



Diamond Sponsors



AUGIWorld

The Official Publication of Autodesk User Group International

March 2014

Tips, Tricks, Hints and Helpers

- Shocking Revit Family Secrets Revealed
- Conflict Resolution in Revit MEP
- Getting Attached with XREFs



BIM Library

ARCAT has the most comprehensive collection of BIM objects you will find, available free of charge and without registration.



The ARCAT BIM Library is also accessible in the ARCAT app, with editing capabilities in the AutoCAD 360 app.



ARCAT

arcat.com



facebook

contents

product focus



6

8 Revit Architecture: Shocking Family Secrets Revealed

16 Revit MEP: Conflict Resolution

18 Inventor: CNC Cut Files with Autodesk Product Design Suites, Part 1

24 AutoCAD: Getting Attached with XREFs

30 3ds Max: Drawing Back the Curtains



30

37 Revit: Getting Started with Revit Macros

42 AutoCAD Civil 3D: Won't You Please Help Me?

46 AutoCAD Architecture: Schedule Tables in ACA

50 Revit Structure: Tips, Tools & Workarounds



37

columns

4 Editor's Note

5 Inside Track

6 CAD Manager: Tips for Your Tricks



50

14 Technology in Practice, with CASE & HP



Cover image:

Cover image: 10 rue de Lancry 75010 Paris by NDCG Architect, 2014 – Daniel Hurtubise. Reuse of full or partial copyright image, in any form, without prior written permission is strictly prohibited. Visit Daniel's blog at <http://www.revitit.com>

Editor's Note



Hello, *AUGIWorld* Reader!

Welcome to another exciting issue of *AUGIWorld* magazine! We have a *lot* in store for you this month! Those who pay attention will notice a plumper magazine this month—56 pages! Now I don't expect this to be the norm, but we have been increasing the variety of content in AW and it really showed in the page count this issue. So let's get on with it!

First up I need to give our cover photographer, Daniel Hurtubise, props once again for a very interesting interior yet exterior building shot. Be sure to read the photo credits on the Table of Contents page for more information on Daniel and this particular image.

Moving on, our focus this month is on tips and tricks and who doesn't love those? But we do move things about in AW on occasion and this month the "Inside Track" column by Lee Ambrosius is right at the front! Lee brings our attention to new solutions by JTBWorld, Cadig, and Redbike Systems—check them out! Then the perennial CAD Manager, Mark Kiker, brings his own set of tips for your tricks in his monthly CM column.

Jetting on, we are next into Revit Architecture with Marcello Sgambelluri and his expose' on shocking family secrets! We then introduce a new column by CASE, "Tech Insights", with their own top ten Revit tricks. Then we have an article on conflict resolution in Revit MEP by Ed Hannabas. And Scott Moyse and Gavin Bath present Part 1 of a 2-part series on the use of CNC cut files with Autodesk products.

Some are really attached to Xrefs and Walt Sparling shows us the why and the how in AutoCAD! And Michael Beall always has a tip to share, this month on LAYER PREVIOUS. Next Joel Bradley draws back and lets the light in on some tips for curtains when using 3ds Max. Then guest author Michael Kilkelly shares how to get started with macros in Revit 2014.

Covering AutoCAD Civil 3D this month is Sam Lucido and he uncovers where to find help in learning C3D. Next we have Melinda Heavrin getting tabular in setting up schedule tables when using AutoCAD Architecture. And Scott Melching brings some bright ideas to his article about Revit Structure.

Okay that's it and that's a lot! So, what's next up? You! You need to turn the page to begin your study in tips and tricks in using Autodesk products!

Take care,

David Harrington

AUGIWorld Editor-in-Chief

Autodesk, AutoCAD, Autodesk Architectural Desktop, Autodesk Revit, Autodesk Building Systems, Autodesk Civil Design, Autodesk Inventor and DWF are either registered trademarks or trademarks of Autodesk, Inc. in the U.S.A. and/or in certain other countries. All other brand names, product names, or trademarks belong to their respective holders.

AUGIWorld

www.augiworld.com

Editors

Editor-in-Chief

David Harrington - david.harrington@augi.com

Copy Editor

Marilyn Law - marilyn.law@augi.com

Layout Editor

Debby Gwaltney - debby.gwaltney@augi.com

Content Managers

3ds Max - Brian Chapman

AutoCAD - Curt Moreno

AutoCAD Architecture - Melinda Heavrin

AutoCAD Civil 3D - Christopher Fugitt

AutoCAD MEP - William Campbell

CAD Manager - Mark Kiker

Column: Inside Track - Lee Ambrosius

Inventor - John Evans

Navisworks - Michael Smith

Product Review - Lonnie Cumpton

Revit Architecture - Jay Zallan

Revit MEP - Todd Shackelford

Revit Structure - Phil Russo

Advertising / Reprint Sales

Chris Porter - salesmanager@augi.com

Kate Morrical - salesdirector@augi.com

AUGI Management

President

David Harrington

Vice Presidents

R. Robert Bell

Desiree Mackey

Scott Wilcox

Treasurer

Donnia Tabor-Hanson

Secretary

Melanie Perry

AUGI Board of Directors

Shaun Bryant

Tommy Holder

Kate Morrical

Michael Smith

Walt Sparling

Scott Wilcox

Publication Information

AUGIWorld magazine is a benefit of specific AUGI membership plans. Direct magazine subscriptions are not available. Please visit <http://www.augi.com/account/register> to join or upgrade your membership to receive *AUGIWorld* magazine in print. To manage your AUGI membership and address, please visit <http://www.augi.com/account>. For all other magazine inquires please contact augiworld@augi.com

Published by:

AUGIWorld is published by Autodesk User Group International, Inc. AUGI makes no warranty for the use of its products and assumes no responsibility for any errors which may appear in this publication nor does it make a commitment to update the information contained herein.

AUGIWorld is Copyright ©2012 AUGI. No information in this magazine may be reproduced without expressed written permission from AUGI.

All registered trademarks and trademarks included in this magazine are held by their respective companies. Every attempt was made to include all trademarks and registered trademarks where indicated by their companies.

AUGIWorld (San Francisco, Calif.)
ISSN 2163-7547



SMARTPURGER 3.5



<http://bit.ly/1bld6Ly>

SmartPurger is batch drawing utility by JTB World that allows you to execute AutoLISP (LSP) and Script (SCR) files, and set purging and saving options that affect the drawing files as they are processed. You can download SmartPurger from the Autodesk Exchange Apps store.

“SmartPurger is a batch and script utility for processing multiple Autodesk® AutoCAD® drawings with full control. Execute LSP or SCR files on multiple drawings. SmartPurger handles crashes so the batch processing continues to the end and leaves bad drawings to be corrected manually. Many purge and cleanup settings available. Useful before backups and distribution of drawings. The file’s modified time stamp can be restored.

SmartPurger is useful if you want to minimize the size of files before a backup or before sending drawings by e-mail, burning to CD

or just to use as a quick viewer of drawings. You have options to save the files to a separate folder or to a different drawing version.

SmartPurger can execute custom script files (.SCR) or AutoLISP files (.LSP, .FAS or .VLX).”

AUTOPARAMETER



<http://bit.ly/1nLgZC1>

AutoParameter is an AutoCAD® add-on by Cadig Inc. that allows you to export and import constraint parameters between AutoCAD and Microsoft® Excel®.

“AutoParameter is designed to link the AutoCAD table cells or Excel worksheet cells to AutoCAD constraint parameters. Also, AutoParameter can export AutoCAD constraint parameters to AutoCAD tables or Excel spreadsheets. And you can import the constraint parameters from an Excel worksheet by using AutoParameter.”

The key features of AutoParameter are:

- ✦ Link the cells of an Excel spreadsheet to AutoCAD constraint parameters.
- ✦ Link the cells of an AutoCAD table to AutoCAD constraint parameters.
- ✦ Export AutoCAD constraint parameters to an AutoCAD table.
- ✦ Export AutoCAD constraint parameters to an Excel Workbook.
- ✦ Import constraint parameters from AutoCAD to Excel.
- ✦ Update the linked AutoCAD constraint parameters.

RDBK_RELOAD ADD-ON FOR REVIT



<http://bit.ly/1fvfNqm>

RDBK_ReLoad is an Autodesk® Revit® add-on by Redbike Systems Pty Ltd that allows you to reload families into a project. You can download RDBK_ReLoad from the Autodesk Exchange Apps store.

“Selectively reload Autodesk® Revit® families into an Autodesk Revit project. ReLoad displays a categorized list of all Autodesk® Revit® families in the project and all are checked for reload by default.

Alternatively, you can choose one or more families from the list.

- ✦ Check a category to reload all families of that category.
- ✦ Expand the category to select individual families.

ReLoad searches for selected families in the Options/File Locations/Places directory list. If a family is not found, you will be prompted to manually select its location.”

Inside
Track

**AUGIWorld
brings
you recent
developments
in Autodesk
and related
software items**

Tips for Your Tricks



Reading the articles in *AUGIWorld's* Tips and Tricks edition can overwhelm you. There are so many good things to glean from the top-notch writers who share their expertise in brief. The topics they cover provide real-world, time-saving information. You may be wondering just how to implement all of the good information and get it in the hands of your users.

Beyond AUGI publications, there are so many places to find ideas on improving performance and production in CAD that it can stagger the mind. What should you share and not share with your staff? What kinds of things should you pass along and what should you try to discourage your staff from using?

You cannot control everything that your office and users can access. You need to know how to approach the sharing and spread of information. You may even have to prevent the migration of tips that actually negate standards. You cannot prevent users from looking online, texting their buddies, posting on AUGI, calling someone where they used to work, or emailing a friend. So what are you supposed to do with the wide array of tips and tricks that come across the wires? How do you protect the files from being corrupted by some trick that someone read?

MAKE SURE YOUR STAFF KNOWS THE STANDARDS

The best defense may be just embedding the standards into your users. Verify that they know what is expected and that they understand the need to hit the target. The bottom line may not be that they follow some exact steps in creating their CAD and BIM files, but that the finished product matches the standards that are set. Having milestone check points along the way can help. Checking the files along the way will ensure that they do not stray too far from the standard.

HAVE STAFF SHARE WITH YOU FIRST

Developing an environment that places you well into the flow of information takes some time. You should nurture your users by interacting with them as much as possible so they feel comfortable sharing their ideas, tips, and tricks with you. Encourage them to come to you with tricks that they think are really effective. Let them know that they can share—and get credit for—their best ideas that you pass on to others. One of the ways you may shut down the flow of information is to not give credit to the person who brought the idea to you. Do that once or twice and people will stop the supply of good ideas flowing toward you. They will continue to share the tips, but you will not know what they are sharing.

You want users sharing their best ideas, but not the troubles that might arise from using them.

DEFINE WHAT SHOULD BE SHARED

As mentioned, you want to share the good stuff that reinforces and supports your standards, production, and processes. Weeding out those items can be a chore if you think you need to review every one of them. There usually is no need for that level of concern. Most of the time the tips and tricks are innocuous. Many are just shortcuts to the same ends that would take many steps. Ninety-five percent of them fall into three categories.

- Category One are the tips that work great and help you move forward.
- Category Two are the ones that work well, but are unusable. They don't really apply to what you are doing and are not useful to you.
- Category Three are the ones that are bad, don't work, make things worse, or even corrupt your files. I have seen the ratio of 2/6/2. Out of 10 tips, 2 will be great, 6 will be of no gain or will not work for you, and 2 will actually be bad for your environment. But the two great ones will be worth all the trouble of the other 8.

SHARE TIPS AND TRICKS, NOT TROUBLES

There are some things you want to watch out for. You want users sharing their best ideas, but not the troubles that might arise from using them—things that, if shared and replicated, can cause problems. Some of those may include tips that set layer names or define blocks. Some of them may change settings or create items in your files that are hard to detect. Things such as Wipeouts and Anonymous blocks should be avoided if they cause problems plotting. I would take a long look at suggestions that include inserting Excel files or database links.

There are many more areas that may cause you concern. Keep these in mind as you review what your users are sharing. When you find someone sharing tips that are not fit for your environment, let them know why. They need to know the story behind the restriction.

SHARE AND SHARE ALIKE

I encourage users to share tips. AUGI is built on sharing. Your firm should embrace sharing efforts also. Users like getting tips and distributing tricks. No one likes someone who hoards information. Collectively, your staff can do great things as long as the sharing is mutual, includes you, and watches out for those items that may cause users to stumble, files to fail, and projects to veer off into problems.



Mark Kiker has more than 25 years of hands-on experience with technology. He is fully versed in every area of management from deployment planning, installation, and configuration to training and strategic planning. As an internationally known speaker and writer, he is a returning speaker at Autodesk University since 1996.

Mark is currently serving as Director of IT for SIATech, a non-profit public charter high school focused on dropout recovery. He maintains two blog sites, www.caddmanager.com and www.bimmanager.com.

Shocking Family Secrets Revealed



INTRODUCTION

The following topics in this article are a culmination of years of secrets that I have used to build my Revit families. Most of it has been shrouded in secrecy. I have kept the secrets to myself, but I want to finally take this opportunity to share them with the Revit community. Read on and enjoy!

HOW TO CUT THE LONG SIDES OF BEAMS

How do you cut the long side of a beam using the cut command and a reference plane when Revit only allows you to cut the short side of the beam? This is illustrated in Figure 1.

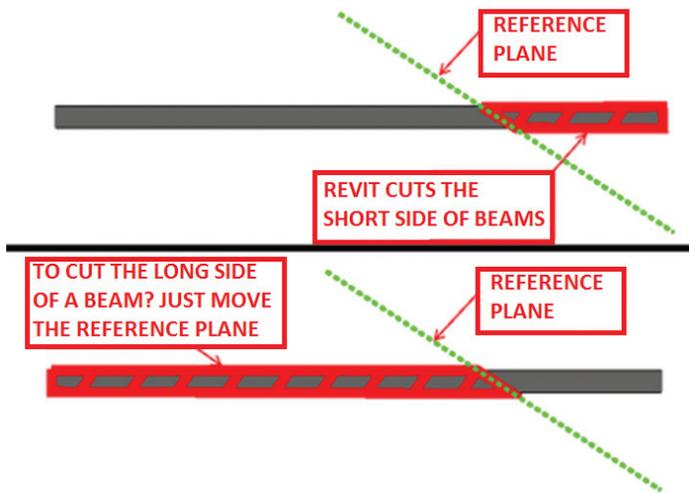


Figure 1: How to cut the long end of a beam

I have seen this question asked again and again on different Revit forum sites and I thought it was time to explain how I have done it in the past.

Follow the steps below and watch the scrolling image to learn how.

- Draw a reference plane at the far end of the long side of the beam to be cut.
- Cut the beam with the cut command.
- Move the reference plane to the desired location and presto! The long side is now cut.

Why did the Revit developers only allow users to cut the short side of the beam by a simple cut and pick command? I don't know, but I am sure the Revit developers had their reasons for making this the current workflow.

IMHO: The important thing to remember when solving Revit problems is to take matters into your own hands, and find ways to solve these problems yourself. Don't wait for the Revit developers to change the program. Instead, change your way of thinking about the program. Having a positive attitude has always helped me solve my Revit problems.

For example, when I solved this particular problem, I didn't get angry at the Revit developers for making this limitation. I thanked them for programming in Revit the ability to MOVE a reference plane, which ultimately helped me solve this problem. Thank you, Revit Developers!

THE JET ENGINE

The following will show how to make the shape of the jet engine. The jet engine was used as part of the 747 jumbo jet modeled in Revit shown in Figure 2.



Figure 2: The Revit 747

How do you make a 747 engine when the profiles that are selected result in the following form?

Well, the developers did not allow profile order to be selected when creating a form. So should you give up, or wait for the developers to add that feature? Right! Take matters into your own hands and create the form, then MOVE THE PROFILE! Follow the steps shown in Figure 3.

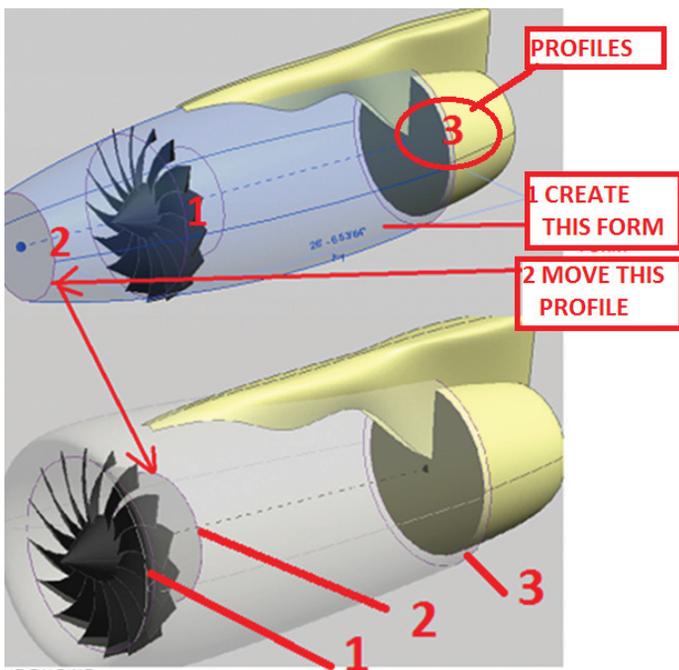


Figure 3: How to move profiles of forms

“RIDE THE RAIL” ROTATION METHOD

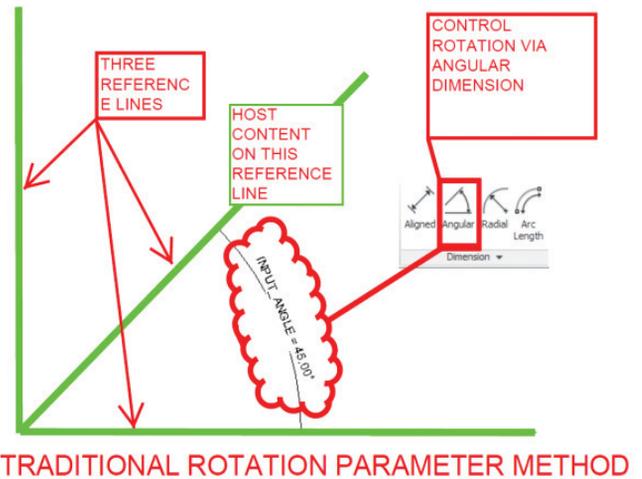


Figure 4: The traditional rotation rig

The traditional method, shown in Figure 4, uses the angular dimension command. It is not a secret that it has been known to sometimes break at a zero value and at other angles. Because of this, I have created an alternative rotation method called “Ride the Rail” for use in the mass and adaptive component family environment.

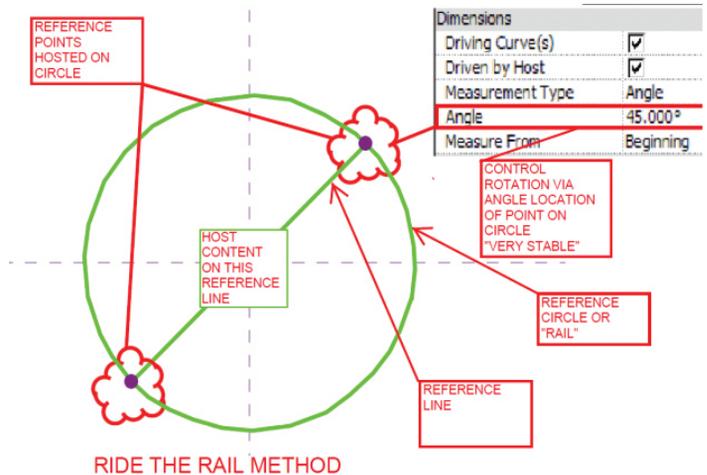


Figure 5: Ride the Rail rotation rig

As shown in Figure 5, the rotation rig, Ride the Rail method does not use the angular dimension. Instead, it uses the power of the reference point to “ride” the “rail” of the circle or curve to control the angle. The next section will show an application of the Ride the Rail rotation method.

INTRODUCTION TO INVERSE KINETICS

Let’s use what we just learned about the Ride the Rail rotation rig and apply it to a piece of construction equipment, shown in Figure 6 to animate the “arms.” This is the same concept as creating bones to drive the motion of character.



Figure 6: The Revit lift family

Draw a reference circle whose center is on the first center of rotation. Then draw a circle where the radius is as large as the next closest rotation point as shown in Figure 7.

Continue to draw circles at each node as shown in the previous step until all the joint locations have nodes with reference circles as shown in Figure 7.

- To add an angle parameter to each swing rig, select the hosted node on the circle and change the measurement type to angle and assign or create an angle parameter.
- To add length parameter, do not use an aligned dimension. Take advantage of the rotational rigs that were just created. Since each segment of “arm” spans from hosted node to hosted node, then to make the “arm” longer just make the radius of the circle a parameter.

To make the bones in Revit, simply create a circle on each successive rotation point and make the element span between these points. This will result in a series of hosted Ride the Rail rotation rigs.

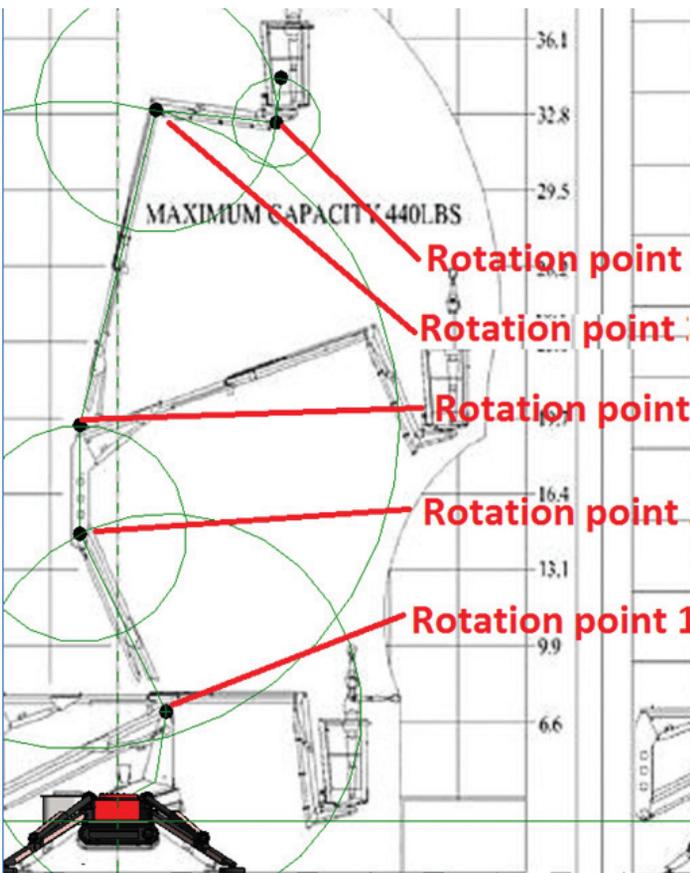


Figure 7: Hosted rotation rigs

Because this is a piece of construction equipment, there will need to be some limits on the parameters that were just assigned. These booms are not allowed to move past a certain point as spelled out in the equipment specifications. Therefore, it is important to add some sort of limitations to the parameters. The fastest way to do this is simply “cut” the circle so that it results in an arc that starts and stops at the limits of the equipment as shown in Figure 8.

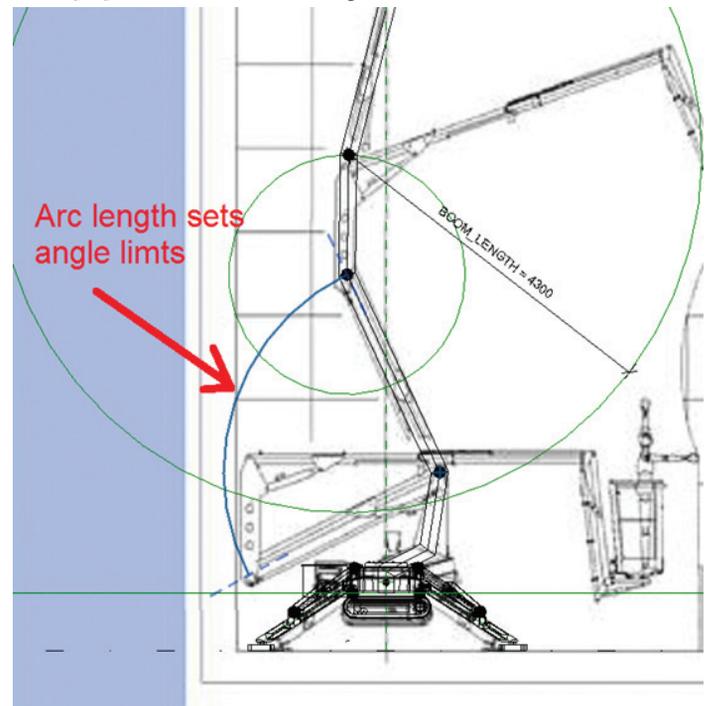
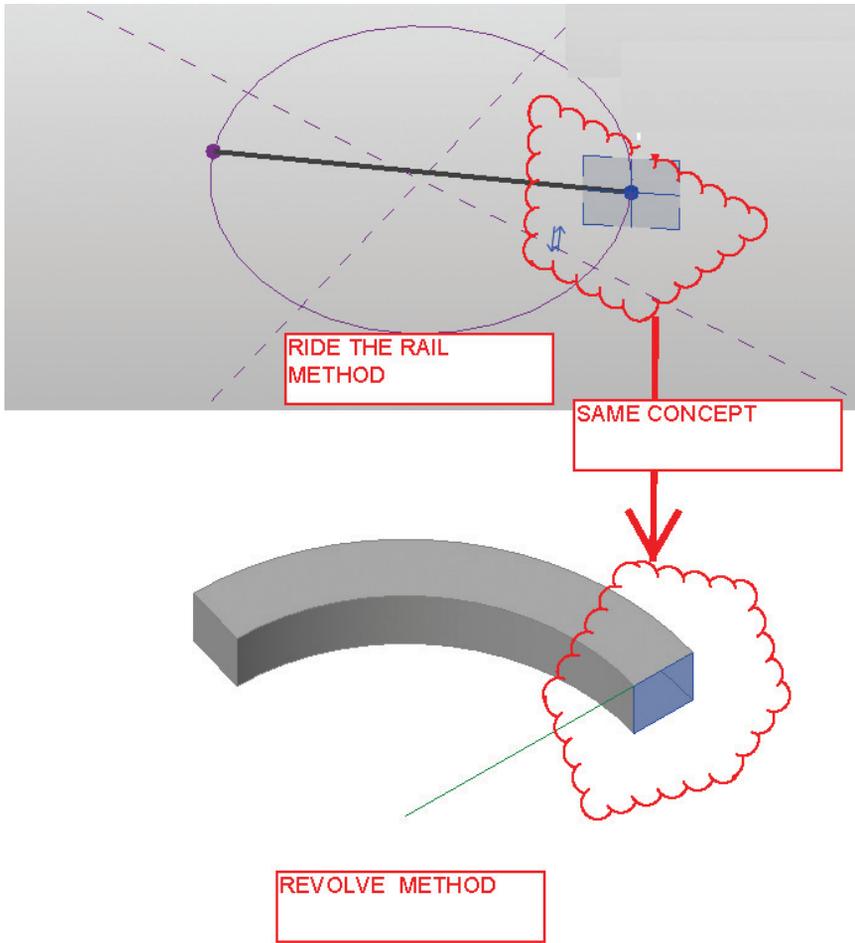


Figure 8: Limiting the rotation of the rig



This fundamental concept made me realize that the Revolve element could have a constant angle value and have its start and end angles move together. Also, you are able to define the end of an open revolve element as a work plane. Yes!

