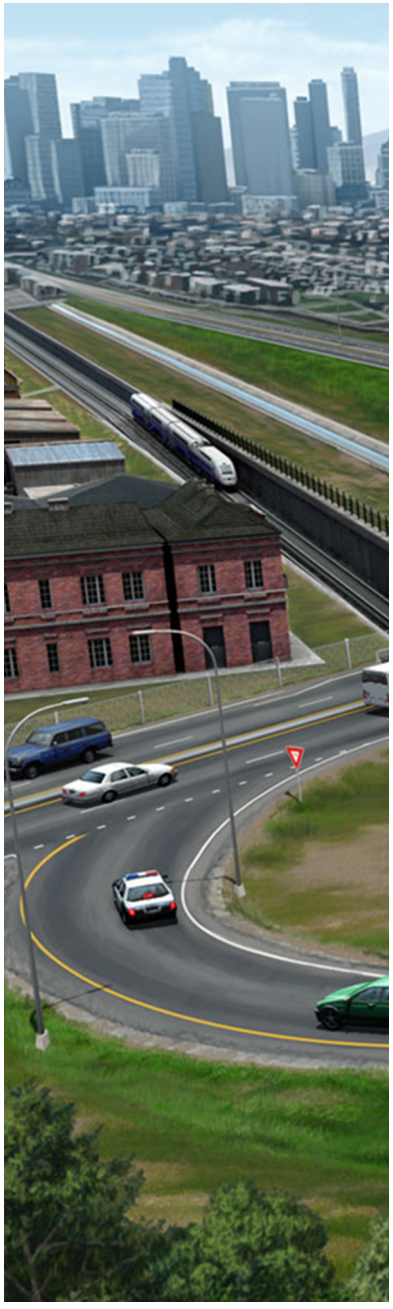


Defining Template End Conditions

This course is for the **2021 Release 1** *version of:*

OpenSite Designer CONNECT Edition

OpenRoads Designer CONNECT Edition

OpenRail Designer CONNECT Edition

About this Practice Workbook...

- This workbook is designed for use in Live instructor-led training and for OnDemand self study. OnDemand videos for this course are available on the *LEARNserver* and through *CONNECT Advisor*.
- This PDF file includes bookmarks providing an overview of the document. Click on a bookmark to quickly jump to any section in the file.
- Both Imperial and Metric files are included in the dataset. Throughout this practice workbook Imperial values are specified first and the metric values second with the metric values enclosed in square brackets. For example: *12.0' [3.4m]*.
- This course workbook uses the *Training and Examples* WorkSpace and the *Training-Imperial* or *Training-Metric* WorkSet delivered with the software.
- The terms "Left-click", "Click", "Select" and "Data" are used interchangeably to represent pressing the left mouse button. The terms "Right-click" and "Reset" are also used interchangeably to represent pressing the right mouse button. If your mouse buttons are assigned differently, such as for left-handed use, you will need to adjust accordingly.

Have a Question? Need Help?

If you have questions while taking this course, search in *CONNECT Advisor* for related courses and topics. You can also submit questions to the Civil Design Forum on Bentley Communities where peers and Bentley subject matter experts are available to help.

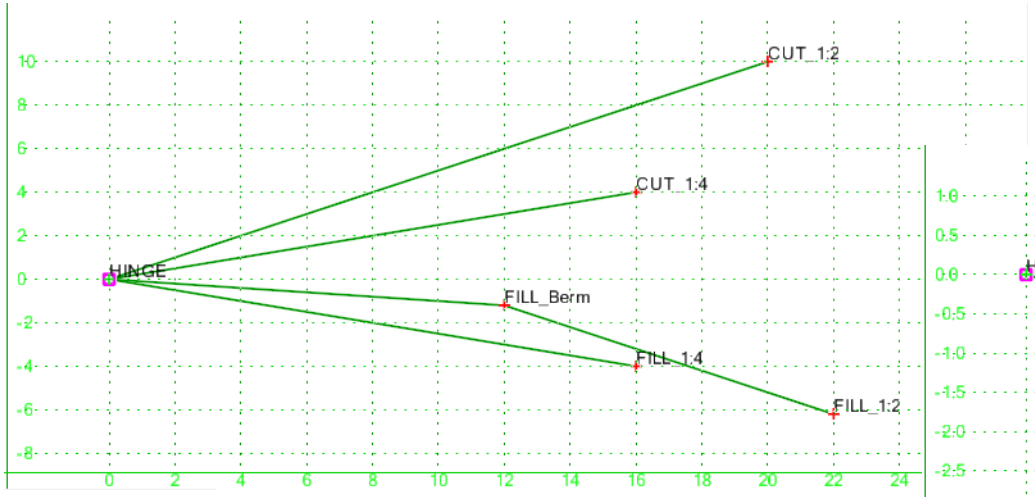
Bentley Institute

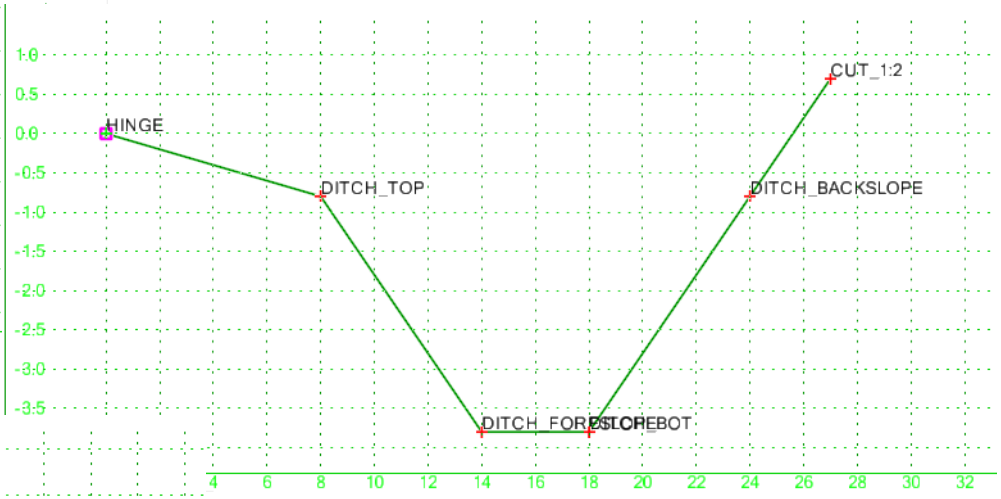
Edition: **05-01**

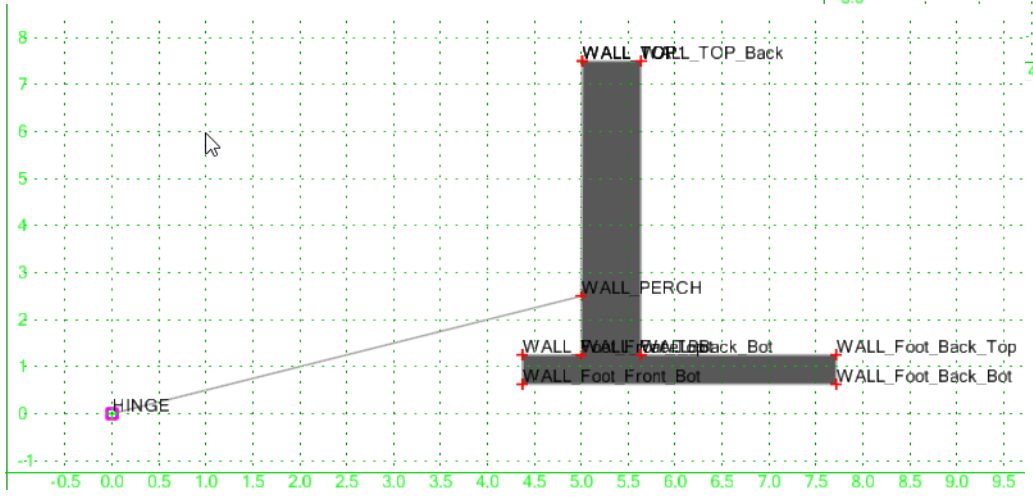
Course Level: **Intermediate**

Course Overview

This course teaches you how to create template components (called end conditions) for various side slope design requirements. This training focuses from the template hinge point to the target intercept point (i.e. existing ground) for both cut and fill conditions. In the following exercises, you will create end conditions with multiple cut and fill slopes, ditches, and retaining walls as well as learn how to add horizontal constraints to target specific features like a right-of-way. You will also learn how to set priorities and test the end conditions to verify their functionality.





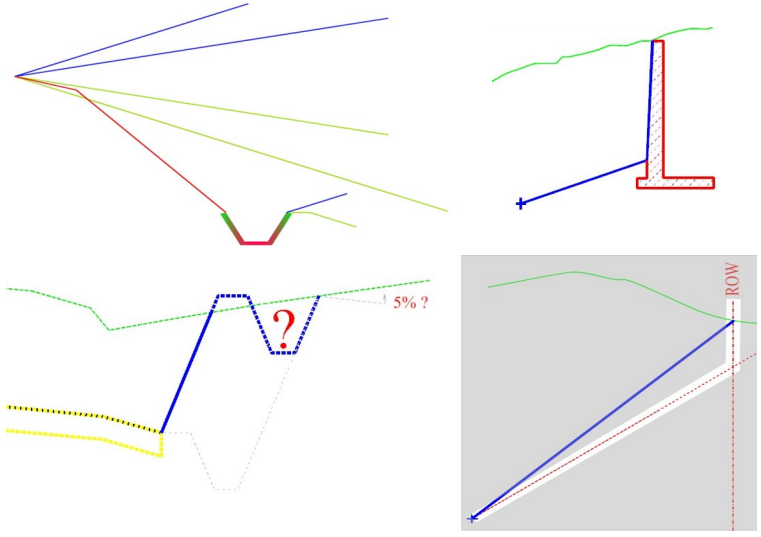


What is an End Condition?

In roadway engineering, the terms “Side Slope” or “Slope Treatment” are commonly used to describe the sloped grading outside the paved roadway. The solutions tend to be highly conditional and can vary dramatically depending on cut or fill heights. The functionality for “Side Slope” design is called an *End Condition* which provides a logical branching and sequencing capability to solve conditional side slope requirements.

Examples of End Conditions:

- Multiple Cut and Fill Slope Solutions. For example,
  - use a *1:6* cut slope if it can find daylight in less than *18’ [6 m]*; else
  - use a *1:4* slope if it can find daylight in less than *32’ [10 m]*; else
  - use a *1:3* slope if it can find daylight in less than *45’ [14 m]*; else
  - use a *1:2* slope to intersection.
- Retaining Walls that seek a horizontal wall location then seek the terrain.
- A Cut Solution that places a Brow Ditch if and only if the existing terrain drains towards the road.
- A Template that automatically forces the daylight line to the ROW limit if the steepest standard slope intrudes beyond the right-of-way line.



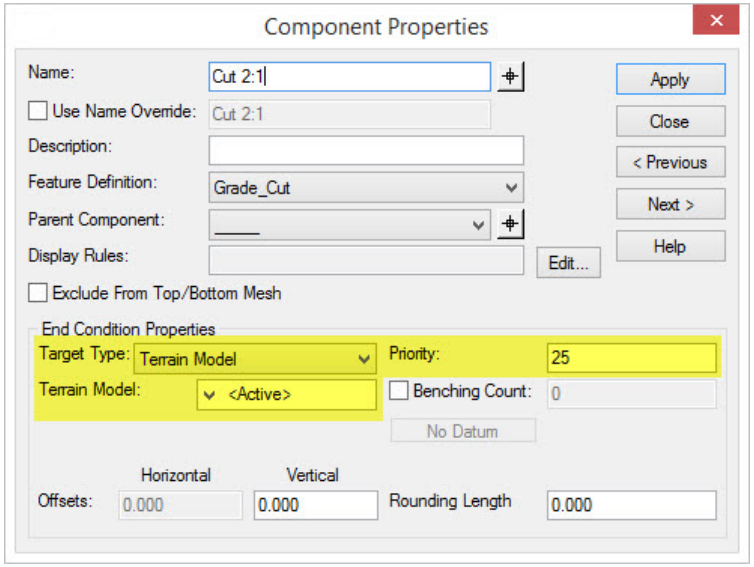
The unique capability of an End Condition Component is that it can seek a target, typically a terrain or feature. An End Condition Component is processed by starting with the first point until an intersection has been found or until all points (segments) have been processed. If the target is found, the End Condition is successful. The Success/Fail Status of an End Condition Component can be used as a logical switch for other End Condition Components to create complex solutions.

A Point in an End Condition Component has a toggle to Seek Intersection. Not every point needs to seek the target.

A point (segment) not seeking the target is **not** considered a failure. It is placed if success is found later in the component. For example, consider a ditch. A ditch foreslope generally does not seek the existing terrain, but does need to be placed if the ditch backslope seeks and intersects the ground.

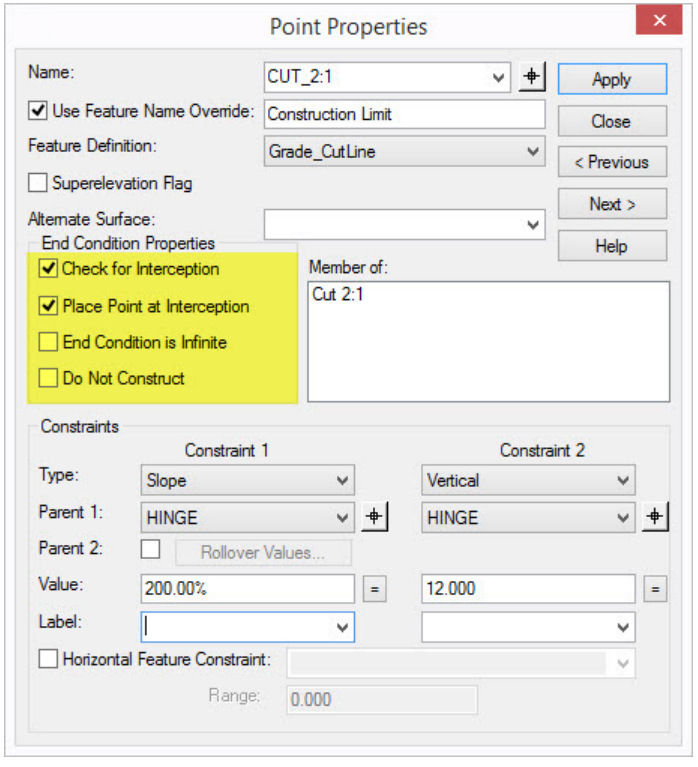
End Conditions Component Points use the same Constraints as other template points such as the Horizontal and Vertical constraints. These constraints limit how far an End Condition can search to intercept the target. If the target is not found within the search range, the end condition fails.

End Condition Component Properties:



- *Target* - Specifies what the end condition is seeking to intercept. End conditions can seek Terrain Models, specified Elevations, and horizontal and vertical geometry elements. Probably the most common target is the *Active Terrain* or Existing Ground. Targets can also be geometry that defines the right-of-way, a special ditch profile, a retaining wall, etc.
- *Priority* - Defines the processing order of multiple end conditions. Given two End Conditions starting from the same point, processing will start with the end condition that has the lowest priority number. For example, check 1:6 slope (Priority 1) before 1:4 slope (Priority 2).

End Condition Point Properties:



- *Check for Interception* - When set, the point will attempt to seek the target specified in the component properties. Sometimes segments such as a Ditch Bottom are not meant to intercept a target. It is meant to be placed only if a later segment finds the target. For these intermediary points, the Check for Intersection checkbox should be clear.
- *Place Point at Interception* - When set, and an interception is found along the line segment, the segment will terminate at the interception point. Otherwise, the segment will extend beyond the interception and a point created at the maximum length as specified by the point's constraints.
- *End Condition is Infinite* - When set, the end condition will attempt to seek a target infinitely until a target is found. Otherwise the component will only seek to the distance specified in the constraints.
- *Do Not Construct* - When set, the point will be solved as the end condition is processed but the point will not be used to draw any component segments. The component is drawn as if the point does not exist. For example, a fixed-width variable-slope segment can be constructed by not constructing the point middle point that sets the width.

Exercise 1 - Build and Test a Multiple Slope End Condition

Description

In this section, you will learn how to create a multi-slope End Condition targeting the active terrain model.

