AUTODESK UNIVERSITY

TR17997-L

Introduction to the Subassembly Composer

Luke Perkins
Stewart – Raleigh NC

Learning Objectives

- Creating a Plan of Attack
- Intro to the Subassembly Composer
- Building our Subassembly – Points, Links, and Shapes
- Importing into Civil 3D
- Beyond the Basics

Description

Welcome to Subassembly Composer for AutoCAD Civil 3D software! This is a lab for those that have utilized AutoCAD Civil 3D software for some time, but are interested in figuring out how to take their corridors to a whole new level. We will develop subassemblies you can put to work as soon as you're home, and give you a methodology for creation that will make your co-workers think you're a professional software developer. Specifically, we will begin by diving into good subassembly design practices, taken directly from our friends in the software development industry. The subassemblies we'll build together in this lab are particularly relevant to those in the land development industry, but can also add value to the steps necessary to create subassemblies for any area of focus in our industry.

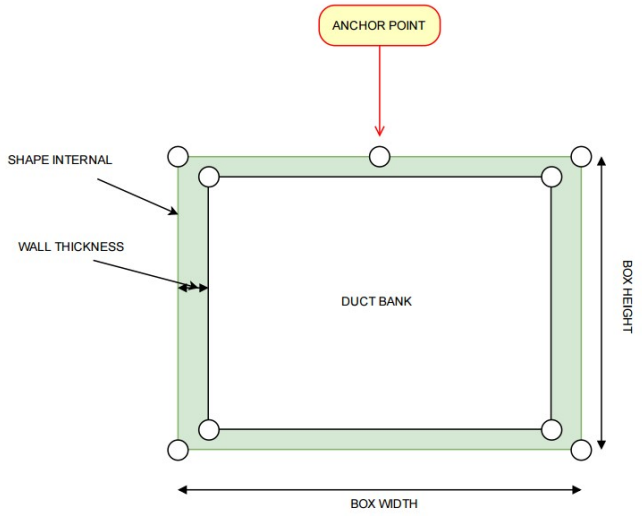
Your AU Expert

Luke Perkins is an Engineering Analyst for Stewart in Raleigh, NC. After graduating North Carolina State University in 2014 he has spent the last two and a half years as a civil engineering designer in the land development industry. During his time at Stewart, he has focused on helping the Land Planning and Design group fully utilize Civil 3D, Infraworks, and Storm and Sanitary analysis for grading, stormwater, sanitary and overall site design. Luke has a passion for technology and has been a part of distinguished design teams on notable projects in North Carolina that are changing the face of Raleigh in the midst of surging growth.

Creating a Plan of Attack

To maximize the effectiveness of the subassembly composer, it is essential to create a schematic of our subassembly on paper; this plan of attack will help organize our thoughts and make us more efficient with our time while constructing the subassembly. Crucial questions to answer when mapping out a potential subassembly:

1. Where do I want my anchor point to be?
2. What parts of my subassembly will users be able to change?
3. What codes should be incorporated to guarantee convention and facilitate surface creation?
4. How can I simplify / Are there any Autodesk PKT's that I can pull from?



Once we have answered these questions and have a working sketch of what our subassembly should look like, create an electronic version of the sketch for documentation and dissemination to fellow designers. This should include a help file that describes in detail the various features of your subassembly that users we'll need to be familiar with

