



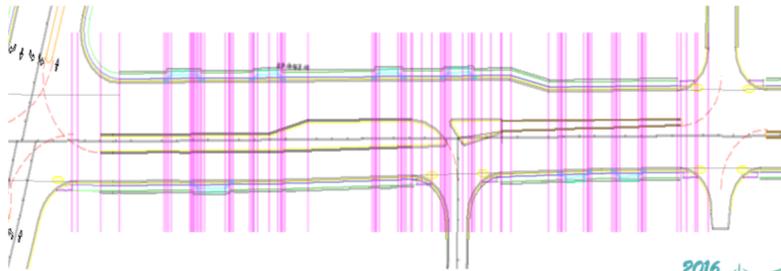
# Comprehensive Template Design

Part 2 – Variable Templates

## Creating Variable Templates

A “Typical” is not typical.

- Conditions change frequently along a corridor that would represent a change in the model.



As you become more comfortable working with templates you will find it is far more efficient to have a single template that will work in multiple situations than it is to have static templates that need to be dropped every time the condition changes. This is especially true in urban sections where the template probably changes every few feet.

There are several ways you can redefine constraints to make a template more flexible for variable conditions like this.

## Hierarchy of Constraints

A points location can be overridden in various ways to produce variable results from a single template.

- Point Constraints
- Horizontal Feature Constraints
- Parametric Constraints
- Point Controls
- Superelevation (specialized point control)



From lowest priority to highest:

**Point Constraints** – Each point can have up to two constraints that define its location in relation to other points.

**Horizontal Feature Constraint** – Each point can have a single Horizontal Feature Constraint that will allow a point to override the location defined by the Point Constraints.

**Parametric Constraints** – Can be created to override Point Constraints within a defined station range. This will override Point Constraints and Horizontal Feature Constraints.

**Point Controls** – A point control can be applied to any point on a template and forces the point to follow a particular geometric element in the file. This will override Parametric Constraints, Point Constraints and Horizontal Feature Constraints.

**Superelevation** – A specialized point control that is created and applied using the superelevation tools. This will override all other constraints and controls.

## Horizontal Feature Constraint

- Overrides the point constraint if a line with the specified Feature found within the defined horizontal range.
- Only elements added as “External References”
- **Range** specifies the maximum horizontal distance to search from the point's current location
  - Negative values will search left.
  - Positive values to the right.
  - A value of zero causes the search to find the closest instance of the Feature.

Point Properties

Name:  Apply

Use Feature Name Override:  Close

Feature Definition:  < Previous

Superelevation Flag

Alternate Surface:  Next >

Member of:

- LT\_PavOverlay\_02
- LT\_ShpOverlay\_In

Constraints

|           | Constraint 1       | Constraint 2    |
|-----------|--------------------|-----------------|
| Type:     | Horizontal         | Slope           |
| Parent 1: | LT_ML_IN_END_SEARC | LT_PVT_OLAY_OUT |
| Value:    | 0.00000            | 2.00%           |
| Label:    |                    |                 |

Horizontal Feature Constraint:

Range:



Horizontal Feature Constraints provide the ability to override a points location when an element with the assigned Feature is found within the search Range. Only elements that have been added to the corridor as “External References” are searched.

An example of how this can be applied to make an milling component follow a milling limits line is:

1. Draw the milling limit lines in the plan view using the Feature “PavementMilling(ML)”.
2. Create a template assigning the Feature “PavementMilling(ML)” as a Horizontal Feature Constraint on the outside points of the overlay component.
3. Create a corridor using template
4. Add milling limit lines in the plan view as Corridor References using the “Civil Tools > Corridor Modeling > Add Corridor References” tool from the Task Menu.

The search distances in the Range field are based on the offset defined in the template not where the parent point may be when applied to a corridor.

## Horizontal Feature Constraint

- Horizontal Feature Constraints can be used to override the point constraints so a point on the template can follow an element with the selected Feature.
  - In this example the horizontal constraint (Constraint 1) defines the width of the pavement but with the Horizontal Feature Constraint defined it will follow the EOP line as it transitions.
- The Range defines the distance from the points location it will search for a line with the Feature “PavementAsphalt(EOPA)”. If found the constraints will be overridden to that elements horizontal offset.