From this I created the “Revolve” rotation method as shown in Figure 10. Note that it uses the preset revolve form parameters of the start and end angles to control the angle—NOT the angle dimension.

THE REVIT PUMPKIN

In this section we will learn how to model the Revit Pumpkin as shown in Figure 11. I built the Revit Pumpkin as an entry in a pumpkin modeling contest (<http://buildz.blogspot.com/2011/10/happy-halloween-iii.html>) and I won first place for scariest pumpkin!

- Open up a new adaptive component family. Insert an image of the side view of a face in the left view of the elevations. Sketch in profile 1 first using the spline thru points command. It should look similar to Figure 12. I chose not to follow the shape of the face exactly since I was already using the “approximate” method. Also, the face was just a guide.

Figure 9: Ride the Rail versus Revolve Rotation method

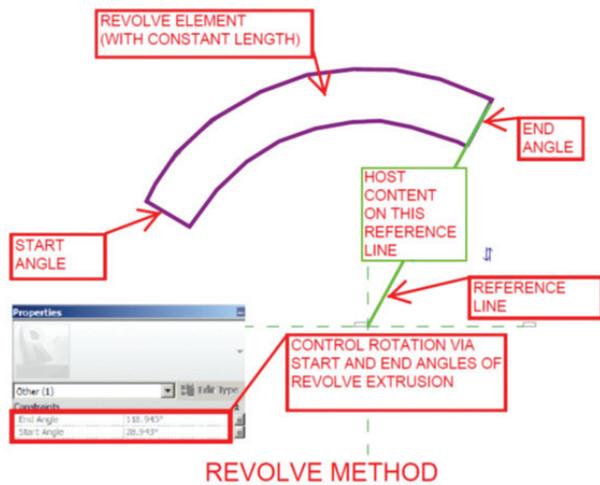


Figure 10: Revolve Rotation method

“REVOLVE” ROTATION METHOD

The Ride the Rail method has one drawback: It cannot be used in the traditional family environment. However, when you apply the principle of the Ride the Rail method to a “Revolve” in the traditional family editor as shown in Figure 9, you get the same result.



Figure 11: The Revit Pumpkin

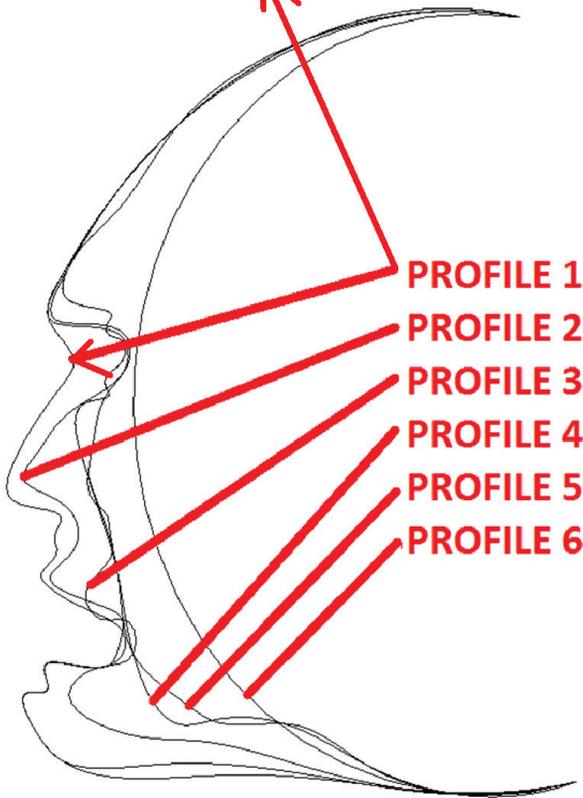
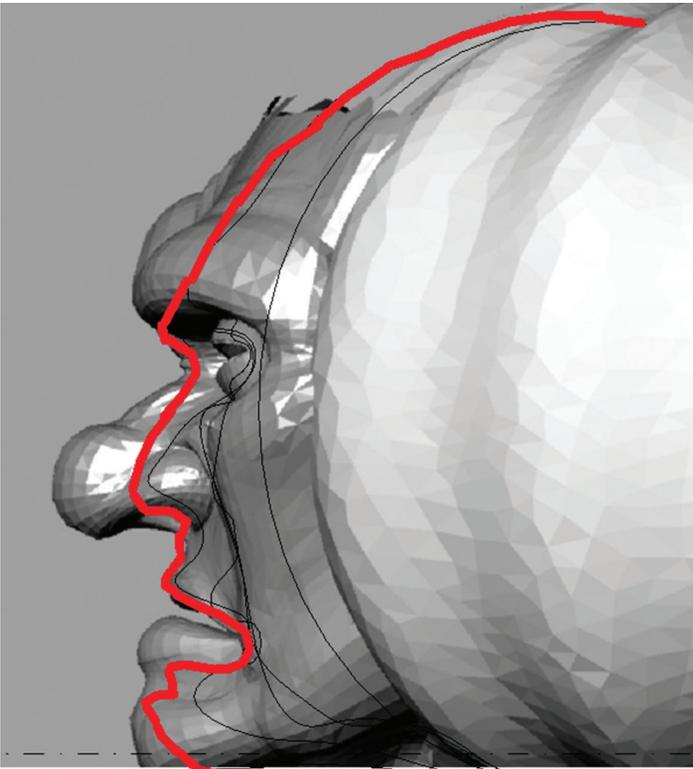


Figure 12: The profiles

- Create the rest of the profiles 2, 3, 4, 5, and 6 in the same view using the previous method.
- Create reference lines in a radial pattern and place the profiles with the dimensions shown. The profiles will be located at 5

degree intervals except profiles 3 and 4, where they are located 2.5 degrees off of profile 2.

Copy profile 1 and paste it into the new adaptive family. Since the profiles were built in the “left” elevation view, then all the profiles will need to be separately pasted to the plan view and “rotated” into place.

Copy profile 2 and paste it into the Revit family. Repeat this process until all the profiles are created. Go back to the plan view and copy and mirror all six profiles.

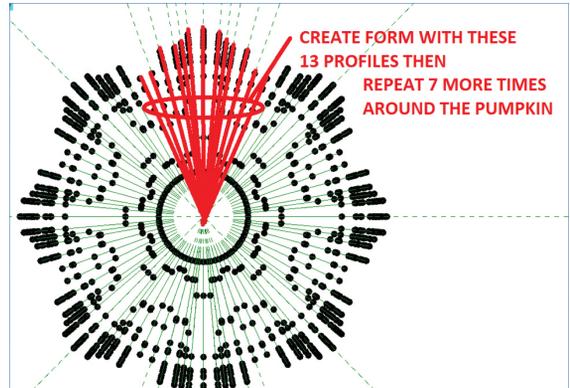


Figure 13: The plan view of profiles

- Select the 13 splines shown in Figure 13 and click “create form.” The final model should look similar to Figure 11.

THE REVIT COW

You will be shown how to model The Revit Cow, shown in Figure 14. This example illustrates how I used Revit and other software to create the Revit Cow. However, this does not mean that you have to choose the software I was using. What is important is that you follow the fundamental steps that are involved no matter what software you are using.



Figure 14: The Revit Cow

Instead of creating the cow model from scratch, which was how I modeled the Revit Cow originally, just take pictures of a cow out on the farm and load it into 123 D Catch, which is free from Autodesk and make your own cow model using the .obj extension.

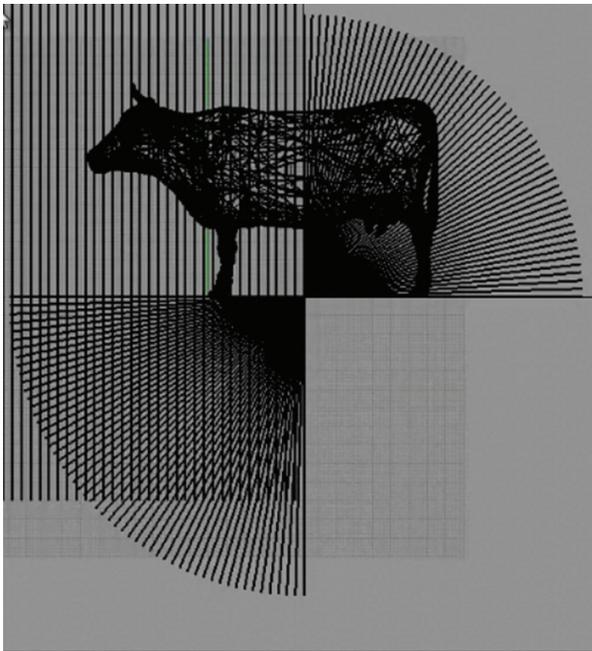


Figure 15: Slicing up the profiles of the cow

Open the .obj file in Autodesk® 3ds Max® and delete any extra mesh or vertices. Save the model and export the file as a dwg.

Open Rhino and import the exported cow from Max as a dwg format into Rhino. Select the dwg (mesh) and use the “mesh to nurbs” com-

mand. Draw a plane that cuts the cow and go to a side view and replicate by array the planes as shown in Figure 15. Select the planes and cow and use the “intersect” command. This will create all the profiles necessary to make a form in Revit. The profiles are radial from the back feet to the middle of the cow as shown in Figure 15. This was set up so that when the form is created in Revit there is less chance of it being successful because there are “smooth” transitions between each adjacent profile.

Note: You could alternatively use 3ds Max to create these profiles by using the spline intersect function.

Once the cow profiles are imported into Revit, simply select the spline thru points command and click the ends of each straight line segment. Repeat for each profile until all the profiles are completed as shown in Figure 16. With all the reference lines selected, click the “Create Form” button.

This is only one half of the cow. If you want to make the other half simply mirror the forms at the centerline of the cow and it will create two halves that make a whole cow. Now that you have seen how all these families were created by revealing their secrets, go out and make some shocking families.

So that does it for this installment of Shocking Family Secrets Revealed. Good luck!

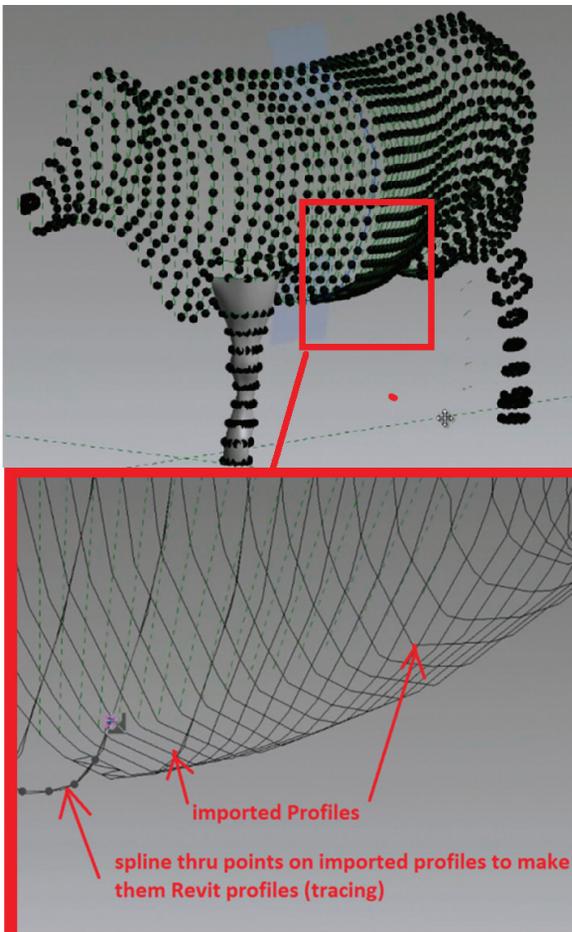


Figure 16: The Revit Cow profiles and nodal points



Marcello Sgambelluri is the BIM Director at John A. Martin & Associates Structural Engineers in Los Angeles, California. He has been using Autodesk products for more than 15 years including AutoCAD, 3ds Max, and Revit. Marcello is heavily devoted to helping advance the use and knowledge of BIM solutions within the AEC community. He is well known for modeling elements and creating workflows that others have thought not possible. He also frequently presents at Autodesk University and at The Revit Technology Conference. He has worked on many well-known projects in the past including the Walt Disney Concert Hall in Los Angeles, CA, the Stata Center at MIT, and the Tom Bradley International Terminal Expansion at LAX. Marcello received B.S. and M.S. degrees in Civil Engineering and is a licensed Civil and Structural Engineer. Marcello can be reached via email: marcellojs@hotmail.com, Twitter: [@marcellosgamb](https://twitter.com/marcellosgamb), or at his blog site: www.therevitcomplex.blogspot.com.

Top Ten Revit Tricks, by CASE

case

In an ongoing effort to improve the building industry, CASE and HP have partnered again—and this time it's to bring you a monthly column focused on getting the best from your Autodesk software. This month you will hear from CASE's most adept Revit experts—the people other

AECO (Architecture, Engineering, Construction, and Owner-operators) companies call upon when they are having a problem with Revit, when they need Revit training, or when they need a custom Revit plugin built. We have asked these experts to share their best tips for using Revit.

ONE: KEYBOARD SHORTCUTS

Caity Jimenez spends a lot of time teaching and using Revit in her role as a building information specialist. She says that setting up, learning, and using keyboard shortcuts saves her an immense amount of time and clicks. One of her most heavily used shortcuts activates and deactivates viewports, a task she does often when laying out sheets. In Revit 2014, viewports can be activated by double-clicking, but there is no fast way to deactivate them. The keyboard shortcut preferences can be accessed by typing KS. Search for “viewport” and assign a shortcut. Caity uses XX and XC because they are normally the closest keys to her hands, but you can assign whatever works for you.

TWO: PREVENT ACCIDENTAL CLICKS

Seth Edwards is moments away from leaving New York to work in our London office, but before he departed he left this tip for those who accidentally and consistently double-click on component families (especially room and ceiling tags) only to have it open up the family. You can change the double-click behavior by going to the Options menu, selecting User Interface, and under the Configure menu changing the double-click options. For “family” set it to “do nothing” on double-click.

THREE: LEARN A NEW WORKFLOW

The workflow section on AEC-Apps has been getting some serious action. Jump over there to see recommended workflows for updating Revit models or linking Revit to Grasshopper. You also can add your own workflow to the mix if you have a process you would like to share with the community (<https://aec-apps.com/workflows>).

FOUR: TEMPORARILY MAKING TEXT DIMENSIONS BIGGER

Temporary text dimensions can appear unworkably small—and they appear to get smaller as you zoom in on them. If these tiny dimensions are taunting you, there is a setting to enlarge the text globally throughout Revit. Caity Jimenez explains that from the Application menu, choose Options and then the Graphics tab. In the Temporary Dimension Text Appearance area at the bottom, choose the font size you prefer. The default is 8, but something like 12 is much easier to work with.





FIVE: REVIT PLUGINS

Don Rudder, Revit API expert and associate director of desktop applications, has built a library of 24 plugins for Revit. They do everything from quickly changing line styles to linking Revit with Excel. Don is constantly adding to this library based on user requests. So if there is some important functionality missing from Revit, Don can probably add it for you. Hop over to <http://www.apps.case-inc.com/> and pick up one of the free Revit plugins.

SIX: LOCKING GEOMETRY FACES

Tim Dumatrait is a building information specialist providing BIM support to clients. In the search for evermore efficient and robust families, Tim recommends locking geometry faces to work-planes when creating families rather than locking them to their profile sketches.

SEVEN: BIM COLLABORATION FORMAT

The BIM Collaboration Format (BCF) is a new XML schema currently making its way towards official adoption by buildingSMART. The schema enables other software packages to mark the locations of problems within a model. Typically these problems would be identified by sending a model to an analysis package that would then return a series of images (or perhaps the entire model) with the clashes identified. Now the analysis software can return a lightweight file transporting users to the problem and describing the issue within their preferred software. Matteo Cominetti, CASE's resident BCF expert, is working hard to develop a BCF plugin for Revit (<http://matteocominetti.com/bcf-revit-plugin/>), and Navisworks (<http://apps.case-inc.com/content/free-bcf-exporter-navisworks-manage-2014>). Try it out the next time you are analyzing your model in Solibri, Tekla, or the like.

EIGHT: BEWARE OF THE FREE LUNCH

Jose Torres recently joined CASE as a senior software engineer. He told us of a time when he downloaded a model of a toilet and tried to link it to the plumbing system of his model. After many failed attempts at connecting a pipe to the tank, he realized the toilet had a slightly skewed connection going into the tank when Revit would only make connections between parallel objects. Based on this experience and others, Jose thinks it's best to tread carefully when downloading free content, particularly for MEP systems.

NINE: CHECK YOUR ROOM HEIGHTS

Rooms are essential for scheduling, especially in a multi-discipline team. Unfortunately rooms created in Revit often don't go full height. Make sure they are the correct height by turning on a cut section and looking for the room fill. Otherwise you can click on the room and set its height in the properties. This ensures all fixtures are correctly assigned to the room and that the room is the right size for MEP calculations.

TEN: A PROPERLY CONFIGURED AND CERTIFIED WORKSTATION

A certified workstation is one of the most important Revit investments you can make. Hardware that is not Autodesk certified may not adequately support advanced product features, and in some cases, may result in problems. Autodesk and HP work closely together to proactively identify and resolve compatibility and performance issues, and certify HP Z Workstations with Intel® Xeon® processors for Autodesk software. HP and Autodesk publish a list of certified workstations (<http://h20331.www2.hp.com/hpsub/us/en/mcad-new-products-1.html>), and HP and AUGI have collaborated on special Revit configurations and pricing for AUGI members (http://h71016.www7.hp.com/html/affiliate/augi.asp?jumpid=va_r2910_go/AUGI/affiliate/071813/taylorlg#tab_3).

ABOUT HP

HP helps you stay ahead of the curve with professional desktop and mobile workstations designed for large and complex datasets, dispersed teams, and tight deadlines. HP Z Workstations with Intel® Xeon® processors deliver the innovation, high performance, expandability, and extreme reliability you need to deliver your 3D CAD projects in less time. For more information, visit the HP Workstations and Autodesk page on the HP website (hp.com/go/autodesk).

ABOUT CASE

CASE exists where building and technology intersect. We combine our experience as architects, engineers, project managers, software developers, and educators with a passion for technology to improve the way buildings are designed, realized, and operated. CASE is a building information modeling (BIM) and integrated-practice consultancy. We provide strategic advising to building design professionals, contractors, and owners seeking to supplant traditional project delivery methods through technology-driven process innovation.



© 2014 Hewlett-Packard Development Company, L.P. The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein. Intel, the Intel Logo, Xeon, Xeon Inside, Intel Core, Core Inside, Intel vPro, and vPro Inside are trademarks of Intel Corporation in the U.S. and/or other countries. All other trademarks are the property of their respective owners. Autodesk is a registered trademark of Autodesk, Inc., in the U.S. and other countries. Screen images courtesy of Autodesk.

Conflict Resolution



This is not an article to help you resolve the conflict with your coworker about who should get the cube with a window. I work as a consultant for Autodesk and I have witnessed a growing trend of “zero conflict models” or “all objects shall be modeled that are larger than one inch” in many contracts for BIM projects. This sounds like a great idea as it will lead to more efficient construction and lower costs in the process. This article is about trying to separate “what is commonly desired” against what is really practical. Architects and structural engineers usually do not object too much and then the MEP engineer comes onboard and messes everything up. So what can the MEP engineer do to address this issue and still be competitive in the marketplace?

In the past year, I have worked with numerous MEP firms that are struggling with satisfying these requirements and still keeping their fees competitive and meeting project deadlines. I work mostly with our Autodesk® Revit® MEP platform for design and Autodesk Navisworks® Manage for collision reporting. However, the methods I will discuss could easily be applied to other software packages.

The majority of engineering firms have based their fees on creating design intent documents, not design build. With the increasing demand for “zero conflict models,” these firms are being tasked with producing something equivalent to “shop drawings.” Even design build firms are being asked to produce more detailed models. This is especially true for the plumbing, piping, fire protection, and electrical disciplines. Engineers are more accustomed to producing diagrammatic documents even though they are creating a 3D model as part of the deliverable.

One of the best examples of this dilemma was a recent conversation with a fire protection engineer on a project. He was supposed to model the entire fire protection system, including the primary routing design. In the past, the secondary routing was typically done by the contractor who actually installed the system. Another example was an MEP firm that was facing increasing criticism over the number of conflicts that were being found in its models. Were these conflicts real or just part of the construction coordination process?

There are several approaches to resolving these problems and still meeting the intent of the contract. The first idea revolves around

clearly identifying, with the other disciplines, what is classified as a conflict. The second idea revolves around creating zones in the model for each discipline. The third idea is probably the most difficult: Having a different standard for conflict resolution in MEP.

Now let’s look at the obstacles that the MEP engineer or designer faces. Pipes and conduit are not typically placed in their actual location in order to produce a clear set of construction documents, which is what is typically used for construction. Ducts and pipes go through walls; pipes and conduit are inside walls; and very few firms place dimensions to the center line of ducts, pipes, and conduit on floor plans.

More and more firms are, in fact, placing dimensions to these objects in their floor plans as part of their documentation process. The next dilemma: The bottom of pipe or duct tag does not report the correct elevation if insulation is added to these objects in Revit MEP. As a result, there is a duct or pipe that is not correctly identified or placed and now when we go to the coordination meeting, this is identified as a conflict. It is just not one, but sometimes thousands. There are several ways to approach this issue.

The first approach involves creating parameters that identify pipes or ducts by their placement. Using Selection Sets within Autodesk Navisworks Manage, you can correctly identify these objects. These items would not be included in collision tests. You can create an instance Shared Parameter or Project Parameter that can be added into runs or systems by using the Tab select method in Revit MEP. At first this may appear like a lot of extra work, but it is not near the amount of work that is involved in going through several thousand conflicts and identifying which ones are true conflicts. By identifying each run by this parameter, countless conflicts are eliminated. These parameters should be clearly identified to the coordinating authority so they know what Selection Sets to create. The instance parameter could be named something like “disposition” or “coordination” and then assigned a value either as a letter or a number.

This method does involve some negotiation and coordination with the other project members, but is a more common sense approach to the underlying problem. The architect and structural team members do not typically have these issues and do not always under-

stand the workflow of the MEP engineer. There have been many other approaches to resolve this issue, but most involve much more work. Some firms have resorted to maintaining two models—one for plan production and one for collision detection. This method requires much more coordination and work. Another approach has been to place pipes in the actual location, which often results in a poor presentation in the plan view, but this is supplemented with a 3D section to clarify their actual placement. This does help with the coordination, but the plan drawings are not acceptable to other participants.

I prefer the second method, which is the creation of zones for the placement of pipes and conduit. These zones can be created using a generic line-based family template. It needs to have both width and height parameters and it is necessary to create reference planes or levels in the model. These zones are placed in the areas the pipes or conduit will occupy, but this method does not require the exact placement of these items. When your model is submitted for the coordination effort, collision tests are conducted using these zones rather than the actual pipe or conduits.

This method is good for most projects unless there are a great number of elevation changes in the architectural model. One of the advantages to this method is that these zones can be placed by technicians or designers that do not have the experience to design actual systems. The designer or engineer can now design pipe and conduit systems within these boundaries established by the zones and also create a correct presentation in the plan view.

The last method is more about negotiation and coordination—allowing the MEP disciplines to model to a different standard concerning the size of modeled objects. Often the other disciplines do not understand the extra effort that is required for the MEP engineer to meet the one inch or larger modeling requirement. I have worked on several projects where we had these discussions to educate other disciplines on the effort involved to meet this requirement and the lack of real value to the overall coordination of the model.

Engineers spend a large portion of their time correcting small pipes and conduit that in reality would have been adjusted in the field as a normal part of the construction process. A one-inch piece of pipe or conduit that hits a 12-inch beam is typically a pretty easy conflict to resolve in the field. This method revolves around education and honest discussions about the amount of effort involved and the return on this investment. This method typically requires a close working relationship with the other disciplines and is one that requires the greatest amount of work up front by the MEP engineer.

I have worked on projects where the tolerance was to 1/16th of an inch due to the large number of small pipes and conduit and the number of each involved. This is not a typical project for most firms and this level of coordination is not a typical situation. Revit MEP is being used on so many different types of projects that Autodesk never envisioned for the software that it requires users to develop processes to fit their own unique needs.

The next issue revolves around the final disposition of the model and what its use will be. Design Build projects by their nature require a greater effort by all disciplines, but this is generally understood when bidding the project. Design Bid projects should not involve the same amount of effort as these projects. The present course is blurring these lines. This again revolves around education, both internally and externally, about the actual needs for the project. Many of the senior MEP engineers that negotiate contracts are not aware of what is being asked of them.

I am not advocating a relaxing of standards for coordination, but rather a more common sense approach to this issue. MEP firms are being asked to deliver more and more for the same fees, often because they have not addressed the issues above. I have been asked repeatedly by MEP firms how to address these concerns, and typically the solution has been to ask the right questions and educate all parties on what they are really trying to accomplish. Walls, doors, windows, beams, and trusses are placed with precision with little doubt to their size and placement in the model. The vast majority of MEP objects are placed with less precision and this is often governed by local codes for their final disposition. Every project is unique and it is difficult to find one solution that works for every situation.

There are many other smaller items that can help with this process in the creation of templates and content that can also help in streamlining the MEP design process. I just don't have the space here to address all the issues. The real solutions revolve around developing a process for a sound understanding internally and then delivering that message to the other disciplines. In the vast majority of MEP firms I have worked with over the past eight years, the senior engineers have not been aware of the amount of effort that is required to meet these increasing standards on their work.

The real purpose of this article is to start the discussion within MEP firms on how they can best address the issues. The many disciplines involved and the variety of projects makes it difficult to find one solution to fit all situations. That does not mean a solution can't be found, but rather that solution does not revolve totally around the software being used, but instead on developing a process to use the software more effectively.