Skills Taught

- Create End Condition Components
- Define Fixed Segments
- Define Segments Targeting Existing Ground
- Set Priorities
- Test End Conditions

Open the Template Library

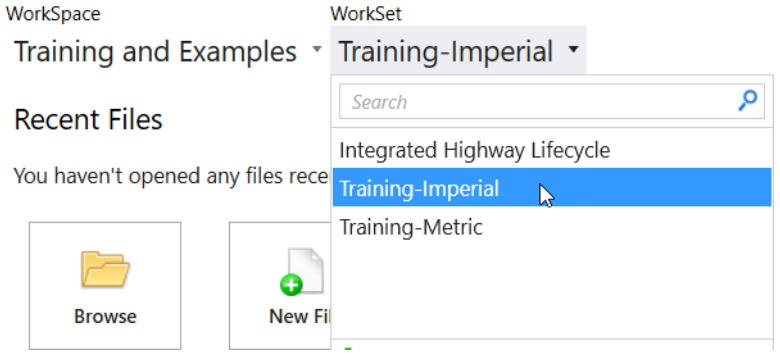
1. Start the software.
2. Set the WorkSpace and WorkSet.

   The WorkSpace and WorkSet define standards that are used by the software, and the ones used for this training are installed during the software installation.

   Typically, the WorkSpace contains organizational standards and the WorkSet contains project standards.

   a. Select **Training and Examples** from the *WorkSpace* drop-down menu.

   b. Select **Training-Imperial** *[Training-Metric]* from the *WorkSet* drop-down menu.



3. Open an existing file.

   a. Select **Browse**.

   b. Browse to *C:\Bentley Training\Defining Template End Condition* or other folder where you unzipped the dataset files.

   c. Select the file **Filename.dgn** *[Metric-Filename.dgn]* and click **Open**.

   Note: If you get a message stating “Incompatible Civil Data”, this is because the training files are “aligned” to OpenSite Designer. Clicking *Yes* will align the file to the software you are using (OpenRoads Designer or OpenRail Designer). This will have zero impact for the training courses. However, note that in production, upgrading the file will make the file read-only in OpenSite Designer. Full information is available at *Bentley Communities - Product Realignment*.

4. Verify or set the *Workflow* to **OpenRoads Modeling**.

5. Select the **Corridors > Create > Create Template** tool.

   The *Create Template* dialog opens, displaying the default Template Library. We will be using a different, course-specific library.

6. In the *Create Template* dialog, select **File > Open**.

7. *Browse to* and *select* the project template library in the class data folder: **Defining End Conditions.itl** *[Metric-Defining End Conditions.itl]*

   Note: When you open the software, the template library designated by the CIVIL_ROADWAY_TEMPLATE_LIBRARY variable is opened, NOT this course template library. If you exit the software, you will have to repeat the above steps to open the course library.

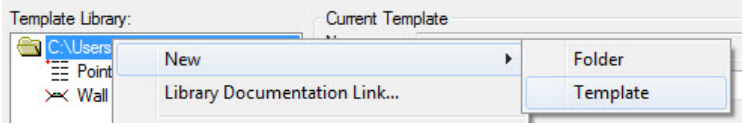
   In production, this is typically a Workset variable so that each project points to the proper agency or project template library.

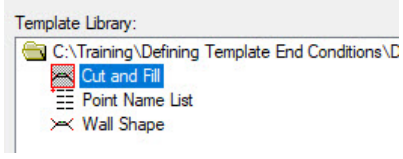
8. Select the *Dynamic Settings* dialog by selecting the icon or going to **Tools > Dynamic Settings**.

   NOTE: The icon may not be visible if there is not an active template.

9. Set the *X* and *Y Step* settings to **0.1** *[0.01]* and ensure that the *Apply Affixes* is unchecked.

10. Create a blank template named **Cut and Fill** by right-clicking on the template library object, and click **New > Template**.



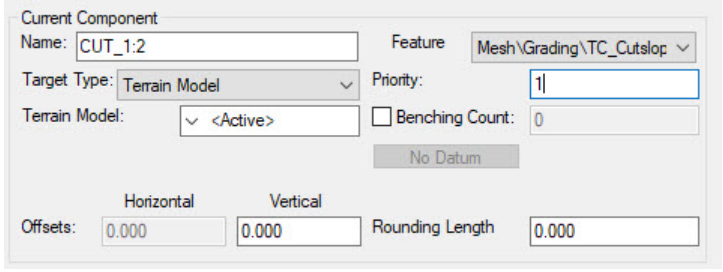


Create a 1:2 Cut Slope End Condition Component

Create an end condition component that seeks the active terrain (most often the existing ground) at a 1:2 slope in cut conditions.

---

1. *Right-click* in the Current Template *window* and select **Add New Component > End Condition**.
2. Define the end condition component properties with the following values:



   a. Type **CUT_1:2** for *Name*.

   b. Set the *Feature* to **Mesh\Grading\TC_Cutslope**.

   c. Set the *Target Type* to **Terrain Model**.

   d. Set the *Terrain Model* to **<Active>**.

   e. Set the *Priority* to **1**.

   We are guessing at this point that this cut solution will be our first priority. This value can be adjusted later as needed. When an end condition has a branch, a point from which two or more End Condition paths begin, the software evaluates each path in a prioritized sequence. If the cut slope has a lower Priority number it will be evaluated first.

3. Place the first point of the end condition component.

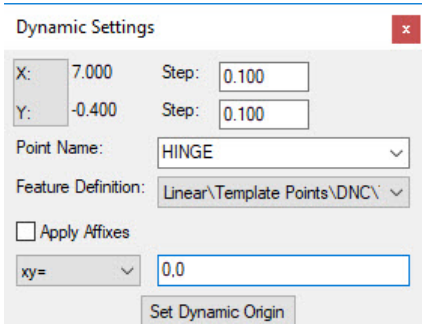
   a. In the *Dynamic Settings* dialog, ensure that the *Apply Affixes* box is clear.

   b. Select **HINGE** from the *Point Name* list.

   The *Feature Definition* will automatically be set to **TL_Draft-DNC**

   c. Select the Precision Input Mode of **xy=**

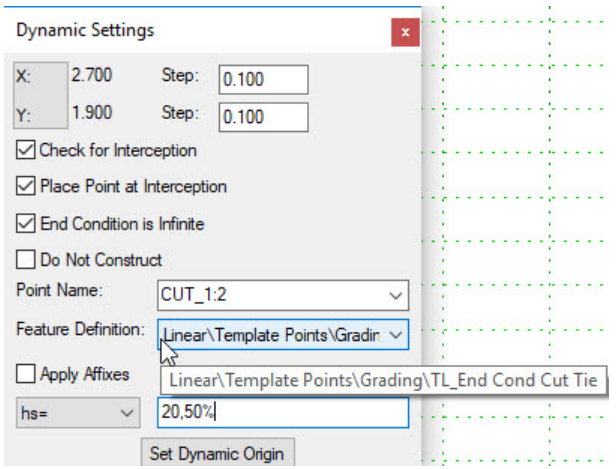
   d. Type **0,0** and click the **Enter** key.



   HINT: You can type in *xy=0,0* or pick *xy=* from the pick list and type in just *0,0*.

   The Hinge point is placed and the component is “drawn” to the cursor, awaiting the next point.

4. Place the second point of the end condition component. In the *Dynamic Settings* dialog:



   a. **Set** the *Check for Interception* box.

      This point will seek to intersect the Active Terrain target specified in the previous step.

   b. **Set** the *Place Point at Interception* box.

      This point will be created at the intersection with the target. It will not be extended to the full extent of the segment as defined by the constraints.

   c. **Set** the *End Condition is Infinite* box.

      This point will infinitely seek an intersection with the target. It will not be limited by the horizontal and vertical constraints.

   d. **Clear** the *Do Not Construct* box.

      This point will be constructed and included in the 3D model.

   e. Select a *Point Name* of **CUT** and edit the name to be **CUT_1:2.**

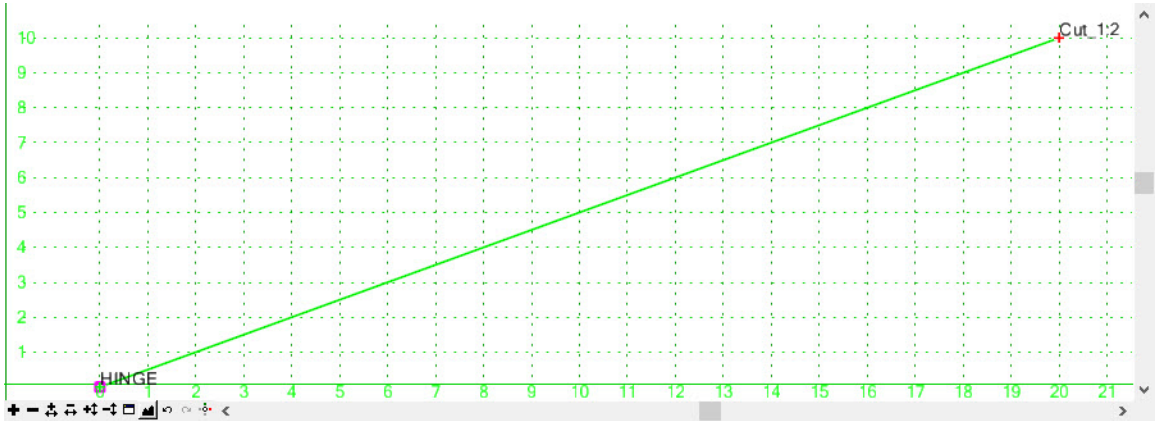
      The *Feature Definition* is automatically be set to **TL_End Cond Cut Tie** when the point name was selected from the list. Selecting the point name from the standards prior to editing the name to ensues the proper *Feature Definition* is selected.

   f. Select the *Precision Input Mode* of **hs=** to enter a horizontal distance and slope.

   g. Type **20,50%** *[5,50%]* and click the **Enter** key.

      This key-in has two constraints defined for the new point relative to the **HINGE** point. A Horizontal constraint at 20’ *[5m]* and a Slope constraint at 50%.

      The **CUT_1:2** point is placed and the component is “drawn” to the new cursor position, awaiting the next point.



h. **Fit** and adjust the *Current Template Window* if necessary.

i. *Right-click* in the *Current Template Window* and select **Finish**.

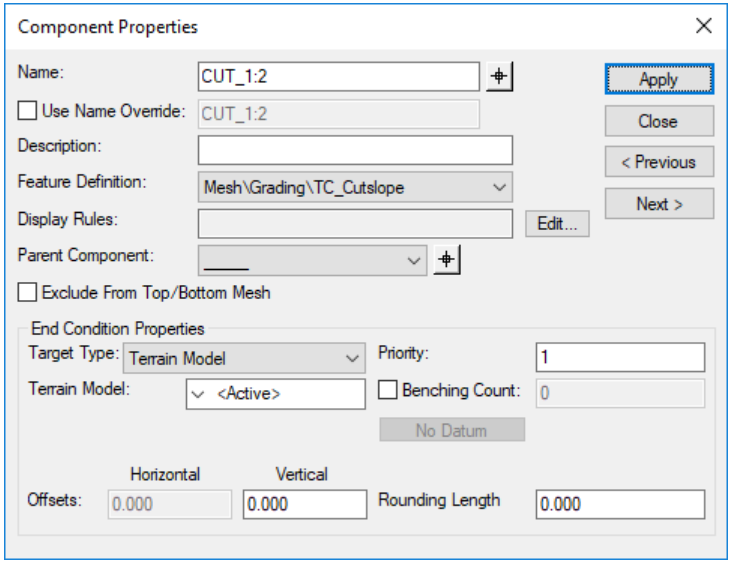
j. **Note:** Precision input values are entered in master units such as feet or meters. You can enter other units with the proper unit specification, such as 6” instead of 0.5.

5. Review the *Component Properties*

   a. *Double click* on the **CUT_1:2** component (green line) to open the Component Properties dialog.

   b. Review the *End Condition Properties* that define the **Target** and **Priority**.

   c. **Close** the *Component Properties* dialog.



6. Review the *Point Properties*

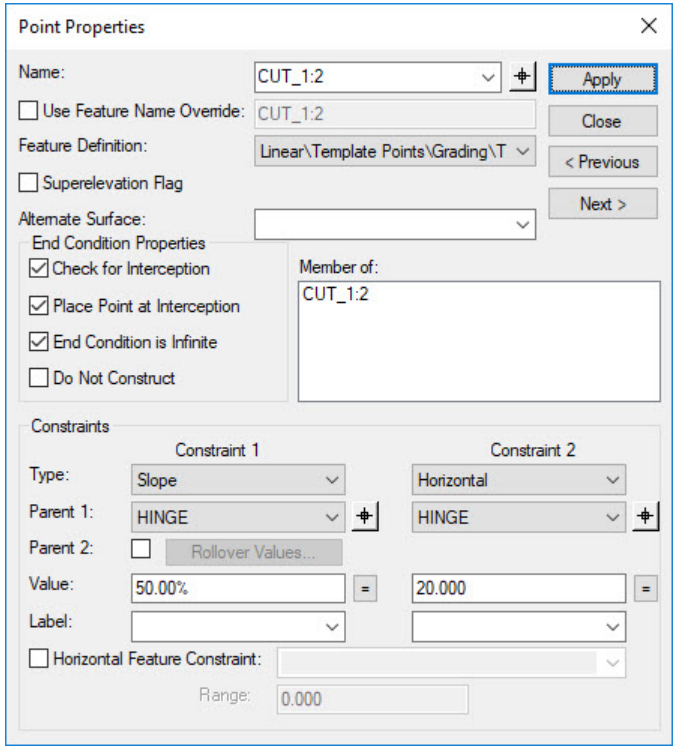
   a. *Double click* on the **CUT_1:2** point.

   b. Review the End Condition Properties.

   c. Review the Constraints that were defined by the *hs=20,50% [5,50%]* key in.

   d. **Close** the *Point Properties* dialog.

7. **Save** the template library.



Test the 1:2 Cut Slope End Condition

In this section, you will learn how to test the CUT_1:2 end condition to see how it will behave when it targets the active terrain model.

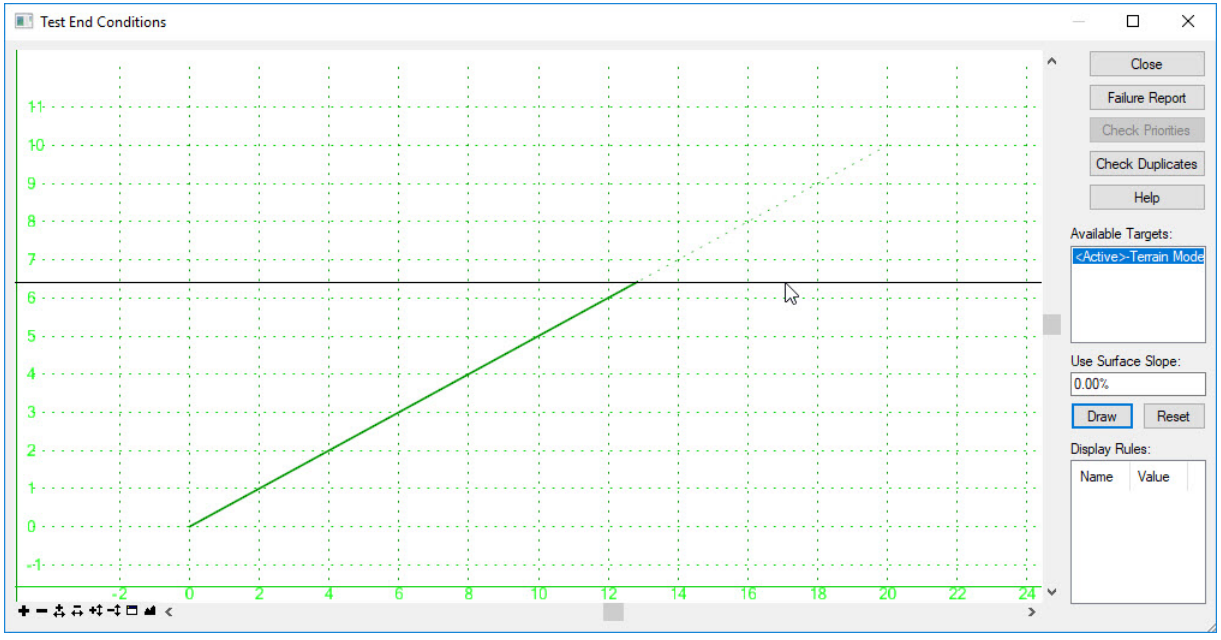
---

1. In the lower right portion of the *Create Template* dialog, *click* the **Test** button. The *Test End Conditions* window will appear.
2. In the *Available Targets* list, click **<Active>-Terrain Model**.
3. *Click* the **Draw** button.
4. Move the cursor vertically up and down in the *Current Template Window*. Notice how the cut slope solves to the target as you move your cursor. The horizontal line that the end condition solves to is the simulated active terrain model with a 0% surface slope.

   Sometimes you may want to test against a non-level surface. To adjust the target surface slope, type a value in the *Use Surface Slope* field, then click on the Draw button again.

5. Verify the template works when in all cut situations, but does not draw in fill.
6. **Close** the *Test End Conditions* dialog.

