BIM for Civil Engineers: Intelligent Hydraulics and Hydrology with Autodesk AutoCAD Civil 3D

W. Curtis Smith, P.E., LEED AP – Gresham Smith and Partners (Nashville, TN)

CI2170

Autodesk AutoCAD Civil 3D software gives us many great features out-of-the-box for our day-to-day hydraulics and hydrology (H&H) modeling needs, but it's not until you dig deeper that you find it can truly bring out the "I" in BIM for civil engineers. Stop limiting your H&H workflow inside of the Autodesk® AutoCAD® software environment to simple area takeoffs. Don't limit the pipe network functionality to what comes out-of-the-box. Learn to use Civil 3D pipe networks, parcels, and catchment objects to greatly improve H&H productivity and drainage map plan production. Empower your team with the tools needed for advanced H&H workflows in Civil 3D as the data is prepared for seamless exporting into the Hydraflow Storm Sewers Extension for Civil 3D or for Autodesk® Storm and Sanitary Analysis software. This class focuses on setting up your drawing templates and shared part catalogs for enhanced H&H modeling. We do discuss workflows, but this class focuses mainly on setting up the tools that are needed to perform the workflows.

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

- Create dynamic labels and expressions for storm systems and hydrology catchment areas
- Customize the part catalog shared parameters for enhanced information modeling and data entry
- Customize parcel styles and user-defined properties for hydrology information modeling
- Prepare standard or specialized drawing templates for a fully integrated hydrology and hydraulics workflow
About the Speaker

*Curtis is registered professional engineer and BIM for civil engineering specialist at Gresham Smith and Partners in Nashville, TN. He was brought into the 500+ employee A&E firm in early 2011 primarily as a civil engineer, but also in hopes of helping the Land Planning division successfully implement a BIM solution. Almost immediately he push to move away from the old CAD platform into the Autodesk platform, particularly into AutoCAD Civil 3D, and began developing from scratch the internal infrastructure and standards needed for successful implementation. For this he was given the company’s annual award for leadership in technology. Curtis brings over 6 years of Civil 3D experience and Civil 3D 2013 Certified Professional status. Along with his day-to-day project engineering duties, he is responsible for the development and maintenance of drawing templates, workflow standards, training, and support for over 50 Civil 3D users in 7 offices across the US.

Email: Curtis_Smith@gspnet.com

Acronyms and Abbreviations

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>BIM</td>
<td>Building Information Modeling (We need to make this ‘CIM’ for Civil Engineers)</td>
</tr>
<tr>
<td>C3D</td>
<td>AutoCAD Civil 3D</td>
</tr>
<tr>
<td>OOTB</td>
<td>Out-of-the-Box</td>
</tr>
<tr>
<td>HSS</td>
<td>Hydraflow Storm Sewers</td>
</tr>
<tr>
<td>SSA</td>
<td>Autodesk Storm and Sanitary Analysis</td>
</tr>
</tbody>
</table>

This handout is very lengthy and detailed. DO NOT try to read and understand it all before the AU session! DO use it as a reference guide in the future.

We will not have time to cover all of this material in detail during the 90 minute lecture. We will try to work through as many examples of these workflows and topics as possible during our short time together at AU. Hopefully this document will sever its purpose as a reference guide to assist you in your future endeavors into these topics.

*Disclaimer: No manual text overrides were used in the creation of these screen captures. What you see are pure BIM for Civil information labels and tables reporting from the model!
Task 1: Create Dynamic Labels and Expressions for Storm Systems and Hydrology Catchment Areas

Going beyond Out-of-the-Box with Labels
Civil 3D labels are the glue that connects the old-school drawings world and the new-school BIM world. We put all the effort into designing a civil infrastructure model in Civil 3D, but at the end of the day we still have to annotate a plan to build from. Civil 3D labels have made this easy for us since the inception of the software. Sure, there’s much we can annotate with the OOTB labels, but with just a little knowledge of engineering industry standard formulas, some basic mathematical equations, and understanding of computer logic we can do so much more.

Label expressions for Civil 3D objects can be found in the Settings tab of Toolspace. Since this is all about H&H workflows, we’re going to focus on pipe network and parcel area labels. Expressions must be written for each unique type of Civil 3D object.

Civil 3D does not come with any predefined label expressions OOTB, but it does give you several object variables, operators, and mathematical functions that you can use to write your own expressions. The Help files are the best place to go to find information on what each operator and function does.
### Operators

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Operator Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>logical equals</td>
</tr>
<tr>
<td>-</td>
<td>logical not equals</td>
</tr>
<tr>
<td>!</td>
<td>logical not</td>
</tr>
<tr>
<td>&lt;</td>
<td>logical less-than</td>
</tr>
<tr>
<td>&lt;=</td>
<td>logical less-than-or-equals</td>
</tr>
<tr>
<td>&gt;</td>
<td>logical greater-than</td>
</tr>
<tr>
<td>&gt;=</td>
<td>logical greater-than-or-equals</td>
</tr>
<tr>
<td>AND</td>
<td>logical AND</td>
</tr>
<tr>
<td>OR</td>
<td>logical OR</td>
</tr>
<tr>
<td>&amp;</td>
<td>binary addition</td>
</tr>
<tr>
<td>^</td>
<td>binary subtraction</td>
</tr>
<tr>
<td>*</td>
<td>binary multiplication</td>
</tr>
<tr>
<td>/</td>
<td>binary division</td>
</tr>
<tr>
<td>+</td>
<td>unary plus</td>
</tr>
<tr>
<td>-</td>
<td>unary minus</td>
</tr>
<tr>
<td>^</td>
<td>power</td>
</tr>
</tbody>
</table>

### Functions

- **ABS(x)**: Returns the absolute value of x.
- **ACOS(y)**: Returns the arc cosine of y.
- **ASIN(y)**: Returns the arc sine of y.
- **ATAN(x)**: Returns the arc tangent of x.
- **ATAN2(y,x)**: Returns the arc tangent of y/x in the correct quadrant based on signs of y and x.
- **CEIL(x)**: Ceiling function: returns the smallest integer that is not less than x. For example: `CEIL(11)` is 11.
- **COS(theta)**: Returns the cosine of theta.
- **COSH(theta)**: Returns the hyperbolic cosine of theta.
- **COT(theta)**: Returns the cotangent of theta.
- **COTH(theta)**: Returns the hyperbolic cotangent of theta.
- **CSC(theta)**: Returns the cosecant of theta.
- **CSCH(theta)**: Returns the hyperbolic cosecant of theta.
- **DEG2RAD(theta)**: Converts theta in degrees to radians.
- **RAD2DEG(theta)**: Converts theta in radians to degrees.
- **DRVSTH(x)**: Returns the derived station from a raw station x, using the station equations. This function is meaningful only for entities that support station equations (alignments, vertical alignments, sample lines, graph profiles, pipes and structures).
- **EXP(x)**: Returns the exponential of x.
- **FLOOR(x)**: Returns the largest integer that is less-than-or-equal to x. For example: `FLOOR(3.5)` is 3.
- **FMOD(x,y)**: Returns the floating point remainder of x/y.
- **GHI2DEG(theta)**: Converts theta in gradians to degrees.
An expression is created by right-clicking on Expressions underneath each label style in Toolspace>Settings, and then click new. The following window appears:

Each label should be given a unique, distinctive name and a description. The name should be kept to a minimum of characters and the description used to describe the full purpose and functionality of the expression. At the bottom of the Expressions window is the formatting option. Each expression should be given the appropriate format to match the desired expression result. Most result formats are self-explanatory and correlate to those formats that may have units assigned to them in the Drawing Settings > Ambient Settings tab. Perhaps the only format that is not self-explanatory is the “Double,” which is nothing more than a unitless value.
OOTB Pipe Parameters Available

OOTB Structure Parameters Available

- Barrel Height
- Barrel Pipe Clearance
- Cone Height
- Floor Thickness
- Frame Diameter
- Frame Height
- Frame Length
- Frame Width
- Headwall Base Thickness
- Headwall Base Width
- Inner Structure Diameter
- Inner Structure Length
- Inner Structure Width
- Rim to Sump Height
- Slab Thickness
- Structure Length
- Vertical Pipe Clearance
- Wall Thickness
- Connected Pipes
- Drawing Scale Conversion
- Energy Grade Line
- Hydraulic Grade Line
- Insertion Rim Elevation
- Known Capacity
- Structure Easting
- Structure Northing
- Structure Offset
- Structure Rotation Angle
- Structure Station
- Sump Depth
- Sump Elevation
- Surface Adjustment Value
- Surface Elevation At Insertion Point
Some things to be aware of when writing expressions for pipe and structures are:

- Expressions may use any standard or custom parameters built into the part catalog, as long as those parameters are numerical values. They may also use the values of other expressions. For Parcels, the user-defined properties may also be used in expressions.

