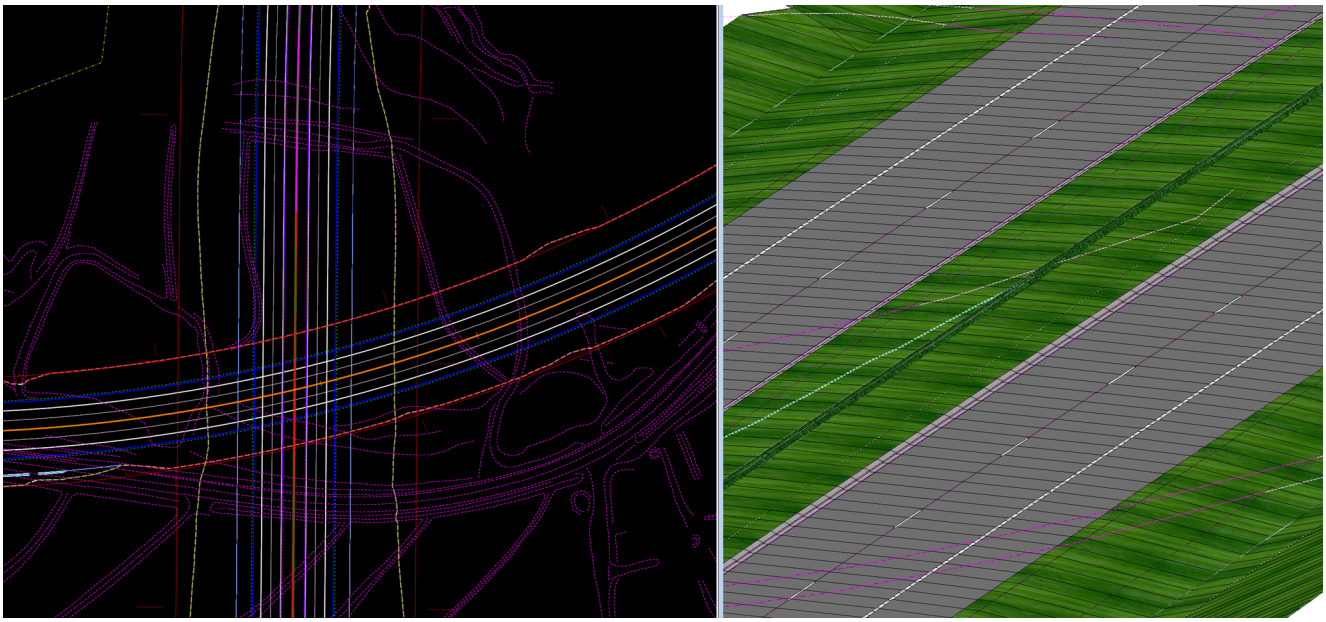


27. Test the template by changing the value of the Parametric Constraint Label **PV_Median_X-Over_Switch** from **1** (one) in its On state to **0** (zero) in its Off default state. Alternate between the **Library** and **Active Template** tabs to see the results of the parametric constraint change.





28. The final step is to move the four (4) seek existing EOP points horizontally back underneath the **+ETI** and **~ETI** points.
29. **Save** template.
30. Test the two (2) templates created in this exercise by creating a corridor. Use the raised median template for the mainline **-L-** and the undivided facility paved shoulder template on **-Y11- (**minor road).



Appendix A – Point Name

CENTERLINE (CL)		
CL	CENTERLINE	Top of finished grade
CL_DRAW	CENTERLINE DRAW	Draw 2D plan graphics and 3D pavement markings.
CL_IC	CENTERLINE INTERMEDIATE COURSE	Top of intermediate course
CL_BC	CENTERLINE BASE COURSE	Top of base course
CL_ABC	CENTERLINE AGGREGATE BASE COURSE	Top of aggregate base course (ABC)
CL_SUB	CENTERLINE SUBGRADE	On subgrade line/bottom of ABC