The *Test End Conditions* dialog is a robust tool to evaluate how End Conditions behave and is available during the design process. The design of the Test End Conditions dialog makes it easy to catch errors before they reveal themselves on a corridor.



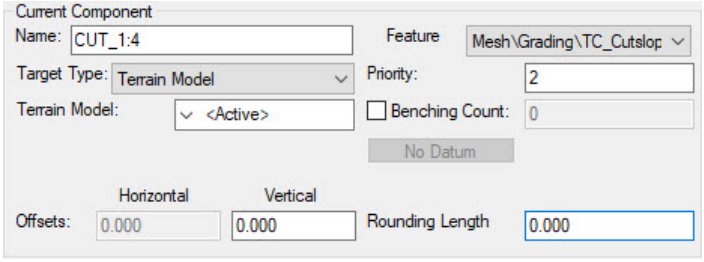
Create a 1:4 Cut Slope End Condition

1. *Right-click* in the Current Template window and select **Add New Component > End Condition**.
2. Define the end condition component properties with the following values:
   a. Type **CUT_1:4** for *Name*.
   b. Set the *Priority* to **2**.
   c. The other settings should still be good:

      Set the *Feature* to **Mesh\Grading\TC_Cutslope.**

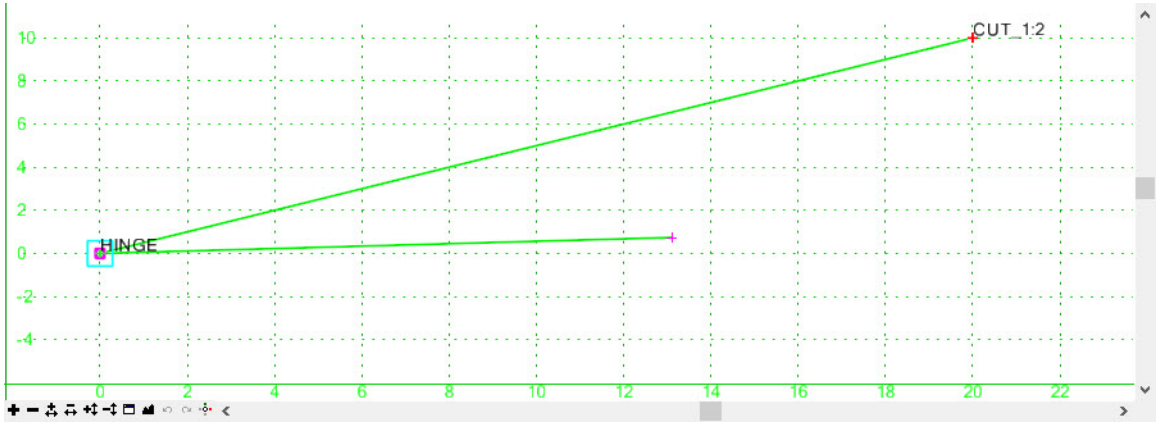
      Set the *Target Type* to **Terrain Model**.

      Set the *Terrain Model* to **<Active>**.



3. Place the first point of the end condition component.
   a. Hover over and click on the **HINGE** point to snap the first point to the same location.

      This merges the starting points of the two end condition components, forming an End Condition “tree”. Solutions are sought for each “branch”, starting with the branch with the highest Priority (having the lowest number).



4. Place the second point of the end condition component. In the *Dynamic Settings* dialog:

   a. Type a *Point Name* of **CUT_1:4**.

   b. **Clear** the *End Condition is Infinite* box.

      This 1:4 slope solution will not infinitely seek the target. It will only be built if it intersects the target within the 16' *[4m]* distance specified by the constraints.

   c. **These settings should still be good:**

      Set the *Check for Interception* box.

      Set the *Place Point at Interception* box.

      Clear the *Do Not Construct* box.

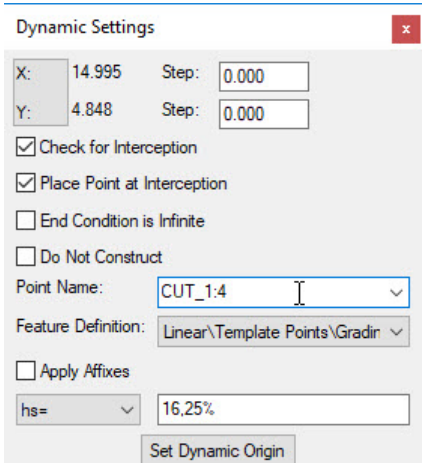
      Set the *Feature Definition* to **TL_End Cond Cut Tie**.

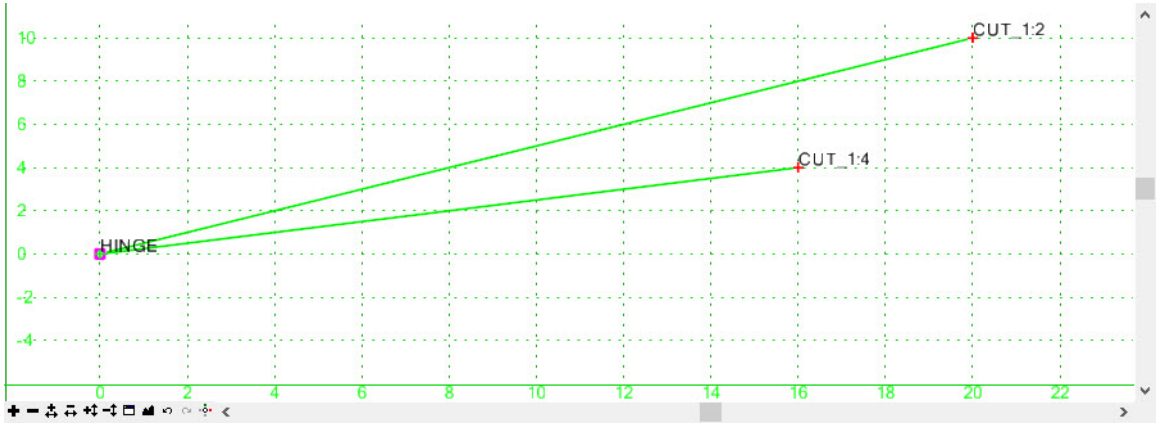
   d. Select the Precision Input Mode of **hs=**

   e. Type **16,25%** *[4,25%]* and click the **Enter** key.

      The **CUT_1:4** point is placed and the component is "drawn" to the new cursor position, awaiting the next point.

   f. *Right-click* in the *Current Template Window* and select **Finish**.



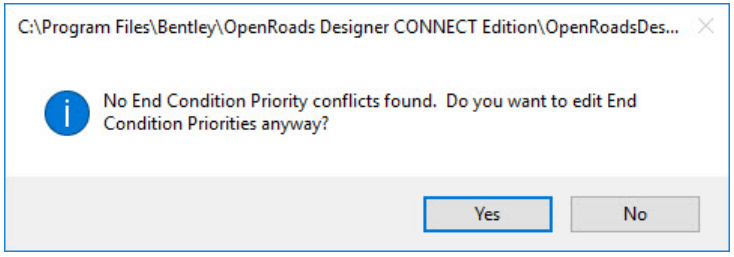


Test the 1:4 Cut Slope End Condition

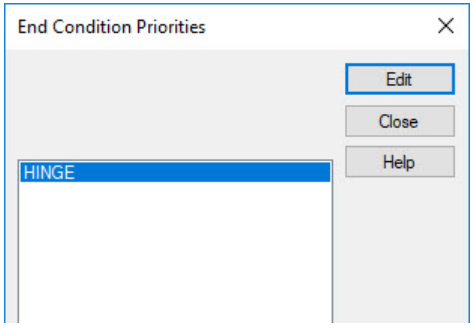
1. *Click* the **Test** button.
2. In the *Available Targets* list, click **<Active>-Terrain Model**.
3. *Click* the **Draw** button.
4. Move the cursor in the *Current Template Window* and notice that the **CUT_1:4** component is never created.

   This is because the **CUT_1:2** solution has a higher priority than the **CUT_1:4** solution so as long as it can be built it will be built. This is not what we desire. The **CUT_1:4** solution should be attempted first, and then the **CUT_1:2** built only if the **CUT_1:4** solution cannot be built.
5. *Click* the **Check Priorities** button in the upper right corner of the *Test End Conditions* dialog to change the priorities of the end conditions.

   NOTE: When there are priority conflicts, you will receive a warning message. You can then choose the *Check Priorities* button to review and edit the priority of each end condition.



6. When the Warning message appears click **Yes** to continue with the edit of priorities.
7. The *HINGE* point is the only location where components branch from a point. With *HINGE* highlighted click **Edit**.



8. Change the *CUT_1:2* Priority to **4** and click **OK**.

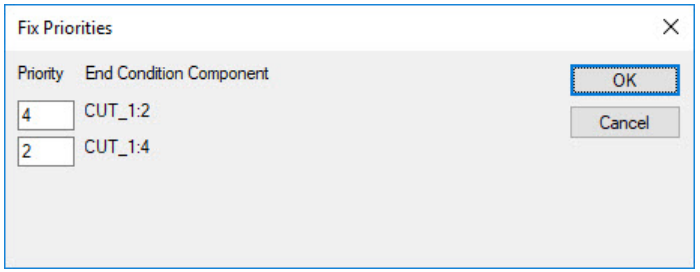
   The **CUT_1:4** solution will now process first because it has the higher priority.

9. Close the *End Condition Priorities* dialog.

10. Verify the template solves for the 1:4 slope up to 16 ft *[4 m]* wide and then the 1:2 slope.

11. Close the *Test End Conditions* dialog.

12. **Save** the template library.



Create a 1:2 Fill Slope End Condition with a Fixed 10% Slope Shoulder

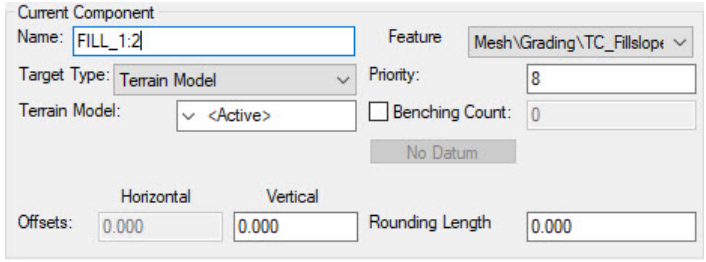
The 50% fill slope will be preceded by a 10% slope shoulder to provide a crash recovery zone.

---

1. *Right-click* in the Current Template window and select **Add New Component > End Condition**.
2. Define the end condition component properties with the following values:
   a. Type **FILL_1:2** for *Name*.
   b. Set the *Feature* to **Mesh\Grading\TC_Fillslope**.
   c. Set the *Priority* to **8**.
   d. These settings should still be good:

      Set the *Target Type* to **Terrain Model**.

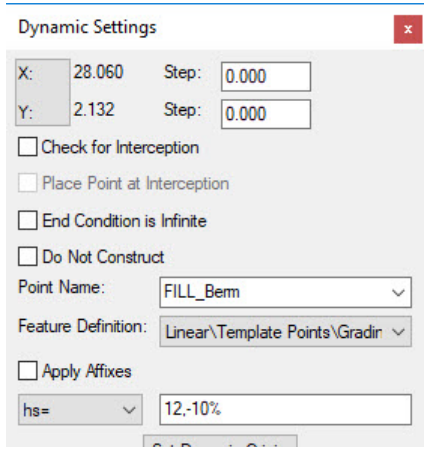
      Set the *Terrain Model* to **<Active>**.



3. Click on the **HINGE** point to place the first point of the 10% Slope Shoulder end condition component.
4. Place the second point of the 10% Slope Shoulder end condition component. In the *Dynamic Settings* dialog:
   a. **Clear** the *Check for Interception* box.

      This point will be created without seeking an interception with the target.
   b. **Clear** the *End Condition is Infinite* box.
   c. **Clear** the *Do Not Construct* box.
   d. Select a *Point Name* of **Berm** and edit the name to **FILL_Berm**.
   e. Set the *Feature Definition* to **TL_End Cond Berm In** if it isn't already.
   f. Select the Precision Input Mode of **hs=**
   g. Type **12,-10%** *[3,-10%]* and click the **Enter** key.

      The **FILL_Berm** point is placed and the component is "drawn" to the new cursor position, awaiting the next point.



5. Place the third point of end condition component which will define the 1:2 fill slope.

   In the *Dynamic Settings* dialog:

   a. **Set** the *Check for Interception* box.

      This 1:2 slope will seek interception with the target.

   b. **Set** the *Place Point at Interception* box.

   c. **Set** the *End Condition is Infinite* box.

   d. **Clear** the *Do Not Construct* box.

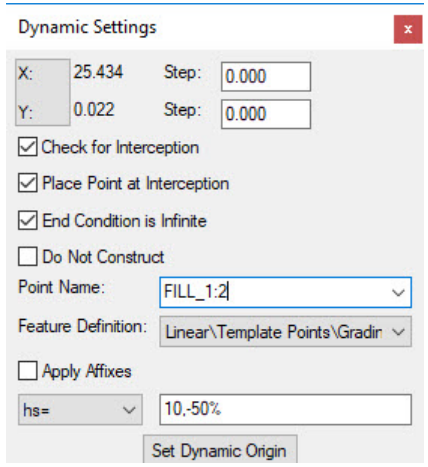
   e. Select a *Point Name* of **FILL** and edit the name to **FILL_1:2**.

   f. Set the *Feature Definition* to **TL_End Cond Fill Tie** if it isn't already.

   g. Select the Precision Input Mode of **hs=**

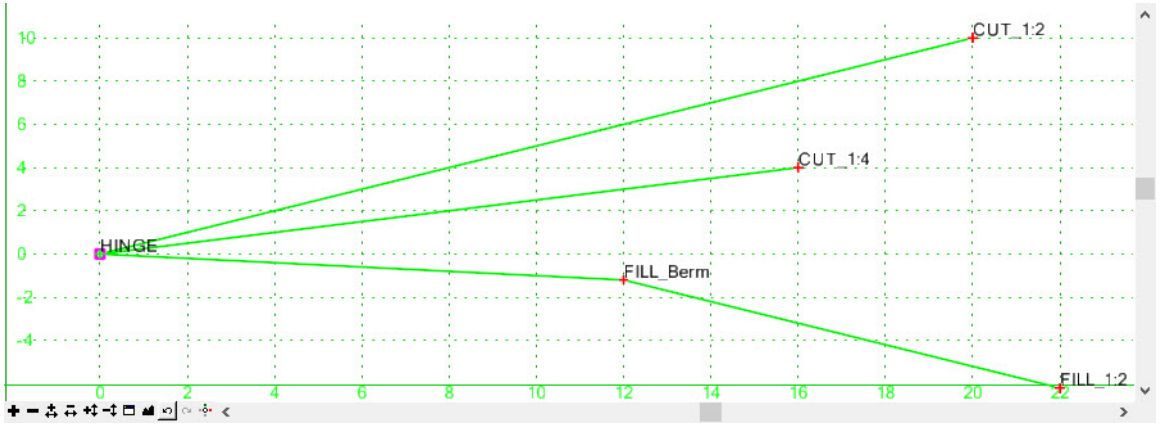
   h. Type **10,-50%** *[3,-50%]* and click the **Enter** key.

      The **FILL_1:2** point is placed and the component is "drawn" to the new cursor position, awaiting the next point.



   i. *Right-click* in the *Current Template Window* and select **Finish**.

   j. **Save** the Template Library.

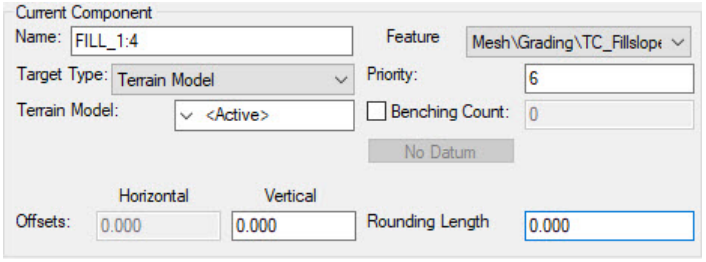


6. **Test** the 1:2 Slope Fill End Condition.

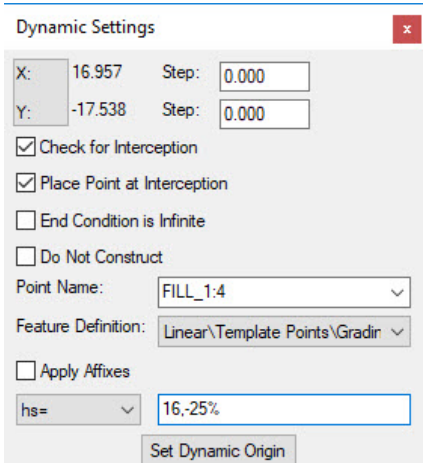
   Verify that the Fill Component does not solve during the flat shoulder/berm segment (because this segment does not seek to intercept the target), but does solve in the -50% segment.

