



# OpenRoads Designer: Templates

## ODOT Template Library

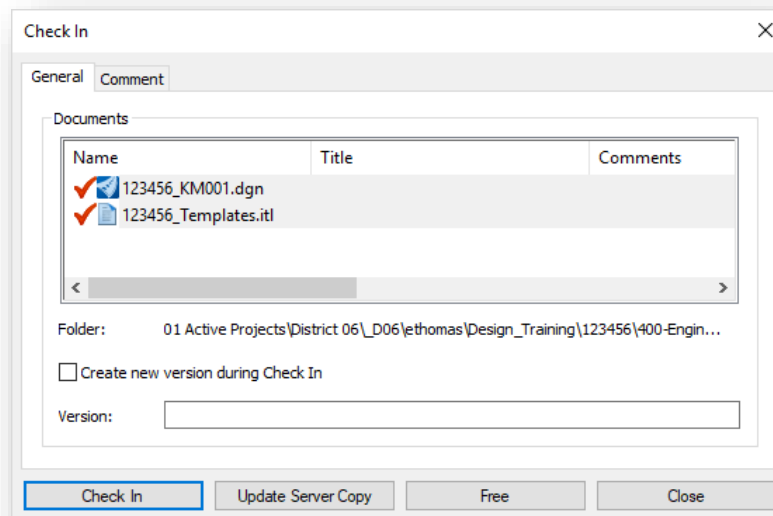
Templates are used to define the typical section for each alignment. Multiple templates can be used to define different typical section designs, or the template geometry can be modified using a variety of corridor editing tools.

When the WorkSet for a new project is created in ProjectWise, the **OHDOT\_Templates.itl** library is copied from the OHDOT Standards to the **990-WorkSet Standards\Template Library\** folder and renamed using the PID number for the project.

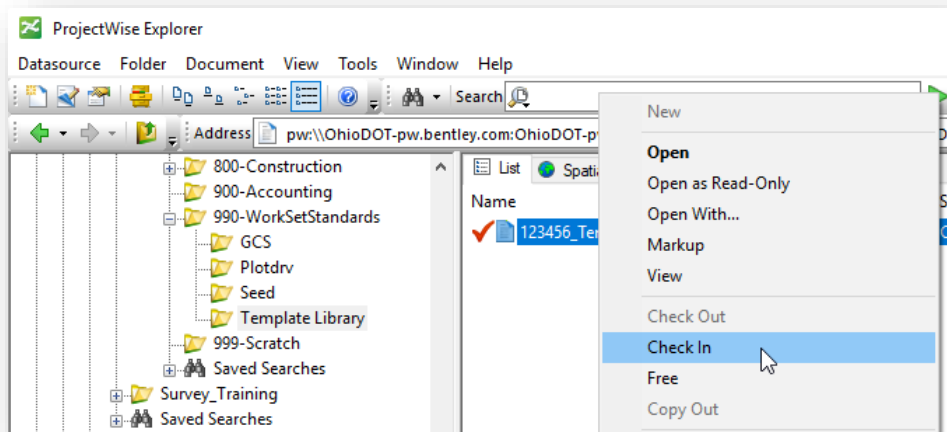
For users working outside of the ProjectWise environment, when the **Startup.mvba** application is processed on a new WorkSet, the template library is copied to the **990-WorkSet Standards\Template Library\** folder as part of the program operation.

## ProjectWise Template Library Check-in/Out

When a template library is opened in ProjectWise, the file is checked out to the current user and cannot be viewed or edited by other users. When the **Create Templates**, dialog is closed, the software does not ask the user to check the template library back into ProjectWise. The user is prompted to check the template library back into ProjectWise when the design file is closed, as shown below.

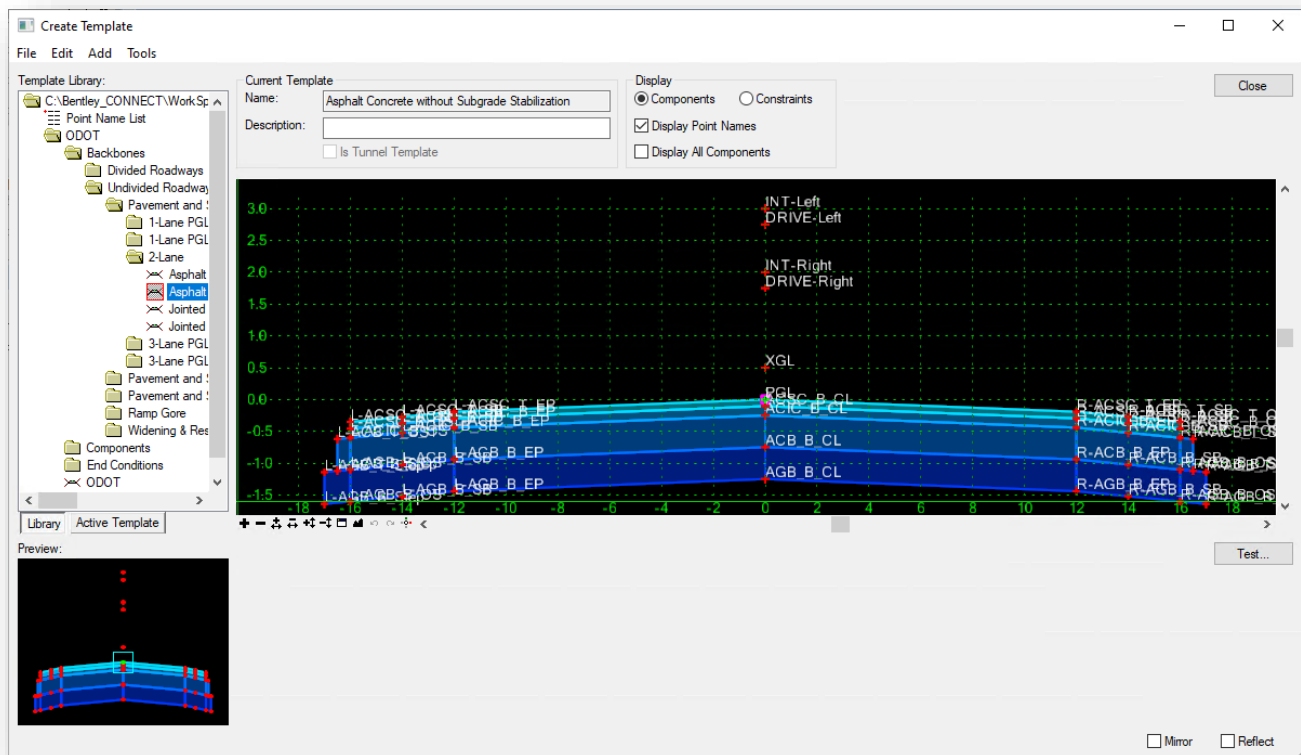


The template library can be checked back in after closing the template library from the **ProjectWise Explorer** dialog as shown below.



## Create Template

Templates are created and edited in the template library by choosing **Template > Create Template** from the **Corridors** tab.



The dialog is split into several areas. Each area is briefly described on the following pages. See the online help for information not presented here.



## Menu Commands

### File

The **File** menu contains familiar commands for creating new Templates and new folders within the template library, as well as commands to save and open template libraries,

### Edit

The **Edit** menu contains commands used when editing templates such as **Undo, Redo, Cut, Copy, Paste,** etc....

### Add

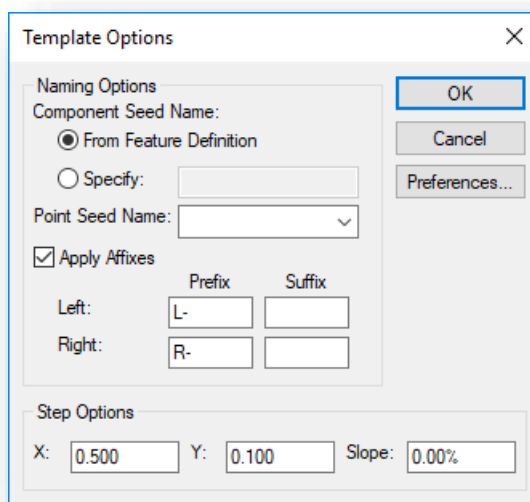
The **Add** menu contains commands to add points and components to the active template.

### Tools

The **Tools** menu contains various tools for template creation. The Options and Dynamic Settings menu items are described on the following page.

### Tools > Options

Selecting **Tools > Options...** displays the *Template Options* dialog which allows you to specify naming and step options for the current template. Use the **Dynamic Settings** dialog, detailed on the following page, to toggle these settings on/off as you design the template.



### Naming Options

#### Component Seed Name

**From Feature Definition** indicates, when selected, the component name is derived from the style specified when the component is created.

**Specify** indicates, when selected, the component name is specified in the field.

**Point Seed Name** allows you to specify the seed name for a point in the current template. Since point names must be unique, the seed name is appended with a different number each time a point is created.



### Apply Affixes

Indicates when checked “on”, that affixes are applied.

**Left Prefix/Suffix** specifies which prefix/suffix should be added to a point or component when the value is on the left of 0.0 (absolute).

**Right Prefix/Suffix** specifies which prefix/suffix should be added to a point or component when the value is on the right of 0.0 (absolute).

### Step Options

These options are used to specify precision step input for components and points. These values can also be accessed from the *Dynamic Settings* dialog.

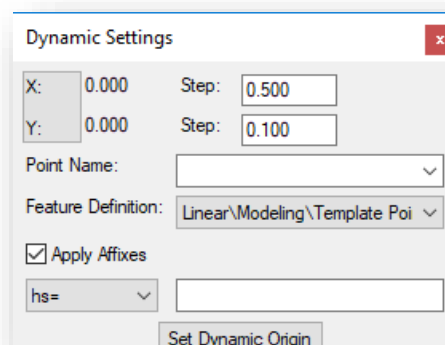
**X Step** specifies the step increment for the X distance from the dynamic origin. If this value is non-zero, then the cursor moves to the horizontal increments of the specified value as measured from the dynamic origin.

**Y Step** specifies the step increment for the Y distance from the dynamic origin. If this value is non-zero, then the cursor moves in vertical increments of the specified value as measured from the dynamic origin.

**Slope** specifies the step increment for slope from the dynamic origin. If this value is not 0, then the cursor moves in slope increments of the dynamic origin. Slope is applicable only when the dynamic settings are in horizontal/slope mode.

### Tools > Dynamic Settings

Select **Tools > Dynamic Settings** to display the *Dynamic Settings* dialog, which is used for precision input of template components and to assign point names and styles when defining components.



**X:Y/X:Slope** toggles between X,Y step lock and X/Slope step lock.

**X Value** displays the current X distance from the dynamic origin.

**X Step** specifies the step increment. If this value is non-zero, then the cursor moves in horizontal increments of the specified value as measured from the dynamic origin.



**Y value** displays the current Y distance from the dynamic origin.

**Y Step** specifies the step increment. If this value is non-zero, then the cursor moves in vertical increments of the specified value as measured from the dynamic origin.

**Point Name** specifies the name of the point to be placed.

**Point Style** allows you to set the style of the point. The style is used to draw feature lines in the plan view. For ODOT projects, the style **CM\_P\_Feature\_Line** should be used as the default style for most template points unless you want to assign a specific style that can be plotted in the display view 3D model.

**Apply Affixes** specifies, when selected, that point name affixes (prefix or suffix) are to be applied to the point name when placing points. The **Affixes** are defined in the *Template Options* dialog.

The **Key-in** pull down is used to specify the type of key in to be performed.

- XY= key in absolute coordinates
- DL= key in delta coordinates from last point placed (defaults to the dynamic origin if it is the first point of a component.
- HS= key in horizontal delta distance and slope from last point placed.
- VS= key in vertical delta distance and slope from last point placed.
- OL= key in delta coordinates from dynamic origin.
- OS= key in horizontal delta distance and slope from dynamic origin.

**Set Dynamic Origin** sets the location of the dynamic origin.



## Template Library

On the left side of the *Create Template* dialog is the **Template Library** area which contains a Windows-like folder structure. This area is used to manage the content of the current template library. Two tabs are supported for the **Template Library** area as described below.

### Library

When the **Library** tab is selected, the **Template Library** displays the contents of the current template library in a familiar Windows folder structure as shown at right. The **Template Library** area also supports common Windows functions like drag, drop, cut, paste, etc.

If you double-click a template name, it becomes the active template.

When a template is selected from the list, the **Preview** window is used to display a preview of the template.

Double-click the template name to set it as the active template.

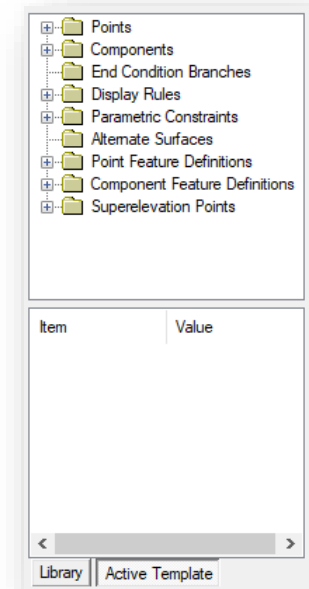
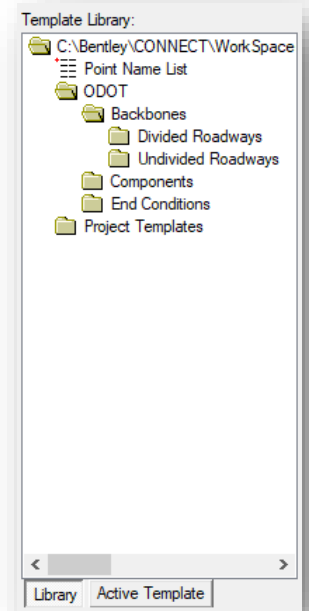
The **ODOT** category contains various components and end conditions that can be used to build templates for your project.

The **Project Templates** category is used to create new templates for the active project.

### Active Template

When this tab is selected, the **Template Library** area displays the contents of the active template as shown at right.

The **Active Template** option is used to review and edit template points, components, and parameters in a table format.



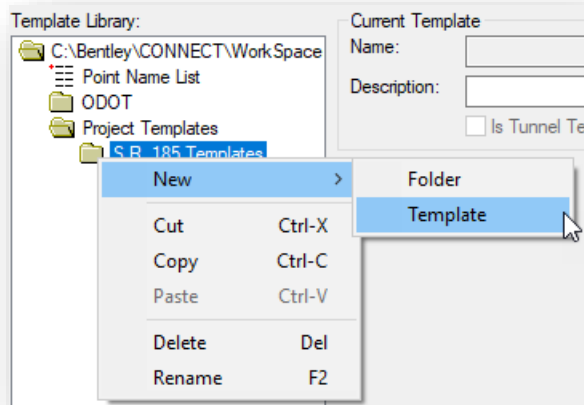


## Creating New Folders and Templates

Templates can be organized into folders in the active template library. The Project Templates folder is intended to be used to create new folders and templates for a specific project.

To create a new folder, right click on the Project Templates folder and choose the **New > Folder** option.

