

BES500065

Structural Engineering Workflows for Interoperability and Lifecycle Analysis

Mark Rocha P.E., S.E.



Learning Objectives

- Overview of the current tools available for interoperability between structural analysis and BIM
- Intro to BHoM, Installation of BHoM, and Intro to BHoM Adapters
- Using Dynamo or Grasshopper to generate a Robot analysis model and a Revit model
- Getting started with Lifecycle Analysis using BHoM to evaluate embodied carbon

Description

Released in 2018, BHoM (Buildings and Habitats object Model) is an open-source platform that enables open workflows between many software packages within the AEC industry. Created by Buro Happold, this tool is especially relevant to structural engineers. With this tool, users can develop structural analysis models using Dynamo, Grasshopper or Excel and then generate structural analysis models to most commercial software packages. Users then can generate Revit models for documentation. In this class, you'll get an overview of the current state of structural analysis workflows and how BHoM can add value to your structural workflows. We will also demo several analyses to documentation workflows to show the capabilities of the tool. Lastly, we will show how users can run Lifecycle Analysis using BHoM to evaluate the embodied carbon of buildings.

Speaker(s)

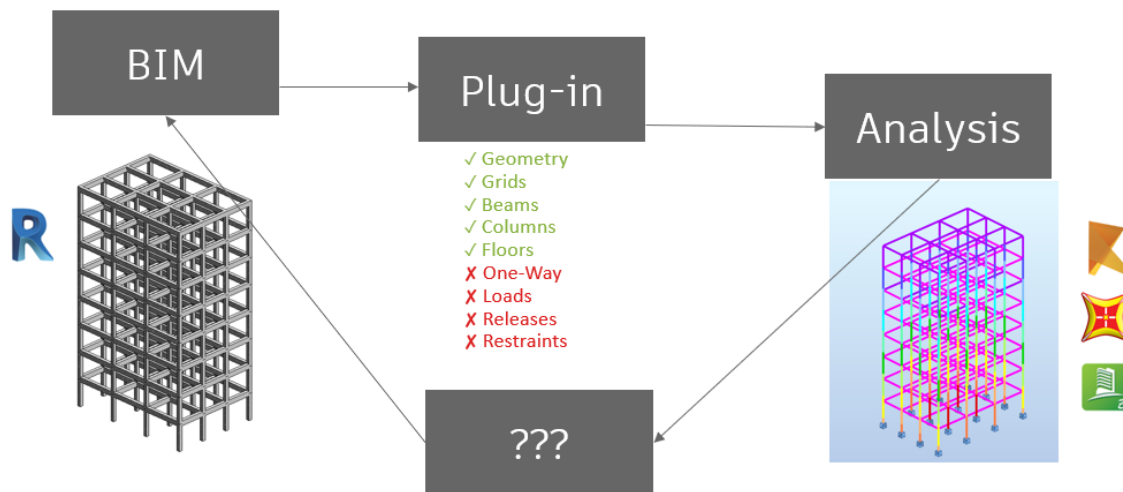


Mark is the Structural Digital Practice Lead at HDR, responsible for the design and analysis of the structural components of the project. Mark's expertise is within BIM, computational design, and structural interoperability workflows to assist project teams. His work ranges with hospitals, courthouses, and laboratories. Mark has spoken at BILT North America on structural engineering workflows and is a guest lecturer at the University of Texas at Austin Architectural Engineering program.

Current Workflows in Structural Engineering

One of the common challenges with engineers and BIM managers is how to move structural analysis data from one software platform to another. This challenge, commonly referred to as interoperability, can be a thorn in the side of engineers who would prefer to focus on design instead of manual data translation. With the advent of Revit and other tools, the ability of translating data has become more efficient and more accessible, however, most of these tools are proprietary or one-off custom software tools.

Typical Process for Structural Engineering:



Typical Process for Engineers

Typical Process for Structural Engineers can be outlined as follows:

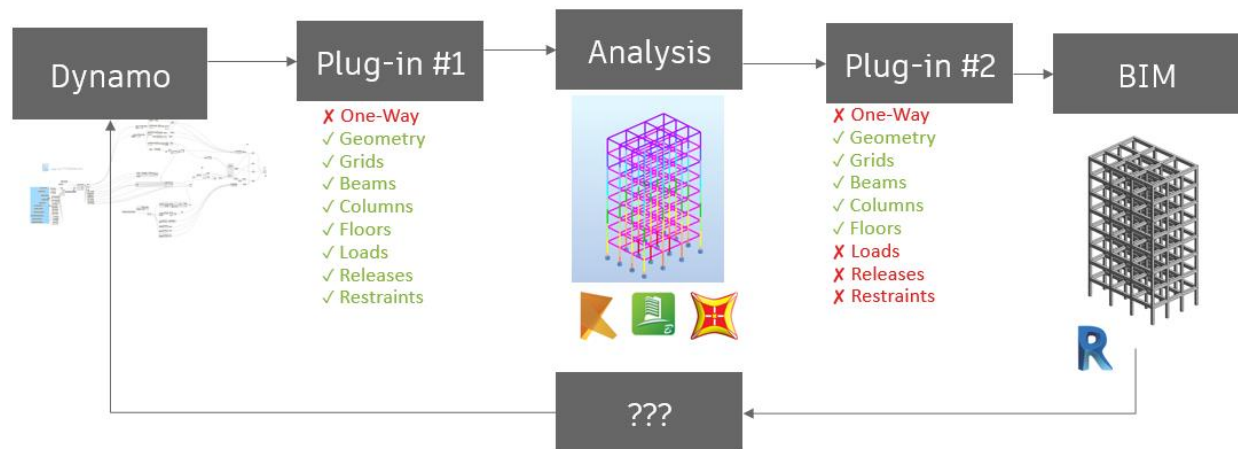
1. **BIM:** (typically Revit) is created with grids, beams, columns and floors.
2. **Plug-in:** This model is then imported to an analysis software using a plug-in. This plug-in can typically import geometry, grids and floors, however, items like loading, releases and restraints are typically assigned with the analysis tool. Most users do not want to spend the time at the beginning of the project adding releases and loads, especially when it can be burdensome to do within BIM.
3. **Analysis:** The engineer then runs an analysis using some type of structural analysis platform. We will discuss these options in more depth in the next section. At this stage, loading, releases and restraints are assigned. While sizes may have been assigned previously, these are refined during this stage.
4. **???:** The challenge occurs on how to update information once the analysis is complete. Typically, this involves manual data entry by selecting elements item by item and updating. Some engineers will delete and recreate their models by using a plug-in. While this can be an effective approach, this has its limitations throughout the project.