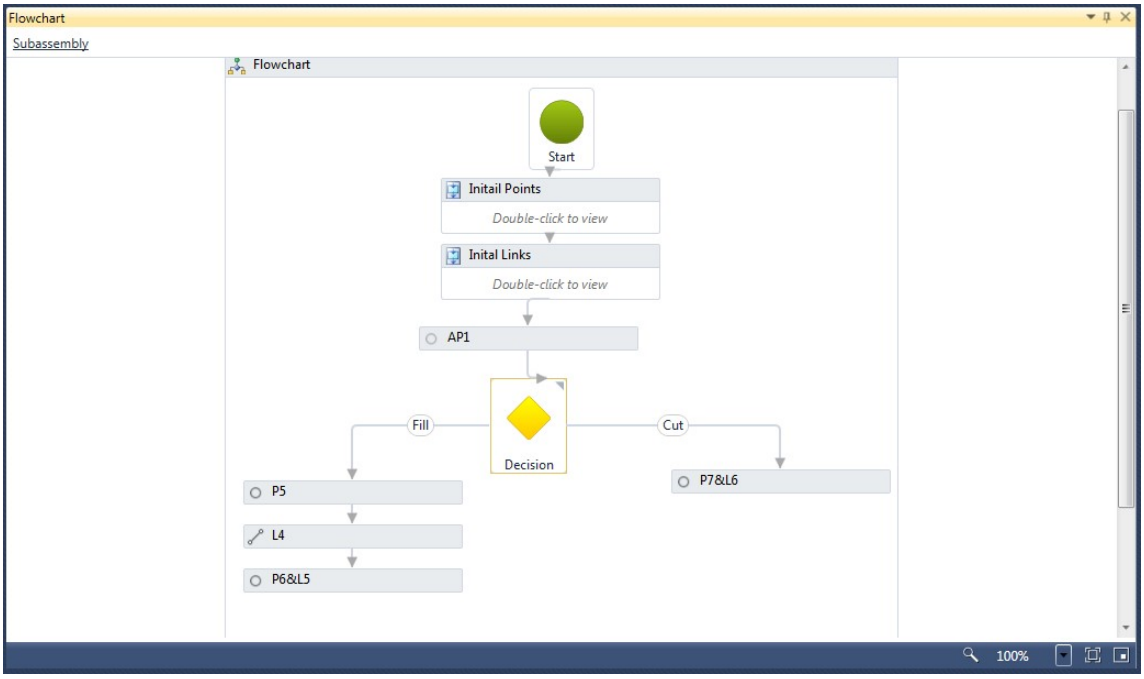
Introduction to the Subassembly Composer

Now that we have an idea of the subassembly we want to create, it's time to implement our design through the Composer. The Subassembly Composer is a simple user interface to help facilitate the creation of subassemblies without have to manually code in Visual Basic.

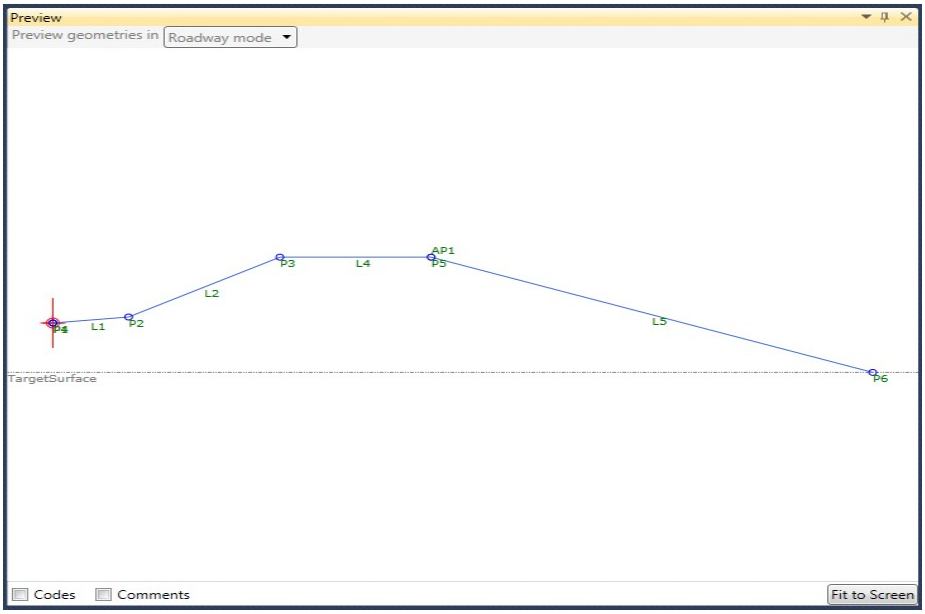
The User Interface

Subassembly Composer's UI is broken into these five major pieces:

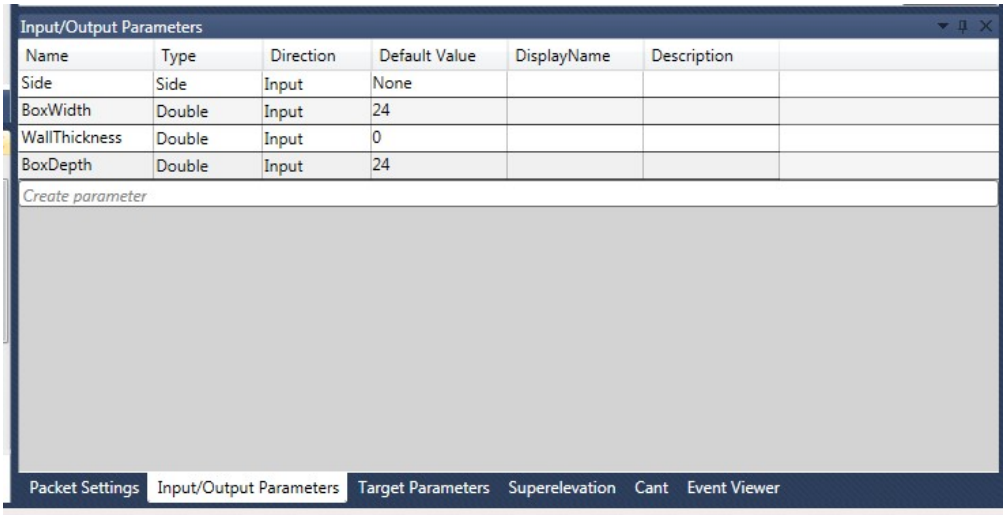
1. The Flow Chart – A visual representation of our subassembly components and our logic used throughout creation.



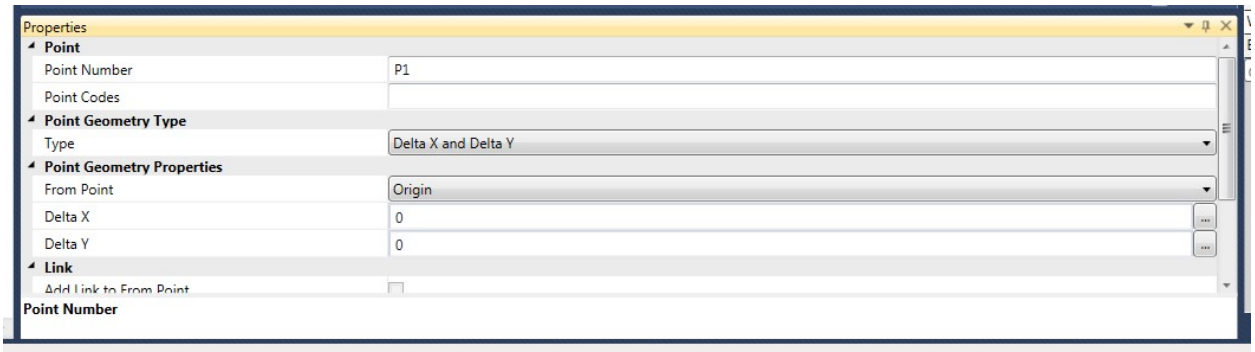
2. The Preview Pane - An illustration of the subassembly we are creating. This preview closely represents what will be depicted in Civil 3D upon import



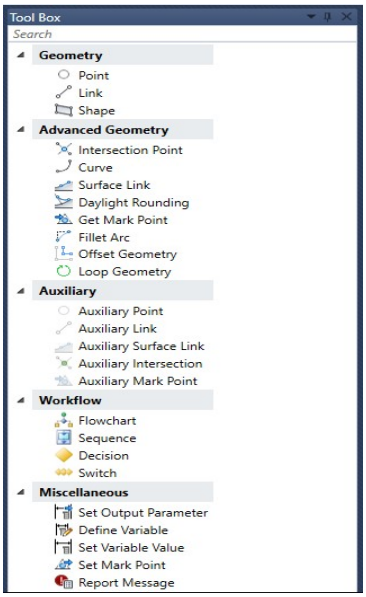
3. Input Pane – This area of the UI (lower right default) is used to build the components of our subassembly and has various options of input depending what is being created.