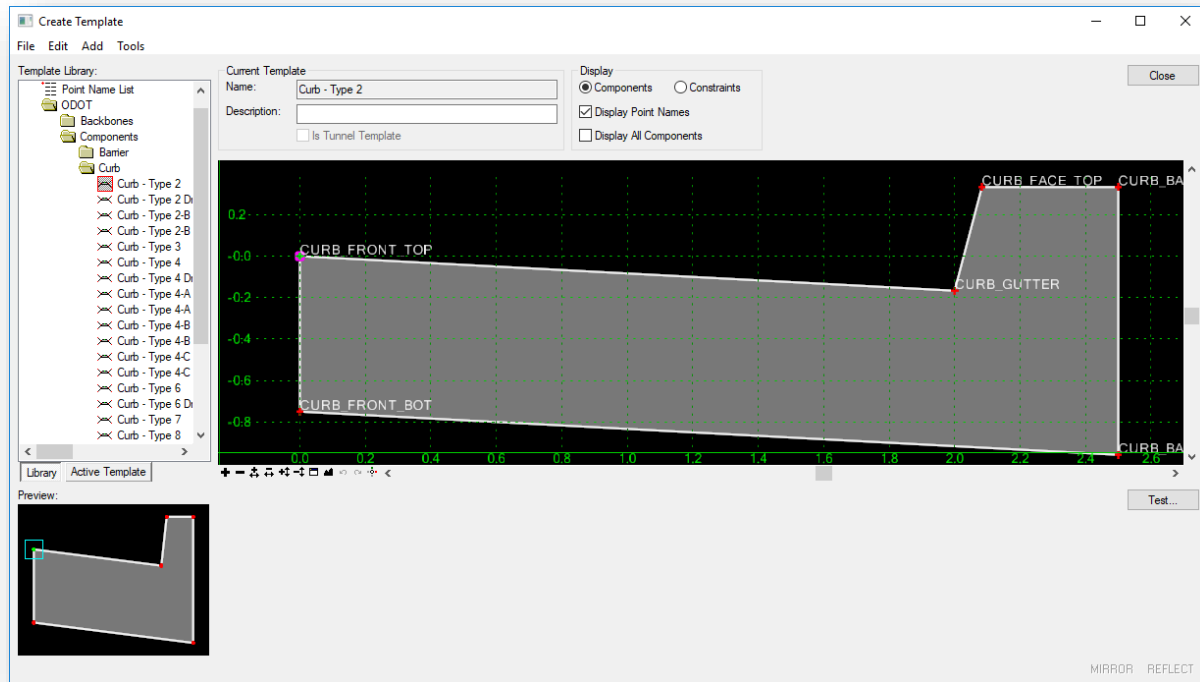
To create a new template, right click on the folder and choose the **New > Template** option.



You can also cut, copy, and paste templates from the ODOT folders to the desired Project Templates folder.

## Current Template

The right side of the **Create Template** dialog is used to display and edit the current template.





### Name

Shows the name of the current template.

### Description

Allows you to review/edit the current template description.

### Display

#### Components/Constraints

This option is used to set the display mode of the template. If **Components** are selected, then the template components are displayed. If **Constraints** are selected, then the point constraints for the template are displayed. Template points are always displayed.

#### Display Point Names

Toggle this option “on” to display the template point names.

#### Display All Components

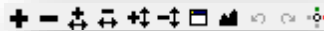
When this option is checked on, any components that are “hidden” are displayed with dashed lines.

### Current Template Window

The Current Template Window is used to display and edit the current template. The window uses its own graphics engine and is not a MicroStation view.

A magenta-colored box is used to designate the origin of the template.

At the bottom of the **Current Template Window** are view commands like the MicroStation view commands.



The two buttons that look like MicroStation View Previous and View Next commands are undo and redo commands used on template operations. The CTRL-Z keyboard shortcut can also be used to undo the last template operation.

Within the **Current Template Window**, you can right-click to access commonly used template functions. You can also double click template components for editing purposes.





## Template Points

When the template is processed using the corridor modeling command, the points of a template are connected to form *breakline* features. Each point will be connected longitudinally to the next template drop location to form 3-D longitudinal breakline surface features. The breaklines are drawn using the symbology defined with the **Feature Definition** that is assigned to each point.

Since the template points are connected to generate the completed surface, establishing a standard name for template features that are used on each template is very important to the quality of the proposed design surface. For example, if a pavement point is named ACSC\_T\_CL on the first template, and it is named PAVT\_SURFACE on the next, the software will not automatically connect these points to generate the breakline feature. ***Following a point naming convention is important to obtain the best results with Corridor Modeling.***

It is equally important to ensure that the feature definition assigned to points is consistent from template to template.

Each template point name must be unique to the template. The software automatically appends a number to the point name if the same name is used during template creation.

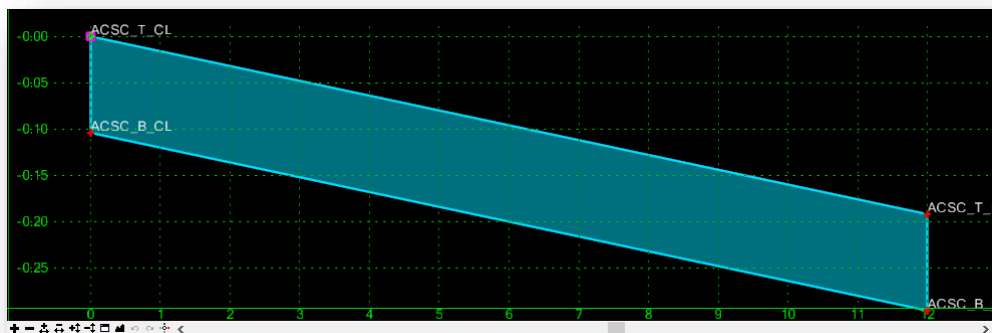
There is no limit to the number of points in a template.

## Template Components

A component is a set of points that define an open or closed shape. Each component, whether open or closed, can represent a different material or area of interest. Components are named and have an assigned feature style. There are 6 types of components as described below:

### Simple Components

A simple component typically represents a section of pavement. It's a closed parallelogram (4 constrained points) that is defined by the slope and thickness.

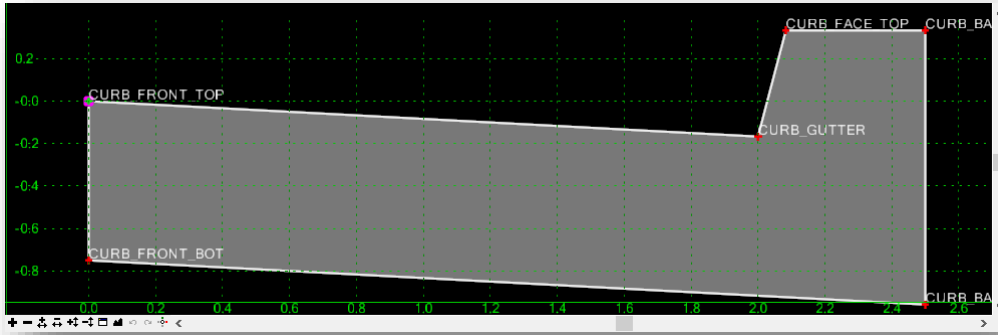


### Constrained Components

A constrained component consists of points that are all restricted to the movement of the first point. A constrained point is typically used to manage the behavior of other points in the template. When a point (parent) is moved, any constrained point (child) also moves. This restriction only affects the offset and



elevation (x,y) of the restrained point and the relationship is unidirectional (movement of child point does not move the parent point). Fully constrained points are displayed with a red + symbol.



### Unconstrained Components

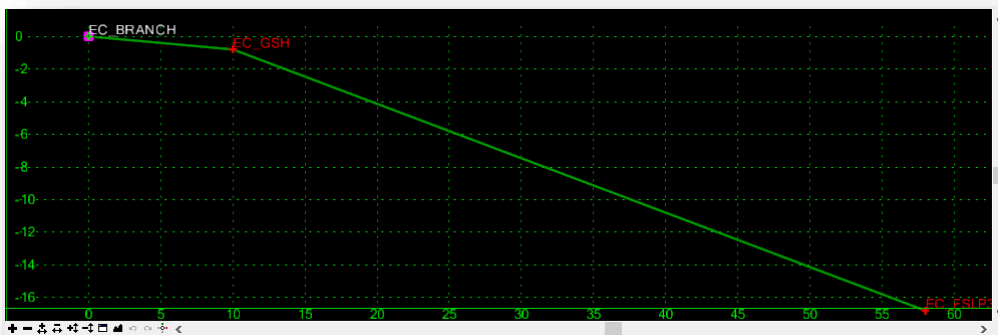
An unconstrained component is open or closed-shaped with no movement restrictions. Moving one point has no impact on the other points that make up the component.

### Null Point

A null point is a template point that is purposely not related to a specific component. It's most often used as a reference for controlling other points.

### End Condition

An end condition is a special open-shaped component that targets a surface, an elevation, or an alignment. End conditions are typically used to define cut and fill slope solutions.



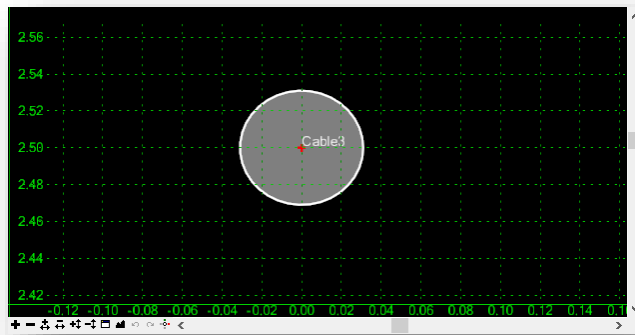
### Overlay/Stripping Components

A specialized component type that is used for defining pavement planning and overlay materials.



## Circle

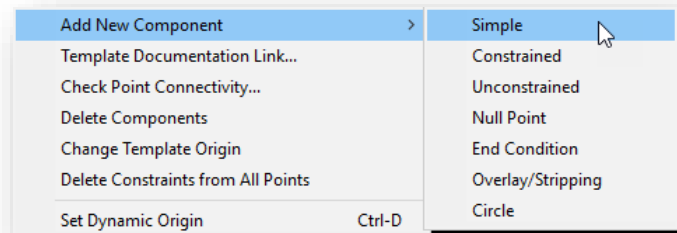
Circular components can be created as shown in the example below.



## Creating Template Components

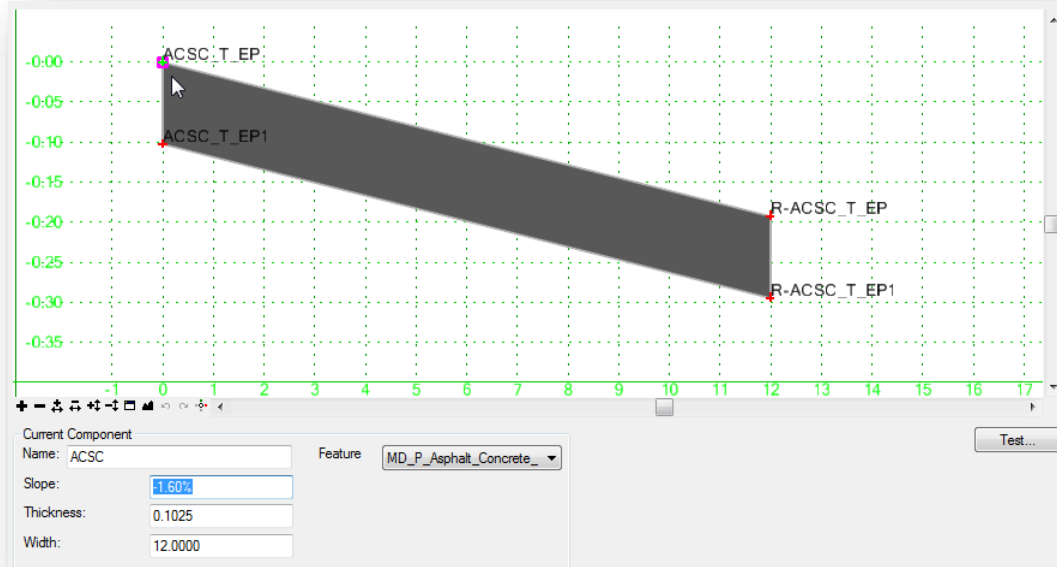
ODOT provides an example template library that contains several commonly used components. These components can be used to build templates. It may also be necessary to make specialized components to meet specific project requirements.

Components are added to the current template by selecting the component that you wish to add from the **Add** pull-down menu, or by right-clicking in the **Current Template** window to access the **Add New Component** menu shown at right.

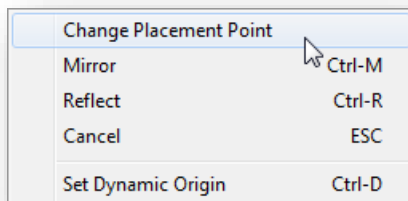


As new components are created, the **Current Component** portion of the *Create Templates* dialog is used to define the **Name**, **Style**, and any other relevant parameters for the component type being created.

In the example below, a simple component is created representing a single lane of pavement.



Before placing the component in the template, various options are available to control how the template is placed by right-clicking in the template window to reveal the menu shown below.



**Change placement Point** can be used to change which component point is used as the point placed at the point you specify. Each time this command is selected, the placement point will move in a clockwise direction to the next point in the component.

**Mirror** will create the original component as well as a mirrored component reflected horizontally about the template origin.

**Reflect** will create a single component reflected about the template origin.

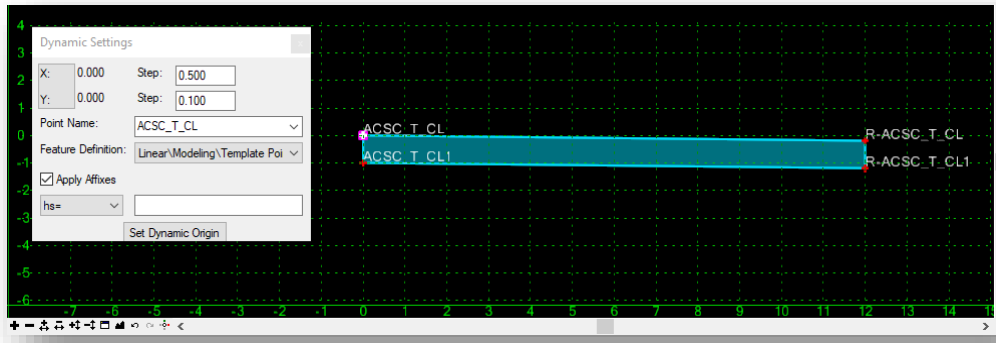
The **Mirror** and **Reflect** options are toggles that are turned on or off and remain active. The on/off status of the **Mirror** and **Reflect** options can also be turned on/off by toggle switches at the bottom right corner of the *Create Templates* dialog.

Select **Set Dynamic Origin** to set the location of the dynamic origin. The dynamic origin is indicated by the purple box in the template display window.



## Point Names and Affixes

When components are added to a template, the points that make up the component are named according to the **Point Name** and the **Pont Style** defined in the *Dynamic Settings* dialog (**Tools > Dynamic Settings**). The **Point Name** is automatically appended with the number 1, and the appended number is automatically incremented for each point in the component as shown in the example below.