- Expressions may only output numbers, and they may only use numbers as variables of the expression. Although logic statements (If/Then) can be written, the output cannot be a text string.

- There is currently no way to read connected pipe data in structure expressions. There is also currently no way to read the pipe expression values in a structure ‘Text for Each’ label component, which is how structure labels show the information for connected pipes.

- If an expression calls for a parameter that is not actually present in a part, e.g. ‘Pipe Diameter’ in an elliptical pipe, then the resulting output will be ???.

- Pipe End and Pipe Start are based on the direction a pipe is drawn in the model; there currently is not a parameter available for upstream end and downstream end of a pipe… although an If/Then expression could be written to determine this for you.

  - PipeDownstreamInvert = IF({Start Invert Elevation}<{End Invert Elevation},{Start Invert Elevation},{End Invert Elevation})
  - PipeUpstreamInvert = IF({Start Invert Elevation}>= {End Invert Elevation},{Start Invert Elevation},{End Invert Elevation})

- The ‘Pipe Slope’ parameter is based on the (start invert – end invert) / total 2D length of pipe in model space. If you use the parameter ‘2D Length – To Inside Edges’ to label the length of pipe, but the pipes in the model are still connected at the center of structures (default connection point), then the lengths and slopes may not match up to what would be determined by a hand calculation… although there is an expression that can fix that too! Let’s build it on the two expressions above:

  - PipeSlopeToInsideEdges = ((PipeUpstreamInvert)-(PipeDownstreamInvert))/(2D Length - To Inside Edges)

The screen capture below shows a pipe that has been labeled with both the ‘Pipe Slope’ parameter and the new ‘PipeSlopeToInsideEdges’ calculated value.
Here is a view of the properties of the pipe shown above.

<table>
<thead>
<tr>
<th>Pipe Properties</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>End Offset</td>
<td>-3.00%</td>
</tr>
<tr>
<td>Pipe Slope (Hold Start)</td>
<td>3.00%</td>
</tr>
<tr>
<td>Pipe Slope (Hold End)</td>
<td>3.00%</td>
</tr>
<tr>
<td>Slope</td>
<td></td>
</tr>
<tr>
<td>Start Invert Elevation</td>
<td>100.00'</td>
</tr>
<tr>
<td>End Invert Elevation</td>
<td>98.55'</td>
</tr>
<tr>
<td>Start Crown Elevation</td>
<td>102.00'</td>
</tr>
<tr>
<td>End Crown Elevation</td>
<td>100.55'</td>
</tr>
<tr>
<td>Pipe Start Easting</td>
<td>5567.2878'</td>
</tr>
<tr>
<td>Pipe Start Nothing</td>
<td>2847.700'</td>
</tr>
<tr>
<td>Pipe End Easting</td>
<td>5613.7549'</td>
</tr>
<tr>
<td>Pipe End Nothing</td>
<td>2848.0032'</td>
</tr>
<tr>
<td>Start Centerline Elevation</td>
<td>101.00'</td>
</tr>
<tr>
<td>End Centerline Elevation</td>
<td>99.55'</td>
</tr>
<tr>
<td>Minimum Cover</td>
<td>0.00'</td>
</tr>
<tr>
<td>Maximum Cover</td>
<td>0.00'</td>
</tr>
<tr>
<td>2D Length - Center to Center</td>
<td>48.47'</td>
</tr>
<tr>
<td>3D Length - Center to Center</td>
<td>48.46'</td>
</tr>
<tr>
<td>2D Length - To Inside Edges</td>
<td>44.47'</td>
</tr>
<tr>
<td>3D Length - To Inside Edges</td>
<td>44.56'</td>
</tr>
<tr>
<td>Start Cover</td>
<td>0.00'</td>
</tr>
<tr>
<td>End Cover</td>
<td>0.00'</td>
</tr>
<tr>
<td>Resize Behavior</td>
<td>Invert</td>
</tr>
<tr>
<td>On Resize, Hold:</td>
<td></td>
</tr>
<tr>
<td>Hydraulic Properties</td>
<td></td>
</tr>
</tbody>
</table>

Now look at the same pipe after the rule has been applied to set pipe end locations to structure inner walls. Notice the 2D Length – Center to Center parameter will only label to the ends of the pipe; so in essence this parameter should be interpreted as ‘2D Model Length’ of pipe.
Here is a view of the properties of this pipe.

Be very careful with this! Municipal agency plan reviewers have been known to pick up on pipe lengths, slopes, and reported inverts that do not match up correctly; and rightly so! The apparent “right thing to do” would be to always set the pipes to inside edges, but that creates some additional problems with connecting pipes and round-tripping with HSS or SSA… which is
a topic for another day. For now, just understand the 2D Length and Pipe Slope parameters so that your label expressions will calculate and output their intended values.

**Pipe Flow Expressions**

The most common engineering formula for gravity flow in a pipe is the Manning’s Equation (also known as the Gauckler–Manning–Strickler formula). Using this expression, we can have Civil 3D dynamically calculate flow capacity of a pipe in a model without the need to export to HSS or SSA for analysis. This can help engineers during the preliminary storm system layout and design process in Civil 3D to get a closer approximation of what the final designed pipe sizing and slopes will be. This also gives a way to annotate the flow capacity of pipes on a plan without the need for adding a custom parameter to pipes for manual input into each, as Civil 3D does not include a part parameter for pipe capacity or a way to transfer this value from HSS or SSA into the Civil 3D model.

The Manning formula is as follows:

\[
V = \frac{k}{n} R_h^{2/3} S^{1/2}
\]

Substituting the standard flow equation of \(Q=V\cdot A\) and the conversion factor \(k\), we can rearrange this equation to:

**Imperial**

\[
Q = (1.4859/n)A(R_h^{2/3})S^{1/2}
\]

**SI System**

\[
Q = (1.00/n)A(R_h^{2/3})S^{1/2}
\]

Where:

- \(Q\) is the volumetric flow rate passing through the channel reach in cfs (m³/s).
- \(A\) is the cross-sectional area of flow normal to the flow direction in ft² (m²).
- \(S\) is the bottom slope of the channel in ft/ft (m/m) [dimensionless].
- \(n\) is a dimensionless empirical constant called the Manning Roughness coefficient.
- \(R_h\) is the hydraulic radius = \(A/P\).
- \(P\) is the wetted perimeter of the cross-sectional area of flow in ft (m).
- \(k\) is a conversion factor of 1 m¹/³/s for SI, or 1.4859 ft¹/³/s for US Imperial

In Civil 3D we will write this equation as follows to determine the full flow of circular pipes:

Circular_ManningsFullFlow =

\[1.4859^*\pi^*\frac{((\text{Start Crown Elevation}-\text{Start Invert Elevation})/2)^2*\text{(1/(Manning Coefficient))}*(\text{((Start Crown Elevation}-\text{Start Invert Elevation})/4)^*(2/3))}{\text{SQRT(\text{Pipe Slope})}}\]

Note that Start Crown and Invert parameters are used in lieu of Inner Pipe Diameter. This is a carry-over from legacy issues with this equation, however it is a good idea to write in this format as it will be required for non-circular pipes since they do have an Inner Pipe Diameter parameter.
For elliptical pipes, this equation becomes more complicated. The wetted perimeter is more complex, and in fact will only be an approximation based on Ramanujan’s equation for the perimeter of an ellipse.

\[ p \approx \pi \left[ 3(a + b) - \sqrt{(3a + b)(a + 3b)} \right] \]

Instead of one big complex expression we’ll use multiple Civil 3D expression equations to determine the full flow in elliptical pipes:

- **Elliptical_PerimeterFull** = \( \pi \left( 3 \times \left( \frac{\text{Inner Pipe Height}}{2} \right) \right) - \sqrt{\left( 3 \times \left( \frac{\text{Inner Pipe Height}}{2} \right) \right) \times \left( \frac{\text{Inner Pipe Height}}{2} \right) + 3 \times \left( \frac{\text{Inner Pipe Width}}{2} \right) \} \)

- **Elliptical_AreaFull** = \( \pi \times \left( \frac{\text{Inner Pipe Height}}{2} \right) \times \left( \frac{\text{Inner Pipe Width}}{2} \right) \)

- **Elliptical_ManningsFullFlow** = \( \frac{1.4859}{\text{Manning Coefficient}} \times \left( \frac{\text{Elliptical_AreaFull}}{\text{Elliptical_PerimeterFull}} \right) \times \left( 1.4859 \times \frac{\text{Elliptical_AreaFull}}{\left( \frac{\text{Elliptical_PerimeterFull}}{2} \right) \^ 2} \right) \times \left( \sqrt{\text{Pipe Slope}} \right) \)

Note that due to the Commutative Law of mathematics, this expression holds true for both horizontal and vertical elliptical pipes for a full flow condition; this would not be the case for any flow equation other than the full flow equation.