4. Properties – Here we have the ability to change the properties of components added from the Tool Box to the flow chart



5. Tool Box – A list of the building blocks we have at our disposal to create the subassembly.



Building Our Subassembly

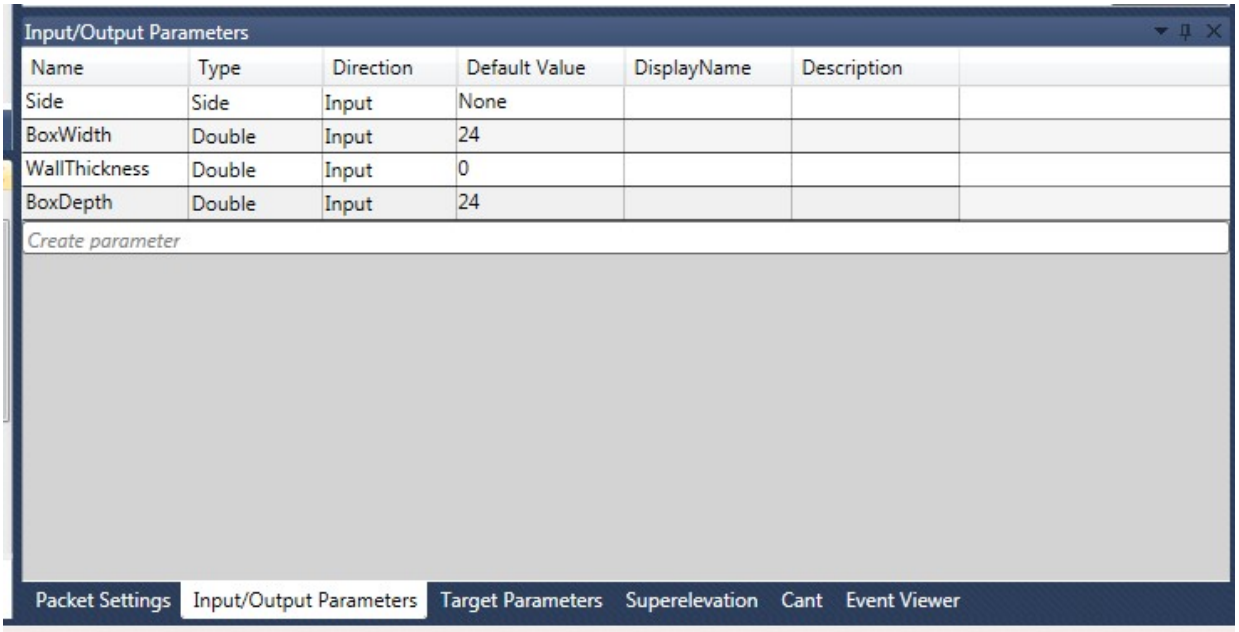
In the tool box on the left, you'll see there are many tools to create our subassembly, but everything that creates the actual build of your subassembly can be simplified down to points, links, and shapes. We'll take our duct bank from the plan of attack sketch a bit further by adding these points, links and shapes through the flow diagram.

Defining Properties

In Packet Settings we can define the name of our subassembly, give our users a description, attach a help file giving the users directions on how to implement our subassembly, and attach an image that will act as a preview to the assembly imported into the Civil 3D tool palette.