Typical Process Challenges

The challenges with the typical process are as follows:

1. Most plug-ins are one-way, meaning it can translate model elements from analysis to BIM or from BIM to analysis. However, most plug-ins fall short with updating elements.
2. There are limited software analysis options available, as not every plug-in is available for every software platform.
3. While many software platforms have plug-ins, not all plug-ins have the capabilities necessary for structural engineers.
4. A common challenge between all tools is the question “How to track changes?”

“Upgraded Process” for Structural Engineering



“Upgraded” process for Structural Engineering

With the advent of parametric design, an “Upgraded Process” has emerged within Structural Engineering. It can be outlined as follows:

1. **Dynamo:** Released in 2016, Dynamo has been a powerful tool for Revit users. One of the benefits is to generate analysis and/or BIM in a parametric environment. There are many examples of generating analysis models using Dynamo on the AU website.
2. **Plug-in #1:** Once the model has been generated within Dynamo, there are several options of using a Plug-in, in this case an Add-on within Dynamo that allows the user to generate an analysis model. Most of the information required, beams, columns, releases, etc. can be generated automatically using a plug-in. Typically, most plug-ins only support a one-way action.
3. **Analysis:** As before, the engineer typically performs structural analysis using the platform of choice. Beams, columns and floor are sized / designed.
4. **Plug-in #2:** The next step becomes translating the model to BIM. This usually involves a different plug-in (Plug-in #2) that may not have all the capabilities of translating from analysis to BIM.

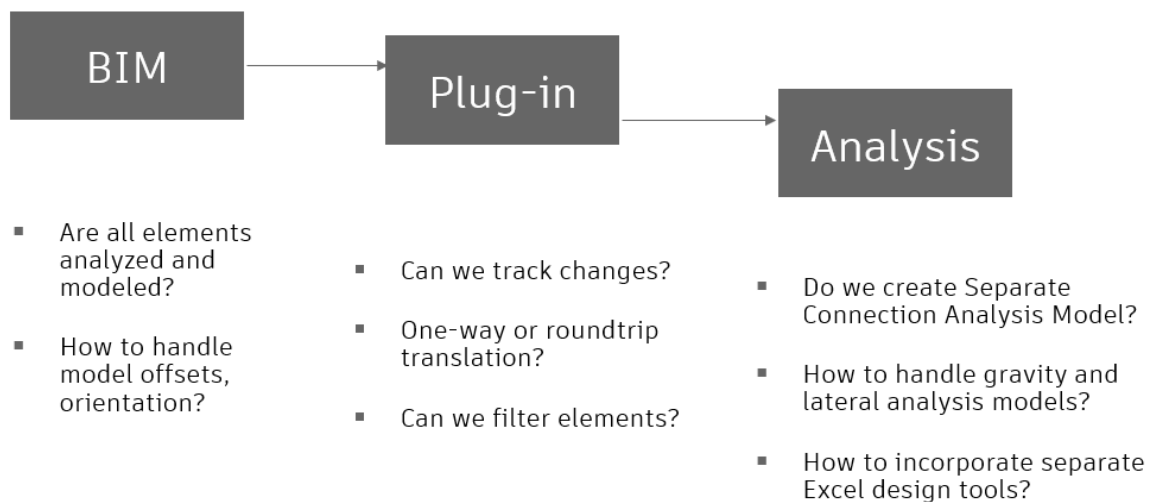
5. **BIM:** By the time the model has been created (i.e., in Revit), items have changed. This leads us to the last phase.
6. **???:** As before, the question remains, “What next?” After the BIM has been created, items must be updated. This typically involves manual entry of every single element.

“Upgraded Process” Challenges

The challenges with the upgraded process include:

1. Multiple plug-ins are typically required, which requires more programs, and can create confusion around project teams.
2. Not all tools have a plug-in.
3. Not all plug-ins are created equal as the capabilities of one plug-in may be more robust than the other.
4. For most users, it is difficult to start and end the process with Dynamo. As project becomes more complex and have multiple pieces of data to manipulate, it can be difficult to manage this parametrically using scripting.
5. In the end, this becomes a manual process for most project teams.

Process Challenges for Structural Engineering



Process Challenges for Structural Engineers

The challenges for structural engineers with BIM, Plug-ins and Analysis are vast. Questions like filtering elements, handling offsets, one-way or two-way translation, and multiple analysis packages add to the confusion for most project teams. While no silver bullet exists, there are some tools that may create better avenues for addressing these questions.

Overview of the current tools available for interoperability between structural analysis and BIM

The current landscape of structural analysis tools changes constantly. Every year, new tools are developed by software vendors, engineers, BIM managers and individuals. We will highlight some of the current structural analysis software, Plug-ins and BIM tools used in the market.

Structural Analysis Software



RAM

[RAM Structural System](#) – is a commonly used structural analysis software used in the US. It is commonly used for steel composite beam projects.



TSD

[Tekla Structural Designer](#) – is a structural analysis software that has strong integration with Tekla and Revit



GSA

[GSA](#) – is a structural analysis software typically used outside the US. It was originally developed by Arup.



Robot

[Robot](#) – is a structural analysis software typically used in Europe. The software has a robust API.



Strand7

[Strand7](#) – is a Finite Element Analysis (FEA) software. The tool is used in several other industries besides AEC.



