

Diamond Sponsors



# AUGI WORLD

The Official Publication of the AUGI Design Community

January 2022

# Trending Technologies

*Also in this issue:*

iLogic Assembly

ArcGIS

3ds Max



DELL PRECISION 3561 MOBILE WORKSTATION

## MORE POWERFUL. MORE SUSTAINABLE.

The Dell Precision 3000 Series mobiles are the world's first workstations made with renewable bioplastics. With up to next-gen NVIDIA® professional graphics and AI-optimization, the 15" Precision 3561 powers design and visualization workflows for the most immersive user experience in an entry-level workstation.



As a special thank you to the AUGI World community,  
you can find exclusive offers from Dell at [Dell.com/AUGI](https://Dell.com/AUGI)

# CONTENTS

## FEATURES

- 6 Civil 3D**  
Grading Optimization
- 12 Inventor**  
Inventor iLogic Notes for Assembly Drawings  
*Republishing an oldie, but a goodie*
- 18 AutoCAD**  
How to Develop an Automation Plan for CAD Systems
- 20 Cad Administration**  
Seven Songs about Scripts  
*Republishing an oldie, but a goodie*
- 26 3ds Max**  
Trending Technologies

## COLUMNS

- 4 Letter from the President**
- 30 Tech Manager**
- 34 Inside Track**



20



26



30

## From the President



### HAPPY NEW YEAR TO ALL OF YOU!

To start the new year off on the right foot, AUGIWORLD brings you “Trending Technologies”.

Our world is a constantly changing place with so much going on that we are often overwhelmed by the amount of information we have available to us. Our goal with this topic is to help the BIM and CAD community focus on some specific technologies that our authors are passionate about.

Regarding CAD & BIM, these are a few of the technology trends that I identified.

- Mobile access so that content can be viewed digitally when needed
- Projects/drawings on the Cloud for better collaboration
- Improved 3D printing for construction manufacturing
- Prefabrication is on the rise
- AI tools like Dynamo and other AI applications for BIM
- Advanced drones and 3D laser scanners
- Energy modeling

I know that some of these have been around for years, but every day we are coming up with better and smarter ways to use them. What technologies do you use? If you are interested in writing an article, let us know!

Tim Berners-Lee (the inventor of the World Wide Web) said this, “*The Web as I envisaged it, we have not seen it yet. The future is still so much bigger than the past.*”

I love his ending statement of the future being bigger than the past! I encourage us all (including myself) to embrace our future. To do this we need to have hope and give others hope, hope in the future and hope in ourselves. We all need to reach out to others around us. You may think your circle of influence is small, but there is so much we can do to improve the lives of those around us. Helping others will improve the immediate future!

Take care,

KaDe

# AUGIWORLD

[www.augiworld.com](http://www.augiworld.com)

### Editors

#### Editor-in-Chief

Todd Rogers - [todd.rogers@augi.com](mailto:todd.rogers@augi.com)

#### Copy Editor

Isabella Andresen - [isabella.andresen@augi.com](mailto:isabella.andresen@augi.com)

#### Layout Editor

Debby Gwaltney - [debby.gwaltney@augi.com](mailto:debby.gwaltney@augi.com)

### Content Managers

3ds Max - Brian Chapman  
AutoCAD - Jim Fisher  
AutoCAD Architecture - Melinda Heavrin  
AutoCAD Civil 3D - Shawn Herring  
AutoCAD MEP - William Campbell  
BIM Construction - Kenny Eastman  
Tech Manager - Mark Kiker  
Inside Track - Brian Andresen  
Inventor  
Revit Architecture - Jay Zallan  
Revit MEP - Nathan Mulder  
Revit Structure - Jason Lush

### Advertising/Reprint Sales

Kevin Merritt - [salesmanager@augi.com](mailto:salesmanager@augi.com)

### AUGI Executive Team

#### President

KaDe King

#### Vice-President

Frank Mayfield

#### Treasurer

Todd Rogers

#### Secretary

Brian Andresen

### AUGI Board of Directors

Brian Andresen	Frank Mayfield
Kimberly Fuhrman	Todd Rogers
KaDe King	Scott Wilcox
Chris Lindner	

### Publication Information

AUGIWORLD magazine is a benefit of specific AUGI membership plans. Direct magazine subscriptions are not available. Please visit [www.augi.com/account/register](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 [www.augi.com/account](http://www.augi.com/account). For all other magazine inquiries please contact [augiworld@augi.com](mailto: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 ©2021 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



# DIGITAL BUILT ENVIRONMENT INSTITUTE



**A Global  
Community for  
AEC Professionals to Learn,  
Grow and Exchange Knowledge.**

Become a member: [www.dbei.org/membership](http://www.dbei.org/membership)

# Grading Optimization

**F**or this article I wanted to take a look at the new Grading Optimization (GO) extension in Civil 3D 2022 software. Some may have heard of it; however, some may not even know it exists! Throughout the course of development, engineers and contractors typically face questions and concerns regarding earthwork quantities. How much cut? How much fill? How do we grade each individual lot? Those answers and many more are crucial to a successful land development project.

With the Grading Optimization extension, you can quickly apply certain design criteria that will help further evaluate the best solution for your project. So, let's look at how you can use GO on your next project.

As this article will be more of a high-level overview/explanation, feel free to email me and I can send a step-by-step outline along with some sample files and videos.

## WHERE DO YOU FIND THE GRADING OPTIMIZATION TOOLS?

Well, good news and bad news. The good news is that if you have the AEC Collection, then you already have Grading Optimization! Bad news? It's only in the AEC Collection!

If you have access to the AEC Collection, the download for GO can be found in your Autodesk Account. Once downloaded, it is then found on the Analysis tab of the Civil 3D Ribbon.

## WHAT IS GRADING OPTIMIZATION FOR CIVIL 3D?

Autodesk Grading Optimization (GO) is an interactive

tool that facilitates grading various land areas such as sites, road interchanges, and around structures. It consists of optimization algorithms that work toward surface smoothness while accommodating user-imposed constraints. These design constraints include grading and drainage elements. The constraints are entered as global parameters and as individual grading element parameters.

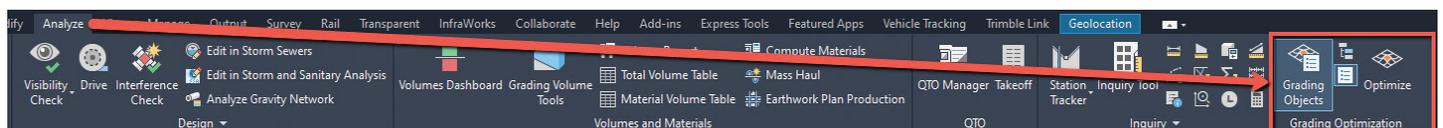
Once optimization begins, the real-time problem solving can be observed. Using several visualization themes for the surface, or by watching the trends of the convergence plots, you can stop and change designs or let it run until a desired grading solution is found. If a solution is not possible, a best approximation is provided within your constraint requirements.

Let's take a look first at the User Interface.

## THE USER INTERFACE

The interface will consist of 4 main parts:

- 1. Grading Object Tool Palette.** This tool palette allows you to tag drawing elements as grading objects for optimization. Each object listed contains default constraint data. You can right-click any object tool to edit the default values applied when used or create additional customized objects.
- 2. Grading Objects Browser.** This tree directory categorically displays each object tagged from the drawing. Selecting a grading object will display the currently specified constraints assigned to it.
- 3. Visualization Display.** This is the viewing area for the optimized grading data. Several display



themes are available to view the surface during and after optimization. The slide to the left of the display exaggerates the contours of the surface to help view subtle changes in elevation.

- 4. Optimization Toolbar.** This bottom band of tools allows you to specify global constraints, mesh and triangle quality, visualization options, and convergence plotting views. Once everything is specified, click Optimize to begin the process.

This workflow describes a typical workflow for using Grading Optimization.

These are hyperlinked to Autodesk help menu for more information and we will explore these steps coming up next.

1. Prepare your drawing.
2. Assign grading objects to your drawing geometry.
3. Open Grading Optimization.
4. Define the system settings for your project.
5. Verify your grading object constraints.
6. Begin optimization
7. Visualize the optimization
8. Return the optimized surface, feature lines, and points to the drawing.

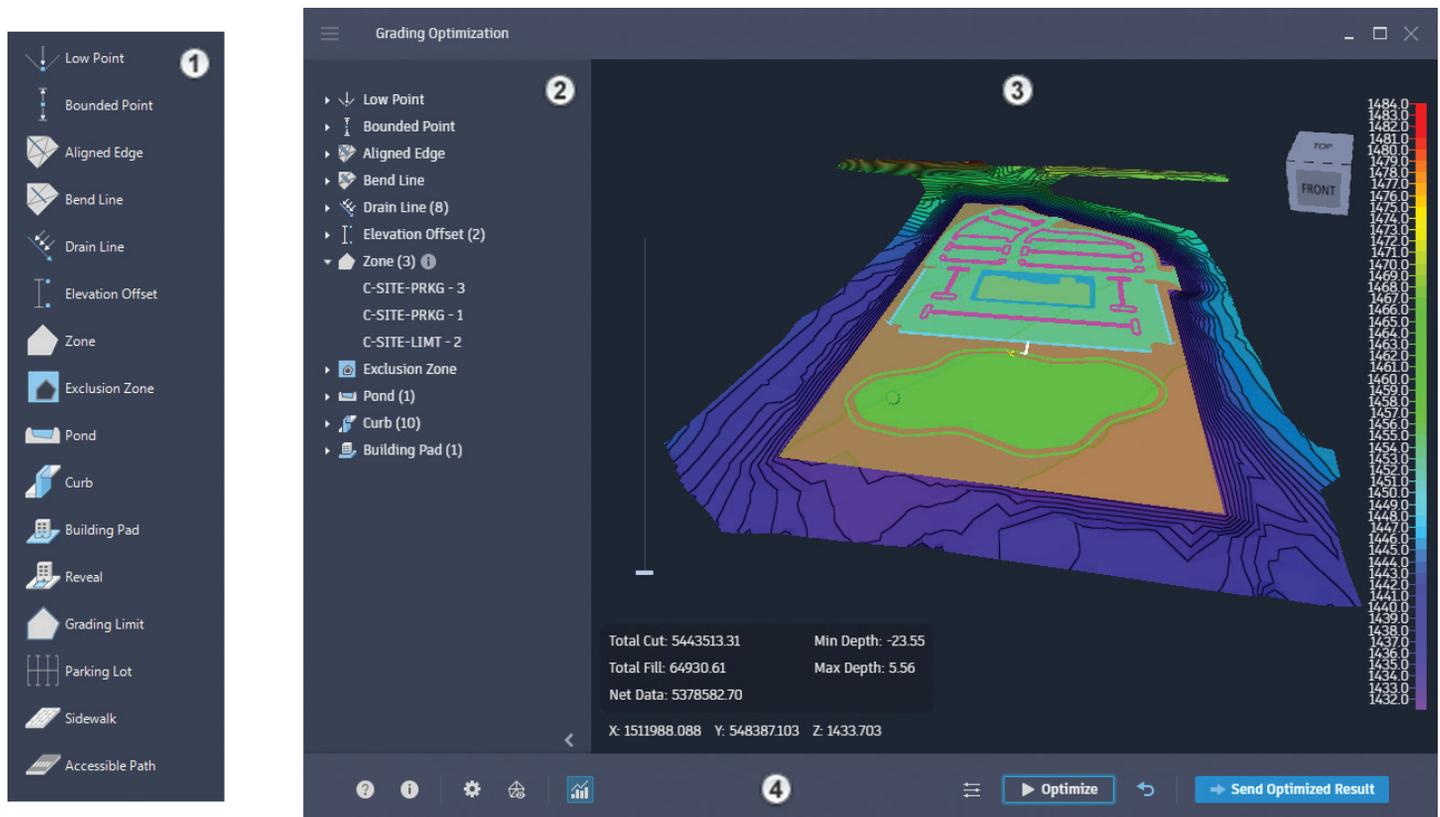
## PREPARE THE CIVIL 3D MODEL

All surface objects are completed with your typical Civil 3D commands for surface creation. Once a site plan or 2D concept plan has been established, Grading Optimization grading objects are then applied to the geometry.