Partially full flow equations would be possible to compute with Civil 3D expressions if custom parameters were added to pipes for flow depth, however this would likely be overkill for a typical H&H workflow. One might wish to compute a maximum pipe flow, which occurs approximately when a pipe is flowing at 90% full, in which case much more complex equations would be required. If you want to tackle it, here’s what you’ll need:

\[ r = \frac{D}{2} \quad h = 2r - y \]
\[ \theta = 2 \arccos \left( \frac{r - h}{r} \right) \]
\[ A = \pi r^2 - \frac{r^2(\theta - \sin\theta)}{2} \]
\[ P = 2\pi r - r\theta \]
\[ R_h = \frac{A}{P} \]

Figure 2. Partially Full Pipe Flow Parameters (more than half full)

\[ y = \text{flow depth} \]

There are lots more equations related to flow, friction losses, etc. in pipes that could be calculated in Civil 3D, however it’s not recommend to overdevelop your templates with this
information since you will still need HSS or SSA to analyze pipe networks… at least for the foreseeable future.

**Logic Expression (If/Then) for Civil 3D Label/Table Output**

Recently a request came in to tackle a problem with the way Civil 3D tables present the information from the model. Tables do not yet have the ability to format a data cell based on the shape or family of a part. Although there’s not much we can do at the moment to have a Civil 3D table label both a circular structure size and a rectangular structure size in the same column (without using the ‘Description’ parameter), we can take advantage of expressions to make some decisions for annotating varying part configurations in the same column.

In our full flow ($Q_{full}$) examples above, we need a way to have Civil 3D determine if a pipe is round or elliptical and report either ‘Circular_ManningsFullFlow’ or ‘Elliptical_ManningsFullFlow’ and report the correct value in the same table column. To do this, we must have a numerical parameter value in the pipes to use in an expression. The value must be capable of differentiating the pipe shape. Unfortunately we cannot do this with the OOTB pipe parameters, so we have to add a custom parameter. Custom parameter creation is covered in the next section of this document. You will see in the next section of this document that a custom pipe parameter has been added to the part catalog for Equivalent Round Size (EQRND). This optional parameter was defined in the horizontal elliptical pipe family in the table that defines inner pipe height for the defined inner pipe width (Part Builder is required to accomplish this). Since EQRND is an optional parameter, it will be available on every part in a Civil 3D file; it just won’t be specifically defined in any catalog parts except elliptical pipes. The default value for EQRND was set to 0.00” (setting default values for optional parameters is covered in the section below). After updating the template standard Parts List to include the default 0 value for EQRND, we’re ready for an expression. This expression will allow logic to determine if a pipe is elliptical (EQRND ≠ 0) or circular (EQRND = 0). The expression is as follows:

\[
\text{ManningsFullFlow\_PipeDecision} = \begin{cases} 
\text{Circular\_ManningsFullFlow} & \text{if EQRND} = 0 \\
\text{Elliptical\_ManningsFullFlow} & \text{if EQRND} \neq 0 
\end{cases}
\]

Now instead of using separate label styles for elliptical or circular pipes, we can simply use the new ManningsFullFlow\_PipeDecision to report the full flow. Although we won’t cover it at this time, it would be possible to embed a second IF/THEN statement into the first in this expression to determine if a pipe is a rectangular culvert.

**Label Expressions for Style Formatting**

You may notice in some areas of the label style composer where numerical inputs are required, such as label attachment Y-offset or Maximum Width, that expressions may be used for the value in lieu of a specified value. The screen cap below shows an example of using an expression called MaxPipeTextWidth for the specified maximum width of the text box for a pipe label. The expression is as follows:

\[
\text{MaxPipeTextWidth} = \frac{\text{2D Length - To Inside Edges}}{\text{Drawing Scale Conversion}}
\]

The Drawings Scale Conversion variable must be used since Civil 3D text is annotative based on drawing scale. Be careful; only one ‘Drawing Scale’ can only be set in model space or in the file drawing settings, but you may have multiple viewport scales you’re working with that are different than the ‘Drawing Scale’.
Below is a screen cap of how this variable will affect the pipe label on varying length pipes. Each label uses the same label style with no manual text edits.

Another useful formatting option would be to utilize an expression in conjunction with the “hide negative value” in the label style composer to format individual label components. In this example two expressions are needed: ‘Q_{full}-Q’ and ‘Negative_{Q_{full}}-Q’. The expressions are as follows:
Q_{\text{full}} - Q = \{\text{ManningsFullFlow\_PipeDecision}\} - \{\text{Flow Rate}\}

\text{Negative}_{Q_{\text{full}}} - Q = (-1) * \{Q_{\text{full}} - Q\}

That’s it for the expressions. The next step is to simply add these expressions to a label style. For this workflow, you will create two separate text components which will both be identical other than their Text Content and color. I’ve added an extra text component in this example that denotes under capacity numbers are in green and over capacity numbers are in red.

The UnderCapacity component contents is shown below. Note the item being labeled is the \(Q_{\text{full}} - Q\) value and the Sign modifier is set to ‘hide negative value’.
The OverCapacity component contents is shown below. Note the item being labeled is the Negative $Q_{full}$ value and the Sign modifier is set to ‘hide negative value’.

The final output will show the appropriate text component in its specified color while the opposite value will be hidden.

Explore other options with these expression workflows. The output can be very beneficial to engineers and designers to both speed up the process and eliminate mistakes during the design. Expressions and styles can take significant time and patience to produce, but once they’re in the template and utilized in standard workflows the effort will pay off.

Just an aside note… SAVE YOUR WORK TO A TEMPLATE. Don’t created these things and then have them forgotten in an old drawing file. Save it to the template for you and others in your organization to use on future projects.
Task 2: Customize the Part Catalog Shared Parameters for Enhanced Information Modeling and Data Entry

Pipe Network Catalog Location
The Out-of-the-Box (OOTB) default location for the Pipe Network Catalog is set to C:\ProgramData\Autodesk\C3D 2013\enu\Pipes Catalog.

It is highly recommended that a network location be chosen for a shared catalog for an organization (or divisions of an organization). All users should not be given write access to this network location, as modifications to parts inside the company standard catalog has potential to drastically affect both existing and future project work. For example, the OOTB Storm Sewers parts list consists of the Reinforced Concrete Pipe (RCP) family, which was found to be inaccurate for many standard pipe wall thicknesses (depending on classification of RCP). It was decided by BIM Manager to create new families for RCP-WallB and RCP-WallC, which have standard wall thicknesses per the American Concrete Pipe Association. Many ongoing legacy projects utilized the original RCP family, so the family could not be deleted from the catalog completely without negatively affecting those projects. The standard Parts Lists in newer template updates used the new RCP-WallC family as the standard, and users were educated via multiple email blasts about the changes.

Always keep backups of original catalog files and periodic backups of files throughout the years of development.

XML File to be Edited
Much of the BIM related content of Civil 3D objects is stored in XML files. These are where we will go to begin customizing the part catalog. The first XML file we will be looking at is in the Aecc Shared Content folder of the Catalog.

The file is titled AeccPartParamCfg.xml, and it houses all of the possible parameters of all structures and pipes in the catalog. For this lesson we will not be digging into Part Builder, but be aware that these edits we are making to the XML file are required before a specific parameter may be assigned and used in Part Builder.
Don't be afraid! At first glance into this file, the XML code can look very intimidating. When you collapse it all down you can see that there are only two sections of information in the file, and when adding part parameters you will need to add information to both of these sections in the file. These are AeccParamDeclaration and AeccParamUsage.

