



Advanced Grading Solutions Using AutoCAD Civil 3D Corridors

David Zavislan, P.E. – Wood Rodgers, Inc.

CV5644 This class takes a focused look at several workflows in Autodesk AutoCAD® Civil 3D® software that use corridors to model surfaces with complex slope geometries. Several examples from actual grading projects will be presented. One example demonstrates the creation of a slope that uses target alignments and profiles to control the placement and grade of a mid-slope bench and drainage channel. Other examples show how to use default and custom subassemblies to model variable height retaining walls.

Learning Objectives

At the end of this class, you will be able to:

- Explore subassemblies and their behavior with a grading perspective
- Explain how to develop assemblies to attain specific design objectives
- Build smarter corridor models by using conditional subassemblies
- Benefit from creating custom subassemblies to solve grading problems

About the Speaker

David is an Integrated Technology Engineer for Wood Rodgers in Sacramento California. Wood Rodgers is a full-service civil engineering firm with offices in Northern California and Nevada. David supports the firm's engineering and document delivery processes - developing CAD tools, training, and standards. David is a licensed civil engineer in California and Colorado. He has spent many years working as a CAD consultant, specializing in civil engineering, surveying, and mapping. Over the span of his career he has held many positions in the engineering industry: CAD manager, structural engineer, staff engineer for a municipal public works department, project engineer and project manager. He also worked several years as a software trainer and consultant for a software reseller.

dzavislan@woodrodgers.com

Overview

The intent of this class is to broaden the scope of corridor modeling functionality beyond the realm of road design. By examining some of the various subassemblies available in the AutoCAD® Civil 3D® program, and the corridor modeling process, this class will expose you to the possible grading solutions available with these tools.

The examples presented in this class are all based on actual grading projects encountered in my work at several engineering firms. Each example is intended to highlight a specific capability of corridors and their application to grading design problems.

Since the duration of this presentation is limited to 90 minutes, not all material can be covered in the detail necessary for you to repeat these lessons on your own. This handout was developed to be supplemental to the presentation. It contains detailed, step-by-step workflows for some of the more complex processes that will be needed to successfully apply these lessons to your own work.

Which Grading Tools Do I Use?

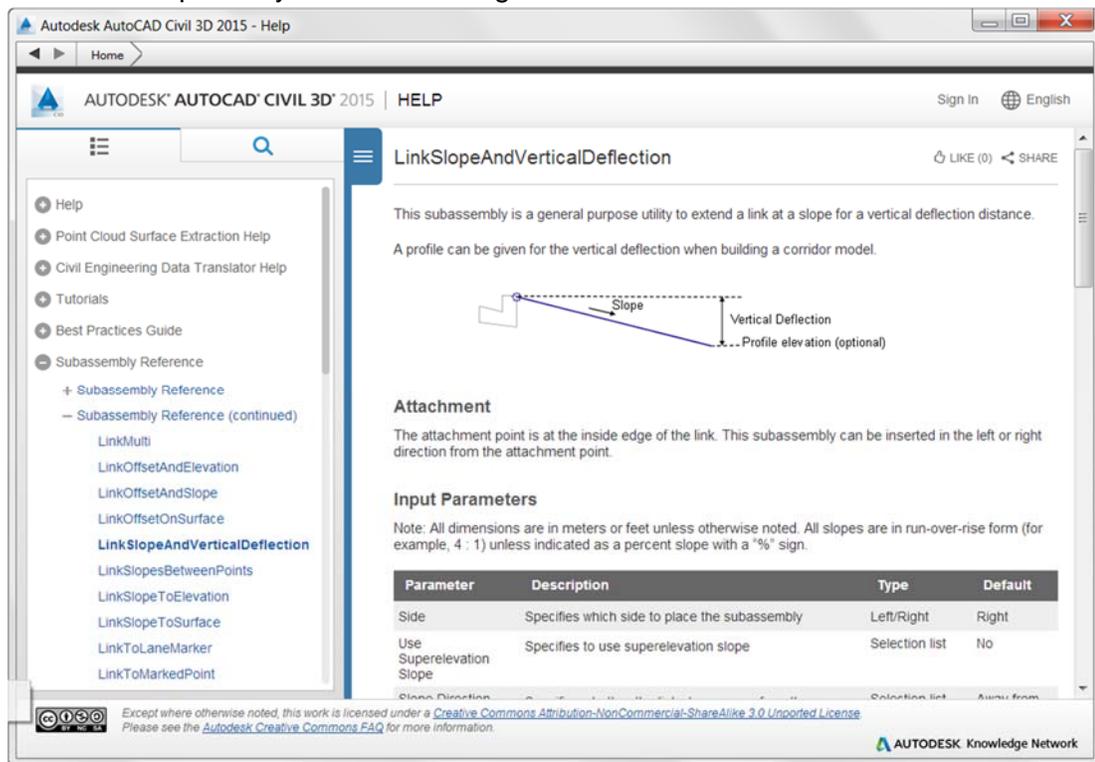
The following table presents a listing of grading tools available in the Civil 3D program. They are organized by complexity, to provide a better idea of where corridor modeling can fit into the grading design process.

Desired Model	Civil 3D Features to Use
Single spot elevation	COGO Point
Single linear element	Feature Line
Single element with another relative line or daylight	Feature line with stepped offset, Or feature line with grading object
Complex vertical geometry like parking lots and site perimeters	Feature line with elevations from surface, Or feature line from alignment and profile.
Multiple linear grade breaks, such as parcels, walls, curbs, ditches, and benches	Alignment, profile and corridor.
Complex models where the offset parameters vary along its length	Corridors with conditional subassemblies or custom subassemblies.

Tips and Tricks

This section provides an overview of key elements to keep in mind to help you successfully use corridors in grading models.

- Probably the most important suggestion is to become familiar with the subassembly Help document. Use it to learn about all the tools that you have at your disposal. The help document for each subassembly explains the available input and target parameters, and how it will behave in the corridor model. At the bottom of each document you will find charts and diagrams that show the code names and which points, links and shapes they have been assigned.

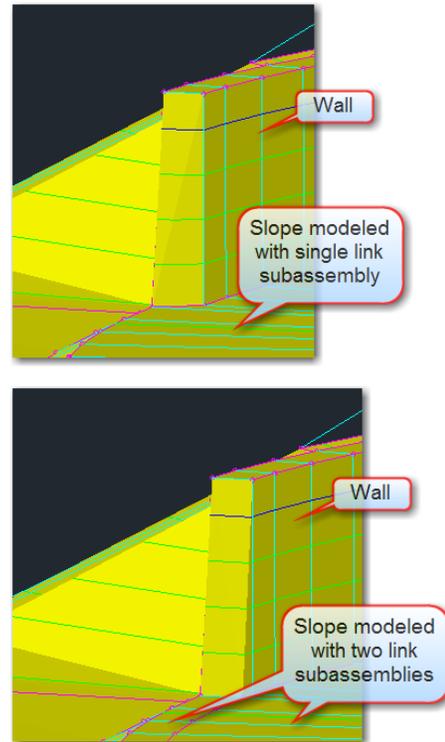


The help files can be accessed by right-clicking on any subassembly in the Tool Palette, by clicking the Subassembly help button on the Construction tab of the Assembly Properties window, or by clicking the Subassembly help button on the Parameters tab of the Subassembly Properties window.

- Use stock subassemblies whenever possible. They will work on any workstation without needing to import custom subassemblies. More advanced behavior can be achieved by including conditional subassemblies.

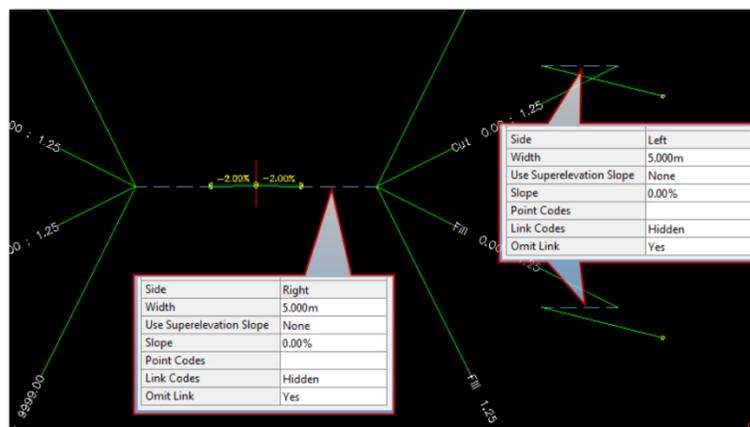
- Match points on assemblies at transitions or regions. This will give you more control over how the corridor surface gets triangulated. This is necessary sometimes, because feature line branching does not always work as expected.

The two images on the right show how a surface model transitions from a flat slope to a retaining wall. In the first image, the slope is modeled with a single link subassembly. This causes an irregular looking transition in the surface model. In the second image, a second link subassembly was inserted into the flat slope region. It had a width that matched the width of the wall. This configuration created a much nicer looking transition from the slope area to the wall.

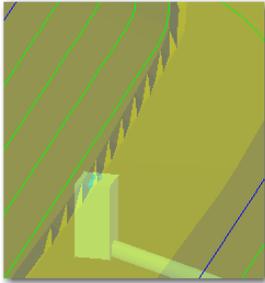


- Use generic links, with the link omitted, to offset the test location of conditional Cut/Fill subassemblies.

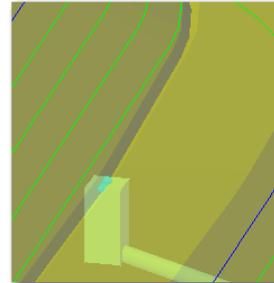
The following image shows this applied to a simple roadway assembly. A generic LinkWidthAndSlope subassembly, with the Omit Link property set to “Yes”, was used to locate the test condition 5 meters from the edge of road. Then, on the end of each condition, another generic link was used to return 5 meters back to the edge of road so that the following subassemblies would continue from there. Changes to the test location are easily made by editing the width property of the LinkWidthAndSlope subassemblies.



- To create more accurate corridor surface models along walls, curbs and ditches, supplement the link data by adding figures as breaklines data.

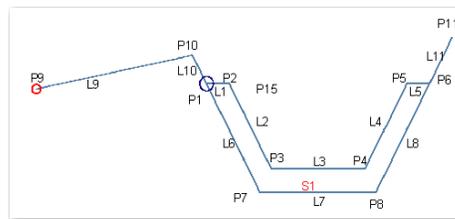


Irregular triangulation in surface modeled with top links only.



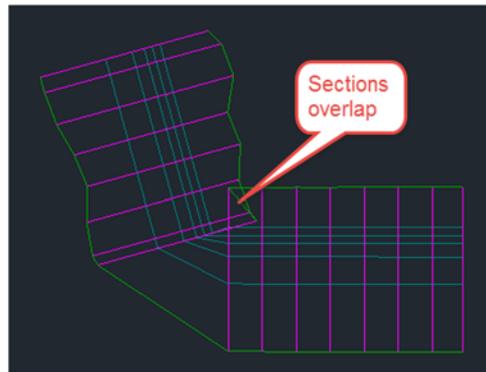
Same model with ditch flowline feature lines added to the surface.

If a subassembly does not have point codes at these locations, the program will not be able to generate figures. This can be corrected by adding marked point subassemblies with your own codes. A good example of where this would need to be done is the flowline location of the SideDitch subassembly. It does not have any point codes at the channel flowline locations (P3 and P4).

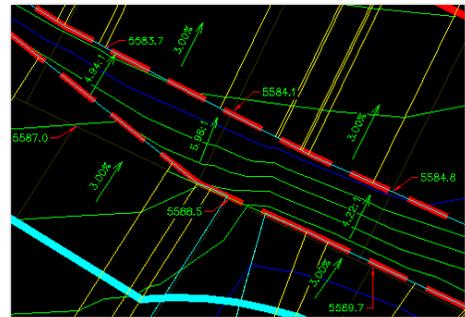


Coding diagram for the SideDitch subassembly.

- Concave arcs and sharp angles can be problems for corridor models. Since corridor modeling is based on evaluating an assembly at increments along an alignment, adjacent sections may overlap near angle points or on curves where the width of the assembly exceeds the radius of the curve. In these situations you may need to split the corridor into regions and supplement the design with feature lines and grading objects.



- Use slope and elevation labels to check your work while building the model. The image to the right shows how slope labels can be used to verify that an interpolated slope was no steeper than 4:1.

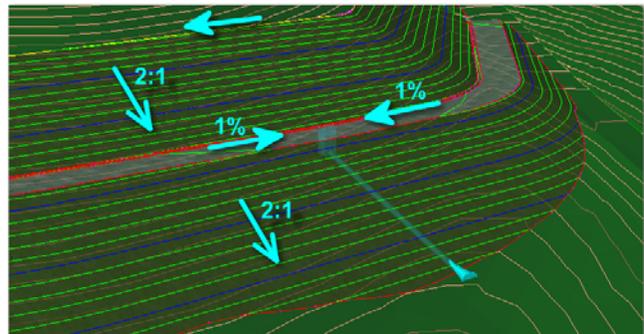


Using Stock Subassemblies to Model a Slope with a Graded Bench

The first grading model that we will look at demonstrates the design capabilities that can be achieved by utilizing some of the stock subassemblies that come with the AutoCAD® Civil 3D® program.

This demonstration shows how to use a corridor to model a fill slope with an intermediate bench. Some of the design criteria for this model includes:

- The top-of-slope hinge line will have fluctuating grades to accommodate site drainage.
- The bench is graded independently of the top-of-slope to accommodate vehicle access and 1% grades towards several drainage inlets.
- In addition to the linear grade requirements, the slopes need to maintain an exact 2:1 gradient as they extend down to the existing ground surface.



Original Grading Solution

The original methods devised for building this model relied only on grading tools. Most of it was created with feature lines. The chosen workflow resembled the process that one would use if the design were sketched out and solved on a sheet of paper with a pencil and scale.

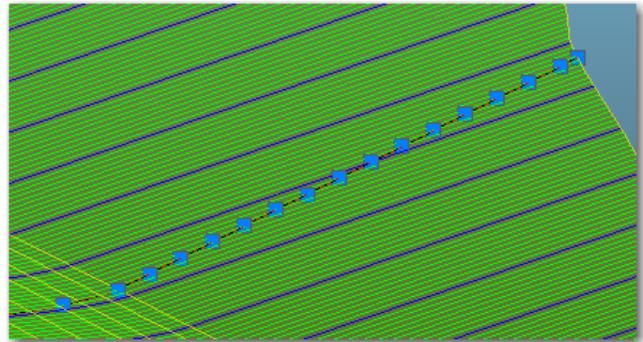
The first step was to create a temporary surface model of the initial 2:1 slope without the bench. A style was assigned to this surface to display contours at a 0.1 foot increments.

A feature line, representing the toe of slope at the bench, was drawn with 10 foot segment lengths between the adjacent, 0.1 foot contours. Since the 10 foot segment lengths were 'eyeballed', this process generated a feature line that was at an approximate grade of 1%.

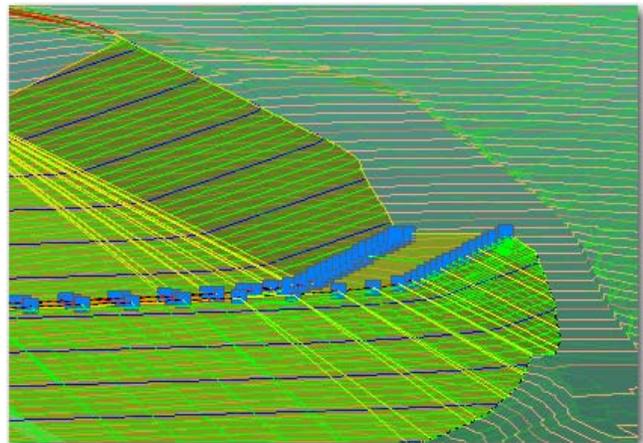
The feature line stepped offset command was used to create the other feature lines needed to model the bench.

The model was completed with a grading object at a 2:1 slope targeting the existing ground surface.

This workflow created a fairly accurate model of the design. However, the construction of the initial bench feature line at 1% grades took a while to digitize. An even more critical drawback to this method was that all of the feature lines were static. Because of this, the lengthy process had to be repeated for every change in the pad, slope, or bench criteria.



Initial bench feature line



Bench modeled with feature lines and grading object.

Corridor Model Solution

With a corridor model we were able to realize several benefits beyond what we were able to accomplish with feature lines alone:

- Corridor modeling broke the process in to separate horizontal and vertical components with alignments and profiles.

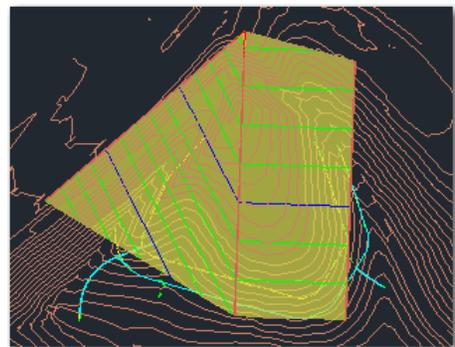
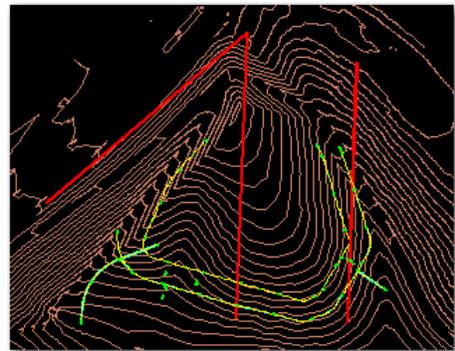
- By using the LinkSlopeAndVerticalDeflection subassembly, the corridor automatically generated the location of the bench where a precise 1% bench grade would intersect the 2:1 slope.
- All of the elements and resulting surface were dynamic. The model could be easily updated with simple edits to target surfaces, alignments and profiles.