LANE LINE (LL)		
LL#	LANE LINE NUMBER	# indicates lane from centerline
LL#-DRAW	LANE LINE DRAW	Draw 2D plan graphics and 3D pavement markings.
LL#_IC	LANE LINE INTERMEDIATE COURSE	Top of intermediate course
LL#_BC	LANE LINE BASE COURSE	Top of base course
LL#_ABC	LANE LINE AGGREGATE BASE COURSE	Top of aggregate base course (ABC)
LL#_SUB	LANE LINE SUBGRADE	On subgrade line/bottom of ABC

EDGE OF TRAVEL (EOT)		
ET(O or I)	EDGE OF TRAVEL OUTSIDE or INSIDE	ETO or ETI
ETO_DRAW	ETO DRAW	Draw 2D plan graphics and 3D pavement markings.
ETO_IC	ETO INTERMEDIATE COURSE	Top of intermediate course
ETO_BC	ETO BASE COURSE	Top of base course
ETO_ABC	ETO AGGREGATE BASE COURSE	Top of aggregate base course (ABC)
ETO_SUB	ETO SUBGRADE	On subgrade line/bottom of ABC

PAVED SHOULDER (PS)		
PS(O or I)	PAVED SHOULDER OUTSIDE or INSIDE	PSO or PSI
PSO_DRAW	PSO DRAW	Draw 2D plan graphics and 3D pavement markings.
PSO_IC	PSO INTERMEDIATE COURSE	Top of intermediate course
PSO_BC	PSO BASE COURSE	Top of base course
PSO_ABC	PSO AGGREGATE BASE COURSE	Top of aggregate base course (ABC)
PSO_SUB	PSO SUBGRADE	On subgrade line/bottom of ABC

PAVEMENT UNDERNEATH CURB AND GUTTER		
CG(O or I)	CURB AND GUTTER OUTSIDE or INSIDE	CGO or CGI
CGO_BC	CGO BASE COURSE	Top of base course
CGO_ABC	PSO AGGREGATE BASE COURSE	Top of aggregate base course (ABC)
CGO_ABC-SH	PSO AGGREGATE BASE COURSE SHELF	Top of aggregate base course (ABC) shelf distance
CGO_SUB	CGO SUBGRADE	On subgrade line/bottom of ABC

GRASS (TURF) SHOULDER		
GS(O or I)_N	GRASS SHOULDER OUTSIDE or INSIDE NORMAL	GSO_N or GSI_N

CURB AND GUTTER BERM		
GB_N	GRASS BERM NORMAL	

SUBGRADE DAYLIGHT (DLT)	
SG(O or I)_DLT	SUBGRADE OUTSIDE or INSIDE DAYLIGHT (GRADED SHOULDER)
SGO_DLT1	When ABC layer is zeroed out for graded shoulder
SGO_DLT2	SUBGRADE OUTSIDE DAYLIGHT (TRENCHED SECTION)
SGO_DLT3	When ABC layer is zeroed out for trenched section

CURB AND GUTTER	
GTO_FL or GTI_FL	Gutter Outside or Inside Flow Line
CBO_FT or CBI_FT	Curb Outside or Inside Face Top
CBO_BT or CBI_BT	Curb Outside or Inside Back Top
CBO_BB or CBI_BB	Curb Outside or Inside Back Bottom
GTO_FB or GTI_FB	Gutter Outside or Inside Face Bottom

SIDEWALK	
SW_FT	Sidewalk Face Top
SW_BT	Sidewalk Back Top
SW_BB	Sidewalk Back Bottom
SW_FB	Sidewalk Face Bottom

END CONDITION	
DBF	Ditch Base Front
DBM	Ditch Base Middle
DBB	Ditch Base Back
C_HNG	Cut Hinge
C_6:1 and LOC_C_6:1	Cut and Limits of Construction Cut (Slope) 6:1
C_VAR and LOC_C_VAR	Cut and Limits of Construction Cut Variable (Slopes) 6:1 to 4:1
C_2:1-FADS and LOC_C_2:1	Cut and Limits of Construction Cut Freeway Arterial Design Slope 2:1
C_2:1 and LOC_C_2:1	Cut and Limits of Construction Cut (Slope) 2:1 – Local Design Slope
F_6:1 and LOC_F_6:1	Fill and Limits of Construction Fill (Slope) 6:1
F_VAR and LOC_F_VAR	Fill and Limits of Construction Fill Variable Slopes 6:1 to 2:1
F_2:1 and LOC_F_2:1	Fill and Limits of Construction Fill (Slope) 2:1