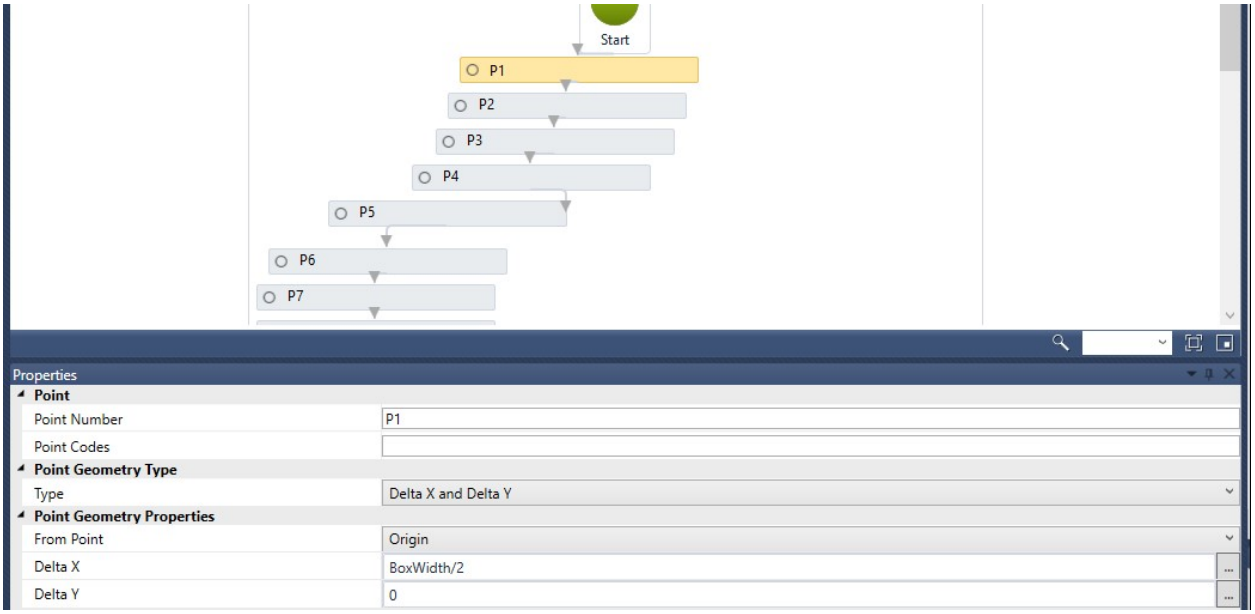
Adding Input Parameters

Input parameters control the user's ability to change components of the subassembly. For our subassembly we'll set up three input parameters. An OuterWidth, an OuterHeight, and a WallThickness (Default units in inches)



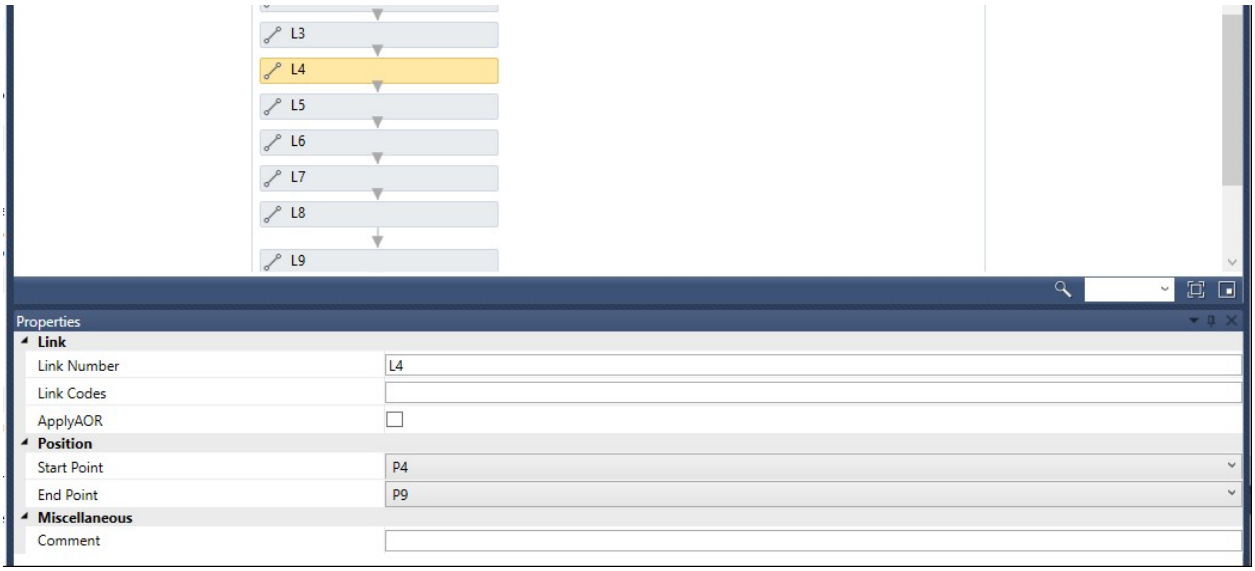
Points

To add a point to the flowchart, drag and drop from the tool palette. With the point selected, you are given a dialog to change the properties of the point. This will influence the geometry of our subassembly by changing the point's location in reference to another point or the origin. We can also use our input parameters to influence this geometry.