The following exercises highlight the steps involved to create a dynamic benched slope grading model.

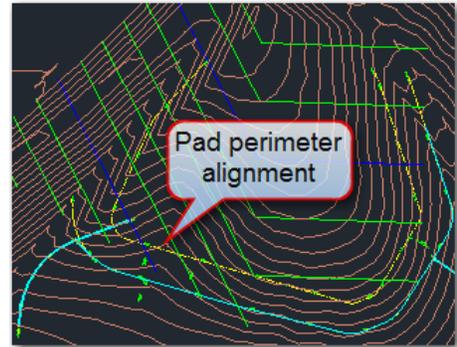
Exercise 1 – Pad perimeter elevations from a temporary pad surface

Almost all grading workflows can be enhanced by breaking the design up into smaller temporary, or intermediate, surfaces. Instead of trying to calculate an irregular grade along the perimeter of the building pad, model the site grades with a simple surface model. Then extract the perimeters elevations from it.

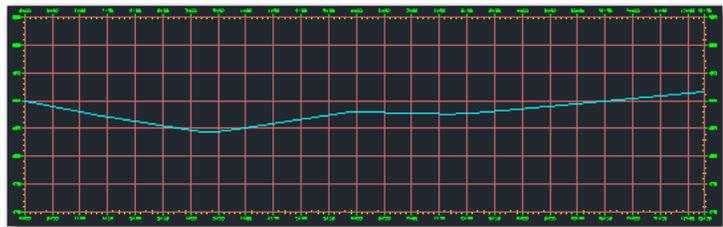
1. Using feature lines, or other methods, create a model of how the pad area will drain. Include control point elevations and hinge lines. Make sure that the extents of the model contain the area of what will be the final pad model. If the temporary pad surface area is smaller than the pad perimeter, the program will be unable to properly extract elevations. For the sample project, the overall site grades were established with three feature lines.
2. Create a temporary surface model from the feature lines.



3. Create an alignment of the pad perimeter. This alignment will be used as the corridor baseline.



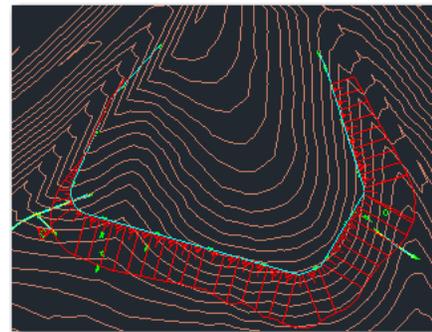
4. Use the **Create Surface Profile** command to create a profile from the temporary pad surface. This profile will be used for the corridor baseline profile.



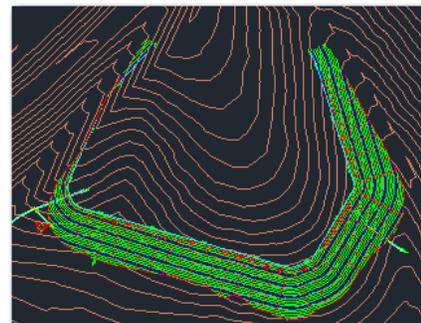
Exercise 2 – Temporary Surface of simple 2:1 Slope

Creating a single 2:1 daylight slope surface will aid in locating the bench profiles and drainage structure rim elevations.

1. Use the **Create Feature Lines from Alignment** command to create a feature line from the pad perimeter alignment and profile created in the last exercise.
2. Use the Grading Creation tools to create a 2:1 slope from the feature line down to the existing ground surface.



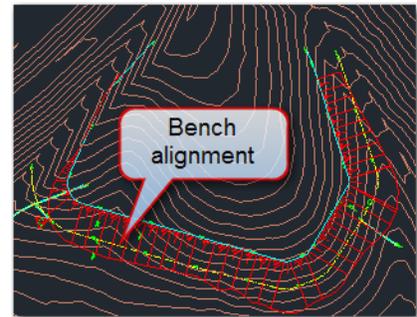
3. Create a surface model from the grading group, or by using the top and toe feature lines as breaklines.



Exercise 3 – Bench alignment and profile

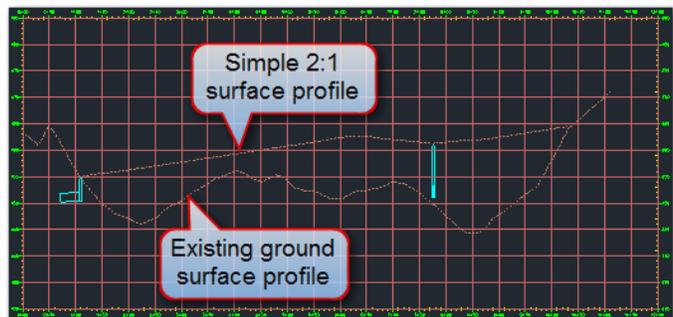
1. Layout an alignment at the approximate location of the bench's inside edge. Place PI points and curves to correspond with similar geometry in the pad perimeter alignment.

This alignment could also be created by offsetting the pad perimeter alignment. However, since the two alignments will not be parallel, the offset transitions may become too difficult to edit.



2. Use the **Create Surface Profile** command to create profiles from the temporary 2:1 slope surface, and the existing ground surface.

Since the bench needs to be graded to drainage inlets, pipes and structures have been added to the model at this time.



3. Using the **Profile Creation Tools**, create a finished grade profile for the bench. Use the simple 2:1 surface profile as a guide for the location. This layout profile will be used as the vertical deflection target for the LinkSlopeAndVerticalDeflection subassembly.



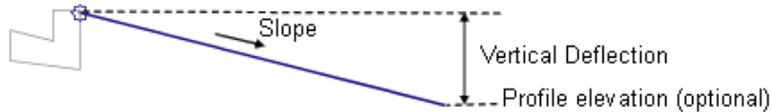
Exercise 4 – Assembly Creation

The power of using corridors for grading comes from how the assembly is constructed and the capabilities of the subassemblies that are used. This example uses the LinkSlopeAndVerticalDeflection subassembly. Looking at the help document for this subassembly we can see that it will allow the use of a target profile to generate the vertical deflection value at each corridor section. It will apply the given slope value to the vertical deflection to calculate the length of the link.

LinkSlopeAndVerticalDeflection

This subassembly is a general purpose utility to extend a link at a slope for a vertical deflection distance.

A profile can be given for the vertical deflection when building a corridor model.

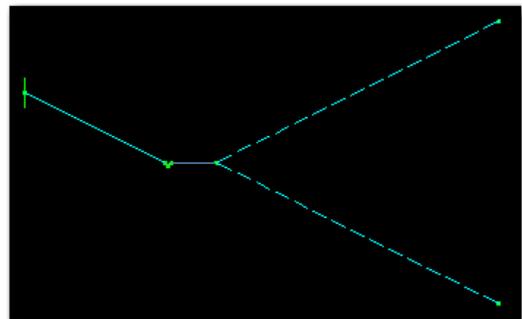


Behavior

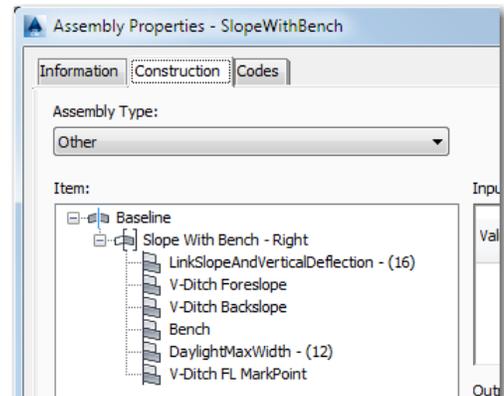
A link is added from the attachment point in the direction of insertion for the given slope until it reaches the given elevation. If a profile name is specified at runtime, the vertical deflection is adjusted to match the profile elevation.

We can exploit this behavior to locate the horizontal offset from the pad perimeter to the finished grade profile elevation of the bench.

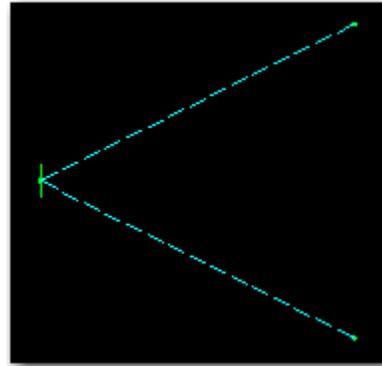
1. Create an assembly with the **LinkSlopeAndVerticalDeflection** subassembly as the first element. Use Generic link subassemblies to model the ditch and bench areas. Finish by adding a **DaylightMaxWidth** subassembly, with the same slope values as the LinkSlopeAndVerticalDeflection subassembly.



2. To make the assembly easier to work with in the corridor, use the **Assembly Properties** command to rename the subassemblies and groups. Give them meaningful and descriptive names. This extra effort will save a lot of headaches later on when you are trying to map targets on complex corridor models.



3. Create a second assembly that has only the **DaylightMaxWidth** subassembly. This will be used in the end regions of the corridor where the daylight slope will be too short for the bench.



Exercise 5 – Corridor Model

Now that we have all of the supporting components created we are ready to create the corridor model.

1. This is a relatively small model, so we are going to set the frequency to **5'** along tangents and curves.

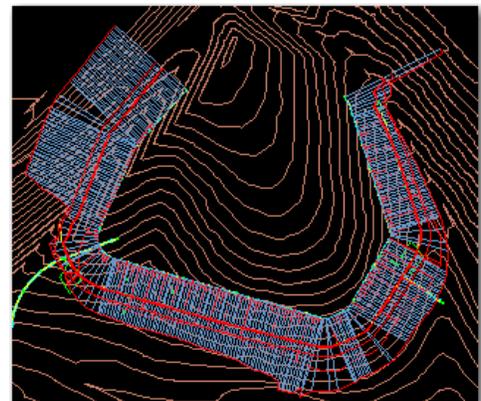
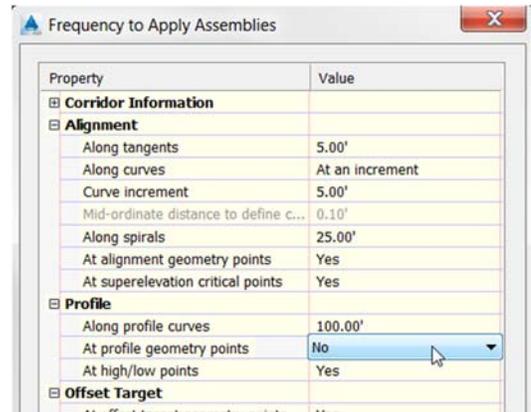
Increase these frequency values during the design process to improve software performance. Then shorten them to improve precision as your model nears completion.

2. Set the At profile geometry points option to **No**.

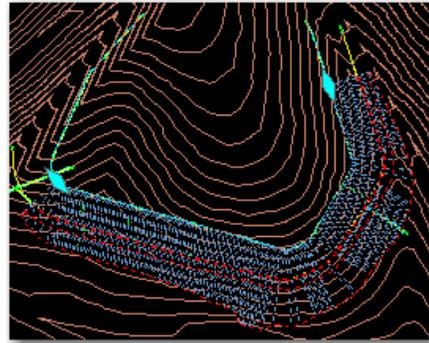
The profile we are using was sampled from a surface, causing it to have a large quantity of grade breaks. Setting this option to No will keep the corridor from sampling a section at every profile grade break.

3. Map the bench profile to the LinkSlopeAndVerticalDeflection subassembly's vertical deflection profile target. This is the profile that defines the cross-slope grade of the bench.
4. Map the existing ground surface to the daylight subassembly.

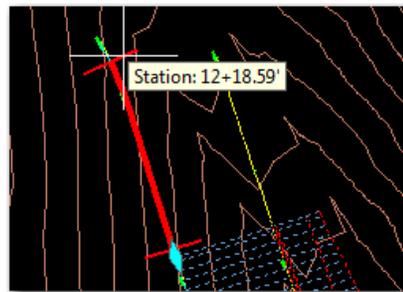
Initially, the corridor region will extend beyond the limits of the Bench alignment. Because of this, you will get "Target objects not found" error messages.



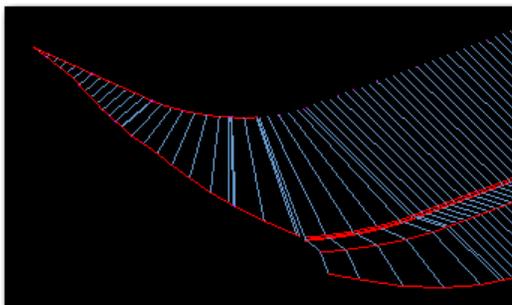
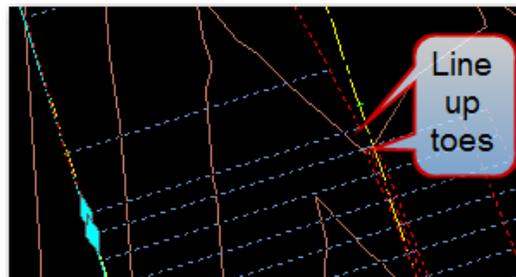
5. Use grips to drag the ends of the corridor region so that the bench slope occurs only in the desired fill areas.



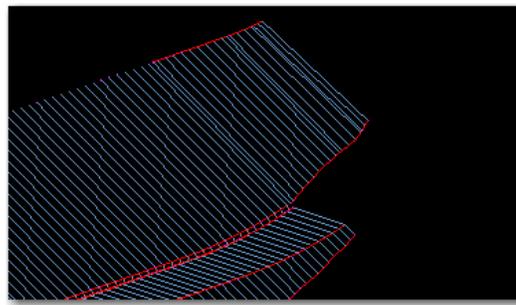
6. Add new regions on both ends of the corridor. Leave a slight gap between the adjacent regions. This makes it easier to grip edit the regions independently.
7. Assign the SlopeOnly assembly to these regions.



8. Use grips to slide the region boundaries along the alignment so that the toe of the slope only region aligns with the toe of slope at the back of the bench.

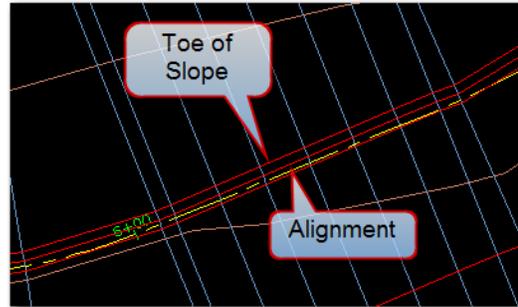


Slope only region near beginning of the benched slope.

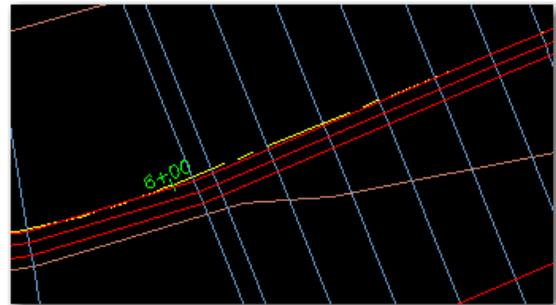


Slope only region near end of the benched slope.

Since we did not know where the toe of slope at the bench would occur when we created the bench alignment they will probably not line up with each other.



9. Edit the bench alignment location so that it aligns with the toe of slope calculated by the corridor model. This can be done by grip-editing the alignment's line and curve segments.



10. Changing the alignment will shift the relative location of the profile. You will need to make edits to the profile to get the desired grades along the bench.
11. Rebuild the corridor after editing the bench alignment and profile.

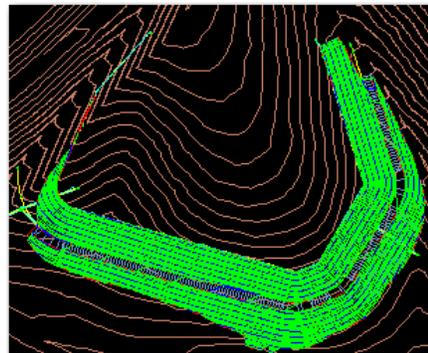
Depending on how far off your original bench alignment was, it may take several iterations of editing to get the alignment location and profile to match the toe of slope calculated by the corridor.

At this point the model is complete. Because of the way it was constructed, design revisions can be implemented fairly easily. As we saw above, the bench alignment and profile can be quickly modified to update the bench grade and location in the corridor. Similar edits can be done to the pad surface and perimeter alignment, which will then be applied when the corridor is rebuilt.

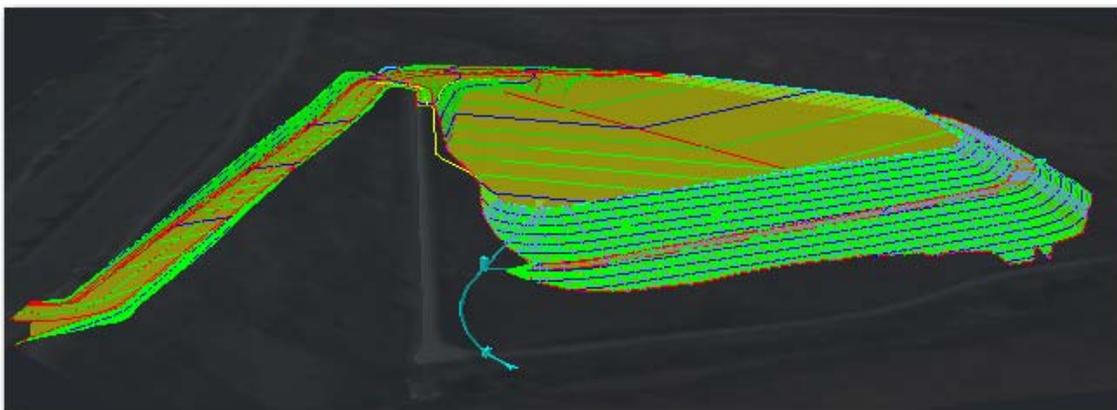
Exercise 6 – Corridor Surface

1. Create a corridor surface from the Top links and Ditch_FI feature lines.
2. Create a boundary using the **Corridor extents as outer boundary** option. This will shrink-wrap a boundary around the perimeter of the model.