Point Properties

Name: RT\_PVT\_EOP\_OUT

Use Feature Name Override: RT\_PVT\_EOP\_OUT

Feature Definition: PavementAsphalt\_pm

Superelevation Flag

Alternate Surface:

Member of:  
RT\_Pavement3:Outside  
RT\_ShoulderPavedOutside

Constraints

| Constraint 1   | Constraint 2    |
|--|-----------------|
| Type: Horizontal   | Slope           |
| Parent 1: RT_PVT_ADJ1_OUT  | RT_PVT_ADJ1_OUT |
| Value: 12.0000   | -3.0000%        |
| Label: EOP_WidthOutRt  | EOP_SlopeOutRt  |
| <input checked="" type="checkbox"/> Horizontal Feature Constraint: PavementAsphalt(EOPA) |                 |
| Range: 20.0000   |                 |



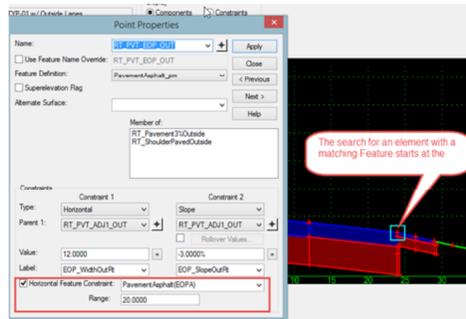
Horizontal Feature Constraints are the most common way to make a point follow a plan element like an edge of pavement as its horizontal location from the centerline varies. This can be done by:

1. Apply “Feature” to the plan element defining its symbology.
2. Edit the template point you want to follow this plan element and select the Horizontal Feature Constraint to match the Feature applied to the plan element.
3. In the point properties define the Range or distance you expect the template to encounter the plan element.
4. When the corridor is created the plan element will need to be added as a Corridor Reference in order for the template to see it.

## Horizontal Feature Constraints

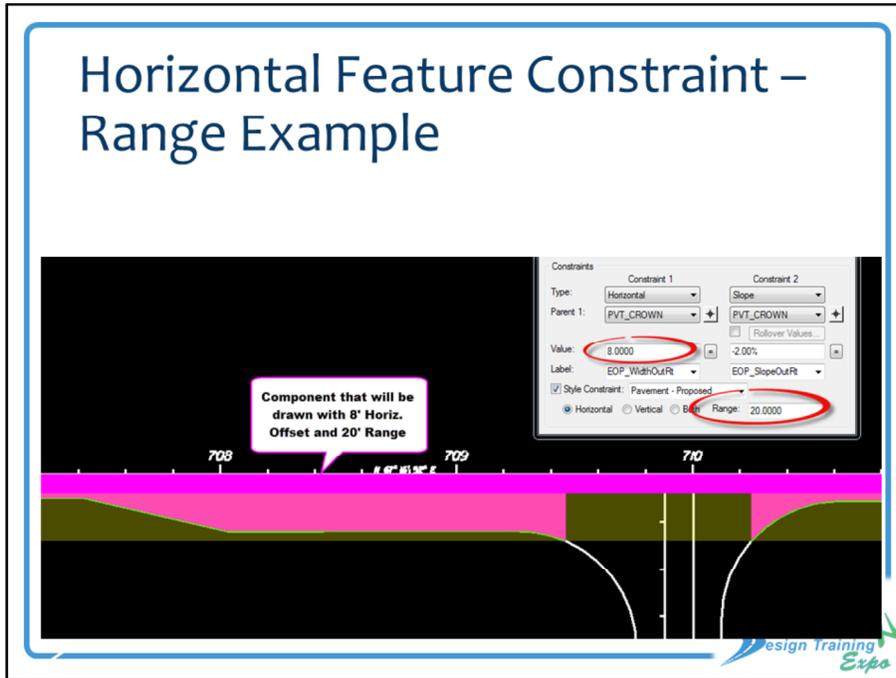
Horizontal Feature Constraints are a common way to make a point follow a plan element like an edge of pavement as its horizontal location from the centerline varies. This can be done by:

1. Apply a “Feature” to the plan element .
2. Edit the template point you want to follow the plan element and select the Horizontal Feature Constraint to match the Feature applied to the plan element.
3. In the point properties define the Range or distance you expect the template to encounter the plan element.
4. When the corridor is created the plan element will need to be added as a Corridor Reference in order for the template to see it.