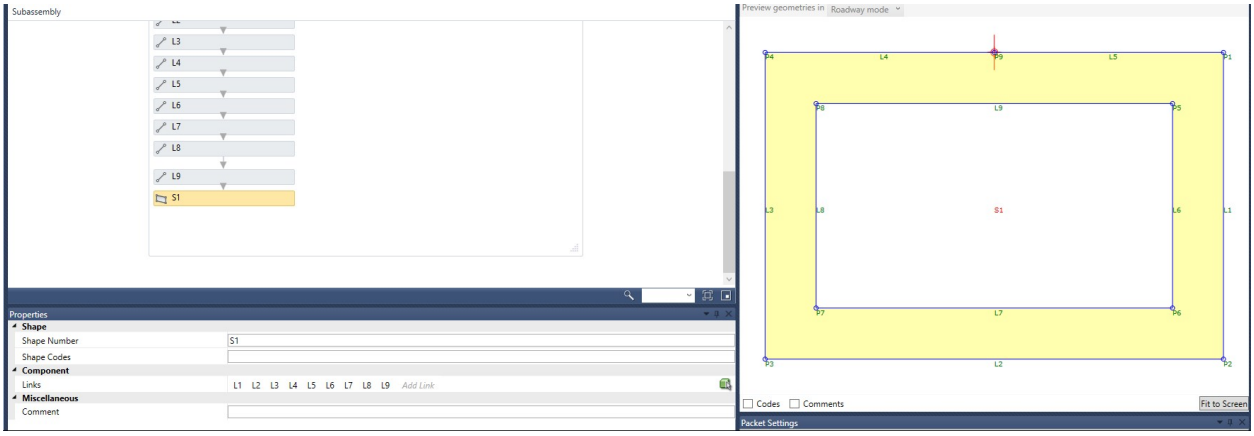
Links

You might notice there is an option to add links between points automatically when building the points. Sometimes this is an easy method of creating links, but in our case we have a situation where we need to create some links manually. Simply drag a link into the flow chart, and select the Start Point and the End Point.



Shapes

To add a shape, drag and drop shape from the Toolbox into the flowchart. The simplest way to create the shape is by selecting the “Select shape in Preview” glyph, and then using your cursor in the preview pane to select the void space created by an enclosed set of links.