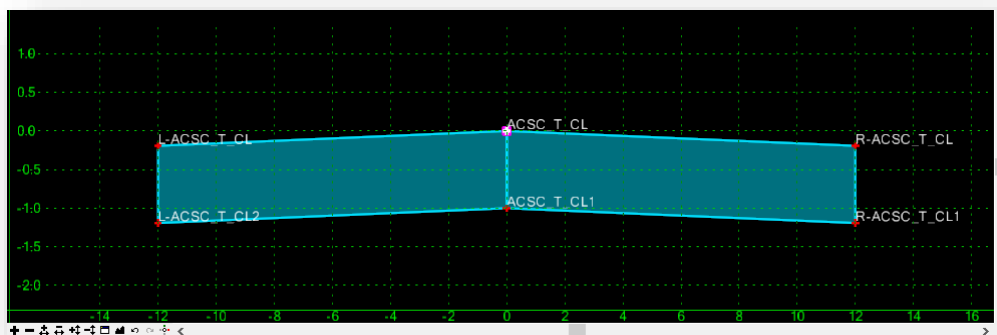


The **Point Name** can be keyed in or selected from the drop-down menu. When selected from the menu, the **Point Style** is automatically assigned for the point according to ODOT standards.

The point and component name designations are a bit cryptic. A listing of the point and component names can be reviewed in the following document in the OHDOT Standards:

[..\OHDOT\Standards\Template Library\OHDOT\\_RoadwayTemplateDefinitions.xlsx](#)

The **Apply Affixes** option is used to add a user defined prefix or suffix to the point name. In the example below a simple component was added with the **Mirror** option toggled on (as the component is being placed, right-click to reveal a menu to choose the **Mirror** option). The prefix of “L-” is applied to the points on the left side of the template. The prefix of “R-” is applied to the points on the right side of the template.



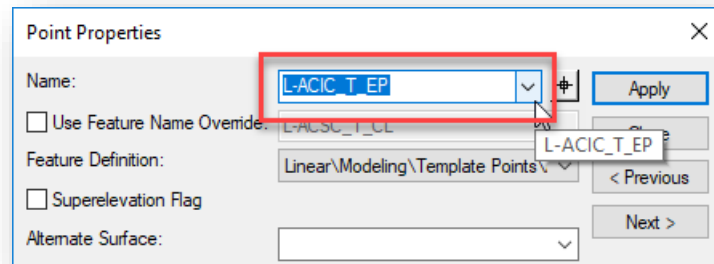
The affixes are defined in the *Template Options* dialog.

**Tip:** As a rule, affixes should not be applied to templates that contain individual components that will be used as “building blocks” for complete roadway templates. This is because the component may be used on either the left or right side of the template. Affixes should always be applied when creating completed templates for use with Roadway Designer.



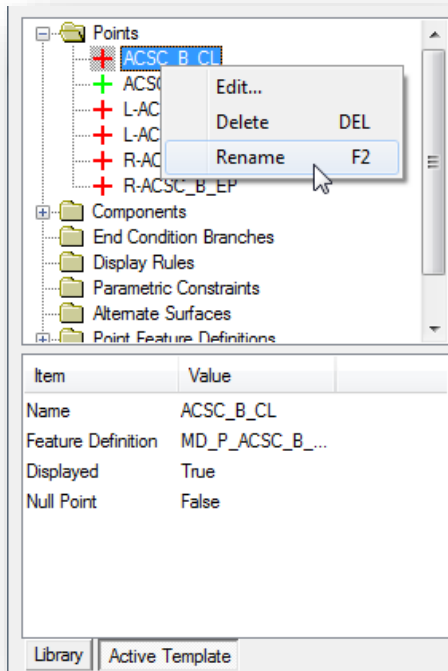
Note that all the points of the component are named using the currently defined **Point Name**. In the example on the previous page, the points were placed using the **ACSC\_T\_CL** (Asphalt Concrete Surface Course Top Center Line) point name. After the component is placed, it is necessary to rename the points and assigned feature definitions to ensure that the points and 3D line strings are drawn using the appropriate standards. This can be done by double-clicking on each point to access the *Point Properties* dialog shown below.

The **Name** can be typed in or selected from the drop-down list. If the **Name** is selected from the list, the associated **Feature Definition** is automatically defined. The *Point Properties* dialog is covered later in this document.



## Reviewing and Editing Point and Component Names

Point and component names can easily be reviewed and edited using the **Active Template** tab as shown below.



Right-click on a point or component name to reveal the menu shown at left. Choose **Rename** to rename the selected element.

The **Item** and **Value** list, shown below the elements, is used to review various parameters but cannot be used for editing.

Choose the **Edit** option for more editing operations.

**Points** and **components** can also be edited by double-clicking on a point or component in the template window. The *Point Properties* and *Component Properties* dialogs are covered later in this document.

**Note:** The **Feature Definition** cannot be edited using the **Active Template** editing options.



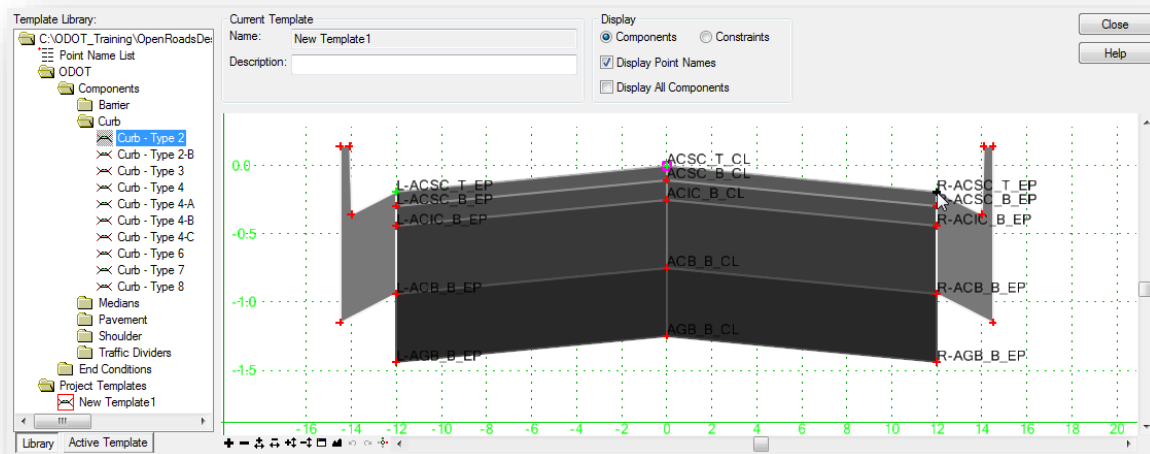
## Drag and Drop Component Operations

A previously created template component can be included in the active template by use of drag and drop operations. Select a previously drawn component from the active template library and drag the component into the **Current Template** window. The component dynamically floats on the cursor. Release the mouse button to drop the component at the desired location.

While dragging the component in the **Current Template** window, right-click anywhere in the **Current Template** window to access the placement options discussed previously (**Mirror**, **Reflect**, etc...)

When connecting two components by dragging a component into the active template, the connecting point will change to a white colored plus sign prior to placing the component to indicate that the points are coincident. The point name from the previously placed component will be retained. Point names can be edited at any time during the template creation process.

In the example below, a previously defined curb and gutter component is placed at the edge of pavement in the active template (**New Template1**) by dragging the **Curb - Type 2** component and dropping it on the right edge of pavement point. The **Mirror** option is turned on, with **Apply Affixes** also toggled on, to place the curb and gutter on both sides of the pavement.





## Component Editing

Components are edited by double-clicking on a component to open the **Component Properties** dialog shown below.

Component Properties

Name:  #

Use Name Override:

Description:

Feature Definition:  ▾

Display Rules:  Edit...

Parent Component:  #

Exclude From Top/Bottom Mesh  Closed Shape

Vertex Fillet Tangent Lengths  
Select points to apply fillet tangent length to:

Name	Tangent Length
ACSC_T_CL	0.000
L-ACSC_T_EP	0.000
L-ACSC_B_EP	0.000

Fillet tangent length:

Apply tangent length

Apply Close < Previous Next >

See the online help for details on each of these parameters.

Issue a right mouse click on a previously drawn component to access the menu shown below.

- Add New Component >
- Template Documentation Link...
- Check Point Connectivity...
- Delete Components
- Change Template Origin
- Delete Constraints from All Points
- Edit Component...
- Insert Point
- Insert Arc
- Unmerge Component Points
- Set Component Display Rules
- Delete Component
- Set Dynamic Origin Ctrl-D

See the online help for details on each of these functions.





## Mouse Wheel Functions While Creating Templates

---

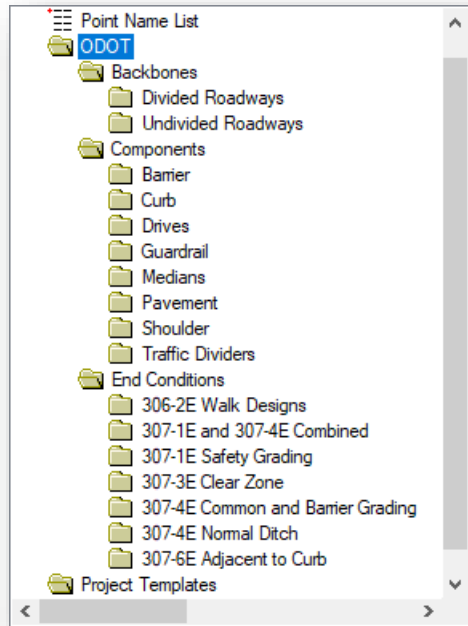
The mouse wheel has the following functions when creating templates:

- Scroll the wheel to zoom in or out in the template window.
- Hold down the **Shift** key while scrolling the mouse wheel to zoom in or out in the vertical direction only.
- Hold down the **Ctrl** key while scrolling the mouse wheel to zoom in or out in the horizontal direction only.



## The ODOT template Library

ODOT provides a template library with many template components defined that can be used as the starting point for creating and editing templates for your project. When a new WorkSet is created, the ODOT template library is copied to the **990-WorkSet Standards\Template Library\** folder.



The ODOT template library has two main folders, **ODOT** and **Project Templates**.

The **ODOT** folder contains various **Components** and **End Conditions** that can be used to create completed templates.

The **Project Templates** folder is an empty folder to be used by the designer to create templates for the project.

The contents of the **ODOT** folder are summarized below:

### Backbones

The “backbone” is defined as the pavement and shoulder portion of the template. The **Backbones** folder contains several common pavement and shoulder configurations for divided and undivided roadways. A variety of configurations are available for superelevated and non-superelevated roadways.

### Components

The **Components** folder contains individual design components that can be used as the “building blocks” for a complete template. A Type 2 Curb and Gutter is an example of a component.

### End Conditions

Several end conditions are defined for cut and fill conditions according to ODOT design standards.



## Template Creation Workflow

---

While the process to create templates will have many variables, the general procedure to create templates is summarized below:

- Open the template library
- Create a new template or copy a template. Templates can be copied from one library to another by selecting **Tools > Template Library Organizer** from the *Create Templates* dialog.
- Drag and drop template backbones or individual components into the active template
- Create/edit the components as necessary
- Merge components as desired
- Review the point names and point name overrides
- Review the component names and component name overrides
- Test the template behavior
- Save the template library



## Advanced Template Design

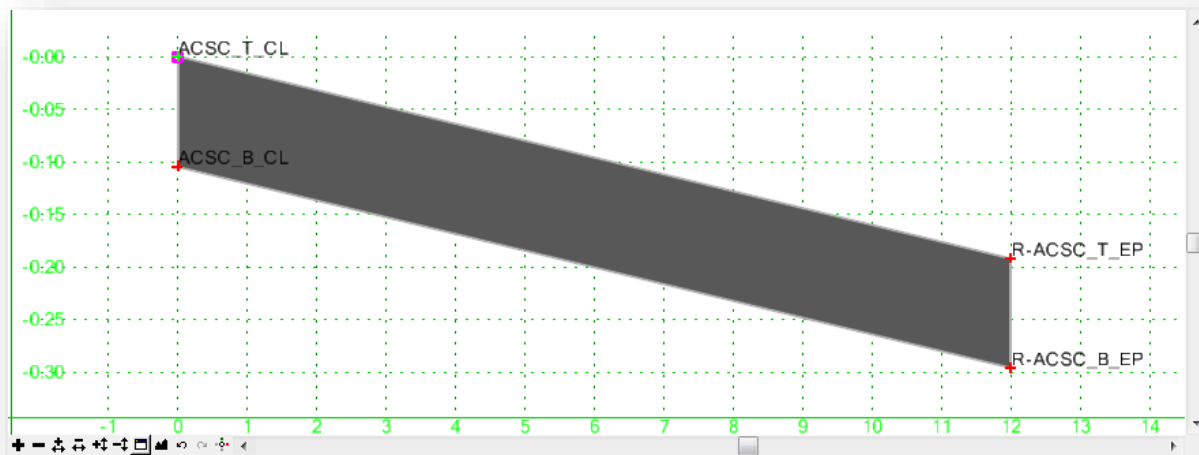
The corridor modeling tools provide a lot of options that can be used to create very complex templates that handle a variety of design scenarios. However, care should be taken not to create overly complex “Franken-Templates” to create a single template to account for all the design elements that occur over the length of the project. Overly complex templates can be hard to edit and even harder to diagnose when things don’t work as expected. It’s better to define a few simpler templates for specific station ranges than it is to define one overly complex template.

In this section, we will take a deeper look at the available template parameters.

### Point Constraints

Point constraints are used to manage the behavior of points in a template. They are used so that if a point is moved in a template, either by the user editing the template or by the application of a horizontal or vertical control during design processing, all the points related to the point being moved behave in a rational and predictable manner.

For example, when a **Simple** component is inserted into the active template, it is comprised of four points which are connected to define the component as shown in the example below.



Three of the four points are constrained with relation to the insertion point. If the original insertion point is moved, the other three points move in relation to the first point.

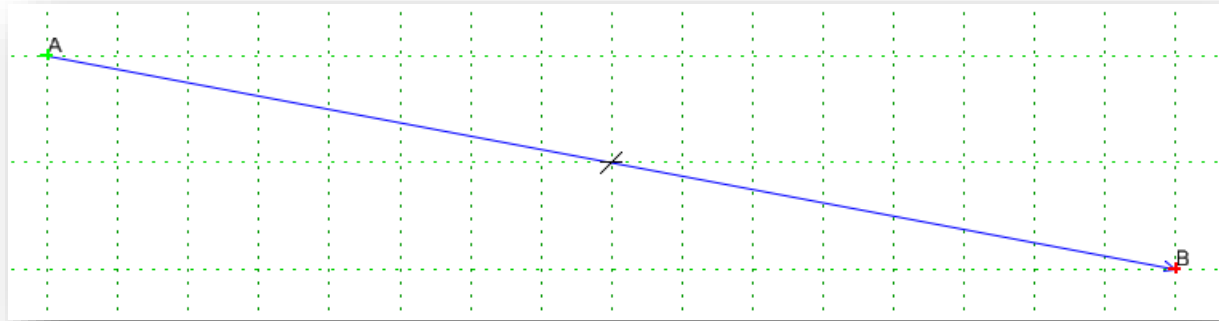
A point can have at most two constraints. If two constraints are defined, the point is said to be “fully constrained”. A point that is fully constrained is represented by a red plus sign. In the example above, points **R-ACSC\_T\_EP**, **R-ACSC\_B\_EP**, and **ACSC\_B\_CL** are fully constrained.

A point that is partially constrained, meaning that it has only one constraint on it, is shown as a yellow plus sign.

An unconstrained point is shown as a green plus sign. In the example above, point **ACSC\_T\_CL** is unconstrained.