SCIA

[SCIA](#) – is structural analysis software typically used outside the US.



SAP2000

[SAP2000](#) – is a commonly used structural analysis software across the globe. It is commonly used in Seismic Design and Non-Linear applications.



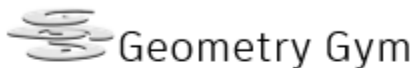
ETABS

[ETABS](#) – is a commonly used structural analysis software across the globe. It is commonly used in Concrete Design and Seismic applications.

Structural Plug-ins



[ISM](#) – is a plug-in for most Bentley products and Revit. It is commonly used as a one-way plug-in from either BIM to analysis or analysis to BIM.



[Geometry Gym](#) – is a plug-in developed by Jon Mirtschin. It has vast functionality when translating Grasshopper models to most structural analysis platforms.



[Konstru](#) – is a plug-in developed by Thornton Tomasetti. It is the only tool that is completely web-based and can translate most analysis tools to common BIM platforms.



[Excel](#) – is a commonly used “intermediary” when translating from BIM to analysis. Some users introduce custom macros to push capabilities even further.

CSiXRevit

[CSiXRevit](#) – is a plug-in for most CSI products and Revit. It is commonly used as a one-way plug-in from either BIM to analysis or analysis to BIM.

Simplex

[Simplex](#) – is a dynamo package developed by Marcello Sgambelluri to read or write from several structural analysis tools.

Rhino.Inside

[Rhino Inside](#) – is a plug-in developed by McNeel to run Grasshopper within Revit. It also includes several other workflows between other platforms.

Tekla Live Link

[Tekla Live Link](#) – is a plug-in developed by Trimble to connect Tekla and Tekla Structural Designer to Grasshopper.

AUTODESK UNIVERSITY

BIM



Revit

[Revit](#) – is a commonly used BIM platform around the globe.



Tekla

[Tekla](#) – is a structural BIM platform typically used for fabrication.



Dynamo

[Dynamo](#) – is a parametric design tool used to connect to Revit and has several packages to connect to other software.



Grasshopper

[Grasshopper](#) – is a parametric design tool used to connect to Rhino and has other plug-ins to connect to other software.



Rhino

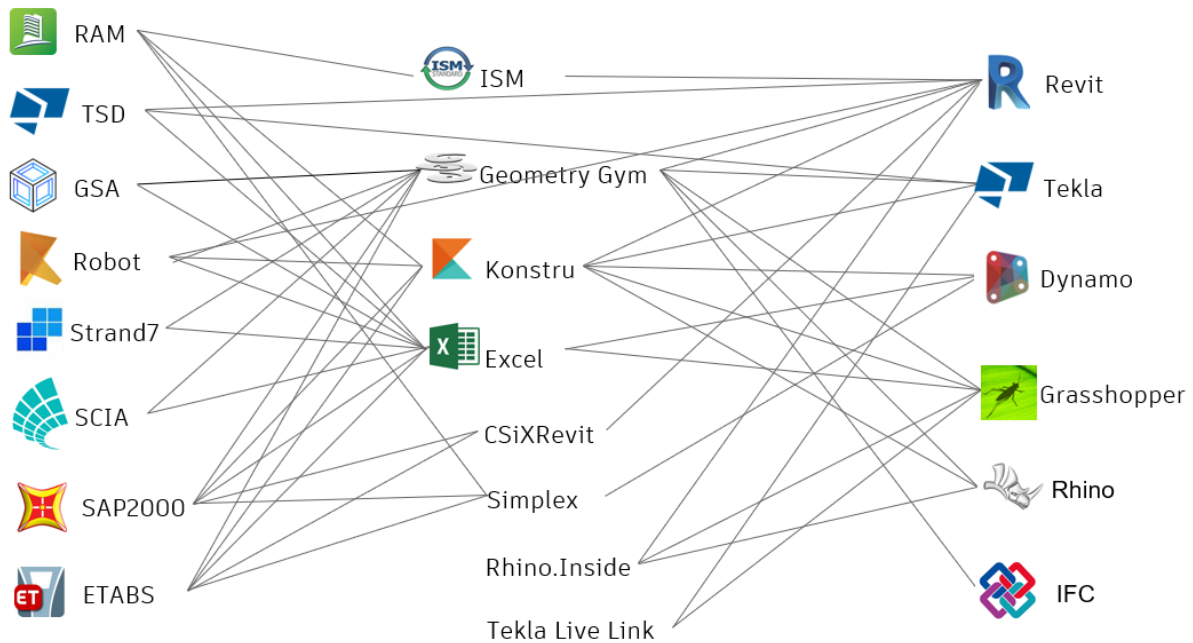
[Rhino](#) – is a 3D CAD tool used in a variety of industries.



IFC

[IFC](#) – is an open-file format BIM standard used in AEC.