Create a 1:4 Fill Slope End Condition

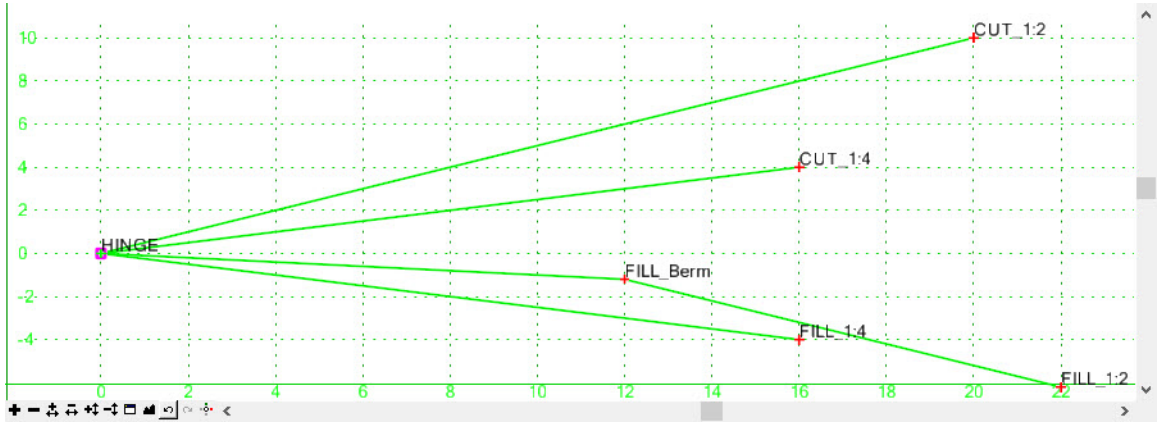
1. *Right-click* in the Current Template window and select **Add New Component > End Condition**.
2. Define the end condition component properties with the following values:
   a. Type **FILL_1:4** for *Name*.
   b. **Set** the *Priority* to **6**.
   c. The other settings should still be valid:
      Set the *Feature* to **Mesh\Grading\TC_Fillslope**
      Set the *Target Type* to **Terrain Model**.
      Set the *Terrain Model* to **<Active>**.



3. *Click* on the **HINGE** point to place the first point of the 1:4 fill slope end condition component.
4. Place the second point of the end condition component. In the *Dynamic Settings* dialog:
   a. Type a *Point Name* of **FILL_1:4**.
   b. **Clear** the *End Condition is Infinite* box.
   c. These settings should still be valid:
      Set the *Check for Interception* box.
      Set the *Place Point at Interception* box.
      Clear the *Do Not Construct* box.
   d. Set the *Feature Definition* to **TL_End Cond Fill Tie.**
   e. Type **hs=16,-25%** *[hs=4,-25%]* and click the **Enter** key.
   f. *Right-click* in the *Current Template Window* and select **Finish**.



The **FILL_1:4** point is placed and the component is “drawn” to the new cursor position, awaiting the next point.



5. **Test** the 1:4 Fill Slope End Condition.

   Verify the template solves for Cut and Fill properly: flatter components before the steeper ones.

6. **Save** the Template Library.

Exercise 2 - Create a Multi-slope Cut and Fill Solution with a Cut Ditch adjacent to the Hinge

Description

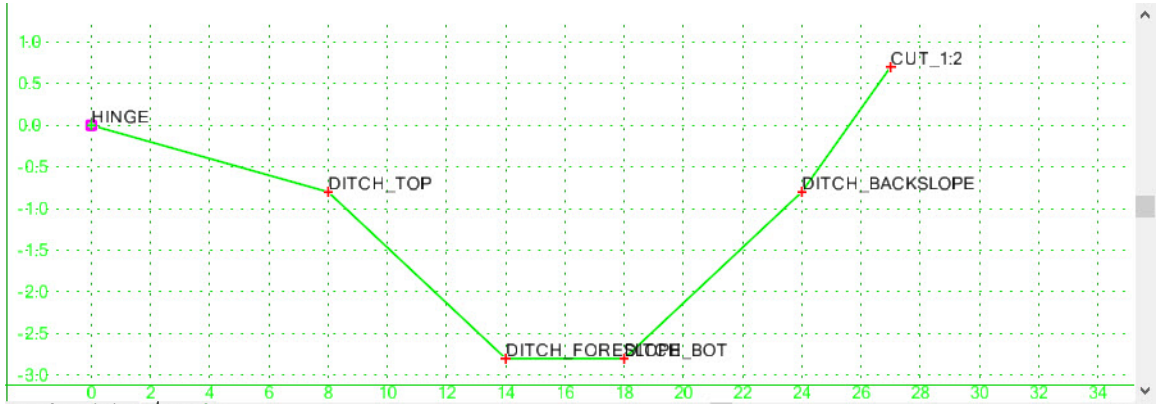
In this section, you will learn how to create a cut End Condition that has a cut ditch adjacent to the hinge and add Parametric Labels so that the ditch geometry can be changed as it is modeled along a corridor. Finally, you will learn how to merge together two separate cut and fill end conditions into a single end condition template.

Skills Taught

- Create Ditch End Condition
- Create Parametric Constraints
- Check and Resolve Priority Conflicts
- Test End Conditions

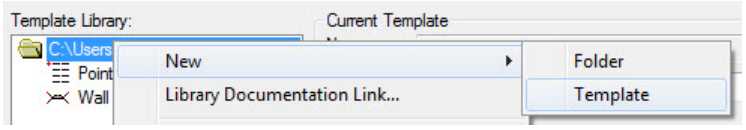
Create a Cut End Condition with a Ditch before the Cut Slope

In this exercise, you will create a cut end condition that has a fixed safety slope and flat bottom ditch before the 1:2 cut slope as shown in the following image.



---

1. *Right-click* on the template library object, click **New > Template**.



2. Name the new template **Cut Ditch - Interior**.
3. Create a new End Condition Component with the following parameters.
   - *Name*: **Cut Ditch Interior 1:2**
   - *Feature:* **Mesh\Grading\TC_Cutslope Berm**
   - *Target Type:* **Terrain Model** - **<Active>**
   - *Priority:* **8**

4. Use the *Dynamic Settings* dialog to define the **HINGE** point at the 0,0 coordinate.

   a. Ensure that the **Apply Affixes** box is clear.

   b. Select a *Point Name* of **HINGE**.

      The Feature Definition should automatically be set to *TL_Draft-DNC*.

   c. Type **xy=0,0** and click the **Enter** key.

      The Hinge point is placed and the component is “drawn” to the cursor, awaiting the next point.

5. Use the *Dynamic Settings* dialog to define the **DITCH_TOP** point which is a fixed segment that provides a crash recovery zone and does not seek to intercept the target.

   a. **Clear** the *Check for Interception* box.

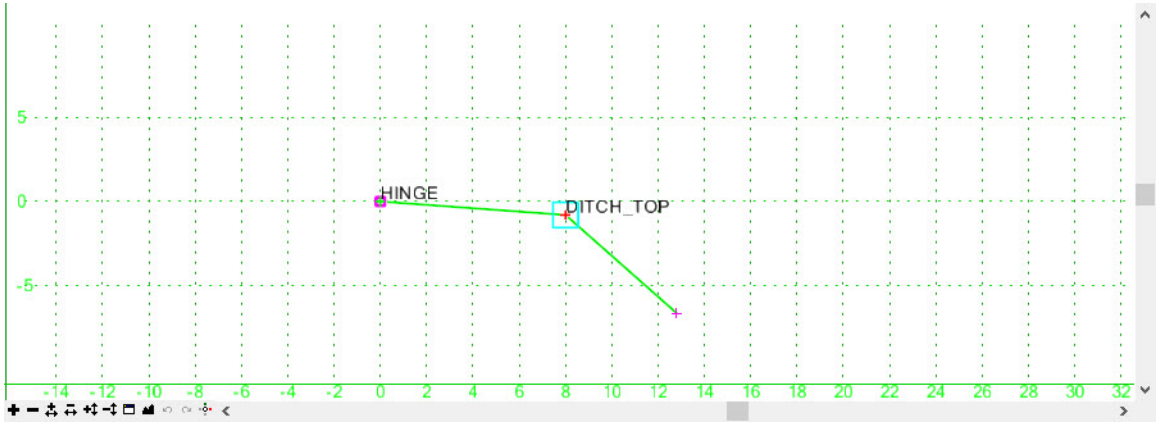
      Do not seek to the target (existing ground) until after the ditch is drawn.

   b. **Clear** the *End Condition is Infinite* box.

   c. **Clear** the *Do Not Construct* box.

   d. Select a *Point Name* of **DITCH_TOP**.

      The Feature Definition should automatically be set to *TL_End Cond Dit Out.*



   e. Type **hs=8,-10%** *[hs=2,-10%]* and click the **Enter** key.

      The **DITCH_TOP** point is placed and the component is “drawn” to the new cursor position, awaiting the next point.

6. Continue defining the points for the flat bottom ditch and back slope. These points (segments) are all fixed and do not seek to intercept the target so the only settings that need to be changed are the Point Names and Feature Definition which are automatically set when the Point Name is selected from the list.

   a. Define Ditch Foreslope

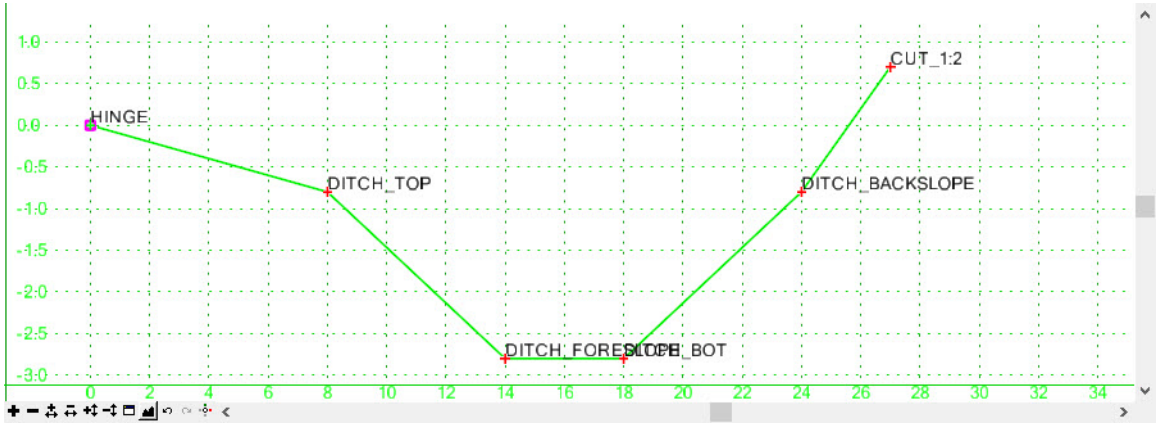
      - *Point Name:* **DITCH_FORESLOPE**
      - **hs=6,-1:3** *[hs=2,-1:3]*

   b. Define Ditch Bottom

      - *Point Name*: **DITCH_BOT**
      - **hs=4,0** *[hs=1.5,0]*

   c. Define Ditch Backslope

      - *Point Name:* **DITCH_BACKSLOPE**
      - **hs=6,1:3** *[2,1:3]*



7. Define the 1:2 cut slope that will seek interception with the target (existing ground).

   - *Check for Interception* = **set**
   - *Place Point at Interception* = **set**
   - *End Condition is Infinite* = **set**
   - *Do Not Construct* = **clear**
   - *Point Name*: **CUT_1:2**
   - *Feature Definition:* **Mesh\Grading\TL_End Cond Cut Tie**
   - **hs=3,50%** *[hs=1,50%]* (**NOTE**: the length is not important to the engineering since the *End Condition is Infinite* box is set.)

8. *Right click* and select **Finish**.
9. **Test** the End Condition and verify the template solves only during the final 50% Cut segment
10. **Save** the Template Library.

Add Parametric Constraints to the Ditch Geometry so that it might be Controlled by Corridors

Ditch geometry varies not only from project to project and from ditch to ditch within a project, but may vary along the length of the ditch. An oversized ditch takes up a lot of width. Defining Parametric Constraints on the Ditch Points allows shrinking or expanding the ditch to meet minimum hydraulic requirements while taking up the least amount of footprint. Corridors can adjust these Parameters station by station.

---

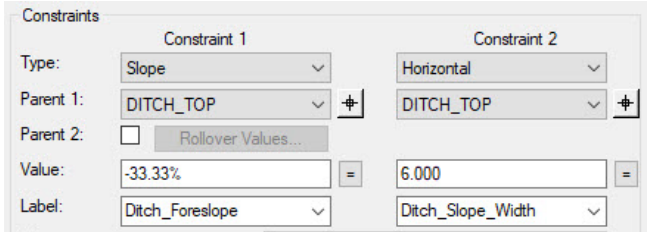
1. *Double-click* the **DITCH_FORESLOPE** point to define parametric constraint label to control the ditch foreslope.

   a. Type **Ditch_Foreslope** for the *Slope Label* to control the ditch slope.

   b. Type **Ditch_Slope_Width** for the *Horizontal Label* to control the width of the ditch slope.

   c. Click **Apply**.

   d. **Close** the dialog.

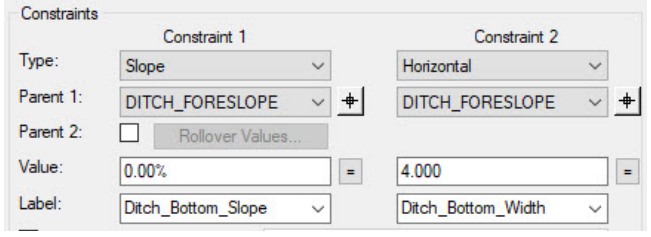


2. *Double-click* the **DITCH_BOT** point to define parametric constraint label to control the ditch bottom.

   a. Type **Ditch_Bottom_Slope** for the *Slope Label*.

   b. Type **Ditch_Bottom_Width** for the *Horizontal Label*.

   c. Click **Apply** and **Close** the dialog.



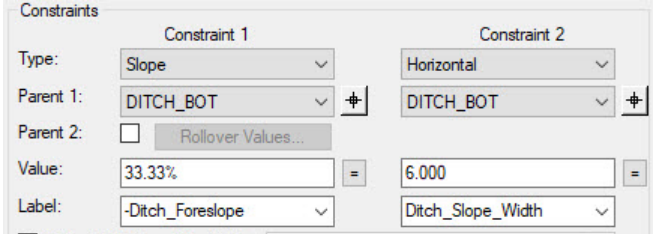
3. *Double-click* the **DITCH_BACKSLOPE** point to define parametric constraint label to control the ditch backslope. We will lock the foreslope slope and the backslope to be symmetric (they will both be defined by the Ditch_Slope_Width and Ditch_Foreslope parametric constraints).

   a. In the *Slope Label* drop-down list, select **Ditch_Foreslope**.

   b. In the *Slope Label* field, **type a minus sign (-)** before **Ditch_Foreslope** because the back slope needs to be in the *opposite* direction than the fore slope.

   a. In the *Horizontal Label* field, select **Ditch_Slope_Width**.

   b. *Click* **Apply** and **Close** the dialog.



4. Test the Parametric Controls

   a. Select the **Active Template** tab.

   b. Browse to the **Parametric Constraints** section and review the defined constraints.

   c. Expand a constraint name to see which points use that constraint.

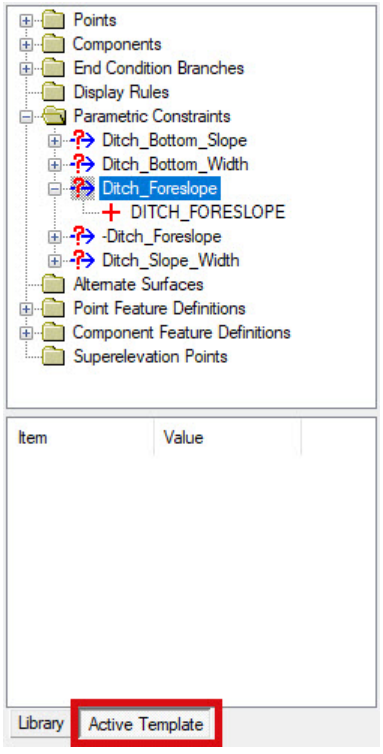
   d. *Double click* on the **Ditch_Foreslope** constraint.

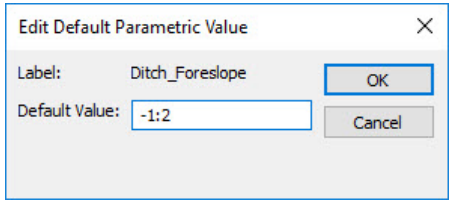
   e. Change the *Default Value* to **-1:2**, press the **TAB** key, and click **OK**.