The Range or search distance begins from the points horizontal location in the template. The template will search from that location the distance defined in the Range property. Positive distances to the right and negative distances are to the left. This creates an envelope in which an element with the defined feature can be found. This keeps the system from finding like elements on the other side of the road.

## Horizontal Feature Constraint – Range Example

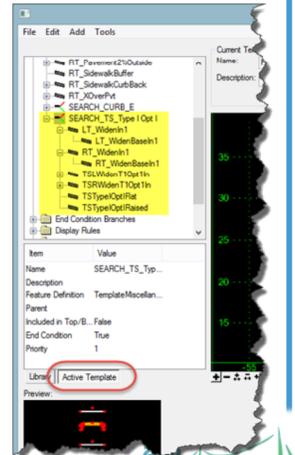


The animation in this slide shows what happens to the edge of pavement when it is located within the search Range and how it defaults back to the defined point constraints when the edge of pavement falls outside the search Range.

## Parent \ Child Relationships

Defining a Parent\Child relationship between related components makes it much easier to manage their display in variable conditions.

- End Conditions, Switches, and Display Rules effecting the parent effect children too.
- Deleting a parent component deletes all children.
- Component Relationships can be viewed as a hierarchical tree using the Active Template tab on the Create Template dialog.
  - Relationships can be created by dragging and dropping on the Active Template tab.

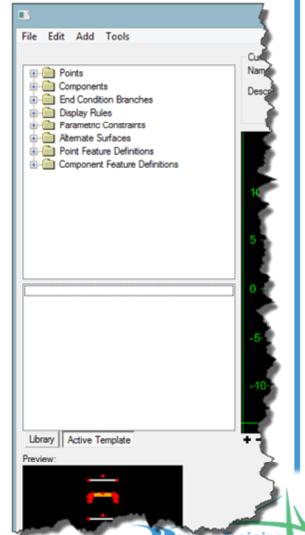


A well structured template will have parent\child relationships defined between components that go together. For example the base component is a child of the pavement component. This becomes more important as variable conditions and display rules are added to the template. When properly structured you can apply the display rule to the parent component only instead of modifying each subcomponent individually.

## Active Template View

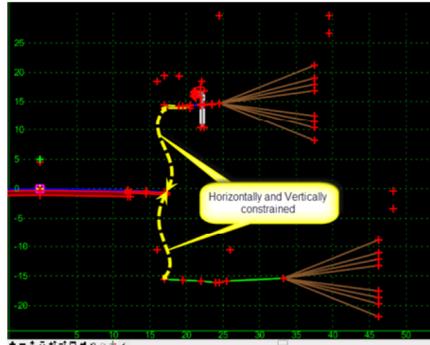
The Active Template view allows you to see and modify properties for all parts of the active template.

- Points - List of all points. Highlights the selected point in the template.
- Components – List of Components in hierarchical tree.
- End Condition Branches – List of end conditions in hierarchical tree.
- Display Rules – List of all Display Rules . Highlights the selected Display Rules in the template.
- Parametric Constraints – List of all Labels defined in the active template.
- Alternate Surfaces – List of all Alternate Surfaces and their components.
- Point Feature Definitions – List of Points categorized by Feature.
- Component Feature Definitions - List of Components categorized by Feature.



## Organizing \ Stacking Conditions

- Instead of stacking all of the end conditions on top of each other they can be offset.
- All end conditions can be attached using Horizontal and Vertical constraints
- Parametric Constraints can be used to override the Horizontal and Vertical constraints to zero values when applied to the corridor.



2016  
Design Training  
Expo

As more conditional components are added to make the template more dynamic, it get very busy and hard to manage if all the components are on top of each other. We recommend placing the conditional trigger set above or below the connection point and then use parametric constraints to move the trigger sets into place when applied to a corridor.



# Switches

“Switch” is a term for using the horizontal distance between two points to switch from one condition to another.