<?xml version="1.0"?>
<!-- AeccParamCfg.xml file must be located in the Shared Content Directory -->

- AeccParamConfigSet

  <!-- AeccParamDeclaration defines both internally managed and extended part data parameters -->
  + AeccParamDeclaration
  + AeccParamUsage
</AeccParamConfigSet>

Use copy and paste, and stick to the format presented in the OOTB parameters that exist in the file.

The photos presented above are how these files will typically open by default in Internet Explorer. For visualization purposes of this document the files will be shown with this XML viewer in most of the screen capture photos. During the AU session we’ll edit them using the basic Notepad program in Windows, but there are several XML editing software packages available for free on the web if you want more organization and colorization of the text during editing.

Note that there are several instances of comments inside of the XML file that simply give comments about content in the file but are not usable to external applications. These are denoted by the characters `<!--`. It is a good idea to use a comment line where your added
content is located in the file. You may also use spacing when editing in Notepad to help organize the user-added data.

We'll start at the bottom with the more simplistic *AeccParamUsage* inputs. In this area you can see if a parameter is defined as a required parameter (*AeccReqParam*), a shape parameter (*AeccShpParam*), or an optional parameter (*AeccOptParam*). We'll stick to adding optional parameters in this lesson for simplicity. Note that optional parameters will appear in the Civil 3D part properties for all parts in a model.

The Pipe parameter data added by GSP BIM Managers is shown below

```xml
<?xml version="1.0" encoding="UTF-8"?>
<Declaration>
  <ParamUsages>
    <ParamUsage name="Pipe_Domain">
      <ParamType name="Pipe">
        <ReqParam name="SweptShape"/>
        <ShpParam name="PipeInnerDiameter"/>
        <ShpParam name="PipeInnerHeight"/>
        <ShpParam name="PipeInnerWidth"/>
        <OptParam name="Material_Type"/>
        <OptParam name="MinCurveRadius"/>
        <OptParam name="FlowAnalysis_Manning"/>
        <OptParam name="FlowAnalysis_HazenWilliams"/>
        <OptParam name="FlowAnalysis_DarcyWeisbach"/>
        <OptParam name="FlowAnalysis_DesignStorm"/>
        <OptParam name="FlowAnalysis_Velocity"/>
        <OptParam name="EquivalentRound"/>
      </ParamType>
    </ParamUsage>
  </ParamUsages>
</Declaration>
```

The Pipe parameter data added by GSP BIM Managers is shown below
The Structure parameter data added by GSP BIM Managers is shown below

```xml
  <AeccOptParam context="EquivalentRound"/>
</AeccPartTypeCfg>
</AeccPartDomainCfg>
- <AeccPartDomainCfg desc="Structure_Domain">
   <AeccReqParam context="StructBoundingShape"/>
   <AeccReqParam context="StructVertPipeClearance"/>
   <AeccShpParam context="StructHeight"/>
   <AeccShpParam context="StructDiameter"/>
   <AeccShpParam context="StructWidth"/>
   <AeccShpParam context="StructLength"/>
   <AeccOptParam context="Material_Type"/>
   <AeccOptParam context="PartSizeValidation_Test"/>
   - <AeccPartTypeCfg desc="Junction Structure" type="Struct_Junction">
      <AeccReqParam context="StructRimToSumpHeight"/>
      <AeccReqParam context="WallThickness"/>
      <AeccReqParam context="FloorThickness"/>
      <AeccOptParam context="StructGrate"/>
      <AeccOptParam context="StructCover"/>
      <AeccOptParam context="StructFrameHeight"/>
      <AeccOptParam context="StructFrameDiameter"/>
      <AeccOptParam context="StructFrameLength"/>
      <AeccOptParam context="StructFrameWidth"/>
      <AeccOptParam context="StructFrameHeight"/>
      <AeccOptParam context="StructBarrelHeight"/>
      <AeccOptParam context="StructBarrelPipeClearance"/>
      <AeccOptParam context="StructConeHeight"/>
      <AeccOptParam context="StructSlabThickness"/>
      <AeccOptParam context="StructRiserHeight"/>
      <AeccOptParam context="StructInnerDiameter"/>
      <AeccOptParam context="StructInnerLength"/>
      <AeccOptParam context="StructInnerWidth"/>
      <!-- Custom Optional Parameters by GSP Employees-->
      <AeccOptParam context="StructGrateSize"/>
      <AeccOptParam context="FlowAnalysis_DrainageArea"/>
      <AeccOptParam context="FlowAnalysis_RunoffCoefficient"/>
      <AeccOptParam context="FlowAnalysis_TimeOfConcentration"/>
      <AeccOptParam context="FlowAnalysis_QCaptured"/>
      <AeccOptParam context="FlowAnalysis_Intensity"/>
      </AeccPartTypeCfg>
  - <AeccPartTypeCfg desc="Inlet-Outlet Structure" type="Struct_InletOutlet">
```

Note that there are separate places you may add the optional parameters for all structure types, only to junction structures, only to inlet-outlet structures (typically endwalls), or only to general structures. These areas are denoted by the `<AeccPartTypeCfg desc='` lines.
Here are the same areas as they appear in Notepad

```xml
</AecParamDeclaration>
</AecParamUsage>

<AecPartDomainCfg domain="Pipe_Domain">
  <AecCRevParam name="SweepShape"/>
  <AecCRevParam type="Pipe" desc="pipe"/>
  <AecCRevParam Context="WallThickness"/>
  <AecCRevParam Context="PipeInnerDiameter"/>
  <AecCRevParam Context="PipeInnerHeight"/>
  <AecCRevParam Context="PipeInnerWidth"/>
  <AecCRevParam Context="MaterialType"/>
  <AecCRevParam Context="MinCurveRadius"/>
  <AecCRevParam Context="FlowAnalysis_Manning"/>
  <AecCRevParam Context="FlowAnalysis_KniefWilliamson"/>
  <AecCRevParam Context="FlowAnalysis_HazenWilliams"/>
  <AecOptParam Context="FlowAnalysis_DesignStorm"/>
  <AecOptParam Context="FlowAnalysis_Velocity"/>
  <AecOptParam Context="EquivalentRound"/>
</AecPartDomainCfg>

</AecPartDomainCfg>

<AecPartDomainCfg domain="Structure_Domain">
  <AecCRevParam name="SweepShape"/>
  <AecCRevParam Context="StructBoundingShape"/>
  <AecCRevParam Context="StructVertPipeClearance"/>
  <AecCRevParam Context="StructHeight"/>
  <AecCRevParam Context="StructDiameter"/>
  <AecCRevParam Context="StructWidth"/>
  <AecCRevParam Context="StructLength"/>
  <AecCRevParam Context="MaterialType"/>
  <AecOptParam Context="PartSizeValidation_Test"/>
  <AecPartTypeCfg type="Struct_Junction" desc="Junction Structure">
    <AecCRevParam Context="StructRimToSumpHeight"/>
    <AecCRevParam Context="WallThickness"/>
    <AecCRevParam Context="FloorThickness"/>
    <AecOptParam Context="SumpFrame"/>
    <AecOptParam Context="SumpGrate"/>
    <AecOptParam Context="SumpCover"/>
    <AecOptParam Context="SumpFrameHeight"/>
    <AecOptParam Context="SumpFrameDiameter"/>
    <AecOptParam Context="SumpFrameWidth"/>
    <AecOptParam Context="SumpFramelength"/>
    <AecOptParam Context="SumpBarrelLength"/>
    <AecOptParam Context="SumpBarrelHeight"/>
    <AecOptParam Context="SumpBarrelPipeClearance"/>
    <AecOptParam Context="SumpCollarHeight"/>
    <AecOptParam Context="SumpCollarDiameter"/>
    <AecOptParam Context="SumpCollarInnerDiameter"/>
    <AecOptParam Context="SumpCollarInnerLength"/>
    <AecOptParam Context="SumpCollarInnerWidth"/>
    <AecOptParam Context="StructureGratesize"/>
    <AecOptParam Context="FlowAnalysis_DrainageArea"/>
    <AecOptParam Context="FlowAnalysis_RunoffCoefficient"/>
    <AecOptParam Context="FlowAnalysis_TimeOfConcentration"/>
    <AecOptParam Context="FlowAnalysis_SQCaptured"/>
    <AecOptParam Context="FlowAnalysis_Intensity"/>
  </AecPartTypeCfg>
  <AecPartTypeCfg type="Struct_InletOutlet" desc="Inlet-Outlet Structure">
    <AecCRevParam Context="StructInletWidth"/>
    <AecCRevParam Context="StructOutletWidth"/>
    <AecOptParam Context="StructInletDiameter"/>
</AecPartDomainCfg>
</AecPartDomainCfg>
```
And here is how they equate to what is seen in the Civil 3D part properties menu

Now let’s look at the less simplistic AeccParamDeclaration inputs. Just like with the others, you should always copy and paste the information from one of the OOTB parameters. First of all, note that the parameters are not organized in different location by “Pipe_Domain” or “Structure_Domain” in this section. Like the others, we should organize our custom parameters with an editor’s comment line; although not shown below you may want to separate pipe and structure parameters with separate editor comment lines.

