

BLD225153

20 Practical Uses of Dynamo for Revit to Improve Team Efficiency

Matthew Anderle

AECOM

Denes Zubcsek

AECOM

Learning Objectives

- Learn how to generate graphs in a clear and organized fashion to make them intuitive for team members to use
- Learn how to apply Dynamo for Revit for managing data
- Discover the application of Dynamo for Revit for automating object placement and manipulation
- Learn how to increase the applicable functionality of graphs across multiple disciplines

Description

Often project teams find themselves executing repetitive work in relation to drafting, data entry, and coordination. If you have ever thought that there must be a better way, or your ears perk up when you hear the phrase "there's an app for that," this will retain your attention. We explore the common, daily uses of Dynamo extension for Revit software to improve project team efficiency, and we'll illustrate how you can create graphs that are functional across multiple disciplines, are project agnostic, and will win you praise at your office. Through the exploration and application of practical Dynamo for Revit workflow concepts, project teams can anticipate improved efficiency and productivity through various phases of the project. The key concepts being explored in this document include:

- Isolating team tasks which could benefit from automation.
- Generating graphs in a clear and organized fashion to ensure they are intuitive for team members to use.
- Managing data through Dynamo for Revit.
- Automating object placement and manipulation.
- Outlining best practices for selecting node sequences in lieu of custom Python code.
- Exploring ways to preview results of Dynamo graphs without fully executing.
- Increasing the applicable functionality of graphs across multiple disciplines.
- Discussing the topics of the distribution and compatibility of Dynamo graphs.

Dynamo provides its users the ability to extend BIM by employing data and logic via a graphical algorithm editor. Because Dynamo is an open source tool, a community of users contributes to its continuous improvement. The 20 practical uses of the software described in this paper will aid teams in streamlining their workflow to improve their productivity.

Speakers

Matthew Anderle is the Building Information Modeling (BIM) director for the Buildings+Places business line of AECOM, with focus on the Americas. He is a BIM and technology evangelist with over 18 years of experience establishing global BIM workflows and standards around content, training, interoperability, and BIM consultation as a service. His experience spans over multiple market sectors with emphasis on large healthcare facilities, data centers, aviation, government projects, and science facilities. Mr. Anderle serves AECOM as a leader in the advancement and efficient implementation of BIM processes for a variety of project types. He manages and directs large distributed project teams to successfully implement BIM collaboration workflows, enabling global offices to work as one entity.

matthew.anderle@aecom.com

Denes Zubicsek is a Project Coordinator for the Architecture department in Columbus, OH and uses his passion for technology and innovation to compliment the practice of architectural design and documentation. He continually pushes the boundaries of visual programming with regard to computational design, quality assurance, and document automation through applications including Dynamo and Revit. Mr Zubicsek works closely with regional and national leadership to assist in the development and propagation of efficient processes throughout AECOM.

denes.zubicsek@aecom.com









Intuitive Graph Organization

It is beneficial for project teams to understand graph organization. The use of a consistent color key aids in identifying node groups that receive and output information, and indicates where to find key graph functionality.

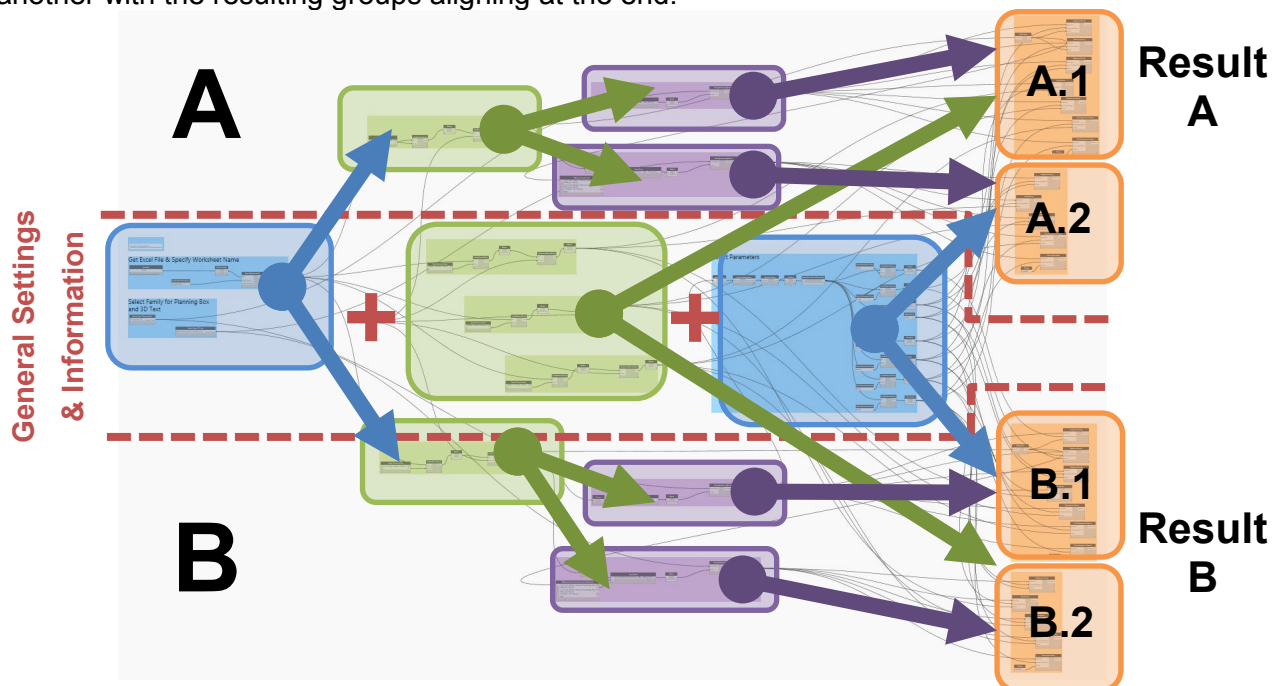
Color Key

AECOM uses a system of color-coding to identify and group Dynamo graphs. The graphs illustrated in this document use the following palette:

	Blue	User Selections, Settings, and Credits
	Green	Discovery and Retrieval of Information
	Gray	Processing Data
	Pink	Creating Elements in Dynamo
	Purple	Creating Elements in Revit
	Orange	Writing Data

Organizational Graph Flow

It is important to arrange your Dynamo Graphs to be intuitive collections of items. This can be accomplished by creating groups of node-routines which perform a particular function within the graph. Arranging these groups in a timeline format will help illustrate flow from one group to another with the resulting groups aligning at the end.