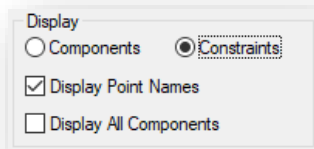


Point constraints are two-dimensional and one-way. Two-dimensional means that the constraints can only affect the points offset and elevation (x and y coordinates in the cross-section view). One-way means there is a child-parent relationship between points. In other words, if point B is constrained by point A, point A is said to be the parent of point B and moving point A will affect point B, but you cannot move point B to affect point A. The example below shows a sample of this where the blue arrow indicates a parent/child relationship from point A to point B:



Constraints can also be labeled so that during processing, the value(s) of labeled constraint(s) may be changed. For example, the template may have a constraint label called **ACSC Thickness** that controls the thickness of the first pavement layer. This value can then be changed over one or more station ranges during the corridor modeling process.

The constraints for all the points in a template can be displayed by choosing the **Display Constraints** option on the *Create Template* dialog.



Constraints are displayed as blue lines between the parent and the child point. At the midpoint of the constraint line, there is a short white line designating the constraint type as follows:

- Horizontal line = Horizontal Constraint
- Vertical line = Vertical Constraint
- Sloped line = Slope Constraint

Multiple lines will be shown for points with two constraints.

Floating the cursor over a point will temporarily display an arrow designating the direction of the constraint (the arrow points to the child point) and a pop-up menu displaying information about the point.



## The Point Properties Dialog

The **Point Properties** dialog is used to review and edit various properties associated with the template points, including the definition of the point constraints. To access the **Point Properties** dialog, in the **Create Template** dialog double-click on the point in the **Current Template** window. The **Point Properties** dialog, shown below, is opened.

The dialog contains the following options:

**Name** displays the point name. The point name can be keyed-in or selected from the drop-down list. If the point name is selected from the list, the corresponding **Feature Definition** is automatically assigned. Point names must be unique to the template.

**Use Feature Name Override** displays the name of the feature that will be created in the surface to correspond to the point. This field is optional. If it is blank, then the point name will be used as the feature name.

- The option is intended primarily for end condition components to create connectivity from one station to the next when the template end conditions change. For example, if you want all surface tie-in points to belong to one feature, then set the feature name of all the ending end condition points on each side of the template to the same name (i.e. all Cut and Fill end points on the right would be given the feature name R-Tie and all the ones on the left would be given the feature name L-Tie).
- If the point has a **Feature Name Override** defined, the point name is displayed in red in the Template window.

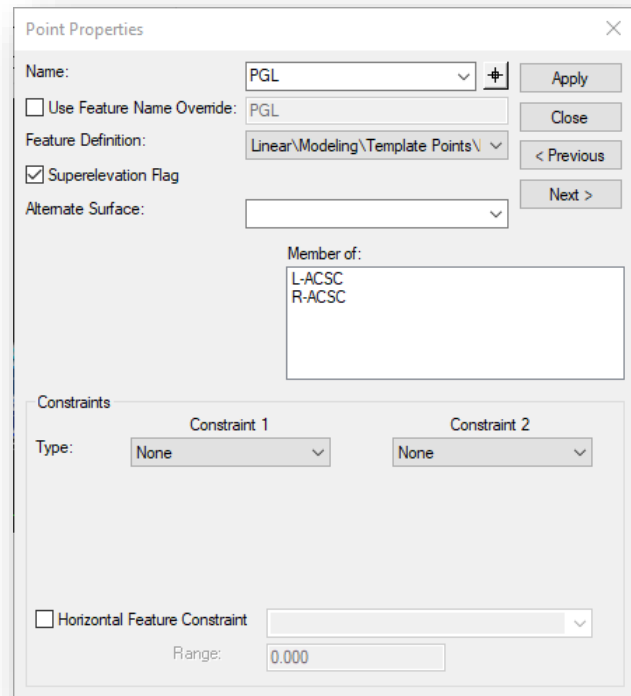
**Feature Definition** is used to define the feature definition of the point. If no style is specified, then it comes from the first component of which the point is a member.

The **Superelevation Flag** is used to identify the point as a candidate to be used for assigning superelevation control lines. This option should only be toggled on four pavement points on the surface of the pavement.

**Alternate Surface** allows you to specify the name of an alternate surface for a point. You may choose multiple alternate surface names. Select from the list of available surfaces in the active template.

**Member Of** indicates in which components the point is included.

### Constraints





The **Constraints** portion of the **Point Properties** dialog allows you to review/edit the constraints on a point.

**Type** specifies the constraint type: **None**, **Horizontal**, **Vertical**, **Slope**, **Vector-Offset**, **Project to Surface**, **Project to Design**, **Horizontal Maximum**, **Horizontal Minimum**, **Vertical Maximum**, **Vertical Minimum**, **Angle Distance**. The constraints are described in more detail below.

Setting up the appropriate constraint types is critical to achieve the desired results when pavement layers, superelevation, and transitioning are introduced to the design.

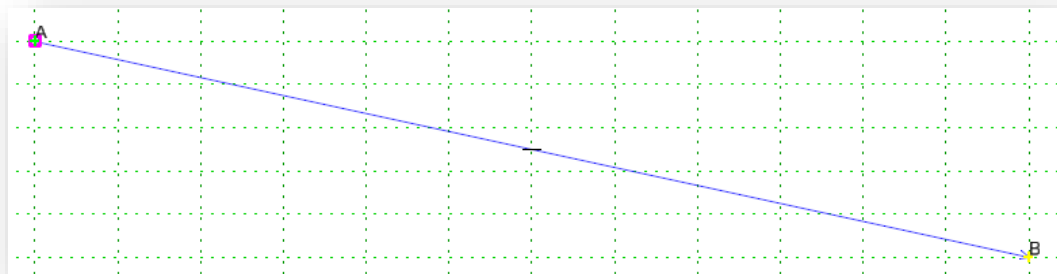
**Label** displays the optional label for the constraint. Constraints that are labeled can have their value changed during design processing. The same label name can be assigned to more than one constraint and more than one point. See Section [10.4](#) for details.

**Horizontal Feature Constraint** is used to allow a point to target elements in the design file that are defined with the specified **Feature Definition** within a specific **Range**. The use of this option is not recommended.

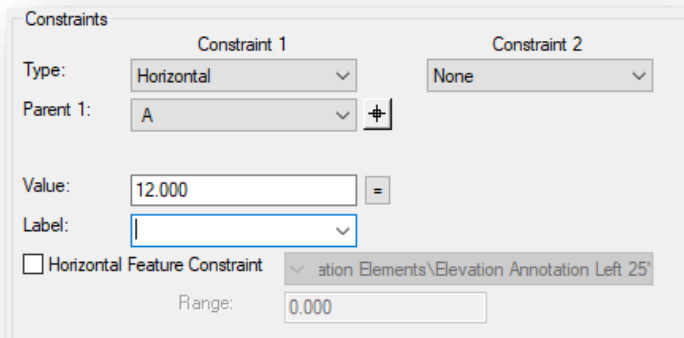
## Point Constraint Types

### Horizontal

The child point remains at the given horizontal distance from the parent point.



In the example above, point **B** has been constrained to point **A** with one constraint as shown below.



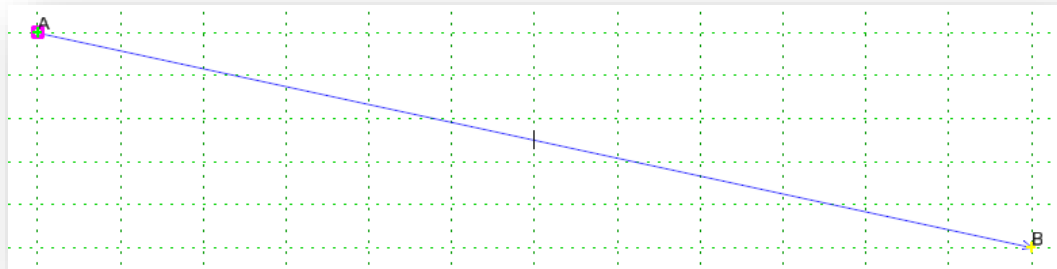
If point **A** is moved horizontally, point **B** will maintain its relative distance from point **A**.

If point **A** is moved vertically, point **B** is unaffected.

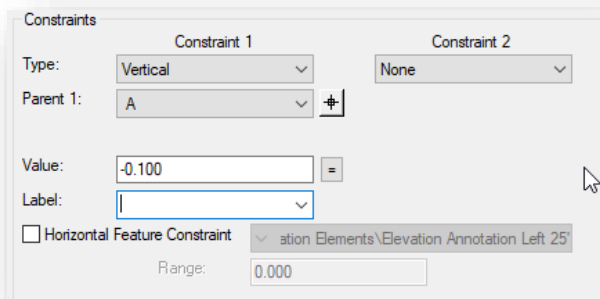


## Vertical

The child point remains at the given vertical distance from the parent point.



In the example above, point **B** has been constrained to point **A** with one constraint as shown below.

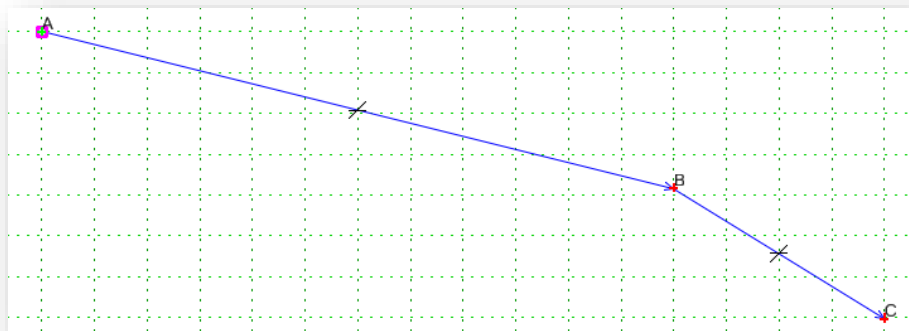


If point **A** is moved horizontally, point **B** is unaffected.

If point **A** is moved vertically, point **B** will maintain its vertical relationship to point **A**.

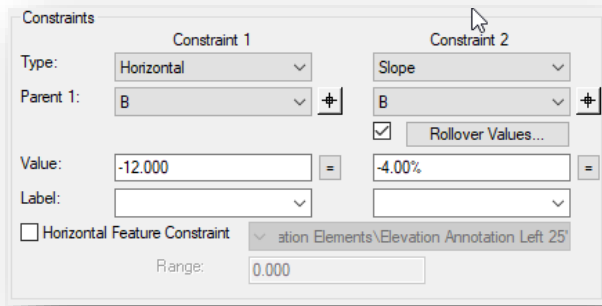
## Slope

The child point will maintain the given slope from the parent point.



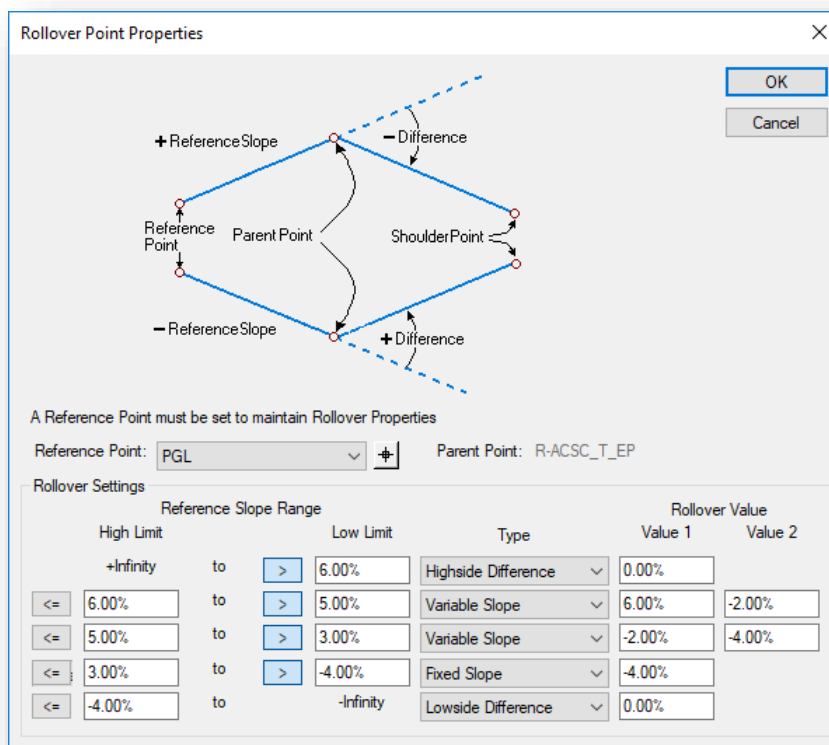
In the example above, point **C** has been defined with a **Horizontal** and a **Slope** constraint as shown below:





Slope constraints are absolute. Slopes going from lower-left to upper-right are positive regardless of whether the child point is to the left or right of the parent.

Slope constraints can also have **Rollover Values** assigned to them. Rollover values are used to set the slope constraint based on a variety of high side and low side slope parameters. A **Reference Point** is specified which defines the controlling slope to the parent point.



The example above documents the **Rollover Values** can as defined in ODOT's example templates for the shoulder break point behavior. See the online help for more information on the available parameters.



### Horizontal Maximum

The child point has two parent points and remains at the given horizontal distance from the parent point that is farthest to the right (has the maximum horizontal or X value).

### Horizontal Minimum

The child point has two parent points and remains at the given horizontal distance from the parent point that is farthest to the left (has the minimum horizontal or X value).

### Vertical Maximum

The child point has two parent points and remains at the given vertical distance from the parent point that is highest (has the maximum vertical or Y value).

### Vertical Minimum

The child point has two parent points and remains at the given vertical distance from the parent point that is lowest (has the minimum vertical or Y value).

### Vector Offset

The child point has two parent points and will be projected onto the vector defined by the two parents. If the offset is not zero, then the child point will maintain a perpendicular offset from the parent vector at the specified offset value. Negative values indicate an offset to the left of the vector defined by the parent points. Positive values indicate an offset to the right.

### Project to Surface

This constraint must be used in conjunction with one of the previously defined constraints. The other constraint will define the projection direction. The child point will then be projected to the surface with the name or parametric label given when the design is processed. If the surface does not exist, or no solution is found, then the point will remain where it is placed in the template.

### Project to Design

This constraint is like the Project to Surface, except that the point is projected to the design surface of the template. A projection value is given to indicate whether the projection is to be to the left or to the right. The point must also be constrained by one of the previous constraints, excluding the Project to Surface, so that a direction for the projection may be determined. A negative value limits the projection to the left of 0; a positive value limits the projection to the right. A value of 0 will seek to the left and to the right of 0 to project the point. If no solution is found, then the point will remain where it is placed in the template.