The first input is the parameter name, which is what will is used inside of Part Builder if you dare to venture there.
Here is a View of Part Parameters inside of Part Builder; Note the custom EQRND parameter

The second and third inputs should always be set to “True” for custom parameters.

```xml
<Parameters by GSP Employees-->
  ="GRT" internal="True" visible="True" unit=
  ="DSTRM" internal="True" visible="True" unit=
  ="ACDA" internal="True" visible="True" unit=
  ="ACRC" internal="True" visible="True" unit=
  ="ACTC" internal="True" visible="True" unit=
  ="ACQC" internal="True" visible="True" unit=
  ="ACI" internal="True" visible="True" unit=
  ="ACV" internal="True" visible="True" unit=
  ="EQRND" internal="True" visible="True" unit=
```

The next input, unit, doesn’t appear to be all that important. For good practice go ahead and designate the proper units for each custom parameter, but don’t expect these to show up anywhere else. You can leave them blank without any negative consequences (that I know of)
The next input, the usage, should be carefully selected to represent the parameter being added. The following are some of the more common usages and their meaning:

- Double_Distance: A decimal number for large distance format (feet or meters).
- Double_SmallDistance: A decimal number for the small distance format (inches or millimeters).
- Double_General: A decimal number with no specific units.
- Int_Distance: A whole number for large distance format (feet or meters).
- Int_SmallDistance: A whole number for the small distance format (inches or millimeters).
- Int_General: A whole number with no specific units.
- String_General: A general text string, which can include any alphanumerical characters.

The next input, index, should always be equal to 0 for custom parameters.
For the last two inputs, match the ‘context’ that was entered in the `AeccParamUsage` context, and enter a description for the parameter. The description will be visible to Civil 3D users and should clearly explain what the parameter is used for.

An optional input may be added to the end of the parameter declaration line that defines if the optional parameter is managed list of values when viewed in the Civil 3D part properties menu. The input is manageList="True" or manageList="False". Leaving this input out altogether is the equivalent of the False setting. If parameters in pipe or structure components have a finite set of options, a manageList="True" setting is recommended. Note that manual inputs in Civil 3D will add items to this list, and anyone who has write access to the part catalog files will be able to add components to this list simply by typing in a value in Civil 3D. The next topic discusses this list in further detail.
Setting the Default Values and Drop-down List Values of Optional Parameters

You may have noticed inside of Civil 3D with the OOTB content that you have access to a drop-down list for part materials when editing a Parts List. You can either type a character string or select one of the predefined options from a list, as shown below.

You can also access the list from within the part properties after the part has been placed in the model, however accessing it from this location requires the managelist="True" value to be defined in the AeccParamDeclaration inputs.

**At the time of developing this document, there were known issues with some parameters not displaying the list drop down in Pipe Properties or Structure Properties dialog boxes. Troubleshooting is ongoing and will be reported in Autodesk Civil 3D forums when a resolution is found.**

This is a good time to note that the “Reinforced Concrete” material that is defined in the OOTB concrete pipe family that you cannot change inside of Civil 3D can be corrected. This material was specifically defined to concrete pipes in release 2012 and has spurred many questions as to why it couldn’t be edited since then. Just open the family in Part Builder and delete the ‘Mat’ parameter from the Size Parameters list. ‘Mat’ is an optional parameter and can be filled in with any text string on any pipe in the catalog. I wouldn’t recommend specifically defining materials in Part Builder so that the users cannot change the value inside of Civil 3D.
You might also have noticed that if you have typed in an alternative string that wasn’t previously in the list, then it was added to the bottom of the list for future use… even if you typed something incorrect! Have no fear, if this list gets out of control you can always correct it. You can also set the order of the values in the list. It’s time to dig back into some XML files.

The files we’re going to edit is called AeccSharedPropertyLists.xml, and there are separate versions located in each of the Pipe and Structure catalog folders. By default, these files are located in C:\ProgramData\Autodesk\C3D 2013\enu\Pipes Catalog\US Imperial Pipes and C:\ProgramData\Autodesk\C3D 2013\enu\Pipes Catalog\US Imperial Structures.

*For those on the SI system, use the \Metric Pipes and \Metric Structures locations.

OOTB, the structure file Shared Property file looks like this

```xml
<?xml version="1.0"?>
<StructTable xmlns:xlink="http://www.w3.org/1999/xlink" version="1.0"
    desc="Part Table">
    <Images>
        <Image xlink:href="Part Reference Image" xlink:type="PathToTmp.jpg"/>
    </Images>
    <Recipe path="/\Metric Pipes\Metric Structures">
        <ColumnConstView desc="Parameter-driven Display" viewName="AeccPartRecipe" viewKey="3d" id="CCV1">
            <Images />
        </ColumnConstView>
    </Recipe>
</StructTable>
```
Your file may already look drastically different if you’ve keyed in a lot of values into Civil 3D in the past. Below is an example of a Structure Shared Property List file that has been updated and managed.

Notice in the screen below that the default settings when adding a part size inside of Civil 3D correlate to the Item id="i0" value inside of the Shared Property List shown in the screen above.
You might also notice one exception to the “i0” default correlation between the two previous screen caps, which is the Grate Size value. This particular part family, Inline or Drain Basin with Rectangular Grate, is a custom part that has the Grate Size (GRT) parameter defined in Part Builder with a list value. When this occurs, the parameter is no longer Optional but a defined value from the part family and the values from the AeccSharedPropertyList.xml file do not apply.

Used the example Shared Property List format above to create your own lists and default values for your Optional part parameters. Note that the Item id value must increase in the list. Also note that if you wish to add new parameters to this list you should must change the id="CCL#" value.

Example Parameter Setup
A new optional part parameter is desired to add a city standard detail number in for a storm structure. The standard details will be STD-001, STD-002, and STD-003.

The following steps should be followed to allow this new optional parameter to be used:

1. Open the company standard catalog, and open the AeccPartParamCfg.xml file (remember to open with an XML editor or Notepad)
2. Add the parameter to the AeccParamDeclaration section by copying one of the existing parameters. This will be a general text string, so copy a parameter with the usage="String_General" so that less edits need to be made. For this example, parameter declaration will be as follows:

   `<AeccDfParameter name="NashStdDtl" desc="Nashville Standard Detail" context="Nashville_StandardDetail" index="0" datatype="String" usage="String_General" managelist="True" unit="" visible="True" internal="True"/>`
3. Add the parameter information to the `AeccParamUsage` section by copying one of the existing lines and pasting. For this property, we want the parameter to work for all structure types, so it will need to be added under the top category. The parameter usage will be entered as follows:

```xml
<AeccOptParam context="Nashville_StandardDetail"/>
```

4. Open the `AeccSharedPropertyList.xml` file located in the Structures folder of the catalog. For this example, we want three standard detail numbers, but by default we want the values to be blank because we may not use these in all cases for putting structures in a model. This is the 9th entry in the file, so the id will be updated to CCL9. We will enter the following content to the bottom of the file:

```xml
<AeccOptParam context="Nashville_StandardDetail"/>
```
5. After the XML files are modified and saved, make sure to restart Civil 3D. These files are loaded upon startup of the software, and the edits won’t be recognized in the currently running instance of the software. After restart, the parameters should be available for input in new parts list and also in existing model parts inside of existing files.