The “Web” of Analysis to BIM



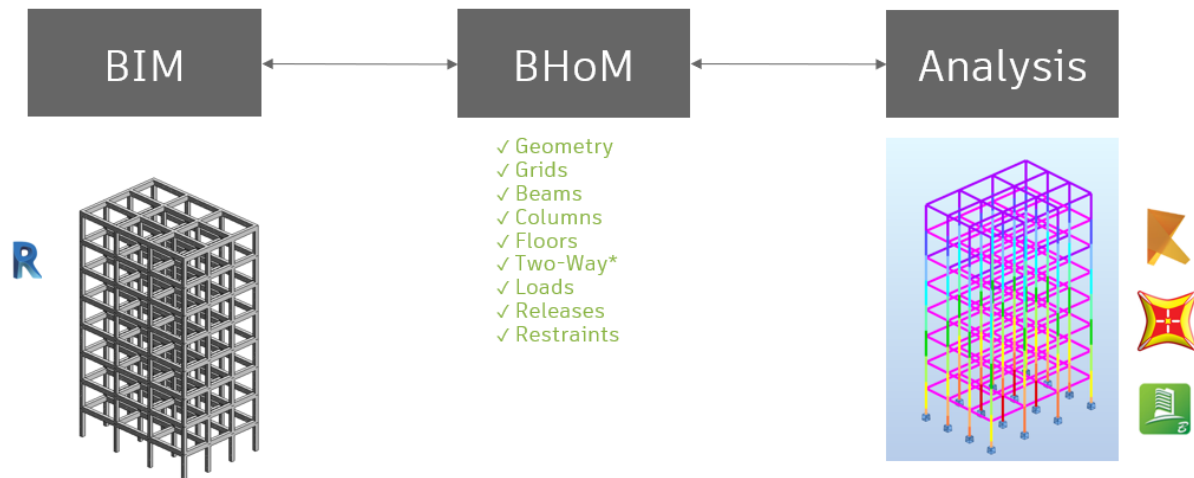
Moving from analysis to BIM can be a challenge for many project teams. While there is no “silver bullet”, certain analysis tools have more advanced plug-ins than others. As projects become more complex and project teams are forced to be more efficient, the correct use of plug-ins will become more critical for the successful outcome on projects.

What is BHoM?

Buildings and Habitats Object Model (BHoM) is an open-source environment that allows users to collaborate between multiple software platforms. Developed by Buro Happold in 2016, this tool was released in 2018 as an open-source project. The tool is hosted on GitHub for users to comment, test, and collaborate regardless of location or company. BHoM allows users to translate data from many different software packages by breaking down the barriers to create a single source of truth. The tool is primarily built using the C# programming language, and the intent is for engineers and designers to allow them to provide, “more design, less code”.

BHoM Process

Using BHoM, the process of moving from BIM to Analysis can be more streamlined.



Note there is an (*) next to Two-Way. While this is the reality on some of the software packages within BHoM, this is a task currently in progress. Each software analysis package is at different stages. Below are some links on BHoM Github Wiki showing the current status of different structural packages:

[BHoM Robot object relations · BHoM/Robot Toolkit Wiki \(github.com\)](https://github.com/BHoM/BHoM/wiki/BHoM-Robot-object-relations)

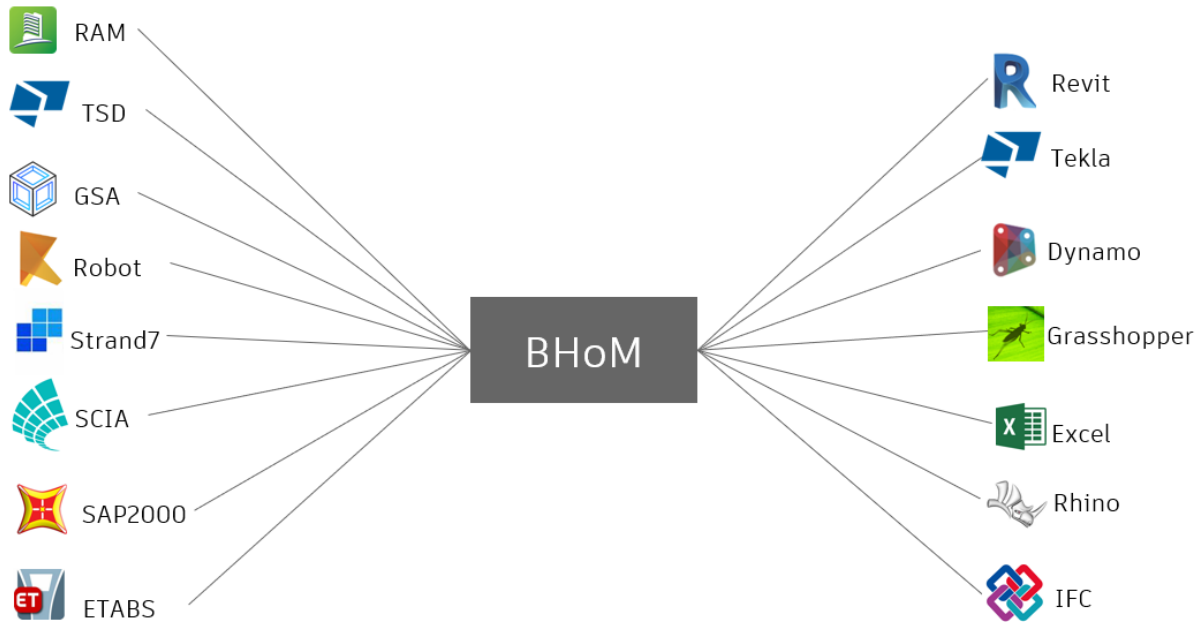
[BHoM SAP2000 Object Relations · BHoM/SAP2000 Toolkit Wiki \(github.com\)](https://github.com/BHoM/BHoM/wiki/BHoM-SAP2000-Object-Relations)

[BHoM ETABS Object Relations · BHoM/ETABS Toolkit Wiki \(github.com\)](https://github.com/BHoM/BHoM/wiki/BHoM-ETABS-Object-Relations)

[BHoM RAM Object Relations · BHoM/RAM Toolkit Wiki \(github.com\)](https://github.com/BHoM/BHoM/wiki/BHoM-RAM-Object-Relations)

BHoM Interoperability

As of today, the BHoM works with the following tools:



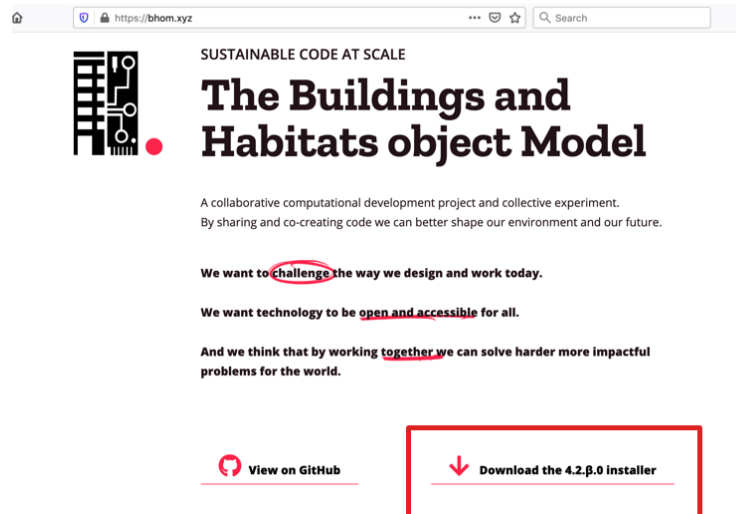
BHoM CURRENT WORKFLOWS

BHoM can be a single source of truth to help project teams streamline their data translation from analysis to BIM and BIM to analysis.