### Angle Distance

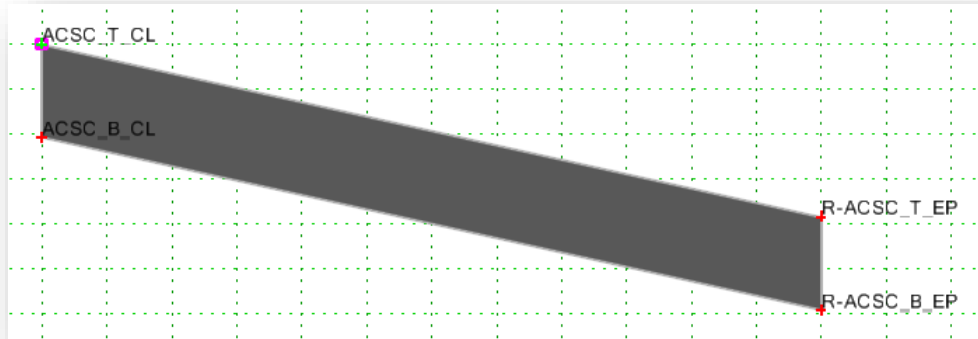
This constraint takes two parent points, a distance, and an angle. The selected point is then fully constrained to the location defined by the first parent, and the angle from the first parent relative to the vector defined by the two parent points. This constraint creates a rigid-body rotation. When selected, no other constraint types are available.



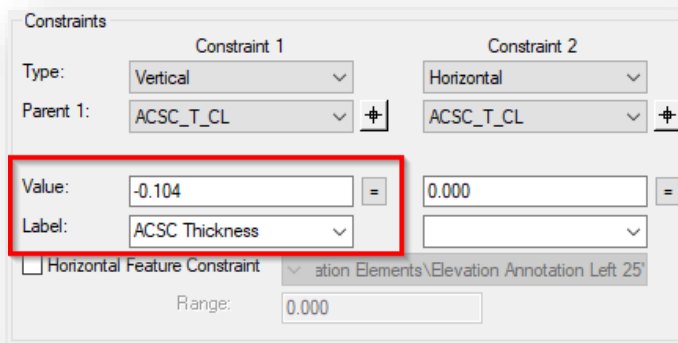
## Constraint Labels

Constraint **Labels** can be assigned for points in your template that may have a variable value in your design, or a different value from one project to another. The value for the **Label** can be modified in the corridor modeling process. Assigning labels will allow your templates to be more flexible and therefore make them useful for more than one project.

In the example below, a simple component has been created representing the Asphalt Concrete Surface Course for one lane of pavement.



The depth of the pavement at points **ACSC\_B\_CL** and **R-ACSC\_B\_EP** has been defined as -0.1042' as shown below.



The thickness of the pavement can be changed by editing the template points, or by adding a **Label**. In the example above, the **Label** “ACSC Thickness” has been assigned for point **ACSC\_B\_CL** and for point **R-ACSC\_B\_EP**.

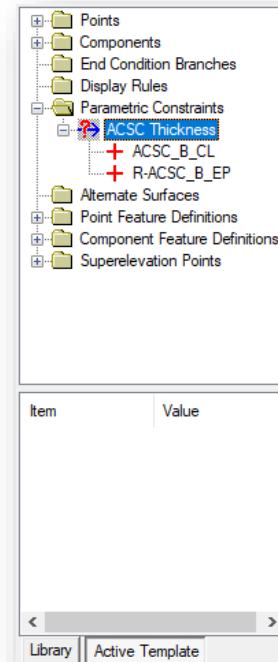
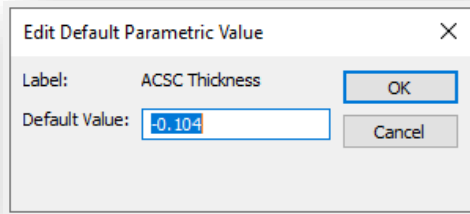
The **Label** name is user definable. The same label name is used for each point in the template that will use the value of the label. Once a label name has been assigned, it is available to be selected from the drop-down label list as other points are defined.



The value of the label can be changed in the **Create Template** dialog, or during the corridor modeling process.

To change the value in the **Create Template** dialog, select **Parametric Constraints** in the **Active Template** option. The points that have been assigned the label are listed as shown at right.

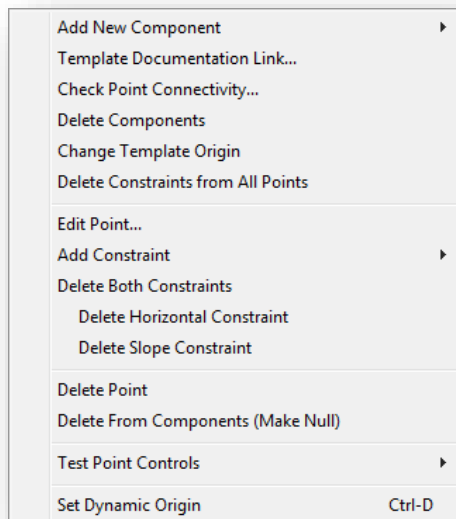
Double-click the name of the **Parametric Constraint** that you wish to edit. The *Edit Default Parametric Value* dialog show below is opened which allows you to edit the value.



The template graphics are automatically updated to reflect the new value.

## Editing Points with the Right-Click Menu

In addition to the *Point Properties* dialog, points can be edited from a pop-up menu that is accessed by right-clicking on a point in the **Active Template** window. The menu is shown below.



Some of these commands are covered later in this training guide. See the online help for documentation of commands not covered in this document.



## Adding Components to the Template

A component is a set of points that define an open or closed shape.

When defining new components using precision input, the sign of the distance is dependent on the parent-child relationship of the points. The parent is always the first point placed when creating template components. The following rules apply:

- A child placed to the right of the Parent point is at a positive distance
- A child placed to the left of the Parent point is at a negative distance
- A child placed above the Parent point is at a positive distance
- A child placed below the Parent point is at a negative distance
- The Slope is the algebraic slope

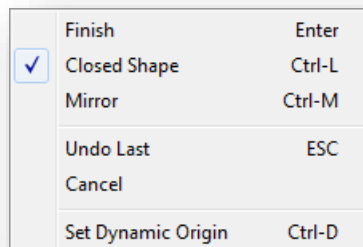
There are six types of components:

- Simple
- Constrained
- Unconstrained
- Null Point
- End Condition
- Overlay/Stripping

Each component type is described in further detail on the following pages.

As you add components to your template, the area under the active template window is used to display various parameters for the component. This is called the **Current Component** area.

Right-click in the current template window at any time during the component creation process to access the menu shown below:



Select **Finish** when you have completed creation of the component.



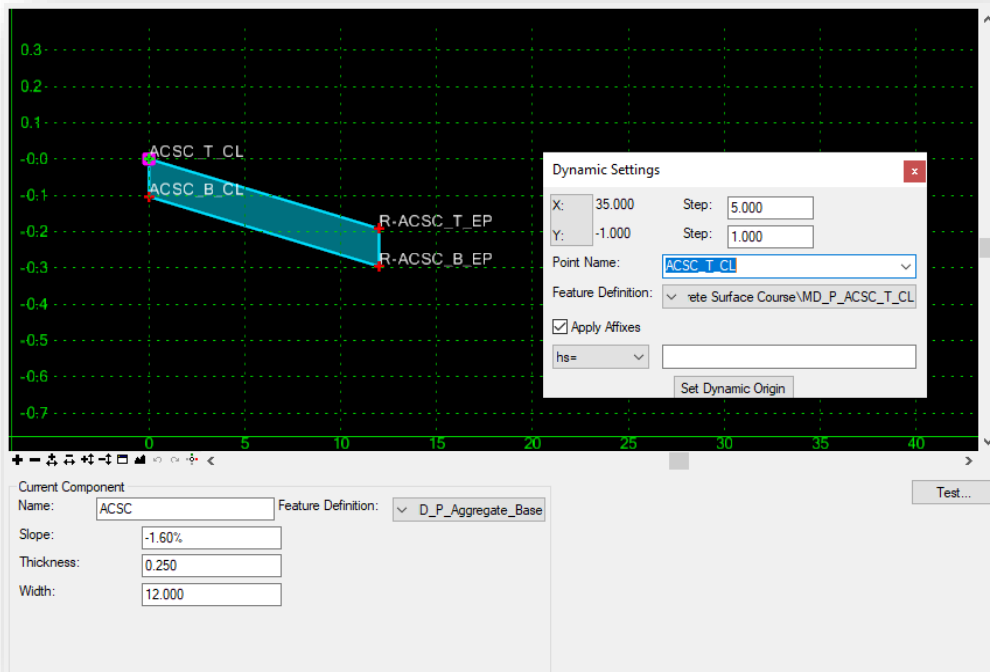
## Creating Simple Components

A simple component consists of four points defined by **Slope**, **Thickness**, and **Width**. The top points are constrained by **Horizontal** and **Slope** constraints. The lower points are constrained to the points directly above them by a **Horizontal** and **Vertical** constraint.

To create a simple component, right-click in the active template window and select **Add New Component > Simple** or select **Add > Simple** from the drop-down menu of the *Create Template* dialog to initiate the command.

Define the **Name**, **Slope**, **Thickness**, **Width**, and **Feature** of the component before placing the component in the template.

The **Feature** is used to set the symbology for the graphics drawn to represent the component in the 3D model.



The *Dynamic Settings* dialog is used to facilitate component placement. The point names are derived from the **Point Name** selected in the *Dynamic Settings* dialog. Each point is appended with a number that is incremented. The points are assigned the **Point Style** feature that is active in the *Dynamic Settings* dialog.

Point names and the feature that is assigned to each point should be reviewed and edited after the component is placed.



## Creating Constrained and Unconstrained Components

**Constrained** or **Unconstrained** components can be used to create any open or closed shapes in the template such as curb and gutter, median barriers, retaining walls, or complex pavement situations such as the shoulder break on the high side of superelevation.

### Constrained Components

To create a constrained component, right-click in the current template window and select **Add New Component > Constrained** or select **Add > Constrained** from the pull-down menu of the *Create Template* dialog to initiate the command.

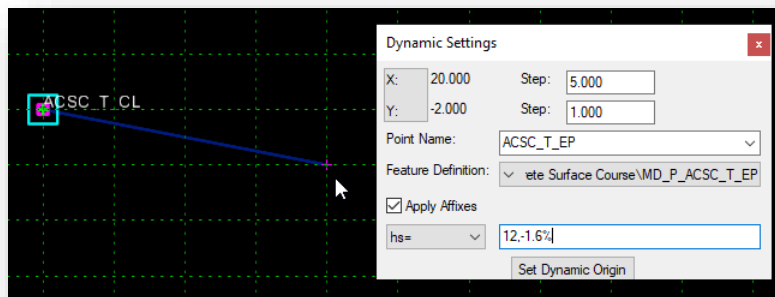
When constrained components are created, both horizontal and vertical constraints are applied to each point.

### Unconstrained Components

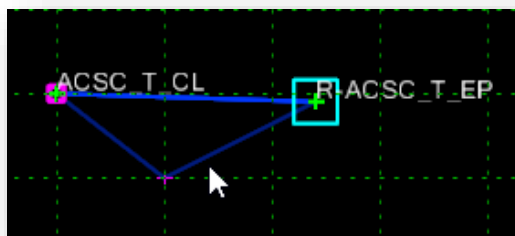
To create an unconstrained component, right-click in the current template window and select **Add New Component > Unconstrained** or select **Add > Unconstrained** from the pull-down menu of the *Create Template* dialog to initiate the command.

When unconstrained components are created, no constraints are applied. This option is useful when you wish to define constraints other than Horizontal and Vertical constraints.

The *Dynamic Settings* dialog can be for precision input when creating the component. In the example below, point **ACSC\_T\_EP** will be created at a defined horizontal distance and slope from point **ACSC\_T\_CL** using a horizontal distance of 12 and a slope of -1.6% by using the **hs=12** option with a value of **12, -1.6%**.

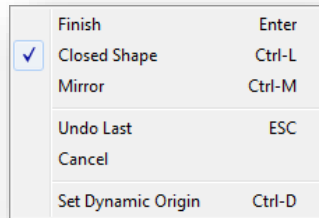


Use the **Enter** key to execute the command.





Right-click in the current template window at any time during the component creation process to access the menu shown below:



- Select **Finish** when you have completed creation of the component.
- Components can be created as open or closed shapes. To toggle on/off the option to placed closed shapes, select **Closed Shape**.
- Select **Mirror** to create two components at the same time. The second component is a mirror image of the one you are defining. The component is mirrored about a vertical line passing through the **Dynamic Origin** location.
- Select **Undo Last** to undo the last point that was added to the component.
- Select **Cancel** to terminate the component creation process.
- Select **Set Dynamic Origin** to specify the location of the dynamic origin.

## Null Points

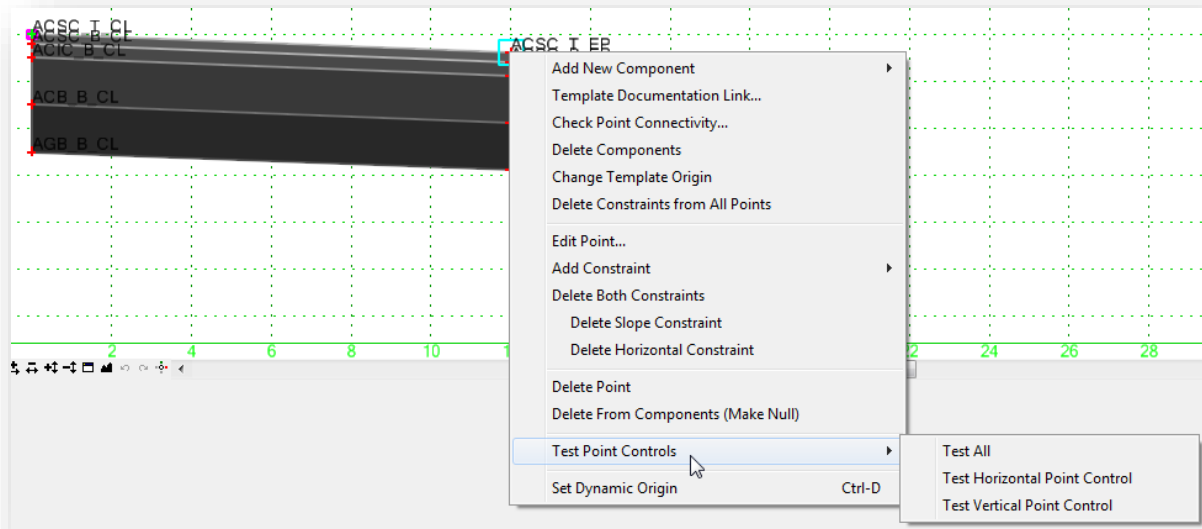
A null point is a template point that is purposely not related to any component. It's most often used as a reference for controlling other points.

To create a null point, right-click in the current template window and select **Add New Component > Null Point** or select **Add > Null Point** from the pull-down menu of the *Create Template* dialog to initiate the command.

## Testing Point Controls

The behavior of fully constrained points can be tested by right-clicking on the point and selecting **Test Point Controls** as shown below.





Select the desired option to test the behavior of a template as the point is moved either horizontally, vertically, or both at the same time.

## ODOT Standard Point and Component Names and Features

As you create templates, each point and component in the template is assigned a **Name** and a **Feature Definition**. It is important to ensure that the points and components are named and assigned features consistently in each template.

Feature definitions for template points and components are defined in **OHDOT\_Feature\_Definitions.dgnlib**, which is attached by a configuration variable.

### Point Names and Features

As the template is applied to the corridor, 3D line strings are drawn in the design file by connecting the points from template drop to template drop by their **Name**. The symbology of these line strings is controlled by the **Feature Definition** that is assigned to each point.

The ODOT workspace for OpenRoads includes several **Feature Definitions** that can be used for template points.