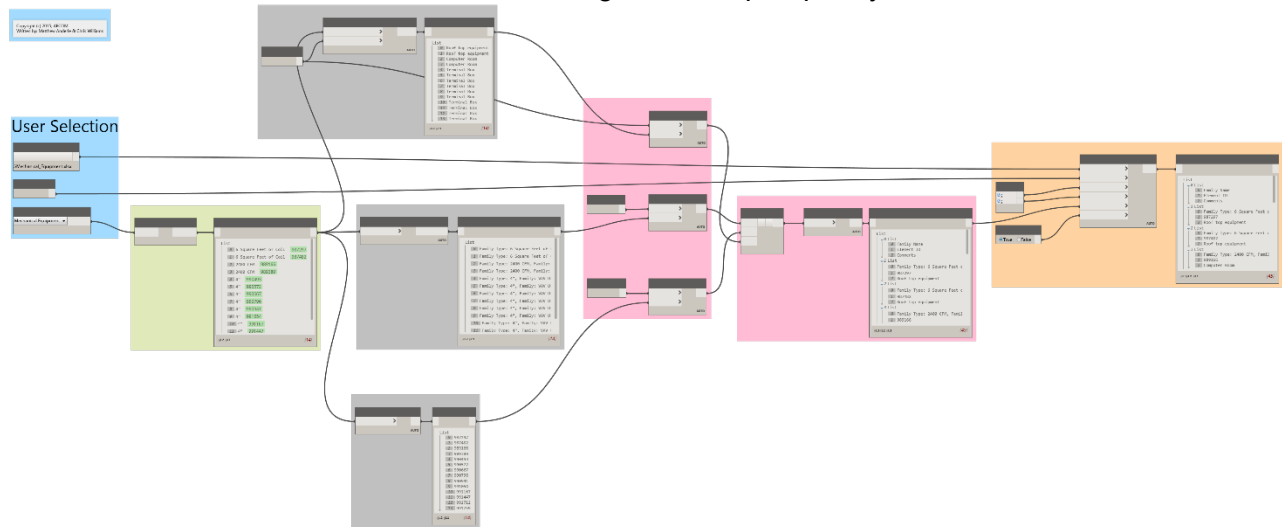


20 Dynamo Graphs

Output Data

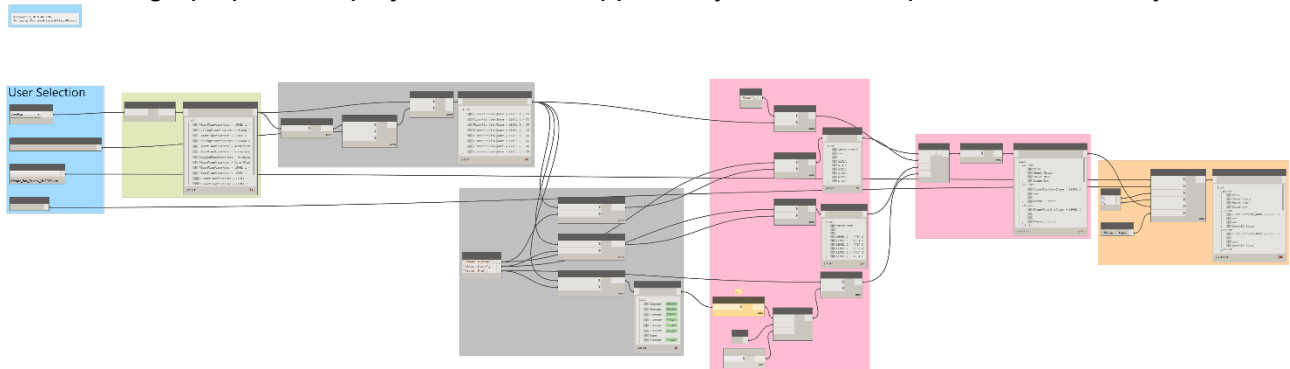
1. Equipment to Excel

This graph selects all elements of a specified category, reports a parameter, and exports to Excel with headers. This can be used to manage data output quickly for evaluation use.



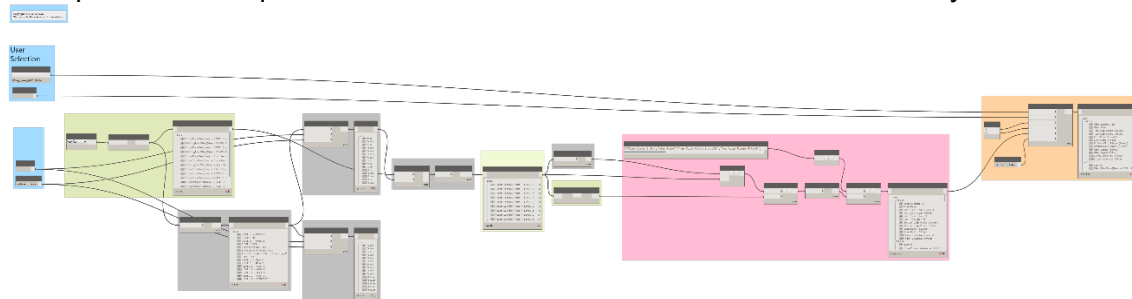
2. Scope Box Review

This graph identifies all view plans and exports sheet number, sheet name, and scope box name to an Excel file with headers. It also provides the ability to filter the view based on a string value. This graph provides project teams the opportunity to review scope box allocation by view.