Be careful with this option, because sometimes the corridor's extent may be greater than the extents of the links that were selected for the surface.



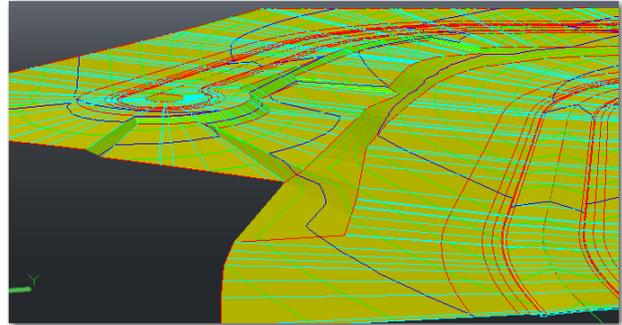
The completed corridor surface model can be combined with other surfaces, feature lines, and grading objects. The following image shows the bench slope surface pasted into the final pad model. The access road and remaining pad details were added as breaklines from feature lines.



Lot Grading with Conditional Horizontal Target Subassemblies

This chapter shows the usage of corridor modeling to create a dynamic surface model of a subdivision grading plan. This workflow is enhanced by incorporating conditional horizontal subassemblies to control the placement of the various lot grading configurations.

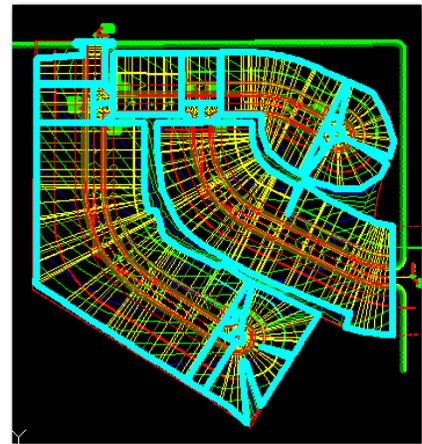
A single corridor was used to model an over-lot (rough) grading plan of a residential subdivision. This type of site grading is used in colder climates where homes are built with basements or crawl-spaces. The design concept establishes final street grades, and provides for the future disposal of left over soil that will be generated when basements are excavated.



Conditional vs. Regions – The original version of this model was created in an earlier version of the AutoCAD® Civil 3D® program in which conditional subassemblies were not available. This required that each possible cross section had to be modeled with individual assemblies, and the corridor needed to be divided into numerous regions wherever the section changed. That model required 12 different assemblies (out of 51 possible combinations) and 25 regions. The large number of regions and assemblies complicated the mapping of targets and made revisions to the design very cumbersome.



In later versions of Civil 3D, with the advent of conditional subassemblies, this same model could be built with fewer assemblies and regions. The number of regions was reduced because lot placement could be controlled with target polylines instead of regions. The model still needed multiple regions to accommodate each of the different baseline alignments, and to model the curb returns and cul-de-sacs. By using conditional subassemblies, this version of the corridor needed only 5 assemblies and 15 regions.



For preliminary design, the model could be further simplified by omitting the curb returns. This was tested on this site and the difference in earthwork volumes was less than 1%.

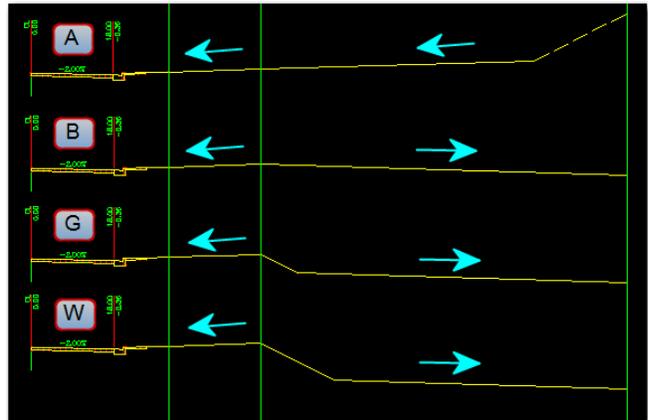
This project had four basic lot types that needed to be modeled with assemblies:

A - Drains from the rear of the lot to the street. The rear lot slope will be interpolated from the elevation of the adjacent parcel at the rear of the lot.

B – This is a split draining lot where the front half of the parcel drains towards the road, and the rear half drains towards the back property line.

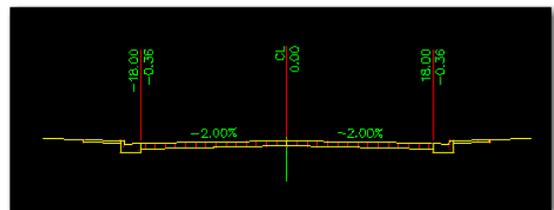
G – This is a garden level lot where the rear portion is four feet lower – exposing half of the basement wall.

W – This is a walk-out lot where the rear portion is 8 feet lower – exposing the entire basement wall at the back of the house.

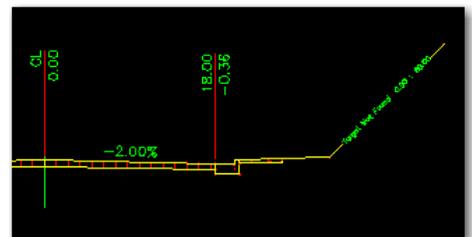


Exercise 1 – Creating the Lot Grading Assembly

1. Start by building the basic roadway configuration out to the right-of-way location. This is the portion of the model that will be identical for all full-roadway regions in the corridor.



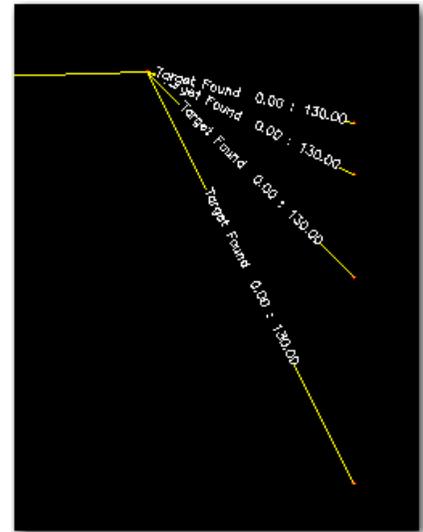
2. The next step is to add a conditional horizontal target subassembly to the right side. It will be used to determine if the assembly stops at the right-of-way, or continues to grade the parcels.
3. Set the target type of this subassembly to be **Target Not Found**. By using this option, target polylines will only need to be created in areas where the lots are not going to be graded from the roadway. This will occur at curb returns and along streets that are adjacent to side lot lines.
4. Set a maximum distance value that will keep it from targeting polylines used to control assemblies in other regions. This is critical if you assign target polylines by layer.



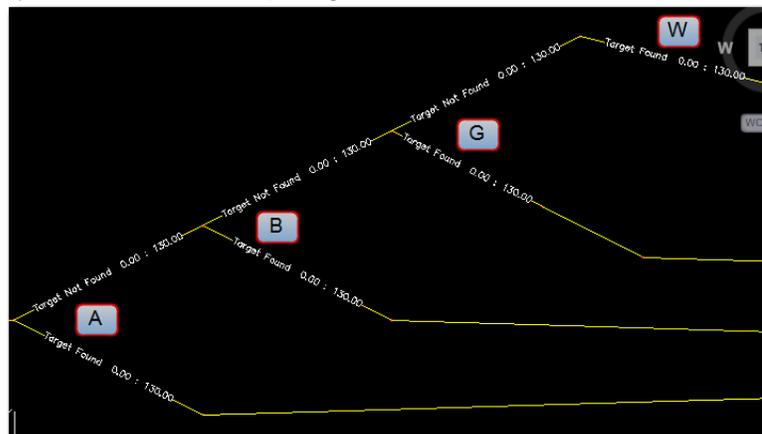
- Add any subassemblies that are common to all lot types. This model has one link subassembly with a 50 foot offset and a grade of 3%. This establishes a common hinge point for all of the parcel types used in this model.



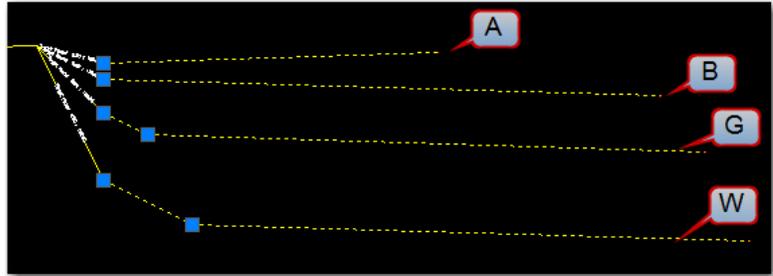
- Add four conditional horizontal target subassemblies - one for each lot type.
- Set the target type of these subassemblies to be **Target Found**. A target polyline will be required in each parcel to indicate which lot grading template to follow. If none of the targets are found, the assembly will end at this location.
- Set a maximum distance value that will keep the subassemblies from targeting polylines used to control assemblies in other regions. This is critical if you assign target polylines by layer.
- Fan out the subassemblies by giving them different layout grades. The image to the right uses grades from 4:1 for the flattest, to 0.5:1 for the steepest.



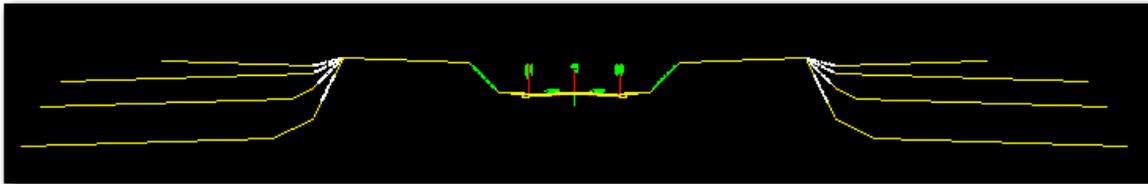
Warning: You will need to be careful when assigning target polylines. If more than one of these conditions is able to locate a valid target, it will create links on top of other links. You can avoid this by cascading the conditional subassemblies off of opposing “target not found” conditional subassemblies. The following image shows how this would look. The assembly will first test for type A targets, if they are not found, it will then test for type B. If they are not found it will then continue up the chain to the type G and W conditions. During corridor creation you will need to map targets to all the conditional subassemblies.



10. Add the link subassemblies to complete the section for each of the four different lot types.



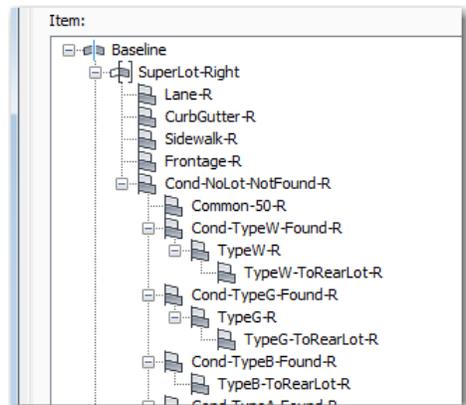
11. Repeat steps 2-10 for the left side of the assembly.



Note: The Mirror subassembly command does not work well with this assembly because most of the elements are being modeled with LinkOffsetAndSlope subassemblies. This subassembly does not have a side property, so the Mirror function will only copy them to the other side. You would still need to edit each mirrored link to put in the correct offset values.

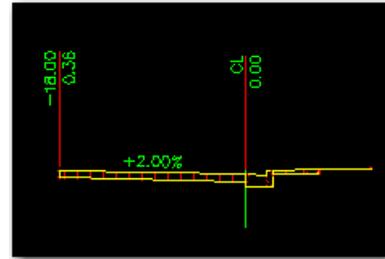
12. Rename the regions to include the assembly name and side.
13. Rename all of the subassemblies to give them names that describe their location in the model. Include an L or R character to indicate side.

Investing five minutes now, to name all of the subassemblies and regions, will simplify mapping targets when editing the corridor model. Plus, it will save you countless hours of trying to figure out why some of the targets may not be working.

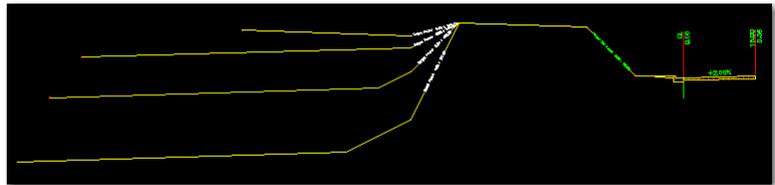


14. This model also required separate assemblies for the curb returns, cul-de-sacs, and the half sections that occur in tee intersections. The following assemblies were created by copying the first one and deleting unneeded elements. Or, new assemblies were created and the subassembly "Copy to" command was used to copy the necessary elements from other assemblies.

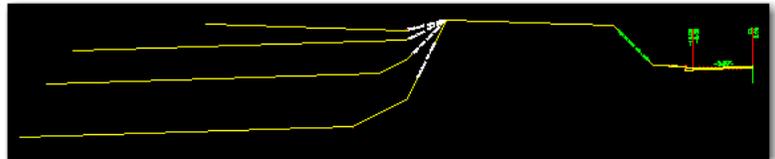
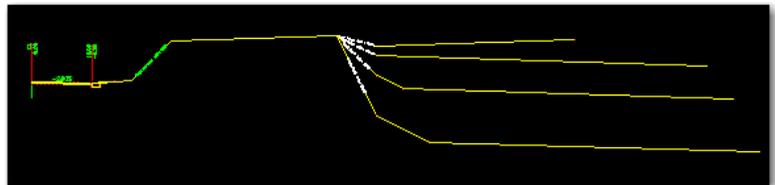
Curb return – Insertion point is edge of pavement. Only grades road to right-of-way.



Cul-de-sacs – Insertion point is the left edge of pavement for clockwise alignment directions. It includes subassemblies for grading lots.

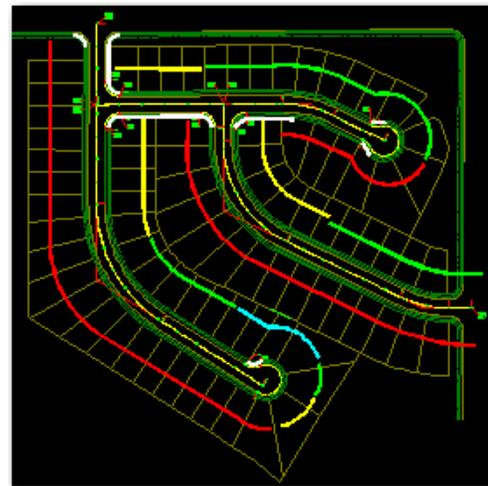


Right and left half sections – Inserted at roadway centerline. These would be used in tee intersections for the portion of the main road that continues through. These two assemblies can be easily created from copies of the full road section assembly.

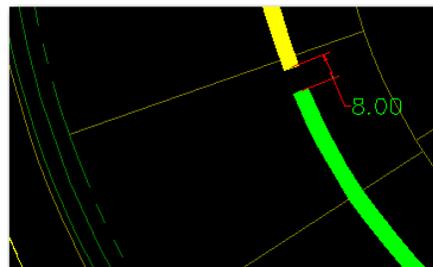


Exercise 2 – Target Polylines

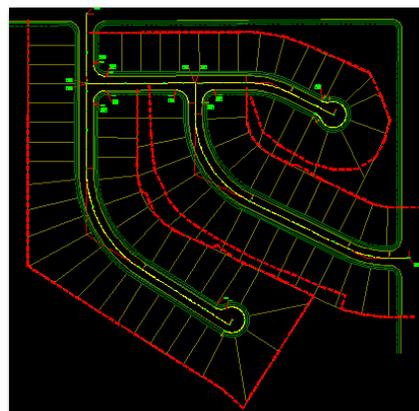
1. Create a unique layer for each of the five types of horizontal target conditions: No Lots (white), Type A (red), Type B (yellow), Type G (green), and Type W (cyan). Assigning AutoCAD index colors can help you quickly identify what lot type the linework represents.
2. Draw polylines, parallel to the lot frontage, through all of the parcels. Draw them on the layer that corresponds to the assigned lot type. Make sure that the polylines are drawn within the maximum offset distance limits that were set on each of the conditional subassemblies.



3. Leave gaps between adjacent segments to accommodate the modeling of the transition slope between different lot types. The image to the right shows an 8' gap between Type B and Type G lots. This will create a 2:1 slope from the type B lot, down to the 4 foot lower type G lot.



4. Create another layer with polylines that define the rear lot locations. These will be used as horizontal targets to set the extents of the subassemblies in the rear portion of the lots. To avoid targeting confusion, you will need to break the polylines at the midpoint of the cul-de-sac regions. This will keep the subassembly from targeting the polyline on the wrong side of the cul-de-sac.



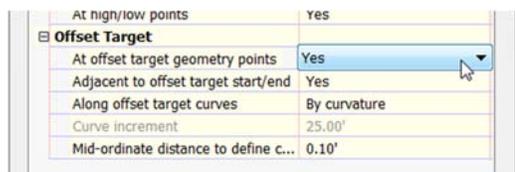
Exercise 3 – Assigning Assembly Targets to Corridor Regions

After creating the corridor with baselines for each of the centerline, curb return and cul-de-sac alignments, and dividing them into regions based on pavement sections; we will need to specify the targets for the conditional subassemblies. This will be done to control how the lots are graded. Offset targets will also need to be assigned to all of the link subassemblies that extend to the rear of the lots.

Using the regional corridor editing tools from the shortcut menu simplifies the target mapping process. Editing by region allows you to visually associate the desired targets within the region boundaries.

For this assembly, it takes about 2-3 minute to map all of the target polylines in a region. With 15 regions in this model, it would take about half an hour to map all of the target polylines.