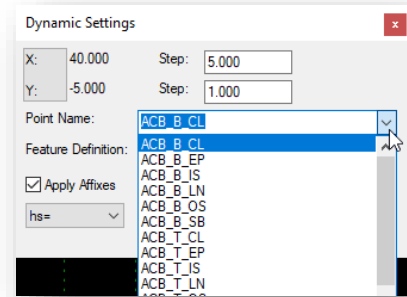
- The point feature definitions can be reviewed by browsing to the following folder in the Civil Standards tab:  
Libraries > Feature Definitions > Linear > Modeling > Template Points

**Features** can be reviewed and assigned in a variety of ways.

- We have already discussed the *Point Properties* dialog which is accessed by double-clicking the point in the **Current Template** window. The **Name** and **Feature Definition** can easily be edited using this dialog.

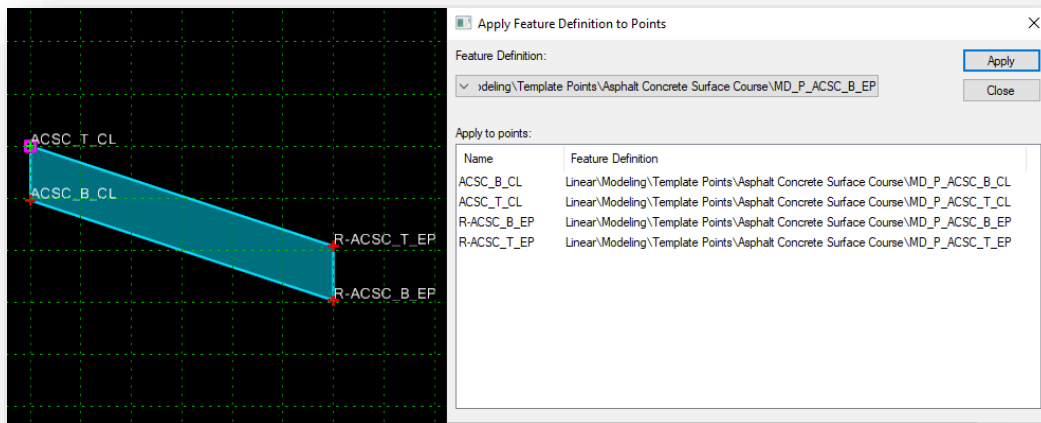


- The *Dynamic Settings* dialog can be used to define the **Point Name** that will be assigned to points as they are created. The **Point Name** field can be used to key-in a name for the point or to select a name from a predefined list as shown at right. It is recommended that you use the point names from this list when designing your templates.



When a point name is selected from the list, the **Point Style**, which is the **Feature Definition**, is automatically set to the appropriate value. The **Point Style** list contains a list of all the **Feature Definitions** available in the OHDOTCEv02 workspace.

- Features can also be assigned to one or more points by selecting **Tools > Apply Feature Definition to Points** from the *Create Template* dialog as shown below.



### Component Names and Features

**Component Names** and **Features** are defined by the user as each component is created.

- The component feature definitions can be reviewed by browsing to the following folder in the Civil Standards tab:  
Libraries > Feature Definitions > Mesh

Component Names and Features can be reviewed and edited in a variety of ways:

- Double-click on a component to open *the Component Properties* dialog.
- Select **Tools > Apply Feature Definition to Components...** to apply a selected **Feature Definition** to one or more components in the template.

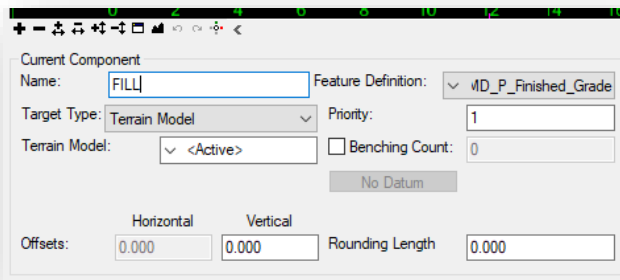


## End Conditions

An end condition is a special open-shaped component that targets a surface, a feature of a surface, an elevation, or an alignment. End Conditions are commonly used for cut and fill definitions. End Conditions are added to the end of simple, constrained, or unconstrained components.

To create an end condition, right-click in the template window and select **Add New Component > End Condition** or select **Add > End Condition** from the *Create Template* dialog's pull-down menu.

When adding an end condition to the current template, the **Current Component** portion of the *Create Template* dialog displays the following options:



The **Name** field is used to specify the user defined name for the component.

The **Feature** is used to set the feature definition for the component.

**Target Type** specifies the type of target the End Condition is seeking. Targets include:

- **Terrain Model** – Seeks an OpenRoads Terrain Model
- **Elevation** – Seeks an elevation
- **Linear Horizontal/Vertical/Both** – Seeks the intersection with a linear element selected from a list.
- **Feature Definition Horizontal/Vertical/Both** – Seeks the intersection with a feature type.

**Horizontal Offset** specifies a horizontal offset from the target. This option only applies if the target is a horizontal target or has a horizontal aspect.

**Vertical Offset** specifies a vertical offset from the target. Applies only if the target is a vertical target or has a vertical aspect. For an elevation target, specify the elevation here.

**Priority** specifies the order in which end conditions are attempted in seeking a solution. Where more than one End Condition starts at the same point, this value determines the order in which End Conditions are attempted to find a solution. Lower numbers, which have a higher priority, are attempted first.

**Benching Checkbox** indicates, when on, the set of segments of the End Condition are repeated until a solution is found.

The **Benching Count** indicates the maximum number of times that benching should occur. If this value is 0, then the repeat count is unlimited, and the benching End Condition must find its target for it to be valid. If any other number is entered, then if the benching is repeated for the specified



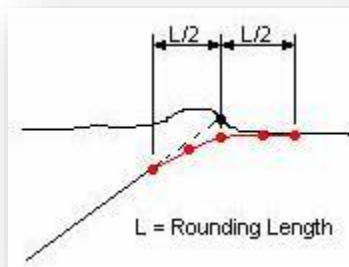
count and if the target is not reached, the benching End Condition will still be considered successful if an End Condition attached to the end of the benching End Condition is successful.

**From Datum** indicates, when on, the benching will use a datum elevation and step elevation to set the benching rather than using the End Condition segments as they are defined.

**Datum Elevation** specifies the elevation that represents the basis for the step elevation. Datum + Current bench elevation =  $n * \text{Step Elevation}$ .

**Step Elevation** specifies the increment for benching when a datum is being used. Each bench elevation is always located at the end of the first segment (or the second point) of the benching End Condition.

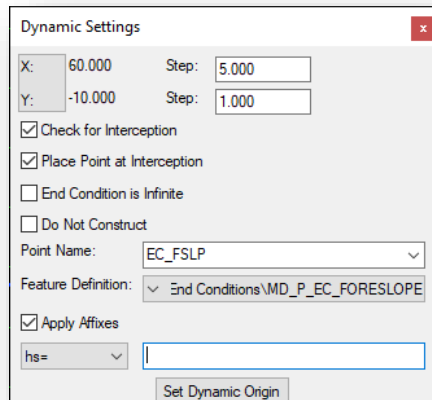
**Rounding Length** specifies the length used to round the end condition. When this value is something other than 0.0, 4 additional points are added to the end of the end condition using a parabolic formula to smooth out the transition of the intersection of the end condition with the surface. This rounding effect is only applied when the target is a terrain model or an elevation. Any other end conditions attached to the end of this one is attached to the end of the rounded section.





## End Condition Point Properties

As you are defining points for an end condition, the *Dynamic Settings* dialog includes additional options for the point definition as shown below.



**Check for Interception** specifies the preceding segment will be checked for interception of the target when the end-conditions are solved. If it is turned off, then the segment will be ignored when checking for interceptions.

**Place Point at Interception** specifies that when an interception is found along the preceding line segment, the line segment will be drawn to the interception. Otherwise, if the interception is found, the line segment will be drawn to its maximum width.

**End Condition is Infinite** specifies the last segment will be of infinite length when attempting to intercept the target. This option is applicable only to the last point on the end condition.

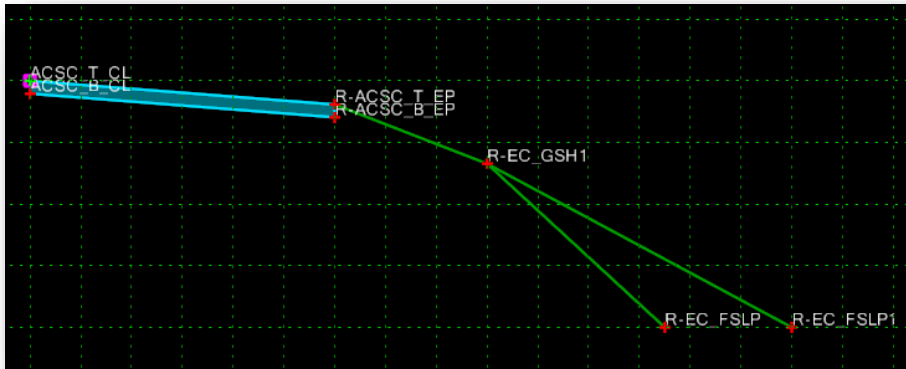
**Do Not Construct** specifies the point will be solved like all other points in the end-condition, but the point will be skipped when drawing the component segments.

These end condition point properties can be edited in the *Point Properties* dialog. Double click on an end condition point to review or edit the properties.



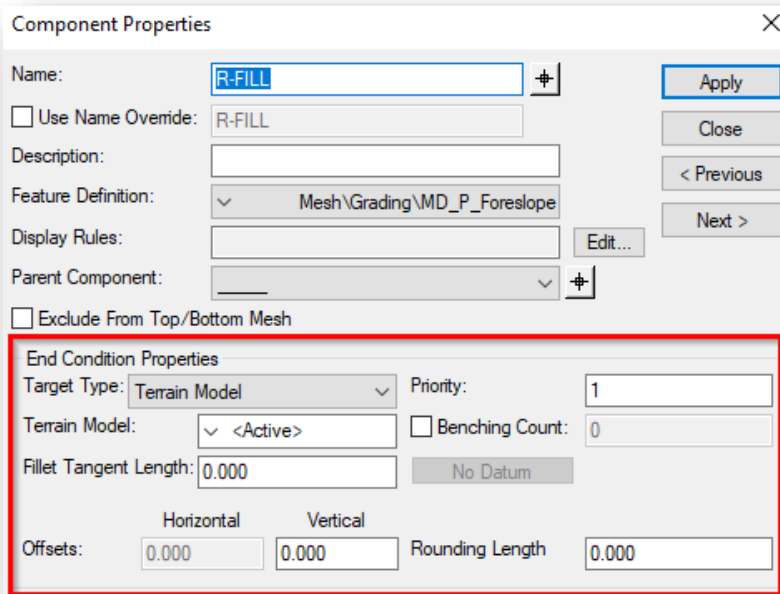
## End Condition Priority

The end condition priority value is used when more than one end condition is defined starting at the same point. A simple end condition consisting of two fill lines is shown in the example below.



When the end condition is processed in corridor modeling, each end condition will be tested in the order specified by the user. The order is determined by the **Priority** value entered by the user during the end condition creation. The most desirable solution, and the first condition to be tested, is given a priority value of 1.

Double-clicking on a component will open the *Component Properties* dialog shown below. If the component is an end condition, the **End Condition Properties** are displayed. The **Priority** is set as shown below.





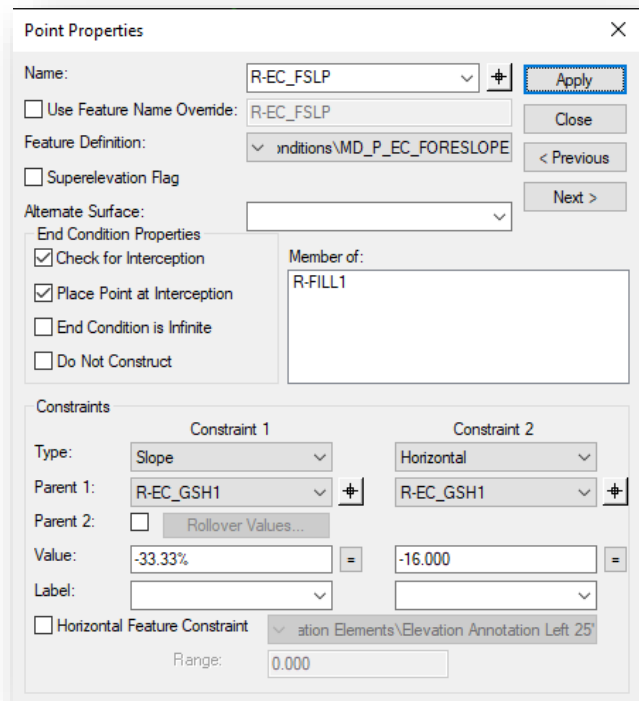
The properties for point R-EC\_FSLP were defined as shown at right. **Check for Interception** is toggled “on”, and **End Condition is Infinite** is toggled “off”. The point is defined with a value of -33.33% Slope for a vertical distance of 16.0’ from the graded shoulder point R-EC\_GSH1.

The properties for the point R-EC\_FSLP1 were defined to specify a 2:1 slope with the **End Condition is Infinite** option toggled on. The priority for the end condition component R-FILL\_2 is set to a value of 2.

With these settings, the end condition R-FILL\_3 will be tested first (Priority=1) for an interception with the surface within the vertical distance of 16.0’. If an interception is found within the vertical distance, the end condition is used for the cross section and a point will be placed at the interception.

If no interception is found within 16.0’ vertically from point R-EC\_GSH1, the condition fails and the software will test the next end condition, R-FILL\_2, which has a Priority value of 2.

Since **End Condition is Infinite** is toggled on for point R-EC\_FSLP1, the solution will solve for an interception with the surface regardless of the vertical constraint definition.

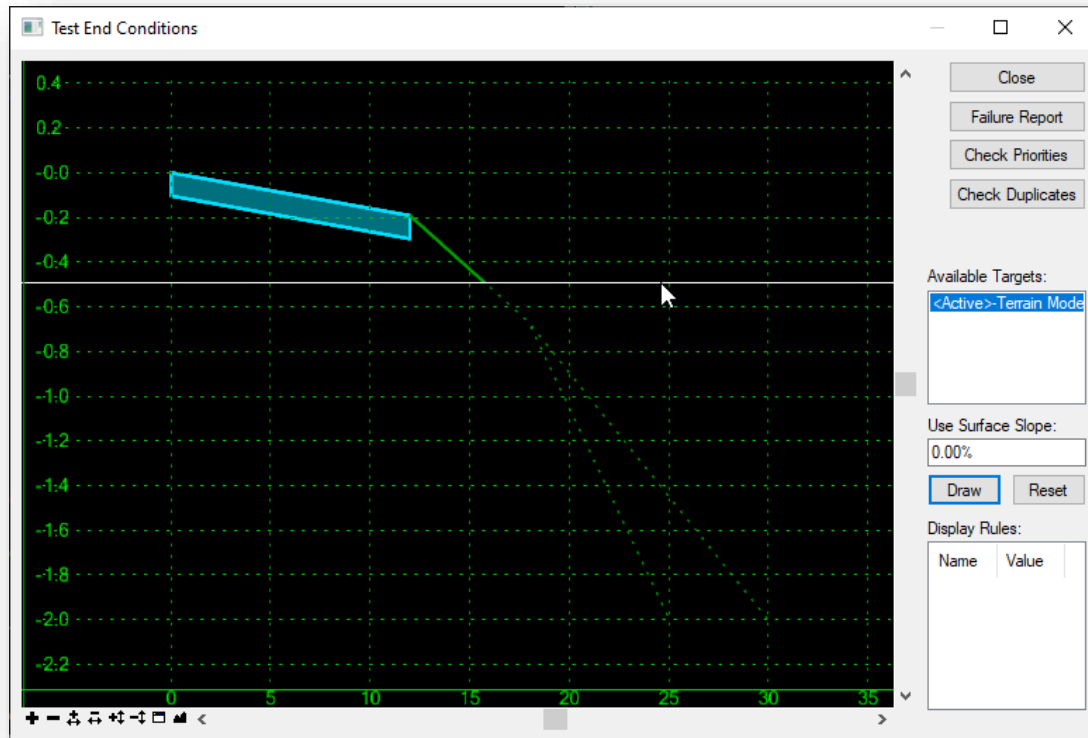




## Testing End Conditions

End conditions can be tested in the *Create Template* dialog before processing the template with corridor modeling. Testing simulates how the end condition will behave when the corridor is processed.