As a business analyst for Autodesk, Ed Hannabas spends his time assisting clients in the utilization of a variety of Autodesk software. He focuses on assisting firms in the adoption and use of Revit and AutoCAD-based software on projects ranging from office buildings and hotels to waste water treatment plants and hospitals. Crossing Architectural, Structural, and MEP disciplines, Ed's duties include training, project set up, and hands-on work, along with the development of standards and workflows for utilizing Autodesk software. Contact Ed by email at ed.hannabas@autodesk.com.

CNC Cut Files with Autodesk Product Design Suites, Part 1

INTRODUCTION

his handout was written as a supplement to the Autodesk University 2013 class given by Scott Moyse and Gavin Bath. They were kind enough to permit it to be formatted in two parts and published here. Some of the information was covered in more or less detail to suit that limited class format.

Part 1 will begin coverage of the various methods of producing 2D CNC cut files or tool paths within Autodesk Inventor®.

In an upcoming issue, Part 2 will discuss using iPropWiz to configure your Inventor design environment and benefit you in the long run. We will also introduce the concept of using Autodesk® Vault to manage your design data, and ultimately your manufacturing data, as well as some methods of presenting that to CAM programmers and staff on the shop floor.

CONTENTS

Introduction

2D CNC Cut Files

Flat Pattern Export

Flat Pattern Export Add-In Example

Export Face

Export Sketch As

Create a Drawing

Setting up a Sketch Text Style

Credits

2D CNC CUT FILES

I know of five distinct ways to get what CNC programmers need out of Inventor. The best method for you depends entirely on what you are cutting and the software you are using to program it. In the past I've had to resort to some fairly lengthy procedures to get what was needed. With building or yacht interiors, the difficulty comes in dealing with the grain direction of the veneers or laminates on parts and subsequent grain matching. As a result I have investigated lots of options and discovered a few tricks along the way.

Flat Pattern Export

As you may well know, in an Inventor sheet metal part you can create flat patterns. Once a flat pattern has been created, you can save out a DWG/DXF to AutoCAD®, which will allow you to develop the cut file further if need be.

There is some scope to run a macro in AutoCAD to do some clever post-processing on the flat patterns to streamline the CAD to CAM process. For a lot of people this would mean getting the part number into the cut file.

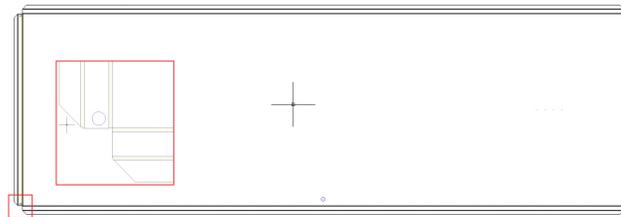


Figure 3: The result of a flat pattern export automates the assignment of layers for some of the model features

Pros:

- + Merge profiles into closed polylines.
- + Spline conversion to polylines.
- + Object recognition: inside and outside, and features on the upper and lower faces.
- + Rebase geometry to 1st quadrant.
- + To a certain extent, ability to control the layers.
- + Semi-automated length, width, and thickness values for BOM and parts lists.

Cons:

- + No support for exporting text with the cut file/flat pattern!
- + No feature recognition, so there's no method to include feature depths in the 2D data.
- + The process has to be carried out manually on each part, so it's the same repetitive process over and over again.
- + Sheet metal thickness won't link through to body extrusion value easily when using multi-body modeling techniques.

Flat Pattern Export Add-In Example

Here is an example of how automating Inventor can make the process of creating sheet metal flat patterns a breeze.

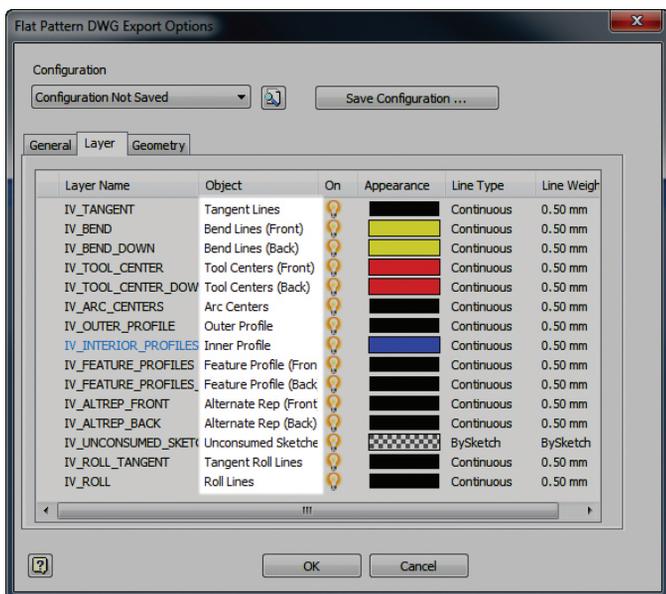


Figure 1: Layer mapping options for flat pattern export

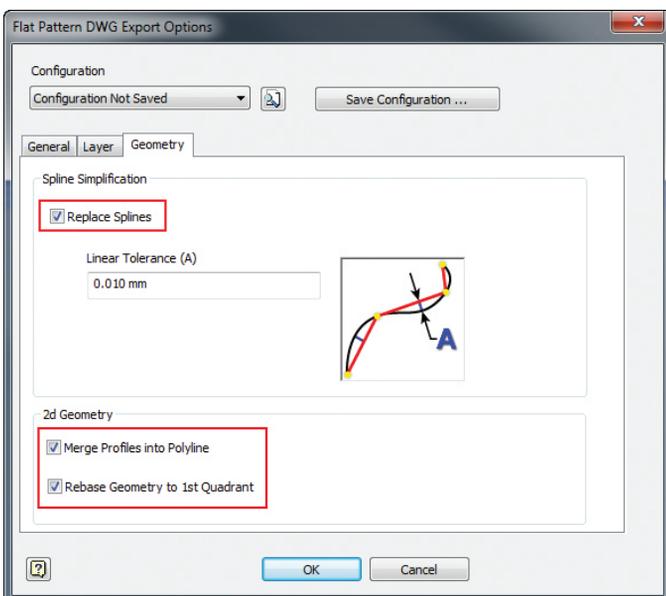


Figure 2: Geometry clean up settings for flat pattern export

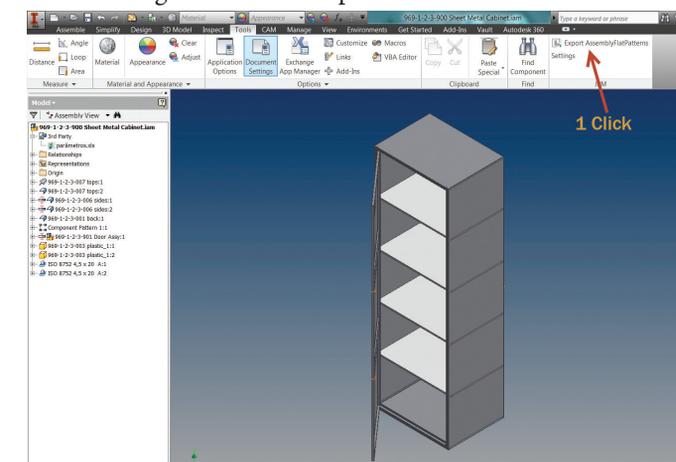


Figure 4: Export assembly flat pattern

Inventor 2014

For many companies, the creation of 2D data for profile cutting can be a hugely repetitive, manual task that costs significant time. We tend to find that the requirements for exactly how the data is presented varies a lot. For this reason, it's difficult to come up with a "one size fits all" approach for automating this task. This particular example is for a manufacturer who requires a table of information in a cut kit as well as 2D DWG files for each component. The tool runs as an Inventor add-in and works in the assembly environment.

With an assembly file loaded, the command is executed and the routine runs through every sheet metal component in the assembly, creates its flat pattern (if it doesn't already exist), and then exports the flat pattern as a DWG.

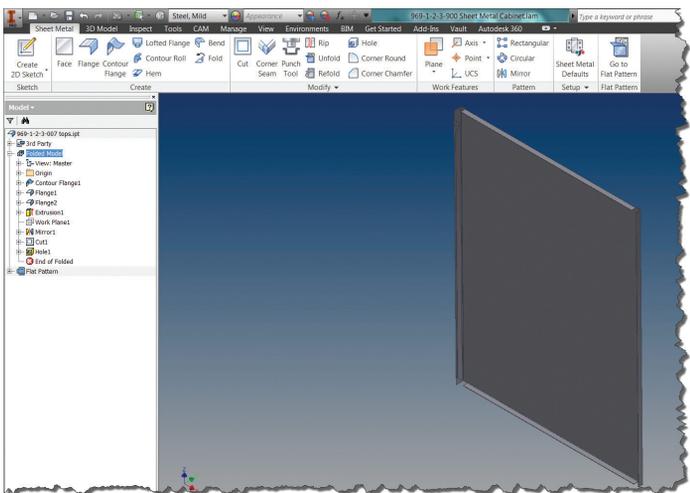


Figure 5: Flat patterns created automatically for all sheet metal components

Once the flat patterns have been created, an email is automatically generated with a HTML table that contains thumbnail images of each component as well as dimensional information, material, and assembly quantities. Additionally, the DWGs are attached.

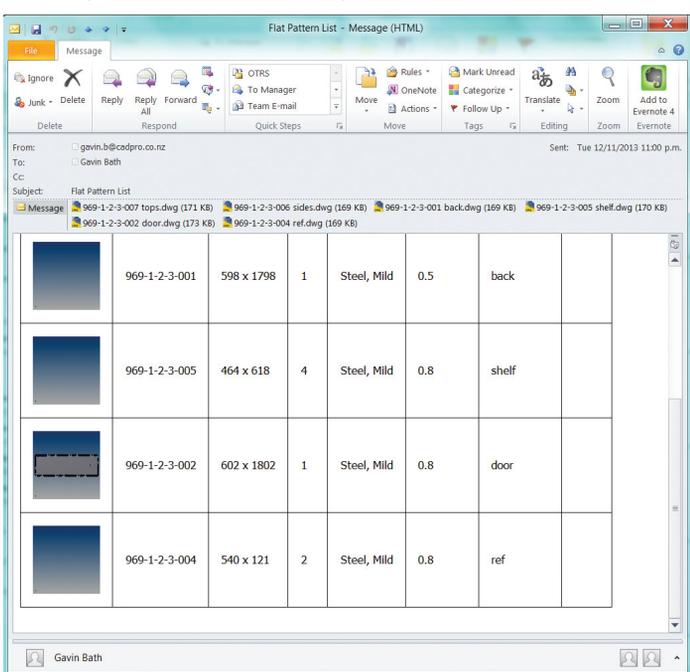


Figure 6: Example email generated by tool

This customization can be very easily edited to manipulate the data that is output; for example, changing from DWG to DXF file format.

Pros:

- ✦ Provides confidence in accuracy of quantities and dimensional information.
- ✦ Really fast and very efficient.
- ✦ Email trail provides job ordering history.
- ✦ Not easy to fudge quantities, add spares, etc. without modifying assembly.

Cons:

- ✦ Not easy to override quantities, add spares, etc. without modifying assembly.
- ✦ Reliance on programmer to edit customization if requirements change.
- ✦ Doesn't handle non-sheet metal parts (unless identified somehow).
- ✦ Doesn't handle grain direction on grain-sensitive materials.

Export Face

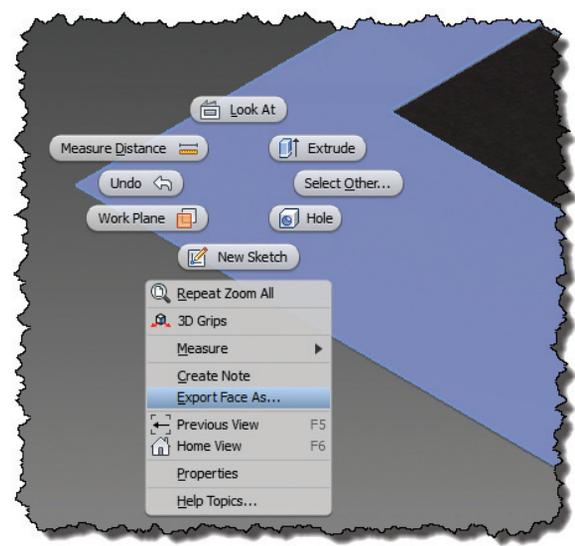


Figure 7: Export face radial menu command

I have a couple of videos showing you how to export a face of your part and a sketch in your part as a DXF/DWG file. People often use it for its simplicity—you can quickly knock up a 2D cut file and fire it off to your CAM programmer.

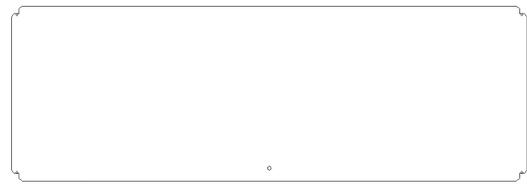


Figure 8: Export face captures the face outline nicely, but it's missing a lot of 'manufacturing' information

Export Face Pros:

- It's quick and simple.
- There are some AutoCAD export options for objects and file type.
- The result is closed polylines.

Export Face Cons:

- Zero layer mapping options like there are in the flat pattern tool.
- It doesn't capture geometry of the entire part, such as rebates, bevels, and hidden features, etc.
- No ability to include text in the export for engraving.
- No ability to color the features by type for export.

Export Sketch As

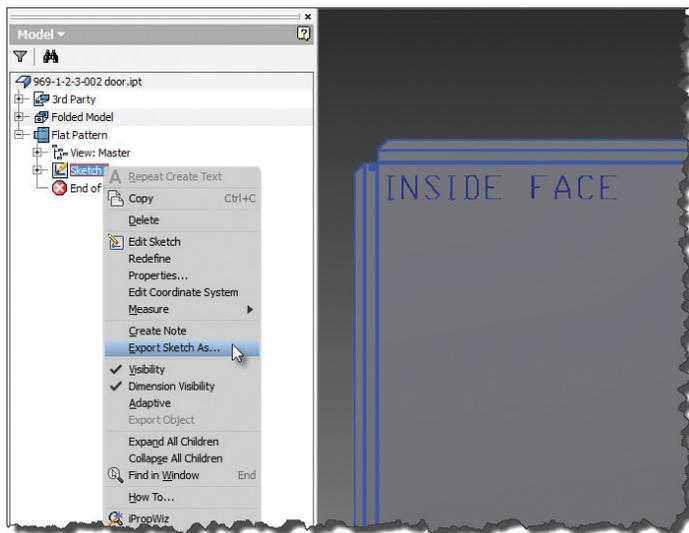


Figure 9: Export Sketch As allows you to capture additional information, which can be added to the sketch

Export Sketch Pros:

- Quick and simple process.
- Some AutoCAD export options for objects and file type.
- The sketch tools allow you to project geometry not included in the Face export.
- Resulting geometry made up of closed polylines.

Export Sketch Cons:

- Zero layer mapping options unlike the flat pattern export tool.
- No ability to export text for engraving.
- No ability to color the features by type for export.

To clarify the engraving text issue, in both situations above you could emboss the text and have it exported with the face or sketch.

However, this results in a text outline instead of a text center line. This is very inefficient for CNC machines to cut when the number is only there for practical rather than aesthetic reasons.

Create a Drawing

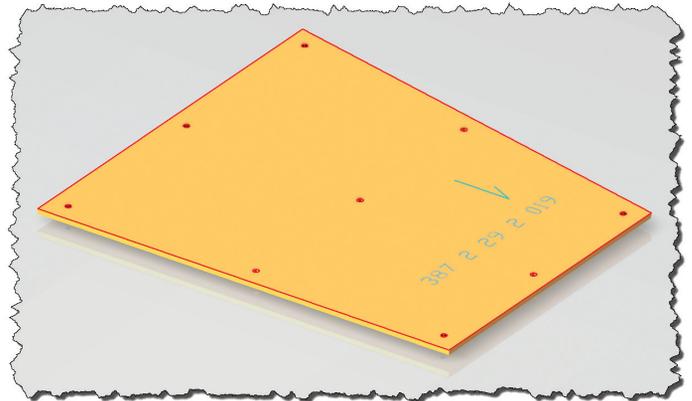


Figure 10: Inventor part with associative sketch cut file

This is my preferred method of generating cut files if you have a requirement for engraving or etching text onto your parts.

The two reasons for wanting to create cut files using the drawing environment are:

1. 2D labeled cut files saved in a single file. In this case you may want to:
 - a. Group your 2D cut files by material and thickness.
 - b. Then use leaders or balloons to access and display your parts iProperties.
 - c. Add instructions to the individual views via icons/symbols or notes for the CAM Programmer(s).
2. 2D cut files via model sketches:
 - a. Group your 2D cut files by material and thickness.
 - b. Display colored model sketches instead of visible edges.
 - c. Display positioned sketch text for CNC engraving.

I'll focus on the second option here:

1. Create your part.
2. Add a sketch for any projected geometry (reference geometry).
3. Add a sketch for 'sketched' geometry and sketch text.
4. Look at the top face (the one with the sketches on it) and set the view cube view as Front. This step is optional, but helps placement of the part views later.
5. "Rinse and repeat" for the other parts.
6. Create a drawing.

Inventor 2014

7. Start placing views, making sure the views are perfectly normal to the sketch faces.
8. Right click on each view in the browser and select 'Get Model Sketches.'
9. Check that all sketches are visible (sketches containing reference geometry won't be visible if they contain 'sketched' geometry as well).
10. Turn off the Visible Edges layer (to reduce/remove duplicate geometry).
11. Save as AutoCAD DWG.
12. In AutoCAD use Quick Select to select geometries by color and add to the appropriate layer.
13. Use the Overkill command to clean up the geometry.
14. Close polylines using the polyline edit command and the multiple option.
15. Check that all the cut files are "water tight" and open only where intended.

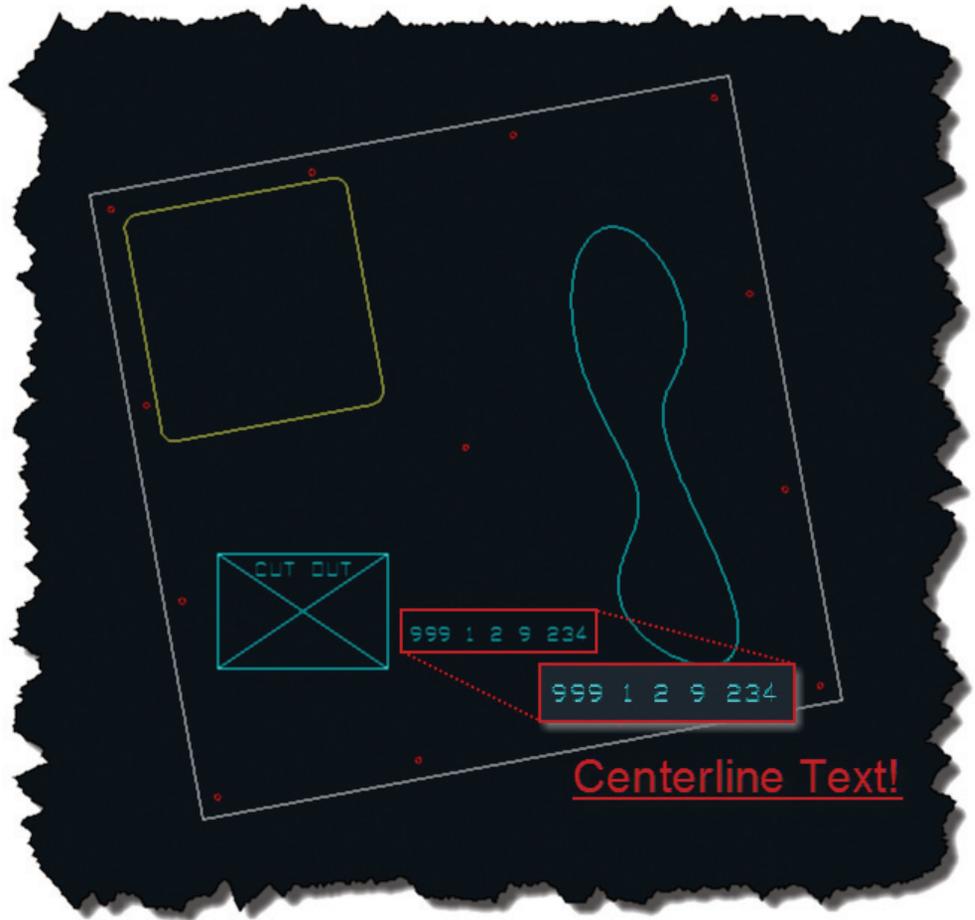


Figure 11: Once exported to DXF/DWG, the txt font becomes centerline text

Following that workflow alone would get you there. However, if you use Inventor DWGs as your company's standard drawing file format, then the following video demonstrating the use of the 'EXPORTLAYOUT' command in AutoCAD, may result in a few less files for you. Thanks to Paul Munford for letting me know about this one.

http://www.youtube.com/watch?feature=player_embedded&v=igPHBF9zbBE

So you may be thinking what a mission it is creating all this lot; however, you can automate the process via an add-in or a macro. So I thought I would share a video of one of our cut file macros at work:

http://www.youtube.com/watch?feature=player_embedded&v=6zj33QGWPI8

What really gets me excited about this workflow is its associativity with the base files. You can do the hard work of creating your CNC cut files while the product is still being designed. Any subtle changes that occur prior to the design being released for construction are associatively updated. All, with very little effort and at the click of a button, are exported en masse to a DXF/DWG for some pre-CAM prep.

Pros:

- + Associative cut files.
- + Allows you to manage exactly what geometries you want in the cut file.
- + Export of the 'txt' font text results in center line text in AutoCAD, which is perfect for CNC machines.
- + Geometry layers and color settings are maintained on Export, which can then be leveraged efficiently in AutoCAD.
- + Batch processing of cut files is possible natively, with zero customization.
- + The result is a single file containing multiple cut files which allows for efficient preparation in AutoCAD prior to CAM import.

Cons:

- + Resulting geometries are individual and open.
- + Splines aren't converted to polylines.
- + Based on the listed workflow, all geometries are on the same layer.
- + The result is a single file containing multiple cut files, which can mean a lot of work splitting up the cut files in the CAM software prior to nesting.

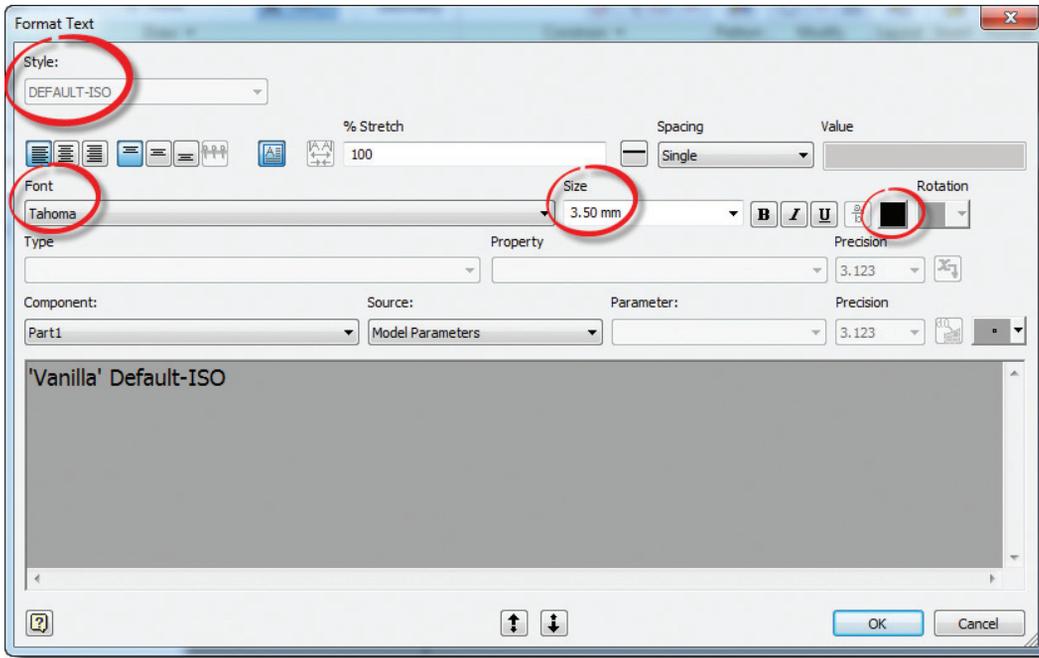


Figure 12: Sick of changing these settings all the time?

Setting up a Sketch Text Style

Do you find yourself swearing at Inventor when you have to repetitively set the font, size, and color of your sketch text in the part environment? It used to really irritate me that the dialog was identical to the one in the drawing environment, but the text style drop-down is permanently greyed out and set to 'Default ANSI' or 'Default ISO' depending on your default standard. I just wanted to pick my own style, which had the presets I use most often and couldn't. But there is a sneaky undocumented workaround.

If you were to look in the styles and within all the Inventor folders on your local drive for any trace of a text style named 'Default ANSI' or 'Default ISO', etc., you won't find anything. So the trick is to create a text style matching the exact name of the style you

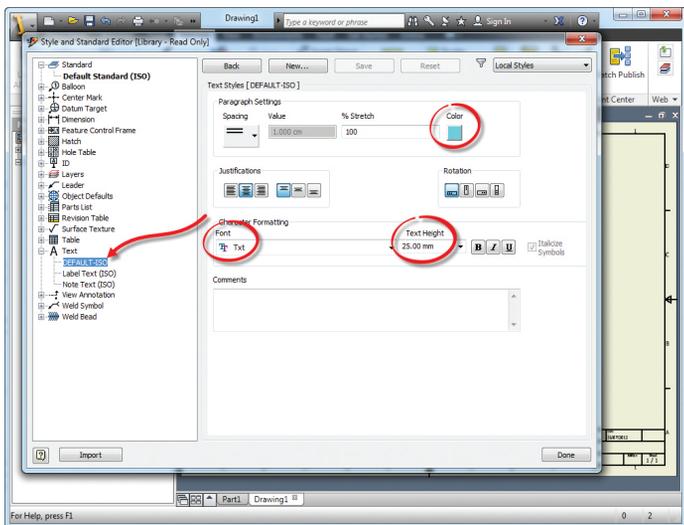


Figure 13: Create a new text style with an identical name as the 'greyed out' text style in the text editor

can't find, then just configure it with the settings you are so fed up with selecting each time you create a new text box.

The next time you create a text box in a sketch, all of your settings will be just so. If you've been creating a ton of CNC cut files requiring text engraving with Inventor, it's at this point you may weep a little (I certainly did).

Thanks to the individual who shared this solution with me a number of years ago via the Autodesk Discussion Forums.

CREDITS

Thank you to Gavin Bath for contributing a huge effort towards this article, and also CADPRO Systems Ltd. for allowing this information to be shared publicly.