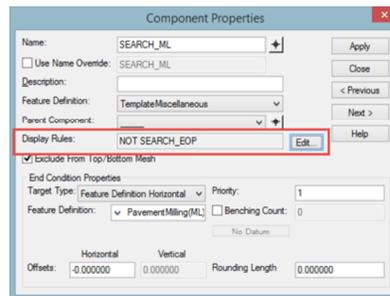
- If the distance between LT\_Switch and LT\_Switch1 is 1' display a Type F Curb and Gutter
- If the distance between LT\_Switch and LT\_Switch1 is 2' display a Type E Curb and Gutter
- If the distance between LT\_Switch and LT\_Switch1 is 3' display a Drop Curb
- Display Rules are applied to each of the possible conditions so as the distance changes one condition is turned off and another is turned on.

| Name       | Type       | Expression             | Test | Value    | Result |
|------------|------------|------------------------|------|----------|--------|
| LT_SWITCH1 | Horizontal | LT_SWITCH1 - LT_SWITCH | =    | 1.000000 | False  |
| LT_SWITCH2 | Horizontal | LT_SWITCH1 - LT_SWITCH | =    | 2.000000 | True   |
| LT_SWITCH3 | Horizontal | LT_SWITCH1 - LT_SWITCH | =    | 3.000000 | False  |
| LT_SWITCH4 | Horizontal | LT_SWITCH1 - LT_SWITCH | =    | 4.000000 | False  |
| LT_SWITCH5 | Horizontal | LT_SWITCH1 - LT_SWITCH | =    | 5.000000 | False  |
| RT_SWITCH1 | Horizontal | RT_SWITCH1 - RT_SWITCH | =    | 1.000000 | False  |
| RT_SWITCH2 | Horizontal | RT_SWITCH1 - RT_SWITCH | =    | 2.000000 | False  |
| RT_SWITCH3 | Horizontal | RT_SWITCH1 - RT_SWITCH | =    | 3.000000 | False  |
| RT_SWITCH4 | Horizontal | RT_SWITCH1 - RT_SWITCH | =    | 4.000000 | False  |
| RT_SWITCH5 | Horizontal | RT_SWITCH1 - RT_SWITCH | =    | 5.000000 | False  |

Switches are basically a series of if statements with a display rule added to each. Typically the if statement measures the distance between two points and assigns a display rule to activate when the distance equals a specific number. This way each possibility can only occur in one condition.

## Display Rules

- A Display Rule is a condition statement that can be assigned to a component. One assigned the component can only be displayed when the conditional statement is true.
  - Evaluated at each interval
  - Affects the component and all children



Use this dialog to add or edit a template display rule.

**Note:** Display rules are not applied in transitions. Transitions see only what is displayed at the beginning and end of the transition at the time the transition is created, and use that information to create the transition. Only components that are displayed are included in the transition.

### Name

displays the rule name. In edit mode, this field is read only. However, you can rename the rule from the Create Template > Active Template tab.

Note: Display rules can not contain a left or right parenthesis, "(" or ")", or any of the following phrases in the rule name: AND, OR, NOT.

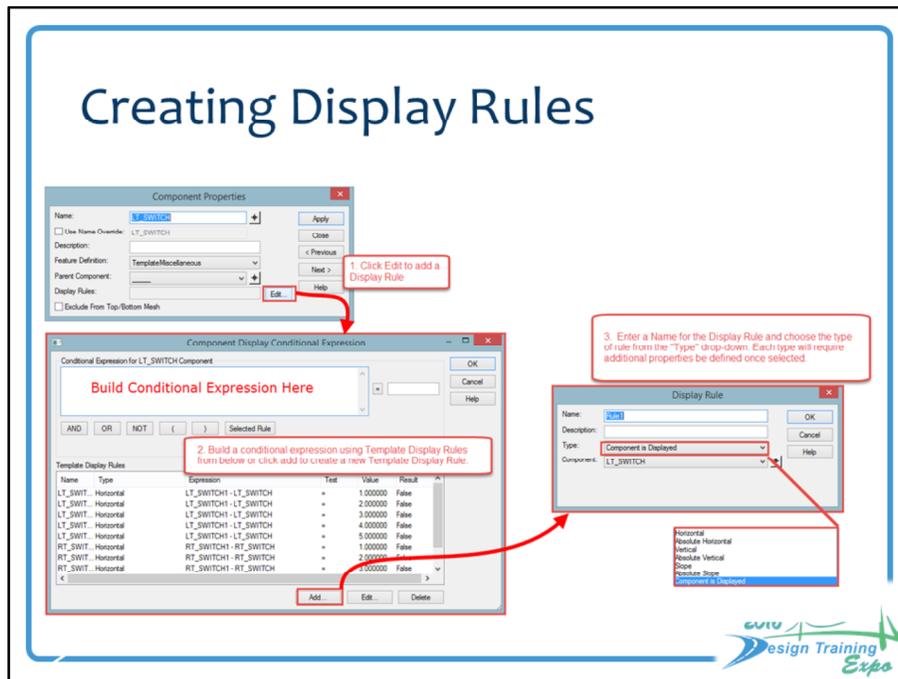
### Description

specifies a description for the display rule.