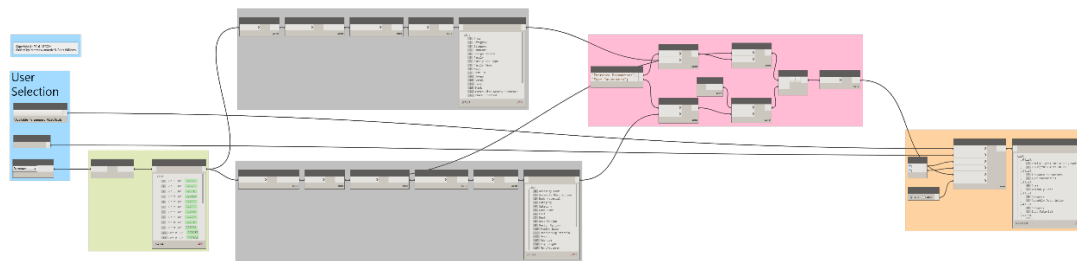
3. View Range Manager

This graph identifies all elements of view plans and exports view range parameters to an Excel file with headers. It also provides the ability to filter the view based on a string value. Accurate and consistent view configurations which are not always controlled by view templates based on team preference require review and validation to maintain visual continuity.



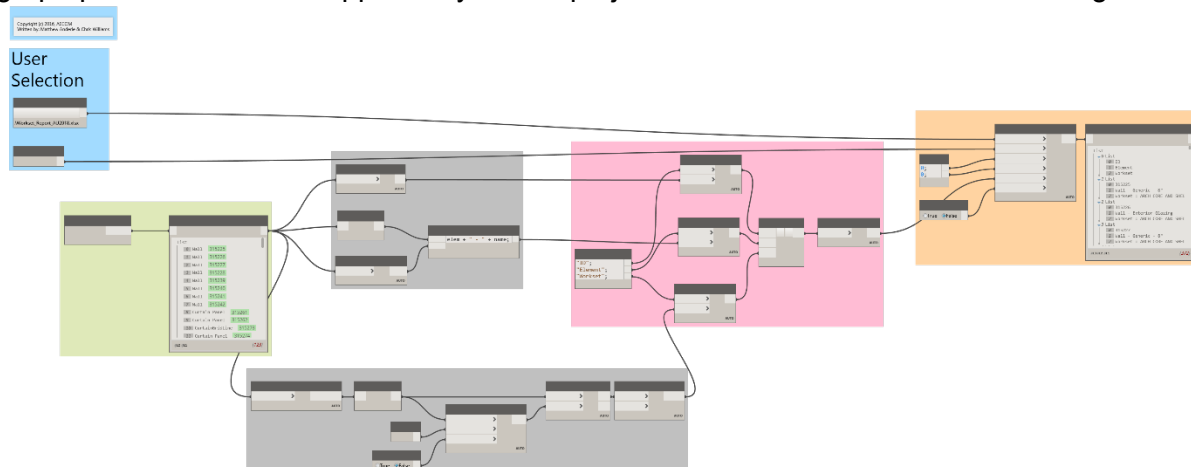
4. Parameter Investigator

This graph reports all available instances and type parameters of a selected category and writes to Excel. This provides an efficient method of reviewing all available parameter of a given element category.



5. Workset Reporter

This graph selects all elements in the active view and creates an Excel file that reports every element and its associated workset. While worksets are not able to be scheduled in Revit, this graph provides a QA/QC opportunity for the project team to review model element organization.





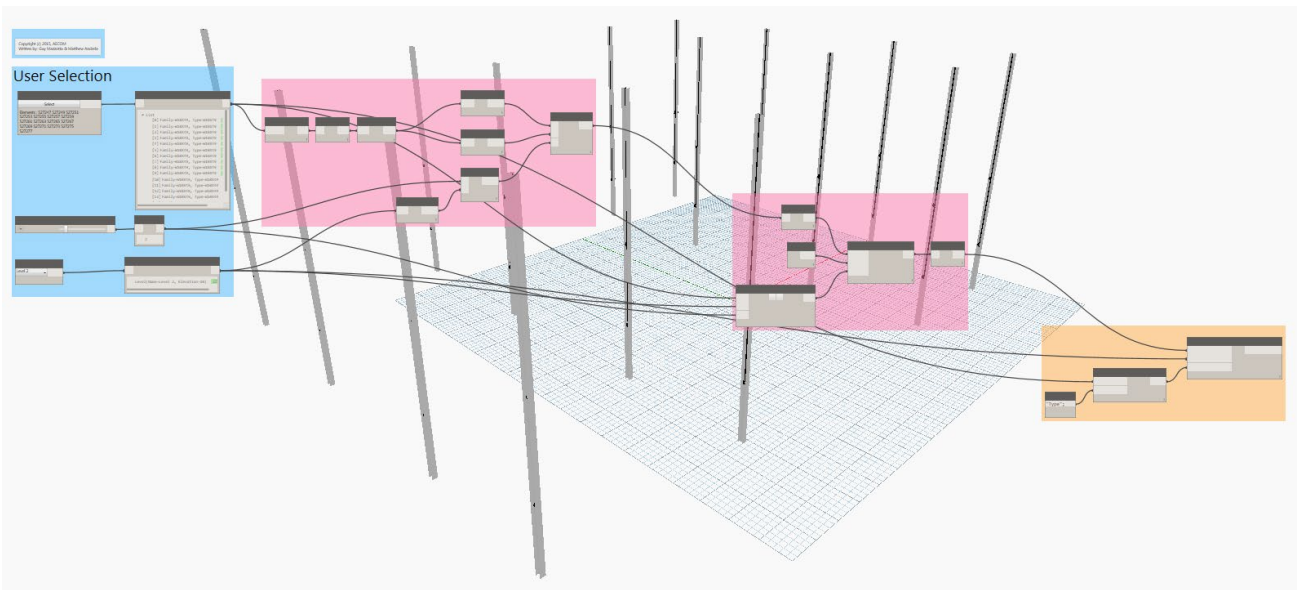
Visualization

6. Column Splicing

This graph utilizes a select node to to:

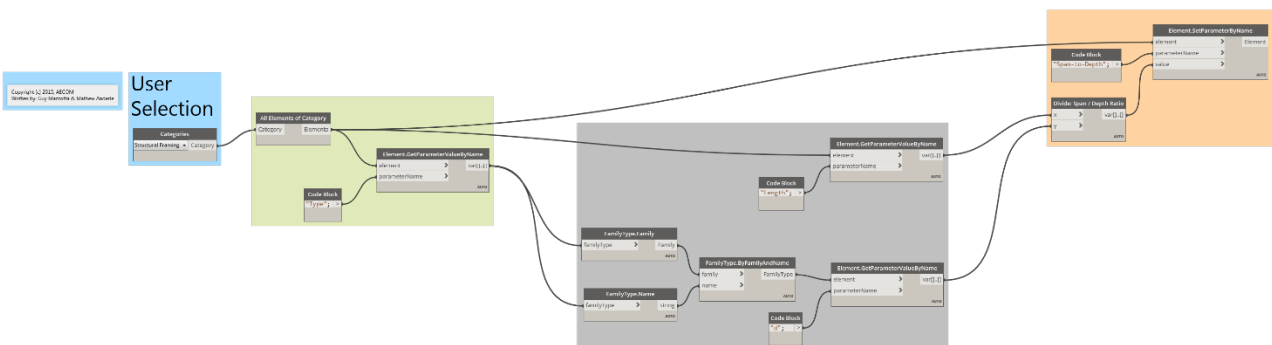
- Select columns
- Input a splice offset above a specified level
- Set the original elevation at top before splicing
- Set level of column
- Set top offset of column
- Modify Revit elements

This graph adds the ability to correct multi-story columns across the project more efficiently and precisely.



7. Span-to-Depth

This graph provides a QA/QC opportunity to verify that span-to-depth ratios are being met in accordance with structural design criteria. This graph uses typical framing spans compared to beam depths and a view filter to suggest areas which may need further validation.

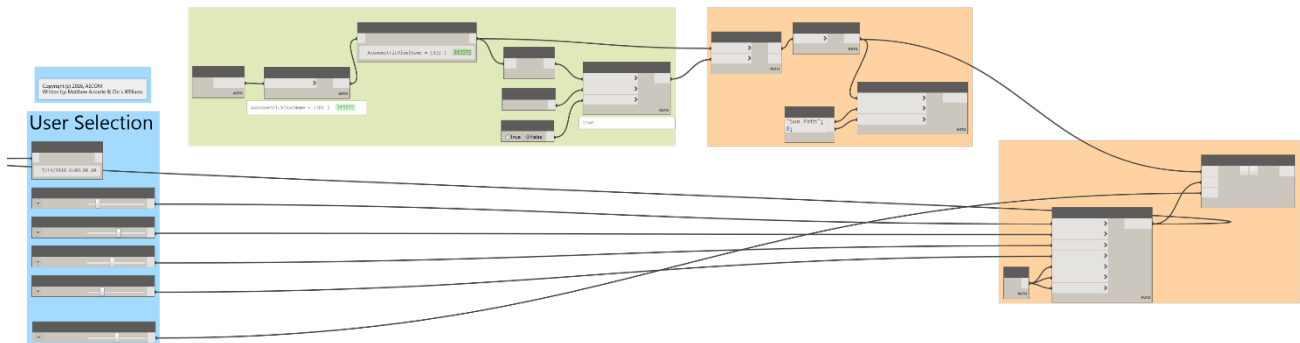




8. Slider Sun Settings

This graph identifies the presence of an axonometric view and if true, allows the user to access a slider scale which modifies sun settings based on date and time inputs for the following:

- Year
- Month
- Day of the month
- 24-hour time period
- Shadow intensity



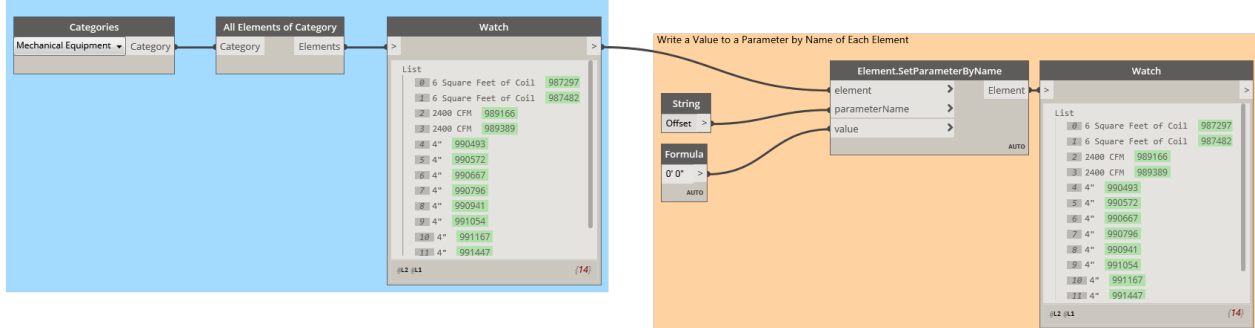
Process Data

9. Write to Mechanical Equipment

This graph retrieves all elements of the category Mechanical Equipment and allows the user to input a value to a parameter providing the team an efficient means to populate large amounts of data without the need to generate schedules.