Installing BHoM

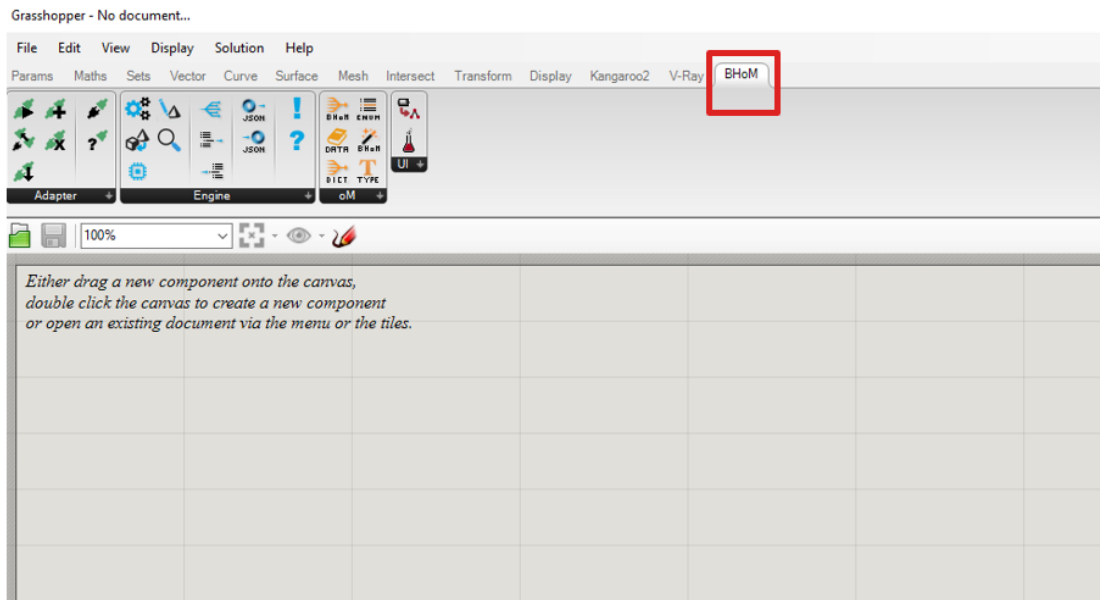
BHoM can be installed at the website – BhoM.XYZ

How do I get BHoM?

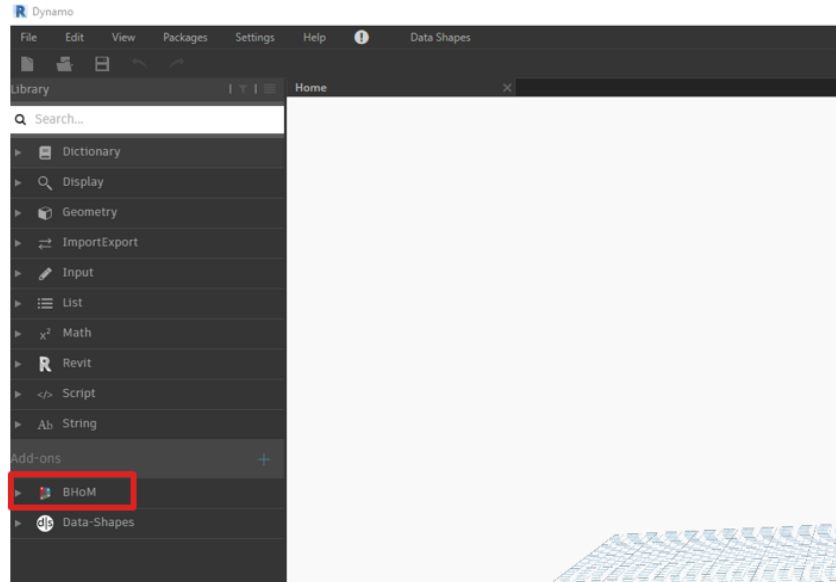


Simple click the install at the bottom right of the screen. The install should happen automatically. It is recommended that Revit, Dynamo, Excel and Grasshopper are closed. Once installed, BHoM should automatically be installed in Grasshopper, Dynamo, and Excel. For more advanced users, BHoM libraries can be accessed using Visual Studio