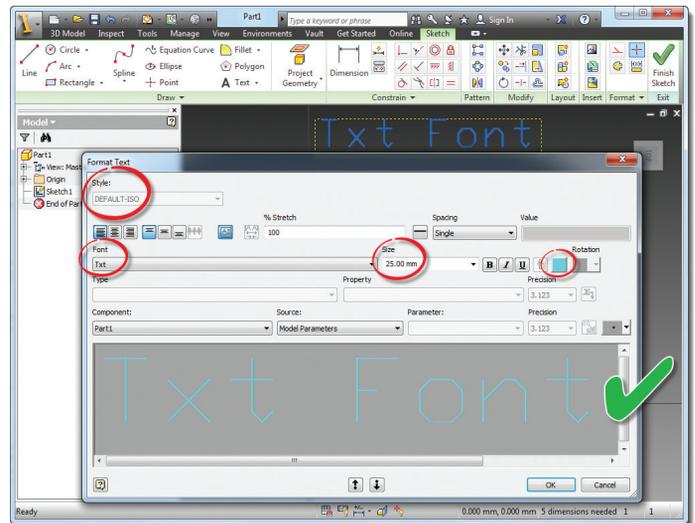


Figure 14: Check that out! Perfect settings every time



Scott Moyle is a Technical Specialist for CADPRO Systems, an Autodesk VAR in New Zealand. His background is in Motorsport Engineering and CNC programming. Scott has been using various Autodesk software products for the last 12 years. He co-owns and blogs on DesignandMotion.net. Scott can be reached using Twitter @scottmoyle or scott@designandmotion.net.

Getting Attached with XREFs



Some of you get pretty excited about and have gotten quite attached to AutoCAD®. One way to make your experience even better is to take your attachment to another level by using external file references. If you are not using External References (XREFs), which have been around for a long time, you are really missing out on a powerful tool. If you are a current user, keep reading—maybe you will find some new things covered here that you have been a bit curious about.

Throughout this article I will refer to the process of attaching external reference files as “XREF/XREFING,” to the attached files as “XREFs,” and to the drawing to which you attach XREFs as the “working drawing.”

WHAT ARE XREFS?

External references are much like the concept of blocks, which was the old school way of handling many of the functions now provided by external references. Rather than being embedded (inserted) in the file, they are externally ‘attached’ or ‘referenced.’ Items that can be attached include DWGs, PDFs, DGNs, DWFs, and IMAGES (BMP, JPG, TIFF, PNG). Different commands can be used to attach the various external files, and these commands start with the file type and the word “ATTACH.” Examples are PDFATTACH,

DGNATTACH, IMAGEATTACH, and DWFATTACH. Or you can just use the XREF command, which will allow you to attach any type from one dialog. The individual commands are great for use in automation efforts (macros, scripts, and LISP), but will not likely be your first “go to” option.

WHY DO YOU XREF?

Because it is the right thing to do, of course! If you are new to AutoCAD and XREFs and wonder what all the fuss is about, XREFs provide a lot of pretty cool features.

- They keep your working file size small
- They allow sharing of files while providing real-time updates
- They are great for coordination

WHAT DO YOU XREF?

As stated above, in what are External References, you see a lot of file types that can be attached to your working drawings. Typical items you would attach include floor plans, title blocks, logos, standard details, survey pictures, and product literature. But if you have any of the file types above that you commonly share in projects, the options are unlimited.

WHY BLOCK INSERT WHEN YOU CAN XREF?

The big difference between working on a background that is a block versus a background that is an XREF is that the external XREF can be worked on by someone else and be shared real time with multiple users or files. External references keep your drawing files small and allow you to share the background with multiple drawings.

EXAMPLE USES

The following are examples of how XREFs can be used to increase your productivity. The one big advantage of every one of these examples is that when you update the XREF, every file that references it is updated.

TITLE BLOCKS

This is probably one of the top two uses of external references. In a typical project the one item that is common among all drawings is the title block. By XREFing the title block drawing, you can make changes to the address, issue date, customer info, and logo in one place and have all of your working drawings updated automatically.

Note that the logo in the title block is another common XREF (IMAGEATTACH). You can also enter all your revision information for the entire job in this drawing as individual layers (i.e., REV-1, REV-2, etc.) freeze them in the title block, and thaw as necessary in the working drawings.

FLOOR PLANS

Probably the top use of XREFs. Through a combination of external references, clipping, and paper space, you can attach multiple plans and plot at different scales all on one sheet.

XREFs can be renamed when attached, and by doing this you can attach the same file multiple times and control the visual aspects of each as if it were a separate file. This is referred to as logical name versus actual name. An example would be attaching a floor plan (FP-1) that has demo and new work layers all in one drawing. By attaching FP-1 and then renaming it in the XREF dialog to FP-Demo, you can freeze and thaw layers as required to show the demo portion of the plan. You then attach (not copy) the FP-1 plan again and freeze and thaw layers as required to show new work. Since each logical XREF has its own layer structure showing up in your layer dialog, you can also change colors and line types.

Note that this feature relies on your VISRETAIN SETVAR being set to "1."

PRODUCT LITERATURE

In some jurisdictions it is required that manufacturers' data, safety information, or product performance be shown on the drawings. Instead of retyping all this data or redrawing it (or creating "sticky backs," for you old-timers) just XREF the PDF or image files into your working drawings.

STANDARD DETAILS

Nearly every set of drawings utilizes some form of standard details. You could have a standard detail sheet with the details XREFed so that the sheet is always up-to-date. If you need to revise a standard detail to be job specific or freeze the details specifics from changing during the course of the job, you can bind those details into your working drawing and edit as required within the current job.

SURVEY INFORMATION

When doing renovation projects, a necessary task is to provide enough information on the existing conditions to allow contractors to be able to see what they face so they can give an accurate bid. Although requiring a field visit is always a good idea, it is not always practical for some projects. A great way to enhance your drawings is to IMAGEATTACH field photos in your drawings and add notes detailing the specifics.

KEY PLANS

When working on large building projects or campus-wide projects, you can attach aerial views or screenshots from your favorite mapping program to visually show your site or campus. For large building projects, you can attach a drawing at a reduced scale of the overall building with hatched areas or layers frozen or thawed to represent specific areas of work.

Any plan changes will automatically update your key plan as the project progresses.

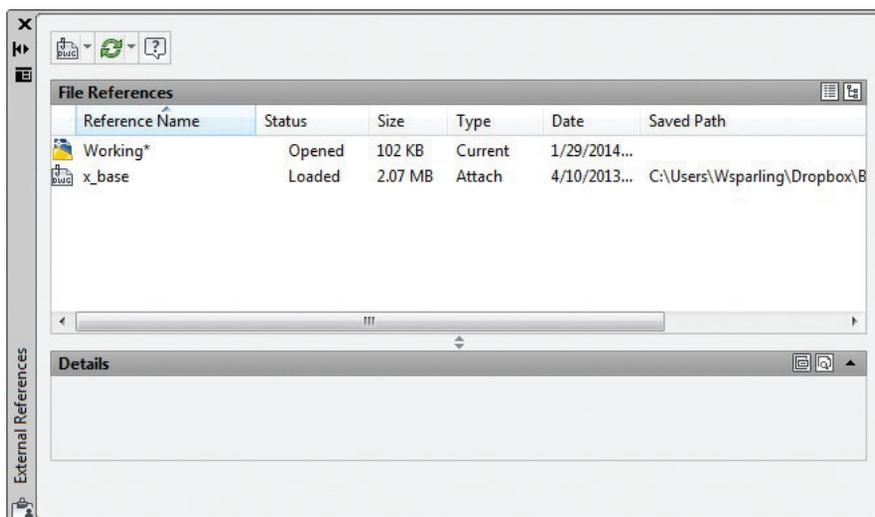


Figure 1: A working drawing with a X_Base drawing (external reference) attached

AutoCAD 2014

If you choose the drop-down menu in the upper left (see Figure 2), you will see the variety of file types that can be attached to your working drawing. Because AutoCAD now has so many external attachment options, the “XREF” OR EXTERNALREFERENCES” dialog gives you a full picture of all your attachments.

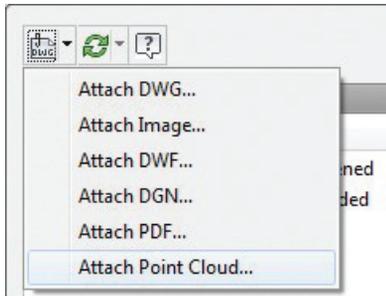


Figure 2: External file attachment options

For daily use, I use the CLASSICXREF command which is “XR” in my shortcut keys to do daily DWG reference functions (see Figure 3).

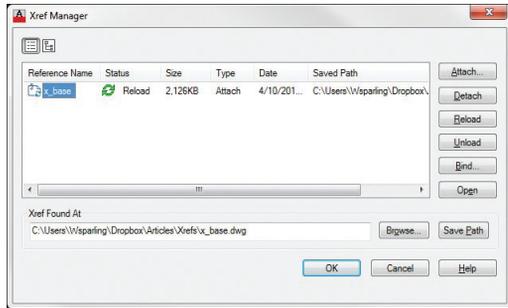


Figure 3: CLASSICXREF Dialog

XREF BINDING

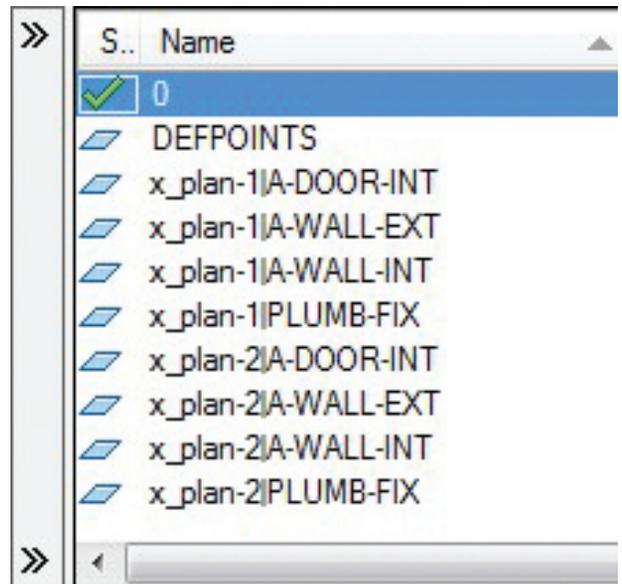
Sometimes you may want to make the externally attached files a permanent part of your working drawing, such as for archiving or sharing with another program that does not understand reference files. Note that images, DWFs, and PDFs cannot be bound.

There are two options for binding external references: Bind and Insert. When choosing the Bind/Bind option, all logical named items in the external reference files get bound with a prefix denoting the original XREF name. These items include layers, line-types, text styles, blocks, dimension styles, and so on. This can make for a very messy listing of the various types, or be very useful, depending on your goals. Bind/Insert merges all the items previously mentioned into the working drawings. I typically prefer this binding option as it reduces the clutter in the drawing and makes it easy to do updates.

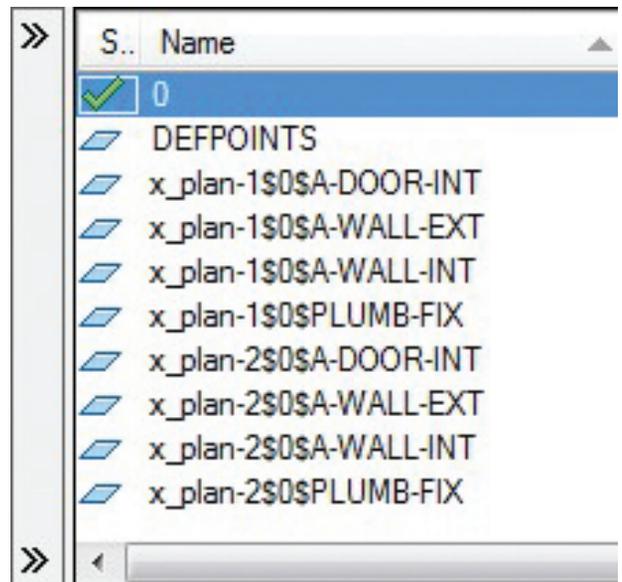
A CREATIVE USE

Have you ever inserted your default North arrow symbol and then had other people copy that to their drawings only to find out later that it’s in the wrong orientation? Then you have to go back through all the plan sheets and fix the North arrow. Bummer! Think about how many times you have to insert a North arrow and rotate it to the right orientation. What if every project had a North arrow drawing that was oriented properly one time and then everybody just referenced it into each plan sheet. An alternative is to put it in the title block, but then it would not apply if the plan were rotated on a sheet.

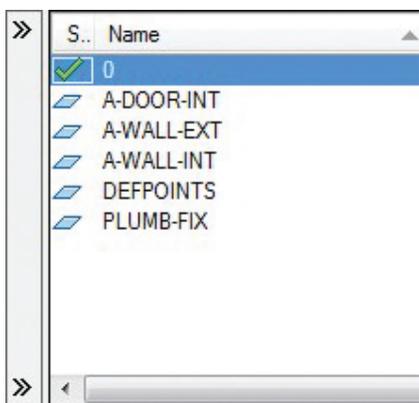
Below are a couple screenshots showing the difference between the Bind/Bind and the Bind/Insert options:



Layers before Binding



Layers after Bind/Bind



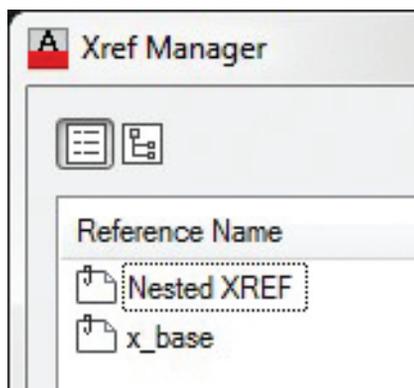
Layers after Bind/Insert

As you can see above, it is much cleaner to use the Bind/Insert option. The additional "\$0\$" symbology will get added to all your linetypes, text styles, blocks, etc.

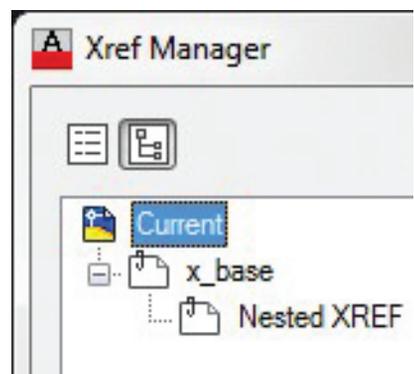
NOTE: If you are having trouble getting files to bind, run an audit and purge on the XREFs and the working drawing. That typically does the trick.

OVERLAYS VERSUS ATTACHMENTS

AutoCAD has the ability to 'nest' XREFs (i.e., attach an XREF that has an XREF attached to it). In the XREF manager (CLASSICXREF) and in the External References dialog, you will see two icons representing two different dialog views. You can see when a XREF is nested using the 'Tree View.'



List View



Tree View

The benefit of this feature is that you can stack XREFs for coordination purposes. Assume you are working on a reflected ceiling plan and you need to coordinate diffuser and sprinkler locations with your lights. You can attach the HVAC plan and the sprinkler plan to your file to do your coordination. You can keep these files attached for future change coordination, but hide them from view.

In the "XREF Manager" dialog, select the file you wish to hide and choose the Unload button. In the "External References" dialog, right-click the file and select Unload. These functions are all part of the normal nesting function of XREF attachments. There is another attachment option called "Overlay" that allows you to attach an XREF to a file and only allow it to be seen in the current file. Note that this does not work for images, DWFs, or PDFs.

Using the above example for coordination, if the electrical designer wishes to power up the lights, he or she could attach the reflected ceiling plan to the lighting plan to do the circuiting. If the HVAC plan and the sprinkler plan were standard XREFs, that would be seen on the plan as well. If there is no need for that, to avoid possible confusion, the HVAC and sprinkler plans could be attached as Overlays and the electrical designer would not even know they existed.

You can change the reference type on the fly. In the "XREF Manager" dialog, select the file you wish to hide and double-click on the word Attach or Overlay under Type and it will change. In the External References dialog, right-click the file and select Attach, and in the pop-up make your change.

Size	Type
50...	Attach
53...	Overlay

XREF Manager



External References

There are a lot of commands and system variables that can be used when using XREFs, but like most other AutoCAD features, you typically need only a few. Below are the most common commands and SETVARS that you would use on a daily basis.

COMMONLY USED EXTERNAL REFERENCE COMMANDS

- REFEDIT – Edit Reference In-Place.
- XCLIP – Allows clipping (hiding portions) of the XREF from view. Great for enlarged plans or partial details. (Note: for other file types: PDFCLIP, DGNCLIP, IMAGECLIP, DWFCCLIP.)
- XBIND– Allows binding of individual items from an external DWG file attachment. Say you want to bring in a block or linetype from the external reference. Use XBIND to bind it and then rename it if you would like.

COMMONLY USED EXTERNAL REFERENCE SETVARS

- VISRETAIN – Controls visibility, color, linetype, lineweight, and plot styles. Should be set to 1.
- XFADECTL – Controls the dimming for all DWG XREF objects. (Fade from 0 – 90 percent, user visual preference).
- XREFNOTIFY – Controls the notification for updated or missing XREFs. Should be set to 1 or 2.
- XREFTYPE – Controls the default reference type when attaching or overlaying an external reference. This depends on use, but the majority of XREFs will be Attachments, not Overlays. See “Overlays Versus Attachments” earlier in this article.

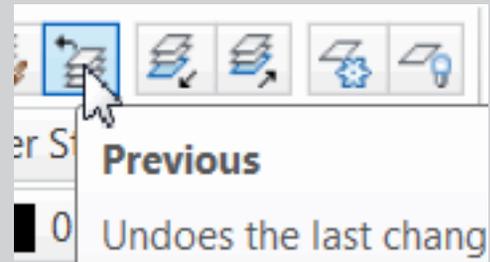
If you are currently using external references then you have already seen the many advantages they provide. If you are not currently using external references, it's time to get attached!



Walt Sparling has worked in the building design industry for 25+ years, starting as a hand drafter. He moved on to CADD in the mid 1980s and then into CADD and networking training and consulting. Walt has served as project manager and designer in the mechanical and architectural realms and currently works with an electrical engineering firm in Tampa, Florida. In his spare time, Walt maintains a couple of blogs and a personal website: FunctionSense.com and waltsparling.com

And Here's How to... Make Layer Previous Work

Layers are the lifeblood in AutoCAD®, and if you aren't really good at managing your layers – and more importantly, knowing how to make them behave—here's how to ratchet up your game. One of the oft-overlooked features on the Layers panel is Layer Previous. At its core, this feature is a layer-specific “Undo.”



HOW TO USE LAYER PREVIOUS

1. Open a drawing you have been working on, then make a new layer [AW_FEB14], give it a color, and make it current; this will be our test layer.
2. Change the color of any other layer in the current drawing, but don't make that layer current.
3. On the test layer, draw a small rectangle somewhere.
4. Click Layer Previous and the rectangle remains, but the color of the layer you changed is restored.
5. Freeze several existing layers; now Rotate the test rectangle.
6. Click Layer Previous and the previously frozen layers are thawed ... but the rectangle is still rotated!

Needless to say, there are several situations where Layer Previous will save the day.



Michael E. Beall (B. Arch.) won the AU 2013 Speaker Award for Hands-on Labs and is an Autodesk Authorized Author. He is the owner of CAD Trainer Guy, LLC in Shelbyville, KY and has been presenting onsite CAD training around the planet since 1982. Contact him anytime at michael.beall@cadtrainerguy.com, on LinkedIn at <http://www.linkedin.com/in/cadtrainerguy/> or give him a call at 502.500.2267.

Top Price/Performance, USA Built!



New K1100M & Mtower LE

@Xi® Computer has been building the most performing AutoCAD® Workstations for over 20 years. The new Intel®Core™ i7 & XEON® based, MTower™ & PowerGo™ are no exception. As NVIDIA® TPP, we are offering **Maximus™ Certified Workstations** combining HPC and Visualization power in a single Desktop. Proudly Custom Built in the USA, sided by a superior domestic technical support, Xi Computers are designed to excel and reliably deliver, even in the most demanding computational environments.



MTower LE

PowerGo XT

MTower 2P64X



- Intel® Core™ i5|XEON E3 v3
- Intel H81/C226 C.S. PCIe 3.0
- 8GB-DDR3-1600 or 1600ECC
- NVIDIA®Quadro® K600 1GB 2H
- 500GB Seagate® 7200 16MB C.
- 5 in 1 Card Reader,Front USB 3.0
- Xi®MTower LE Small Form Factor
- MS Windows® 7/8.1 Pro® -Linux®

- Intel Core i7 4800MQ 3.7GHz
- up to 32MB Cache, 4/6 Cores
- 8GB DDR3 1600MHz to 32GB
- NVIDIA®Quadro® K1100/3100M
- 500GB SATA6 Hybrid 8GB SSD
- 17.3" 1920x1080NR LED
- Speakers, Camera,1394+ Mike
- Microsoft® Windows® 7/8.1 Pro®

- Intel Xeon®Processor, New E5-V2
- 10/20MB L3Cache,C602 Chipset
- 16GB DDR3-1600 ECC to128G
- NVIDIA®Quadro® K2000 2GB
- 250GB Samsung® EVO™SSD
- 27" 1920x1080 LCD Monitor
- Xi®MTower 720/1000W HAF
- MS Windows® 7/8.1Pro®-Linux®

XEON E3-1230 3.7 Ghz TB \$1,199
4-Core i5 4430 3.2 Ghz TB \$ 999

17.3" 1GB K1100M \$1,899
17.3" 4GB K3100M \$2,379

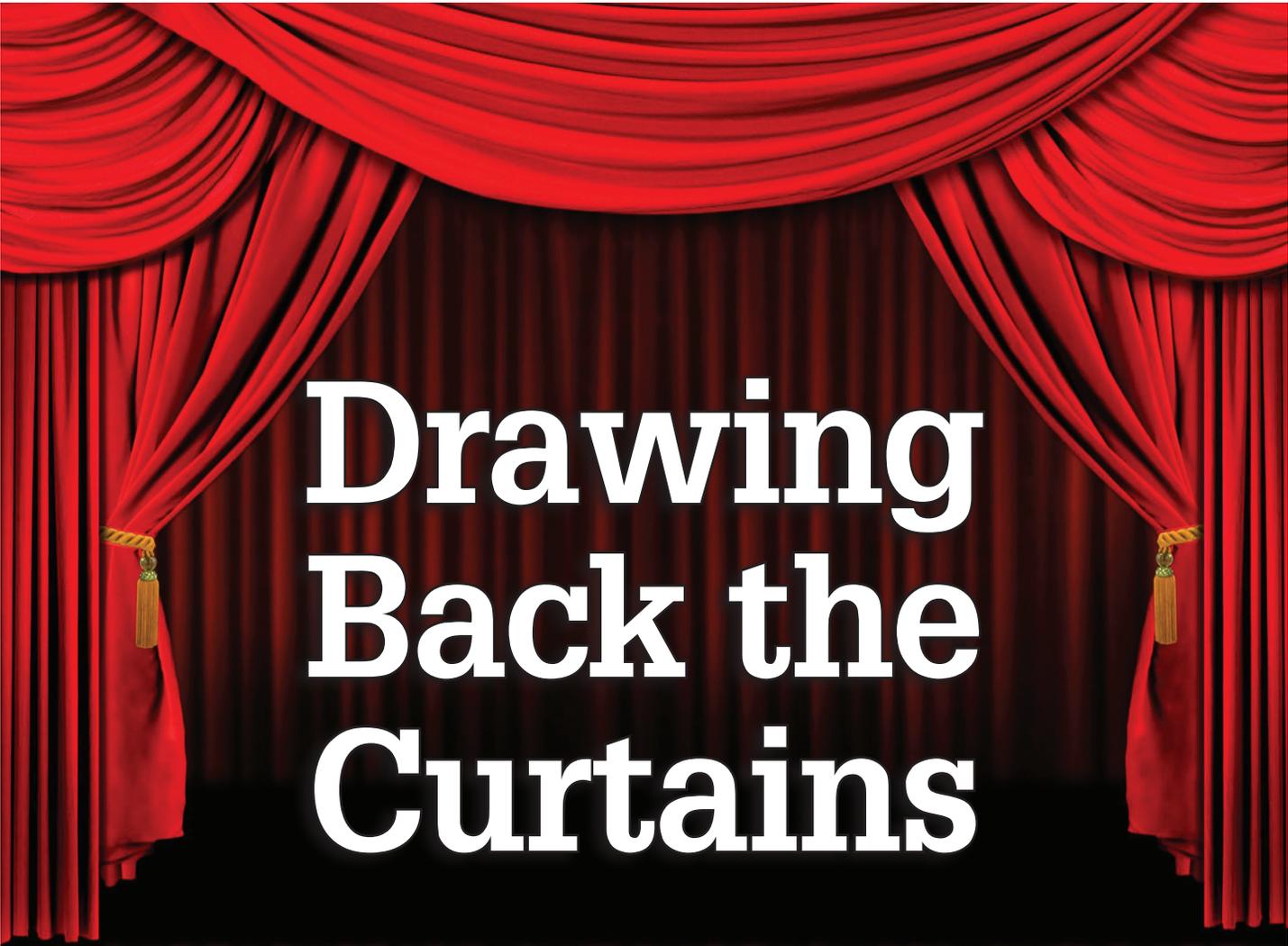
1x4-C E5-1620v2 3.9GHzTB \$1,999
2x8-C E5-2687Wv2 3.4GHz \$6,949

NVIDIA®Tesla™ K40 & Quadro®K5000/K6000 NVIDIA®MAXIMUS™TECHNOLOGY Certified NEW Xeon Phi™ Coprocessor with 61 cores

Xi®Certified for AutoCAD®2014, 3ds Max®, Inventor®, SolidWorks®, Maya® & Revit®. Custom built in the USA from 100+options to match your applications and your budget. Business Lease from \$ 59/month.

www.xicomputer.com

1-800-432-0486



Drawing Back the Curtains

Hi, everyone, Lynda.com and Infinite Skills author Joel Bradley here with a quick tutorial that walks us through the process of modeling a simple set of tied-back curtains using some of 3ds Max's Cloth Simulation tools, specifically the Cloth modifier itself.

SETTING UP THE PROJECT FOLDER

To be able to follow along with the steps we will take here you will need to download the supplied project files from this link: www.vrayelite.co.uk/AugiWorld/Project_Files.zip.

Once you have the zip file download you can then extract the contents of the archive to a local or portable disk location of your choice. Just be sure to make a note of where you put them as you will need to navigate to the root folder in just a moment or two.

Once extracted, we can now fire up Autodesk® 3ds Max® to set up and make use of the 3ds Max Project structure provided in the download. To do that, let's:

- Come up to the customize menu at the top of the 3ds Max UI and from there open up the 'Preference Settings' dialog.

- Inside there we want to first switch over to the Files tab, and then make sure that the 'Convert local file paths to relative' option has a check mark in it.
- Finally, click OK to save the changes and exit the dialog.