Some of these objects likely already exist in your project at even a conceptual stage, such as building pads, roadway materials (curb, sidewalk, etc.), parking lots, and ponds. The existing geometry can be used to apply grading objects. However, some of the more conceptual objects will need to be planned and sketched such as drain lines, low points, zones, and elevation offsets. The objects influence the Grading Optimization and can simplify problematic areas.

Once applied, the grading object data is added as a property set to your drawing geometry.

At minimum, you typically need 2 items, an Existing Ground surface and 2D AutoCAD linework. The majority of this is just standard AutoCAD commands such as lines, polylines, and the occasional Civil 3D COGO Point or Feature Line. The image below shows which items from AutoCAD can be used as GO items.



New Grading Object	Select in the Drawing	Select in the Tool Palette	Press Enter	Adjust Settings in Dialog
Low Point	COGO point, AutoCAD point	↓ Low Point	Press Enter	Low Point Properties Dialog
Bounded Point	COGO point, AutoCAD point	↓ Bounded Point	Press Enter	Bounded Point Properties Dialog
Aligned Edge	line, polyline, 2D polyline, 3D polyline, or feature line	↻ Aligned Edge	Press Enter	Aligned Edge Properties Dialog
Bend Line	line, polyline, 2D polyline, 3D polyline, or feature line	↻ Bend Line	Press Enter	Bend Line Properties Dialog
Drain Line	line, polyline, 2D polyline, 3D polyline, or feature line. <b>Note:</b> Drain lines cannot be a closed loop to work correctly.	↻ Drain Line	Press Enter	Drain Line Properties Dialog
Elevation Offset	line, polyline, 2D polyline, 3D polyline, or feature line. <b>Note:</b> As sketched, the start of the line will be the higher elevation and the end of the line will be the lower elevation.	↻ Elevation Offset	Press Enter	Elevation Offset Properties Dialog
Zone	Must be closed geometry: polyline, 2D polyline, 3D polyline, feature line, hatch, or parcel.	🏠 Zone	Press Enter	Zone Properties Dialog
Exclusion Zone	Must be closed geometry: polyline, 2D polyline, 3D polyline, feature line, hatch, or parcel.	🚫 Exclusion Zone	Press Enter	Exclusion Zone Properties Dialog
Pond	Must be closed geometry: polyline, 2D polyline, 3D polyline, feature line, hatch, or parcel.	🌊 Pond	Press Enter	Pond Properties Dialog
Curb	Can be open or closed geometry: polyline, 2D polyline, 3D polyline, feature line, hatch, or parcel. Closed geometry creates curbs and open geometry creates islands.	🚶 Curb	Press Enter	Curb Properties Dialog
Building Pad	Must be closed geometry: polyline, 2D polyline, 3D polyline, feature line, hatch, or parcel.	🏠 Building Pad	Press Enter	Building Pad Properties Dialog
Reveal	Reveals are placed on the edges of building pads. Refer to the Reveals topic for the placement procedure.	🔍 Reveal	Press Enter	Reveal Properties Dialog
Grading Limit	Must be closed geometry: polyline, 2D polyline, 3D polyline, feature line, hatch, or parcel.	🏠 Grading Limit	Press Enter	Zone Properties Dialog
Parking Lot	Must be closed geometry: polyline, 2D polyline, 3D polyline, feature line, hatch, or parcel.	🚗 Parking Lot	Press Enter	Zone Properties Dialog
Sidewalk	Must be closed geometry: polyline, 2D polyline, 3D polyline, feature line, hatch, or parcel.	🚶 Sidewalk	Press Enter	Zone Properties Dialog
Accessible Path	Must be closed geometry: polyline, 2D polyline, 3D polyline, feature line, hatch, or parcel.	🚶 Accessible Path	Press Enter	Zone Properties Dialog

**Note:** Geometry cannot intersect with itself to be an eligible Grading Object.

## ASSIGNING GRADING OBJECTS

You place grading objects by assigning them to Civil 3D drawing elements. Once assigned, the grading object information is saved as Property Sets in the drawing.

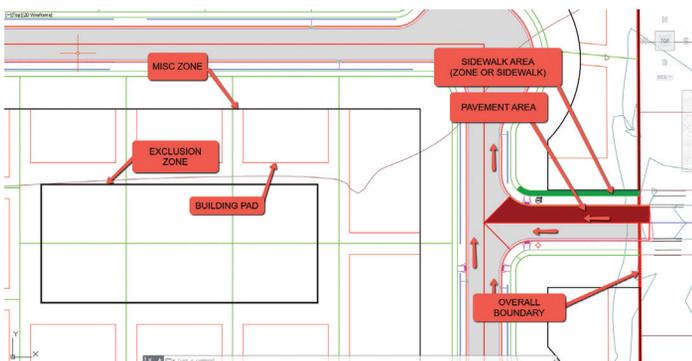
1. Open your drawing.
2. Click Analyze tab > Grading Optimization panel > Grading Object Tools to open the Grading Object Tools Palette.

This tool palette is similar to others in Civil 3D. You can right-click a tool to cut/copy, paste, and rename to create your own preconfigured tools.

3. Continue adding grading objects in your drawing by selecting the appropriate tool and applying it to your geometry.

*Tip:* If you need to go back and change constraints on an object that you created, select it in the **Grading Objects Browser**. You can press and hold Ctrl while selecting multiple objects of the same category. Use the Grading Object Properties dialog to edit their properties.

4. To reference all the grading objects as you refine your drawing, use the **Grading Objects Browser**. Click Analyze tab > Grading

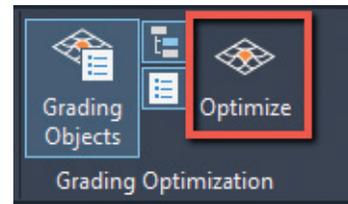


Optimization panel > Grading Objects Browser.

*Tip:* You can click Analyze tab > Grading Optimization panel > Grading Objects to display both the Grading Object Tools Palette and the Grading Objects Browser.

## OPEN GRADING OPTIMIZATION & DEFINE SYSTEM SETTINGS

1. Click Analyze tab > Grading Optimization panel > Optimize.

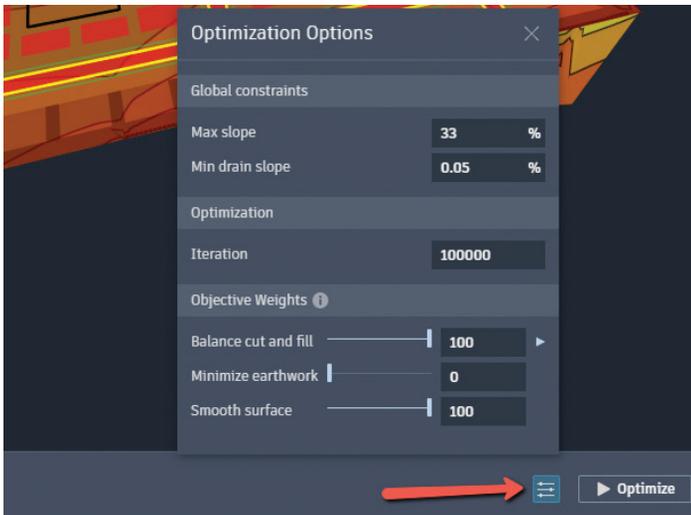


2. You must select a surface; in this case you typically select your Existing Ground.
3. The Grading Optimization interface will then open.
4. Review your project Optimization settings.

There are three types of optimization settings that you can use to adjust to best optimize your grading project: model rendering settings, global constraint and objective defaults, and visualization settings. Review each of these before moving forward on new projects.

- **Modifying Model Settings**
- **Modifying Global Optimization Settings**
- **Modifying Visualization Settings**

I won't go over each of these setting, simply select and review the settings that you are presented.



Before running the optimization, take a quick look at the Optimization Options.

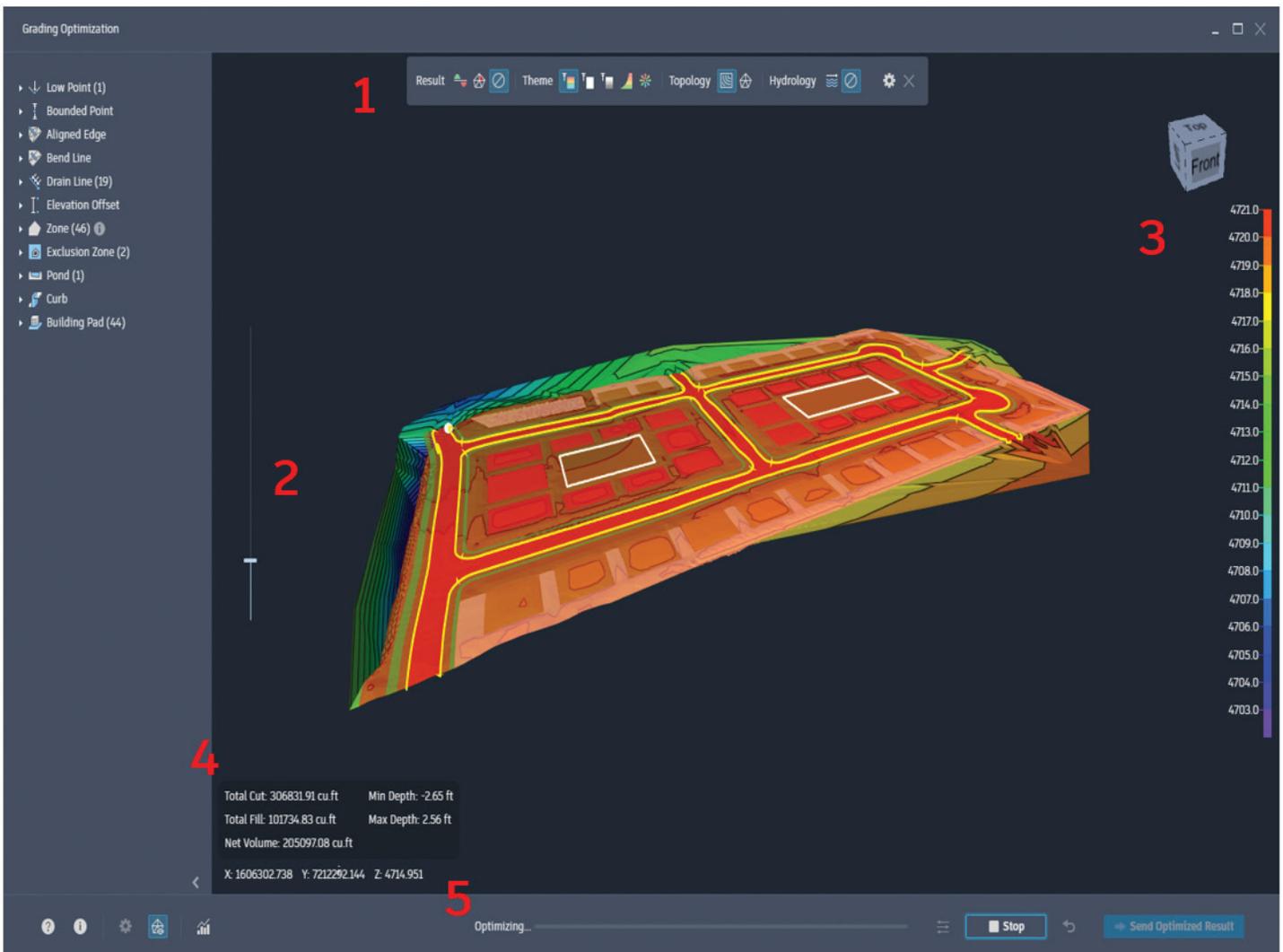
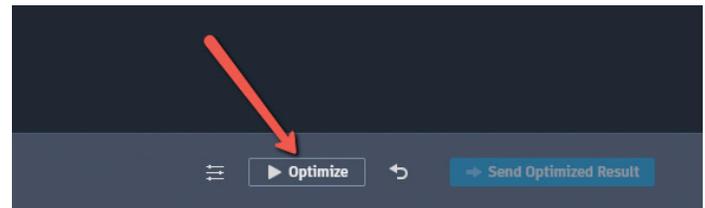
In the Optimization Options you can tell it how many iterations to go through before finalizing the results! You can set this number low (+/-200) for