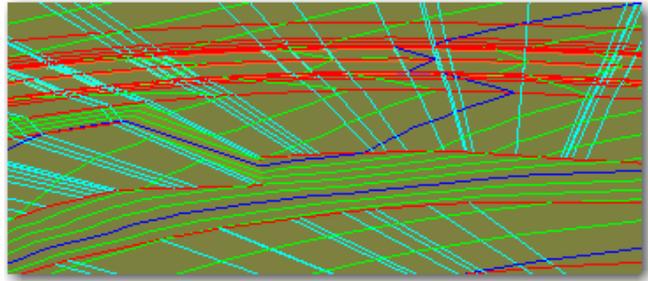
After adding all of the targets, edit the frequency for all regions. To accurately model the transitions between different lot types, set the At Offset Target Geometry Points frequency option to **Yes**. This will add corridor sections at all of the target polyline vertices and endpoints.



Exercise 4 – Revising the Model

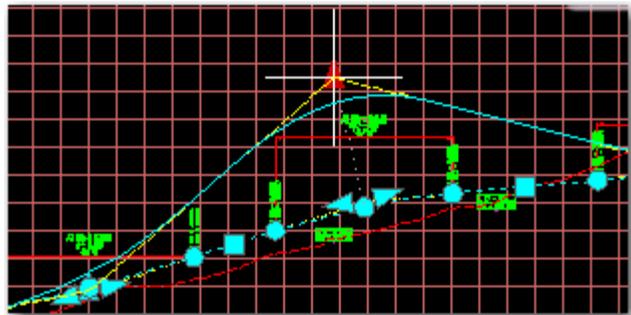
After investing a significant amount of time in building assemblies, creating a corridor with multiple baselines and regions, and targeting dozens of polylines, we are ready to reap the benefits of our efforts. By creating a site grading model in this manner; we will be able to quickly edit elevations of entire streets just by editing the corresponding profile. Or, we could easily change lot types just by grip editing and shifting the locations of target polylines.

1. The image to the right shows the initial corridor and surface contours. Note the location of the type G lots on the left, and the B lots to the right. Also note the height of the slope at the rear of the lot.

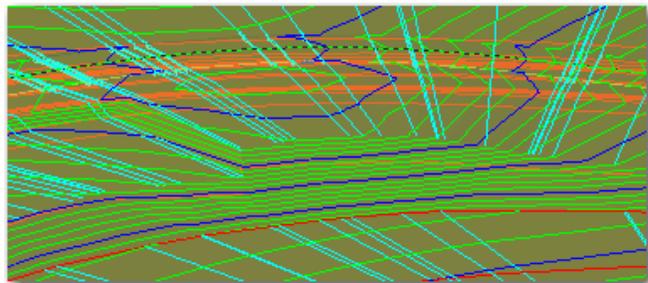


2. Grip editing the layout profile of the baseline alignment, a PVI was raised 10-15 feet.

Note: For this demonstration the profile was edited in an area that does not coincide with any cul-de-sacs or curb returns. If it did, the affected profiles would also require editing to ensure seamless transitions between the regions.

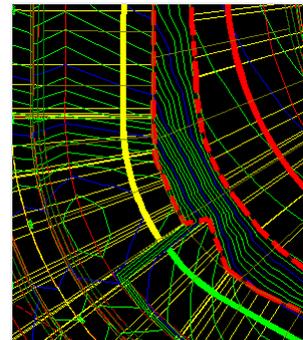


3. **Rebuild** the corridor.
The lots will retain their same basic shape relative to the raised alignment. The slope at the back of lots should now be a lot higher, and steeper – displaying more contours.

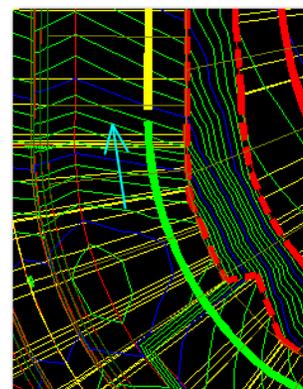


The height of the rear lot slope can be reduced by converting the lots from Type B to Type G (garden level).

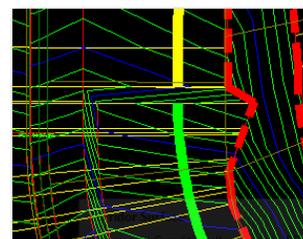
- 4. ZOOM into the area of the model where the lots were raised with the previous profile edits. A yellow polyline is being used to control the placement of Type B lots, and a green polyline is controlling the placement of the Type G lots.



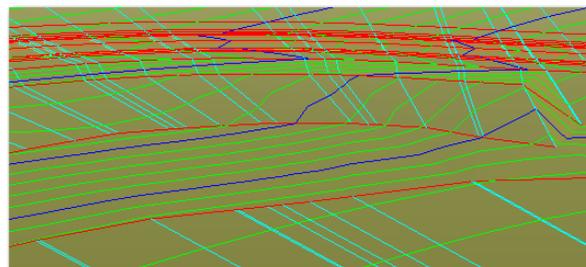
- 5. By editing the yellow and green polylines, the gap was moved several property lines to the north.



- 6. Extending the rear lot polyline will create a smoother transition between the Type G and Type B lots.



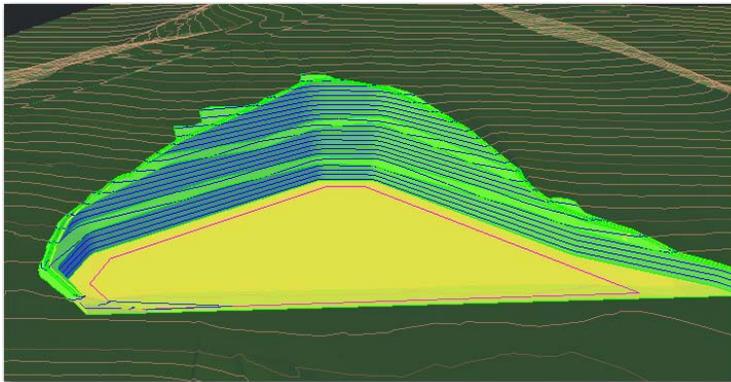
- 7. After editing the polylines, the model was updated by rebuilding the corridor and surface.



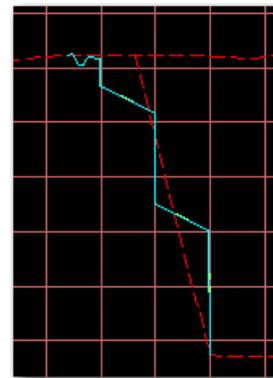
Walls with Conditional Cut/Fill Subassemblies

The last chapter demonstrated how to use conditional horizontal target subassemblies to control grading conditions with target polylines. That method simplified the model editing process, but still required human input to determine the locations where changes in assemblies occurred. This chapter will improve upon this process by using conditional cut/fill subassemblies to automatically determine where changes in the assembly need to occur.

The design example for this chapter is a cut slope model with three retaining walls. The walls will be separated by ten foot wide, sloped benches. The client wanted the tops of the walls to have undulating profiles to mimic the hills on the horizon. The final design element is a ditch at the top of the cut to catch drainage from the existing slopes above the site.

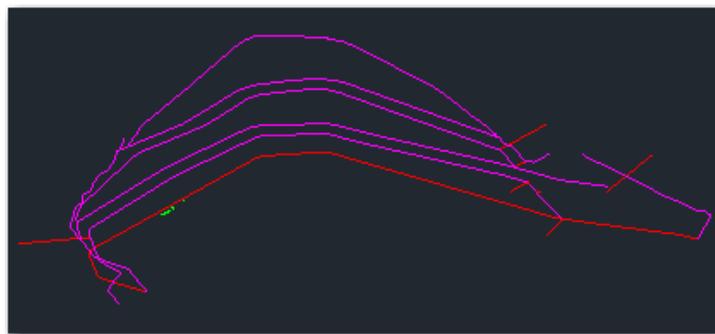


Model view



Section view

Feature line model – We originally attempted this model with grading tools alone. It contained dozens of feature lines that became very difficult to manage and edit. Every change of the first wall's height required a recalculation of the end locations of the walls above it. It was also very difficult to model and edit the curved undulations of the wall tops with feature lines.



Model view of feature lines

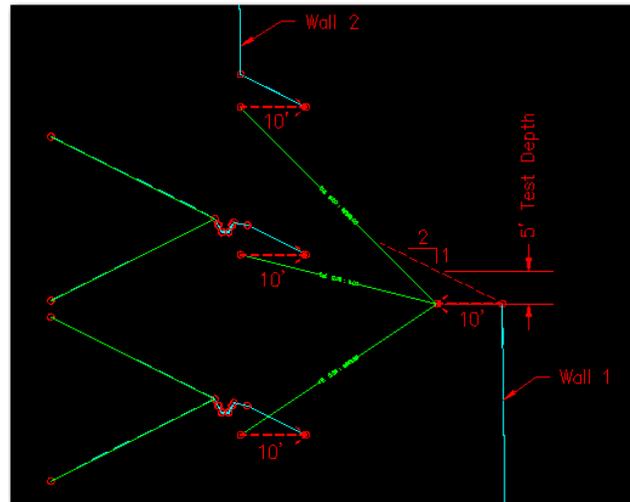
Superimpose Profiles – The superimpose profile function allowed for each wall to have its own profile view while still being able to visually reference the heights of the other two walls. The following image shows a portion of the profile view for the middle wall (Wall2), with the other two wall profiles superimposed.



Offset to Conditional Cut/Fill – For this model we needed to test, where the existing ground surface elevation is, at the upper end of a ten foot wide, 2:1 bench slope.

Since the conditional cut or fill subassembly is only able to check the depth to a surface directly above or below its insertion point, we needed a way to offset 10 feet to where the end of the bench would occur. At this location we could test for two cut conditions:

1. Between 0 and 5' of cut, the model will add a segment that targets the surface at a 2:1 slope.
2. Any cut greater than 5 feet will require the fixed 10' wide bench slope with the addition the next wall element.



A good way to create and manage the offset values is with LinkWidthAndSlope subassemblies. For this model they were created with a width of 10' and a slope of 0%. The side property was set to "Left" for the link that went from the top of the wall to the conditional subassembly. Similar links were used with the side property set to "Right", for the returning offset from the ends of the conditional subassemblies to the next component in the section. Once the assembly had been constructed, all of the 'offset' subassemblies were selected, and their Omit property was changed to "Yes" – to prevent them from displaying in the corridor model.

Exercise 1 – Alignments

Separate alignments were needed for each retaining wall to allow for accurate profiling of their tops. Since all three walls were to remain parallel, this was a good application for offset alignments.

1. Use the **Alignment Creation Tools** command to create the main alignment for the base of the first wall. This alignment will have two profiles. The first profile will control the base of the wall at the edge of the pad. The other profile will control the height of the first retaining wall.



2. Use the **Create Offset Alignment** command to create the other two wall alignments from the first.
Set the number of left offsets to **2**, and the right offsets to **0**.
Set the left incremental offset distance to **10**.

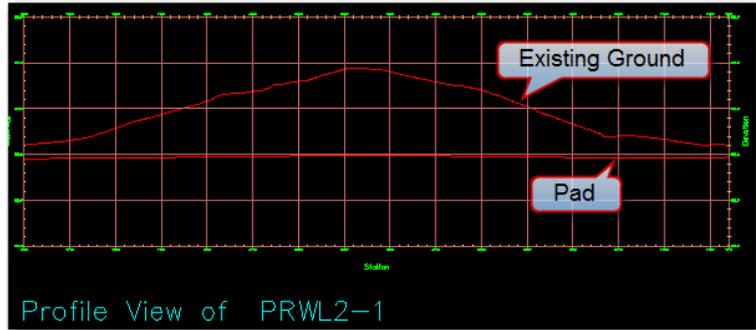
The drawing will now contain three parallel alignments. As offsets, any edits made to the first alignment will automatically be applied to the other two.



Exercise 2 – Surface Profiles

An existing ground surface profile and profile view were created for each alignment. Another surface was created to define the pad elevation and grades. This surface was also sampled for a profile on the first alignment. It will be used for controlling the corridor baseline.

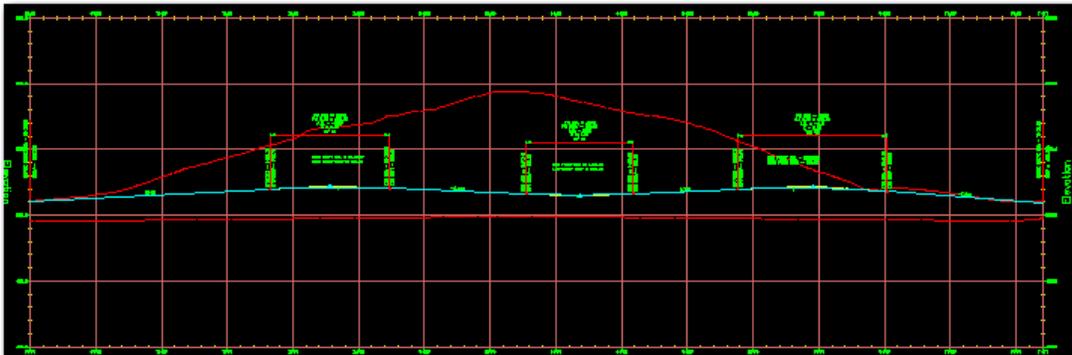
The image to the right shows the profile view, and profiles, for the first wall alignment.



Exercise 3 – Layout Profiles

Create layout profiles to represent the top of wall elevations.

1. Use the **Profile Creation Tools** to construct the top of wall profile for the first wall alignment.



The above image shows the layout profile for the first wall. It was created with fixed line segments and free circular curves.

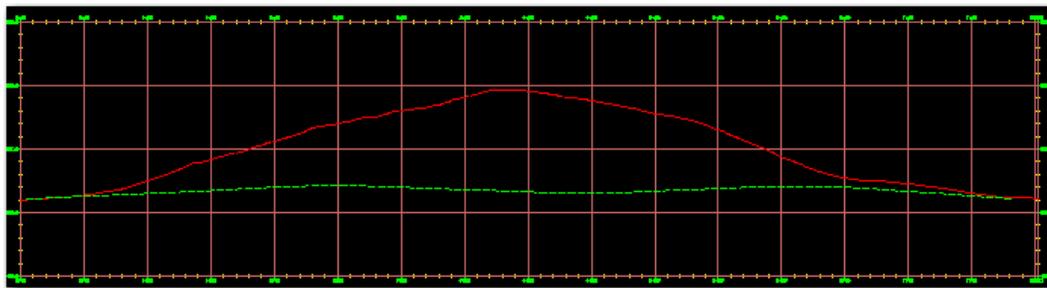
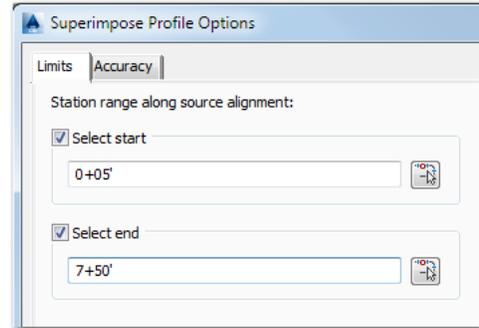
2. Repeat this process for the other two profiles. You may want to superimpose the first wall profile into the others as a guide, prior to laying them out.

Exercise 4 – Superimpose Profiles

Superimpose layout profile from each alignment into the other two profile views.

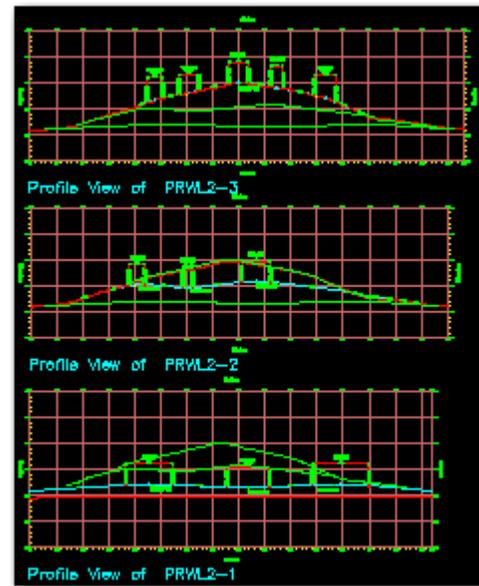
1. Run the **Create Superimposed Profile** command.

Because of the curvature of these alignments, there are potentially two stations that the start and end of the selected profile could project to. Shortening the station ranges at the ends should keep the superimposed profile from crossing back on itself in the selected profile view. It may take you a couple of iterations to get the correct values.



First wall profile projected into second wall profile view.

2. Repeat this for the other top of wall profiles and the other profile views.



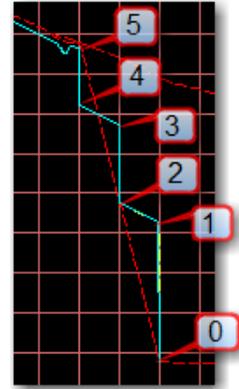
Exercise 5 - Assembly Creation

This exercise highlights the key steps in the process of creating a multi-tiered cut slope, with a combination of walls and benches. All of the components are created using the stock subassemblies that were installed with the AutoCAD® Civil 3D® program. The key elements of this assembly will be the conditional cut/fill subassemblies that are used to determine when to put in the next wall level, or when to end the section by extending the bench to the existing

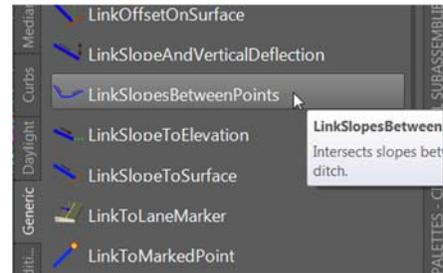
ground surface. Once the first wall and bench elements have been created, the pattern can be quickly replicated with the “Copy to” command from the subassembly shortcut menu. Since most slopes have a finite number of benches, this method is much easier than trying to create a custom subassembly with the Subassembly Composer.