      TIP: It is important that you press the **TAB** key or the new value may not be read.

   f. Review the End Condition and notice the new foreslope that has been applied.

   g. Set the *Default Value* for the **Ditch_Foreslope** back to **-1:3**.

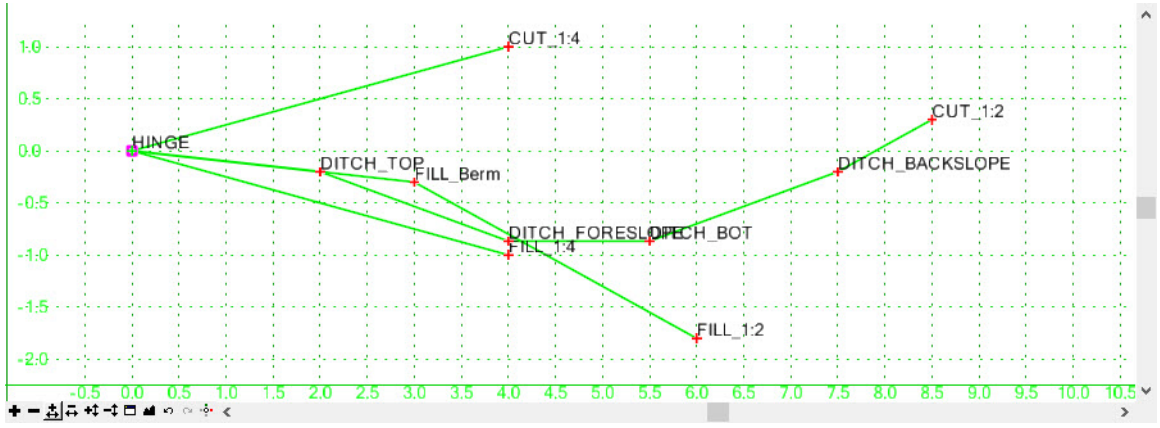




Create a Template with both Fill and Cut with Ditch End Conditions

1. Select the **Library** tab.
2. Copy the existing Cut and Fill template
   a. *Right-click* on the **Cut and Fill** template, click **Copy**.
   b. *Paste* a copy of the template into the library and rename it **Cut and Fill with Ditch**.
   c. *Double-click* it to make it the Current Template.
3. Delete the **CUT_1:2** component which will be replaced by the cut with ditch.

   HINT: Right-click and select **Delete Components** then drag the cursor over the component(s) to be deleted.
4. Add the Cut Ditch end condition
   a. Single click on the *Cut Ditch Interior* template and drag it into the **Cut and Fill with Ditch** template making sure the **HINGE** points merge.

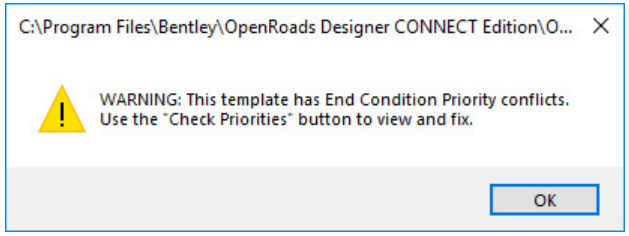


Test the Fill and Cut with Ditch Template and Fix Priority Conflicts

1. **Test** the End Condition.

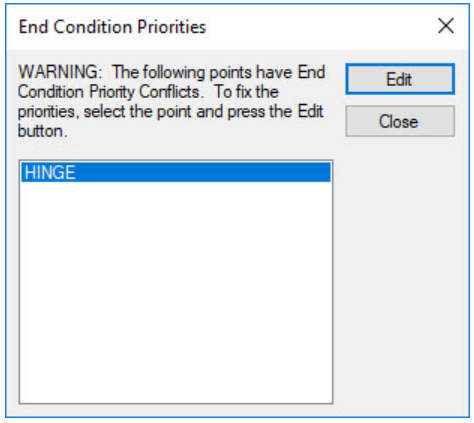
   If any priority conflicts are detected a warning dialog appears..

2. If the Priority Conflict dialog appears, click **OK** to continue.



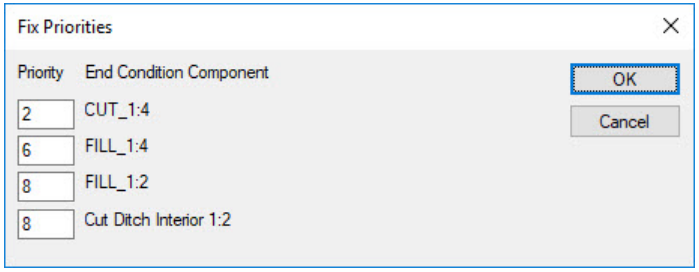
3. In the *Test End Conditions* dialog, click **Check Priorities**.

   The *End Condition Priorities* dialog lists all the Points with End Condition Components with identical priorities.

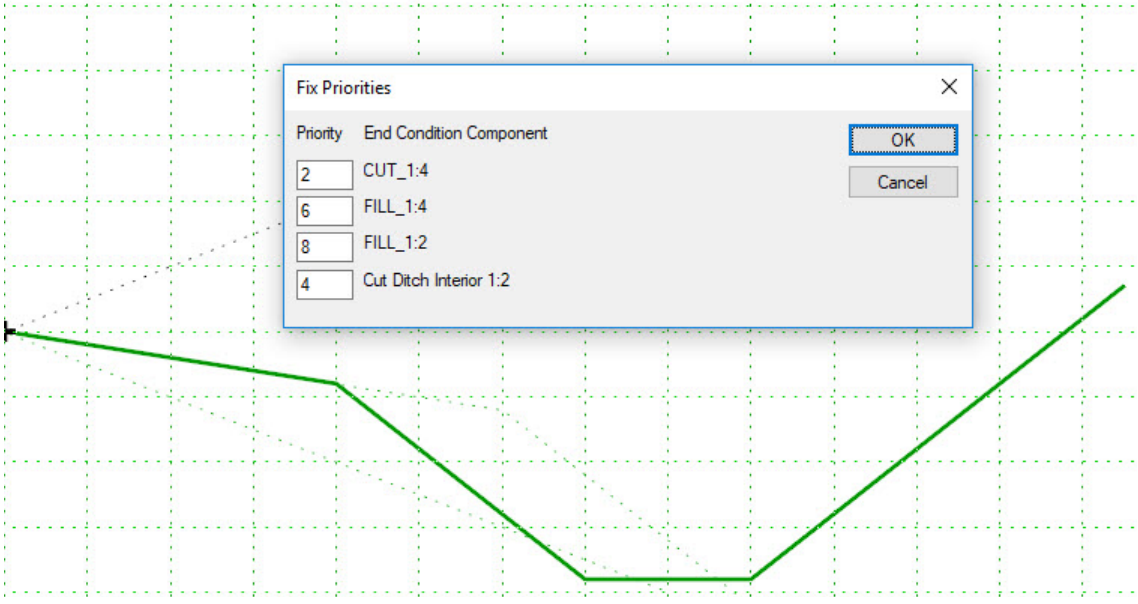


4. Click **Edit**.

   The *Fix Priorities* dialog lists all the components starting from the Hinge point and their priorities.



5. Change the Priority of the *Cut Ditch Interior 1:2* from **8** to **4** so that it solves before the **Fill_1:2** condition.



6. Click **OK** to accept the change and close the dialog.
7. **Close** the *End Condition Priorities* dialog.
8. **Test** the template.
9. **Save** the Template Library.

Exercise 3 - Build a Constant Width/Varying Slope Clear Zone

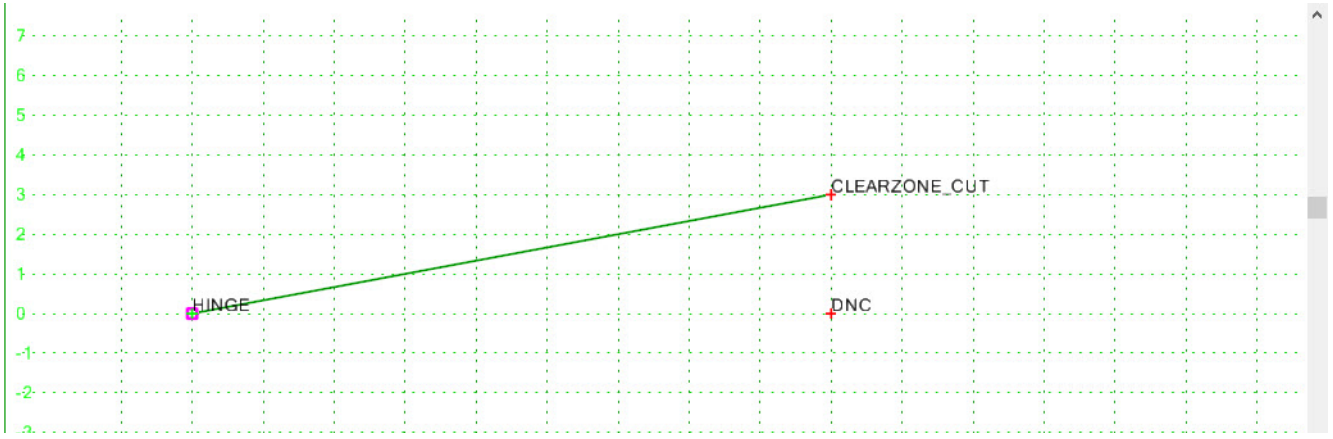
Description

In this section, you will learn how to create an end condition that has a fixed width and variable slope. This type of end condition is used in situations such as a clear zone grading or right-of-way grading where the horizontal position of the tie-in is known and the slope is variable.

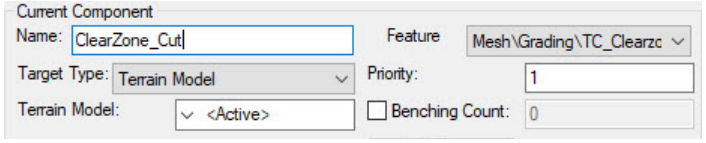
Skills Taught

- Master the *Do Not Construct* point property

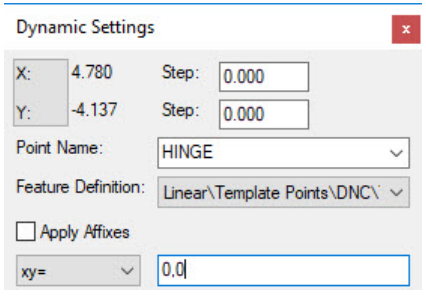
Create End Condition with Variable Cut Slope



1. Create a new template named **ClearZone**.
2. Double-click the **ClearZone** template to make it active.
3. Create a new End Condition component with the following parameters.
   - *Name*: **ClearZone_Cut**
   - *Feature:* **Mesh\Grading\TC_Clearzone**
   - *Priority:* **1**
   - *Target Type:* **Terrain Model** - **<Active>**



4. Use the *Dynamic Settings* dialog to define the **HINGE** point at the 0,0 coordinate.
   a. Select a *Point Name* of **HINGE**.
   b. Type **xy=0,0** and click the **Enter** key.



5. Use the *Dynamic Settings* dialog to define the **DNC** clear zone width point which is 18' *[6m]* from the **HINGE** point. The point does not seek to intercept the target and is not constructed as part of the final 3D model. It is used to located the edge of the clear zone from which an intersection with the ground is targeted.

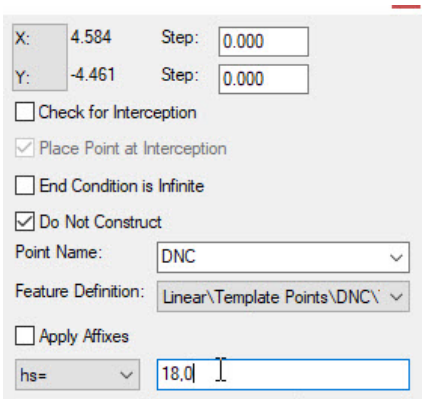
   a. *Check for Interception* = **Clear**

   b. *End Condition is Infinite* = **Clear**

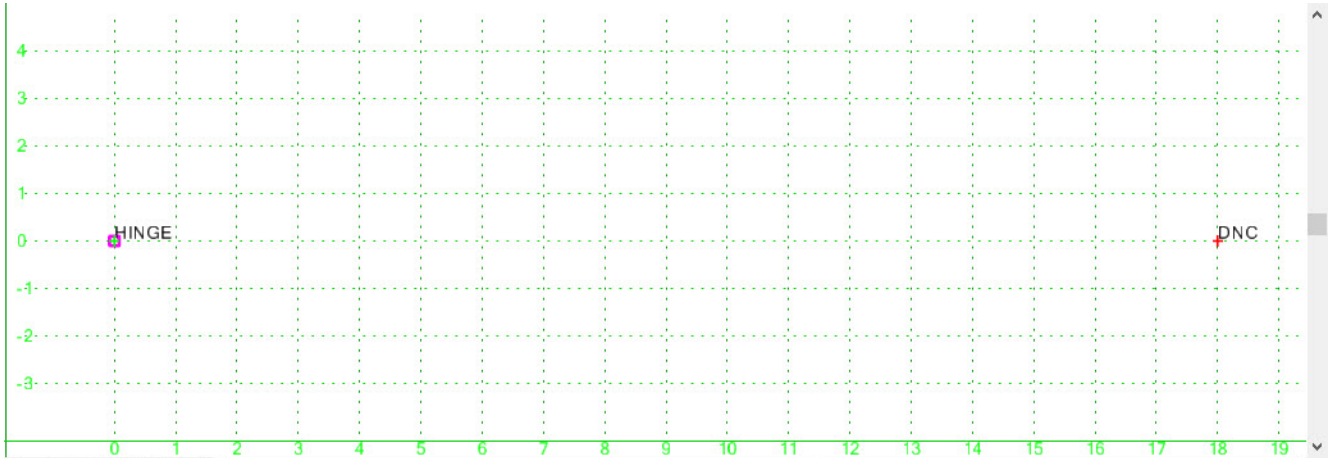
   c. *Do Not Construct* = **Set**

   d. *Point Name*: **DNC**

   e. **hs=18,0** *[hs=6,0]*



   The point is placed at the same elevation as the **HINGE** and offset by the clear zone distance. Notice that the line segment is not drawn from the **HINGE** to the **DNC**. This is due to *Do Not Construct* setting on the **DNC** point.



   Note: Adding a Parametric Label to the Horizontal Constraint - such as Clear_Zone_Width - would make it very easy to adjust the clear zone width along and between corridors.

   From this point the end condition will seek interception with the target (existing ground) vertically up or down. Connecting the point where the vertical line intersects the target with the **HINGE** is the variable cut slope desired.

6. Create the Cut condition point at the interception with the ground.

   a. *Check for Interception* = **Set**

   b. *Place Point at Interception* = **Set**

   c. *End Condition is Infinite* = **Clear**

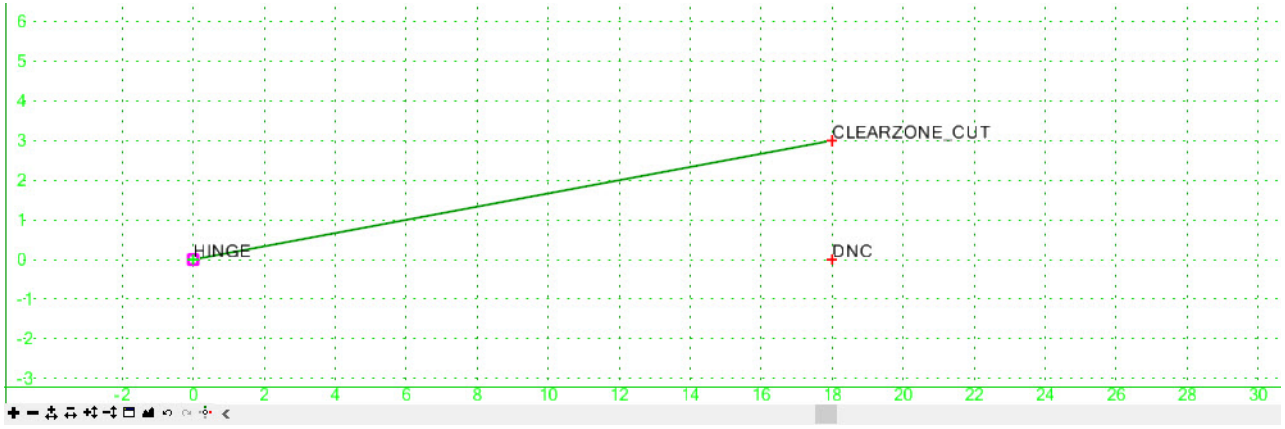
   d. *Do Not Construct* = **Clear**

   e. *Point Name*: **CLEARZONE_CUT**

   f. **dl=0,3** *[dl=0,1]*

      Seek to intercept the target (existing ground) within 3' *[1m]* vertically.

   g. **Right-click** in the Create Template window and select **Finish**.