Enabling this option makes sure that any files you may want to add to the project from this point forward will be given file paths that are relative to the project root, as opposed to being given hard-wired local paths.

Now we are ready to set up our 3ds Max project folder. To do that, we:

- Come up to the Quick Start toolbar found in the top left corner of the 3ds Max UI and there click on the 'Project Folder' icon.
- In the 'Browse For Folder' pop-up that appears we need to navigate to the Project_Files folder that we extracted to disk a few moments ago.
- Once located, all we need to do in order to set this as the Project folder is left mouse click to select and highlight it and then click OK to lock that selection in place, dismissing the 'Browse For Folder' dialog at the same time.

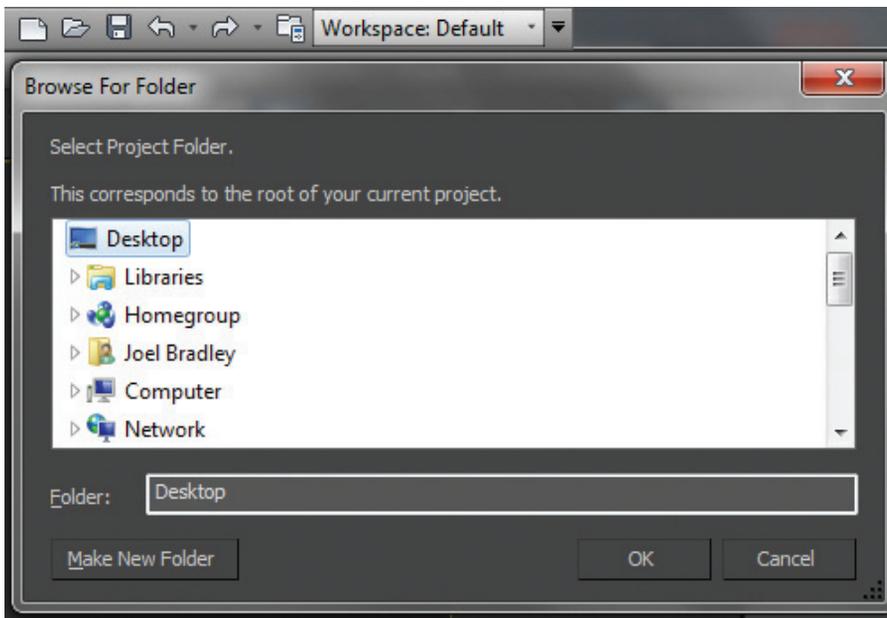


Figure 1: Project Folder button in Quick Start menu

OVERVIEW OF OUR START SCENE

With the Project Folder correctly set up, we can now open up the 3ds Max scene file with which we will start our tutorial. To do that:

- Come back up to the Quick Start toolbar and this time click on the 'Open File' icon. This will take us straight to the 'scenes' folder located inside the Project structure.
- In here, left-click to open up the 'Curtain_Start.max' scene file.

If you now scrub the time slider down at the bottom of the 3ds Max UI, you should be able to see that we currently have three animated boxes in the scene, each of which contracts along its own local X axes as time moves forward. These are the objects that we will eventually be using as 'cloth driver' objects that will essentially pull our curtains into place for us.

We also have two simple geometry planes in the scene, one will serve as our curtain whilst the other will be used to create a pelmet that in real life would hide the rail or pole upon which the curtain hangs. Here we will use it to disguise the fact that we have no rail or pole geometry in the scene.

You will also notice if you select each of them in turn that these planes have quite high polygon counts. This is by design as it will help us produce better deformations in the geometry once we start to simulate and fold the curtains.

If you have used 3ds Max's cloth tools before, perhaps having worked through a number of tutorials already, you may be wondering why we are keeping Quad subdivisions in the meshes rather than using the more often recommended Delaunay triangulation pattern. We are doing so in this instance because

quads actually give us better final deformations than would come from the triangulated mesh of modifiers such as 'Garment Maker.' The uniformity of the quads just seems to help the curtains fold in a more interesting manner. And finally we are using the Iray renderer in the scene so that lighting and sampling turn out at fairly decent quality levels without us having to worry about parameters and settings. Of course, the render times we get may well be longer than desired, especially if we follow a workflow that requires us to make a number of test renders throughout a project. As we will only be taking one or two renders here though, this set up should work out fine.

With the set up out of the way, let's make a start at creating our Cloth objects, working first with our curtain geometry.

TURNING THE CURTAIN INTO A CLOTH OBJECT

The first thing we will need to do, of course, is turn the curtain into something that 3ds Max can recognize as a cloth object for simulation. To do that, we:

- With the GEO_Curtain object selected, come over to the Modify tab in the Command panel and from the modifier drop-down list apply a Cloth modifier.

This instantly makes a wealth of commands available to us in the Command panel—an extremely important one being the 'Object Properties' button. In order to access the controls that give us the ability to assign specific cloth properties to our geometry, we need to:

- Click the 'Object Properties' button, which will open up the Object Properties panel.
- From the 'Objects in Simulation' list on the left, we can select the GEO_Curtain object and then from the very top of the parameter section right-click the 'Cloth' radial button. This tells 3ds Max that we want to count this particular object in the simulation as Cloth.
- From the 'Cloth Properties' presets list found in the drop-down just below, choose the 'Cotton' option and then lower the U&V Compress values to 50 (these should be linked by default) to help give us more realistic folds in the material.
- We also want to add a little more density to the cloth to help it hang a little better, so let's increase that parameter from 0.01 to 0.02.



Figure 2: Object Properties dialog

As we already have a scene set up using geometry to represent real-world solid objects such as the walls, we will need to tell 3ds Max that these need to collide with the cloth inside the simulation. To do that, while still inside the Object Properties dialog, we need to;

- Click the 'Add Objects' button found just above the 'Objects in Simulation' list, and from the 'Add Objects to Cloth Simulation' dialog that appears, select the GEO_Room and GEO_Window_Sill objects, and then click the Add button.

Now, although our two objects look as though they are currently selected in the 'Objects in Simulation' list, this is a bit of a bug in 3ds Max that is trying to kid us, so before making any changes we just need to de-select and then re-select them both. We can easily

tell when we have them correctly selected as the 'Inactive' radial button becomes selected.

- Next, we need to click the 'Collision Object' radial button found down at the bottom of the Object Properties dialog and then enter an 'Offset' value of 0.35.
- To finalize the settings we have entered, click the OK button at the bottom of the dialog.

NOTE: Be sure NOT to click the X in the top right corner of the dialog as this will effectively cancel everything that we have done and lose the settings.

LINKING THE CURTAIN TO THE DRIVER OBJECTS

With the object types correctly added to the simulation, we can now set about linking our curtain geometry to the animated driver objects in the scene. To do that, in the 3ds Max Command panel, we:

- Expand the Cloth modifier at the top of the modifier list using the + button found just before its name and from the options select 'Group' sub-object level.
- As we need to get a clear view of things here, let's hit the F key to switch over to a front view and then press F3 to make use of wireframe mode in the viewport.
- To get nice and close to the action we can select the Zoom Region tool from the controls in the bottom right of the 3ds Max UI and in the viewport, draw a region around the animated driver object (box) found at the top of the curtain.
- After right-clicking to exit the Zoom tool, make certain that the selection method is set to 'Rectangular' and then click and drag to make a rough selection of all the curtain vertices that sit within the visible volume of the driver (Box) object as seen in Figure 3.

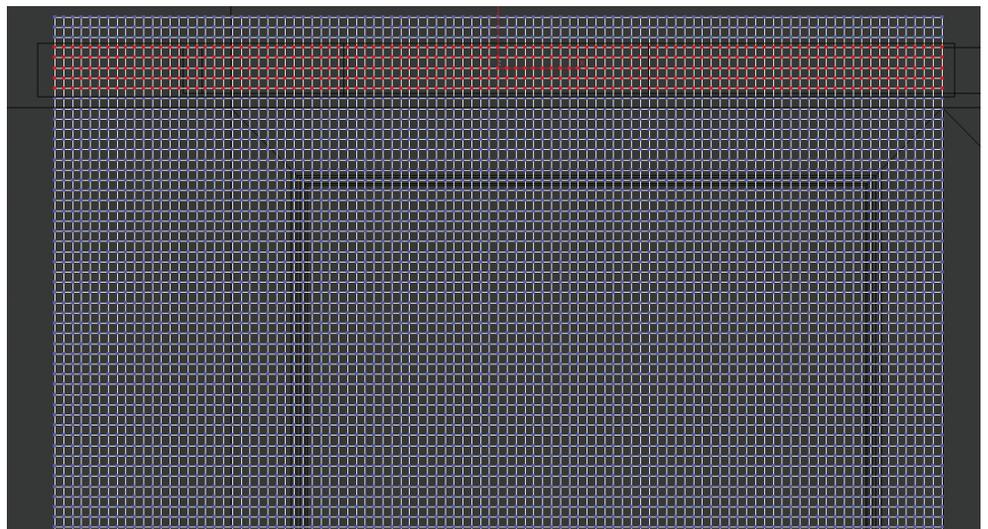


Figure 3: Selected verts

- Back over in the Command panel, we now need to click the 'Make Group' button found in the Group rollout, giving it a name in the 'Make Group' dialog that describes what the group is doing. 'Top_Node' should do nicely here, and then once we are done we can click OK to exit.

We now have an unassigned group (as seen in the group list window), which means that it currently has no behavioral instructions attached to it. To alter this we need to:

- Click the 'Node' button from the options that have now become active in the Command panel and then click in the viewport to select the GEO_Top_Cloth_Driver object. (You will now see driver object's name appear next to the group in the list window.)

Note: Alternatively, having clicked the 'Node' button we could hit the H key and then select the driver object from the Pick Object dialog that appears.

Of course, we now need to follow through and use the same basic procedure for the cloth driver object that is positioned at the mid-point of our curtain. To do that, let's:

- Re-position our view by means of a middle mouse click and drag, making certain that we place the 'GEO_Mid_Cloth_Driver' object at around about the middle of the view.
- Again we need to make a selection of vertices inside the volume of the driver object, but as this geometry is at an angle in the view, we will need to switch our selection tool and use either the 'Fence' or 'Paint' Selection options.
- In my case I am going to switch over to the 'Paint selection' tool and then use it to create the selection seen in Figure 4.

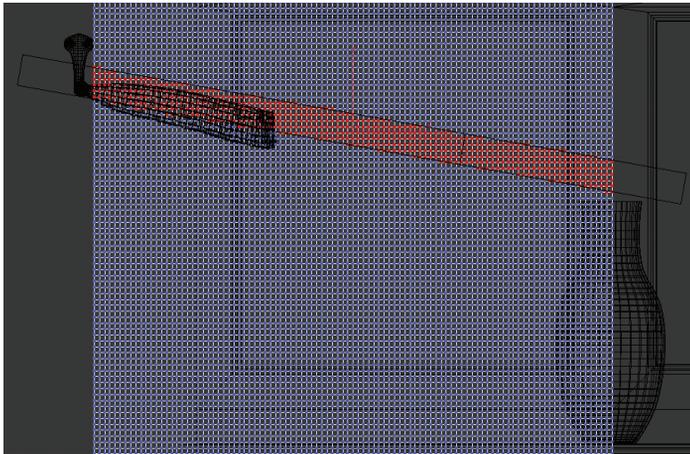


Figure 4: Selected mid verts

If you get a bit carried away with the selection you can always hold down the Alt key and then paint to de-select certain vertices.

- We do need to turn our selection into a Group, so again let's click the 'Make Group' button over in the Command panel, name it 'Mid_Node,' and then click OK.
- To apply the required behavior to the new group we need to click the 'Node' button and then in the viewport left-click to select the 'GEO_Mid_Cloth_Driver.'

- Finally, up in the modifier list left-click on the top level 'Cloth' label in order to exit modifier sub-object mode.

SETTING THE OPTIONS AND SIMULATING THE CLOTH

Now we need to apply the final set of parameters that will allow us to simulate our curtain into place. To do that:

- Still in the Command panel, scroll down in the Cloth modifier parameters to the 'Simulation Parameters' rollout and from there increase the Subsample value to 3 to improve the accuracy of our curtain simulation.
- As we are folding the cloth in on itself, we will also want to put a check in the Self Collision option and set its numeric value to 1.

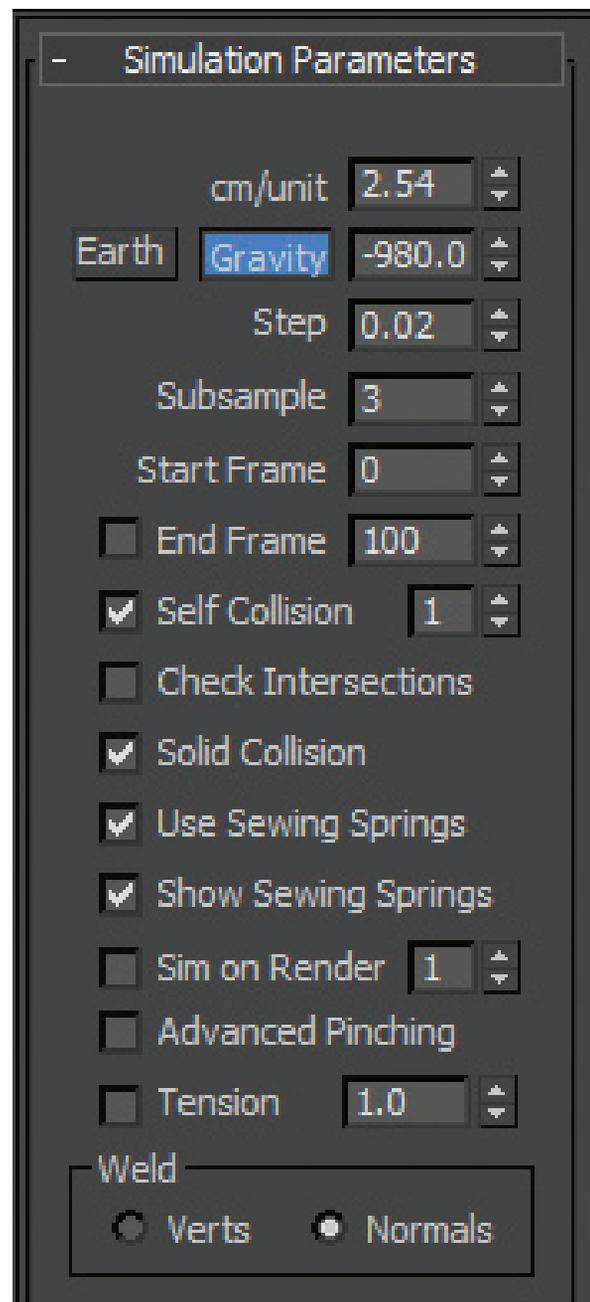


Figure 5: Final simulation parameters

Note: If you are wondering why we aren't also enabling the 'Check Intersections' option, the simple truth is that this feature is discontinued and so no longer functions in a newly created simulation such as this.

- We also need to ensure that 'Gravity' is enabled and set at an Earth actual value of -980cm (-9.8 if working meters).
- Finally, we can come back up to the top of the rollout, hit the 'Simulate' button, and then wait for our curtain to be drawn into place. (You may want to hit the C key and switch back to the Camera view here to watch this progress.)

Once the simulation has run its 300-frame course, because we are only looking to create static curtain models here, we can use some very simple 3ds Max functionality to create a matched pair.

- From the Tools menu, use the 'Snapshot' tool with the 'Single' and 'Mesh' options chosen to create a static copy of our mesh in its current form.
- Next we drag the time slider back to around frame 275 (adding a little bit of variation to the way our cloth looks) and again use the Snapshot tool, giving us two static pieces of curtain geometry.
- To tidy up the scene somewhat, let's either right-click on the still selected simulated curtain and then use the 'Hide Selected' option from the quad menu, or simply delete it from the scene.

Note: If we simply hide the mesh, then our cached cloth simulation will be saved with the 3ds Max file, bumping up both its size and load times by quite a bit. In my case, I am going to delete it from the scene in order to keep files sizes down.

To place one of our curtains over on the opposite side of the window:

- Select one of the snapshot meshes in the scene and from the main toolbar click on the Mirror tool.
- We can set to the mirror axis in the pop up dialog to X, choose 'No Clone' from the Clone Selection options, and then in the Offset field set a value of around about 71cm in order to nicely separate the two curtains.

Note: If desired we could at this point add a shell modifier to give the curtains some thickness for this demonstration, though we will forgo that step and instead select and hide our animated driver objects and come up to the main toolbar and hit the Render button.

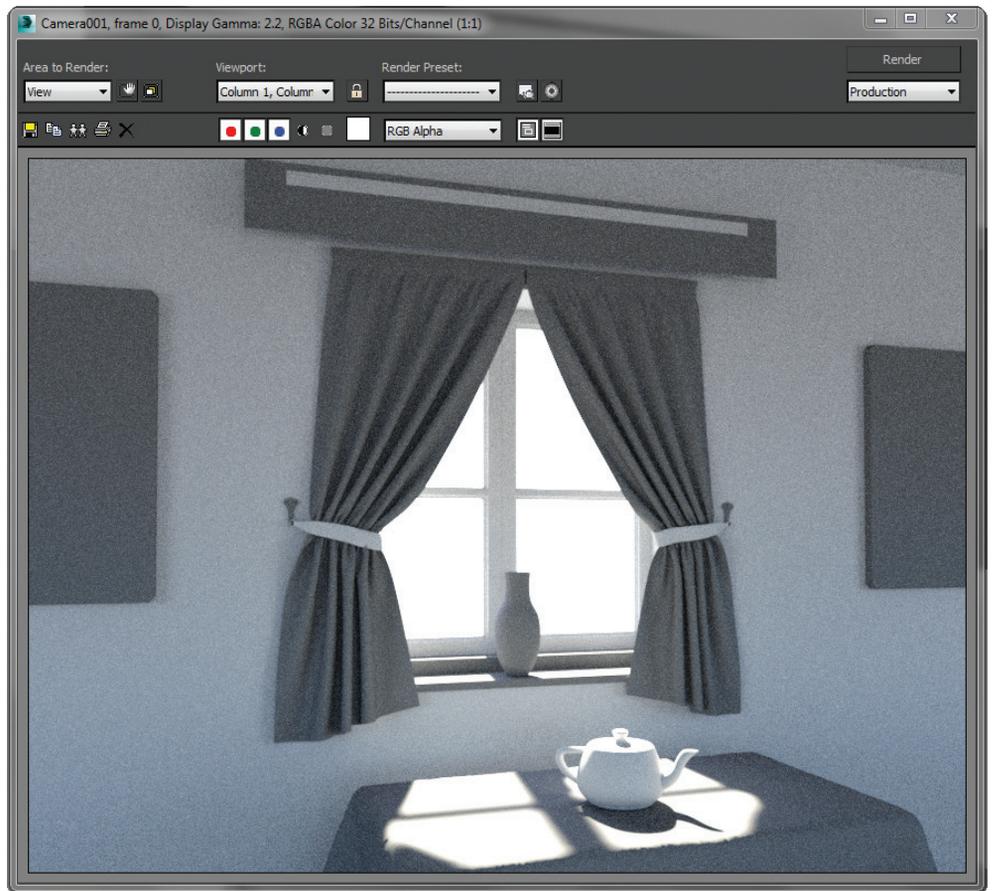


Figure 6: Curtains in final simulated position

TURNING THE PELMET INTO A CLOTH OBJECT

Turning our second geometry plane into a Pelmet requires following a set of steps that are pretty much identical to those used in the creation of our curtain. The only noticeable difference will be the way in which we select and group our vertices for attaching to the animated driver.

- With the Pelmet plane selected, let's come over to the Modify tab and from the drop-down list, apply a Cloth modifier.
- In the 'Object' rollout, click the 'Object Properties' button, select the 'GEO_Pelmet' entry from the list on the left, and then click to select the 'Cloth' radial button.
- From the 'Presets' drop-down we can again choose the Cotton option and lower the U&V Compress values to 50 to help achieve better deformation in the cloth.
- Once done, click the OK button to finalize the settings and dismiss the dialog.

As the Pelmet geometry is nowhere near the wall (being moved into place by hand once the simulation is done), there is no need for us to add any collision objects to this particular simulation.

LINKING THE PELMET TO THE DRIVER OBJECTS

To link vertices in the Pelmet geometry to the driver object:

- In the modifier list, access the cloth modifier's sub-object options via the + button and select 'Group' sub-object level.
- Again, we will probably want to make sure that we are in a front view so let's hit the F key to switch over to that and at the same time use F3 to ensure that we are in wireframe mode.
- Now we can Zoom in and Marquee select the first two vertices of the top row, come over to the Command panel, and from there click the 'Loop' button found in the 'Group' rollout.

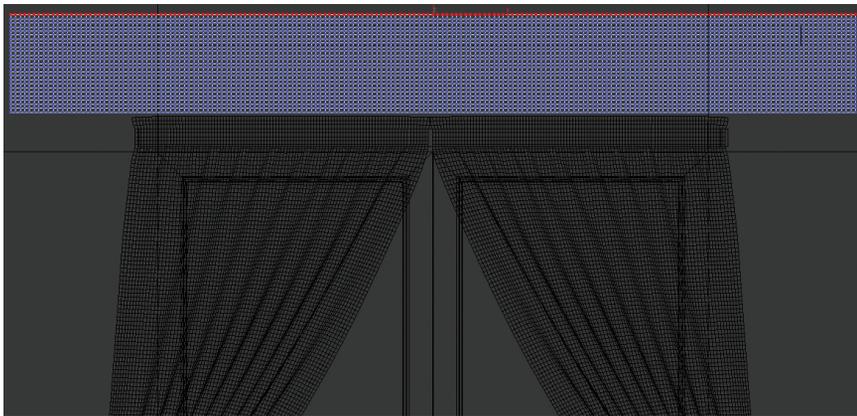


Figure 7: Selected top row verts

- From the top we want to count roughly 7 to 10 rows down and then, whilst holding down the Control key (which we need to keep down for the rest of this selection operation), again select the first two vertices in a row and then click the Loop button. You should end up with a selection that looks something like Figure 8.

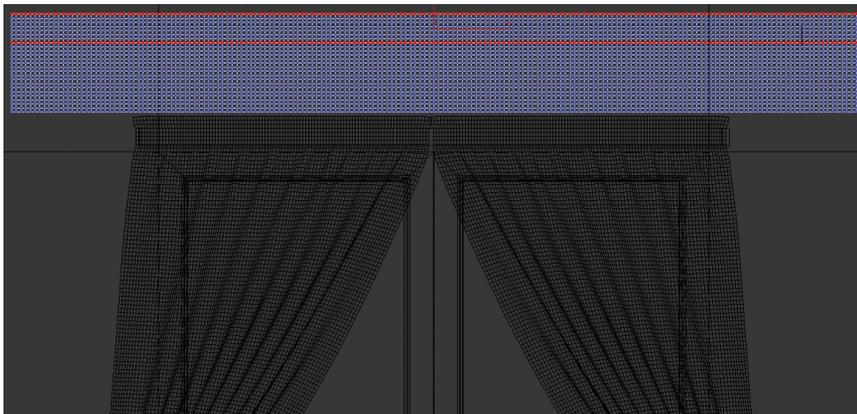


Figure 8: Double row selected verts

- Next we need to select some vertical columns, doing so roughly every 9 to 13 vertices along.

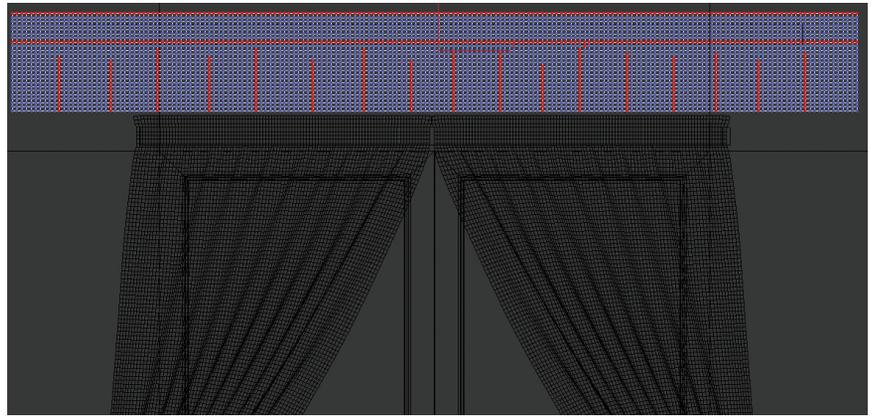


Figure 9: Selected verts

Note: Although we want the selection to come close to the already selected second row of verts, it is better if they don't actually connect up AND if the number of vertices in the column is varied. Once done we should end up with a selection that looks like Figure 9.

This selection pattern has been used to help us create a much more complex set of folds than was needed for the initial curtains.

- Now we can click the 'Make Group' button in the Command panel Group rollout and give it the descriptive name of Ruffle_Node, clicking OK once we are done.

Again we now have an unassigned group.

- To assign a behavior we can click the 'Node' button and then select the GEO_Pelmet_Driver object in the viewport (or again use the H key to pick the geometry from a list).

And with that we have now set up our Pelmet. All that remains is to:

- Exit sub-object mode on the cloth modifier and then click the Simulate button in the Command panel's 'Object' rollout.

Note: Even though the Pelmet plane is less dense (in terms of polygons) than our curtain geometry, because the selection we are using is much more complex we can expect this simulation to take quite a bit longer previous. Ultimately, the end result should be well worth the wait.

FINAL TOUCHES TO THE PELMET

Typically a Pelmet has a front and two sides, effectively boxing off the curtain pole or rail. To save ourselves a little time and to reduce the complexity of the simulations we are working, we have quite obviously been working with just a single flat plane to create our Pelmet.

3ds Max 2012

To fake the box effect somewhat we can perform some simple modeling operations on the pelmet geometry and bend each of the ends into place. To do that let's:

- Take a snapshot of the Pelmet and then delete the simulated object from the scene.
- Before performing modeling operations I like to convert my geometry to an editable poly by right-clicking and choosing the Convert > Convert to Editable Poly command.

Note: This is just a personal preference, not an essential step. We could just as easily add an Edit Poly modifier to the geometry and then perform the following operations from there.

- Next we want to switch to a Top view so let's hit the T key to do that and at the same time make sure that we are in wire-frame mode using the F3 key
- Let's enter 'Vertex' sub-object mode and then marquee select 3 to 5 rows of vertices on the far left of the Pelmet object as shown.