1. Develop a subassembly naming plan before creating complex subassembly like this.

For this project each wall face was given a pair of numbers to indicate the bottom and top of each wall. The numbering started at 0 for the base and incremented up to 5 for the top of the upper wall. All subassemblies were then named based on the points that were spanned, or the point number that indicated the location of the element in the model.

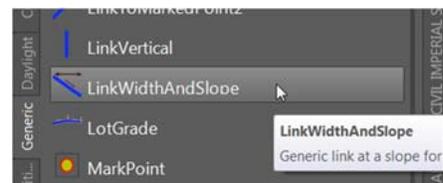


2. Use the **LinkSlopeAndVerticalDeflection** subassembly to model the retaining wall. This subassembly has a slope property can be changed to represent the face of different types of walls. It also has a target component that will allow the end elevation value to be extracted from a profile.

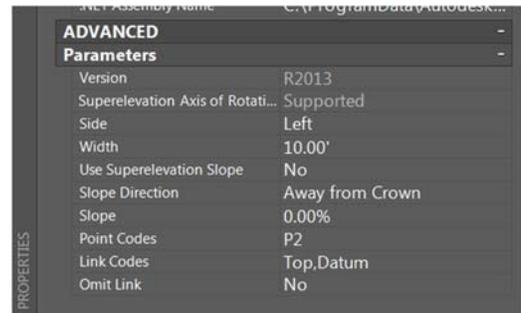


The assembly will now need to test the depth, relative to the existing ground surface model, to determine if the location is deep enough to warrant the addition of another level of wall. The test needs to occur at a location ten feet to the left – where the base of the next wall would be located.

3. In the Tool Palette window, select the **LinkWidthAndSlope** subassembly. It is located on the **Generic** tab.



4. Set the side to **Left**.
5. Enter a width value of **10**.
6. Set the user superelevation option to **No**.
7. Set the slope direction to **Away from Crown**.
8. Enter a slope of **0%**.
9. Leave the default point and link codes.
10. Set the omit link option to **No**. (We will change this later, when we are finished building the assembly.)

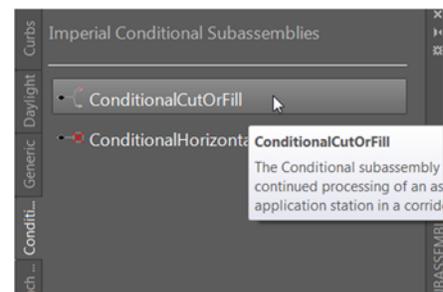


11. Select the marker point on the end of the previous subassembly.

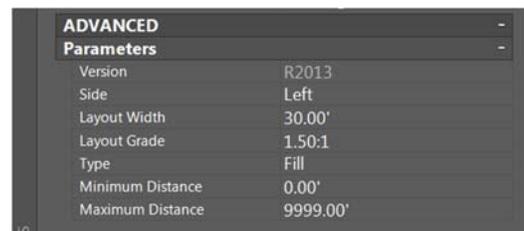


12. Press **Enter** once to finish the command.

13. Select the **Conditional** tab in the Tool Palettes window.
14. Select the **ConditionalCutOrFill** subassembly.

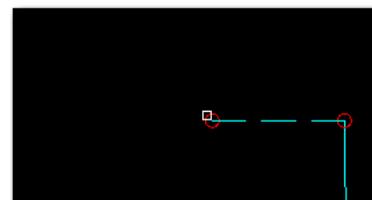


15. In the Properties window, set the side to **Left**.
16. Enter a layout width of **30**.
17. Enter a layout grade of **1.5:1**
18. Set the type to **Fill**.
19. Enter a minimum distance of **0**, and a maximum of **9999.0**. This should accommodate all fill conditions.

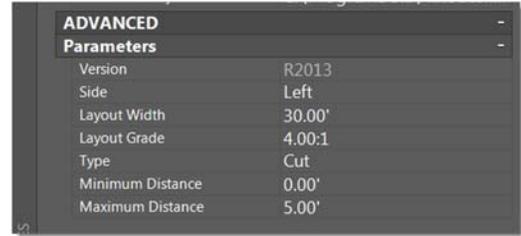


20. Select the marker point on the end of the previous subassembly.

Do not end the command yet.

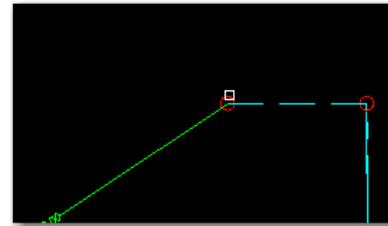


21. In the Properties window, change the layout grade to **4:1**.
22. Change the type to **Cut**.
23. Change the maximum distance value to **5**.

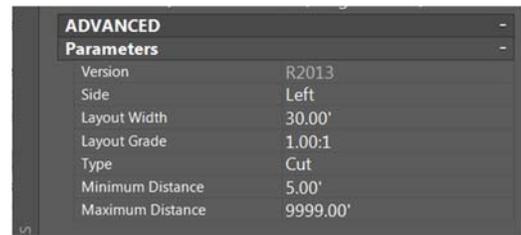


24. Select the marker point on the end of the horizontal, LinkWidthAndSlope subassembly.

Do not end the command yet.

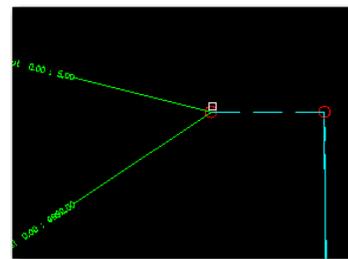


25. In the Properties window, change the layout grade to **1:1**.
26. Change the minimum distance value to **5**.
27. Change the maximum distance value to **99999.0**.

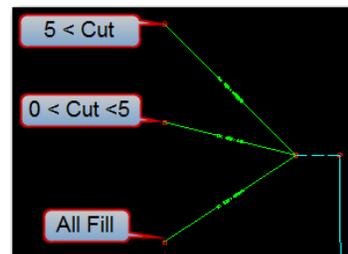


28. Select the marker point on the end of the horizontal, LinkWidthAndSlope subassembly.

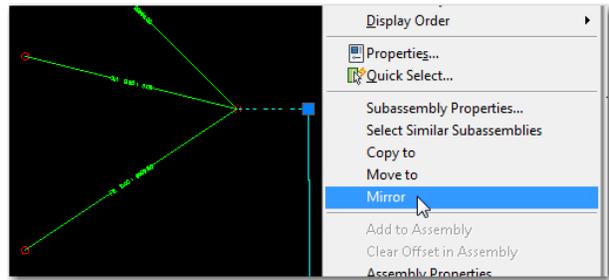
29. Press **Enter** once to finish the command.



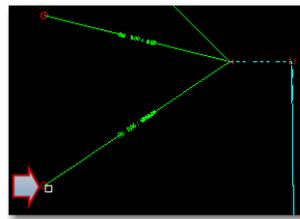
The assembly should now have three conditional subassemblies at this location – one for all fill conditions, and two that divide the cut conditions at a depth of 5 feet.



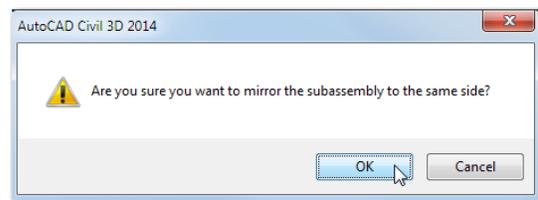
- 30. Select the horizontal, LinkWidthAndSlope subassembly.
- 31. Right-click and select the **Mirror** command from the shortcut menu.



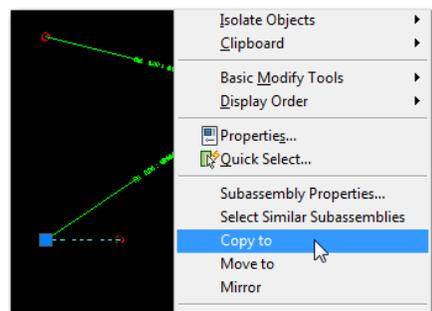
- 32. Select the point on the end of the conditional fill subassembly.



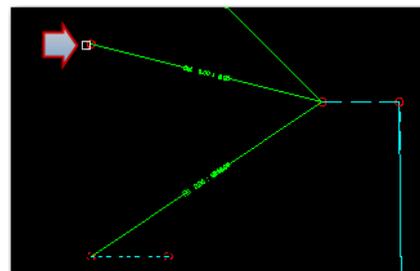
- 33. Click the **OK** button on the warning dialog.



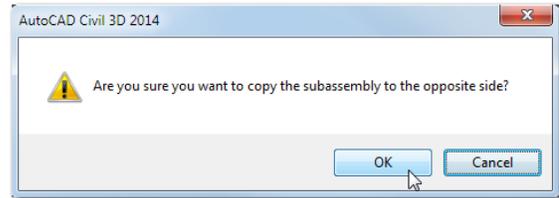
- 34. Select the subassembly that was created with the previous Mirror command.
- 35. Right-click and select the **Copy to** command from the shortcut menu.



- 36. Select the point on the end of the first conditional cut subassembly.

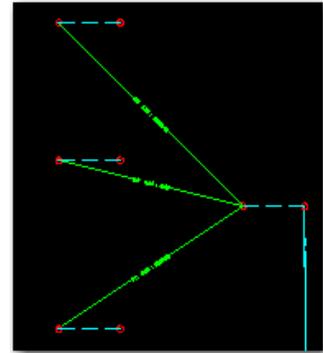


37. Click the **OK** button.

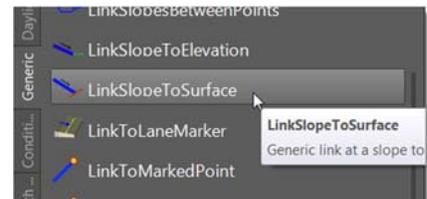


38. Repeat steps 34-37 to add a horizontal LinkWidthAndSlope subassembly to the end of the third conditional subassembly.

All three conditions should now have a link that extends 10 feet to the right. This will locate the subsequent subassemblies back at the top of the first wall.

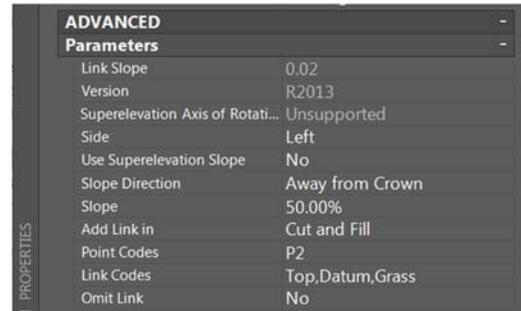


39. Return to the **Generic** tool palette tab.
40. Select the **LinkSlopeToSurface** subassembly.

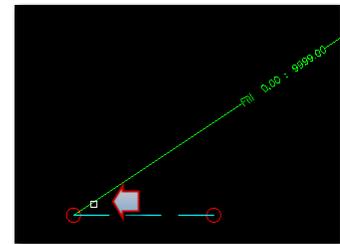


In the Properties window:

- 41. Set the side to **Left**.
- 42. Set the use superelevation option to **No**.
- 43. Set the slope direction to **Away from Crown**.
- 44. Enter a slope value of **2:1** (50%).
- 45. Set the add link option to **Cut and Fill**. Setting it to this value will allow it to be copied correctly to any of the cut or fill conditions.
- 46. Add any custom point and link codes.
- 47. Set the omit link option to **No**.

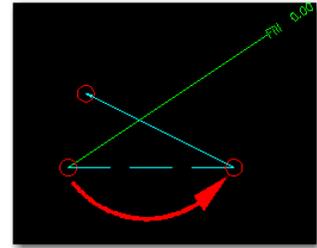


48. Select the marker point on the right end of the fill condition subassembly. If we select the end of the horizontal, LinkWidthAndSlope subassembly, Civil 3D will automatically switch the side property of the LinkSlopeToSurface subassembly to the right.



49. Press **Enter** once to finish the command.

50. Use the **Move to** command, from the shortcut menu, to move the LinkSlopeToSurface subassembly to the right end of the horizontal, LinkWidthAndSlope subassembly.

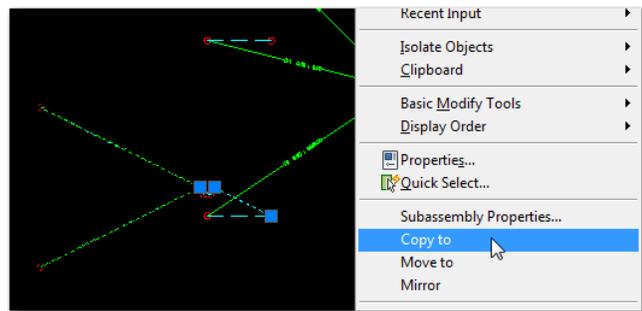


51. Finish this branch of the assembly by adding **SideDitch** and **DaylightMaxWidth** subassemblies.

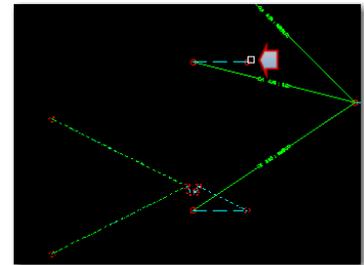
52. This model will use the exact same configuration in the 0 to 5 foot cut condition.

53. Select the LinkSlopeToSurface, SideDitch and DaylightMaxWidth subassemblies.

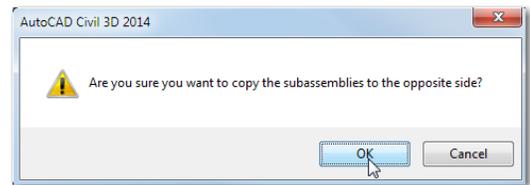
54. Right-click and select the **Copy to** command from the shortcut menu.



55. Pick the right end of the horizontal, LinkWidthAndSlope subassembly that is attached to the 0-5 foot cut condition.

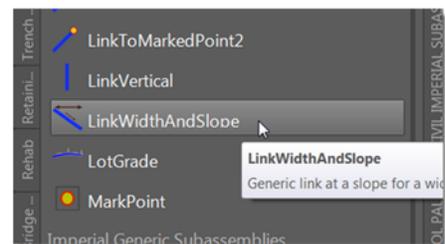


56. Click the **OK** button.

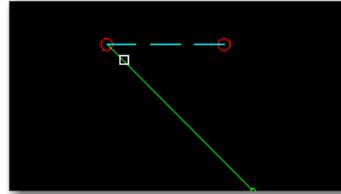


For the greater than 5 feet of cut condition we need to add a 10' wide bench.

57. This will be modeled with a **LinkWidthAndSlope** Subassembly.

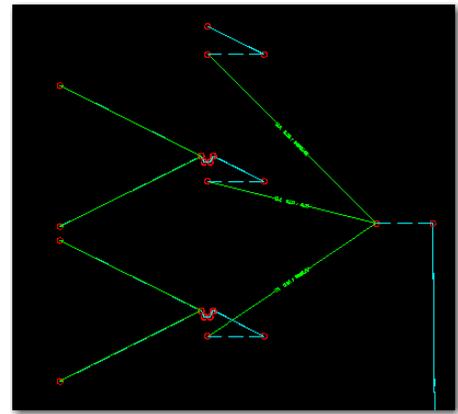


58. To prevent Civil 3D from changing the side to Right, select the end of the conditional subassembly for the greater than 5' of cut condition.

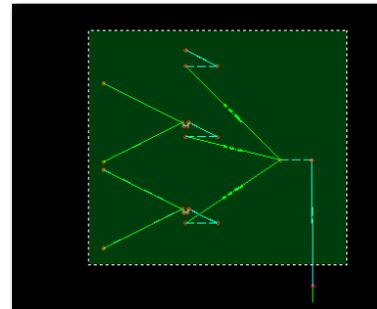


59. Use the **Move to** command from the shortcut menu to move the LinkWidthAndSlope subassembly to the right end of the horizontal, LinkWidthAndSlope subassembly.

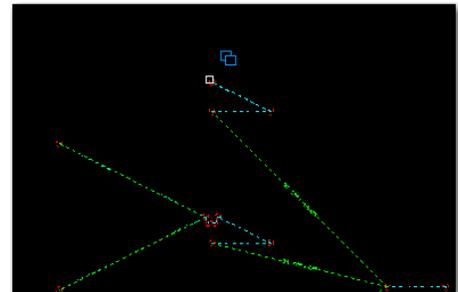
We now have the basic layout of a single tier. It starts at the base of the first wall and continues to the base of the next wall. Subsequent tiers can now be created simply by copying all of the subassemblies in this one. Just be sure to use the subassembly “Copy to” command, instead of the AutoCAD COPY command.



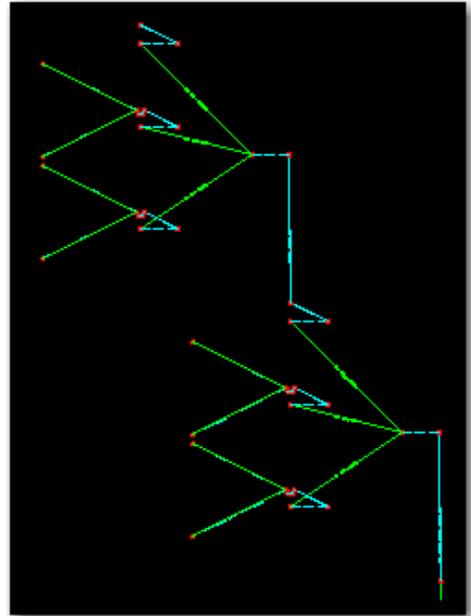
60. Select all of the subassemblies.
61. Right-click and select the **Copy to** command from the shortcut menu.