   The "surface segment" runs directly between the **HINGE** and the **CLEARZONE_CUT** point.

7. **Test** the template.

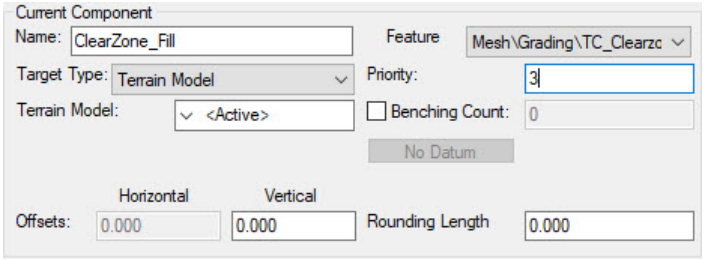
   The template should solve with a variable slope to the Clear Zone width only in cut conditions and only when the depth of the cut is less than 3' *[1m]* because that is the constraint that was placed on the **CLEARZONE_CUT** point.

Add Variable Fill Slope to End Condition

This is identical to the Clear Zone Cut solution, with the exception of the seek direction.

---

1. Create a new End Condition component with the following parameters.
   - *Name*: **ClearZone_Fill**
   - *Feature Definition:* **Mesh\Grading\TC_Clearzone**
   - *Priority:* **3**
   - *Target Type:* **Terrain Model** - **<Active>**



2. Click on the **HINGE** point to place the first point of the end condition component.
3. Place the Point that defines the clear zone width.

   Just like before, this point uses the *Do Not Construct* option so that it will not be created in the 3D Model. This point defines the width of the clear zone and is used to determine the actual cut and fill slopes that will intersect the target (existing ground) at this clear zone width.

4. Use the *Dynamic Settings* dialog to define the **DNC** point which is 18’ [6m] from the **HINGE** point. The point does not seek to intercept the target and is not constructed as part of the final 3D model. It is used to located the edge of the clear zone from which an intersection with the ground is targeted.

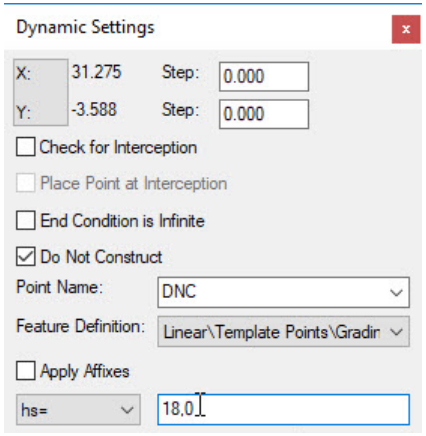
   a. *Check for Interception* = **Clear**

   b. *End Condition is Infinite* = **Clear**

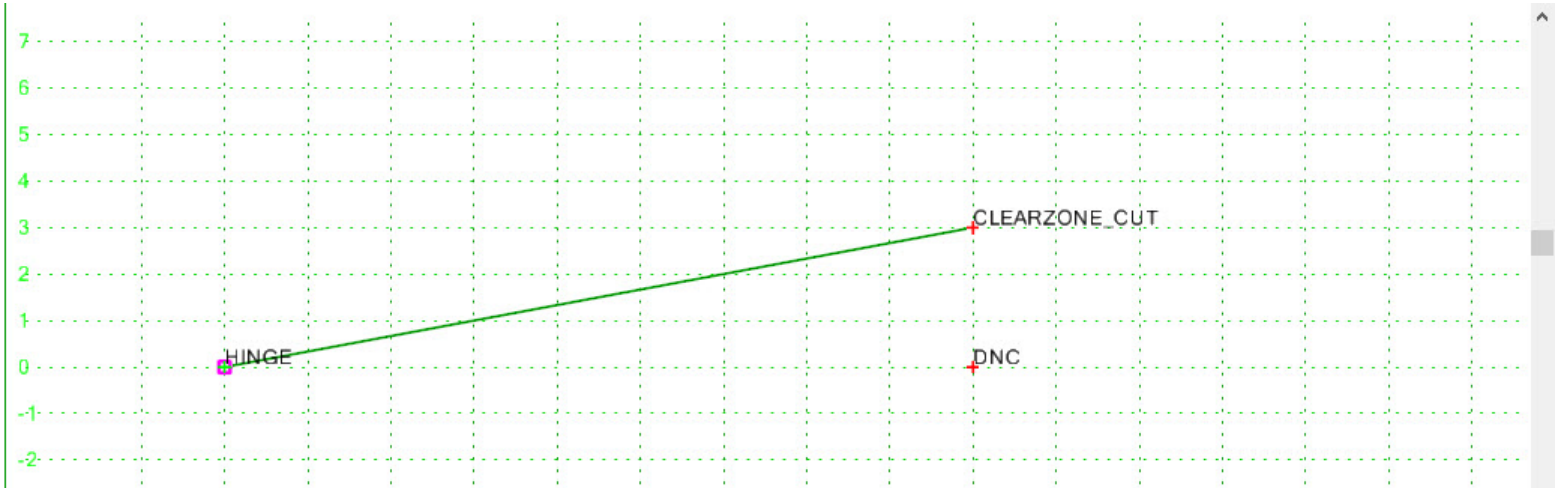
   c. *Do Not Construct* = **Set**

   d. *Point Name*: **DNC**

   e. **hs=18,0** *[hs=6,0]*



   The point is placed at the same elevation as the **HINGE** and offset by the clear zone distance. Notice that the line segment is not drawn from the **HINGE** to the **DNC**. This is due to *Do Not Construct* setting on the **DNC** point.



   From this point the end condition can seek interception with the target (existing ground) vertically up or down. Connecting the point where the vertical line intersects the target with the HINDGE is the variable cut or fill slope desired.

5. Create the Fill condition point at the interception with the ground.

   a. *Check for Interception* = **Set**

   b. *Place Point at Interception* = **Set**

   c. *End Condition is Infinite* = **Clear**

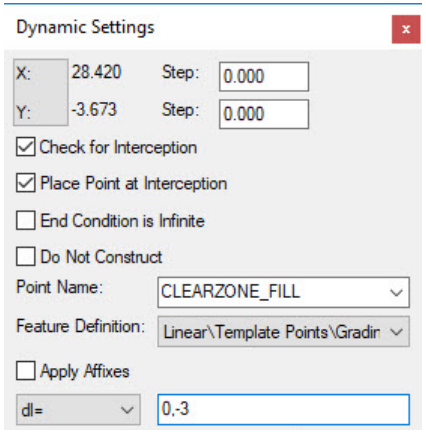
   d. *Do Not Construct* = **Clear**

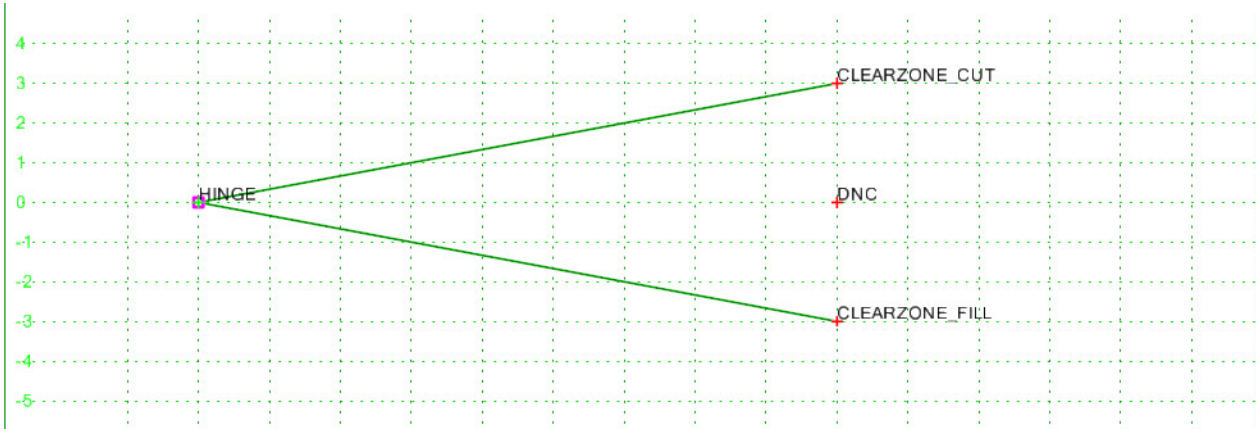
   e. *Point Name*: **CLEARZONE_FILL**

   f. **dl=0,-3** *[dl=0,-1]*

      Seek to intercept the target (existing ground) within -3' *[-1m]* vertically.

   g. *Right-click* and select **Finish**.





The "surface segment" runs directly between the HINGE and the CLEARZONE_FILL point.

6. **Test** the template.

   The template should solve with a variable slope to the Clear Zone width in both cut and fill conditions but only when the depth of the cut or fill is less than 3' *[1m]* because that is the constraint that was placed on the CLEARZONE_CUT and CLEARZONE_FILL points.

7. **Save** the template library.

Exercise 4 - Build and Test a Template that Places a Cut Wall

Description

Right-of-way restrictions are common on many projects and because of this walls are sometimes needed where slopes fall outside the limits of the right-of-way. Walls tend to be intermittent features along a corridor, seldom running its full length. A single template can run the full length of a Corridor, placing walls only when it encounters a Wall feature. One common technique in managing project design is to use 2D features to store the horizontal locations of the retaining walls along the corridor. A template can be created that uses the standard side slope when it does not find a retaining wall feature but uses a retaining wall solution when it does.

In this section, you will learn how to create a Cut End Condition that draws a Wall when it encounters a 2D Wall Feature. This will be done using a Feature Definition Horizontal Target to check for a horizontal wall feature. If found, a 50% slope will be placed to the wall location and then a near-vertical segment representing the inside wall face will intercept the existing terrain. If a wall style is not found, then a standard 50% cut slope component will attempt to find the existing terrain.

Skills Taught

- Target horizontal position of a Feature
- Use an existing Component in an End Condition
- Parent - Child Components

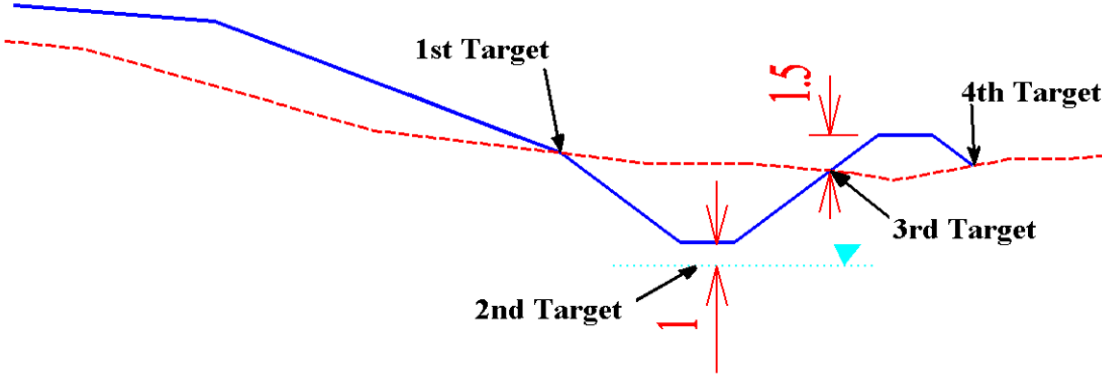
Seeking Multiple Targets

Some side slope solutions require targeting multiple targets. One example is a ditch at the bottom of a fill slope. The fill slope extends to the target (existing terrain) and then add an exterior fill ditch is built. A more elaborate example might be to fill to the terrain, put in a ditch to within one foot of the water table and finish with an exterior berm.

These are examples of serial targets: once an initial target is found additional earthwork is performed prior to tying to another target – which may or may not be the same target.

In this and upcoming exercises, we will build templates that seek a wall feature (for the wall bottom) and then the active terrain (for the wall top). We will also build a template that seeks a ROW line and then seeks the existing ground.

The software logic follows the engineering logic. If the first target is not found, further operations are not attempted. If, for example, you need a wall bottom feature, but it does not exist, there is no point to seek the terrain for the top.



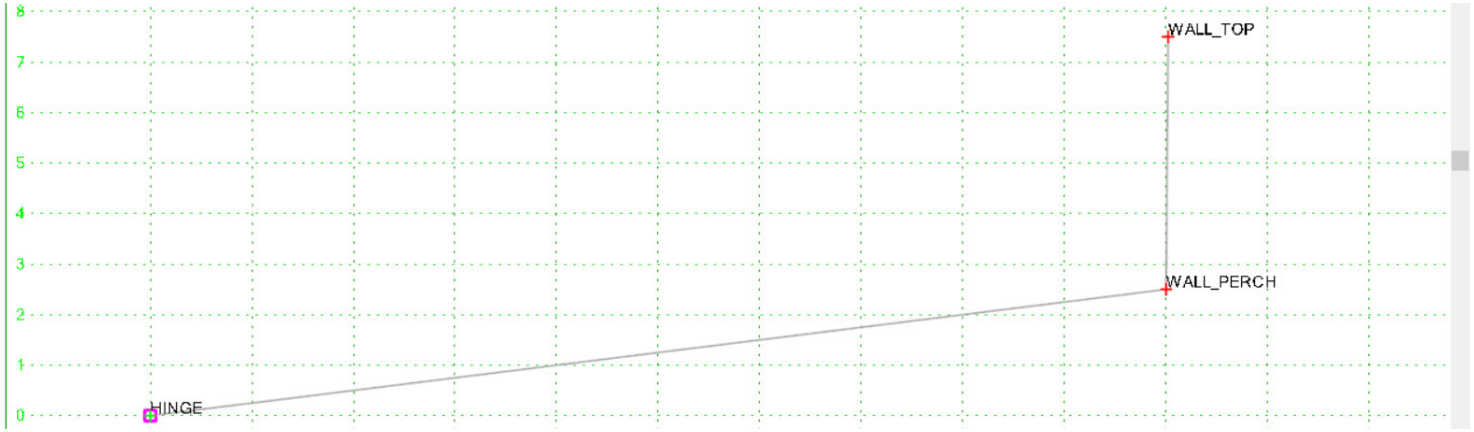
Any component failure in an End Condition path stops the path processing.

End Condition Path Rules

- When a component Succeeds, keep going.
- Once a Component in a path fails, the whole Path fails. Processing returns to the next path at a branch point, if one exists.
- A Component or Path that has no Seeking segments counts as a Success, but is only placed if the remainder of the path succeeds.

  An example of this is might be a single-segment End Condition component from shoulder to the branching point from which a series of related EC Components start.

Create the End Condition Component that Seeks the Wall Feature



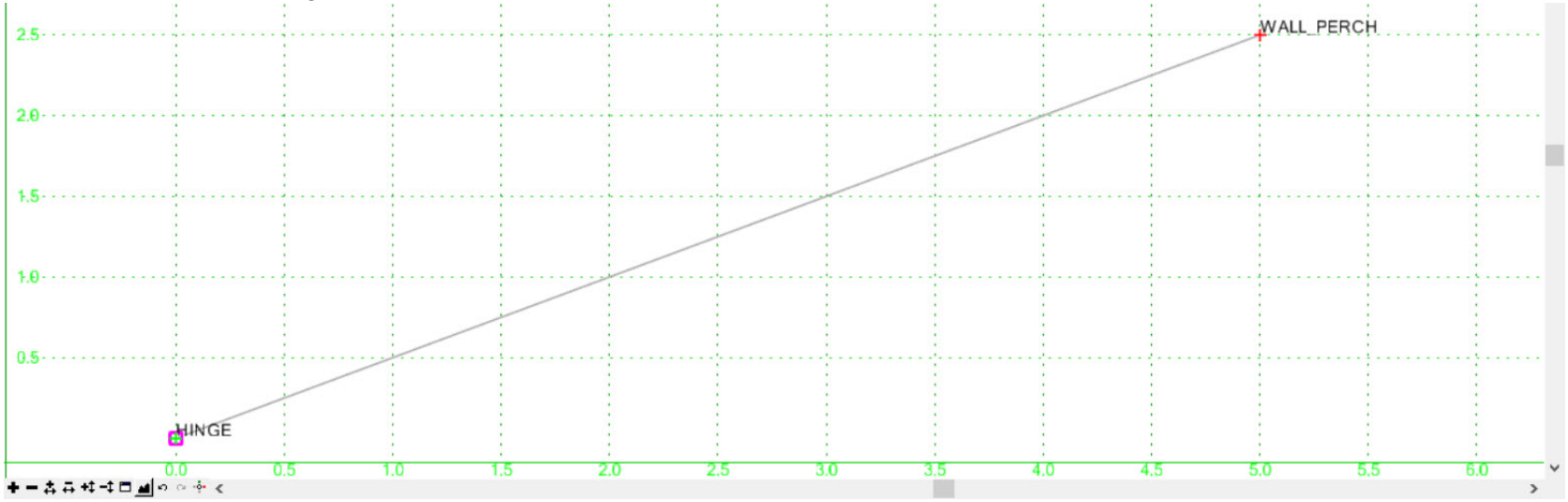
1. Create a new template named **Cut Wall**.
2. Create a new End Condition component with the following parameters.
   - *Name*: **Cut Wall**
   - *Feature:* **Mesh\Structural\TC_Ret Wall-Cut**
   - *Target Type:* **Feature Definition Horizontal**
   - *Feature Definition:* **Linear\Walls\Wall**

     The component will target the horizontal location of any feature with a Feature Definition of *Wall*.
   - *Priority:* **1**
3. Use the *Dynamic Settings* dialog to define the HINGE point at the 0,0 coordinate.
   a. Select a *Point Name* of **HINGE**.
   b. Type **xy=0,0** and click the **Enter** key.