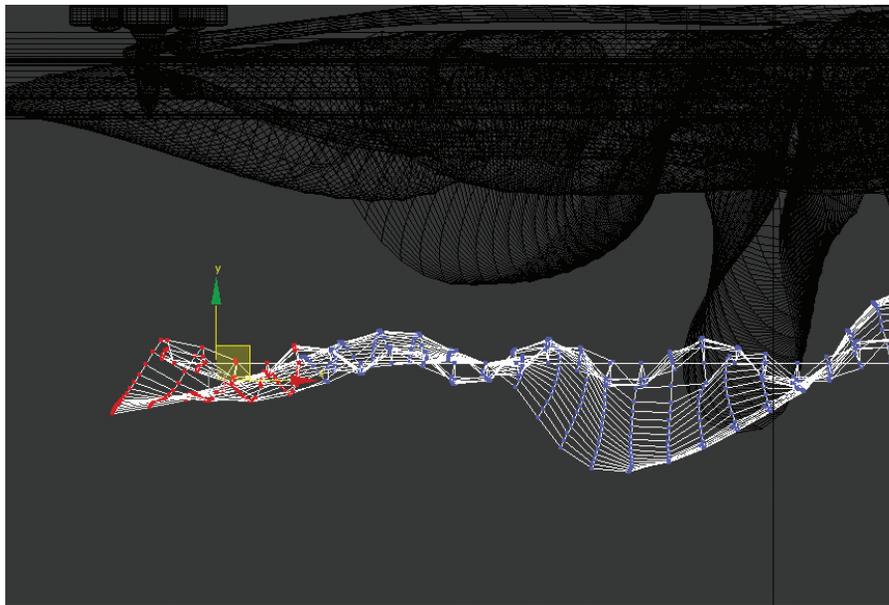


Figure 10 – Selected verts

In the 'Soft Selection' rollout we need to first enable 'Soft Selection' and then set its 'Falloff' radius to 7cm.

With the Rotate tool active, rotate the selected vertices roughly minus 70 degrees on the Z axis, essentially bending them round the corner.

We need to perform the same operation over on the opposite side of the Pelmet so let's select a group of vertices over on the right hand side of the geometry and this time rotate them around the positive Z axis.

Finally, let's move our Pelmet into place using the 'Move' tool, setting it just above our curtain, and then finish off by taking a final render. The end result should look like something like Figure 11.

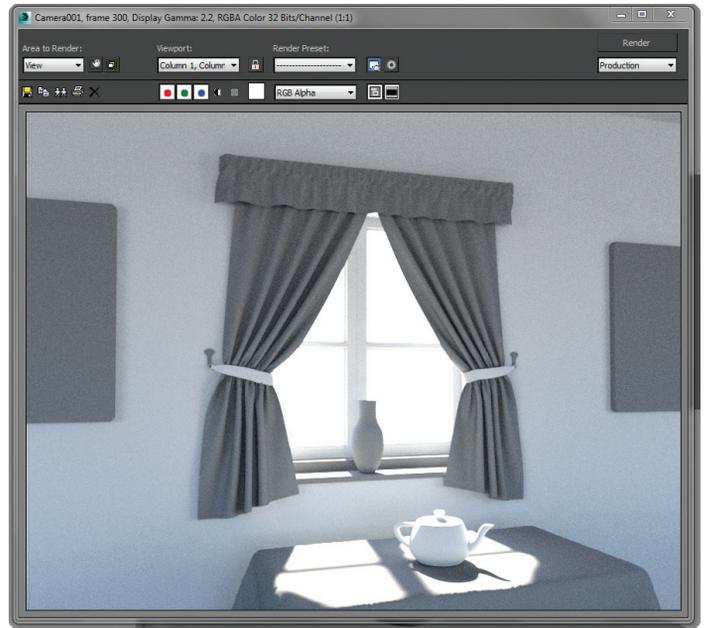


Figure 11: Final render

And with that, we are done. The idea here has been to introduce you to a basic methodology for creating a set of curtains using 3ds Max's Cloth simulation tools. The next step is for you take what we have covered here and experiment with both the settings and the technique to produce something unique and wonderful of your own.



Joel Bradley is a self-taught 3D generalist who has been using 3D and graphic design applications since he was 9 years old, cutting his teeth on programs such as truespace and bryce 3d before moving onto 3ds Max. He has used his acquired skills to work in a number of industry areas including producing content for the web, print, visualization, and multimedia, to name a few. For the past four years, Joel has focused his attention on the production of video training for applications such as 3ds Max, Maya, SketchUp, V-Ray, Photoshop, After Effects, Combustion, Composite, and others.



INTRODUCTION

Let's face it, sometimes working in Autodesk® Revit® can be a chore. There are some tasks that are just plain boring, or repetitive, or both. Renaming sheets, renumbering rooms, and changing family names are all necessary tasks, but they don't exactly make for exciting work, especially if you have to do it more than once.

Fortunately, Revit provides tools for automating these kinds of repetitive tasks. By writing macros, you can custom tailor Revit to the specifics of how you work. You can fine-tune your tools to do your work faster and more accurately.

WHAT IS A MACRO?

A macro is a series of user-created commands and functions that are written using the Revit application programming interface (API) and created in SharpDevelop, an open-source integrated development environment (IDE) integrated into Revit. Macros are stored inside Revit project files and are run directly from Revit. Unlike macros in other applications, Revit macros cannot be recorded. You must write out all the code for your macros.

CHOOSING A LANGUAGE

One of the great things about writing macros in Revit is that you can choose from four different languages to write your code. Since the Revit API is built on the Microsoft .Net framework 4.0, you can write macros in either the C#, Visual Basic .Net, Python, or Ruby programming languages. All these languages compile into the same intermediate language so you have full access to the Revit API regardless of the language you choose. You can build the exact same macro in terms of functionality using any of the supported languages. Choosing a language is largely a matter of personal preference and learning style.

The examples in this article are written using VB.Net, which is a great language for beginning programmers. It evolved out of Microsoft Basic and retains Basic's friendliness toward new programmers. VB.Net code is more forgiving than other languages and is easier to read and understand.

One additional benefit to learning to write macros in Revit is that you can put these new programming skills to use on other aspects of your job. Once you have an understanding of the language, you can write code for other applications or create websites or stand-alone applications. The benefits of learning to write code are vast and will pay you many dividends throughout your career.

Revit 2014

THE REVIT 2014 SDK

To get started, you should first install the Revit 2014 SDK. The SDK (software development kit) contains help files and sample code that will assist you as you learn to program macros. The Revit 2014 SDK can be installed from the main page of the Revit installer or it can be downloaded from the Autodesk Developer Network website: <http://usa.autodesk.com/adsk/servlet/index?id=2484975&siteID=123112>.

The SDK will install on your hard drive and create the following folders and files. Take some time to review the files. The macro samples in particular are useful when you start creating your own macros.

Add-In Manager	9/10/2013 10:20 PM	File folder	
Macro Samples	9/10/2013 10:21 PM	File folder	
Revit Server SDK	9/10/2013 10:21 PM	File folder	
Revit Structure	9/10/2013 10:21 PM	File folder	
RevitLookup	9/10/2013 10:21 PM	File folder	
REX SDK	9/10/2013 10:21 PM	File folder	
Samples	9/10/2013 10:21 PM	File folder	
Structural Analysis SDK	9/10/2013 10:21 PM	File folder	
Autodesk Icon Guidelines.pdf	4/24/2013 8:28 AM	Adobe Acrobat D...	4,960 KB
Getting Started with the Revit API.docx	4/24/2013 8:28 AM	Microsoft Word D...	173 KB
Read Me First.doc	4/24/2013 8:28 AM	Microsoft Word 9...	40 KB
Revit Platform API Changes and Addition...	4/24/2013 8:28 AM	Microsoft Word D...	84 KB
RevitAddInUtility.chm	4/24/2013 8:28 AM	Compiled HTML ...	219 KB
RevitAPI.chm	4/24/2013 8:47 AM	Compiled HTML ...	39,028 KB
RevitAPI.chw	9/18/2013 1:42 PM	CHW File	4,734 KB

Figure 1: The SDK folder

YOUR FIRST MACRO!

Ready to create your first macro? As you'll see, the process is very straightforward. Follow these steps and you'll be on your way to macro mastery.

1. Open the Macro Manager

In Revit 2014, create a new project file. Once you have created the file, open the Macro Manager by selecting the Manage ribbon and clicking the Macro Manager icon.

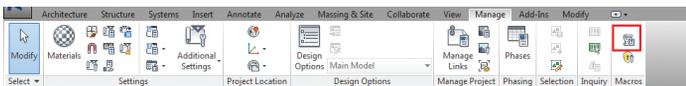


Figure 2: The Manage ribbon

Macros can reside either in the active project file or within the application. Macros saved within the project file can be used by any user who opens that file. Macros saved in the application are saved to the user's Revit configuration. These macros can be used on any model file, but only by the user who created the macro.

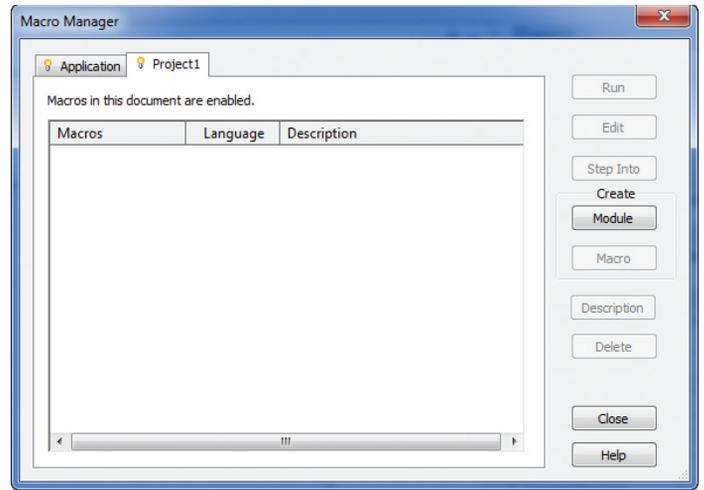


Figure 3: Macro Manager

2. Create a New Module

To create a macro, you must first create a new module. A module is simply a collection of macros. Each macro within the module must have a unique name and all the macros within the module must compile without errors.

Click the "Project 1" tab then click the Module button in the "Create" section. In the "Create a New Module" dialog box, title your module "MyFirstModule." Choose VB.Net as the module's language then click OK to create the module.

Once Revit has created the module, SharpDevelop will launch in the background. SharpDevelop is the integrated development environment that's built into Revit for programming macros.

3. Create a New Macro

Now that we have a module, we'll create a macro to go in the module. Make Revit the active application and click the Macro button in the "Create" section of the Macro Manager dialog. In the Create a New Macro dialog, title your macro "MyFirstMacro" and set the language to VB.Net. Click OK to create the macro.

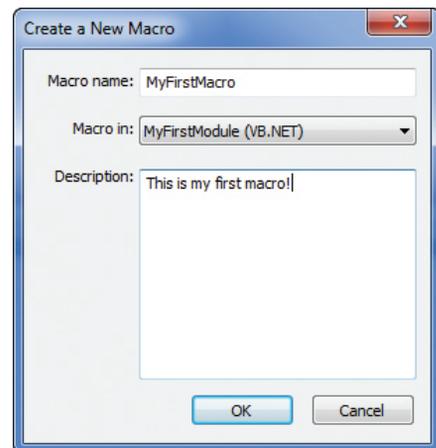


Figure 4: Create macro

4. Write Your Macro

Back in the Macro Manager dialog, select “MyFirstMacro” from the list and click the “Edit” button. This will bring you to SharpDevelop. In the SharpDevelop code window, you’ll see standard code that is automatically created. Scroll to the bottom of the screen and you will see the starting code for “MyFirstMacro.”

```

1  *
2  * Created by SharpDevelop.
3  * User: michael
4  * Date: 2/1/2014
5  * Time: 9:37 PM
6  *
7  * To change this template use Tools | Options | Coding | Edit Standard Headers.
8  *
9  Imports System
10 Imports Autodesk.Revit.UI
11 Imports Autodesk.Revit.DB
12 Imports Autodesk.Revit.UI.Selection
13 Imports System.Collections.Generic
14 Imports System.Linq
15
16 <Autodesk.Revit.Attributes.Transaction(Autodesk.Revit.Attributes.TransactionMode.Manual)> _
17 <Autodesk.Revit.DB.Macros.AddInId("4FE34658-3A83-4897-9A1B-760E5B0712F8")> _
18 Partial Public Class ThisDocument
19
20     Private Sub Module_Startup(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Startup
21
22     End Sub
23
24     Private Sub Module_Shutdown(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Shutdown
25
26     End Sub
27
28     Public Sub MyFirstMacro()
29
30     End Sub
31 End Class
32

```

Figure 5: Startup code

Your first macro is going to simply pop-up a message box in Revit. It only takes one line of code. Between “Public Sub MyFirstMacro()” and “End Sub”, type the following:

```
TaskDialog.Show("My First Macro","This is my first macro!")
```

5. Build Your Macro

Once you’ve typed the code, you’re ready to build the macro. All macros must be built or compiled before Revit can run them. In the SharpDevelop pull-down menus, select “Build” then “Build Solution.”

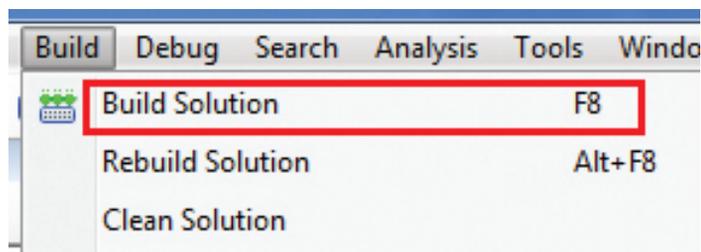


Figure 6: Build Solution

Any errors in your code or warnings will be highlighted in the Error and Warnings window located at the bottom of the SharpDev interface. If you don’t see the Errors window, click the “View” pull-down then select “Errors” from the list.

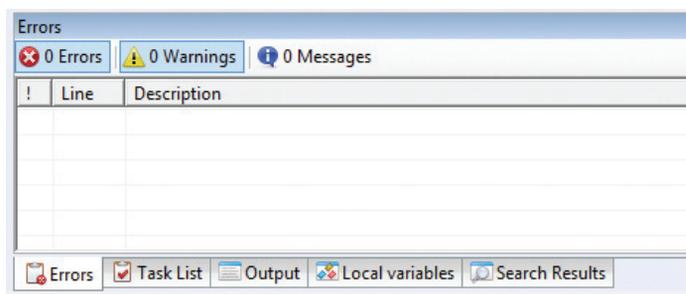


Figure 7: Errors window

If you have an error, double check your code and make sure your spelling and syntax are correct. The code window will list errors by their line number so they’re generally easy to pinpoint.

6. Run the Macro

Revit will only run macros that have compiled without any errors. If your macro compiled correctly, go back to Revit and open the Macro Manager dialog (Manage > Macro Manager). You should see “MyFirstMacro” in the list for “MyFirstModule.”

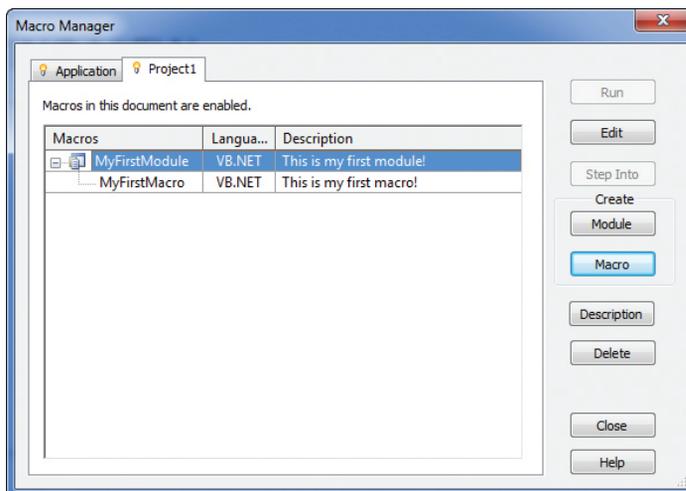


Figure 8: The Macro Manager dialog

Select “MyFirstMacro” then click the Run button. This will execute your macro. You should see the following on your screen:

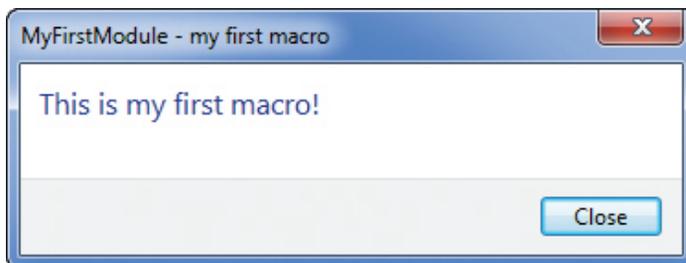


Figure 9: My First Macro

Revit 2014

You did it! You wrote and compiled your first macro. To take this further, you can modify the code to report something more useful. If you want the code to report the current file's filename, change the code to the following:

```
TaskDialog.Show("My First Macro", "The current filename is " & Me.Document.Title)
```

The "Me.Document" object contains data pertaining to the current model file. For a variation, change "Me.Document.Title" to "Me.Document.ActiveView.ToString" to see the current project file's active view.

MORE MACROS

Our first macro was useful for illustrating the process for creating a macro, but let's take what we just learned and put it to use on a macro that actually does something useful. Here's the scenario for the next macro: You're adding a new sheet into a sequence of sheets and you need to do some renumbering. You could just do this manually, but it's pretty tedious and you might miss a sheet in the process. Instead of doing it manually, let's write a macro that gets the current sheet's number and increments it by one automatically.

One of the first things we need to do when writing this macro is import the Microsoft.VisualBasic namespace. This will allow us to use certain Visual Basic functions, such as IsNumeric. Since sheet numbers can contain letters, we need to test the last three digits of the sheet number to make sure it contains only numbers.

```
Imports System
Imports Autodesk.Revit.UI
Imports Autodesk.Revit.DB
Imports Autodesk.Revit.UI.Selection
Imports System.Collections.Generic
Imports System.Linq
Imports System.Diagnostics
Imports Microsoft.VisualBasic
```

Figure 10: Imports statement

Any time we make a change to the model via code, we need to create a transaction. A transaction creates a lock on the model so no other changes can occur while your change is taking place. The transaction code is very straightforward. It must wrap around the code that is making the change to the model. If you try to run your macro without the transaction code, you'll get an error. Here's what the transaction code looks like:

```
'create transaction
Using t As New Transaction(Me.Document, "your transaction name")
    If t.Start = TransactionStatus.Started Then
        'make your changes here

    End If

'commit changes
t.Commit

>alert the user that sheet has been incremented
TaskDialog.Show("Alert", "Hey, your changes were made!")

End Using
```

Figure 11: Transaction code

Note that any line beginning with an apostrophe is a comment line. These types of lines are not compiled by SharpDevelop. Comment lines are used to describe the macro and to serve as a reminder for what the code does.

The flow of the sheet renumbering macro goes like this: The code starts by determining if the current view is a sheet. If it is, the code gets the last three digits of the sheet number. It checks if the last three numbers are really numbers. If they are, the code creates a new variable that is the last three numbers plus one. The code then creates a transaction and updates the current sheet's sheet number. As a last step, the macro alerts the user that the sheet number has been updated.

To test this yourself, create a new macro and type the following code:

```
Public Sub IncrementCurrentSheet()
    'variables
    Dim curSheet As ViewSheet
    Dim curSheetNum As String
    Dim lastNum As String
    Dim newNum As Integer
    Dim newSheetNum As String
    Dim incValue As Integer
    Dim curView As View

    'set variables
    incValue = 1
    curView = Me.ActiveView

    If TypeOf curView Is ViewSheet Then
        'current view is a sheet so get sheet number
        curSheet = curView
        curSheetNum = curSheet.SheetNumber

        'get last three digits of sheet number
        lastNum = curSheetNum.Substring(curSheetNum.Length-3)

        'check if last digit is numeric, if so then increment
        If IsNumeric(lastNum) Then 'requires importing Microsoft.VisualBasic namespace
            'increment sheet number
            newNum = CInt(lastNum) + incValue

            'create new sheet number using incremented number
            newSheetNum = curSheetNum.Substring(0,curSheetNum.Length-3) & newNum.ToString

            'create transaction and update sheet number
            Using t As New Transaction(Me.Document, "update sheet number")
                If t.Start = TransactionStatus.Started Then
                    curSheet.SheetNumber = newSheetNum
                End If

                'commit changes
                t.Commit

                'alert user that sheet has been incremented
                TaskDialog.Show("report", "Incremented sheet " & curSheetNum & " to " & newSheetNum)
            End Using
        Else
            'alert user that sheet number is not numeric
            TaskDialog.Show("report", "Last three digits of sheet number are not numbers. Cannot increment this sheet.")
        End If

    Else
        'alert the user that the current view is not a sheet
        TaskDialog.Show("report", "Please make the current view a sheet.")
    End If

End Sub
```

Figure 12: Increment sheet code

Some variations of this macro would be to automatically renumber all the sheets in the project file instead of just the active sheet. Another option would be to renumber a specific, user-defined series of sheets, such as only the A2 or M5 sheets.

Another useful macro is one that converts model lines to walls. This macro shows you how to prompt the user for information and how to create Revit objects. The macro also uses element filtering, which is how the API collects objects of a certain type. Here's the code:

```
Public Sub WallsFromLines()
    'prompt user to select level for new walls
    Dim curLevel As elementId
    curLevel = Me.Selection.PickObject(ObjectType.Element, "Select level for new walls").ElementId

    'create collection of all line elements in the current project file
    Dim curCollector As New FilteredElementCollector(Me.Document)
    curCollector.OfCategory(BuiltInCategory.OST_Lines)

    'loop through the elements - if element is a model line then create wall
    For Each curElem In curCollector.ToElements
        If curElem.Name Like "Model Lines" Then
            'convert element to model line
            Dim curLine As ModelLine
            curLine = curElem

            'create transaction and insert wall using current wall type
            Using t As New Transaction(Me.Document, "add new wall")
                If t.Start = TransactionStatus.Started Then
                    'create new wall
                    Dim newWall As Wall
                    newWall = Wall.Create(Me.Document, curLine.geometrycurve, curLevel, False)
                End If

                'commit changes
                t.Commit
            End Using
        End If
    Next
End Sub
```

Figure 13: Walls from lines

The code starts by prompting the user to select a level object. The selected level will be the level onto which the new walls are created. The `Me.Selection.PickObject` command prompts the user and puts the selected level object in the `curLevel` variable. The code then creates a collection of all line elements in the project using the `FilteredElementCollector`. You can use `FilteredElementCollectors` to create selections of any object type in Revit. Once the lines are collected, the code loops through each line and checks if it is a model line. If it is, the code creates a transaction and uses the current line's geometry to create a new wall. Once the wall is created, the code commits the changes then moves on to the next line.

To expand this macro, you could have the code create different wall types for different line types. You could also program it to automatically create the walls on a specific layer rather than prompting the user for the level. A third variation could use a linked AutoCAD® drawing rather than model lines.

GOING FURTHER

So what other tasks can you automate with Revit? Pretty much anything you can think of! Good candidates for macro automation are tasks that are fairly standardized or require lots of data input. Some examples could include:

- Update all window family instances with manufacturing data from a spreadsheet.
- Check that all the doors in fire-rated walls are actually fire-rated doors.
- Insert components based on coordinate data in a spreadsheet.
- Rename all custom families with a specific prefix for your company before sharing the model.
- Automatically place specific view types on a sheet.

That's just a short list of tasks that could be automated. What tasks do you do on a regular basis that you would like to see automated? Could you write a macro that would do most, if not all, of the tedious work?

CONCLUSION

Learning to write macros in Revit can drastically improve your efficiency with the software. A well-written macro can complete a once tedious task with the click of a button. As your programming skills grow, so will your ability to do the seemingly impossible. Programming macros takes time and patience, but it's a rewarding endeavor. Don't be intimidated by the code. Just work systematically and take small steps and you'll be well on your way to macro mastery.

ADDITIONAL RESOURCES

In addition to the information provided in the Revit 2014 SDK, you can find many online resources for learning the Revit API. Here is a list of some websites and resources I've found to be helpful:

Blogs

- Boost Your Bim – <http://boostyourbim.wordpress.com/>
- The Building Coder – <http://thebuildingcoder.typepad.com/>
- Spiderinnet – <http://spiderinnet.typepad.com/blog/>
- The Proving Ground – <http://wiki.theprovingground.org/revit-api>

Online Forums

- AUGI – <http://forums.augi.com/forumdisplay.php?218-Revit-API>

Online Courses

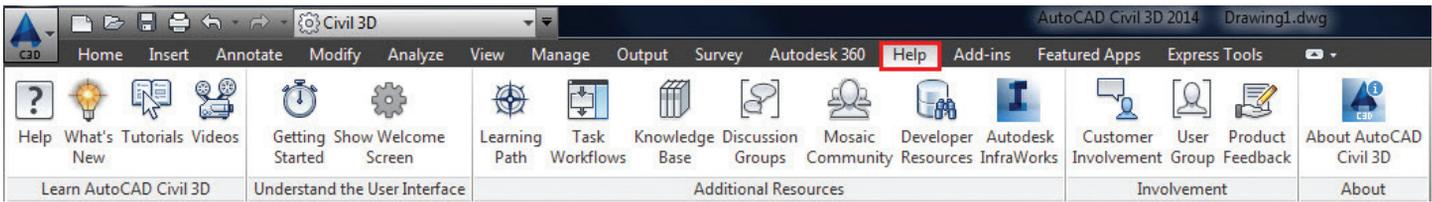
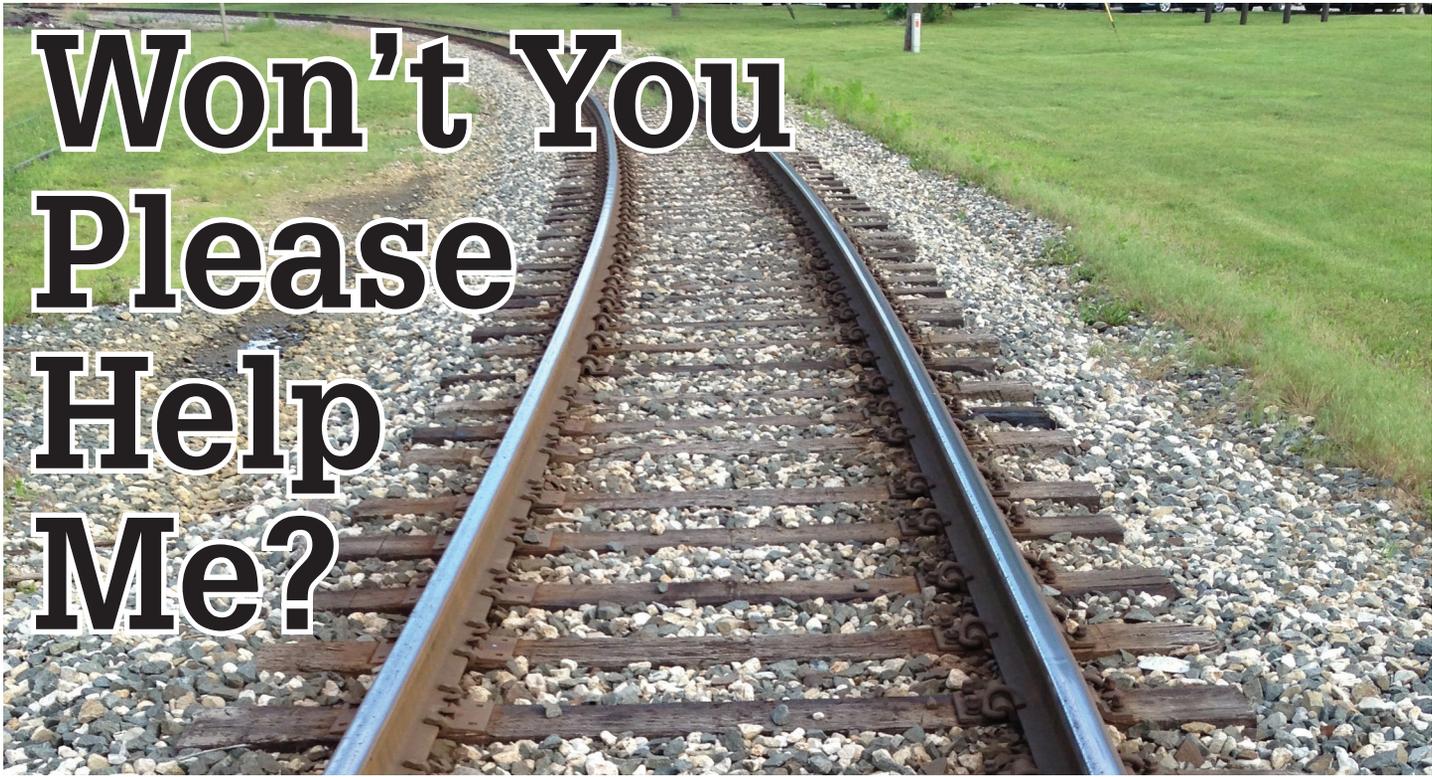
- Learn to Program the Revit API – <https://www.udemy.com/revitapi/>

Books

- Autodesk Revit 2013 Customization with .Net How-to by Don Rudder



Michael Kilkelly AIA is a principal at Space Command (www.spacecmd.com), an architecture and consulting firm in Middletown, Connecticut. Prior to founding Space Command in 2012, Michael was an associate at Gehry Partners in Los Angeles. Michael received his B.Arch from Norwich University in 1995 and his SMArchS from MIT in 1999. Michael writes about architecture and technology on his blog at www.archsmarter.com. You can contact Michael at michael@spacecmd.com.