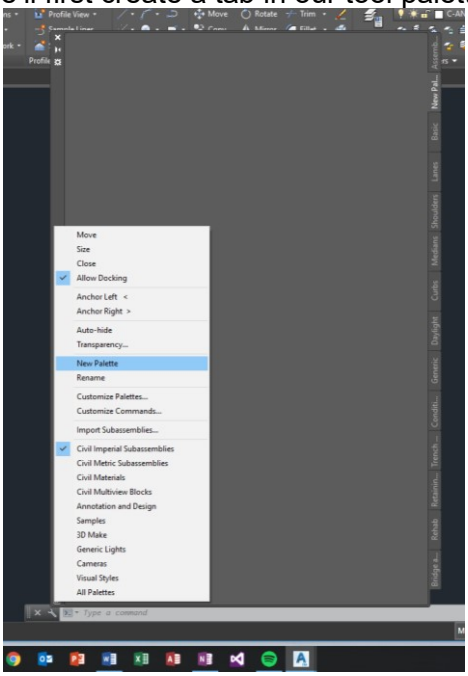
Importing into Civil 3D

Export from Subassembly Composer

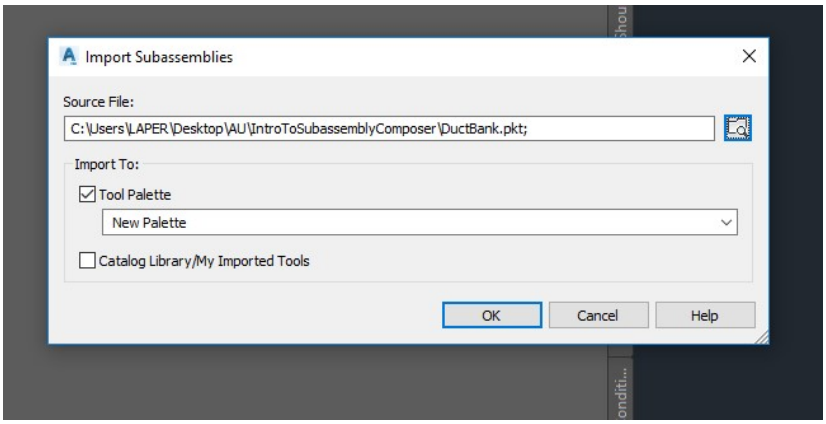
Save the work we've completed in Subassembly Composer as a PKT file to a smart location.

Import into Civil 3D

To import our subassembly, we'll first create a tab in our tool palette:



Next we'll right click the tool palette pane and import a custom subassembly:



Codes

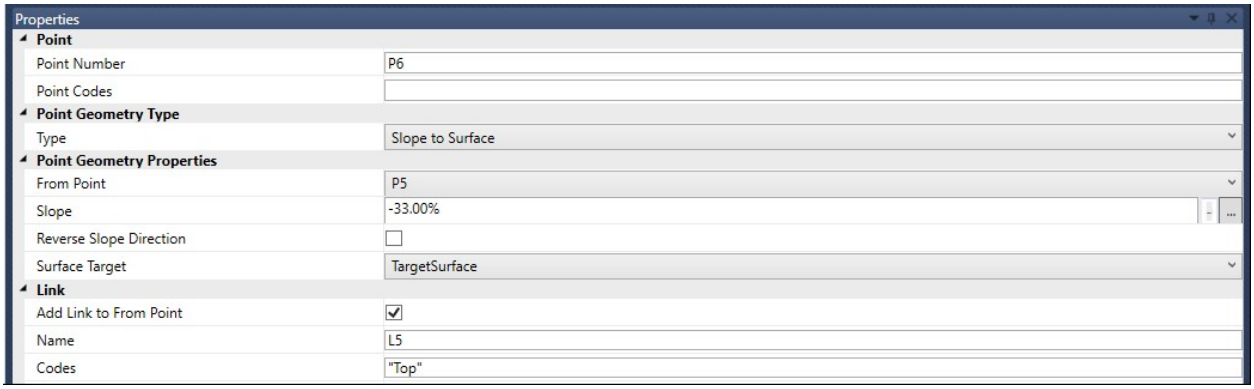
Codes tell civil 3D how to consume the data from the Subassembly Composer. A familiar code "Top" can be added to links that are designed for surface creation to generate our corridor surface in Civil 3D

Beyond the Basics

We've created a subassembly and imported it into Civil 3D successfully, but there's so much more the Subassembly composer has to offer. From complex geometry to decisions, switches and targeting, next we will look at examples of building a subassembly of moderate difficulty.