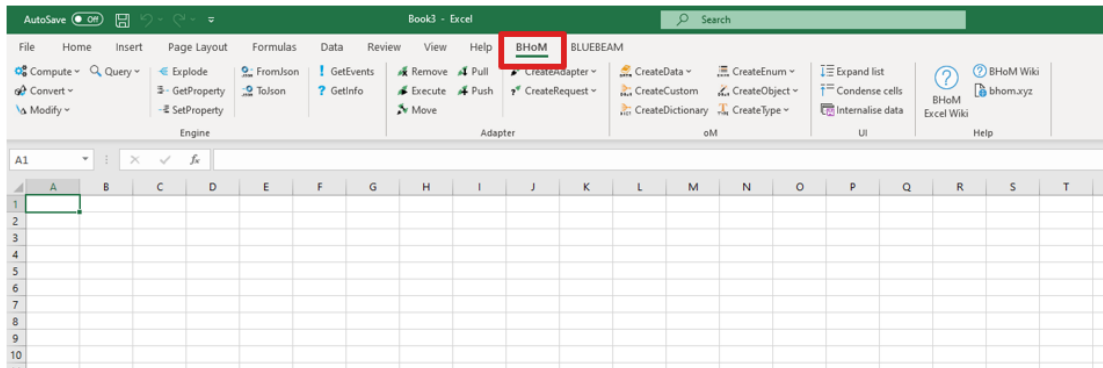
Installs in Grasshopper



Installs in Dynamo

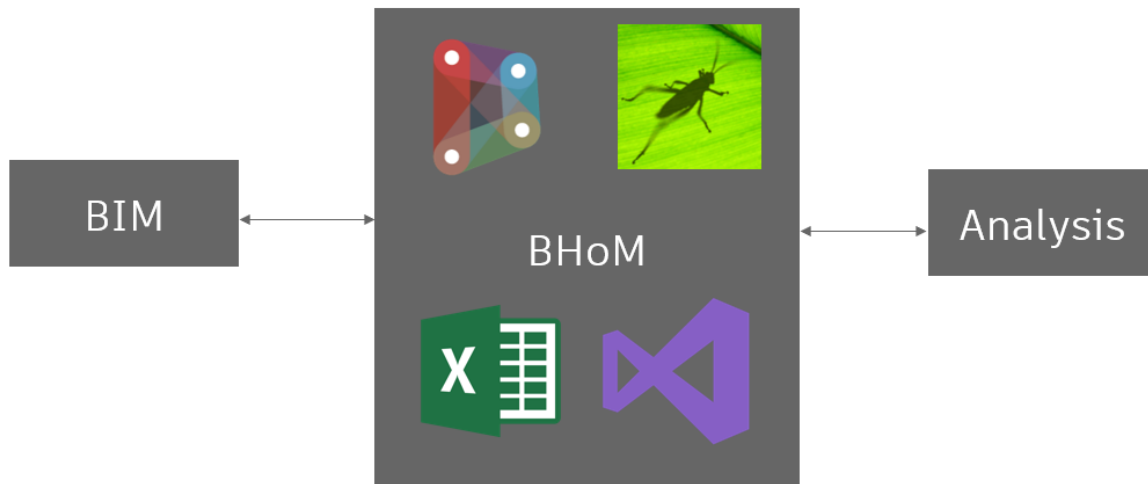


Installs in Excel



BHoM Adapters

BHoM is built using adapters. You can think of adapters as the connection point between BHoM and other software. A unique feature of BHoM is that it is not a stand-alone software, meaning it cannot be open and run like Revit. Instead, BHoM uses Adapters – such as Dynamo, Grasshopper, Excel or Visual Studio to connect with other software.



BHoM Toolbar

Within BHoM, there are 3 major toolbars used: Adapter, Engine and Object Model (oM). We will take a quick look at each.

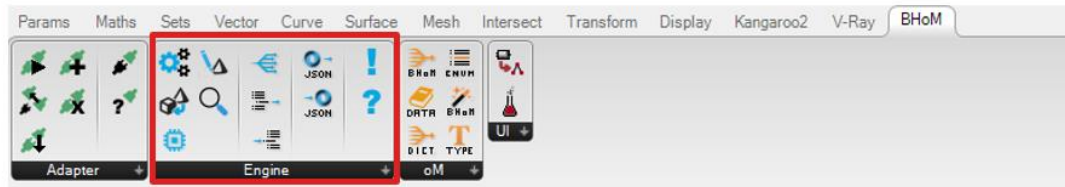
BHoM Adapters



The **Adapter Panel** allows you to:

- Connect to other software
- Push Data / Pull Data to other software (i.e., push beams to Robot, read columns from Revit)
- Execute commands (i.e., run analysis)

BHoM Engine



The **Engine Panel** allows you to:

- Modify a BHoM Object (i.e., flip bar orientation)
- Get or Set Property (i.e., get a beam section or set a beam section)

BHoM Object Model



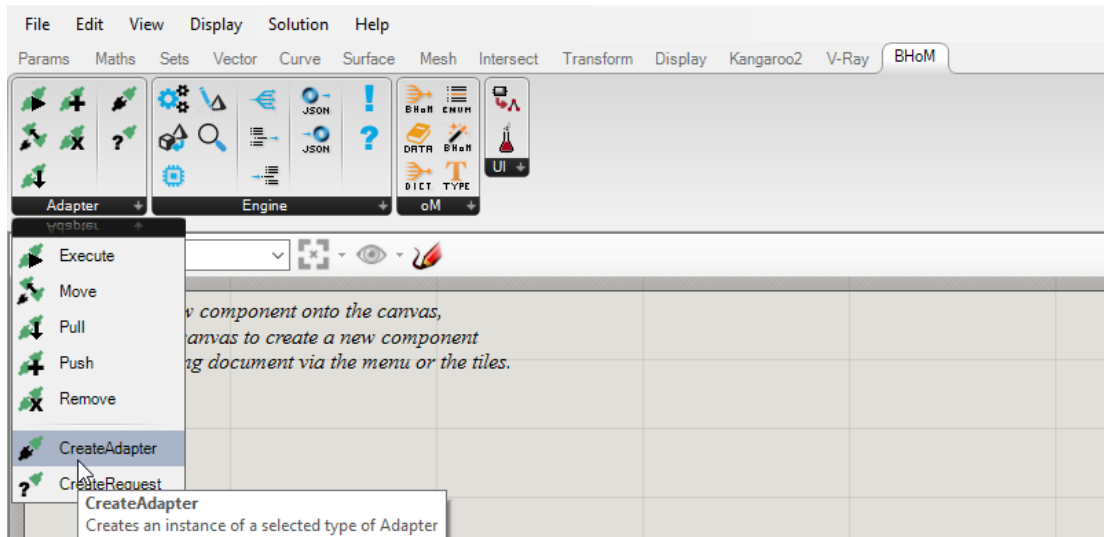
The **oM or Object Model Panel** allows you to:

- Create a BHoM Object (i.e., create a bar, node, panel, load)
- Create BHoM Library (i.e., W-section library)

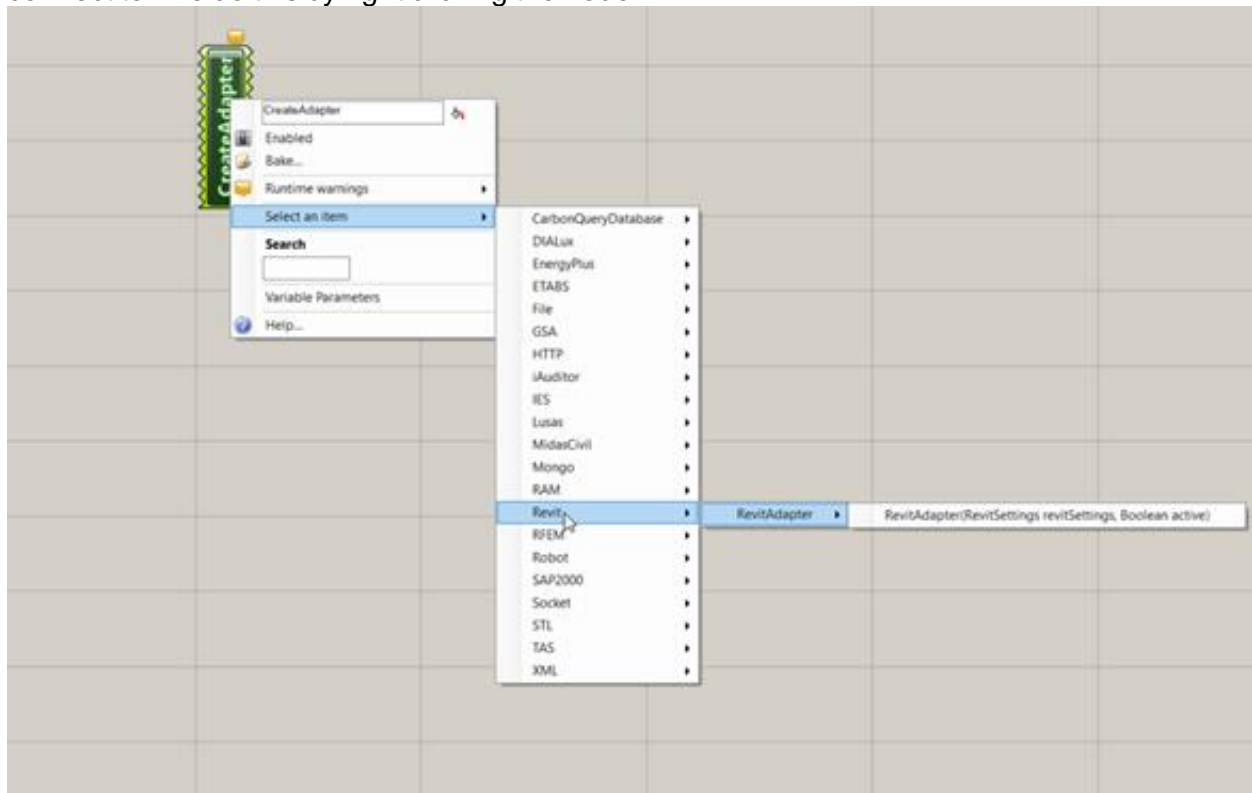
Create a BHoM Adapter

To create a BHoM Adapter, navigate to the BHoM Bar (this can apply to Grasshopper, Dynamo or Excel) – Under the Adapter Tab – Click the Icon for “Create Adapter”

Grasshopper - No document...



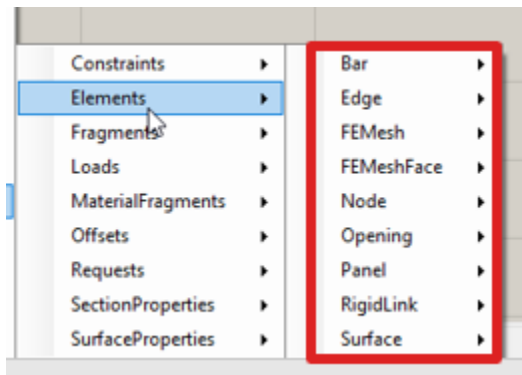
Once the adapter is on the canvas, we can now create an Adapter for the software we desire to connect to. We do this by right clicking the node.



Helpful Hint: Many BHoM nodes will need to be “right clicked” to assign correctly. Instead of creating individual Grasshopper or Dynamo components for every single software, they are grouped in ways to reduce the clutter for the user. Keep this in mind as you explore BHoM.

BHoM Analytical Library

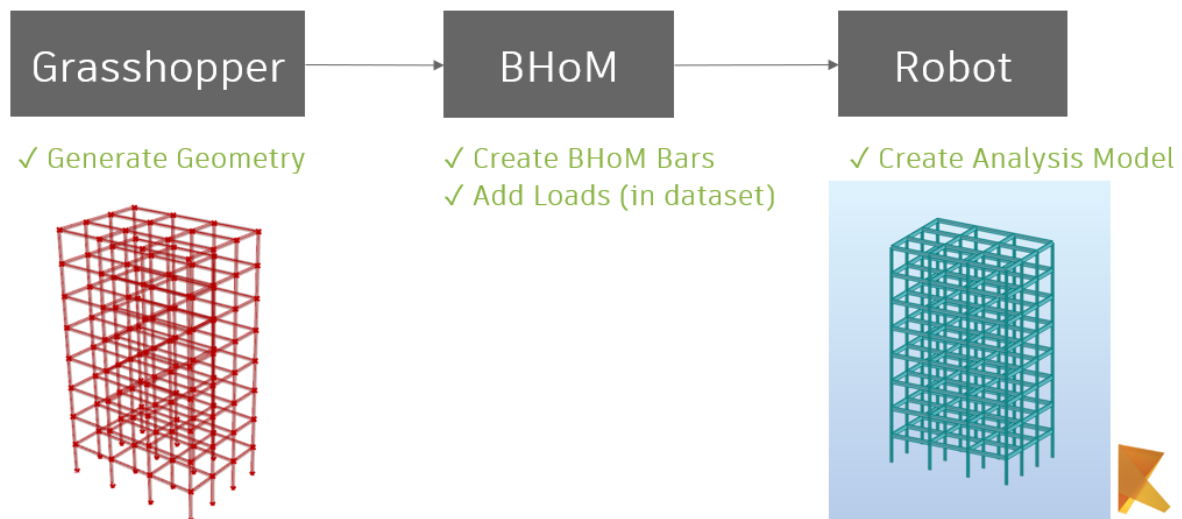
Within BHoM, there are connections to many different structural packages. However, the different structural packages have different nomenclature for their different elements. For example, an analytical bar in Robot is a Bar, in SAP2000 it is a frame, and in RAM it is a beam (or column). The BHoM developers attempted to streamline this process by simplifying the “noise” between packages. Below is an example on the nomenclature of many structural elements.