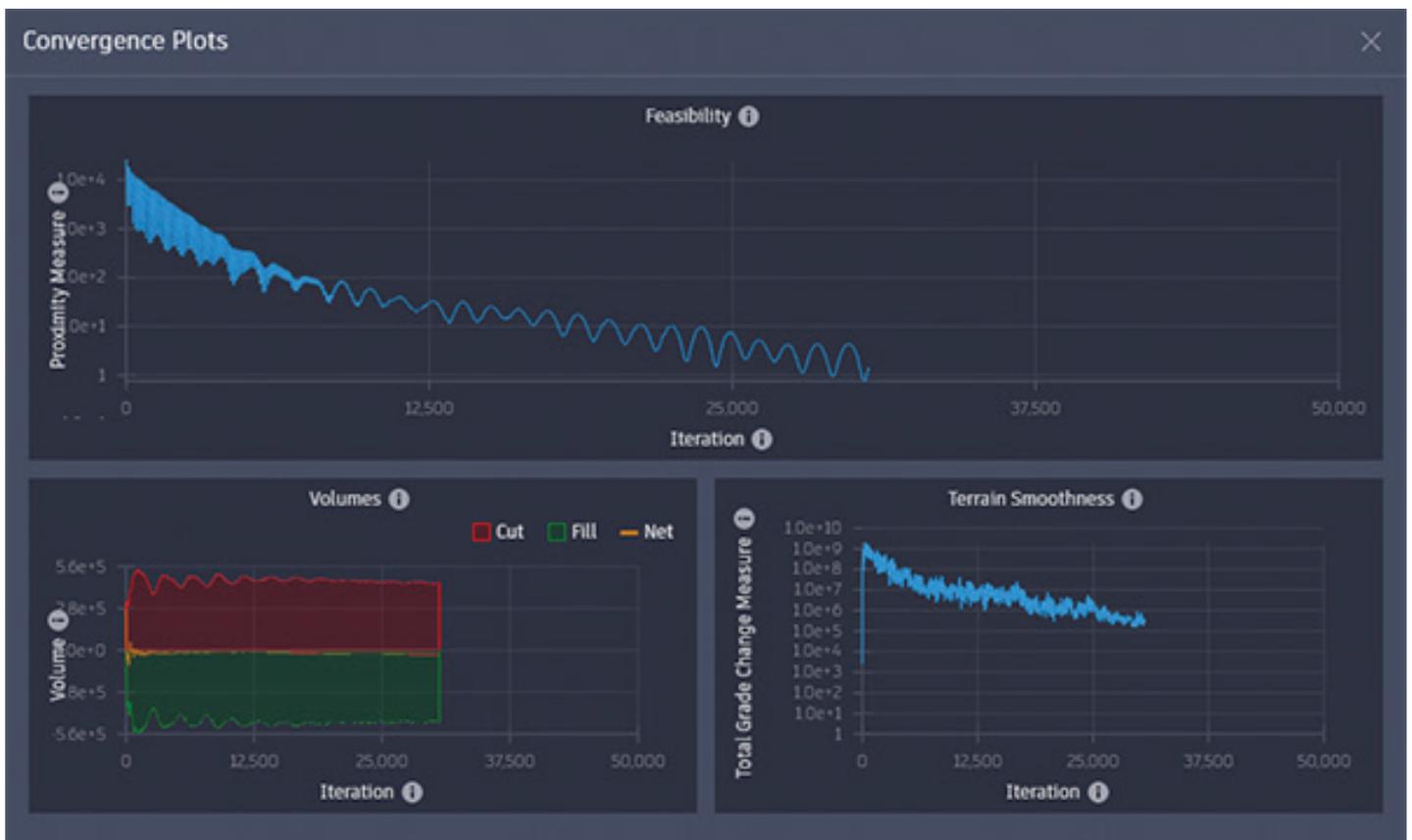
quick results, or really high (+/-100000) and then go to lunch or go home, because it could take a while!

Before moving on to the Optimization of the project, explore the user interface and all the settings. The GO dialog box is a floating window, independent of Civil 3D and can be put on a second monitor if needed.

### BEGIN OPTIMIZATION & VISUALIZE

You start the optimization process by clicking Optimize on the Optimization Toolbar.





### Visualization Display Interface

5. **Visualization Toolbar.** Several display themes are available to view the surface during and after optimization. See Visualization Toolbar.
6. **Vertical Exaggeration Slide.** The slide to the left exaggerates the elevations of the surface to help view subtle changes in elevation. See Contour Slide.
7. **Legend Bar.** The bar to the right displays the color depictions of the different ranges of information in each visualization. These can be modified in the Visualization Settings dialog.
8. **Optimization Status.** This status window displays the current surface data during the optimization process.
9. **Cursor Position.** The XYZ surface coordinates of the position of the cursor.

You can move around, adjust views and exaggeration and view realtime the results as it goes through the grading iterations!

You can also turn on the convergence plot graphs and see even more valuable information! You can hit stop at anytime to accept the results, or let it finish the optimization.

## CONVERGENCE PLOT ANALYSIS

The Convergence Plot dialog gives other perspectives during optimization.

**Feasibility** - Be sure the feasibility line is tracking downward. if it levels off before reaching zero, the desired constraints and objectives cannot be met 100%. Review problem areas in the Violation View, and reassess your plans.

**Volumes** - This chart helps you identify any fill that needs to be brought in or cut that needs to be removed from the site. This chart identifies both and tracks a net volume.

**Terrain Smoothness** - This chart tracks the optimization attempts to minimize the grade change between adjacent triangles on the surface.

Observing the optimization in the Convergence Plot allows to you to see when fewer changes to the surface elevations are occurring. In these cases, it may make sense to stop the optimization instead of waiting for only a slightly better result.

### Progress Bar

The progress bar displays progress in relation to the maximum number of iterations. If Grading

Optimization converges to a solution within the constraints before the maximum iterations, then it stops.

## RETURN THE OPTIMIZATION DATA

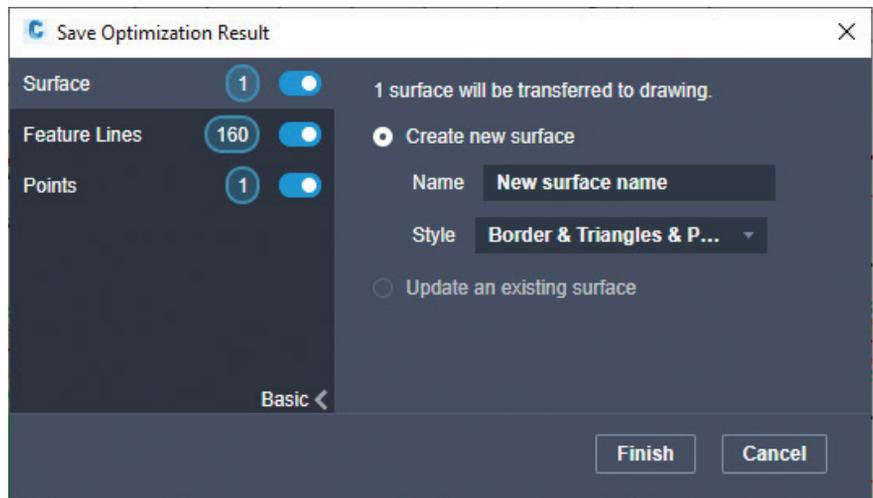
Once you feel comfortable with the results of your analysis, you can return the data back to Civil 3D!

1. Once you click **Send Optimized Result** in the Optimization Toolbar (bottom right), the Save Optimization dialog opens.
2. You then select **Surface** if you want to send the optimized surface back to your drawing and then do one of the following:
  - Select **Create New Surface**, name it, and select its surface style.
  - Select **Update an Existing Surface** and specify the surface to be overwritten.
3. You can also select **Feature Lines** if you want to send any feature lines created in the optimization process back to the drawing. If so, select **Save Feature Lines to Site** and do one of the following:
  - Select **Create a New Site**, name it, and select a feature line style.
  - Select **Update an Existing Site** and specify the site to be overwritten.
4. You then select **Points** if you want to send any points created in the optimization process back to the drawing as a point group. If so, select **Save Optimized Points to Point Group** and do one of the following:
  - Select **Create New Point Group**, name it, and select a point style.
  - Select **Update an Existing Point Group** and specify the point group to be overwritten.

Now take a look at what's been imported!

## CONCLUSION

When I first saw GO in action, I thought it would be a bit overwhelming to set everything up. But once I got into



it on my first couple projects it really wasn't that bad.

As far as use cases go, I see GO being used more for site plans (parking lots w/ buildings, etc.) but have also had good results when applying it to a subdivision. I also see it more of a preliminary grading, to where you get it close to what you want, then fine tune the design utilizing all the feature lines and other geometry that has been imported from GO.

And as usual, the best way to learn GO is to.....



Thank you!

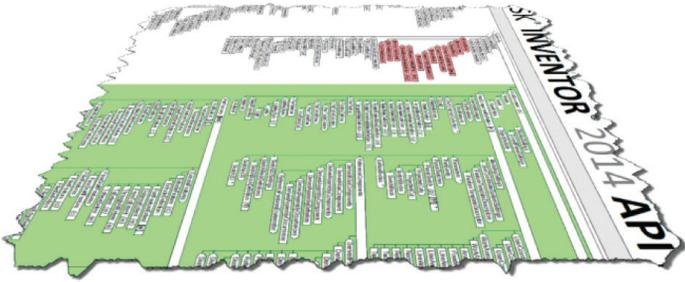


*Shawn has been a part of the design engineering community for roughly 15 years in all aspects of design, construction and software implementations. He has implemented and trained companies across the Country on Civil 3D and other infrastructure tools and their best practice workflows. Shawn can be reached for comments or questions at [sherring@prosoftnet.com](mailto:sherring@prosoftnet.com).*

# Inventor iLogic Notes for Assembly Drawings

It has been some time since I was regularly programming. My background was C++ and I was moving to C# when, well, I started writing; it appears as though there is only enough time to either research engineering software or research software API.