62. Select the end of the LinkWidthAndSlope that was placed on the greater than 5 feet of cut condition.

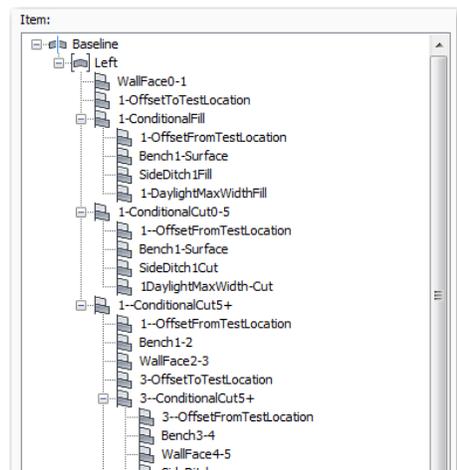


We now have two tiers of our wall system completed. We could keep repeating steps 60-62 to create additional wall and bench tiers.



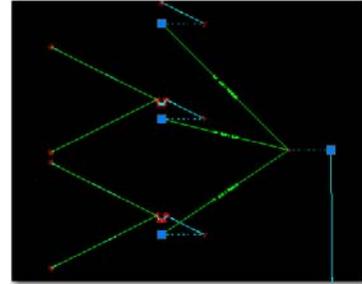
For the uppermost tier, we only need to add the wall, side ditch and daylight assemblies to the top end of the last bench. Use the subassembly “**Copy to**” command to copy these elements from other locations in the assembly.

63. Edit the assembly properties, or each subassembly’s properties, to give meaningful names to each of the subassemblies. Investing time in this step will make the target mapping process much easier when building and editing the corridor model.

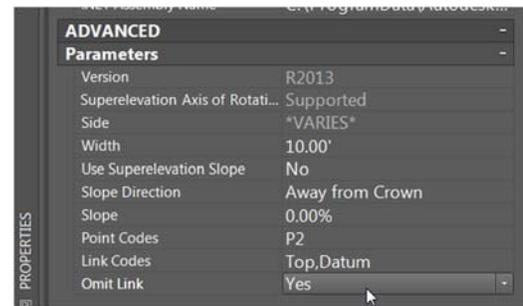


The final step is to hide the offset subassemblies to keep them from being constructed as part of the corridor model.

64. Select all eight of the horizontal links that are used to offset to and from the conditional test locations.



65. Change their omit link property to **Yes**.



Exercise 6 – Corridor Creation

1. Select the **Corridor** command from the Home ribbon tab.
2. Enter a meaningful name and description for the corridor.
3. Select the main, base of wall alignment. This will become the corridor baseline.
4. Select the profile that represents the elevation at the base of the wall. This example uses a surface profile that was generated from a surface model of the proposed building pad.
5. Select the wall assembly that was created in the previous exercise.
6. Set the target surface to the existing ground surface. This will be assigned to all subassemblies that target surfaces. For this model that will include the ConditionalCutOrFill, LinkSlopeToSurface, and DaylightMaxWidth subassemblies.
7. Toggle **on** the Set baseline region parameters options. This will open the Baseline and Region Parameters dialog after clicking the OK button.
8. Edit the corridor frequencies.
9. Set the frequency along tangents to 25.
10. Set the frequency along curves to 5. This will add more corridor sections along the curves to provide more detail in these areas.
11. Set all of the specific location options to **Yes**.

Note: For larger corridors, keep the frequency values larger, and possibly turn off some of the supplemental sampling options. This will increase the corridor rebuild speed and improve system performance while you are editing the model. Once you have refined the

model to the optimal design, reduce the frequency distances and enable more of the supplemental sampling options to create a more precise model.

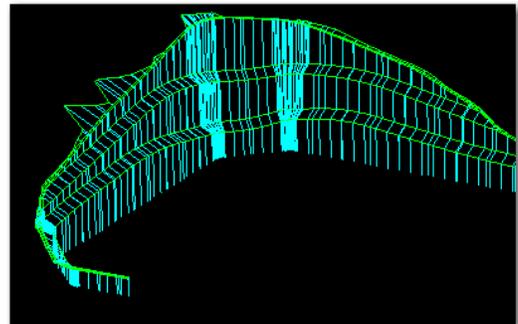
12. Edit the corridor targets.

13. Verify that all of your surface targets were properly set to the existing ground surface model. If they were not, pick on the object name field of the Surfaces row, where it reads “<Click her to set all>”

14. Besides the surface targets, the only other targets that need to be set for this assembly are the top of wall profiles. They will control the height of the LinkSlopeAndVerticalDeflection subassemblies that model the retaining wall faces.

Target	Object Name	Subassembly
Target Alignment	<None>	DaylightMaxWidth
Target Alignment	<None>	DaylightMaxWidth
Slope or Elevation Targets		
Vertical Deflection Profile	<input checked="" type="checkbox"/> PRWL2-1-PRWL2-1-Top	WallFace0-1
Target Profile	<None>	1-OffsetToTestLocation
Target Profile	<None>	1-OffsetFromTestLocation
Target Profile	<None>	1--OffsetFromTestLocation
Target Profile	<None>	1--OffsetFromTestLocation
Target Profile	<None>	Bench1-2
Vertical Deflection Profile	<input checked="" type="checkbox"/> PRWL2-2-PRWL2-2-Top	WallFace2-3
Target Profile	<None>	3-OffsetToTestLocation
Target Profile	<None>	3--OffsetFromTestLocation
Target Profile	<None>	3--OffsetFromTestLocation
Target Profile	<None>	3--OffsetFromTestLocation
Target Profile	<None>	Bench3-4
Vertical Deflection Profile	<input checked="" type="checkbox"/> PRWL2-3-PRWL2-3-Top	WallFace4-5

15. After rebuilding the corridor, use the **Object Viewer** to inspect the model.

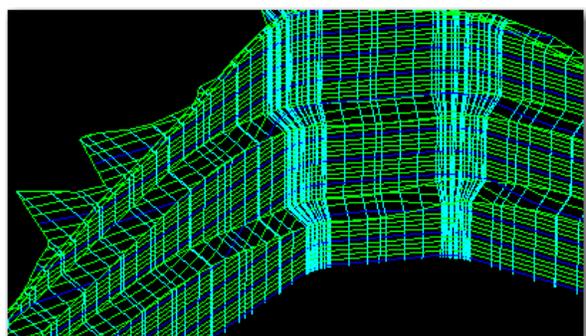


You should be able to identify sections where the conditional subassemblies evaluate differently. Wherever the depths were less than five feet you should see a 2:1 slope to the side ditch. Where the cut depth is greater than five feet you should see a vertical link for the face of a wall.



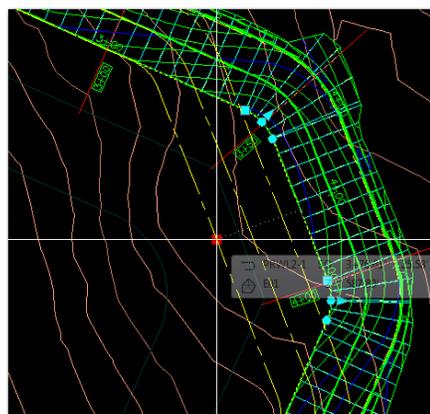
Exercise 7 – Corridor Surface

A corridor surface can be created from assembly Top links with a boundary derived from the corridor extents.

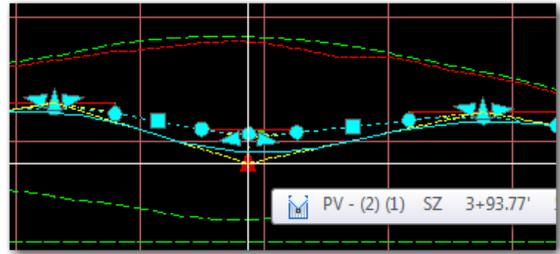


Exercise 8 – Model Editing

1. You can grip edit the baseline alignment to change the horizontal location of the walls. Since the other two alignments were created as offsets, they will also move - maintaining their ten foot relative offset distances.

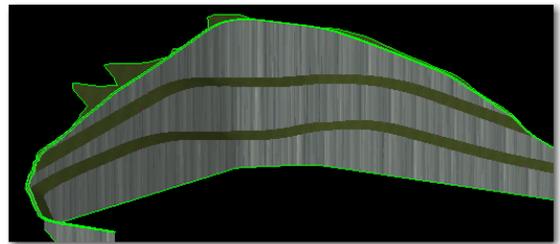


2. Edit the top of wall profiles to align them with the updated surface profiles.

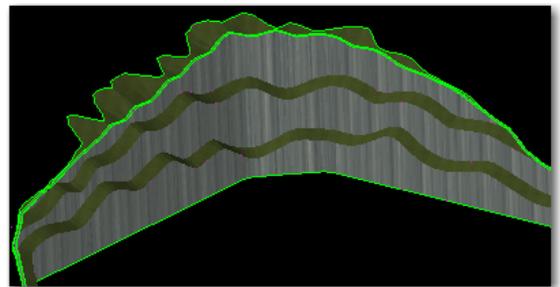


3. **Rebuild** the corridor when you are finished editing the alignments and profiles.

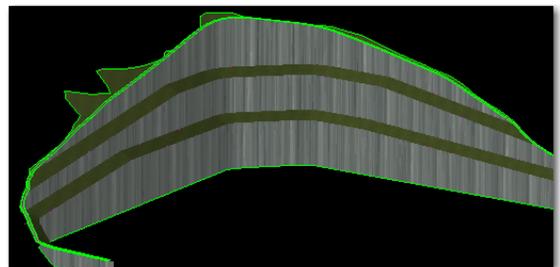
The image to the right was created with the view mode set to the Realistic visual style, and a code set that assigned render materials to link code names.



Another editing option would be to create additional layout profiles for each wall alignment. Change the target assignments to the alternatives to quickly try different configurations.



Alternative 2 – extra wavy profiles.



Alternative 3 – Straight profiles.

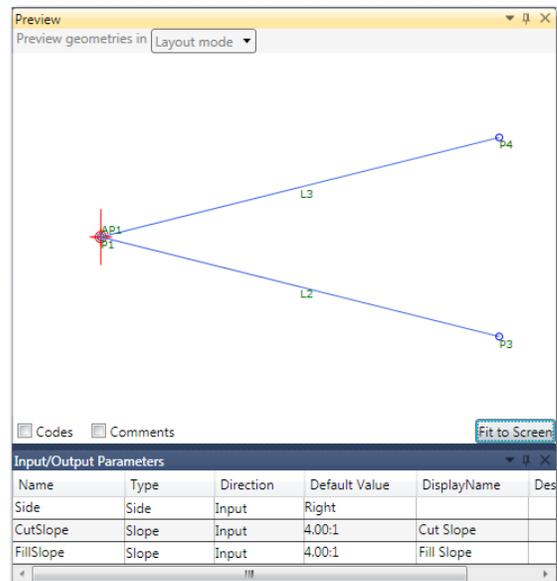
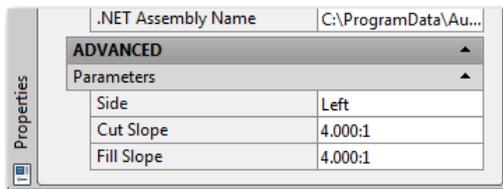
Custom Grading Subassemblies

The Subassembly Composer program is a very powerful tool that allows you to create intricate geometries or automated subassemblies that can be used to solve complex design problems.

However, as seen in the previous chapters of this handout, the stock subassemblies that install with the AutoCAD® Civil 3D® program can be combined to solve a wide variety of modeling conditions. Intelligence can be added by incorporating conditional subassemblies at various locations in the assembly. Because of this, I have needed to only create a few custom subassemblies for grading models. Most of these have been fairly basic and were created to fill gaps in the functionality provided by the stock subassemblies.

The first is a simplified daylight subassembly. It has a couple of features that distinguish it from the stock daylighting subassemblies.

- It is simpler. It has only three user properties: Side, Cut Slope, and Fill Slope. This makes insertion and editing much easier because the user does not have to dig through dozens of settings to find the ones that they want to modify.



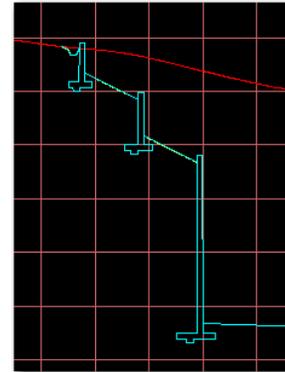
- In addition to being simpler, the cut and fill slope parameters are not limited to positive values. Entering negative slope values will allow the subassembly to chase a grade up slope when in a fill condition, or down when in cut. As shown in the image to the right, the daylight slope from the retaining wall is in a fill condition, but it is going up at 20:1 (5%). This works well as long as the target surface is steeper, and covers large enough of an area for a solution to be found.



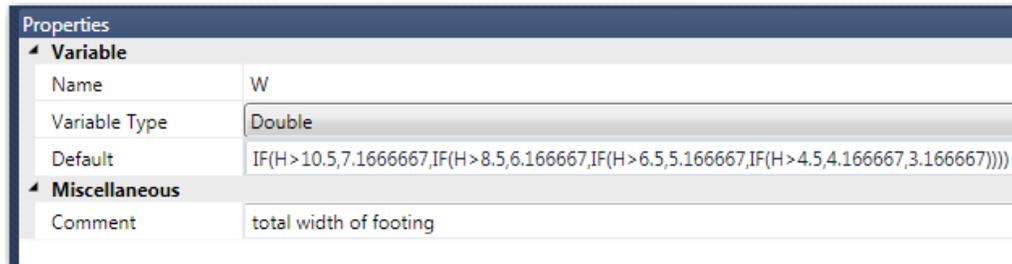
Another custom subassembly I created was for grading typical Caltrans retaining walls. This was done because the default retaining wall subassemblies, installed with the Civil 3D program, can only derive their height by targeting a surface - similar to the way a daylighting subassembly works. Because of this, they are difficult to use on the interior portions of the assembly, and are best suited for insertion on the ends.

Subassembly Composer was used to create a custom retaining wall subassembly with several elevation targets that allow its height and footing depth to be controlled with profiles.

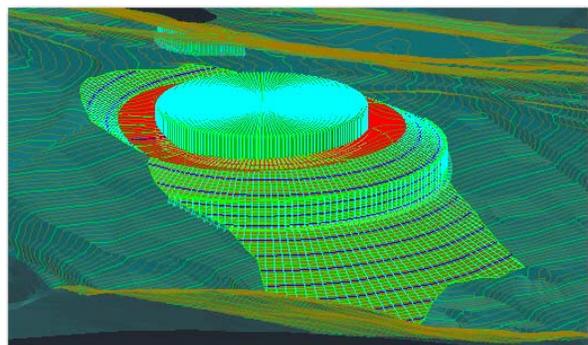
The image to the right shows the section view of three custom retaining wall subassemblies connected together in a single assembly. The heights of all three walls are being generated from target profiles (similar to the model in the previous chapter).



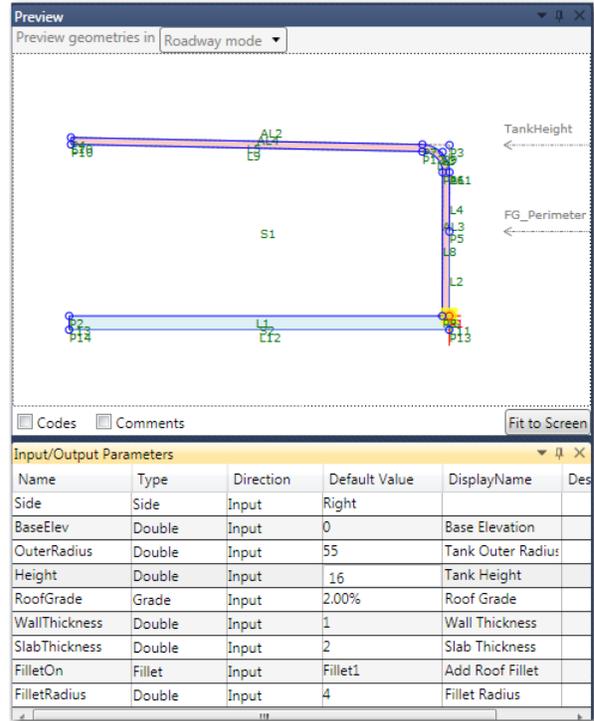
In addition to the elevation targets, the footing size and key dimensions get calculated based on the wall height. This was achieved by incorporating 'lookup' tables with If statements. Footing dimensions were created as variable definitions with expressions. A lookup format was achieved by nesting multiple IF functions, where the false expression evaluates the subsequent IF statement. The following image shows the expression used to return the footing width based on the height (H) of the wall.



An extreme example of a custom grading subassembly was a semi-buried water tank model. This subassembly addressed very specific modeling criteria. It was done mainly to test corridor grading capabilities, and has limited functionality for general site grading models.

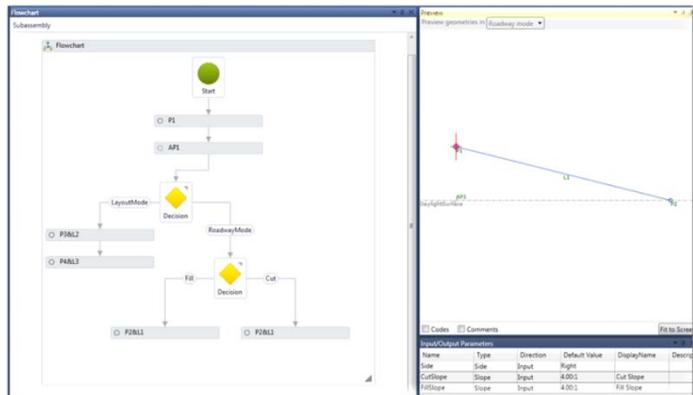


The key features of this subassembly include the parameter driven construction of the tank geometry; and the elevation targets that could be applied to control the grading around the perimeter of the tank and the overall height of the tank. These features allowed for quick iterations through different tank sizes, foundation depths, and to easily control the grading and drainage in the area adjacent to the tank.