### Type

displays the rule type.

# Creating Display Rules



To create a Display Rule click the Edit button on the parent component's properties dialog.

This will open the Component Display Conditional Expressions dialog. On this dialog you can build a Conditional Expression from a list of previously created Display Rules that can be evaluated then click OK. If there are no Display Rules listed yet, or if the Display Rule you need does not appear in the list, click the Add button on the bottom of the dialog.

This opens the Display Rule dialog. On this dialog you can enter a name and a description for the new Display Rule you are creating. Then you can choose from a list of display rule types and define the properties associated with the type.

Types:

## Horizontal

specifies the difference between the x values of the point (pt1.x - pt2.x).

## Absolute Horizontal

specifies ABS ( pt1.x - pt2.x ).

## Vertical

specifies pt1.y - pt2.y.

**Absolute Vertical**

specifies  $ABS ( pt1.y - pt2.y )$ .

**Slope**

specifies the slope from pt1 to pt2.

**Absolute Slope**

specifies the absolute value of the slope from pt1 to pt2.

**Component is Displayed**

list existing components. The rule is evaluated to True if the indicated component is displayed.

**Between (1st point)**

specifies the first point for the conditional test. Does not apply for type Component is Displayed type.

**And (2nd point)**

specifies the second point for the conditional test. Does not apply for type Component is Displayed.

**(Expression)**

specifies the conditional expression (<, <=, =, >=, >).

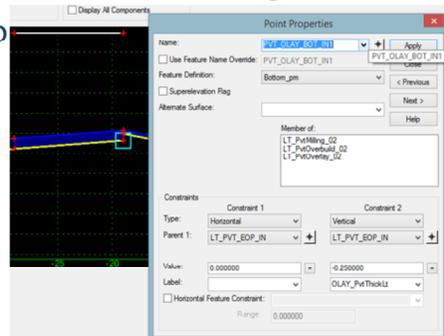
**(Value field)**

specifies the value that the expression is evaluated against.

## Defining Parametric Constraints (Labels)

Parametric Constraints use the Label property defined on a points constraints to override the defined value over a specified station range.

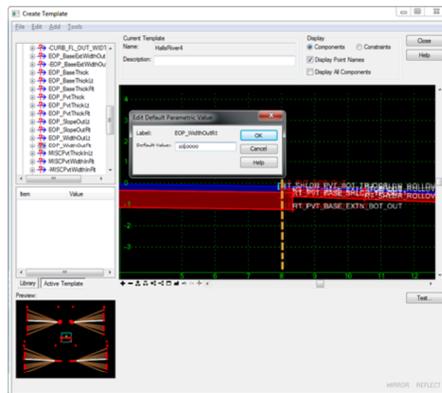
- Labels for common constraints are the same.
  - Pavement Depth
  - Lane Width
  - Switches



# Parametric Constraints

To edit Parametric Constraints on the Create Template dialog:

1. Select the “Active Template” tab.
2. Expand the Parametric Constraints folder
3. Double-click the Label name
4. Enter a value in the “Edit Default Parametric Value” dialog
5. Click OK.



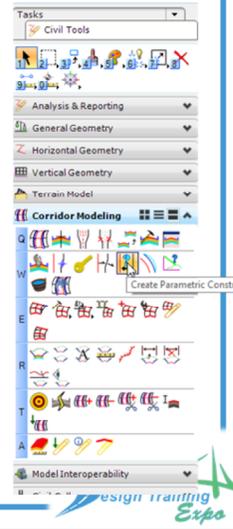
This allows you to reset the values for all points using the selected label at once.

For example, Pavement Thickness could be defined on the inside and outside of every pavement component in the template. For a six lane section with cross overs Pavement Thickness could be defined 14 time or more in a single template. If the pavement design changes you can quickly reset all of those values at once without having to edit each point’s properties to redefine the vertical constraint.