I have been on a project that demanded a lot of drawings, and after I got through revising about 20 drawings, I decided that digging into iLogic would be a sound investment. What follows is a collection of my notes and references relating to the iLogic Assembly document functions that I used to automate drawing creation.



## INTRODUCTION

It is very important to understand the Autodesk Inventor® assembly document structure and hierarchy. Each component in an assembly is a document. Each document is simply a reference to a file that is, or will be upon saving, stored on a file server or local drive.

However, documents can be interpreted in numerous ways. In our case, documents represent parts, assemblies, spreadsheets, drawings, etc. Inventor has class objects that have intelligence built in to deal with how an assembly document, for example, will behave.

We will look briefly at raw document references, and then take a bit of time stepping into the assembly document container hierarchy.

For the scope of this discussion, I will limit that object hierarchy as follows:

**Document -> AssemblyDocument ->  
AssemblyComponentDefinition ->  
ComponentOccurrences -> ComponentOccurrence**

and

**ComponentOccurrencesEnumerator**

## JUST GIVE ME EVERYTHING

First, we can get all file references by calling for 'AllReferencedDocuments'. This function is basically a document dump.

**Base Object Class: Document**

**Type: Function Call**

**Object Function: AllReferencedDocuments**

**Returns: Document object**

This function does a jam-up job of returning all FILE instances within any document object. Only one instance per file is returned, regardless of how

many times a part is used in an assembly.

This function is not reserved to assembly files and will work on any document class-based object. This makes it a great, universal front-end build for code that needs to work in a broad range of Inventor documents.

Caution should be used to provide significant bounds checking as to what type of document is being returned, as AllReferencedDocuments will return anything at all—spreadsheets, part files, you name it.

Example (working with only the part files in a document):

*'Get the active assembly document.'*

```
Dim oAsmDoc As AssemblyDocument
oAsmDoc = ThisApplication.ActiveDocument
```

*'Iterate through all of the documents referenced by the assembly.'*

```
Dim oDoc As Document
For Each oDoc In oAsmDoc.
AllReferencedDocuments
```

*'Verify that the document is a part.'*

```
If oDoc.DocumentType = kPartDocumentObject
Then
Dim oPartDoc As PartDocument
oPartDoc = oDoc
```

```
{-----
```

*'Do something with the part document here*

```
{-----
```

```
End If
```

```
Next
```

```
...
```

This is great when you want everything, but I need to see some structure, and having some refinement would be nice.

## ASSEMBLY COMPONENTS

This is what cleaned up my code tremendously. I am dividing into parts so that it is (hopefully) easier to digest.

Assembly component occurrences are all components in an assembly, which contain occurrences, or each component. This section will discuss the use of these, and the functionality associated with extracting each portion.

## Assembly Component Definition

The complete component container is similar to the CAD bills of material (BOM), except that the actual BOM definition data is contained separately inside this structure.

**Base Object Class:** *AssemblyDocument*

**Type:** *Standard Object*

**Object Definition:** *AssemblyComponentDefinition*

**Returned By:** *AssemblyDocument.ComponentDefinition Function*

Access to the assembly component structure is performed through this object. We need to define the 'AssemblyComponentDefinition' object and load it with data from a call to the assembly document's 'ComponentDefinition' function.

Example:

*'Get the active assembly.'*

```
Dim oAsmDoc As AssemblyDocument  
oAsmDoc = ThisApplication.ActiveDocument
```

*'Get the assembly component definition.'*

```
Dim oAsmCompDef As  
AssemblyComponentDefinition  
oAsmCompDef = oAsmDoc.ComponentDefinition
```

...



Now that the Assembly Component Definition is filled, it is time to do some digging.

## Component Occurrences

As you might expect, these are the containers that each component document is present in.

If a part definition occurs 10 times in an assembly, there are 10 component occurrences, and all 10 will show up in this container. The benefit here is that skeleton and reference files are not part of the 'Assembly Component Definition' structure, so they don't get in the way here.

**Base Object Class:**  
*AssemblyComponentDefinition*

**Type:** *Collection Object*

**Object Definition:** *ComponentOccurrence*

**Referenced at:** *AssemblyComponentDefinition.Occurrences*

The occurrences collection is the whole enchilada of component occurrences, which you can access directly.

Now we can define a 'ComponentOccurrence' object, and use it to inspect each 'ComponentOccurrence' that exists in our 'AssemblyComponentDefinition'.

In this example, I used a "For Each" function to pull each component.

Example:

*'Iterate through all of the Part Occurrences'*

```
Dim oOccurrence As ComponentOccurrence  
For Each oOccurrence In oAsmCompDef.  
Occurrences
```

*'Set Reference to Occurrence Name'*

```
Dim oOccName As String  
oOccName = oOccurrence.Name
```

*'Show each name in a dialog'*

```
MessageBox.Show(oOccName, "Document Name")  
Next
```

...

This is a great way to quickly run through the assembly and get every component. What follows is more functionality to be used when you want to be specific about what you are getting.

## Occurrence Enumeration

This object goes hand in hand with the next function. While its complete functionality is still not quite understood by me, this object acts as a container to catch a collection of component occurrences, and disseminate them by type into an

intelligent object with appropriate functions.

### **Base Object Class: ComponentOccurrences**

#### **Type: Collection Object**

#### **Object Definition: ComponentOccurrencesEnumerator**

#### **Referenced At: itself as dimensioned object**

In the last example, we bypassed the need for this and iterated through our assembly component definition's collection of components directly, like a book, one page at a time. However when we want the assembly component definition to hand us a big list of components, we need somewhere to put them. The Inventor API has provided this container for that purpose.

We need to define the 'ComponentOccurrencesEnumerator' object for the next section.

Example:

*'Define the Component Occurrence Enumerator*

```
Dim oLeafOcCs As
ComponentOccurrencesEnumerator
```

### **All Leaf Occurrences**

Leaf occurrences are part files in an assembly, the end of any branching in the structure.

#### **Base Object Class: AssemblyComponentDefinition**

#### **Type: Function**

#### **Function Call: Occurrences.AllLeafOccurrences**

#### **Returns: ComponentOccurrence**

This function returns the very same component occurrences we iterated through previously; however, this function only returns a collection of those components that represent the end of assembly branches, part or 'leaf' objects.

While we were able to peruse the assembly component definition's components like reading a book, in this function the assembly component definition object will dump an appendix in our lap.

Here, we will fill our component occurrence enumerator by a call to the component definition's 'AllLeafOccurrences' function. Then define

another component occurrence to represent and investigate each leaf occurrences that is in our filled enumerator.

Example :

*'Create the Enumerator to catch all the leaf occurrences of the assembly.*

```
Dim oLeafOcCs As
ComponentOccurrencesEnumerator
oLeafOcCs = oAsmCompDef.Occurrences.
AllLeafOccurrences
```

*'Iterate through the occurrences and print the name.*

```
Dim oOcc As ComponentOccurrence
For Each oOcc In oLeafOcCs
MessageBox.Show(oOcc.Name, "Occurance Name")
```

### **All Referenced Occurrences**

The Inventor API will also permit users to extract all occurrences of specific Inventor documents.

#### **Base Object Class: AssemblyComponentDefinition**

#### **Type: Function**

#### **Function Call: AssemblyComponentDefinition.Occurrences.**

#### **AllReferencedOccurrences(Document)**

#### **Returns: ComponentOccurrences**

This function will return all instances of the specified document at any level within a component definition.

This example takes a file name, and returns all occurrences of it that exist. It uses a call to opened documents expecting that if it exists, Inventor has it opened. Some good bounds checking could be applied to catch any error associated with an unopened file.

Example:

*'Get the active assembly.*

```
Dim oAsmDoc As AssemblyDocument
oAsmDoc = ThisApplication.ActiveDocument
```

*'Get the definition of the assembly.*

```
Dim oAsmCompDef As
AssemblyComponentDefinition
oAsmCompDef = oAsmDoc.ComponentDefinition
```

' *Get the document to find occurrences for.*  
Dim sDocName as String

sDocName = "C:\designandmotion.ipt"

Dim oDoc As Document  
oDoc = ThisApplication.Documents.  
ItemByName(sDocName)

' *Get the occurrences that represent this document.*  
Dim oOccs As ComponentOccurrencesEnumerator  
oOccs = oAsmCompDef.Occurrences.  
AllReferencedOccurrences(oDoc)

' *Iterate through the Occurrences*  
Dim oOcc As ComponentOccurrence  
For Each oOcc In oOccs

' -----  
' *Do Something here with the Occurrence*  
' -----  
Next  
...

## CLOSING THOUGHTS

I used this code to automate the production of drawings, running out as many as 20 at a time right out of an assembly drawing file. I added substantial enhancements so that each drawing contained views, parts lists, as well as automating various key iProperties values.



I hope this helps you get a jump start on iLogic assembly code as it did for me. If you'd like more information on Inventor, iLogic, or even how we

built the drawing generator, stop by Design & Motion. <http://designandmotion.net>

## References:

*Special thanks to the Mod The Machine team and their article "Accessing Assembly Components" from 2009:*

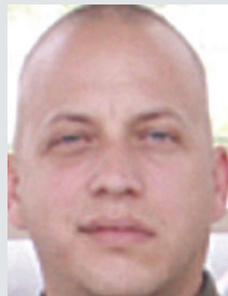
[http://modthemachine.typepad.com/my\\_weblog/2009/03/accessing-assembly-components.html](http://modthemachine.typepad.com/my_weblog/2009/03/accessing-assembly-components.html)

*The Autodesk Inventor 2014 API chart:*

<http://images.autodesk.com/adsk/files/Inventor2014Model.pdf>

*Mod the Machine Article "Understanding File References" from 2008:*

[http://modthemachine.typepad.com/my\\_weblog/2008/11/understanding-file-references.html](http://modthemachine.typepad.com/my_weblog/2008/11/understanding-file-references.html)



*John Evans is an Autodesk Certified Inventor Professional living in the Florida Panhandle, where he provides technical troubleshooting at Gustin, Cothorn, and Tucker, Inc. His career through the Aerospace Design, manufacturing, and maintenance spans 24 years and includes a tour in the USAF. John now works as a design consultant and author from his company John Evans Design and manages the blog "Design and Motion", where he combines his passions: Autodesk Inventor, simulation, and motion control. He is a regular attendee of Autodesk University and has recently joined Tekni Consulting as a contributing author for the Creative Inventor training series. He can be reached at [john@johnnevansdesign.net](mailto:john@johnnevansdesign.net)*

# 16C Intel® i9-12900KF in Stock!



## Xi® M Tower™ 2P64 & 2P64X

Desktop Supercomputers with Dual CPU scalability for AMD Epyc® and Intel® Xeon® platforms, up to 128 Cores & 2TB Reg. ECC. Available with multiple NVIDIA® Quadro® RTX® A6000 w/48GB & GeForce® RTX 3090 w/24GB V-RAM GPU.



## Xi PowerGo™

Our PowerGo CAD Laptop line starts with the 8-core Intel i7-11800H and RTX 3050 from \$1378. The XL model with Quadro T1200, A2000, A3000 & A5000 is priced starting at just \$1727.00. The XT goes up to i9-11900K, RTX 3080-16GB & 128GB DDR4.