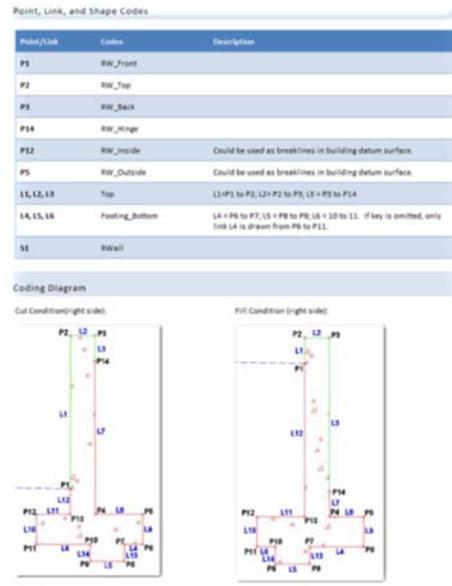


Tips and Tricks for Creating Custom Subassemblies

- Start simple. The adjacent image shows one of the first subassemblies that I created with the Subassembly Composer. It is a simple daylight subassembly that creates a single daylight link in either cut or fill (as described previously). Exercise 1, at the end of this section, contains all of the steps to build this subassembly.

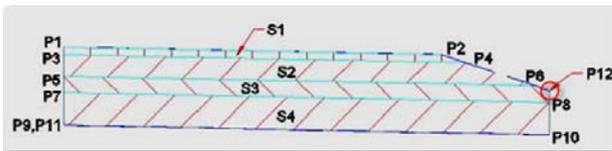


- Before building a subassembly, create a document that is similar to the help documents available with the stock subassemblies. Plan out how the subassembly will behave, list the input parameters and targets that will be used, and sketch out the geometry with point and link numbers.

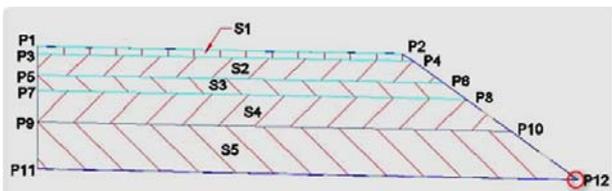


- Use the same code names that Civil 3D uses with the stock subassemblies. This will insure that your current code sets will work with any new subassembly that you create.
- If you have conditions that change the shape of the subassembly, make sure that you use the same point number for the outside point where other subassemblies will be attached.

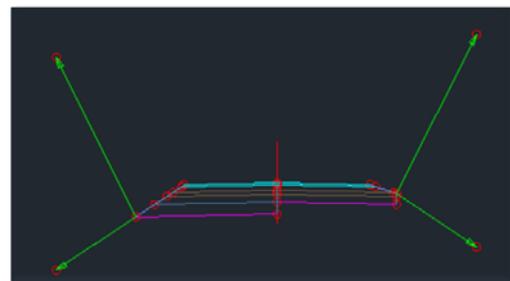
The following images show a highway subassembly, that has different geometry configurations for cut and fill. In the images you will see point number P12 assigned to different locations. It is at the ditch flowline point while in cut and the toe of the shoulder slope while in fill. When a daylight subassembly is attached at this point, it will shift to the correct position as the corridor alternates between cut and fill conditions.



Point codes in cut.

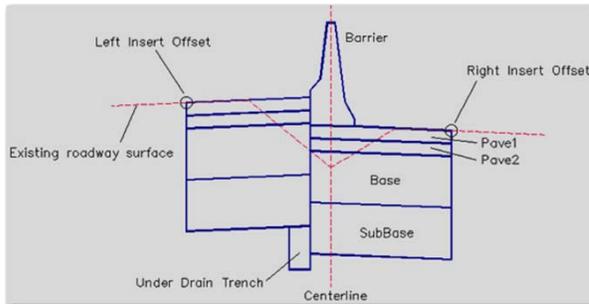


Point codes in fill.

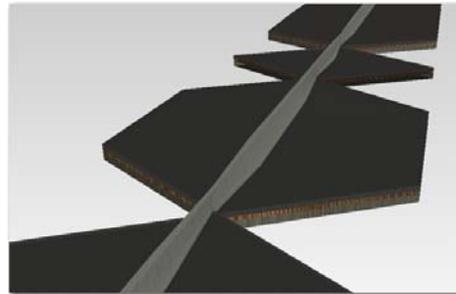


Subassembly in layout mode with daylight subassemblies attached.

- Don't use variables that will mirror asymmetric components. This following images show a median lane widening subassembly that was created from the stock OverlayMedianAsymmetrical subassembly. This subassembly calculates the point offset locations based on the right side of the highway being lower. If the left side is lower, it multiplies the offset values by negative one. When viewed in cross sections this subassembly will look fine. However, when feature lines or solids are extracted, they will crisscross as the low side condition alternates between left and right.



Subassembly layout based on right side being lower.

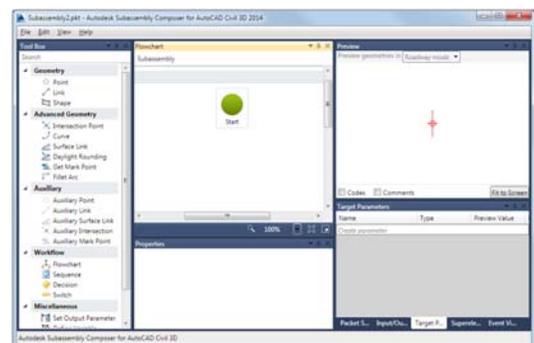


Resulting solids created from a "mirrored" subassembly with the Corridor Solids Extractor.

Exercise 1 – Creating a simple daylight subassembly

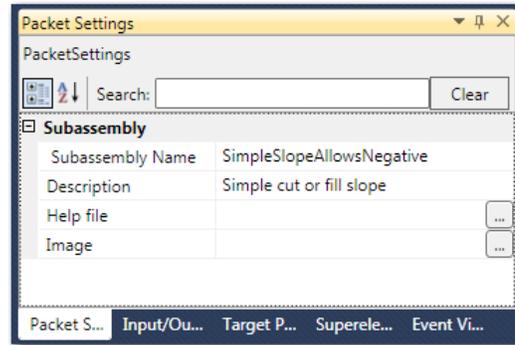
For those of you who have never used the Subassembly Composer program, this exercise will walk you through the steps to create a simple daylighting subassembly. Basic concepts of input parameters, targets, and decisions will be demonstrated.

1. Launch the Subassembly Composer program with a new, blank subassembly.

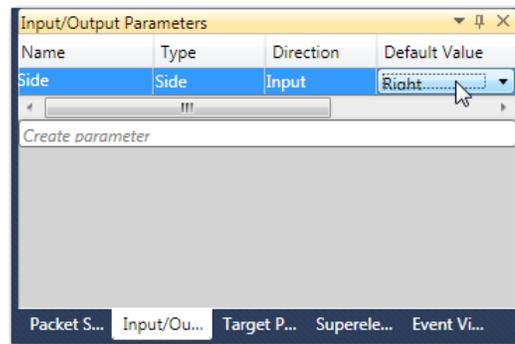


2. In the lower right panel, select the Packet Settings tab.
3. Enter the name **SimpleSlopeAllowsNegative**.
4. Enter a description. This will appear in the subassembly's tool tip when you hover over it in the Tool Palette window.

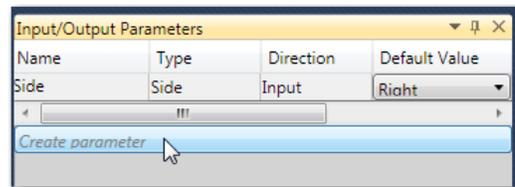
If you have created a help document or thumbnail image, you can add them on this tab.



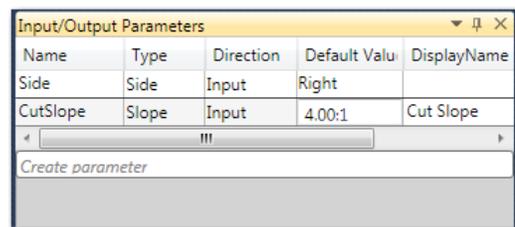
5. Select the **Input/Output Parameters** tab.
6. Set the default value for the side parameter to **Right**. All subassemblies will have a Side parameter. If the subassembly is not side specific, set the default to None.



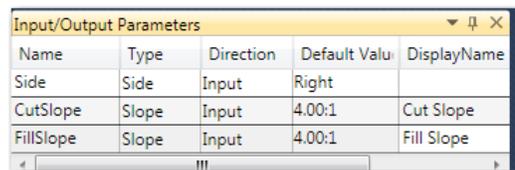
7. Pick the **Create parameter** button, located at the bottom of the list.



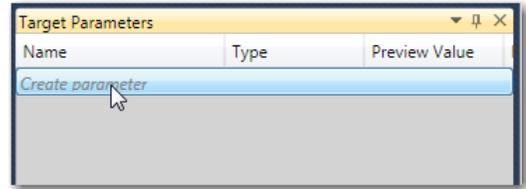
8. Enter the name **CutSlope**. This is the property name that will be referred to in the Subassembly Composer.
9. Set the type to **Slope**.
10. Set the direction to **Input**.
11. Enter a default value of **4:1**.
12. Enter a display name of **Cut Slope**. This is the name that will be displayed in the Properties command in Civil 3D.



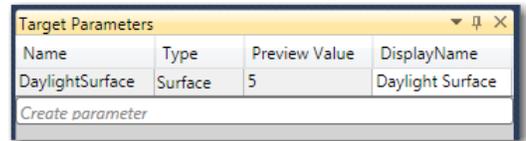
13. Repeat steps 7-12 to add a **FillSlope** parameter.



14. Select the **Target Parameters** tab.
15. Pick the **Create parameter** button.



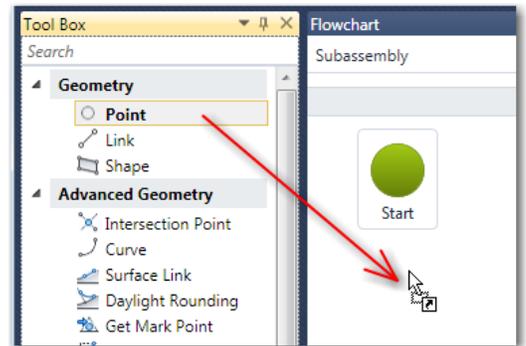
16. Enter the name **DaylightSurface**. This is the name that will be referred to in Subassembly Composer.
17. Set the type to **Surface**.
18. Enter a non-zero preview value. This will affect the display in the Preview panel.
19. Enter the display name **Daylight Surface**.



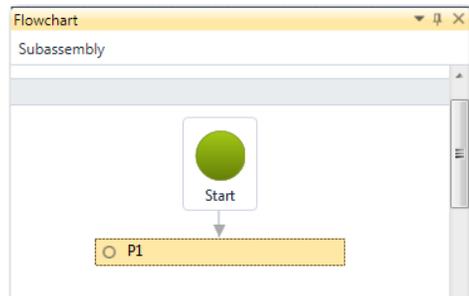
We are done setting up the input and target parameters. The remainder of this exercise will utilize the Tool Box, Flowchart and Properties panels. As we progress, we will check the results in the Preview panel.

This assembly will have a couple of decision points added to its flowchart. It is best to start the subassembly by adding any points, links, or variables that will be utilized in all of the decision branches. For this subassembly, the only common point will be the initial start point.

20. Pick and drag the **Point** object from the Tool Box window into the Flowchart window.



21. When you release your mouse button a point will be added to the flowchart. A flowchart link will automatically be added from the Start element. The point will be assigned the next available point number.



22. Select the P1 point in the flowchart. Its properties will be displayed in the Properties panel.

23. Enter the point code “**Hinge**”. To be interpreted as a text string, instead of a parameter or variable, the code must be within quotation marks. Multiple codes can be entered by separating them with commas. Each code string would be enclosed within its own set of quotation marks.

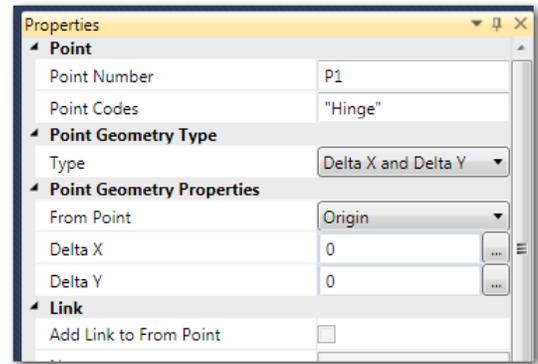
24. Set the type to **Delta X and Delta Y**.

25. Set the from point option to **Origin**. This will reference the subassembly’s insertion point.

26. Enter a delta X value of **0**.

27. Enter a delta Y value of **0**.

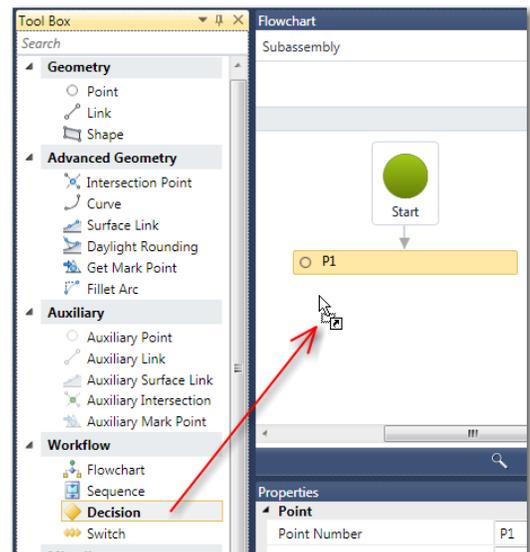
28. Leave the Add link to From Point property toggled **off**.



The next few steps will add a decision point to determine if the subassembly is being used in assembly layout (layout mode) or in a corridor (roadway mode). If it is in layout mode, we are going to have it draw both cut and fill slopes, just like the daylight subassemblies that installed with Civil 3D.

You do not need to create a layout version of the subassembly. Civil 3D will use the roadway mode of the subassembly in layout view. Based on the default parameter settings it will display either the cut or the fill portion of the subassembly.

29. Pick and drag the **Decision** object from the Tool Box window into the Flowchart window.



30. With the Decision element selected, edit its properties in the Properties panel.

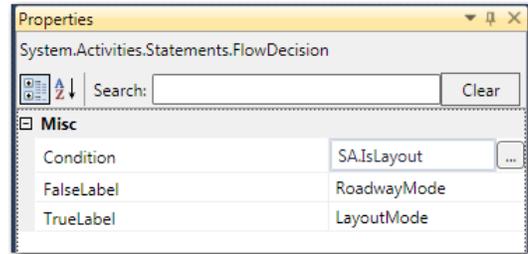
31. Enter the following expression in the condition field: **SA.IsLayout**

This expression is a .NET function of the subassembly (SA) object. It will evaluate to True if the subassembly is being displayed on an assembly layout.

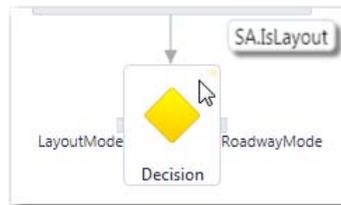
32. Enter a false label value of **RoadwayMode**.

33. Enter a true label value of **LayoutMode**.

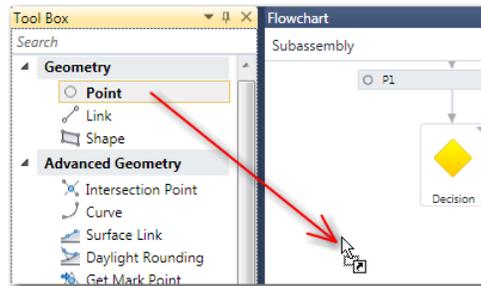
The two label values will be displayed in the flowchart, making it easier to read.



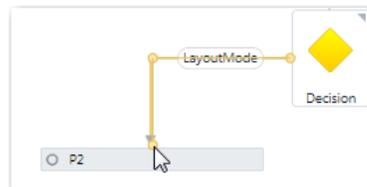
When you hover your cursor on the decision element in the flowchart panel, a tool tip containing the expression will appear, and the true and false connection tabs will display.



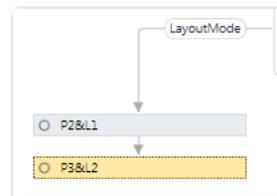
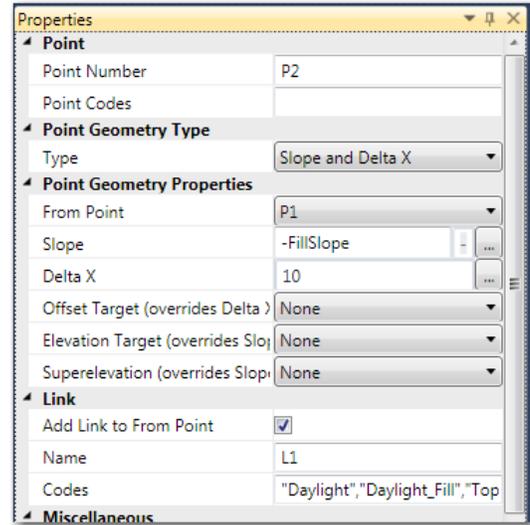
34. Pick and drag the **Point** object from the Tool Box window into the Flowchart window.



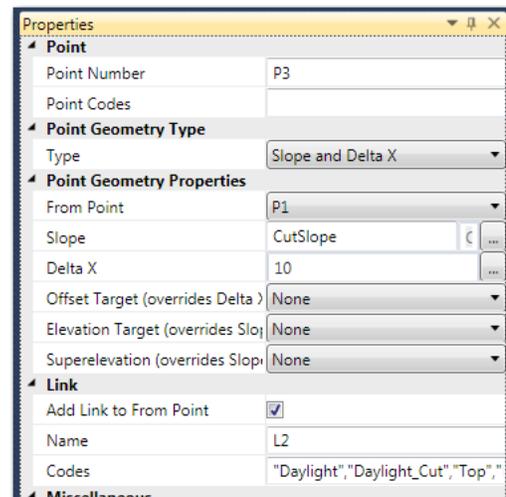
35. Grip-edit the flowchart connector that comes out of the LayoutMode side of the decision. Hook it to the tab on the top of the point that was just added.