MEDIAN	
CL_PGL	Centerline Profile Grade Line
M_NULL BW_L or _R	Median Null Point Base Width Left or Right
M_DP	Median Ditch Point
M_NULL SLP_L or _R	Median Null Point Slope Left or Right
M_NULL MAXS_L or R	Median Null Point Maximum Slope Left or RIght
M_CP	Median Crown Point (Raised Median)

TRIGGERS AND SWITCHES	
SK_EP_L or _R	Seek Existing Pavement Left or Right
SK_EPI or EPO	Seek Existing Pavement Inside or Outside
SK_EPO_HMAX	Seek Existing Pavement Outside Horizontal Maximum
SK_EPI_HMIN	Seek Existing Pavement Inside Horizontal Minimum
TR_ETI_SHEAR	Trigger Edge of Travel Inside Shear
TR_ETO_SHEAR	Trigger Edge of Travel Outside Shear
TR_ETO_SHEAR-KPM	Trigger Edge of Travel Outside Keep Pavement Markings
SW_MD_PVMT	Switch Median Pavement

Appendix B – Parametric Constraint Label

Name	Default	Description
BM_Width	10	Width of Berm (C&G Section)
CGO_Width Gutter	2	Width of Gutter Outside C&G
DR_Height	0	Height of Draw component line
LN_AUX Inside Slope LT	-2%	Slope of Left Inside Auxiliary Lane
LN_AUX Inside Slope RT	2%	Slope of Right Inside Auxiliary Lane
LN_AUX Inside Width LT	0.0025	Width of Left Inside Auxiliary Lane
LN_AUX Inside Width RT	-0.0025	Width of Right Inside Auxilary Lane
LN_AUX Outside Slope LT	2%	Slope of Left Outside Auxiliary Lane
LN_AUX Outside Slope RT	-2%	Slope of Right Outside Auxiliary Lane
LN_Width	12	Width of each Lane
MD_Ditch Slope	16.6667%	Slope of Median Ditch
MD_Slope	1%	Slope of Median Crossover Pavement
MD_Tie Offset	15 or 30	Offset Distance of Median per side (1/2 Median Width)
PV_Depth ABC	-0.5	Depth of ABC Pavement Layer
PV_Depth Base Course	-0.375	Depth of Asphalt Base Course Pavement Layer
PV_Depth Intermediate Course	-0.25	Depth of Intermediate Course Pavement Layer
PV_Depth Overlay	-0.125	Depth of Surface Course Overlay
PV_Depth Surace Course Wedge	-0.125	Depth of Surface Course Wedge
PV_Depth Surface Course	-0.25	Depth of Surface Course Pavement Layer
PV_Median_X_Over_Switch	0	Switch to turn on (1) or off (0) median pavement
PV_Shear Inside LT	-0.5	Switch to shear (< -1) at EOT Left Inside
PV_Shear Inside RT	0.5	Switch to shear (> 1) at EOT Right Inside
PV_Shear Outside LT	-0.5	Switch to shear (< -1) at EOT Left Outside
PV_Shear Outside RT	0.5	Switch to shear (> 1) at EOT Right Outside
PVI_Width PS Shelf ABC	-0.5	Width of Shelf ABC Inside Paved Shoulder
PVO_Width PS Shelf ABC	0.5	Width of Shelf ABC Outside Paved Shoulder
SHI_Width Normal	-10	Width of Normal Shoulder Inside
SHI_Width Paved	-4	Width of Paved Shoulder Inside
SHO_Width Normal	8 or 12	Width of Normal Shoulder Outside
SHO_Width Paved	4 or 10	Width of Paved Shoulder Outside
SS_Slope Cut	50%	Maximum (Steepest) Cut Slope
SS_Slope Fill	-50%	Maximum (steepest) Fill Slope
SS_Width Ditch	12 or 18	Width of Roadway Ditch (Front Slope)