## Xi NetRAIDER™ & BladeRAIDER™

The most cost effective building blocks for Data Centers, Render Farms, AI Simulations and Machine Learning computing. 1U/2U/4U & 6U racks with great scalability to 128 cores AMD Epyc, 2TB ECC to 8x NVIDIA Ampere® A100, Ampere A6000 & GeForce 3090.



## Xi M Tower PCIe/64/64TR

Quiet, powerful, compact, award winning, cost effective, state of the art workstation available with the latest 16-Core Intel i9-12900KF professionally over clock all P-Cores @ 4.9GHz or the 16-Core AMD Ryzen 5950X @ 4.9GHz & the 64-Core Threadripper Pro to 1TB Reg. ECC 3200MHz. @Xi certified for 100+ CAD applications including AutoCAD, 3ds Max, Adobe CC, CATIA, Cinema 4D, Inventor, Creo, Revit, & SOLIDWORKS.

Impeccably built, guaranteed & supported in the USA. Our expert IT Architects will customize the fastest workflow solution for your application and budget.



www.xicomputer.com

800-432-0486

# How to Develop an Automation Plan for CAD Systems

I have been automating CAD systems for 20 years, and I'm here to tell you: you're probably doing your CAD automation process wrong.

The process I've seen most often goes something like this: A CAD technician goes to the CAD manager and complains about a task that takes a long time to complete. A decision is made to automate this task to increase efficiency. Sometimes the CAD manager can write a script or create a plugin that aids in the completion of the task. Sometimes the CAD manager will have to hire a developer to produce the plugin. This leads to a system that has a handful of helpful but compartmentalized plugins—all of which need to be maintained and may or may not improve your CAD system. In businesses large and small, I see this scenario time and time again.

Two years ago, I made the jump from CAD manager to full-stack developer. Picking up the knowledge of business processes showed me that many places are making the same mistake and have been looking at AutoCAD automation through the lenses of a CAD technician for way too long. Approaching it from a business process point of view has taught me the real value is in automating the whole process-- not just automating away problem areas.

## **PREREQUISITES:**

Before you decide on automating your CAD system, I suggest using this small list of tasks to help better align your end result. The better job you do with these tasks, the better your automation plan will be.

- 1. Define CAD standards:** If your company has not developed CAD standards, they need to be created and put into action before any effective automation can take place. The most critical areas to standardize are file storage directory structure, file naming convention, symbol library, templates, and layers.
- 2. Define your complete process:** Typically, I use Visio to draw up flow diagrams to aid in this process. The process should start with file creation and document each step it takes to complete the drawing and output to the end result file. (Note: only include the high-level drafting tasks.)

## THE ANNOTATION PLAN:

Now that you have CAD standards and a documented working process, you can begin looking for areas you can automate. I separate my plan into three parts: **file system, drafting/design acceleration**, and finally, **file output**.

### PART I: FILE SYSTEM

The file system is the foundation of your CAD system but is so often overlooked. This is an area you simply cannot get away from. No matter the size of the project, your files must be created, stored, and organized for retrieval. I have seen so many easily avoidable issues drag a team down had they just used a solid file system. Files have been lost, old design files have been delivered to clients, and people have lost untold amounts of time looking for the file they know exists but just cannot find. A good file system should abstract away as much of the complexity as possible. It should handle file creation, naming, renaming, merging, and deletion. Ideally, the application controlling the file system should be available inside and outside of your CAD applications.

### PART II: DRAFTING ACCELERATION

This is typically the area where most CAD system automation is focused. The biggest issue I've found is everyone wants to automate only the hard parts. While automating the difficult drafting or design tasks is important, this isn't the first thing you should automate. Remember: if it's difficult to manually do this task, it is going to be just as

difficult (and potentially expensive) to automate it. Remember those process flows you created as a prerequisite? Here's where those components come into play. Layout your plans vertically so you can see all of them at once. Now line up each step that is the same for each drafting process. Now go through each step in your process and make a guess at how long you spend on each task. Start with tasks required on 100% of your files. Can any of these tasks be automated completely? Automating these easier tasks should be simpler and much more affordable than automating the more complex tasks. This approach also builds confidence in your automation plan, and should help when it is time to get funding for your more complex automations.

## PART III: OUTPUT

Eventually your CAD files will need to be output into a format that anyone can read. This could be plotted on paper or published to PDF for digital transmittal. A common issue I run into is ensuring the latest version of the CAD file is published and stored in the proper location. This location should be setup and managed as part of the file system, but automating the publishing task must be handled by the CAD system. The tricky part is determining when an AutoCAD file should be published. This process will vary based on your environment but automating this process is necessary.

The best automation plans have components that build on each other. It abstracts mundane tasks from the user so they can work on more important tasks. Most importantly, this type of plan starts every project off on a solid foundation.



*Jason was a CAD Technician, CAD Manager, and IT Specialist for 17 years before making the switch to being a full-time full stack developer. His current role is lead developer for a consulting team creating web applications, Autodesk Plugins for AutoCAD, Revit, Civil 3D, Navis Works, and desktop applications.*

# Seven Songs about Scripts and How They Improve a CAD Manager's World



**T**he first CD I ever purchased with my allowance money was the debut studio album by the Spice Girls. I remember playing it over and over on my boom box as I rode my bike around in a circle on the driveway. I soon acquired more CDs, and it eventually became a big issue for me as an eight year old to hop off my bike to manually change the CD when it was done playing.

Then one day, I discovered my dad had acquired a five-disc CD changer and my world was transformed. Everything was easier and more automated. I didn't have to take extra time to manually change the CD. I had a process that did the work for me, and I was able to go on with my day.

What does all this mean? It means that a very simple script is a lot like the five-disc CD changer in the story (except way more powerful). This article will show how simple scripts can help CAD managers reach their automation goals in a carefully selected playlist of Seven Songs about Scripts.

## 1. "WHAT'S A SCRIPT? (BABY DON'T HURT ME)"

The first song on the list asks the most important question. What exactly is a script? A script is a powerful tool that can be used to automate tasks for

## SCRIPT #1 - DEPLOYMENT ASSEMBLY SCRIPT

(copy & paste in a separate text file in Notepad)

*Note: Don't forget to change file paths prior to testing!*

```
REM Copies all your custom content to
theDeployment Pack
robocopy „C:\Source“ „C:\Destination“ /e /mir
```

```
REM Copies the Version of Deployment Pack
robocopy „C:\Source“ „C:\Destination“ „*.txt“ /e /mir
```

```
REM Copies the Tool Palette Group File
robocopy „C:\Source“ „C:\Destination“ „Profile.aws“
```

```
REM Copies the Tool Palette Files
robocopy „C:\Source“ „C:\Destination“
„somefile*.atc“
```

```
REM Copies the Tool Palette Image Files
robocopy „C:\Source“ „C:\Destination“ /e /mir
```

the aspiring CAD manager. The tasks can include copying and pasting files, overwriting files, backing up files, naming files, and more. Regardless of which Autodesk® product you are using, there can potentially be a lot of customization files to manage. For AutoCAD® MEP, having a custom CUIx file, custom Content Builder content, user profile, various automation files, tool palettes, etc. can yield more than 1,500 files to deploy. (I would know... I counted!)

Manually deploying all these files can lead to mistakes, not enough files being accounted for, and countless hours troubleshooting why your sweet custom setup just does not work on the end user's computer as well. You can write a script to automate the task of copying in files where they are supposed to go, so you don't have to do it manually. No fancy programs are required. You can simply use Microsoft® Notepad® (or Notepad++ if you have it) to create a simple script.

It is important to note that scripts are powerful and that is not something to be taken lightly. Incorrectly using scripts can potentially overwrite files you didn't intend to overwrite, and that will hurt. In this article, I will describe the approach I take to writing scripts. However, I highly recommend certain safety measures if you are new to the concept. In other words, don't try this at home unless you incorporate the following steps.

- Save scripts created in Notepad as text files (.txt) until you are ready to test. (Save as .cmd or .bat later.)
- Right click>Edit to edit any script that does not have .txt as a file extension. Simply clicking on a script file can run it!
- Set up local test folders so you can see what is happening when you run your scripts before you configure them to be the file paths you actually need them to be. This is the most important tip because syntax matters!

## 2. “DEPLOY ME MAYBE”

The next song on the list addresses how to use scripts to create a local deployment assembly pack, which is basically an assembly of all the files that will eventually be distributed. To do this, simply open up Notepad to make use of the following concepts shown in Figure 1.

```
ROBOCOPY “Source file path” “Destination File Path”
ROBOCOPY “Source file path” “Destination File Path” “Specific File”
ROBOCOPY “Source file path” “Destination File Path” “Any File with Wildcard”
```

Figure 1: Concept

Figure 1 shows three different ways to use Robocopy (or robust copy) to copy files from the source file path to the destination file path.

- Copy all contents of the source file path and put them in the destination file path.
- Copy one specific file from the source file path to the destination file path.
- Copy any files that have part of a file name from the source file path to the destination file path (this comes in handy if you don't want to spell out several specific files).

Creating a deployment assembly pack may sound redundant, but being in a CAD management role means I'm testing things all the time. I don't want to accidentally include test files or future development files while deploying content to others, so I make all my files copy into a nicely formatted deployment pack with subdirectories that house many different custom files. This deployment assembly pack acts as a final checkpoint before everything is sent to the server. This is important because once your files hit the server, they are considered “live” and

## SCRIPT #2A – HOST TO SERVER SCRIPT

(copy & paste in a separate text file in Notepad)

*Note: Don't forget to change file paths prior to testing!*

REM Creates Backup File of Existing Deployment Pack on the Server  
 robocopy "E:\Server Deployment Location" "E:\Server Deployment\_Backup\Backup\_%date:~/=%" /e

REM Copies New Deployment Pack on the Server  
 robocopy "C:\Deployment Pack Assembly" "E:\Server Deployment Location" /e /mir

## SCRIPT #2B – HOST TO SERVER SCRIPT-POWERSHELL

(copy & paste in a separate text file in Notepad)

*Note: Don't forget to change file paths prior to testing!*

REM Creates Backup File of Existing Deployment Pack on the Server Using PowerShell

```
powershell -command „& {$source = ,E:Server Deployment Location';$destination = ,E:\Server Deployment_Backup\Backup.zip';&'add-type' -assembly ,system.io.compression.filesystem';[io.compression.zipfile]::CreateFromDirectory($source,$destination)}“
```

```
rename „E:\Server Deployment_Backup\Backup.zip“ „Backup_%date:~-10,2%%date:~-7,2%%date:~-4,4%.zip“
```

REM Copies New Deployment Pack on the Server  
 robocopy „C:\Deployment Pack Assembly“ „E:\Server Deployment Location“ /e /mir

ready for end users to retrieve. See Figure 2 for a simplified script on assembling a deployment pack for AutoCAD MEP.

Notice there are a few additions to the main concepts shown earlier.