As a CAD manager, I often have to mentor and train the technical staff on how to use the basic functions of AutoCAD® and AutoCAD® Civil 3D®. I don't have all the answers so I rely on the Autodesk Community, blogs, and the programs' Help feature for assistance. One thing that has become clear to me is that we no longer receive the books and pdf files from Autodesk with all of that good reference material that fills out our office shelves.

Remember the days when Help was not on the ribbon? You would have to select from the menu pull-down or type help at the command prompt. When was the last time you took a look at the Help tab on the ribbon? I know for me, I was pleasantly surprised at all of the resources available when I selected that tab. Let's take a tour through the Help tab on the ribbon and see if there is anything there that you are missing.

HELP

The first section on the ribbon is the Help button. This will bring up the online help menu from Autodesk (You can download the offline help as well.) Selections are Contents, Search, and Favorites, as shown in Figure 1.



Figure 1: Civil 3D Help

WHAT'S NEW

With each release of Civil 3D comes a variety of new information and add-ons that can be reviewed upon installation of the software. The section named "What's New" will launch the splash screen where you can find information on all of the new features in the software. I find it extremely helpful to run through all of the documentation to see if there is anything that can help you with workflow and your daily activities. You will also find short tutorials including video of how to use some of the more detailed features of the program. Figure 2 below is a video example showing how to create a pressure pipe network in Plan View.

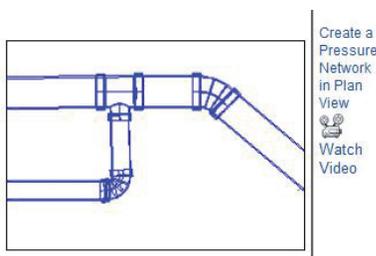


Figure 2: What's New

The What's New Section contains jump categories such as Design, Data Exchange, Analysis, Survey, Railway Design, Corridor Design, Annotation, and Visualization.

TUTORIALS

The Tutorial section will walk you through some main points in Civil 3D. I like to take a look at the Points, Surfaces, Profiles, and Grading sections first, but going through the entire list is good practice. Figure 3 shows a portion of the tutorial section in Civil 3D where you can access various types of tutorials.

Welcome to the AutoCAD Civil 3D

The following tutorial sets are included with AutoCAD Civil

- [Getting Started](#). Learn about the application workspace
- [Points](#). Learn how to work with coordinate geometry (C land surfaces.
- [Point Cloud Tutorials](#). Learn how to work with point cloud are obtained from LiDAR scanning
- [Surfaces](#). Learn the basics about creating and working
- [Survey](#). Learn the basics about importing, creating, ma
- [Project Management](#). Learn how to use the AutoCAD C data shortcuts and Autodesk Vault.
- [Alignments Tutorials](#). Learn about alignments, which ar
- [Profiles](#). Learn the basics about viewing and designing alignment.
- [Parcels](#). Learn the basics about creating and editing pr parcels.
- [Grading](#). Learn how to design the finished grade for lan retail sites.

Figure 3: Welcome to AutoCAD Civil 3D

VIDEOS

Who doesn't like to grab some popcorn and watch a good Civil 3D video? This section provides videos (some with audio) to help your learning experience, as shown in Figure 4.

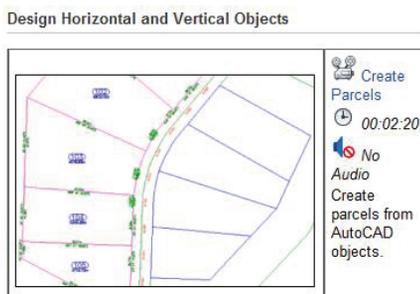


Figure 4: Welcome videos

GETTING STARTED

These tutorials, designed for beginners, start with the application workspace and some important design tools and tasks. From Help: "In Civil 3D®, design data is organized as object collections in the Toolspace window. In Toolspace, the Prospector tab displays the hierarchy of design objects, such as points, surfaces, and alignments. The Settings tab displays a hierarchy of object styles, including styles for labels and tables." These videos are more for the entry level user to get acquainted with the interface, but it's always good to take a look back and refresh your skills.

WELCOME SCREEN

Have you ever heard the question, "Whatever happened to that Welcome Screen I turned off when I installed Civil 3D"? Typically you can type welcome screen at the command prompt and get it back. The Civil 3D 2014 Welcome screen isn't a new feature, but it has changed over the years. It may be a tool that you want to visit from time to time. It provides access to helpful bits of information as well as links to communication tools. Figure 5 shows a portion of the welcome screen as it appears when you launch the program.

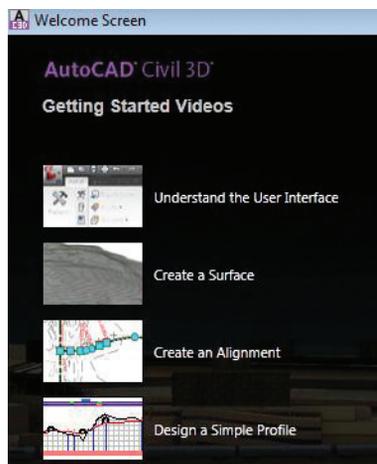


Figure 5: Welcome screen

LEARNING PATH

Learn, Connect, Explore. That was the theme of Autodesk University 2014. The learning path section is where you can find all resources tied into one menu. A good start for beginners or even long-time users to take a look at some features of Civil 3D you may not know about.

TASK WORKFLOWS

When I began my class at AU2013, I mentioned workflow. I wanted to provide examples of macros, but relate those to workflow and how you go through a project. This section does the same thing, going through several topics and figure out how you would complete each task as if it were assigned to a project. Figure 6 shows some examples of the workflow section of help.



Figure 6: Task workflows

KNOWLEDGE BASE

This section displays the Civil 3D support website. Figure 7 below shows where you can run a knowledge base search and move to specific product information.

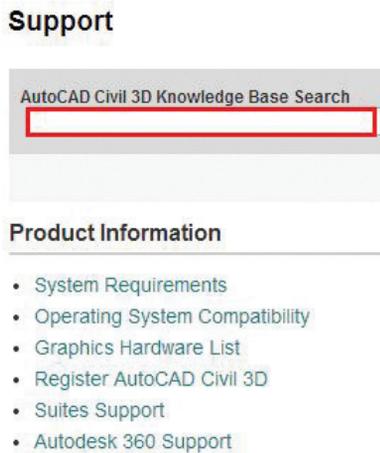


Figure 7: Knowledge Base support

DISCUSSION GROUPS

Civil 3D is a powerful civil engineering tool incorporating a dynamic engineering aspect to design. Discussion groups help you find people who have run into a problem similar to yours and have found a solution. These groups connect people from all over the world in one place to collaborate on Civil 3D design problems and solutions. This section takes you to the Autodesk Community (Figure 8) where you can use forums, read articles, and search ideas collaborating with your peers.

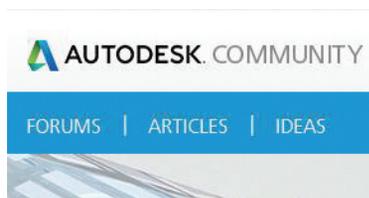


Figure 8: Autodesk Community site

MOSAIC COMMUNITY

When selecting Mosaic Community, you will be taken to the Mosaic website where you will see the following statement: “Thank you for visiting. In an effort to consolidate our community and social efforts to more popular networks, we will be migrating content sharing to Facebook, Twitter, and YouTube on January 31, 2014.” The time has come to say goodbye to Mosaic and hello to Project Chronicle. Figure 9 shows an image from Autodesk Labs where Project Chronicle first began as an Autodesk Research project.

AUTODESK LABS: PROJECT CHRONICLE



Figure 9: Chronicle

Autodesk Labs, part of the Office of the Chief Technology Officer (CTO) Scott Sheppard, is a public process for sharing innovative new technologies and collaborative development. If you are unfamiliar with Autodesk Labs you can start by visiting Scott’s blog at http://labs.blogs.com/its_alive_in_the_lab/autodesk-labs/

Here you will find all the information on what projects are being worked on, as well as some past successes. Project Chronicle is a free technology application from AutoCAD that allows users to create videos and share files (Chronicles) with other users. These videos are stored on a secure website where they can be shared publically or made private for you to share with your company or in a classroom environment.

The Chronicle recording utility as shown in Figure 10 allows users to capture workflow from within Autodesk products. Simply hit the Record button and you are ready to go. So when you have that next great idea, write it down or make a Chronicle and publish it online for future use. How many times have we done something we wish we would have recorded so when that scenario comes up again we’ll have an online resource to rely on? Project Chronicle is the answer. It is an extremely useful tool where experts can showcase their skills, allowing others to get the help they need when working with Autodesk software.



Figure 10: Chronicle Recorder

AUTODESK INFRAWORKS

If you attended Autodesk University 2013, you certainly noticed quite a few classes on Infracworks. This button will take you to the About section of Infracworks on the web. This product is very powerful, combining the use of Civil 3D, GIS, Mapping, and other tools to provide stunning visuals for presentation and production.

DEVELOPER RESOURCES

The Autodesk Developer Network section will take you to the topics used by developers and lists the APIs directly related to enhancing the functionality of Civil 3D. There are three APIs available for customizing AutoCAD Civil 3D, and are listed below and on the developer resources website:

1. .NET API allows you to write plug-ins to AutoCAD Civil 3D in any .NET language.
2. COM API can be used in creating VBA macros, which you can run from the AutoCAD Civil 3D command line.
3. Custom Draw API (in C++) is an extension of the AutoCAD ObjectARX API that allows you to customize the way AutoCAD Civil 3D renders objects.

CUSTOMER INVOLVEMENT

Remember the image shown below (see Figure 11)? Yes, the one you unchecked after installing the software indicating that you don't want Autodesk to collect data? If you select yes, you agree to participate in the Autodesk Customer Involvement Program (CIP). Data and system usage is collected by Autodesk and reviewed, which helps them personalize the experience with the software and improve future releases of the product. Simply check Yes as shown in Figure 11.

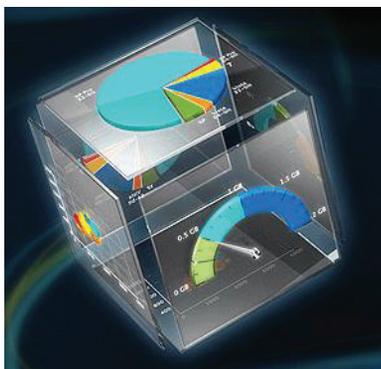
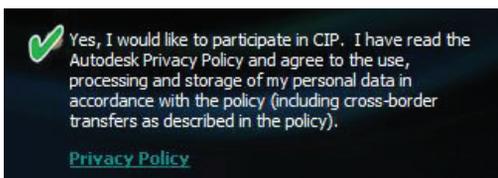


Figure 11: Customer Improvement Program



USER GROUPS

User groups can be a valuable resource for those day-to-day problems we all encounter during projects. Find a group, join, and collaborate with your peers on the issues with the software you use. Meet new people who share your same interests and passions. Find those people you can talk with about AutoCAD, Civil 3D, Plant 3D, Infracworks and other techy things, or just hang out.

PRODUCT FEEDBACK

As with many items we purchase, we can provide feedback with Autodesk products. This will give us the opportunity to voice our opinion on the stability, ease of use, and overall functionality of the product(s) we use. All of this data, as with customer involvement, will be reviewed and taken into account to improve the next release of the software.

CONCLUSION

In life and in the workplace it can be hard to ask for help—even during those times when deep down we know we need it. Pride and ego can get in the way. Helping others in the CAD world or life in general will make us all sleep better at night.

For many years (too many to count), I was that guy who sat in the corner office and tried to figure things out on my own. Then one day someone offered me help and I accepted. It was then I realized that there are so many good people and resources to assist us in our daily routine. You don't have to take this journey alone. Take that leap of faith and trust you will find the right person or place to go to. Use the resources available and network with colleagues to become a better person, designer, and overall good guy or gal.

Using the help functions in Civil 3D can only benefit you as a professional in your career and give you the ability to mentor and provide guidance to those who need it. It will be a rewarding experience for you and the people you meet along the way.



Sam Lucido is a CAD Services Manager with Haley and Aldrich, Inc. He has over 20 years of experience involving design, user support, and customization. Sam is professionally certified in AutoCAD 2011-2014 and recently spoke at AU2013. He uses his vast knowledge about AutoCAD to help provide support to engineering and design teams with monthly tip sheets and WebEx trainings. You can find him at CADProTips.com and he can also be reached at slucido@haleyaldrich.com

Schedule Tables in

ACA

Schedules are tables you can insert in drawings to list information about selected objects in your building model. Objects are made up of properties that contain data. Schedule tags provide an efficient tool for collecting the property data attached to the objects for display in a schedule table.

You can create schedules with varying levels of detail by defining and attaching sets of properties to object styles or to individual objects and then extracting and displaying the data in a schedule table. You can produce basic schedule tables using the default tools provided with the software.

OVERVIEW

Before delving into schedule tables, it is important to understand some of the terminology associated with them. Here's a little overview of some important terms that apply to creating and managing schedule tables.

- Schedule Tags** – You can use project-based or standard schedule tags in your drawings to graphically display the property data of an object. By linking the schedule tag to a property in a property set, you report property data of the object. When you anchor the tag to an object to which the property set is applied, the value of the property displays in the tag. The information in the tag is updated if the object or if the property changes.

- **Schedule Tools** – AutoCAD® Architecture provides default tools for project-based and standard wall, door, and window schedules on the Scheduling tool palette and in the Content Browser. Selecting one of these tools that has a style and other properties predefined allows you to quickly place a schedule table in your drawing.
- **Schedule Styles** – A schedule table style specifies the properties that can be included in a table for a particular object type. The style also controls the table formatting, such as text height and spacing, columns and headers. Display properties in the style control the visibility, layer, color, linetype, and linetype scale of table components.
- **Property Sets** – A property set is a user-definable group of related object properties. When you attach a property set to an object or a style, the property set becomes the container for the property data associated with the object. Property sets are specified using property set definitions.
- **Property Set Definitions** – A property set definition is a documentation object that specifies the characteristics of a group of properties that can be tracked with an object or style. Each property has a name, description, data type, data format, and default value.
- **Property Data Formats** – A property data format is specified for each property definition within a property set definition to control how the data for that property displays in a schedule table, in a schedule tag, or on the property palette. Property set definitions and schedule table styles use property data formats to control the display format of values for each property.

- **Default Format** allows you to specify the format you want for your new schedule table style. This includes text appearance, matrix symbol, and cell size.
- **Applies To** allows you to specify which objects you want the schedule table style to track. This could be as simple as a polyline or a door. This could also be several ACA objects, depending on what information you wish to include in your schedule table.
- **Columns** allow you to add columns to represent properties that are reported in the schedule table style. You can also add column headings, edit column data, and edit column placement in your style (see Figure 1).
- **Sorting/Grouping** allows you to specify the sort order of each row within the schedule table style. You can also group columns together with this feature and specify if you would like to display subtotals for the group.
- **Layout** allows you to specify the format of the table title, the column headings, and the matrix column headings.
- **Classifications** allow you to assign a group of named properties to various objects. They assist in controlling how objects are displayed and scheduled.
- **Display Properties** allows you to specify the visibility, line type, layer, and other display properties of the schedule table style you are creating.

CREATING SCHEDULE TABLE STYLES

Schedule table styles are used to control the appearance and the content of schedule tables. A schedule table style for the type of schedule table you want to create must be contained in the drawing. When a schedule table style is copied into a drawing, data formats and property set definitions specified in the style are also copied. Property data formats and property set definitions will be discussed shortly.

Like many entities of ACA, schedule table styles are created and edited in the Style Manager under the Manage tab of the ribbon. To create a new style, expand Documentation Objects, right-click on Schedule Table Styles, and click new. Enter a name for the new style and hit enter.

Next, you will edit the options for the schedule table style. The eight tabs you have to choose from are as follows:

- **General** is where you would add a description, if desired. You can also click on Notes and add a note and/or a reference document.

Once your style has been created, you can drag and drop it onto your tool palette for quick access. You can also add the schedule table to the Annotation tab on the ribbon by using the CUI. I highly recommend doing this if you plan to use your new schedule table style frequently.

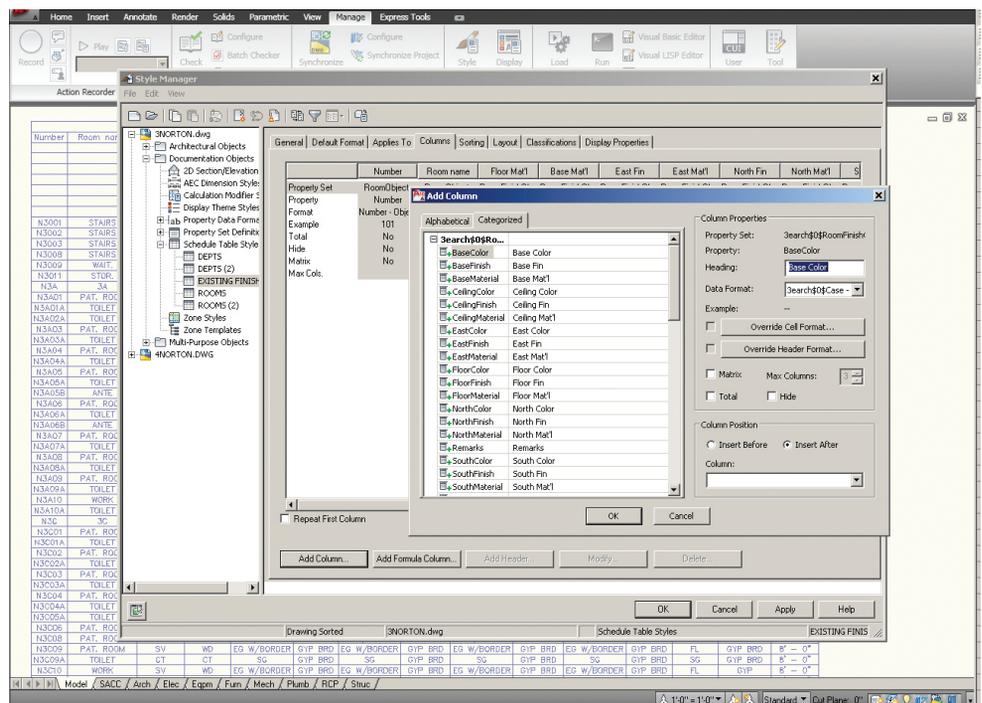


Figure 1: Schedule Table Style – Add Column

PROPERTY DATA FORMATS AND PROPERTY SET DEFINITIONS

Before you create a schedule table, you will need to attach the property sets that are referenced in the schedule table style to the objects and object styles. These attached property sets become the containers for the data that will appear in your schedule table. A schedule table extracts the data from objects and displays it in the table. Data is not saved in the table itself.

Property set definitions are created and edited in the Style Manager under the Manage tab of the ribbon. To create a new property set definition, expand Documentation Objects, right-click Property Set Definitions, and click new. Enter a name for the new definition and click enter. As with the Schedule Table Style, you will want to check which entities your new Definition Applies To. Now you will want to click on the Definition tab and add Property definitions as needed (see Figure 2). Click Apply and OK when you are finished. Property Set Definitions are added to objects through the properties palette, extended data tab.

A property data format is applied to each definition within a property set definition. Property data formats are created and edited in the Style Manager under the Manage tab of the ribbon. To create a new property data format, expand Documentation Objects, right-click Property Data Formats, and click new. Enter a name for the new format and click enter. Now, you will want to click on the Formatting tab. Here, you will

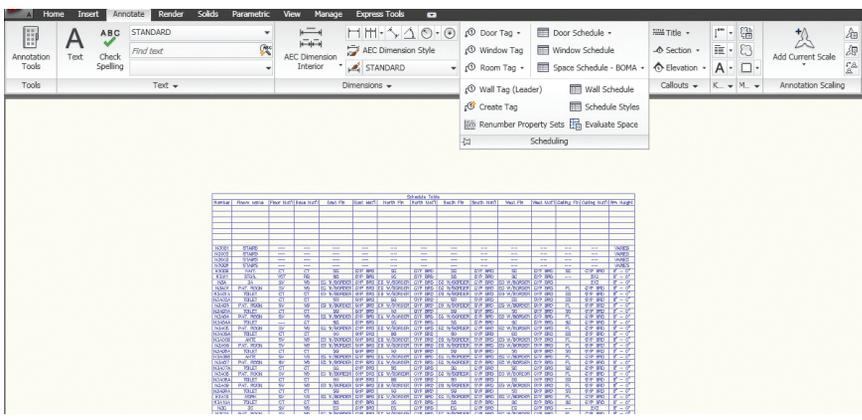


Figure 3: Annotation ribbon

need to specify how you want the formatting to appear. Fill in all information pertinent to the format you are creating. Click Apply and OK when you are finished.

INSERT A SCHEDULE TABLE

Begin by opening the Annotation tab of the ribbon or opening your tool palette (depending on where you placed your new schedule table style) and selecting the Schedule Table you created. Next, select the objects you wish to include in the schedule table or you can press enter to schedule an external drawing. Objects selected that are not of the type specified for the schedule table will automatically be filtered out of the drawing. Next, you will need to specify in the drawing area the insertion point for the upper-left corner of the table, or you can press enter to scale the schedule table to the current drawing scale (see Figure 3).

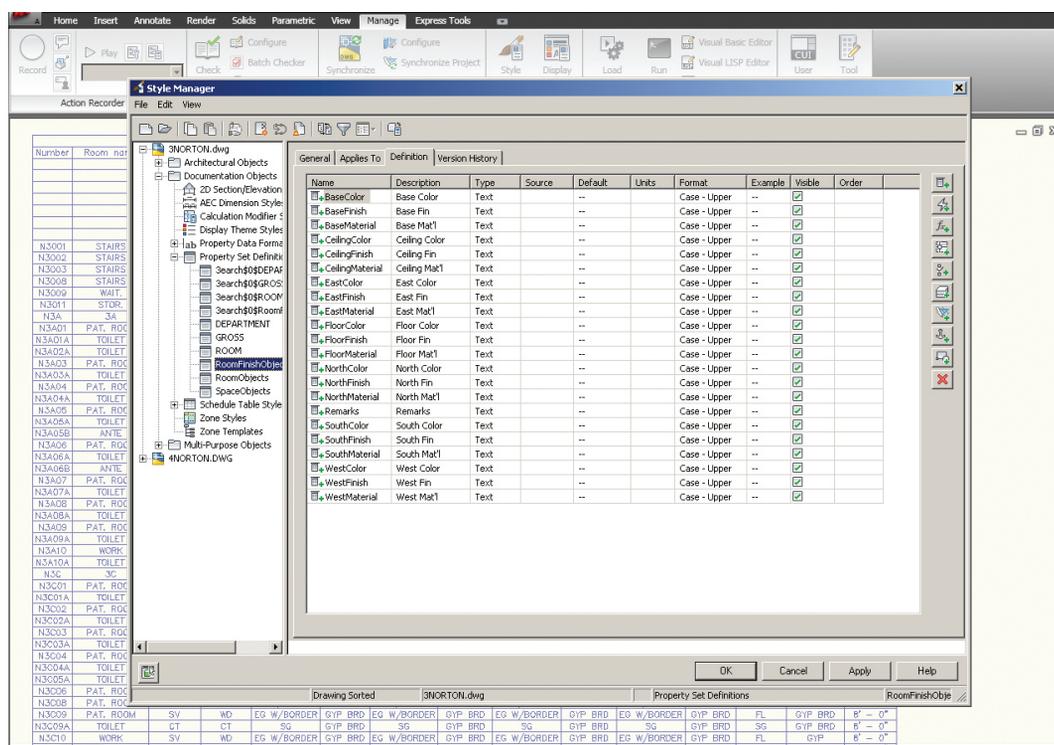


Figure 2: Define a property data format

If your schedule table contains question marks in any of the cells, the property set definition that contains that property is not attached to an object or object style. If you have empty cells or dashes within cells, this indicates that the property set definition is attached, but data is either not available or is not entered for that object or object style.