4. Place a point at the bottom of the wall where the target (horizontal feature) intersects the 1:2 or 50% slope.

   The point is placed if the target feature is intercepted. The point seeks an interception for an infinite distance and a slope 1:2 or 50% slope is created from the **HINGE** to the bottom of the wall.

   - *Check for Interception* = **Set**
   - *Place Point at Interception* = **Set**
   - *End Condition is Infinite* = **Set**
   - *Do Not Construct* = **Clear**
   - *Point Name:* **WALL_PERCH**
   - *Feature Definition:* **Linear\Walls\Wall_Perch**
   - **hs=5,50%** *[hs=2,50%]*
   - *Right-click* to **Finish** adding to the component.



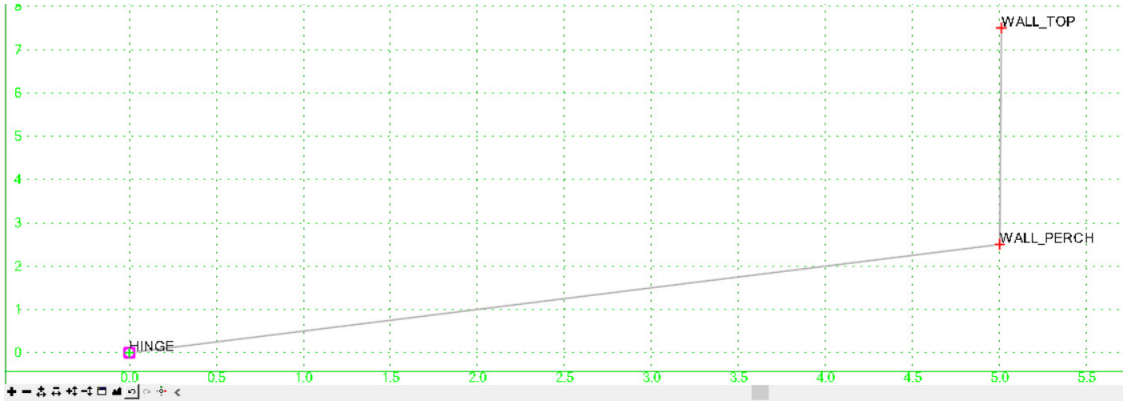
5. **Test** the component and observe that the 50% slope ends at the horizontal location of the feature.

   NOTE: If the template is not solving to the wall feature, check the end condition component properties and be sure that the wall target feature definition is set to **Linear\Walls\Wall.**

6. Create a new End Condition component with the following parameters.
   - *Name*: **Cut Wall Face**
   - *Feature:* **TC_Ret Wall-Cut**
   - *Target Type:* **Terrain Model - <Active>**
   - *Priority:* **1**
7. Place a point at the **WALL_PERCH** point, ensuring a merge.
8. Place the **WALL_TOP** point where the wall intercepts the target (existing ground).

   The **WALL_TOP** point infinitely seeks the active terrain or in other words there is no limit to the height of the wall.

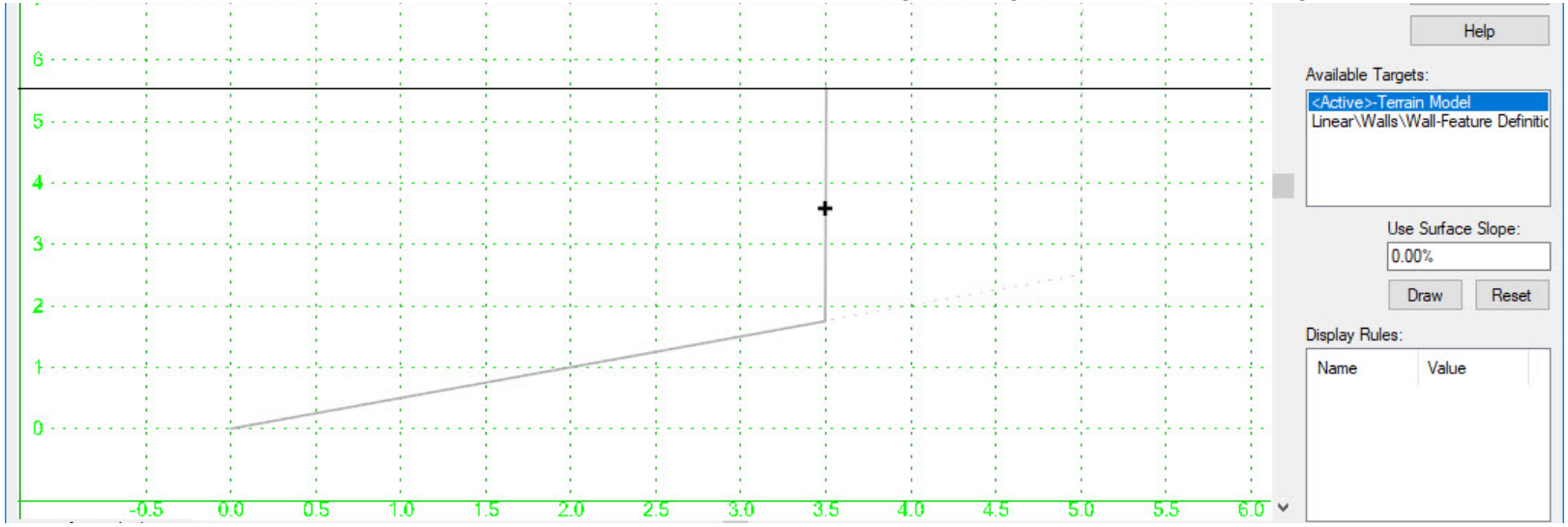
   - *Check for Interception* = **Set**
   - *Place Point at Interception* = **Set**
   - *End Condition is Infinite* = **Set**
   - *Do Not Construct* = **Clear**
   - *Point Name:* **WALL_TOP**
   - *Feature Definition:* **Wall_Top**
   - **dl=0.01,5** *[dl=0.01,3]*
   - *Right-click* to **Finish** adding to the component.



Previously, we have tested simple single-target End Conditions. As templates with multiple target types are generally more complex, testing them is even more important.

The Test End Conditions dialog Available Targets lists all the targets in the active template. Each of these can be selected and drawn.

9. **Test** the template.
   a. Select the **<Active> - Terrain Model** target.
   b. Click **Draw** and click in the window to define the location of the terrain.
   c. Select **Linear\Walls\Wall-Feature Definition** target.
   d. Click **Draw** and move the cursor about the window to see how the slope and wall react to the combination of the two targets.
   e. Click in the window to lock the location of the wall.
   f. Select the **<Active> - Terrain Model** target again.
   g. Click **Draw** and move the cursor about the window to see how the wall height changes as the terrain changes.



10. **Save** the Template Library.

Add a Wall Shape to the Cut Wall End Condition

End Conditions cannot be closed shapes.

There are times when a closed shape needs to be used in conjunction with an End Condition solution. Closed shapes can be used for volumes (and included in End Area Volume calculations) for such things as concrete ditch linings and retaining wall structures. Closed shapes can be drawn as a Simple, Constrained or Unconstrained components.

An End Condition component point being merged onto by a closed shape retains its functionality. For example, a terrain-seeking daylight point would continue to seek the terrain whether it is a member of only an end condition component or if it is a member of an end condition component and a retaining wall shape component.

All components have a property called Parent. When a component's Parent does not display, the component does not display. When using closed shape component in an end condition component, the most straightforward way to tie its display status to the End Condition status is to make the shape component a Child of the End Condition component.

Add a Wall Shape to the Wall Solution

In this section, we will add a previously-created wall shape to the Wall End Condition.

---

1. Review the **Wall Shape** template by setting it active.
2. *Move* the **WALL_TOP** point and observe that the wall grows and shrinks about the **WALL_PERCH** point in proportion to the all height.
3. *Double-click* the **Cut Wall** template to make it active.
4. Add the wall shape to the End Condition.
   a. *Single click* on the **Wall Shape** template and drag it into the **Cut Wall** template making sure the **WALL_PERCH** points merge.

      Next, we need to have the WALL_TOP point in the Wall Shape be controlled by the WALL_TOP point in the End Condition.

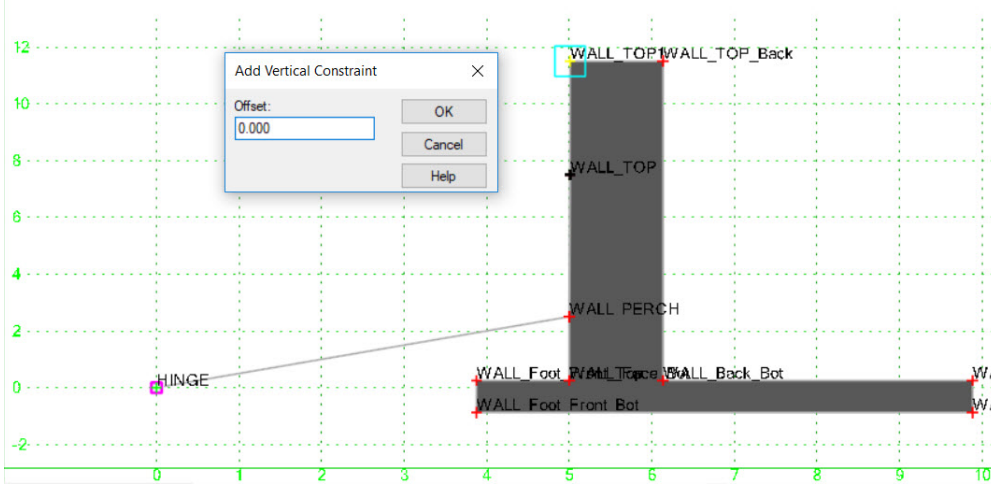
   b. *Right-click* the top front point of the wall shape, **WALL_TOP1**. Click **Add Constraint > Vertical**.

      NOTE: The **WALL_TOP** point in the Wall Shape was automatically suffixed with a number to maintain unique point names when it was copied into the End Condition.

   c. At the *"DB: Select Parent Point, RST: Exit"* prompt, click the **WALL_TOP** point.

   d. Set the Vertical *Offset* to **0**.

   e. Click **OK**.



5. **Test** the template.

   The wall shape should adjust with the wall face component.

   Note that the wall shape always displays. We need it to display *only* when the Cut Wall Face component is drawn.

6. Close the *Test End Conditions* dialog.

   The Wall Shape component needs to be a child of the Wall End Condition. This way the wall shape will not show unless a cut wall is needed.

7. *Double-click* on the **Struc_Wall** component (the wall shape).

8. In the Component Properties dialog:

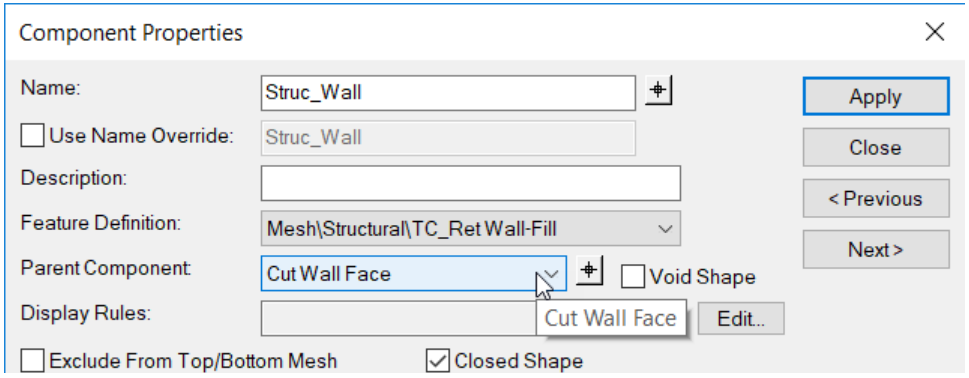
   a. select **Cut Wall Face** as the *Parent Component*.

   b. Click **Apply**.

   c. **Close** the dialog.

9. **Test** the template. The wall shape is no longer drawn when the wall End Condition does not solve.

10. **Save** the template library.



Exercise 5 - Create a Forced Right-of-Way End Condition

Description

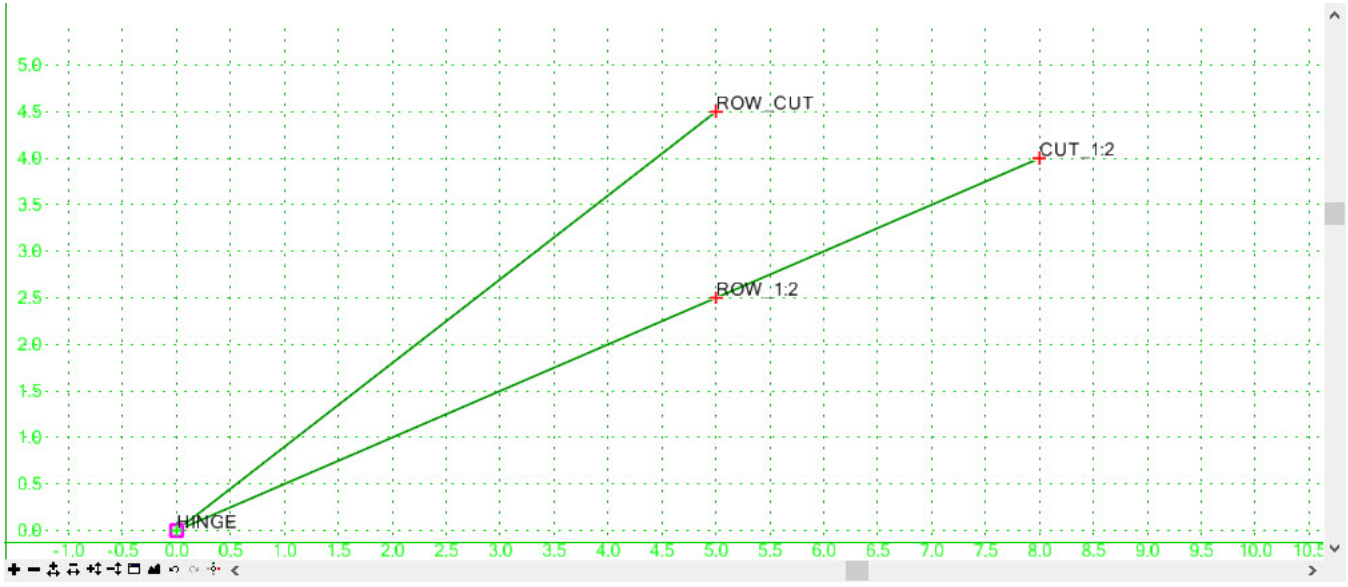
Side slopes tend to come in standard slopes: 1:6, 1:4, 1:3, 1:2. Sometimes there is not enough right-of-way to accommodate even the steepest “standard” slope. One solution is to force the slope to tie at the right-of-way.

In this section, you will learn how to build a template that solves for a 50% cut slope, unless the 50% cut slope lies outside the ROW line, then it will solve directly from the Hinge to the surface at the right-of-way line.

The way to solve for this in End Condition logic is to seek a solution where the ROW feature exists and the surface lies above the 50% cut line. If that fails, solve with a regular 50% cut solution.