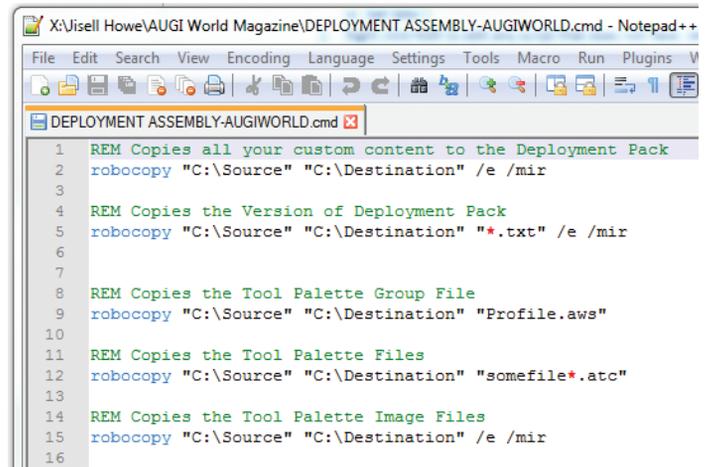


Figure 2: Deployment assembly script

- **REM** to allow for embedded comments/notes that help with troubleshooting and ensuring everything is accounted for
- Options or switches like /e and /mir to tell the scripts to copy in all subdirectories and make the destination files look exactly how the source files look (mirror/overwrite if necessary)
- “Version” of the deployment pack, which is just a simple text file with the date and details of the update noted within it (Figure 3)

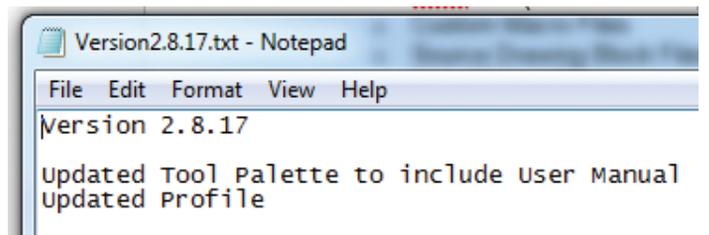


Figure 3: Deployment version

Following this script example will create a nicely formatted deployment assembly pack that pulls files from wherever you have them on your local machine. This deployment assembly pack will allow you to verify its contents in a well-organized way before it is sent to the server. Treat this script as something that is nice and beneficial to have, but not required. I mainly did this because I had reasons to keep many of the out-of-box AutoCAD file paths for my custom files to also reside.

## 3. “SEND THE FILES UP BEFORE YOU GO-GO”

Making hundreds of files available on the server makes it easier for rolling out deployment packs (read: your company standards), because any other

method of distributing those same files could be very cumbersome. It is important to house these files on the server (for distributing purposes) because something could always happen to individual computers and local files where your customizations should ultimately reside.

As soon as you verify the contents of your deployment assembly pack, get them up to the server via a host to server script. Create a host to server script by simply writing the first Robocopy line from Figure 2 and modifying the paths so the source is your deployment assembly pack and the destination is your server deployment location path.

#### 4. “BACKUP FILES THAT I USED TO KNOW”

It is important to get this next song stuck in your head because when you send files to the server, you also need to account for file backups of the “old” content that is already there. Your host to server script should make backups of old content before the new content from the deployment pack assembly gets sent to the server.

Figure 4 shows an example of how this is done. Notice that the first Robocopy line generates a backup file with the current date in a different location. There are other ways to format file names, but this is just one way to create a dated backup file with old content in it that you can easily reference if needed in the future.

Also notice that I don’t have all content mirroring (overwriting) from the server deployment location to the server deployment backup location. I wrote the first line that way because I don’t want the files to be overwritten. I want backup files to keep generating with all the respective subdirectories, so I do not have a /mir switch after that line. Now they’re just some files I used to know.

```

1 REM Creates Backup File of Existing Deployment Pack on the Server
2 robocopy "E:\Server Deployment Location" "E:\Server Deployment_Backup\Backup_&date; /m%& " /e
3
4 REM Copies New Deployment Pack on the Server
5 robocopy "C:\Deployment Pack Assembly" "E:\Server Deployment Location" /e /mir

```

Figure 4: Host to server script

#### 5. “POWERSHELL™ SCRIPT OF THE HEART”

If you want to be extra crafty with your backup files, you may want to turn this song up. PowerShell™,

#### SCRIPT #3 – SERVER TO USER SCRIPT

(copy & paste in a separate text file in Notepad)

*Note: Don’t forget to change file paths prior to testing!*

```

REM Copies Deployment pack from Server to
End User Computer
robocopy „E:\Server Deployment Location“
„C:\End User Computer\Your Custom Setup“
/E /MIR

```

```

REM Copies Tool Palette Group file to End User
Computer
robocopy „E:\Server Deployment Location“
„C:\Users\%username%\AppData\Roaming\
Autodesk\MEP 2016\enu\Support\Profiles\
Profile“ „Profile.aws“

```

```

REM Copies Tool Palette Files
robocopy „E:\Server Deployment Location\
Custom Files\Tool Palettes\AUGIWORLD“ „C:\
End User Computer\Users\%username%\
AppData\Roaming\Autodesk\MEP 2016\enu\
Support\WorkspaceCatalog AME (USImperial)\
Palettes“ „AUGIWORLD*.atc“

```

```

REM Copies Tool Palette Image Files
robocopy „E:\Server Deployment Location\
Custom Files\Tool Palettes\AUGIWORLD\
images“ „C:\End User Computer\
Users\%username%\AppData\Roaming\
Autodesk\MEP 2016\enu\Support\
WorkspaceCatalog AME (USImperial)\
Palettes\images“

```

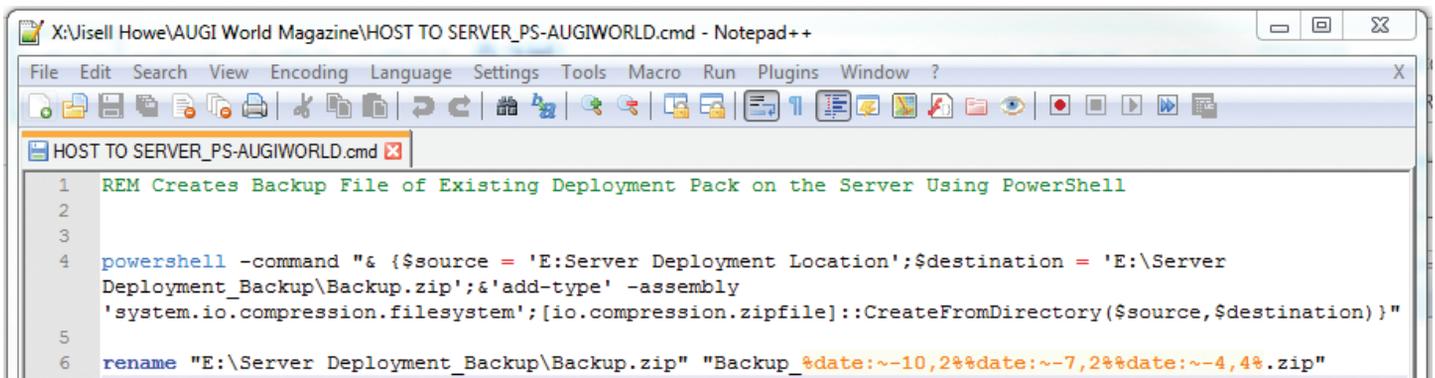
```

REM End of Script
echo msgbox „Your AutoCAD MEP is up to
date!“ > „%temp%\popup.vbs“
wscript.exe „%temp%\popup.vbs“

```

another tool that is likely already on your computer, is a very powerful method for automating tasks through scripts. In fact, it is so powerful, its default settings actually prevent anything from running automatically with it.

Take note that with great power comes great responsibility. I want to address this in the topic of scripts because the .cmd files I’m showcasing have some limitations. More specifically, my backup files are rather large, and I want to compress and zip



```
X:\Jisell Howe\AUGI World Magazine\HOST TO SERVER_PS-AUGIWORLD.cmd - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
HOST TO SERVER_PS-AUGIWORLD.cmd
1 REM Creates Backup File of Existing Deployment Pack on the Server Using PowerShell
2
3
4 powershell -command "& {$source = 'E:Server Deployment Location';$destination = 'E:\Server
Deployment_Backup\Backup.zip';&'add-type' -assembly
'system.io.compression.filesystem';[io.compression.zipfile]::CreateFromDirectory($source,$destination)}"
5
6 rename "E:\Server Deployment_Backup\Backup.zip" "Backup_&date:~-10,2&date:~-7,2&date:~-4,4%.zip"
```

Figure 5: Host to server script using PowerShell

```
13 REM End of Script
14 echo msgbox "Your AutoCAD MEP is up to date!" > "%temp%\popup.vbs"
15 wscript.exe "%temp%\popup.vbs"
```

Figure 6: Server to User Script with Message Box

them. During my research, I did not see a way to do this exclusively with .cmd files and commands, so I looked into how to use PowerShell as another tool in my quest for automation.

As I mentioned, the default settings may prevent scripts from actually running through PowerShell itself. While you could change or bypass these settings, your IT department may discourage this action. For that reason, I looked into using PowerShell in an interactive way within my .cmd files.

Figure 5 shows how this is done. Keep in mind that for the sake of the screenshot, I just made the text wrap around to the next line. Otherwise, just placing content on different lines can make a difference, just like the order of content in your script can make a difference.

While PowerShell itself can be an article topic all on its own, the key takeaway is that there is a source and destination defined for the backup file to grab files and ultimately reside, a means to compress said file, and then some code to rename the newly created zip file with today's date.

I purposely showed a different way to format the date in the file name this time around. There are different possibilities in formatting the date because just telling the file to rename with the system variable %DATE% tends to come back with forward slashes (i.e., 03/08/2017), which is not allowed in Windows file names.

Now just add in your Robocopy line to retrieve and copy files from your deployment assembly pack

location to your server deployment location at the end of this host to server script, and you won't be singing the blues much longer.

## 6. "SOMEWHERE OVER THE NETWORK"

Somewhere over the network, your files will fly. And your automation dreams will come true . . . as long as you write this final server to user script that gets everything onto any end user's computer.

This last step is crucial because your Autodesk software depends on it. The concepts and syntax are the exact same as Figure 2 except you are specifying the server deployment location as the source file path and the various file paths on your end user's computer as the destination file paths. Just like %DATE%, you can use %USERNAME% as a "placeholder" in your destination file path when you are trying to deploy files to any end user computer (i.e., C:\Users\%USERNAME%\AppData . . .).

This script is the one that the end user can click on and run to get the latest and greatest from the server. One point I want to add is that the end user might not know what a script is or what it is doing. For this reason, I added in one last bit of code to my server to user script that actually notifies the user with a popup message box that the transfer of files is done. In terms of syntax on Figure 6, it can be inferred that "echo" is just a means of displaying something, and in this case it is displaying the text that the user's AutoCAD MEP is up to date in a fun popup message box (Figure 7).