This is extremely useful for complex end condition definitions with multiple solutions. To test the end condition, select the **Test** command, located below the **Current Template** window. The *Test End Conditions* dialog is opened.



The software will check for conflicts in the priorities defined in the template before opening the *Test End Condition* dialog. If conflicts exist, a warning is issued.

The dialog displays the current template. Non-solved end conditions, and their child components, are displayed in dashed lines. Normal components and solved end conditions are displayed as solid lines.

To test the end condition, select one of the available targets and then select the **Draw** button. Move your cursor over to the testing window. The end conditions will change from a dotted line to a solid line indicating the solution for the target interception.

**Failure Report** activates the *Results* dialog. When you fail to get a solution on one or both sides, the results show which components failed and which end condition start points were not solved.

**Check Priorities** tests the template for priority conflicts. When conflicts exist, a message is displayed indicating there are end conditions that start from the same point and have the same priorities. The *End Condition Priority Conflicts* dialog is displayed. If no conflicts exist, a message box indicating no conflicts is displayed.

**Available Targets** lists all the targets applicable to the current template. You can select the desired target and draw it in the display.





**Use Surface Slope** specifies the slope of the surface that is being placed. This is useful in certain instances where an end condition intersects a surface more than once, such as a ditch profile.

**Draw** draws the selected target. Surfaces and elevations are drawn as horizontal lines. Features and alignments are drawn as a “+” mark. As you draw a target, the display dynamically updates to show the solution(s).

**Reset** clears all the drawn targets.

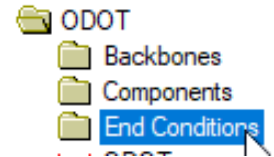
**Display Rules** list the display rules for the current template. Left click the **Value** to toggle between **True** and **False**. Press the **Reset** button to restore the original **Display Rules** values.



## OhioDOT Released End Conditions

The following section will help explain how to use and work with the Template end conditions that are released in the OHDOT standards.

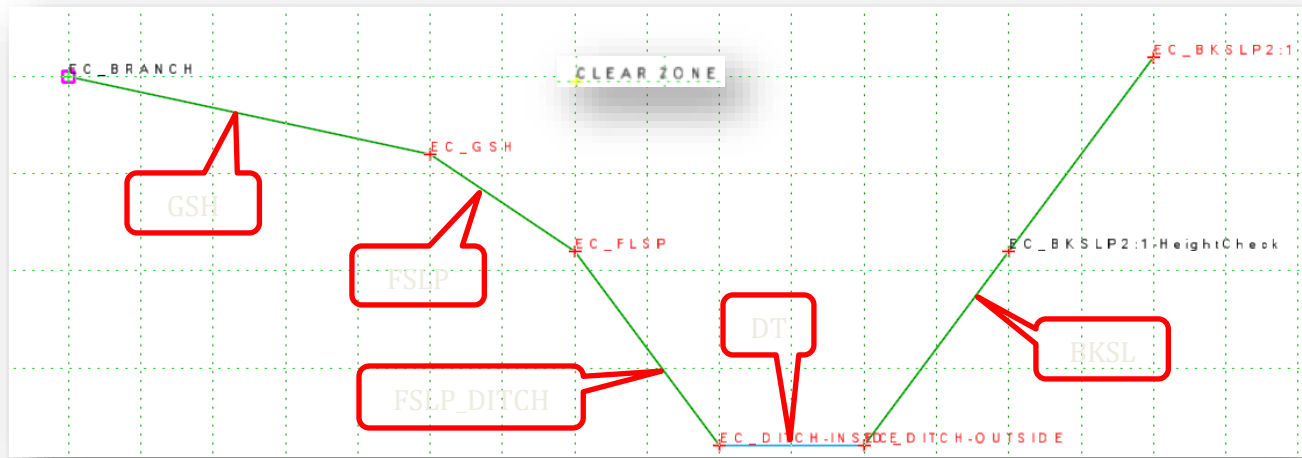
The end condition templates are based off section 300 of the Location & Design Manual – Volume 1 – Roadway Design. These templates can be found within the ODOT\End Conditions folder inside the worksets itl file.



### Standard Points and Components for End Conditions

While end conditions or grading can get very complicated in certain situations these templates were set up to cover most of the standard cases and can be used as a starting point for more complicated situations. To make sure all points and components will connect correctly from drop location to drop location the following points and components have been set up.

#### Grading with Ditches



See tables on next page for more details on the points and components that make up this template.

- Note that the foreslopes consist of three components. A Graded Shoulder, Foreslope and Ditch Foreslope. If only two different foreslopes are needed, then instead of deleting the foreslope component, simply change the constraints on the EC\_FLSP point so that the point moves to be at the same location of the EC\_GSH point. This is done so that if another template is used in the corridor that uses three foreslopes the two template drops will be able to connect all the points and components together and not leave gaps in the model.
- Note that in some cases the EC\_FLSP point will represent the clear zone location. In this situation the CLEAR ZONE point should be used to set this distance and the EC\_FLSP point should be horizontally constrained about the CLEAR ZONE POINT.
- Note the EC\_BKSLP#:1-HeightCheck point is an optional point that can be added to the BKSLP component to represent the minimal ditch depth. Set the check for interception checkbox to be



checked off to ensure the end condition fails if the minimum depth is not met, to ignore this remove the point or check on the check for intersection checkbox.

- Note when adding multiple end condition branches to a template, simply add a number to the end of point/component names and increment that for additional branches. Ensure override names remain the same.
- Note the last point in an end condition branch should always have an override name of TIE. It can be used as a basis for construction limits.
- Note backslope points can optionally contain the slope in the name.

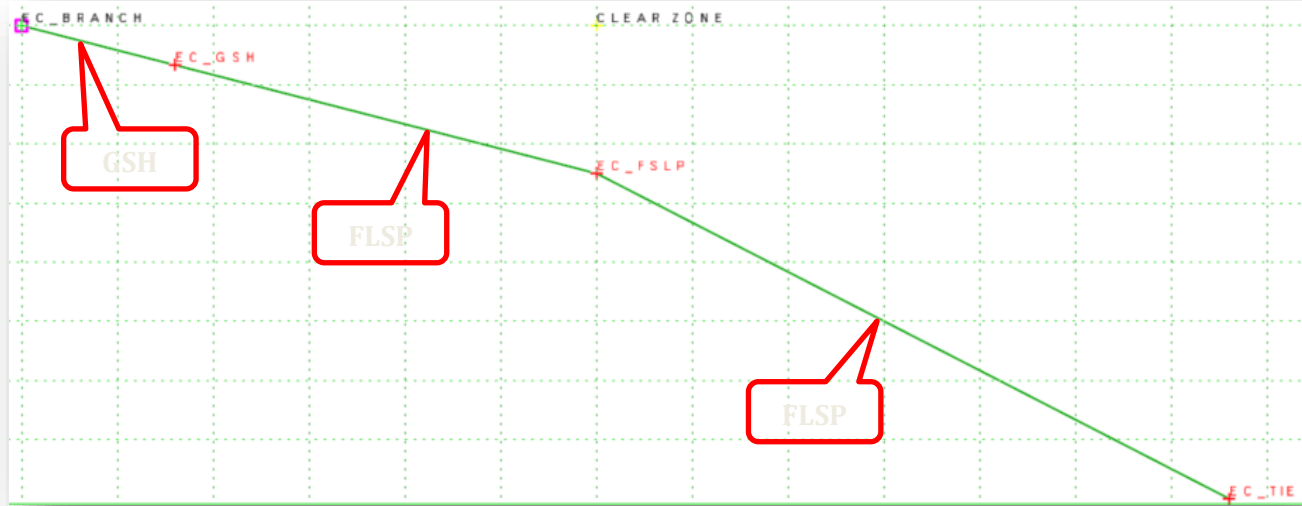


Point Name	Override Name	Desc	Constraint	Type	Parent	Label	NOTES
EC_BRANCH	--Merge with edge of shoulder point--						
EC_GSH	GSH	point for end of graded shoulder	Constraint 1	Slope	EC_BRANCH	GSH_SLOPE	Graded Shoulder slope
			Constraint 2	Horizontal	EC_BRANCH	GSH_WIDTH	width of graded shoulder
EC_FSLP	FSLP	point for end of foreslope, if not needed change constraint #2 value to 0, keep for connection to other templates	Constraint 1	Slope	EC_GSH	FSLP	Foreslope
			Constraint 2	Vertical	EC_GSH	FSLP_DEPTH	Depth of foreslope, if this point needs to match Clear zone then change to be horizontally constrained about CLEAR_ZONE Point
EC_DITCH-INSIDE	DITCH-INSIDE	point for end of ditch foreslope/inside bottom of ditch	Constraint 1	Slope	EC_FLSP	DITCH_FSLP	Foreslope of ditch
			Constraint 2	Vertical	EC_FLSP	DITCH_DEPTH_NEGATIVE	depth of ditch
EC_DITCH-OUTSIDE	DITCH-OUTSIDE	point of outside bottom of ditch	Constraint 1	Horizontal	EC_DITCH-INSIDE	DITCH_WIDTH	width of ditch
			Constraint 2	Vertical	EC_DITCH-INSIDE		leave at 0
EC_BKSLP#:1	TIE	point for end of backslope	Constraint 1	Slope	EC_DITCH-OUTSIDE	DITCH_BKSLP	backslope
			Constraint 2	Vertical	EC_DITCH-OUTSIDE		height of backslope
EC_BKSLP#:1-HeightCheck		check point to check min depth of ditch- OPTIONAL	Constraint 1	Slope	EC_DITCH-OUTSIDE	DITCH_BKSLP	backslope
			Constraint 2	Vertical	EC_DITCH-OUTSIDE	DITCH_DEPTH	min. depth of ditch
EC_CZONE	CLEAR_ZONE	Point to represent clear zone - OPTIONAL	Contraint1	Horizontal	EC_BRANCH	CZ_WIDTH	Sets width of clear zone, can be used to move EC_FLSP point

Component Name	Override Name	Description	Parent
GSH	GSH	Graded Shoulder	
FSLP	FSLP	Foreslope-keep for connection to other templates	GSH
FSLP_DITCH	FSLP_DITCH	Ditch Foreslope	FSLP
DT	DITCH	Ditch	FSLP_DITCH
BKSLP	BKSLP	Backslope	DT



## Grading without Ditches



See tables on next page for more details on the points and components that make up this template.

Unlike the ditch template which had three separate components for each part of the foreslope, there are only two components, the graded shoulder and the foreslope. Where the foreslope component consists of a middle point to allow an additional slope change. This was done to ensure components from drop location to drop location connect. If only two different foreslopes are needed, then instead of deleting the point, simply change the constraints on the EC\_FSLP point so that the point moves to be at the same location of the EC\_GSH point.

- Note when adding multiple end condition branches to a template, simply add a number to the end of point/component names and increment that for additional branches. Ensure override names remain the same.
- Note the last point in an end condition branch should always have an override name of TIE. It can be used as a basis for construction limits.



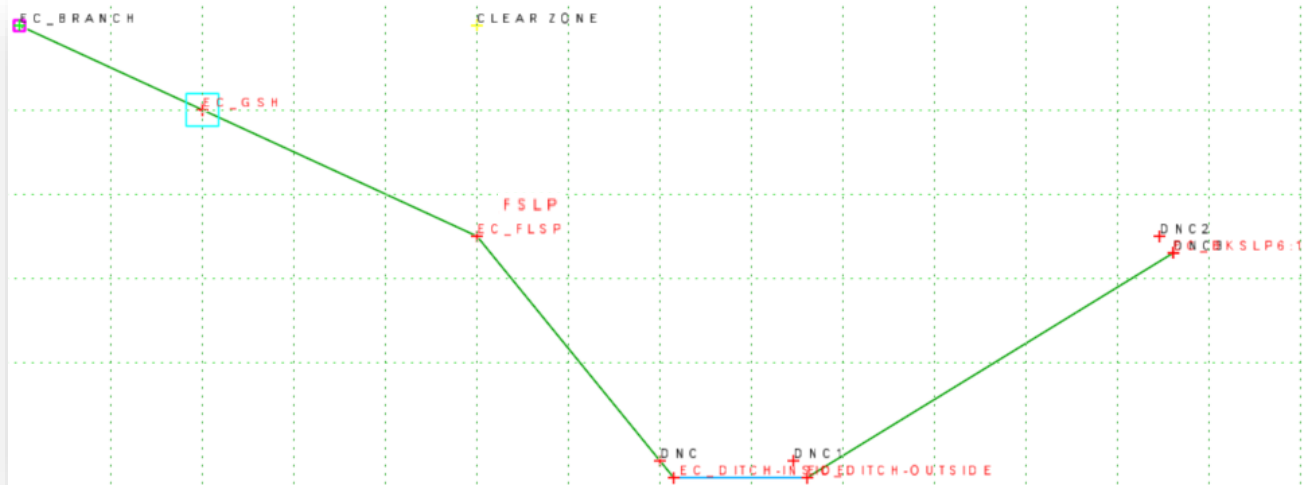
Point Name	Override Name	Desc	Constraint	Type	Parent	Label	NOTES
EC_BRANCH	--Merge with edge of shoulder point--						
EC_GSH	GSH	point for end of graded shoulder	Constraint 1	Slope	EC_BRANCH	GSH_SLOPE	Graded Shoulder slope
			Constraint 2	Horizontal	EC_BRANCH	GSH_WIDTH	width of graded shoulder
EC_FSLP	FSLP	point for slope change of foreslope, if not needed change constraint #2 value to 0, keep for connection to other templates	Constraint 1	Slope	EC_GSH	FSLP	Foreslope
			Constraint 2	Vertical	EC_GSH	FSLP_DEPTH	Depth of foreslope
EC_TIE	TIE	point for end of foreslope	Constraint 1	Slope	EC_FSLP		Foreslope
			Constraint 2	Vertical	EC_BRANCH		Overall depth
EC_CZONE	CLEAR_ZONE	Point to represent clear zone - OPTIONAL	Constraint 1	Horizontal	EC_BRANCH	CZ_WIDTH	Sets width of clear zone, can be used to move EC_FLSP point

Component Name	Override Name	Description	Parent
GSH	GSH	Graded Shoulder	
FSLP	FSLP	Foreslope	GSH



## Fixed Ditch Templates

These templates are set up to force a minimum ditch depth. Once the backslope becomes shorter than the minimum depth then the ditch will be pushed lower in order to maintain the required minimum depth of the ditch. These templates are useful when dealing with fill conditions.