Skills Taught

- Test multiple slope conditions



Create an End Condition that Targets Right-of-Way Geometry

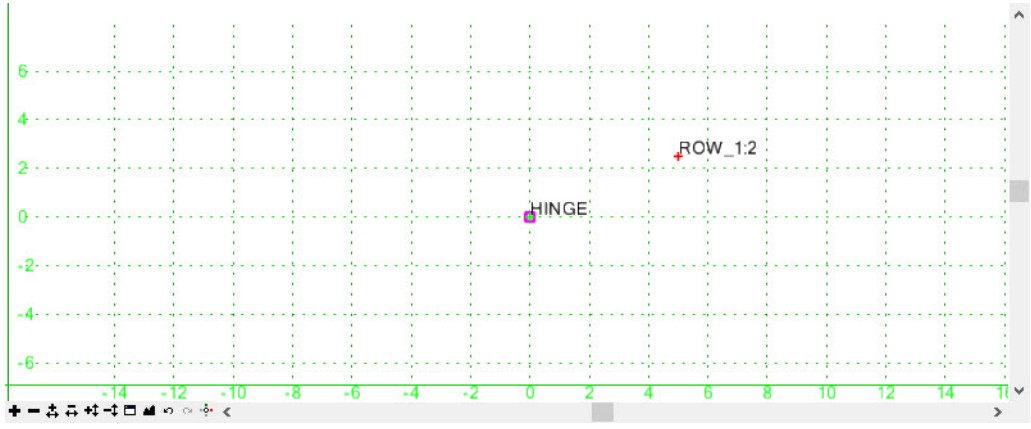
1. Create a new template named **Forced ROW**.
2. *Double-click* the **Forced ROW** template to make it active.
3. Create a new End Condition component with the following parameters.
   - *Name*: **ROW SEEK**
   - *Feature:* **Mesh\Grading\TC_Cutslope**
   - *Target Type:* **Feature Definition Horizontal**
   - *Feature Definition:* **Linear\Right of Way\E_ROW**

     The component will target the horizontal location of any feature with a Feature Definition of **E_ROW**. If the component fails to find a ROW line, there is no right-of-way line, the component and the component "branch" fail.
   - *Priority:* **1**
4. Use the *Dynamic Settings* dialog to define the HINGE point at the 0,0 coordinate.
   a. Select a *Point Name* of **HINGE**.
   b. Type **xy=0,0** and click the **Enter** key.

5. Place a point at the right-of-way where the target (horizontal feature) intersects the 1:2 or 50% slope.

   The point is placed if the target feature is intercepted. The point seeks an interception for an infinite distance and a slope 1:2 or 50% slope is created from the HINGE to the right-of-way.

   - *Check for Interception* = **Set**
   - *Place Point at Interception* = **Set**
   - *End Condition is Infinite* = **Set**
   - *Do Not Construct* = **Set**
   - *Point Name:* **ROW_1:2**
   - *Feature Definition:* **Linear\Template Points\DNC\TLC_Draft-DNC**
   - **hs=5,50%** *[hs=2,50%]*
   - *Right-click* to **Finish** adding to the component



Add an End Condition Seeking the Active Terrain above the 50% line at the ROW Location

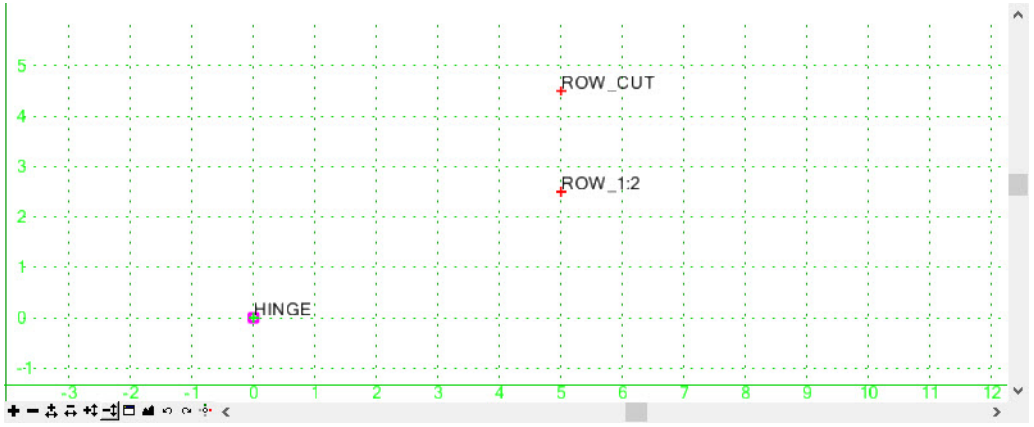
If this component is activated it means that the ROW line was found. If this component fails to find the existing ground, then it means the surface is below the 50% slope threshold.

---

1. Create a new End Condition component with the following parameters.
   - *Name*: **CUT SEEK**
   - *Feature:* **Mesh\Grading\TC_Cutslope**
   - *Target Type:* **Terrain Model - <Active>**
   - *Priority:* **1**
2. Place a point at the **ROW_1:2** point, ensuring a merge.
3. Place a point at the right-of-way where the target (horizontal feature) is intersects the 1:2 or 50% slope.

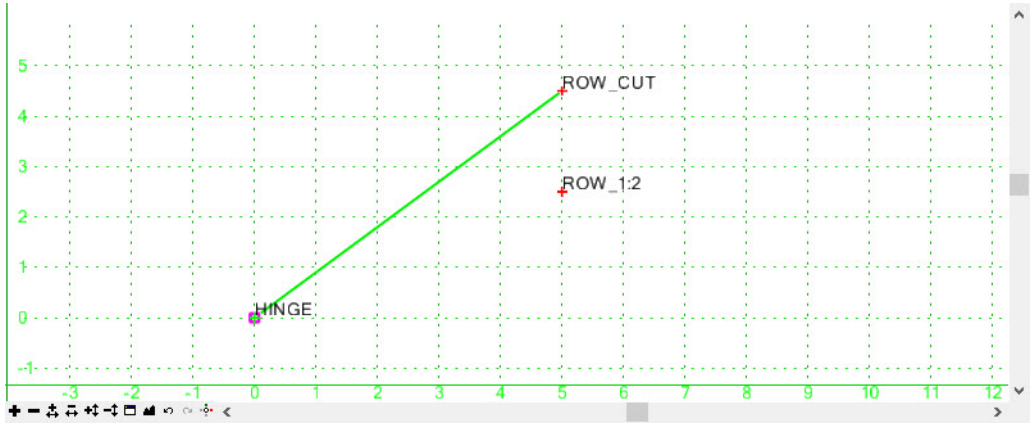
   The point is placed if the target feature is intercepted. The point seeks an interception for an infinite distance and a slope 1:2 or 50% slope is created from the HINGE to the right-of-way.

   - *Check for Interception* = **Set**
   - *Place Point at Interception* = **Set**
   - *End Condition is Infinite* = **Set**
   - *Do Not Construct* = **Clear**
   - *Point Name:* **ROW_CUT**
   - *Feature Definition:* **Linear\Template Points\Grading\TL_End Cond Cut Tie**
   - **dl=0,2** *[dl=0,1]*
   - *Right-click* to **Finish** adding to the component



Because the ROW_1:2 point is set to Do Not Construct, neither End Condition will display because only one point in each is drawn. A Component between the Hinge and ROW_CUT point needs to be added.

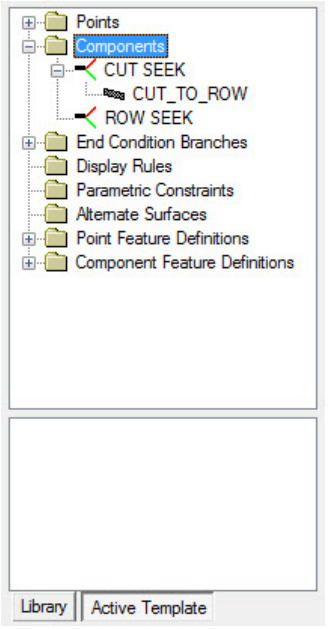
4. Create a new *Unconstrained* component to connect the HINGE and ROW_CUT with the following parameters.
   - *Name*: **CUT_TO_ROW**
   - *Feature:* **Mesh\Grading\TC_Cutslope**
5. Place a Point at the **HINGE** point.
6. Place a Point at the **ROW_CUT** point.
7. *Right-click* to **Finish** adding to the component.



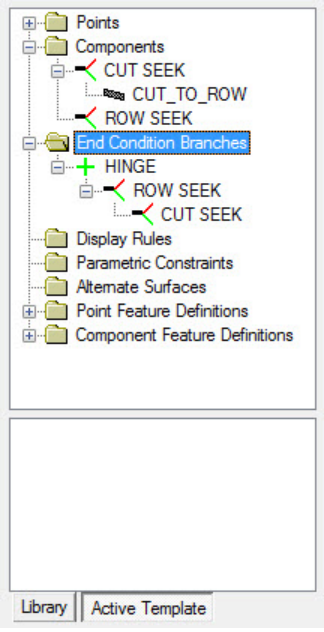
8. *Double-click* on the **CUT_TO_ROW** component.
9. Set the *Parent Component* to **CUT SEEK**.

   We want to make the display of the CUT_TO_ROW component conditional on the success of the ROW solutions.
10. **Test** the template.

It may be unclear how the three components created in this exercise relate to one another. The *Active Template* tree view shows components relationships to each other. The Components and End Conditions Branches leafs are listed in “tree” format showing the parent child relationships.

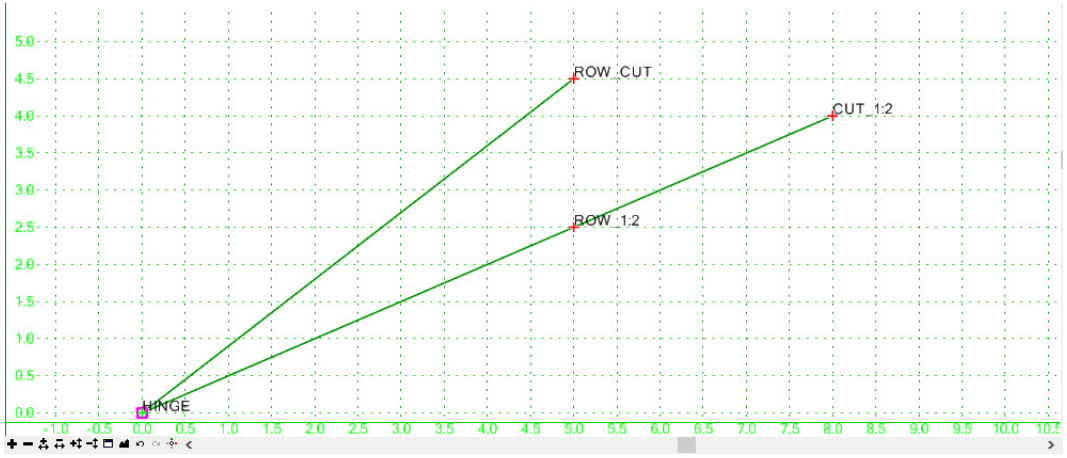


11. Select the **Active Template** tab.
12. Expand **Components**.
    - There are three components in this template named CUT SEEK, ROW SEEK, and CUT_TO_ROW.
    - The CUT_TO_ROW component is a child of the CUT SEEK component meaning it will only be built when the CUT SEEK component is successful.
13. Expand **End Condition Branches**.
    - There is only one branch in this template that begins at the HINGE point.
    - The first segment is between the HINGE and ROW SEEK.
    - A second segment begins at the ROW SEEK and goes to the CUT SEEK point.
    - If the first segment (ROW SEEK) fails, the second segment is not built.



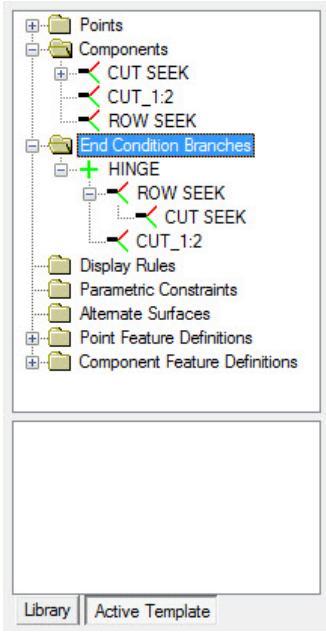
Add an End Condition for the 1:2 Cut Slope Inside the Right-of-Way

1. Continuing in the *Forced ROW* template, create a new End Condition component with the following parameters.
   - *Name*: **CUT_1:2**
   - *Feature:* **Mesh\Grading\TC_Cutslope**
   - *Target Type:* **Terrain Model - <Active>**
   - *Priority:* **4**
2. Place a point at the **HINGE** point, ensuring a merge.
3. Place a point that intercepts the Active Terrain target at a 1:2 or 50% slope inside the right-of-way target.
   - *Check for Interception* = **Set**
   - *Place Point at Interception* = **Set**
   - *End Condition is Infinite* = **Set**
   - *Do Not Construct* = **Clear**
   - *Point Name:* **CUT_1:2**
   - *Feature Definition:* **TL_End Cond Cut Tie**
   - **hs=8,50%** *[hs=3,50%]*
   - Right-click to **Finish** adding to the component



4. **Test** the template. Verify that it solves in all cut situations:
   - Cut - No ROW Feature
   - Cut at 50% with ROW to the outside of the daylight
   - Cut at the Surface at the ROW feature, inside the 50% daylight point.

5. Review the End Condition Branches in the Active Template tree.
   - There are now two branches from the HINGE point.
     - The ROW SEEK branch that looks for the ROW geometry.
     - The CUT_1:2 branch that seeks to intercept the target at a 1:2 slope.
6. **Save** the template library.



Skills Assessment

The questions below will test your retention of the skills covered in this course.

---

1. End Conditions can be closed shapes.
   a. True
   b. False
2. Targets are defined as a Point Property.
   a. True
   b. False
3. Every End Condition component has a Target.
   a. True
   b. False
4. At least one point in an End Condition component must seek a target.
   a. True
   b. False
5. The Place Point at Interception check box must be selected in order for a point to be placed if the target is found.
   a. True
   b. False
6. There can be gaps in the sequence of Priorities at a branch. For example, 2, 5, 9, 10, are valid Priorities at an End Condition Branch.
   a. True
   b. False
7. If an End Component in an End Component path succeeds, the path succeeds and the branch stops processing.

a. True

b. False

8. Every End Component in a Path must succeed in order for the Path to succeed and the End Components in the Path to be placed.

   a. True

   b. False

9. A point not seeking intersection is considered a failure.

   a. True

   b. False

10. As soon as a point in an End Condition component succeeds, the component is considered a success and stops processing other points in the component.

    a. True

    b. False

11. Arguably the primary advantage of Feature Definition targets versus Linear Targets is that with Feature Definition: it is easy to target and manage lots of data by function-specific Feature Definitions --ROW, Retaining Walls, utilities, etc.

    a. True

    b. False

Skills Assessment - Answers

The answers to the skills assessment questions are highlighted below.

---

1. End Conditions can be closed shapes.
   a. True
   b. **False - End Conditions cannot be Closed shapes. Closed Components can, however, be attached to any template point or points.**
2. Targets are defined as a Point Property.
   a. True
   b. **False – Targets are Component Properties.**
3. Every End Condition component has a Target.
   a. **True – Every End Condition component has a Target.**
   b. False
4. At least one point in an End Condition component must seek a target.
   a. True
   b. **False – An End Condition component with no points seeking the target is allowed and is considered a success.**
5. The Place Point at Interception check box must be selected in order for a point to be placed if the target is found.
   a. True
   b. **False – A point will be placed as long as the Do Not Construct check box is not set. Place Point at Interception when set places the point along the template segment where the target is found. When clear the point is placed at the full segment length regardless of where along the segment the target is found.**
6. There can be gaps in the sequence of Priorities at a branch. For example, 2, 5, 9, 10, are valid Priorities at an End Condition Branch.
   a. **True – There can be gaps in the sequence of Priorities at a branch. For example, 2, 5, 9, 10 are valid Priorities at an End Condition Branch.**

b. False

7. If an End Component in an End Component path succeeds, the path succeeds and the branch stops processing.

   a. True

   b. **False – Processing continues with the next Component in the Path.**

8. Every End Component in a Path must succeed in order for the Path to succeed and the End Components in the Path to be placed.

   a. **True – Every End Component in a Path must succeed in order for the Path to succeed and the End Components in the Path to be placed.**

   b. False

9. A point not seeking intersection is considered a failure.

   a. True

   b. **False – A point not seeking intersection is not considered a failure. It is considered a conditional success: it allows continuation of processing. If further points succeed, it is placed with the other successful point(s).**

10. As soon as a point in an End Condition component succeeds, the component is considered a success and stops processing other points in the component.

    a. **True – As soon as a point in an End Condition component succeeds, the component is considered a success and stops processing other points in the component. Processing immediately begins for the next component, if any, in the path. Note that this is different than how Components process.**

    b. False

11. Arguably the primary advantage of Feature Definition targets versus Linear Targets is that with Feature Definition: it is easy to target and manage lots of data by function-specific Feature Definitions --ROW, Retaining Walls, utilities, etc.

    a. **True**

    b. False

Summary

In this course you have learned how to create template end condition components for various side slope design requirements including multiple cut and fill slopes, ditches, and retaining walls. You have also learned how to add horizontal constraints to target specific features like a right-of-way, to set priorities, and to test the end conditions to verify the end condition functionality.