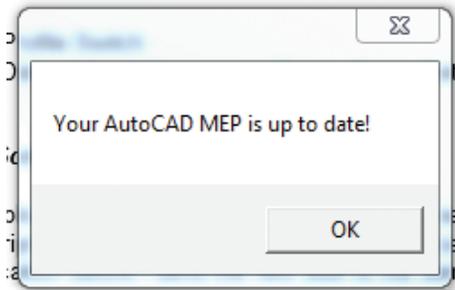
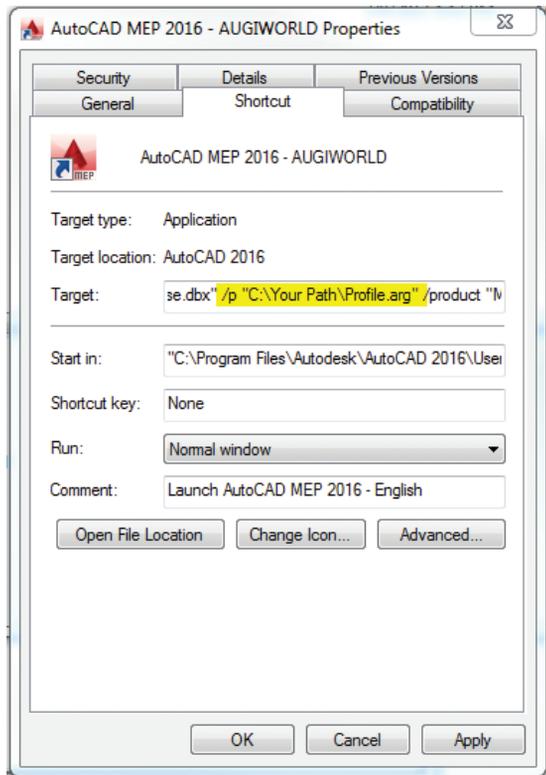


Figure 7: Message box

Keep in mind these scripts are just vehicles to transport files from one place to another. Additional setup may be required if you need AutoCAD on the end user computers to now read a bunch of new file locations for your custom content via a user profile. I created a copy of the AutoCAD shortcut and added a profile switch (See Figure 8), so it always runs the user profile I tell it to run.

The user profile drives a lot of the AutoCAD Options file locations, and it's imperative that the end user's AutoCAD setup is pointing to the correct file locations to run everything smoothly. I include this modified shortcut in my deployment pack for the end user to use when everything is deployed. Once these final points are squared away, you'll have reached the end of the rainbow where there are all sorts of other opportunities to use scripts for automation.





# Trending Technologies

**3**Ds Max and 3D art are tools for marketing that are used to sell ideas and products. In architecture, for example, the top visual companies have media teams who film, hire actors, produce stock video, and create entire productions to sell their luxury condos in the skyscrapers of Manhattan. They build the commercials and 3D is just one part of the production. All to capture the viewer's attention, and to do this, it is essential to maintain a pulse on the technologies helping to produce better media for 3D art and 3D content. So let's explore a few

technologies making their way into the world of 3D and 3D art to produce eye-catching results.

## PHOTOSHOP NEURAL FILTERS

Neural filters for Photoshop are powered by machine learning using Adobe Sensei. These filters reduce the time necessary to perform edits to imagery. Using a Neural Filter, Figure 1 from Adobe's site demonstrates the ability to add a smile to an actor's face.

## Original Image



Figure 1a: The original image with no filters applied.

## Smile Created using Liquify



Figure 1b: Liquify uses existing pixels from the image to adjust the smile.

## Smile Generated by Neural Filters



Figure 1c: Neural Filters generates new pixels to adjust the smile.

This technology is brand new, in beta, but growing. To access the filters select the Filter menu and then Neural Filters. Once there, you can download the

filters and view the waitlist displayed in Figure 2.

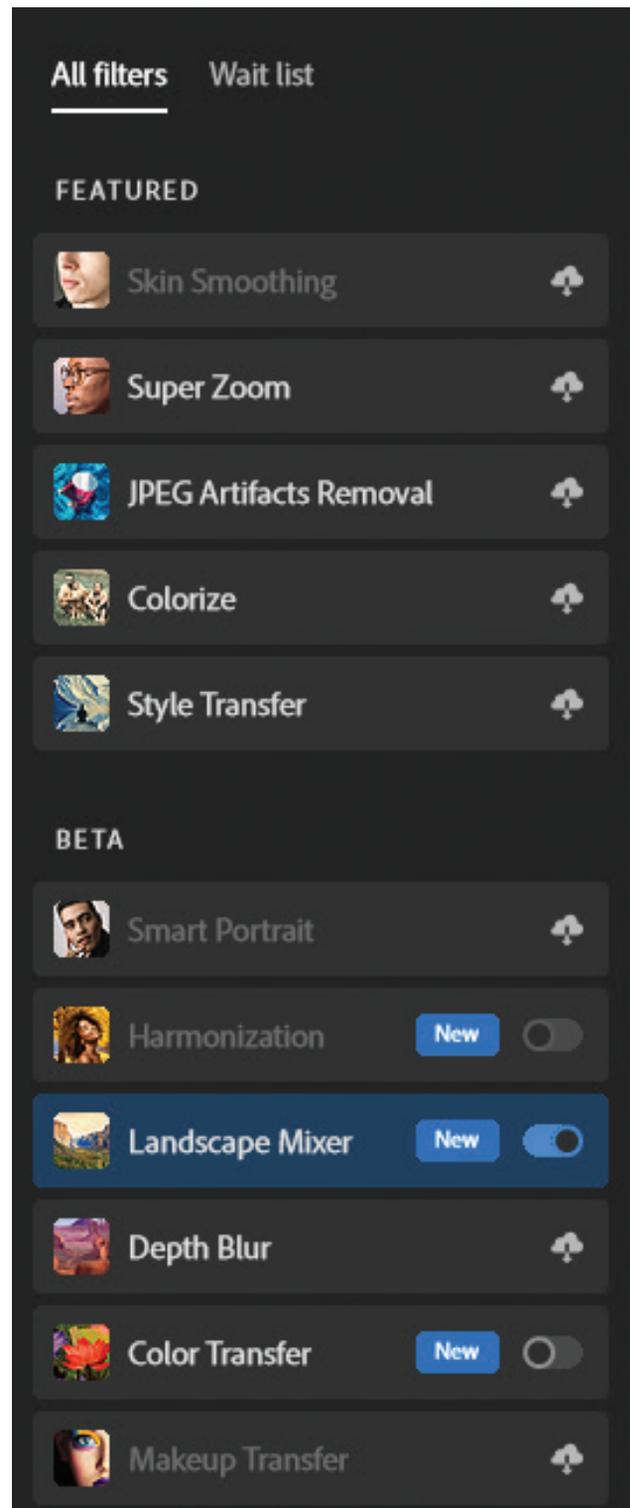


Figure 2: Wait List

Figure 3 displays using the Landscape Mixer to convert a photo to sunset. 3D artists can quickly generate incredibly unique and enticing video content with relatively simple (although tedious) effort by applying these filters to a video frame by frame.



Figure 3: Landscape Mixer

## EB SYNTH

EB Synth is another piece of software in beta that helps convert video to appealing imagery. See Figure 4, where the creator (Alicexz.com) converted the original video to the appealing style displayed.

## EMBERGEN

The last one I want to share is EmberGen. Atmosphere plays an essential part in 3D work and presentation, and for concept art and illustration capturing a mood and relaying that to your audience is essential. For 3D artists, atmospheric effects have been traditionally time-consuming and challenging to manage and implement. However, EmberGen developers made light the heavy lifting providing us with powerful software to implement



Figure 4: EB Synth by Alice X. Zhang

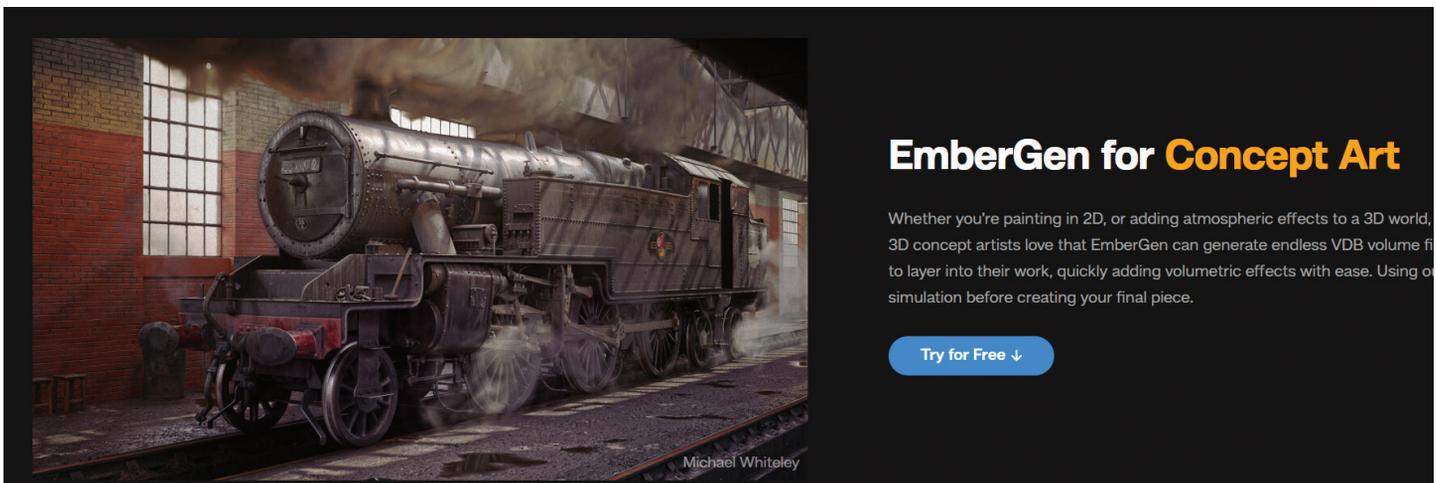


Figure 5: EmberGen for Concept Art

various effects to our scenes with an equally light learning curve. The software uses a node-based system to generate results and applies effects to mesh objects or display results influenced by geo. It even works with animated meshes. See Figure 5 for an example of effects applied to the train.



Brian Chapman is a 3D Artist, Autodesk Authorized Developer, and Application Specialist in Las Vegas, Nevada. He can be reached at [pictelforge@gmail.com](mailto:pictelforge@gmail.com)

# Making the Call

## Part Two



**B**ack in October, I brought up the subject of making decisions. How sometimes they get bogged down or become fuzzy as teams and individuals take too long to decide. I discussed how to speed things up and move quicker. Now for the flip side...

### **SLOWING THINGS DOWN**

There are decisions that are made so quickly that they seem like the whim or dictates of one person in the room. When a call is made too quickly, others are unable or unwilling to challenge a selection that was made. The choice is made so quickly that some

don't even know what was decided. Moving at fleet speed is good, unless there is a good reason to slow it all down. You need to do some due diligence. Sometimes you just need to slow it down. But no one wants to see people drag their feet and avoid progress. How can you slow things down and have a valid reason to do it?

## DOING DUE DILIGENCE

Good old Merriam-Webster provides two definitions for Due Diligence:

1. law: the care that a reasonable person exercises to avoid harm to other persons or their property
  - a: such diligence as a reasonable person under the same circumstances would use
  - b: the use of reasonable, but not necessarily exhaustive efforts
2. business: research and analysis of a company or organization done in preparation for a business transaction (such as a corporate merger or purchase of securities)
  - a: the care that a prudent person might be expected to exercise in the examination and evaluation of risks affecting a business transaction
  - b: the process of investigation is carried on... for the purpose of providing information with which to evaluate the advantages and risks involved

On a grand scale, it is the best practice for large decisions, like mergers or acquisitions, or major purchases. But I like to do it for most everything I purchase. It helps avoid the allurements of the "new and shiny". It provides a framework for a good outcome and makes sure that some thinking goes into everything. It involves reasonable efforts in a process of investigation by prudent people. Sounds good to me.

It is proper to slow down any decision making so that due diligence has time to play out. If you put this into practice on every decision then you can slow things down as needed on any decision. The requisite effort needed to provide this level of care depends on the scale of the project or purchase. More purchase money to be spent or expanded project scope would call for greater due diligence.