Credits
Copyright (c) 2015, AECOM
Written by: Matthew Anderle & Chris Williams

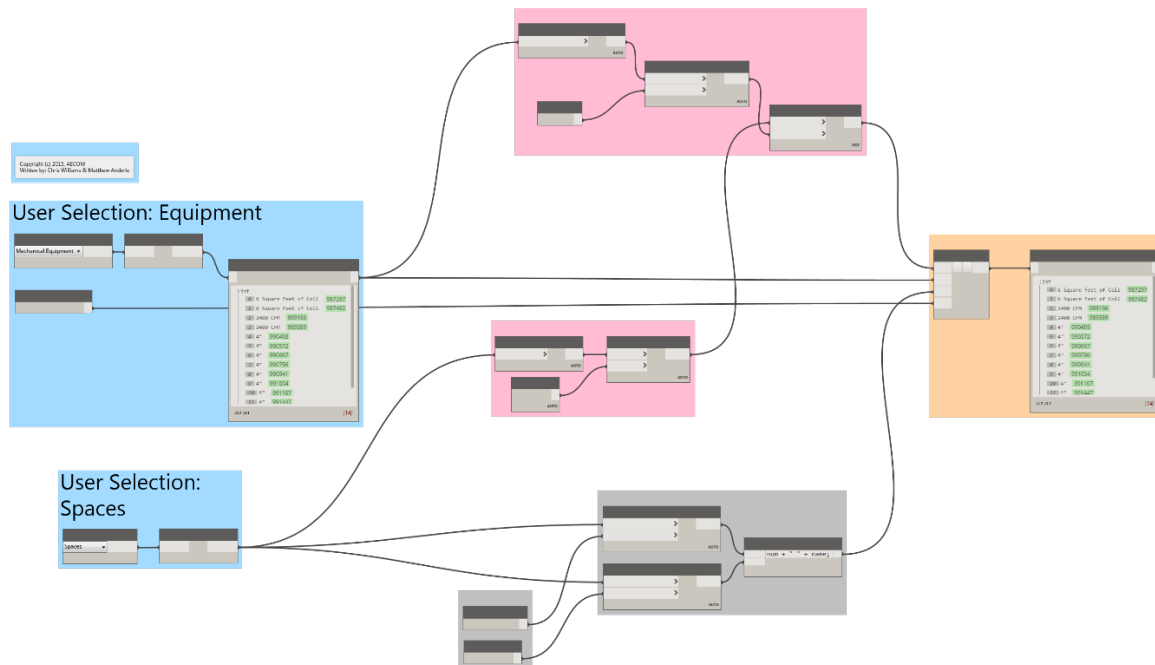
User Selection





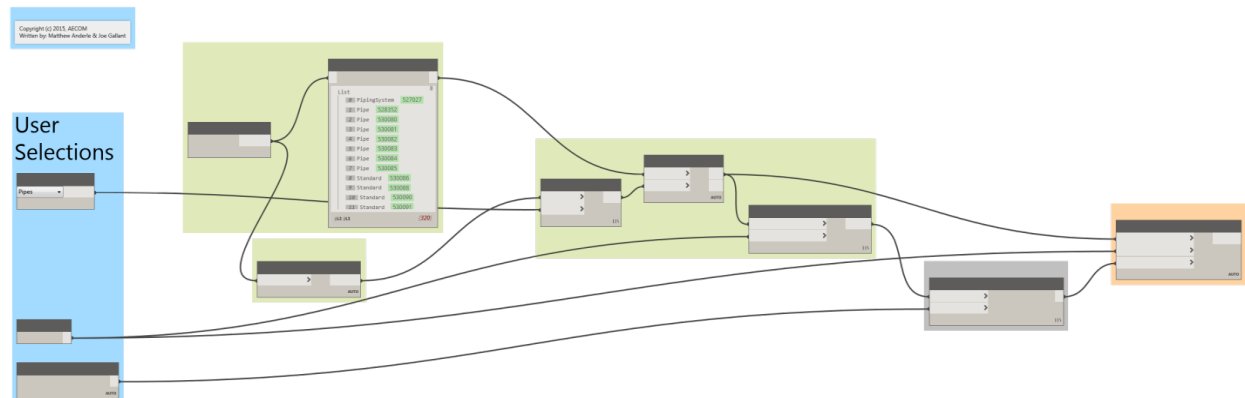
10. Auto-Set All Equipment Locations by Space Name

This graph sets all elements of a Space or Room category, creates a bounding box of those Spaces or Rooms, and evaluates whether another category is contained within the bounding box. It populates parameters from the Space or Room to the specified element category. Use this graph to set the equipment location where room and/or space identifiers are not included in families.



11. Pipe Height Adjust

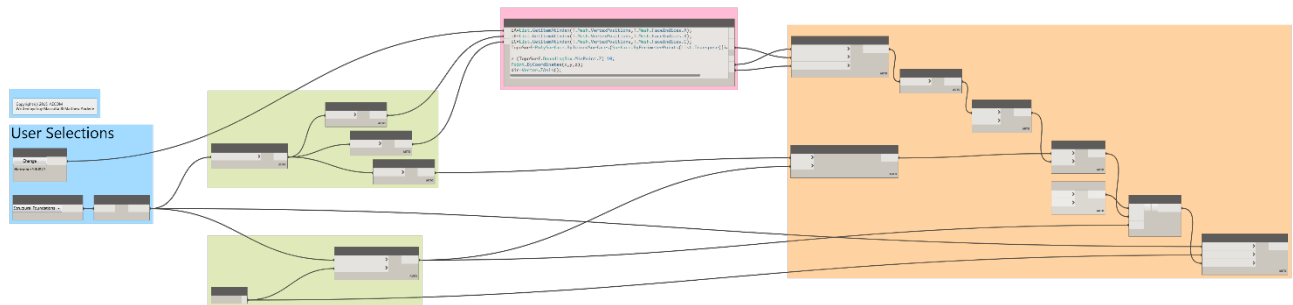
This graph selects all elements of the Pipe category in the active view and adjusts the current offset height to a new user input offset height. This graph will quickly adjust piping distribution systems when a change in height between floors occurs.





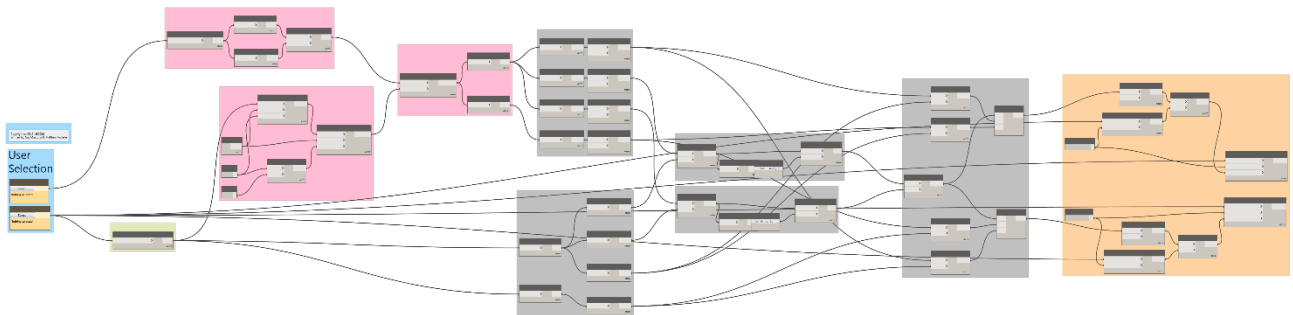
12. Caissons to Topography

This graph allows the user to select a Revit topographic element and adjust all structural foundations depth-to-rock parameter to the selected topographic element. This graph allows the structural foundations to adapt to changes or refinement in topographical surveys quickly and accurately.