One last thing that bears repeating… It is important to have a centralized location for the company standard part catalog. These parameters will not be recognized by users who do not have access to the standard part catalog or do not have the standard part catalog set correctly inside of their running instance of Civil 3D. Educate your users on the importance of using the company standard catalog. If you’re sending BIM files out of house for use, be sure to send the catalog as well. After setting the catalog to the centralized location, Civil 3D must be restarted for the XML files to be read from the newly pathed catalog location.

---

**ColumnConstList desc="Nashville Standard Detail" dataType="string" unit="" name="NashStdDtl" id="CCL9" visible="1" context="Nashville_StandardDetail" index="0"**

<Item id="i0"></Item>
<Item id="i1">STD-001</Item>
<Item id="i2">STD-002</Item>
<Item id="i3">STD-003</Item>
</ColumnConstList>
Task 3: Customize Parcel Styles and User-defined Properties for Hydrology Information Modeling

Setting up Sites for H&H Workflow with Parcels
As with feature lines, parcels on the same site will interact with one another. No single area of a site may exist in more than one parcel. This workflow will require separate sites to do overall watersheds and surface-type watersheds in the same file. If you wish to do multiple hydrology calculations in a single file, such as Rational Method and SCS method, you will need separate sites for each.

Note that in Toolspace you can see information about the parcels in these sites by clicking the main Sites category.

Setting Up the Parcel User Defined Properties
The most important step to using Parcels for H&H workflow is to setup the user defined properties to hold the H&H information (the "I" in BIM!). If you are not an engineer or are not well-versed in hydrology, ask an engineer or hydrologist for assistance during the early stages of development of these properties and the calculations involved.

!!Warning!! User Defined Properties do not sort alphabetically. Use Classifications to enhance sorting of the User Defined Properties. Note, Classifications do not sort either, but they will help group the various properties for ease in locating the properties for information input. Also note, User Defined Properties and Classifications may not always appear in the same order. Resorting has been observed while adding additional User Defined Properties or Classifications and after AUDIT on the file.
Note the random order of both Classifications and Properties

Try to plan out a good naming convention during development. Feel free to use the one above for SCS and Rational hydrology. Also, be sure to set a default property value that will be commonly used when appropriate. For calculation purposes default areas should be set to 0.

User Defined Properties and Classifications are created from the Settings tab of Toolspace. Right-click on User-Defined Property Classifications to create a new Classification. Right-click on a Classification to create a new User Defined Property.
There are a few things to note about Classifications:

- There is a classification called “General” that does not appear in the list by default, however if you look in the properties of a created parcel you will see the General Classification with Parcel Number, Parcel Address, and Parcel Tax ID. You can create another Classification called “General” and add properties to it. These new properties will be added to the General Classification that already exists.

- A classification will not show up in Parcel Properties until there is a User Defined Property created inside of the Classification.

- The “Unclassified” Classification may be deleted (if there no parcel label expressions already created); however it will recreate itself when parcel label expressions are created.

Creating the Area Label Expressions for Automated Calculations

The whole point of creating all of those User Defined Properties is so that they may be utilized for labels, tables, and automated calculations. Forget the calculators and spreadsheets; let the model do the math for you!
Parcel expressions for H&H calculations are created in Toolspace under Parcel Label Styles>Area. When creating expressions, be sure to give each one an easy-to-understand name and a thorough description. A good BIM Manager / Template Developer should always keep in mind those who may be coming into these settings afterwards.

Note the expression named SCS Calculated Impervious Area… we’ll come back to that later in this document.

Below is an example for calculating the weighted CN for a delineated watershed;
Creating the Parcels
Before beginning the process of creating parcels, make sure that the Sites, Parcel Styles, and Area Label Styles have been fully developed. Refer to Task 4 below titled “Prepare Standardized or Specialized Drawing Templates for a Fully Integrated Hydrology and Hydraulics Workflow” for more information on setting up styles for H&H templates.

Throughout the process of creating watersheds and catchment area parcel objects the recommended workflow is to create a catchment from surface or delineate an area with polylines, and then use the command Create Parcels From Objects to create 1 or more areas at a time. When using this command you also have the option to create from Xref objects; this is useful for creating from closed property lines, building lines, and interior parking lot islands.
Unionizing Parcels
Many times you will need to unionize several smaller parcel areas into one large parcel area. For example, you may have 20 curb islands in a parking lot that need to be joined into the overall watershed pervious area. Use the Parcel Creation Tools > Parcel Union to accomplish this.
Follow the prompts to select the destination parcel first, then select all of the parcels you wish to add to the destination parcel. Notice from the screen caps that colors are present in the destination parcel, but white in the other. We keep our H&H parcel styles set to Byblock for colors so that we can use various colors to better organize catchment areas in the file.

Input the Parcel Areas and CN / C values into the User Defined Properties
After delineating and unionizing all of your sub-areas of an overall catchment, input your individual areas and curve number / runoff coefficient values into the User Defined Properties that have been setup in the H&H template.
Discover New Workflows; Improve Your Templates!
This workflow is great, but I think we can do better! This next workflow addition came after the initial development and deployment of the H&H template with the SCS workflows that were originally setup. Instead of putting this section above, under the section “Creating the Area Label Expressions for Automated Calculations,” it was purposely put here to call to attention that workflows can always improve. Never stop developing, and always update the template with improvements so others in your organization may take advantage of them!

It is recommended that an expression be written for the most common surface type area to save the step of delineating one area type or hand computing that area type based on ‘Parcel Area – sum of all other area types’ if you do not create sub areas for all the surface types. For example, impervious surface is generally the most common surface type in a commercial development, and the SCS Curve Number is typically 98.

We’ll do the following steps to have Civil 3D automatically compute the impervious surface type so we do not have to delineate it:

- Create the expression to calculate the impervious area.
- Display the calculated impervious area in the Parcel Area Label Style.

- Modify your weighted CN equation to utilize the new calculated impervious area.
Let’s go back and modify the User Defined Properties of the overall watersheds we were working on earlier to remove the impervious value input as SCS – Area 1. This is not a normal step in the process, but it is necessary for times such as this when the workflow improvement is made in the middle of a project.

That’s it. The resulting weighted CN is the same. Now we have 8 separate wooded or meadow surface type inputs available for use without the need to delineate impervious for the SCS workflow.

**Modifying Parcel Watersheds for Revisions**

Modifying a parcel is just like modifying a feature line. Use the contextual ribbon for geometry editing. Insert or Delete PI and grip editing are the most commonly used tools.
Use the Fit Curve tool to create and modify curve segments
You cannot delete a parcel itself without using a workaround (create a new site, move parcel to site, and delete site). Most of the time you will be able to simply delete parcel segments to get rid of unnecessary areas of a parcel. Select the segment, and hit the delete key.
With each modification, you should notice the areas updating.
This is a good time to note… Parcels can be tricky to understand and even finicky at times! Notice above that the parcel style is SCS Subsheds – Meadow, and the color of area fill red while the parcel segment is white. The segment is set to BYBLOCK color, just as the fill is, but it is not taking the red color. This is because parcel segments act as separate objects from the parcel area itself. Area labels and area fills take on the same ByBlock object properties, however they are controlled in two separate style areas (parcel style and parcel area label style).

Spend some time editing parcels and styles to get to know how all the pieces interact. As covered in the section above, modify your templates as your workflows evolve and you find out how all the pieces interact with one another. Document your findings and template revisions! Remember, you have to teach these workflows to the users when you’re finished and share your template revisions so no one is surprised (confused) after the changes are made.

Some other Parcel issues that may be encountered are:

- Empty parcels floating around the edges of your work area. This could happen when deleting or editing segments that are not fully closed; a parcel escapes out into space. If this happens you can try to delete it with the Move to Site workaround above, or just leave it and set the area selection label style to <none> from the parcel properties window.

- An overall Site parcel will appear that consists of the addition of all parcels in a particular Site. In this example below, the parcel has created and displayed itself. This is not a standard parcel, and User Defined Properties cannot be assigned to it. It is recommended to create a _No Display style for the parcel area label so that it does not appear on your drainage maps. Alternatively you could give it a different area label style so that it denotes the sum of all the catchment areas on the plan, without the user-defined parameters in the label.
In model space the overall Site parcels appear as shown

However, in Toolspace they do not appear at all

- You may encounter tiny parcels in areas around other parcel segments. This usually is due to imperfect geometry (i.e. two parcel segments from adjoining parcels not lying exactly on top of one another). Because parcels on the same site cannot occupy the
same space and any single segment created inside of a parcel will interact with and split the overall parcel, it is recommended that no overlapping parcel segments be created. Use single parcel segments two split into two parcels instead of overlapping segments.