It is important to note that property data formats, property set definitions, and schedule table styles cannot be changed through RefEdit. Changes made through RefEdit seem to work, but the drawing will revert to the previous settings when saved back to the xref file. If you are using an xref file and need changes to be made to the schedule table, you will need to open the xref drawing directly and make changes there.



Tips, Tools & Workarounds

Part of the beauty of Autodesk® Revit for me is its unparalleled ability to allow practically anything to be created. As a detailer, I use Revit for geometry. The accurate size and placement of steel components is paramount to the success of my business. The speed and agility with which I can do this is also of key importance. That being said, my Tips & Tricks rendition will be slanted toward the creation or modification of components to achieve the geometry required. It will also use techniques recently made possible in Revit 2014 (i.e., double-clicking an element to edit it, a task not as easily performed in earlier releases of Revit).

WALLS

During the process of logging more than 15,000 hours in Revit, I have learned that there are ways to modify Revit elements... then there are ways to modify them easily. Take walls, for example. My first inclination was to use Revit's wall opening tool (Structure>Opening>Wall) because, after all, I wanted to cut an opening in a wall.

What I quickly discovered with this tool is I basically have one shot to get it right, or I will spend some time finessing the opening to achieve the correct size and placement. This involved a lot of added dimensions, lines, alignment tool, etc. WHEW! Finally... I have a wall opening. Nine more to go.

The EASY way to create wall openings is to simply Edit the wall. Double-click the wall, and simply draw the openings you want, as many as you want, in whatever shape you want, and close the wall and you have your openings. The benefit of creating wall openings this way is that you can manipulate the lines using the same commands on the Annotate and Modify panels that you would use if you were drafting the wall and its openings in 2D. Make a mistake? Simply Edit the profile and change it. This technique can also be used for shaping the top, sides, or bottom of a wall, as long as it is straight, but it will give a warning on how to properly control the height and depth of walls. If your wall is curved in plan view, not to worry, I will cover editing the top of that wall later.

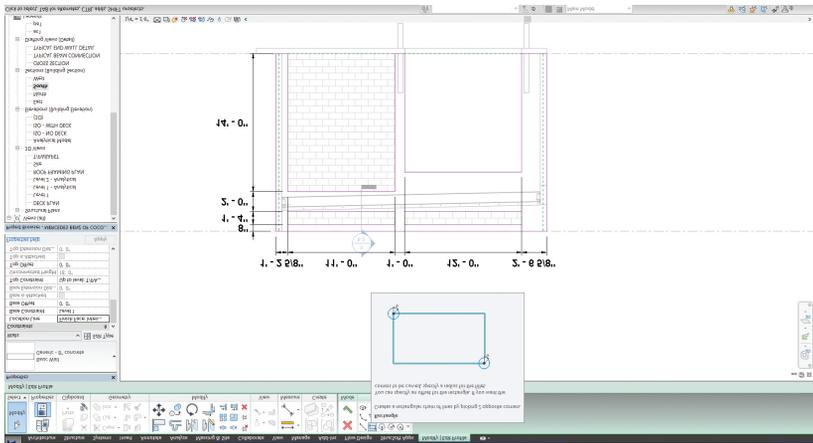


Figure 1: Wall openings

BEAMS

Just as with walls, beams sometimes require cut. In similar fashion Revit has a cutting tool for this as well (Structure>Opening>By Face). Typically, this tool works well for most applications...until you get to a moment connection cut. The problem with using the standard tool for this is you must extend your lines beyond the edge of the beam. If not, and you are slightly inside the beam perimeter (even an amount small enough that you cannot see it visually), it simply will not cut. The cutting shape will remain there, invisible until you actually roll over it with your mouse. By the time you become aware of this, you may have a few attempts at cutting stacked on top of each other #foundoutthehardway. In any event, if conditions are not just right, it can be cumbersome.

A great solution to this (and beneficial in other areas of Revit as well) is the use of Voids to cut elements. In order for this to work, a few stars have to align; not to worry, I will also cover that shortly. As you can see in Figure 2, the end of the beam has been prepared for a moment weld, including the bevels, rat holes, and even down to the backing bars by a Void cutting family.

VOID CUTTING TOOLS

So far, a lot of talk about void cutting tools, but no real information has been provided. Your approach to creating a void cutting tool will depend on the intended application.

If you have one cut, something you may not use again, simply model it in place (Structure>Component>Model In Place). Typically, void cutting tools are void extrusions because they are the most versatile of the shapes, allowing height, width, and depth control. You can create it as a void form, or if you forget and create a solid extrusion, simply select the extrusion and toggle between solid and void in the Properties Browser. If your cutting tool is one that will be used frequently on other projects, create a Void Family.

THE BENEFITS OF FAMILY

The benefits of creating families in Revit are many, including repeatability, stability, and performance. Modeling in place, too often, will increase model size and slow performance. When creating a void cutting family, always create it as a Generic Model - Face Based. When inserted into the project, this type of family can be placed on: Face (Vertical and Horizontal), Level, or Work Plane, allowing every

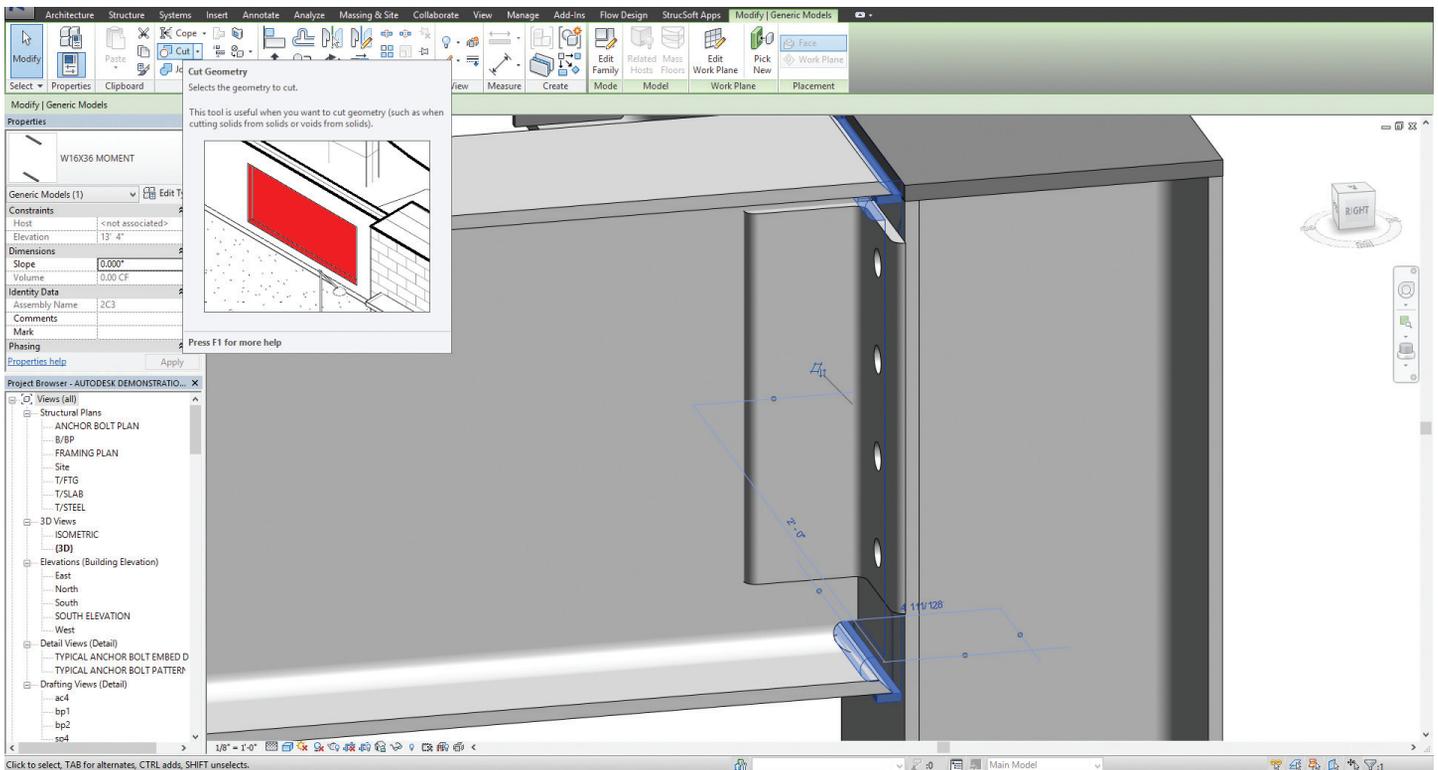


Figure 2: Beam cutting techniques

Revit Structure 2014

scenario to be covered. When creating a void family, be sure to create it to go below the depth of the face, to ensure you properly cut all the way through your element. Because a family is a file, be sensitive to this fact: The deeper the void, the larger the file. Don't use a 1'-0" long void to cut a 1/4" material thickness.

Finally, if you are adept at family creation and using parameters, create your void family with parametric controls. This will allow you to have one family that will cover many different sizes of cuts. We have created a simple tool in this manner that allows me to control height, length, and depth. I use it frequently to cut joist/beam pockets in walls, web holes in beams, precast plan recesses in walls, holes (or even depressions) in floors and roof. Additionally, we also have a parametric hole cutting tool that allows me to enter Hole Diameter, Hole Count, Hole Spacing, Cut Depth. When nailer bolt holes are required in a beam or perimeter angle, whether it is 2 holes or 100, one simple tool cuts all of them with two clicks of the mouse.

A final note about void cutting tools in general: Linear stiffener angle will not allow a void to cut it, as loaded into the project. A workaround for this is to Edit the family and change its Family Category to Structural Framing, and check "Cut with voids when loaded," in the Family Parameters list. Reload it into the project and it can then be cut.

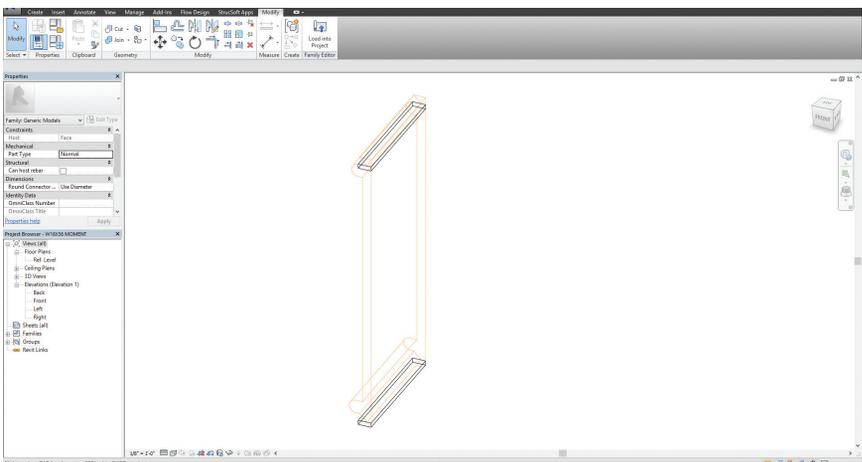


Figure 4: Combination families

FACE of your element. This can be a beam web, if you are using a web cutting tool, or it can be the beam top flange, as is the case with our moment connection cutting tool.

There are few limitations to the complexity and number of included items in a family such as this. A little time spent on a family on the front end will allow you to perform multiple cuts and placement of items with just a couple of mouse clicks. See SCHEDULING for how to make your family schedulable.

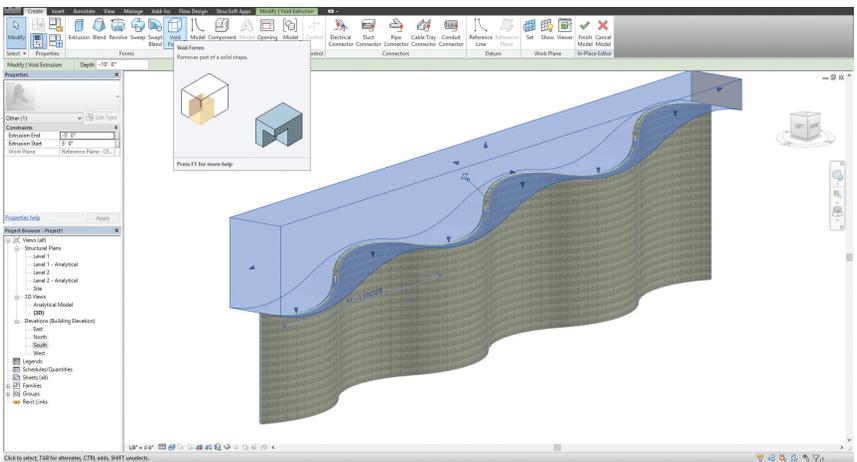


Figure 3: Void cutting tools

FAMILY PARAMETERS

We use Revit to detail structural steel, something that is admittedly not its primary function. Due to that, we are put in the enviable position of having to create our own content. I say it is an enviable position because when it is thrust upon you, there is no better way to learn what TO do...and what NOT to do. Understanding the difference between Instance and Type parameters, and their uses, is liberating in the sense that it gives you complete control over the use of your family. For our cutting tool shown in Figure 3, we use Instance Parameters because we want to use one tool to create several cuts of different sizes and depths.

COMBINATION FAMILIES

As shown in Figure 2, the moment connection prep was cut with a tool that included a void and a couple of solid extrusions. If you have a need to do this, or perhaps you need to cut the web of a beam for mechanical systems and wrap it with flat bar or angle, create the void family, and include the added elements to it. It is a very simple way to ensure that the shape you cut and the added solid elements are always correctly placed. When creating this in a face-based family, remember to use your Reference Level as the

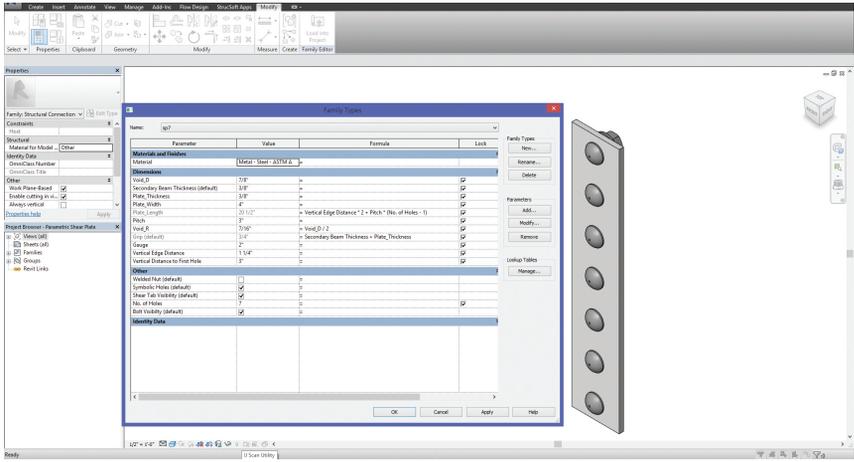


Figure 5: Family parameters

Revit Structure 2014

ing implications, and will eventually show up in areas such as this, often long after the family has been created. It can be changed, but be aware that it can adversely affect the existing elements in your project. A little planning, on the front end, goes a long way.

ASSEMBLIES

A little-used area of Revit is the Assembly option. If you use Revit LT, it does not have this option. The Assembly tool is vital to the work our office produces every day, so it precludes us from using Revit LT. Consider your intended use before buying software; it can become very important when it is too late. As mentioned before, we create most of our content, such as base plates, cap plates, shear plates, connection angles, haunches, stabilizer plates, stiffener plates, etc. Those elements have their own data associated with them, as mentioned in the above paragraph.

Our model is complete; now what do we do with it? Believe it or not, that was a serious question in our office when we first began using Revit. I am a little embarrassed to say that we didn't know about the Assembly tool then, so when the question arose, we simply set about re-drawing the entire structure in 2D Drafting Views. It was a six story building. You can imagine our elation when we discovered the Assembly tool! Now, when we create a model, it will likely contain beams and columns, each with varying numbers of elements attached to it.

There are two ways to create an assembly: 1) Select all the elements that will be included in that assembly, as it will be shop fabricated and shipped. On simple projects this can be done with a simple selection box; however, one day you will come upon a very crowded area; and 2) When you come upon this crowded area, select the main member, create the assembly, then edit the assembly and select the add option. This will allow you to selectively add one item at a time, so you don't end up with unwanted elements in your assembly. When this is achieved, you can then create Assembly Views by right-clicking on the Assembly in the very bottom of the Project Browser, and selecting "Create Assembly Views." Note that several options will be available and you will have to settle on what works for you.

In our office we typically select: 3D Ortho, Front, Back, Left, Right, Top, Bottom, and Part List. Some of the views will not be used, but we select these to give us options on how we want to display the part in the view. The most complex side may be on the back of the beam, or it may require more than one view, as is the case with many columns.

A quick tip to figure out Front versus Left is to look at the 3D Ortho view. On the cube, you will see Front and Right, as well as corresponding South and East directions. This will allow you to select the proper view and is essential for column placement, both on the fab drawings and erection drawings. That being said, perhaps the most comprehensive view created is the Part List. This is the list our office uses as a Bill of Materials. Currently, it requires several steps to manipulate it to be what we want, but we are also working on changing that (if you want further direction on how to

change that, email me and I will be glad to elaborate further). The key to making this list work as a Bill of Materials is the data you previously entered, as discussed in the previous paragraph, SCHEDULING. Just as the assembly combines all of the physical elements of a column/beam, the part list combines all of the data associated with those elements.

One note about assemblies: You cannot include an existing assembly in a new assembly (i.e., if you have created a base plate as an assembly, you cannot include it in a column. To do that, you must first copy the base plate and paste it back to the same location, then immediately create an assembly. You will then have two in the same place (and get a warning)—one as a standalone assembly and one to be included in the major assembly. Be cautious with this, however, as Revit will count all of them, and your schedule count will be off due to that).

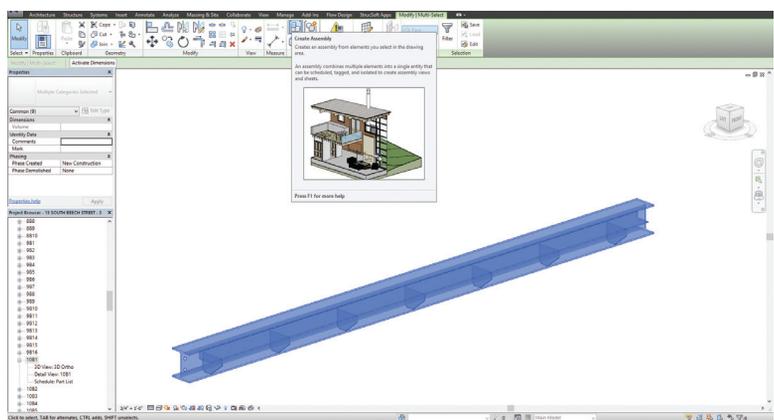


Figure 8: Creating assemblies

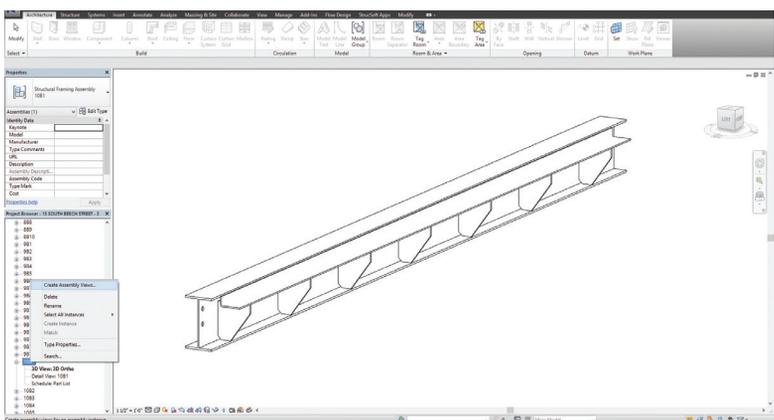


Figure 9: Creating assembly views

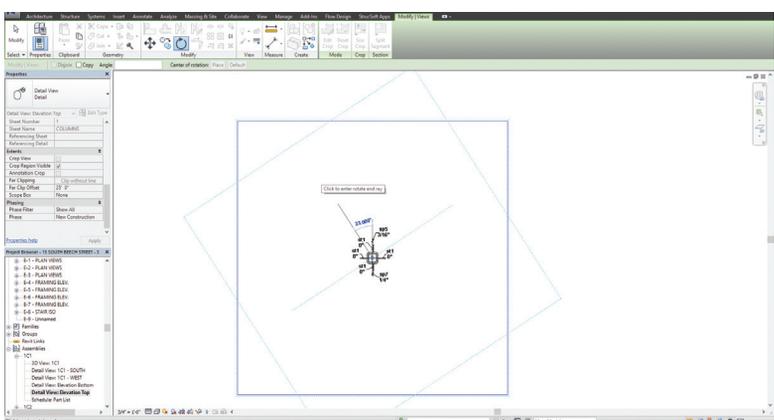


Figure 10: Rotating views

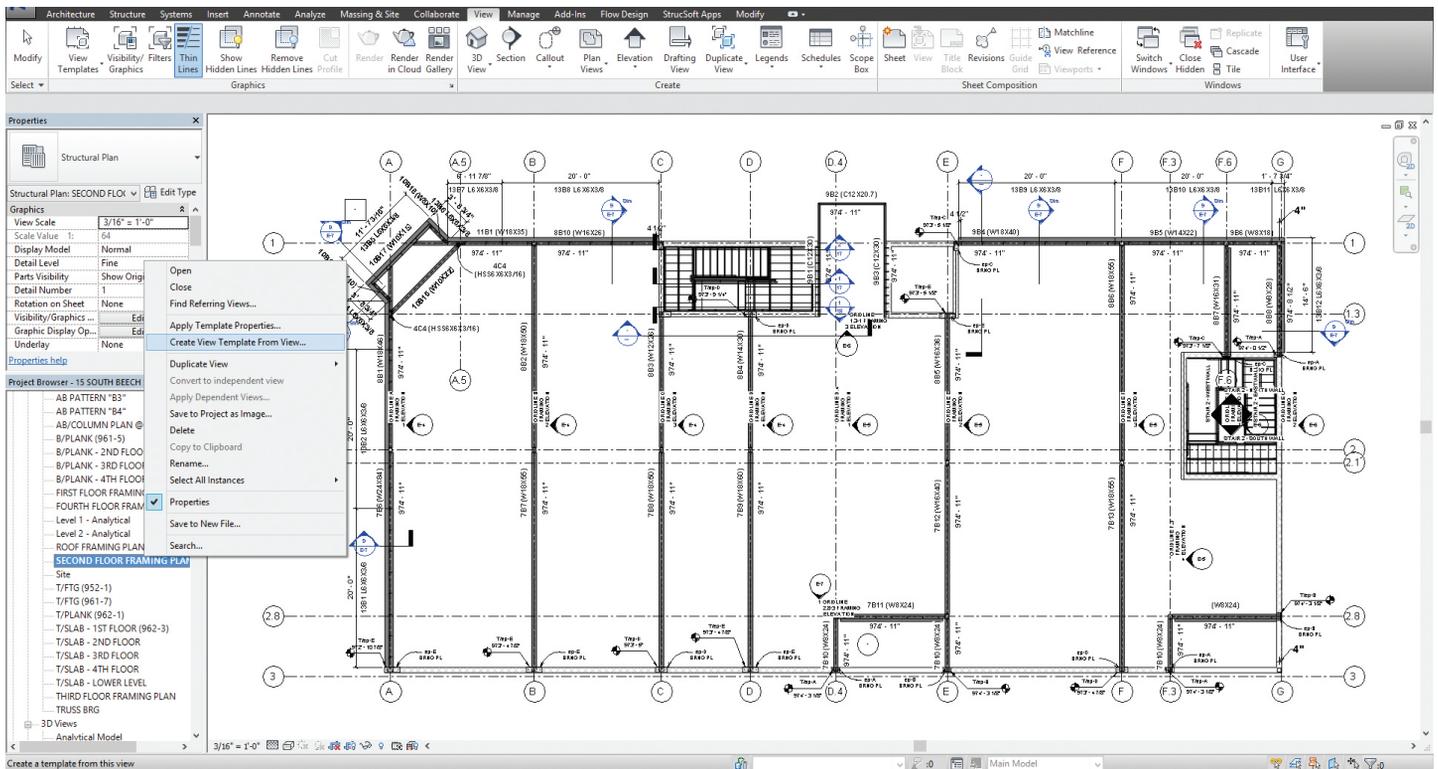


Figure 11: Easy view templates

ROTATING VIEWS

By now you have tried all of the tips & tricks listed above and you have a nice, neat column assembly ready for use. If the most beneficial view of the column was on the right side (East), then you have a curve ball coming. When Revit creates assembly views, the default top and bottom views are oriented to the South view. You may NEED to use an East, West, or North view, as well you should; but the top and bottom views don't know that. A simple workaround for this is: Go to your top view, select "Crop Region Visible," select the crop region in the drawing window, and use the rotate tool to rotate the view (without rotating the model element). Two key actions here are: 1) Select ONLY the crop region (not the model element); and 2) You must rotate the OPPOSITE direction of your intended direction (if you want to see the bottom or top view). I cannot explain the WHY behind the opposite rotation, other than it works. When doing this, always pay close attention to your model element to ensure that the process works correctly. This can be a bit confusing at first, but eventually it becomes routine.

EASY VIEW TEMPLATES

There are a couple of ways to create view templates. One way is to trudge through the laundry list of items provided in the view template window. An easier way is to simply modify the view you are working on to reflect what you want to see. In most cases you will find that this can be achieved through the options at the bottom of the drawings window (scale, detail level, display, visual style, and so on). If you don't want to show a particular category, simply right-click (the element)>Hide In View>Category. When the view is set to your preferences, select the view in the

Properties browser, Right-click>Create View Template From View. Name the view template and you now have the view template you want, without the laundry list.

FINAL WORD

As mentioned before, part of the beauty of Revit for me, as a detailer, is its unparalleled ability to allow practically anything to be created. With many options come many workarounds. The tips & tricks I have listed here are certainly not all inclusive, and may not fit every application. Revit users are a unique breed, in that they take pride in their program and the stunning designs it allows them to create. To that end, many users have also created their own workarounds that better fit their needs. In any event, these are the workarounds that help our office be productive; I hope you find something in here that will help you become the best Reviteer you can be. Keep Calm...and Revit On!



Scott Melching is a former fab/erect company owner with 25 years' experience in the industry. An avid Revit enthusiast and Revit Architecture Certified Professional, he is currently the owner of ST Detailing & Design, Inc. www.stdetailinganddesign.com, American Stair Design www.americanstairdesign.com, and Phoenix Design Studios www.phoenixdesignstudios.com. Living the beach life in sunny South Florida, Scott can always be reached at: Scott@STDetailingandDesign.com

When professional problems demand professional solutions...



the answer is black and white.

www.boxxtech.com



"Intel and the Intel Iris logo are trademarks of Intel Corporation in the U.S. and/or other countries."