13. Match Elements to Reference Plane

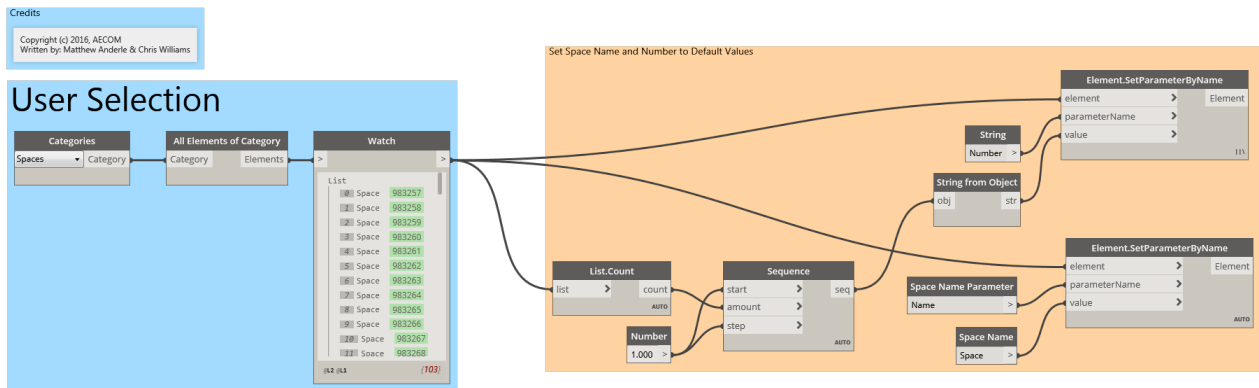
This graph adjusts the start and end offsets of each structural member to a reference plane. This can be used to match sloped roofs or slabs.





14. Space and Room Data Clear

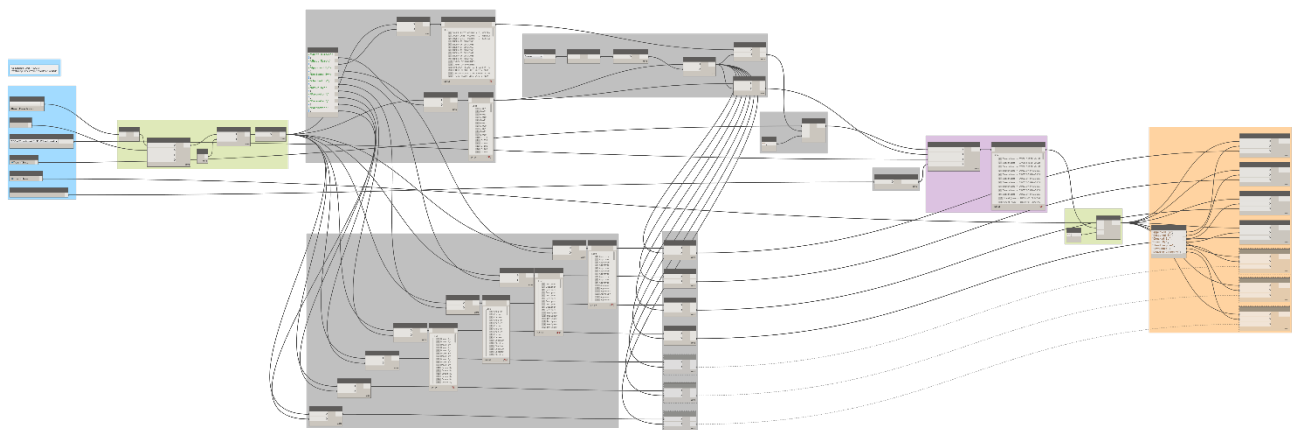
This graph selects all Spaces and resets the name parameter to a user input value; and the number parameter with a user input starting number. This can be used to reset spaces to a sequential numbering system and can be used as a starting graph for multiple data writing purposes.



Creation

15. Sheet Creator

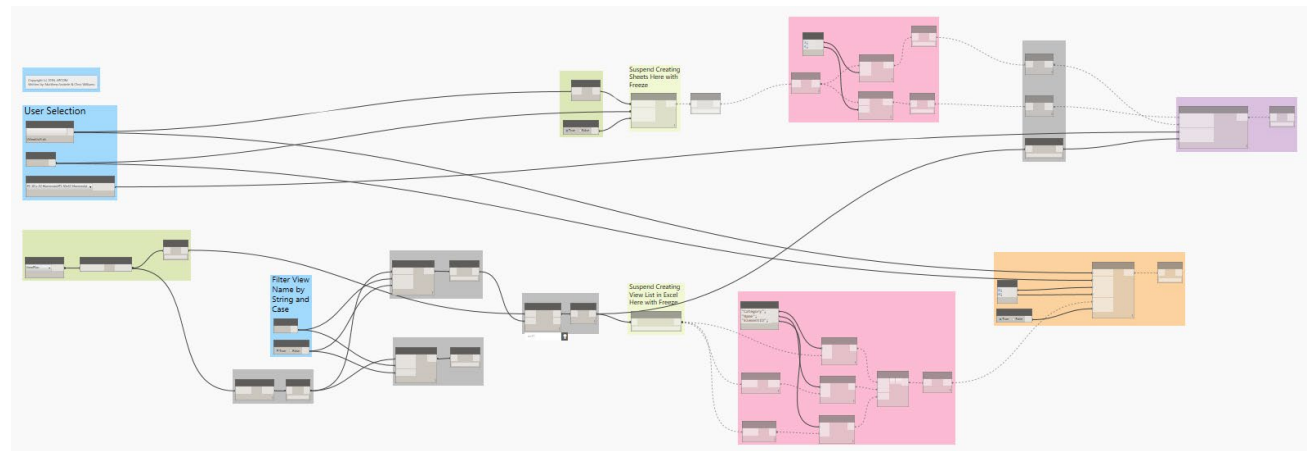
This graph references a user generated Excel file with a predefined sheet index, allows the user to select a title block family, and inputs a series of parameter values, filters any existing sheets, and creates the remaining list as new sheets in Revit. This can replace or compliment sheet set creators and provide the project team an efficient approach to sheet generation.