These templates have additional do not construct (DNC) points. These points are needed to set up the template to push the ditch to lower elevations in order to meet minimum ditch depths.



Point Name	Override Name	Desc	Constraint	Type	Parent	Label	NOTES
EC_BRANCH		--Merge with edge of shoulder point--					
EC_GSH	GSH	point for end of graded shoulder	Constraint 1	Slope	EC_BRANCH	GSH_SLOPE	Graded Shoulder slope
			Constraint 2	Horizontal	EC_BRANCH	GSH_WIDTH	width of graded shoulder
EC_FSLP	FSLP	point for end of foreslope, if not needed change constraint #2 value to 0, keep for connection to other templates	Constraint 1	Slope	EC_GSH	FSLP	Foreslope
			Constraint 2	Vertical	EC_GSH	FSLP_DEPTH	Depth of foreslope, if this point needs to match Clear zone, then change to be horizontally constrained about CLEAR_ZONE Point
DNC		Do not construct point in line with Ditch foreslope/inside ditch	Constraint 1	Slope	EC_FSLP	DITCH_FSLP	Foreslope of ditch
			Constraint 2	Vertical	EC_FSLP	DITCH_DEPTH_NEGATIVE	Negative Depth of ditch
DNC1		Do not construct point parallel with outside ditch	Constraint 1	Vertical	DNC		
			Constraint 2	Horizontal	DNC	DITCH_WIDTH	Width of ditch
DNC2		Do not construct point parallel with backslope	Constraint 1	Slope	DNC1	DITCH_BKSLP	Backslope of ditch
			Constraint 2	Vertical	DNC1	DITCH_DEPTH	Depth of ditch
DNC3		Do not construct point lines up with end of backslope	Constraint 1	Slope	DNC2	DITCH_FSLP	Foreslope of ditch
			Constraint 2	Vertical	DNC2		
EC_BKSLP#:1	TIE	point for end of backslope	Constraint 1	Horizontal	DNC3		
			Constraint 2	Vertical	DNC3		
EC_DITCH-OUTSIDE	DITCH-OUTSIDE	point of outside bottom of ditch	Constraint 1	Slope	EC_BKSLP#:1	DITCH_BKSLP	Backslope of ditch
			Constraint 2	Vertical	EC_BKSLP#:1	DITCH_DEPTH_NEGATIVE	Negative Depth of ditch
EC_DITCH-INSIDE	DITCH-INSIDE	point for end of ditch foreslope/inside bottom of ditch	Constraint 1	Vertical	EC_DITCH-OUTSIDE		
			Constraint 2	Horizontal	EC_DITCH-OUTSIDE	DITCH_WIDTH_NEGATIVE	negative width of ditch
CLEAR_ZONE		Point to represent clear zone - OPTIONAL	Constraint1	Horizontal	EC_BRANCH	CZ_WIDTH	Sets width of clear zone, can be used to move EC_FLSP



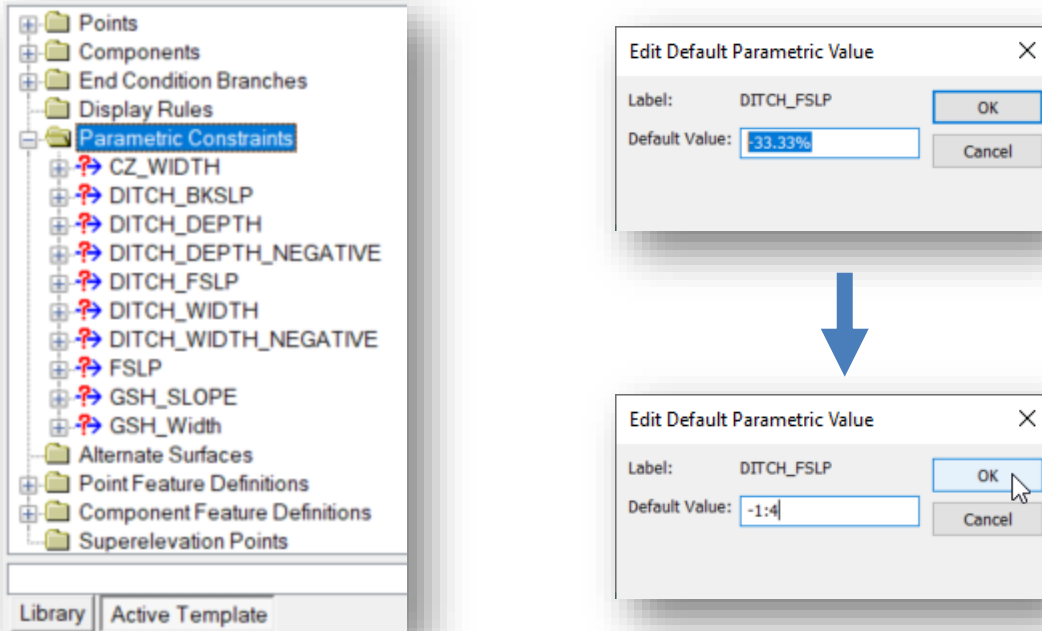


Component Name	Override Name	Description	Parent
GSH	GSH	Graded Shoulder	
FSLP	FSLP	Foreslope-keep for connection to other templates	GSH
DNC		Hidden component needed to push ditch down to maintain min. ditch depth	FSLP
BKSLP	BKSLP	Backslope	DNC
DT	DITCH	Ditch	BKSLP
FSLP_DITCH	FSLP_DITCH	Ditch Foreslope	DT

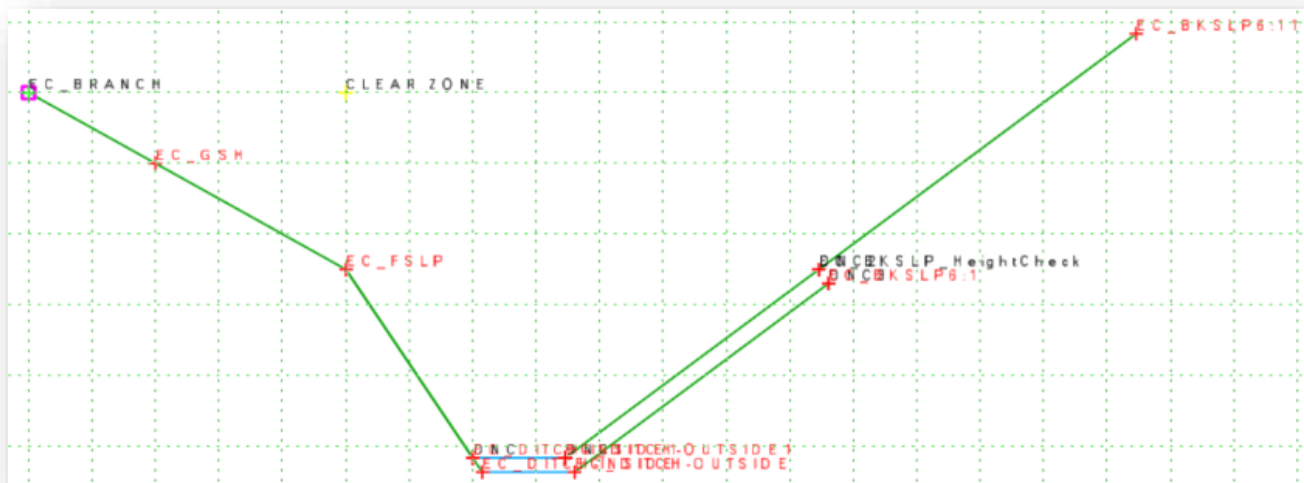


### Modifying Fixed Ditch Templates

Since these templates are more complicated to edit, it is recommended you make edits via the Parametric Constraints (Labels). Simply select the Active Template Tab and then expand the Parametric Constraints Section. Double click on a given Constraint (Label) to change its value. For example, to change the Ditch foreslope, double click on the DITCH\_FSLP constraint and a dialog will open displaying the current value. Change to the desired value and click okay.



The above screen shots of fixed ditch templates only show the fixed ditch part of the template. The actual templates delivered with the workspace take the [Grading with Ditches](#) template and combines it with the fixed ditch template. An example is shown below. Notice that some points are duplicated and end with a 1, that is okay. The Grading with Ditches part of the template has a higher priority than the fixed ditch part.





This way, the backslope will search up to solve first and once the backslope reaches the minimum ditch depth the fixed ditch solves and pushes the ditch deeper to maintain the minimum ditch depth.

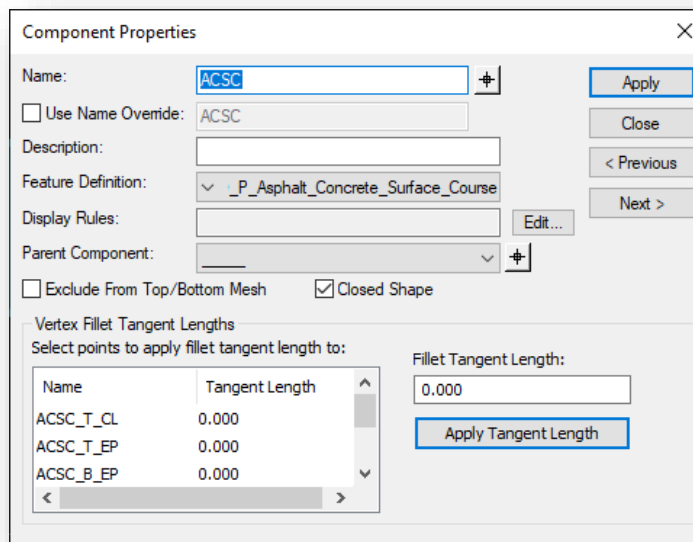
### Using Fixed Ditch Templates

Since these ditches are more complicated it is recommended you use them as a starting point to get the ditch profile. Do not try to add point controls off these templates. The simple workflow is as follows.

1. Push fixed ditch template along alignment
2. Open Profile of inside ditch line. There might be bumps in profile and it might be desired to smooth out the profile. REMEMBER when drawing in a new profile to always stay at or below the current profile. The current profile represents the highest elevation that profile can be in order to solve. Going above that profile will cause your end condition to not solve in those areas.  
Also keep in mind the lower you draw in your profile, the further the horizontal distance of the ditch will be from the alignment.  
Make sure to complex and give a feature definition to your drawn in profile.
3. Project profile to centerline alignment.
4. Remove rules from projected profile!
5. Replace fixed ditch template with equivalent non-fixed ditch template
6. Add a vertical point control to the ditch point based off the projected ditch profile.

## Component Properties

The properties of the component can be edited by double-clicking on the component. The *Component Properties* dialog is opened.



### Name

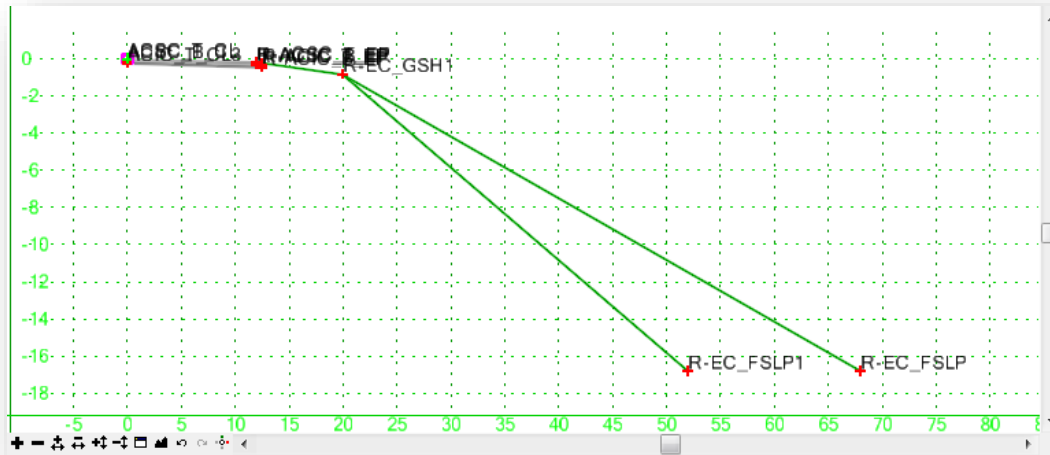
Specifies the name of the current component. Select a different component using the locate button.

### Use Name Override

The name override is used to specify the name used by the software to connect components from one template drop location to the next.



The option is intended to create connectivity from one station to the next when the template components change. For example, a template may contain more than one solution for a fill condition as shown below.



Two end conditions are defined for the fill slopes named R-FILL2 and R-FILL3. When the templates are processed, the software will evaluate each template drop location to determine whether the R-FILL2 or the R-FILL3 component is used. 3D Mesh elements are created by connecting all the components by the name of the component. All the components named R-FILL2 will be connected to create a 3D Mesh element, and all the components named R-FILL3 will be connected to create another mesh element. This leaves a gap in components at the locations where the fill slope transitions from 3:1 to 2:1 since the components named R-FILL3 will not be connected to the components named R-FILL2. The solution is to assign the same **Override Name** for each component.

### Description

Specifies an optional description of the component.

### Feature Definition

Specifies the feature style of the component. This option is used for display and to define the component's material. ODOT has defined a specific group of styles to be used for components prefixed with the word "Component". When the components are plotted in MicroStation, either as a 3D model, or as cross sections, the symbology defined by the selected Style is used. Care should be taken to select the appropriate style for each template component.

### Parent Component

Specifies the parent component. If a component has a parent component, then it is only displayed if the parent component is displayed. The parent component can be either a non-end condition component, or an end condition component. Specify the component or identify it using the locate button.



The **Active Template** listing can also be used to display and edit the parent child relationship between points and components. In the example at right, the **Components** category is open to show the parent/child relationship between the components in the tree. Drag and drop operations can be used to move one component under another.

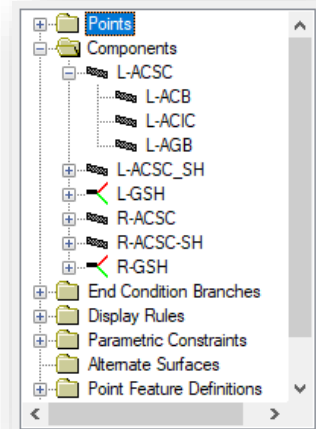
**Note:** The component names are listed alphabetically, not from top to bottom as drawn in the template.

### Exclude from Top/Bottom Mesh

Specifies that the component will be excluded from the top and bottom mesh elements that are created in the corridor modeling process.

### Close Shape

This option is only available for non-End Condition components. Only closed shapes can be used to compute component quantities.





## OpenRoads Software Version

---

This document was prepared using the following software version:

OpenRoads Designer CONNECT Edition – 2021 Release 1 Update 10 – Version 10.10.01.03

This document was updated using the following software version:

OpenRoads Designer CONNECT Edition – 2021 Release 2 Update 10 – Version 10.10.21.04

## Contacts

---

For any questions, suggestions, or problems with this document please contact the ODOT Office of CADD and Mapping Services by use of the following form on the ODOT website:

[https://odot.formstack.com/forms/cadd\\_servicerequest](https://odot.formstack.com/forms/cadd_servicerequest)

---