## Applying Parametric Constraints

- To apply a parametric Constraint to a corridor select “Civil Tools > Corridor Modeling > Create Parametric Constraint from the Task Menu.
- After selecting the corridor you will be prompted for :
  - Start station
  - Stop station
  - Constraint Label
  - Start Value
  - End Value

|                  |                          |
|------------------|--------------------------|
| Lock To Start    | <input type="checkbox"/> |
| Start            | 546+34.39 R1             |
| Lock To End      | <input type="checkbox"/> |
| Stop             | <input type="checkbox"/> |
| Stop             | 495+00.53 R1             |
| Constraint Label | OLAY_PvtThickLi          |
| Start Value      | -0.250000                |
| Stop Value       | -0.250000                |



Parametric constraints can also be used to change one or more labeled constraint values of a template while the template is being processed through the corridor.. Click Tools > Parametric Constraints. This allows you to use only one template to handle a number of different conditions. The process involves two steps.

1. Add labels to the constraints that are to be controlled. This is done in the Create Template command.
2. Specify the new constraint value(s) and the station range over which they will be applied on the corridor. This is most commonly used to change pavement thickness over a specified range.

All constraints can contain parametric labels.

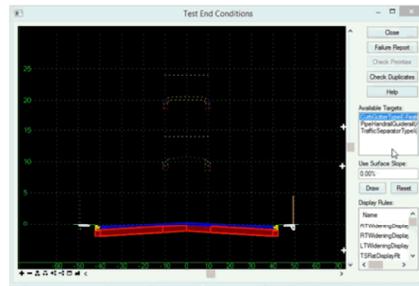
For example, if the milling depth changes in the middle of the project you don't need to do multiple template drops. You can do a single template drop and add a Parametric Constraint each time the milling thickness changes.

## Trigger Lines

“Trigger Line” is a term for the use of End Conditions as display controls.

Trigger Lines are used to make a template dynamic by including multiple possibilities in a single template.

- Reduces the number of template drops required along a corridor.
- Especially useful in urban areas where conditions change frequently.



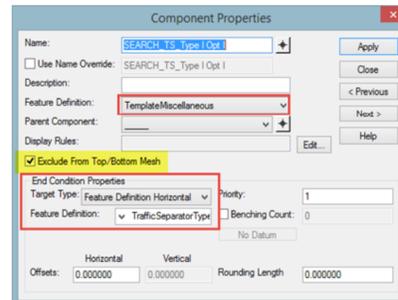
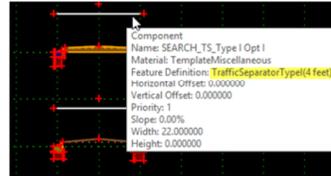
Trigger Lines are End Condition components used to search for “Corridor References” with a specific Feature.

End Conditions are only displayed when the defined condition is true.

All child components of an End Condition are displayed when the End Condition is True

## Creating Trigger Lines

- An End Condition (Trigger Line) is configured to search for a specific target.
- Make Trigger Line the Parent of related components.
- All related components of that Trigger Line will be displayed when the Target is located and they will not be displayed when the when the Target is not found.
- Exclude Trigger Lines from triangulation.



To create a Trigger Line you can right-click in the editor and select “Add New Component > End Condition” and draw a line horizontally from where you want to begin the search to a point in the direction of the search. You could then set the Target Type to “Feature Definition Horizontal” and the choose the desired target Feature Definition.

A default Trigger Line is provided in the FDOTSS4.ITL file. It is named “Standard Search Line (Trigger)” and can be found in the End Conditions folder. To use this predefined trigger:

1. Drag this component into your template
2. Set the target feature in the trigger line properties.
3. Edit the component you want to be controlled by the trigger line and make the trigger line the parent component.

## Contact Info



Jimmie Prow  
CADD Support Coordinator  
Phone no: (850) 414-4863  
Toll Free no: (866) 374-3368 extension 4863

email        [ecso.support@dot.state.fl.us](mailto:ecso.support@dot.state.fl.us)  
                 [jimmie.prow@dot.state.fl.us](mailto:jimmie.prow@dot.state.fl.us)  
web:         <http://www.dot.state.fl.us/ecso>