16. Place Views on Sheets

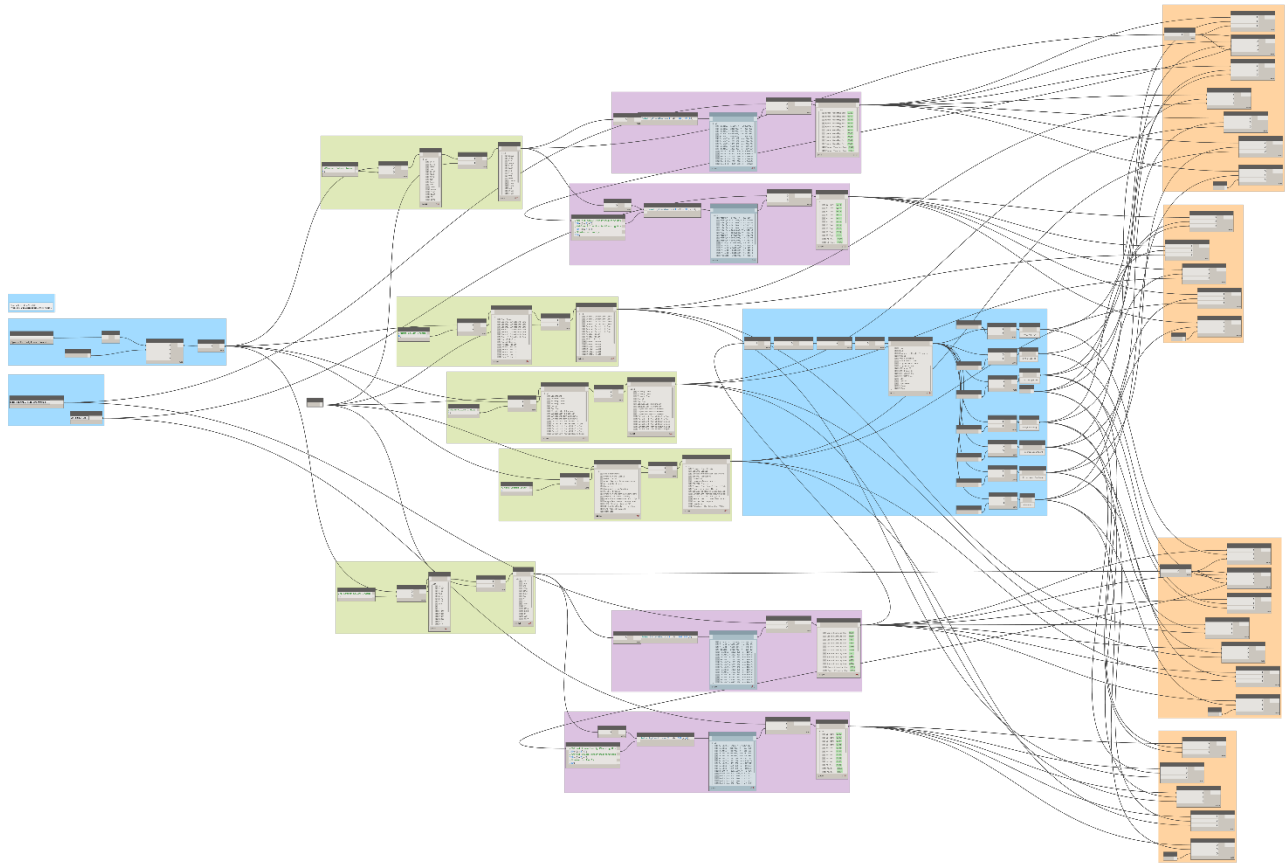
This graph has two functional paths: one which captures a view list that can be filtered by the project team and outputs the data to Excel; the other path reads the Excel file with the addition of the sheet name and number and creates new sheets with the selected title block, and places the corresponding view on the sheet.





17. Excel Square Footage-to-Family Instance

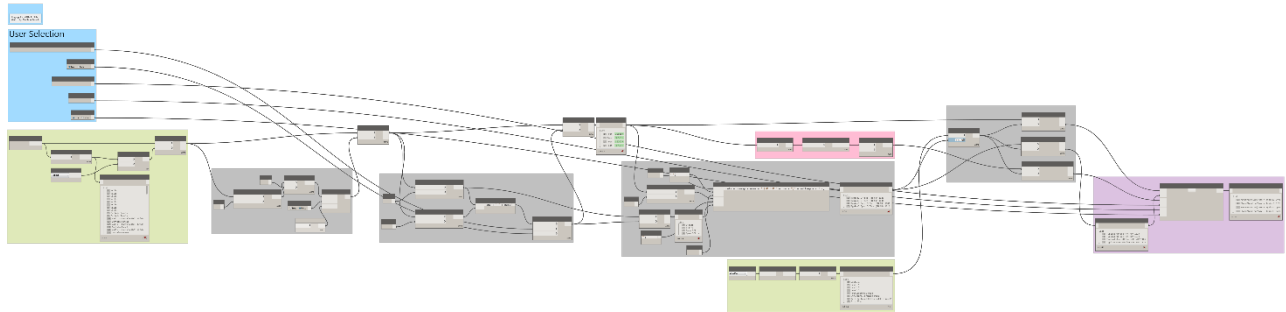
This graph reads an Excel file with design program data, which is divided into two columns: one with planned size and the other with existing size of rooms or departments based on area. This graph then places a family instance, sizing it to match the planned and existing areas. A three dimensional text family accompanies each family instance. Parameter data is then transferred from the Excel file to each family. This graph provides graphical planning tools for the project team to study program adjacencies and quickly assess planning strategies.