I want to move forward thoughtfully and without undue delay. I typically start with some basic questions and try to keep it as simple as I can. If they can be answered yes or no to these questions, that is great. Others may require greater digging. If so, then keep going. Question beget more questions and the investigation widens.

Here are some example questions I use when buying technology:

**Appropriate Use** – is this the right tool for the job?

**Best in Class** – is this the best tool for the job?

**Uniqueness** – are there other tools that we already own that do the same thing?

**Data Storage** – is it in the cloud? secured? backed up?

**Team Choice** – are key stakeholders on board?

**Share-ability** – Can others collaborate/see/use the data/files?

**Access** – Can those who need access, get it? Can we have differing levels of access?

**Security** – Can we lock out those who should not see the files/data?

**Version Control** – does the tool support document versioning?

**Scalability** – can everyone use it just as easily as one person? What if 100 people were using it at the same time?

**Software Compatibility** – does it play well with our other software tools?

**Hardware Compatibility** – Does it run on our current platforms (all variations)?

**Extensible** – can we add additional features via programming efforts? Does it have an API?

**Workflow** – does it enhance/embrace/adjust/overturn our workflow?

**Training** – is it easy to use by the average user? Is training provided?

**Licensing** – is it named user/subscription/concurrent user?

**Administrative** – who is the champion for this tool? Who will manage the user accounts?

**SSO** – how is user authentication managed?

**Lost Opportunity** – if we do not buy this tool, what will happen?

While the list looks long, it can be pared down depending on the tool you are reviewing. It might be a short conversation or email thread with the vendor or staff. It can be done quickly, or it can open up good conversations and investigation channels that might uncover that the tool is not a good fit.

One of the best questions that most people overlook or do not think through completely is “How will this be used?” You should have people think about how tools will at a deep, process level. Again, it can be done quickly. Just have them picture how the tool would fit, who would use it (and who will push back), when it would be used, what would stop being used because it replaces, how you get it in everyone’s hands and get them fully trained... think from cradle to grave in a project environment. This can uncover areas that might cause problems or failures.

**THE GOAL OF DUE DILIGENCE IS TO MAKE GOOD CHOICES AND PURCHASES, NOT TO SLOW THE PROCESS DOWN. IT SHOULD INCREASE SUCCESS NOT STIFLE IT.**

## **SLOW DOWN TO FIND AND REDUCE THE RISKS**

What if some of the questions in your review bring up causes of concern? Then you slow down again... You need to define the impact and reduce the risk if possible. Just knowing the potholes in the road allows you to fill them in or avoid them. You cannot eliminate all risk, but you can identify it and measure the exposure to risk that might impact success.

Since you cannot eradicate all risk, you can categorize it a “possible” or “probable”. Just because something “might” happen does not mean it will. If you can measure the probability, then you can prepare for troubles before they come. By reducing risk, you increase the chances for positive impact.

## **KNOW THE EXPECTED GAINS AND DEAL BREAKERS**

If you can define productivity gains, then you can frame a conversation of where the new tool helps. Try to quantify the reduction in hours over time, the increase in accuracy, the flow of data, the avoidance of known headaches and more. For each of these, anchor them in reality, not your dreams. They need to be defensible and measurable after the tool is in place.

Just because the new tool does not meet every expectation does not mean it will not improve your productivity. Know your deal breakers and be ready to walk away from technology that just does not improve things enough right now. Check back later and maybe future upgrades will improve it.

## **GO DOG GO**

Not just a fun book to read to your kids... but lessons to learn about green lights and red lights. When entering into decisions and making the call you should be ready to speed things up if you see green lights, or slow them down when you see red. With practice you can know which gear you should be in as you head down the technology highway.



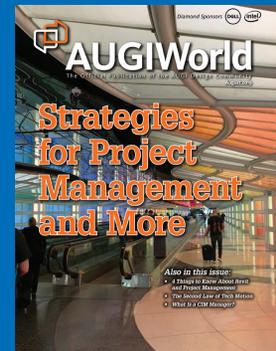
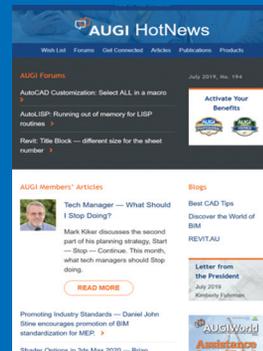
*Mark Kiker has more than 30 years of hands-on experience with technology. He is fully versed in every area of management from deployment planning, installation, and configuration to training and strategic planning. As an internationally known speaker and writer, he is a returning speaker at*

*Autodesk University since 1996. Mark is currently serving as Chief Technology Officer for SIATech, a non-profit public charter high school focused on dropout recovery. He maintains two web sites, [www.caddmanager.com](http://www.caddmanager.com) and [www.bimmanager.com](http://www.bimmanager.com).*

# Connect with AUGI Members

Increase your revenue by advertising with AUGI and reaching its 400,000 Members.

AUGIWorld Print  
AUGI HotNews Digital



AUGI.com Website

Thousands of visits per month

Forum Advertising

Active membership participation

- Advertising Email Blasts
- Targeted Mailings
- Industry Leading Google Results Site

 **AUGI** Visit [www.AUGI.com/advertise](http://www.AUGI.com/advertise)

Welcome to *AUGI*WORLD Inside Track! Check out the latest opportunities to advance your skills, processes, and workflows in your firm with the most current AEC-related software and hardware updates available.



## FUTURE

<https://apps.autodesk.com/RVT/en/Detail/Index?id=4627376098925758972&appLang=en&os=Win64>

**Autodesk Revit: 2022, 2021, 2020, 2019**

Future is a plug-in for Autodesk® Revit® that contains a set of features that can significantly speed up the creation of models, as well as the release of documentation for engineers, architects and BIM specialists. The plugin provides several ways to improve the efficiency of work in Revit.

We list just a few of the 36 functions: Sheet numbering, Copying sheets, Reinforcement of walls/columns/beams, Copying properties, Autosizing, Remove reinforcement constraints.

The main feature of the program is its gradual filling and optimization. The program is provided with a paid subscription. By purchasing a subscription, you get access not only to the functions that have been created at the moment but also to those that will be added to the program over time. No intrusive notifications, automatic updates, affordable price and simple interface. There is a 30-day trial version, as well as an educational one - a free license key is issued for students. Functional characteristics of the software:

Shared - tools for a specialist of any profile

1. Numbering of sheets
2. Copy parameters
3. Marking. Active view.
4. Marking all elements
5. Crop a view
6. Print PDF
7. Copy schedules
8. Print sets
9. Put \ Delete signature
10. Room parameters
11. Export tables to Excel
12. Importing tables into Revit
13. Copy sheets
14. Work sets
15. Duplicate values
16. Transfer properties to stained glass

17. Copy openings
18. Cut openings
19. Autosave

Constructive - tools for structural engineers working with reinforced concrete structures

20. Wall reinforcement
21. Number of frames
22. Update mark of assemblies
23. Copy assemblies
24. Column reinforcement
25. Beam reinforcement
26. Stairs reinforcement
27. Reinforcement of floor openings
28. Area reinforcement
29. Pass properties to rebar
30. Remove \ Install reinforcement constraints
31. Pile numbering
32. Enable \ Disable analytics

Dimensions - annotation tools for any professional

33. Dimension line
34. Connect. Dimensions.
35. Move. Dimensions.
36. Autosize. Floors.

Software performance:

1. A unique technique for constructing reinforcement
2. A large number of optimization tools in one program
3. Fast installation
4. Support for multiple versions of Autodesk Revit
5. Simple and intuitive interface
6. Technical support directly with the developer
7. Multilingual interface



## QUICKFIELDEDIT

[https://apps.autodesk.com/ACD/en/Detail/Index?id=3273393530658835904&appLang=en&os=Win32\\_64](https://apps.autodesk.com/ACD/en/Detail/Index?id=3273393530658835904&appLang=en&os=Win32_64)

If you have some news to share with us for future issues, please let us know. Likewise, if you are a user of a featured product or news item and would like to write a review, we want to know. [brian.andresen@augi.com](mailto:brian.andresen@augi.com)

**Autodesk AutoCAD, AutoCAD Electrical, AutoCAD Mechanical, AutoCAD Architecture, AutoCAD MEP, Civil 3D®, AutoCAD Plant 3D: 2022, 2021, 2020, 2019, 2018**

QuickFieldEdit is an AutoLISP plug-in to make the AutoCAD fields much easier to use.

It replaces the object linked to a field while all the other field's settings remain the same - property, format, precision, conversion factor, prefix and suffix. Works for fields in single-line text, multiline text, block attributes and tables. No complicated dialogs - just two clicks and the field shows the previously selected property of the new object.

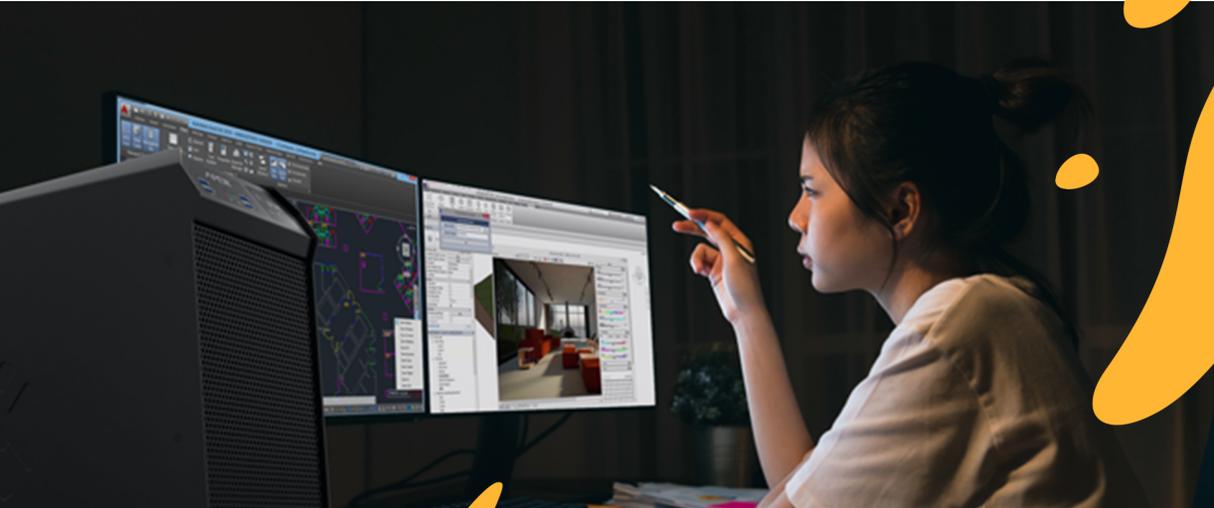
## **AUGIWORLD brings you recent developments in Autodesk and related software items**



# Powerful Revit Performance from Anywhere

Download our free eBook and discover

## BOXX Cloud



Purpose-built for AEC, BOXX Cloud frees architects, engineers, and BIM managers from desk side computers, enabling them to work remotely with secure, seamless, state-of-the-art, cloud-based performance for Autodesk Revit and other CAD applications.

Combining security and high-performance with leading expertise, BOXX Cloud virtual private cloud service is the remote work solution that makes more sense than ever before, so download our free eBook filled with vital information you won't find anywhere else.

[Get the eBook »](#)

**BOXX**

**AMD**

**NVIDIA**

**intel**

(888) 302-0223  
512-852-0400  
[boxx.com/augi](http://boxx.com/augi)