- Parcels may display strange looks to them, as if the style is not being recognized. No good explanation has been found for this, but it has been noted that the issue does not occur as often if dividing parcel segments are endpoint snapped to a grip of the outer segment line rather than being nearest snapped to anywhere on the outer segment.

- If you disconnect multiple parcel segments by editing a segment, one of the parcels will disappear while the other takes over the overall space created by the disconnect. You cannot get the user defined property information back when this occurs. When reconnecting the parcel segments, you will need to reenter the information, as a brand new parcel will be created with the last used parcel style and area label style used. The parcel that is created by reconnecting will also take the current layer that is active instead of the Civil 3D parcel object layer, which can create problems for Bylayer settings or Select Similar command. To reiterate the importance of the bullet point above, snapping new parcel segments to an endpoint grip (or inserting a PI on the intersecting segment after the fact) will help to reduce these problems when editing parcel areas. If the parcel segments are two different objects, then select them both before moving the grip.

After 2 or 3 projects using this workflow you should pick up on a few Parcel quirks and know what to do in your workflow to avoid them or fix them.

Parcel Area Tables
As with Pipe Network tables, Parcel tables can also be a useful tool in presenting an abundance of information all in one neatly organized format. Adding Civil 3D Parcel Area tables is found in the Annotate tab of the Ribbon.
Table styles may vary greatly depending on the type of information your organization or local regulatory requirements require you to present. More than likely you will only need a table style for overall SCS Method watersheds and overall Rational Method catchments.

When the Table Creation window is first presented you are given the option to include all parcels using a particular label style or to select individual parcels to include. When choosing to add by label style you are also given the option to add existing parcels only or to add existing and all future parcels created with the chosen label style(s). You will need to select at least one parcel using the object selection tool from the window (green box with cursor). The other parcels with the SCS Watersheds – Full Area Label will also be included when the table is made.

Note that once a parcel is included in a table it will not be removed if the area label style is changed. The selection rule says it will add existing and add new, but it will not delete automatically. Have no fear! The “Add Items” and “Remove Items” tools from the Contextual Ribbon actually work with parcel areas (unlike with pipe network tables).
For the template default it is recommended to include all subshed areas, as pictured above, and let users delete table columns as needed. For this example only the SCS AREA 1 and SCS AREA 2 labels are needed, and we need to add the Calculated Impervious Area that we created after we improved the workflow.
Also note that in the table style editor there is a sorting option where the table may be sorted by any column in the table. For this example the sorting column has been set to column 1 so that the parcels will sort alphabetically by name. Sort will occur as most standard computer logic occurs, numerically first then alphabetically. The most reasonable setting to have in the standard template would be to sort by parcel name (column 1 in the screen cap above), however you may also wish to sort by properties such as Hydrologic Soil Group, weighted CN, total area of watershed, etc.

The Final Product
Creating Catchment Objects from H&H Parcels
To round-trip data to and from HSS or SSA, we need to convert all of our work developing these drainage maps with parcel objects to a format that Civil 3D can use to transfer the data through the .stm file. Catchment objects do just this. They also offer a workflow to produce SCS time of concentration flow paths.

The original concept developed for utilizing Parcels for H&H workflow spawned from the limitations of the original release of Catchments in Civil 3D 2012. In 2012, once Catchment objects were created they could not be modified. In 2013 Catchment objects could be modified after creation, however as of 2014 they still do not have as robust functionality for watershed and catchment maps as parcels. They do have label expressions capability, but without a way to add your own information parameters such as Parcel user-defined properties there is not much you can do for calculations. They also do not yet have dynamic table capabilities.

Although we cannot dynamically link the Parcels and Catchment objects created with this workflow, the catchment objects are very easy to produce from the Parcel delineation. It’s as simple as exploding the Parcels of your catchment map… wait what!!!!?? Yes, explode the parcels. Fortunately Parcels will not be erased when exploded. It’s the equivalent of extracting boundaries from a surface; the Civil 3D object is retained and a ‘dumb’ 2D AutoCAD object is brought out. !!!Warning!!! Be sure you explode only the Parcel and NOT the Parcel segments. Parcels are selected by picking the area label or by selecting in Prospector under the Site it was created in, as shown below.
Here is what will come out of this Parcel after exploding

We created the Mtext from the label, a polyline from the label border, a hatch pattern from the hatch area fill, and a single polyline from the Parcel boundary. You might be thinking, “That’s going to be a pain to clean up all the text, hatches and polylines created from the text border… all I want is the border to create a Catchment Object from.” It’s actually very simple to clean up if you’ve set layers appropriately in your parcel and parcel area label styles. Each object that comes out will be on the layer defined in the style, so you can use the Select Similar tool to pick them all at once and delete.
Below the text box polyline is selected; the layer is shown

Here the parcel boundary polyline is selected; it took on the parcel object layer

Note also that any time you explode a parcel the bounding polyline will always be closed. This is perfect for creating a Catchment Object.

Knowing this, here is the workflow to quickly obtain all of the bounding polylines (or 3D polylines if you had elevations set on the parcel vertices):

1. In Prospector, drop down the Site that the desired Parcels are in, click the Parcels heading, shift-select all the parcels in the displayed in the Prospector list, right-click and click ‘Select’.

---

54
2. With all parcels selected, use the EXPLODE command.
3. Select any text, polyline text border, and hatch patterns that came out of the exploded parcels. Right-click and choose Select Similar, and then delete.

Now it’s time to create the Catchment object and link it to the structure. The first step for creating Catchment objects is to create a Catchment Group. This can be done from either right-clicking catchments in Prospector or from the Analyze tab of the Ribbon. There’s nothing special about creating a Catchment Group, just name it. You may choose to use separate groups for separate networks or separate areas of a single network.

Next you will use the Create Catchment from Object command.

It’s a good idea to use the Selection Cycling tool here, as you will be selecting from overlapping polylines, parcel segments, and other Catchment Objects. You must pick your polyline catchment boundaries one at a time to create the Catchment Objects. Each time you pick one, Civil 3D will ask to define a flow path. You can do this if you have already drawn in Tc flow paths, but more than likely you will just skip this by pressing Enter or Escape.
Here you will get the typical Civil 3D window for defining an object. Give the Catchment a name; I recommend naming it to correlate with the structure name it will link to. Since you already have your weighted C values calculated from the H&H Parcel workflows, you can read that off of the Parcel area label and assign it as the runoff coefficient. If you need to move around in the model space to find the label, use the Reference pipe network structure object selection button to put you into model space. While you’re there go ahead and select your structure that the catchment will be referenced to. Note that the pipe network will need to exist in the drawing file; it may be data shortcut referenced but not Xreffed. Assign your Catchment style and Catchment label style, and since you won’t be needing the polyline anymore choose ‘Erase existing entities’.
Before you leave from here you can also switch over to the Flow Path tab and define the Time of Concentration for the catchment. Here I’ve use a User-defined Tc of 5 minutes, as is many times the case with highly impervious commercial development projects such as this example.

Unless you are creating Tc flow paths at the time of creating the catchments, I’d recommend you use Edit Command Settings in the Settings tab of Toolspace to modify the default value of the Time of Concentration from TR-55 to User-defined, as shown below. This will save you several clicks throughout the workflow if you typically use user-defined Tc values.
That’s it! You now have a catchment tied to your structures, and the Catchment object data will transfer into HSS or SSA for final analysis of the storm system. If you’re exporting to HSS, I’d recommend you spend the time to define as much information in the Civil 3D model as possible that will transfer to HSS… don’t overlook the Inlet Location – On Sag or On Grade value in the structure properties.
Task 4: Prepare Standardized or Specialized Drawing Templates for a Fully Integrated Hydrology and Hydraulics Workflow

So what’s in a successful H&H template?