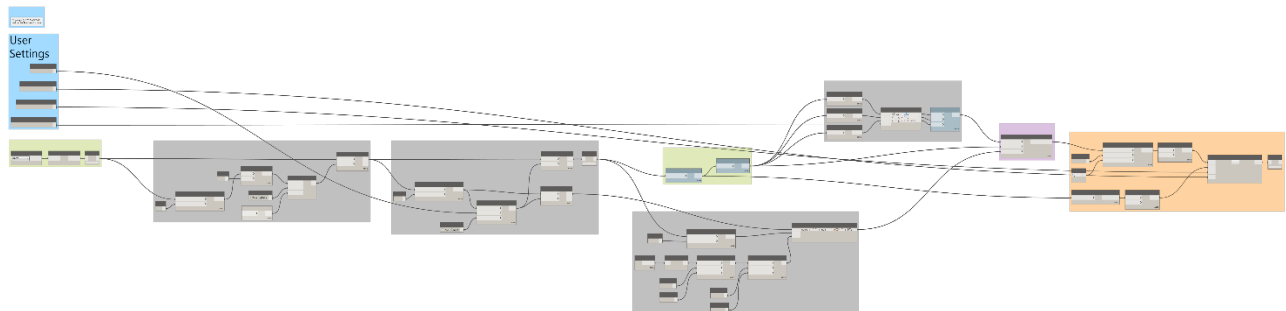
18. Create Room Plans Cropped to the Room

This graph captures all rooms in a view and filters that list against user defined name or number queries and generates a new floor plan with a crop region set to a user defined offset value. A view template can also be applied at the time of view creation. Project teams can use this graph to quickly generate typical room floor plan views and apply a common view template.



19. Create Room Axonometric Views Section Boxed to the Room

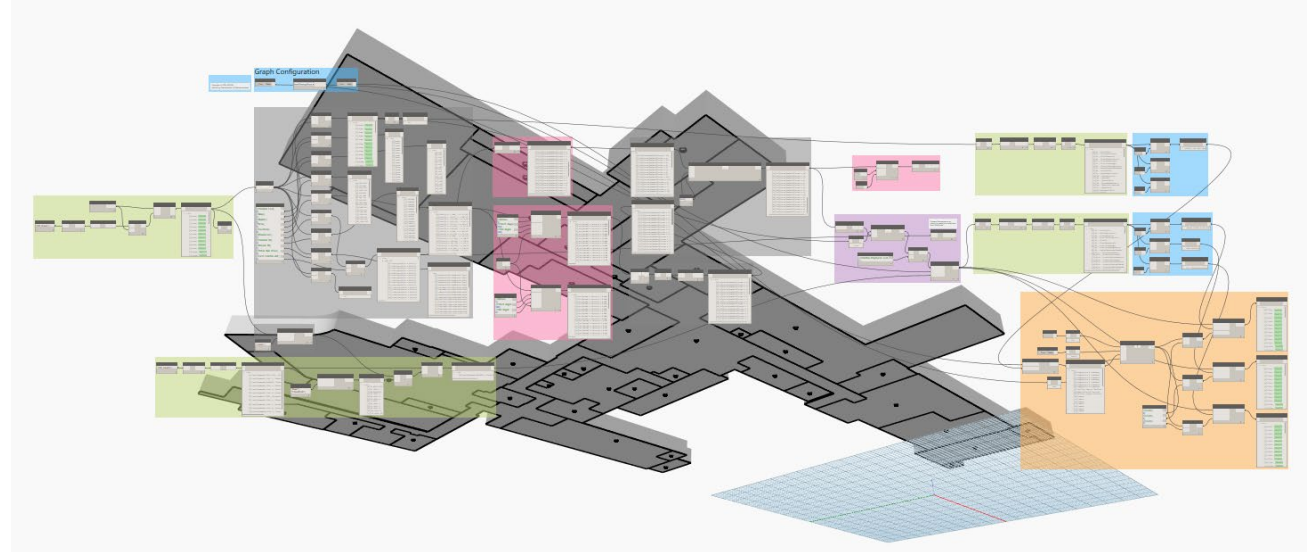
This graph captures all rooms in a view and filters that list against user defined name or number queries and generates a new axonometric view, section boxed to a user defined extent outside the room, with user defined offset values for eyepoint height. A view template can also be applied at the time of view creation. Project teams can use this graph to quickly generate typical room axonometric views and apply a common view template.





20. Area Boundary-to-Floor Element

This graph identifies areas per level, generates boundary curves in Dynamo, and converts those boundaries into new floor element objects while transferring data from the original area object to the corresponding floor. This allows the project team to visually represent area stacking diagrams in three dimensional views. The use of view filters against departmental parameter data can quickly color code these stacking diagrams.



Conclusion

By integrating Dynamo for Revit workflow concepts into the execution of a project, data entry and repetitive work can be streamlined to improve efficiency and productivity across all disciplines, and through various phases of the project. Proper graph generation, automation, and data management will allow project teams to isolate tasks, facilitate collaboration, and enhance communication to optimize project delivery.

For a Limited Time Download the Dataset! <https://bit.ly/2RskFvQ>

Helpful References

Dynamo

<http://dynamobim.org>

<https://forum.dynamobim.com/>

Revit API

<https://forums.autodesk.com/t5/revit-api-forum/bd-p/160>

Python Standard Library

<https://docs.python.org/2/library/index.html>

Jeremy Tammik – The Building Coder

<http://thebuildingcoder.typepad.com/>

C# & VB.Net Code Converters

<http://www.developerfusion.com/tools/convert/csharp-to-vb/>

<http://converter.telerik.com/>