36. Select the P2 point element in the flowchart.
In the Properties panel:
37. Set the type to **Slope and Delta X**.
38. Set the from point to **P1**.
39. Enter a slope value of **-FillSlope**. This will apply the fill slope input parameter in the negative Y direction.
40. Enter a delta X value of **10**. This will be the default width displayed during assembly layout in Civil 3D.
41. Leave all of the target options set to **None**.
42. Toggle **on** the Add Link to From Point option.
43. Enter the codes **"Daylight", "Daylight_Fill", "Top", "Datum"**
44. Drag-and-drop another **Point** element. Place it below the previous one.

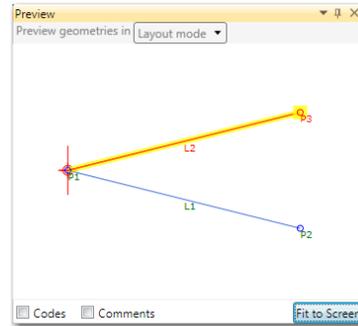


- In the Properties panel:
45. Set the type to **Slope and Delta X**.
 46. Set the from point to **P1**.
 47. Enter a slope value of **CutSlope**. This will apply the cut slope input parameter in the Y direction.
 48. Enter a delta X value of **10**.
 49. Leave all of the target options set to **None**.
 50. Toggle **on** the Add Link to From Point option.
 51. Enter the codes **"Daylight", "Daylight_Cut", "Top", "Datum"**



52. To verify the model, set the Preview panel to display geometries in **Layout mode**. This is set at the top of the panel.
53. Click the **Fit to Screen** button.

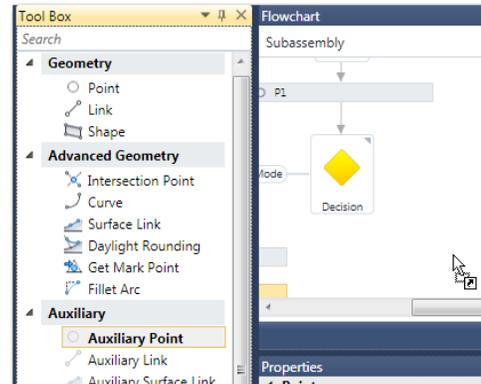
Items selected in the flowchart will be highlighted in the preview panel.



We are now ready to create the elements that will define how this subassembly will be constructed in the corridor model. We will start by adding an auxiliary point that gets its Y value from the target surface. This point will be used in a decision element to determine if the subassembly is in cut or fill.

Auxiliary elements are used internally by the subassembly to calculate or locate other point and link elements. They will not appear to the user within the Civil 3D program.

54. Drag-and-drop an **Auxiliary Point** element to the flowchart window. Place it below and to the right of the ISLayout decision.



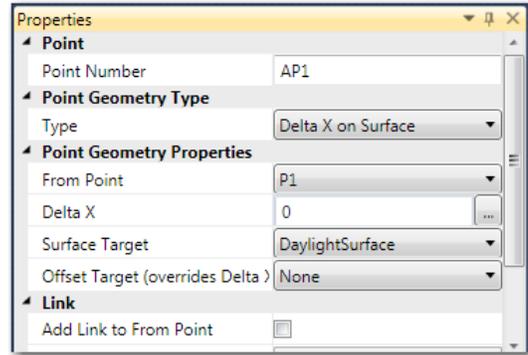
It will automatically be linked to the last point element that was added to the flowchart.

55. Grip-edit the link line so that it connects to the RoadwayMode tab of the decision element and points to the top tab of the auxiliary point.



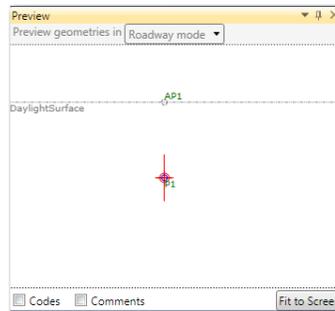
In the Properties panel:

56. Set the type to **Delta X on Surface**.
57. Set the from point to **P1**.
58. Enter a delta X value of **0**. This will sample the target surface directly above or below the subassembly's origin point.
59. Select the **DaylightSurface** target parameter.
60. Leave the offset target set to **None**.
61. Leave the "add link to from point" option toggled **off**.

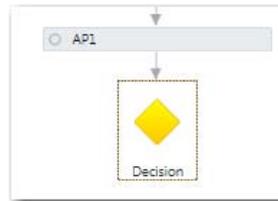


62. Change the Preview panel to Roadway mode.

You should see the AP1 point aligned with the P1 point, and at the same elevation as the DaylightSurface target.

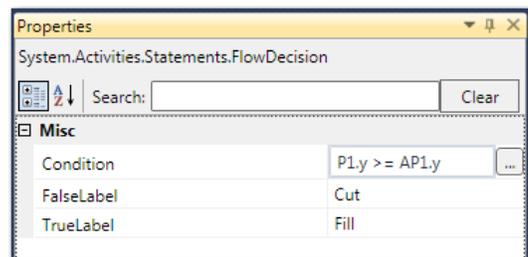


63. Drag-and-drop another **Decision** element to the flowchart window. Place it below the auxiliary point. The correct flowchart link should be added automatically.

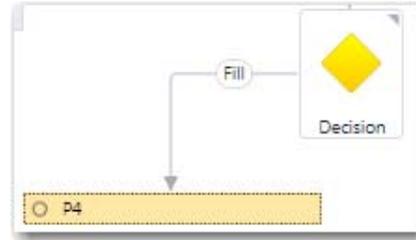


In the Properties panel:

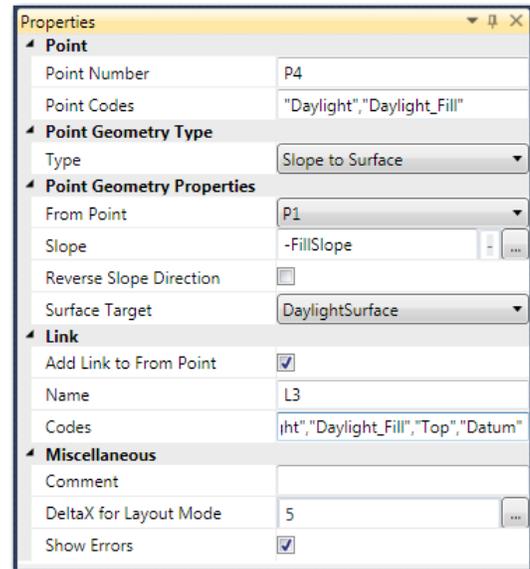
64. Enter the following expression in the condition field: **P1.y >= AP1.y**
It will evaluate to true if the Y coordinate from point P1 is greater than or equal to the Y coordinate from the auxiliary point AP1.
65. Enter a false label of **Cut**. (P1 is lower than AP1)
66. Enter a true label of **Fill**. (P1 is higher than AP1)



67. Drag-and-drop another **Point** element to the flowchart window. Place it below and to the left of the last decision element.
68. Drag the fill flowchart link to the top of the new point element.



- In the Properties panel:
69. Enter the point codes:
"Daylight", "Daylight_Fill"
 70. Set the type to **Slope to Surface**.
 71. Set the from point to **P1**.
 72. Enter the slope value **-FillSlope**. This will apply the FillSlope input parameter in the negative Y direction.
 73. Set the surface target to the **DaylightSurface** parameter.
 74. Toggle **on** the add link to from point option.
 75. Enter the link codes:
"Daylight", "Daylight_Fill", "Top", "Datum"

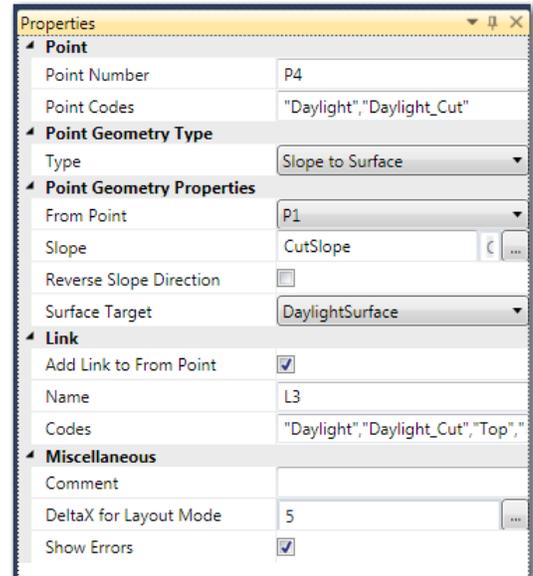


76. Drag-and-drop another **Point** element to the flowchart window. Place it below and to the right of the last decision element.
77. Drag the flowchart link from the Cut tab of the decision to the top of the new point element.

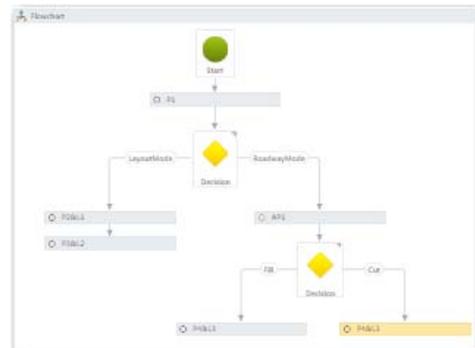


In the Properties panel:

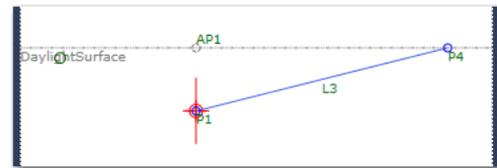
78. Change the point number to **P4**. This way the subassembly will have the same code number in both cut and fill.
79. Enter the point codes: **"Daylight","Daylight_Cut"**
80. Set the type to **Slope to Surface**.
81. Set the from point to **P1**.
82. Enter the slope value **CutSlope**. This will apply the CutSlope input parameter in the Y direction.
83. Set the surface target to the **DaylightSurface** parameter.
84. Toggle **on** the add link to from point option.
85. Change the link name to **L3**. This way the subassembly will have the same link name in cut and fill.
86. Enter the link codes: **"Daylight","Daylight_Cut","Top","Datum"**



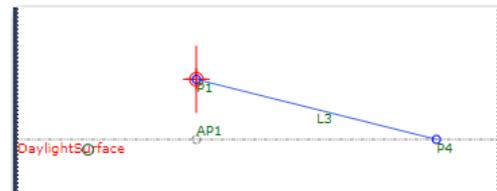
The finished flowchart should appear similar to the image to the right.



87. In the Preview panel, drag the DaylightSurface target up and down to make sure that the P4 point and L3 link are generated in both cut and fill.



Cut



Fill

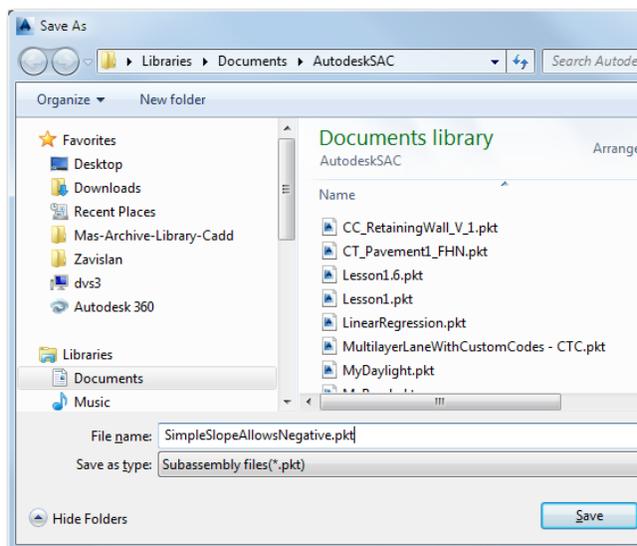
88. Select the **File > Save** menu command.

89. Enter the name
SimpleSlopeAllowsNegative.pkt

This is the same name that we entered on the packet settings tab.

The default file location is in the user's Documents\AutodeskSAC folder. You will probably want to copy the packet file to a network location so that it can be installed on all users' workstations.

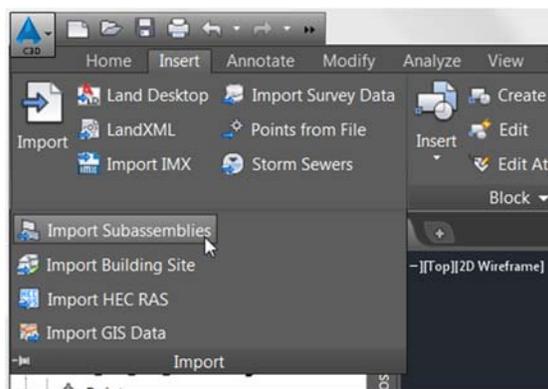
90. Click the **Save** button.



Exercise 2 - Importing a Custom Subassembly into Civil 3D

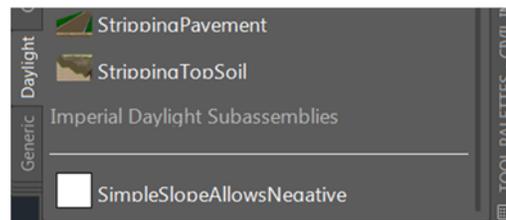
This exercise shows how to import a custom subassembly into the Civil 3D program. This process will need to be performed for each user in the office that will be working with corridor models using this subassembly.

1. The **Import Subassemblies** command is located in the expanded **Import** ribbon panel of the **Insert** ribbon tab.

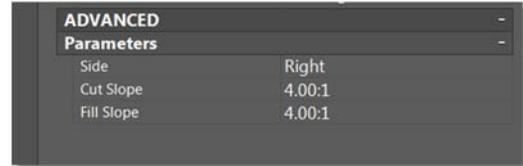


After importing, the new subassembly should appear at the bottom of the palette that you selected during the import process.

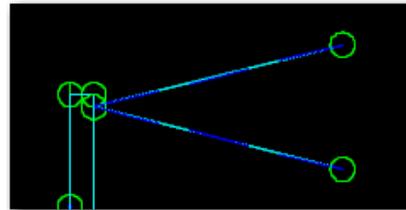
You can add separators and text to better organize the palette.



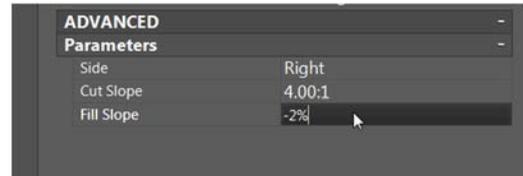
- Test the subassembly out by adding it to an assembly.
The Properties window should display the three input parameters that were defined.



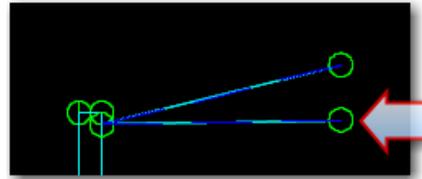
- Attached to the assembly, it should display both cut and fill links as defined in the layout mode.



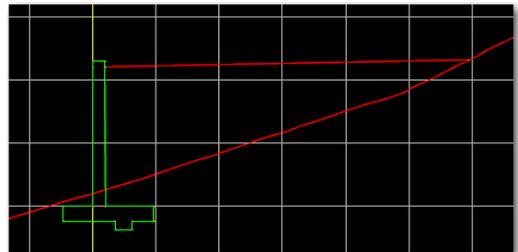
- Try editing the subassembly's properties - changing the Fill slope to -2%.



- The fill link should now appear slightly above horizontal.



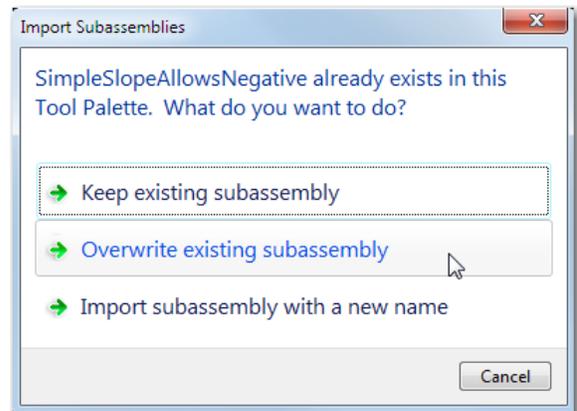
- Assign the assembly to a corridor model.
- Edit the models target mapping and make sure that there is a DaylightSurface target parameter.
- Assign the existing ground surface to that target.
- Edit the corridor sections to verify that the subassembly is behaving as expected.



Exercise 3 – Re-Importing an Updated Subassembly

This exercise shows how to re-import an existing subassembly. This usually needs to be done several times as you debug and update your custom subassemblies. Since this is a somewhat cumbersome process, it is recommended that you do your debugging in one drawing file, with a single assembly, and a simple corridor.

1. Close all sessions of Civil 3D.
2. Open a new session of Civil 3D with an empty drawing. This is done because you cannot re-import a subassembly if it is being used in any of the currently opened drawing files.
3. Select the **Import Subassemblies** command. It is located in the expanded **Import** panel of the **Insert** ribbon tab.
4. Browse to and select your subassembly pkt file.
5. Click the **Open** button.
6. Toggle **on** the import to Tool Palette option.
7. Select the palette that you want to add the subassembly to.
8. Click the **OK** button.
9. Select the **Overwrite existing subassembly** option.



The subassembly should appear again at the bottom of the palette.

10. Open the drawing containing your corridor model.
11. Replace any instances of your custom subassembly by inserting it from the tool palette.
12. Edit the corridor properties and rest any targets.