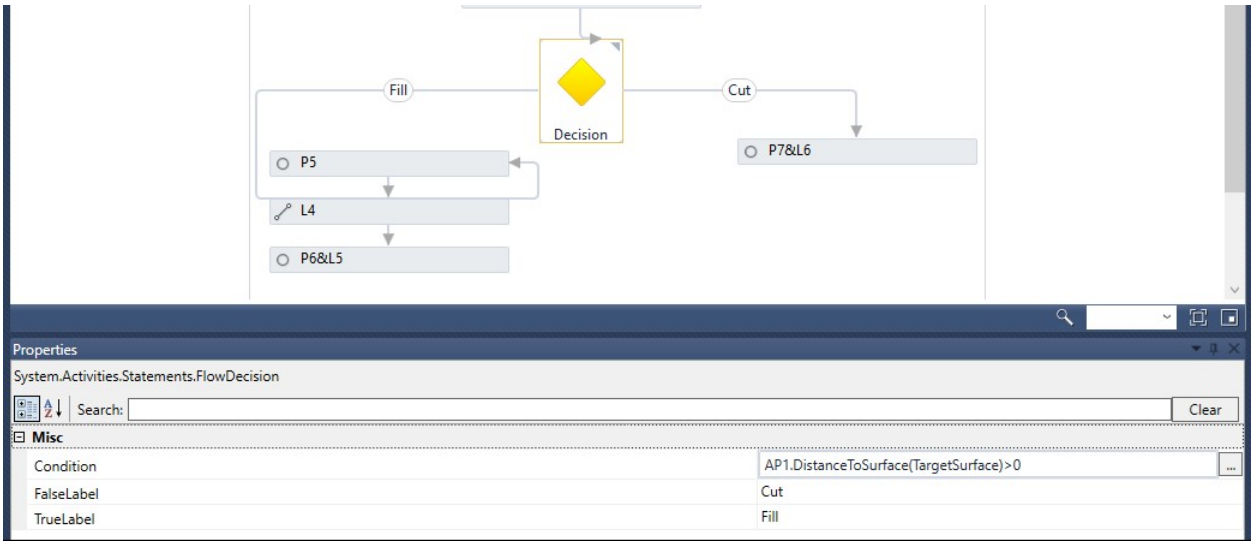
Targeting

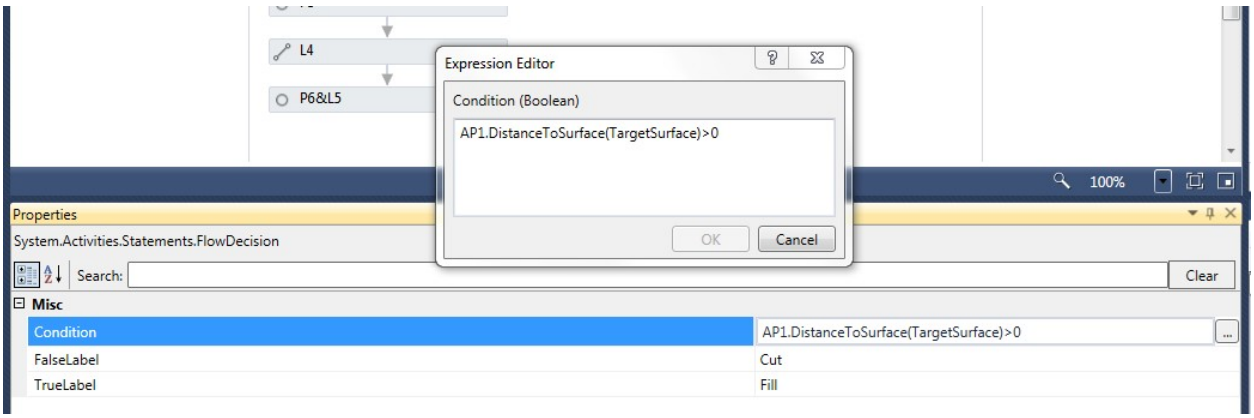
Very similarly to adding input parameters, targeting is achieved by adding an elevation, offset, or surface targeting parameter. It is important to note that without a decision we are limited to one-dimensional surface targeting.



The Decision

Decisions are used to represent an If/Then statement in SAC. The most popular use of the decision is to figure out if our subassembly should tie to a surface in either cut or fill. To create this decision we'll drag and drop a decision into the flow chart, and with a little VBA code, create a simple statement that will analyze whether our point is above or below the input surface. This will split our flow chart and allow us to continue building on either side of the decision.





Summary

The subassembly composer has many other features. Some that will require practice and further knowledge to master the different logic possibilities of the flow chart as well as taking points, links and shapes further to create complex geometry. However, with these few concepts we've covered in this course subassemblies can be created that far exceed the abilities of pre-packaged subassemblies that ship with Civil 3D. With a little imagination, collaboration and creativity the possibilities are endless!

Additional Resources

Using codes in SAC for accurate quantity take-offs:
http://au.autodesk.com/au-online/classes-on-demand/class-catalog/2015/autocad-civil-3d/ci10120#chapter=0

Complex Geometry:
http://au.autodesk.com/au-online/classes-on-demand/class-catalog/2015/autocad-civil-3d/ci9963-l

Aviation:
http://au.autodesk.com/au-online/classes-on-demand/class-catalog/2015/autocad-civil-3d/ci10209#chapter=0