- Civil 3D Feature Settings
- Good starting Parts Lists
- Rule Sets
- Pipe and Structure Styles
  - Plan, Profile, and Crossing Pipes
- Pipe and Structure Label Styles
  - Label Expressions to perform calculations
- Pipe and Structure Table Styles
- H&H Parcel Styles
- Parcel Area Label Styles
  - Hydrologic watersheds
  - Hydraulic calculation catchments
- Parcel Area Table Styles
- Catchment Styles

Civil 3D Feature Settings

This is a very important part of successful workflows that often gets overlooked. This often stays unchanged from the OOTB defaults.
Utilize the Default Profile Label Placement settings to speed up labeling productivity.

Pay very close attention to the Storm Sewers Migration Defaults if you want to round-trip data with HSS or SSA. When Part Catalogs are updated, be sure to update these settings.
Parts Lists
Keep a thorough Storm Sewers Parts List in the company standard template. Discuss with users the importance of managing the Parts List and swapping parts in lieu of changing part parameters from within the model. Round Tripping between HSS and SSA does not work if the correct sizes are not in the Parts List.

It is highly recommended to have both Ex. and Prop. options in the default parts list, as many projects involve tying new systems into existing systems. Don’t worry about overcrowding a Parts List in the template; it’s much easier to delete part families and sizes than to add.

Rule Sets
Rule Sets can be as basic or as detailed as you’d like, depending on your typical workflow. At a minimum, it is recommended to have a min-max cover and min-max slope for each particular
pipe material in your standard parts list. If you want to get really in-depth, make a rule set for individual pipe sizes that sets min-max slopes based on the full flow velocities (e.g. 2 ft/sec to 20 ft/sec).

It is not recommended to use the Set Pipe End Location rules to set pipe ends to inside edges of structures for storm sewers. This will cause issues during the round-tripping process with HSS and SSA. It will also cause issues when creating alignments from network and connecting new pipes to structures with other pipes not connected at the center. You may use a label style for pipe length to inside edges of structures to correctly label to real-world lengths, but be sure to create an expression to label the pipe slope based on a calculation of IE up, IE down, and 2D length from inside edges of structure. The default pipe slope label will not do this, and slopes may not correctly match the pipe lengths and invert.

**Pipe and Structure Styles**

Have clear and concise structure styles for different structure part families. Due to the complexity of Part Builder, using user-defined blocks as plan and profile view representations of structures is many times the best answer.

Have styles for both existing and proposed for each type of pipe and structure style.

Single Line pipes are old-school. If your organization is still using single line, try to get buy-in to change the standard. It doesn’t paint a clear picture on the plan, and in the BIM world it’s no longer a time consumer or potential for missing with revisions to have double line pipes. If they’re small pipes on a very large scale plan… maybe.
Pipe and Structure Label Styles
Have clear and concise naming conventions for your standard label styles. It is recommended that Storm and Sanitary (and Water, Gas, Data, etc. if not yet using Pressure Networks) be clearly denoted in a label style name. Existing and Proposed styles are also recommended to easily denote which structures and pipes are existing and which are proposed.

Utilize design information label styles in models to assist in the design. As shown in the examples above, there may be an abundance of information that you wish to have in a particular label style that is used only during the design process, while other standard label styles will be used in plan production.

Let the labels read the BIM information as much as possible. Try to limit the amount of “dumb” text in a label style.

Keep in mind that the Label Entire Network in Plan or Profile can only utilize one pipe and one structure label style at a time. Try to find ways to optimize and limit the need to add individual part labels or to perform manual text overrides, such as the example above for having a single label style that can decide if a pipe is round or elliptical and label accordingly.

Remember that structure label styles are somewhat limited when it comes to inflowing and outflowing pipes. In the case of multiple inflow pipes or multiple outflow pipes, there is no way to override text for one pipe in a structure label without overriding the others… I just thought of a good Wish-list item.

Pipe and Structure Table Styles
Perhaps the most variable styles you will deal with are the pipe and structure table styles. Many reviewing agencies will have specific requirements for what is desired in a pipe and structure table. These tables not only serve to display final plan production information, but they can also help during the design process to have an overall look at the pipe networks since neither the Panorama view of pipes and structure nor Toolspace offer much flexibility in terms of what information is displayed.

Tables are limited when it comes to annotating varying part types. For example, there is currently no way to add a single column that will report either “Length x Width” or “Diameter” depending on the shape of the structure. It’s best to use the part Description to overcome these challenges for part sizes in a table. Also remember that any number of Optional Parameters may be added to your standard part catalog, which can all be reported in a table.

There are some things that may be done with logic expressions, such as the circular vs. elliptical pipe full flow decision expression example above, to control the values reported in a table.
Have an easy-to-understand naming convention for the table styles to point users in the right direction.

**H&H Parcel Styles**

Have defined styles for each type of watershed or catchment area delineation in your organization’s standard workflow inside of the H&H template. Discuss with the decision makers of your organization how catchment and watershed maps should look. You have several visual options using Parcel styles, such as boundary linetype, hatch or no hatch, hatch partial fill area or hatch full area, etc. As with all Civil 3D styles management, use clearly defined naming for styles to avoid confusion of users. Use the test, “If a new user starts work tomorrow and looks at the style names in a list, will he/she be able to figure out the appropriate style to use without asking someone else?” It is recommended that the Overall and Subshed naming convention coincide with the Sites naming convention. Remember to set the most commonly used style as the default style in the Feature Settings.
**H&H Parcel Area Label Styles**
The Parcel Area Label is where all of the information for a watershed/catchment area is displayed. Take some time with others in your organization to determine how you would like the labels to look for a final watershed map that will be delivered to a client or reviewing agency.

Have a clear and concise naming convention that coincides with the H&H parcel style. A _No Display style may be necessary to retain in the template for various cases stated previously.

**H&H Parcel Area Table Styles**
Like pipe network tables, these H&H area table styles may vary greatly depending on a particular client or reviewing agency.

Remember that any User Defined Property or Area Label Expression may be reported in a table.

**Catchment Object Styles**
Follow the same recommendations as H&H Parcel Styles. Although SCS hydrology workflow generally won’t require a Catchment Object workflow to link the information to storm structures, you may wish to have an SCS hydrology style in the template so that you can take advantage of the Time of Concentration Flow Path styles in Catchment Objects.

**Develop a standard workflow**
Learn about the standard workflows of engineers and designers in your organization before developing the H&H templates. This can drastically change your development of parts lists, rules, styles, expressions, and settings. Hold roundtable discussions early on in template development to achieve common goals and ideas for standardizing the templates and workflows.

Keep an open mind about varying workflows for varying uses. For instance the workflow presented here on creating Parcel catchment areas, then creating Catchment objects from those for HSS and SSA export is great for proposed land development; however for large-scale SCS hydrology it would be better to create Catchment objects from the GIS surface, explode those catchments, then convert those bounding areas to Parcels for to further watershed map creation.
Hold regular training sessions to get as many users on the same page with workflow as possible. This is very difficult with Civil 3D because there are so many different ways to achieve a single end goal, but it is important to the success of your organization to get all users performing the standard workflows in an organized and similar manner. The tools presented in this document are a major leap from the traditional method of performing H&H area take-offs with basic polylines and spreadsheets or making watershed maps with hatch patterns and “dumb” text. It is common in many engineering firms to need to pull in someone other than the original engineer / designer to make modifications to a design. Failure to provide the necessary training to all users may result in significant time loss to recreate work using alternative workflows; or even worse, having an inexperienced user corrupt a file or disorganize it beyond repair.

**Evolve Your Standards to the BIM Software; Remember You Can't Please Everyone**

Try your best to develop styles and standards that make everyone happy, but remember you can’t please everyone. There will always be that guy who wants the bottom of structure to appear at the outer bottom edge of the lowest connected pipe, no matter how many times you explain that it isn’t realistic without showing the floor thickness. There may also be that one engineer who wants the inverts called out on the bottom of the structure instead of on top with the other structure data. Hold roundtables and discuss these items with all parties, allow questions and suggestions, and work through these issues on the front end. Do whatever it takes to keep the CAD technicians from drawing dumb text and leaders just to make that guy happy.

We are now in year 11 of Civil 3D. Most of the old 2D Civil drafting standards should be gone by now, and the legacy standards and workflows should NEVER hold back advancement and the evolution of the industry.

**Good luck with your new BIM for Civil standards and workflow development. In closing, I leave you with my two favorite quotes; one of my own.**

*The only thing worse than training employees and losing them is to not train them and keep them!*

- Zig Ziglar

*In the world of business technology the opposite of evolution is not regression, but extinction.*

-WCS