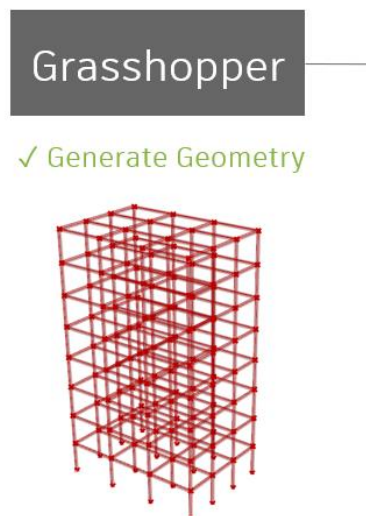
Using the BHoM: Grasshopper to Revit

In the first example, we will be creating a Grasshopper model with geometric components (a simple tower), we will then use that geometry to create BHoM analytical bars and assign sizes and loads, and finally we will generate a Robot analysis model.

BHoM – Create Robot from Grasshopper



Step 1: Generate Geometry

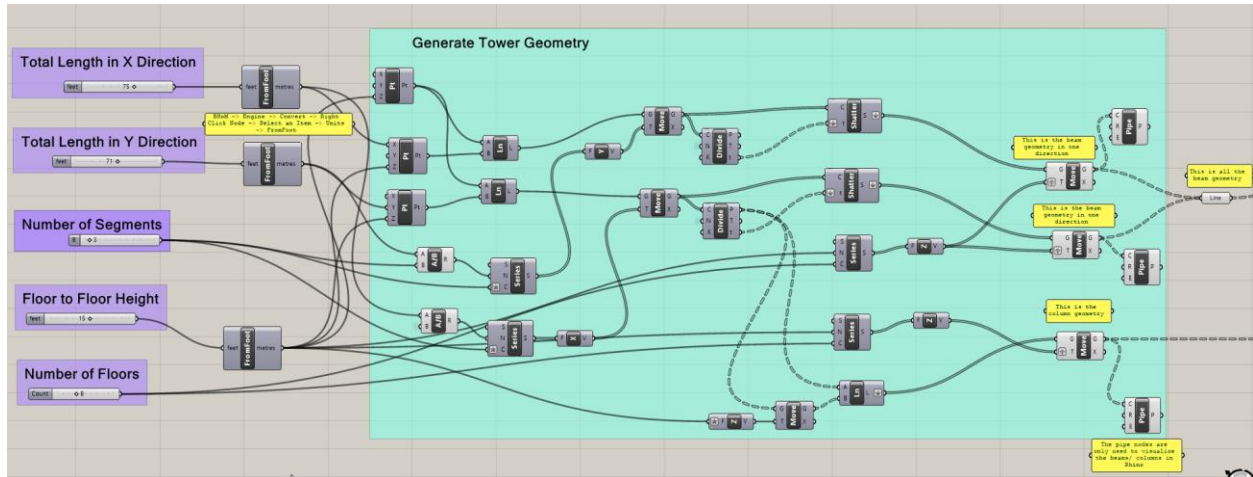


Steps to Open Script:

1. Open Rhino (The example was built on Rhino 6, but Rhino 7 should work just fine)
2. Type in Grasshopper in the command line.

3. Grasshopper will open – you should see a blank canvas.
4. In the dataset, open the file “Example1-CreateRobotFromGrasshopper”.
5. As you make adjustments, the script will automatically update.

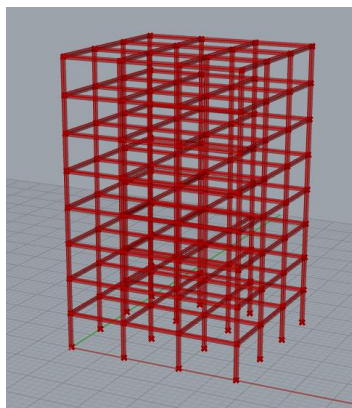
Our first step is to generate geometry within Grasshopper. Creating this tower script is outside of the scope of the class, however, I would encourage anyone wanting to learn more about Grasshopper to try and recreate this script from scratch.



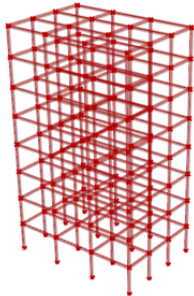
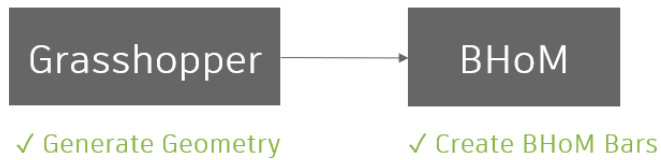
Helpful Hint:

BHoM units are SI by default. If we want to use Imperial units, we need to use the node “FromFoot” which will automatically change the units from feet to metric, in this case meters. Keep in mind units – especially with loads.

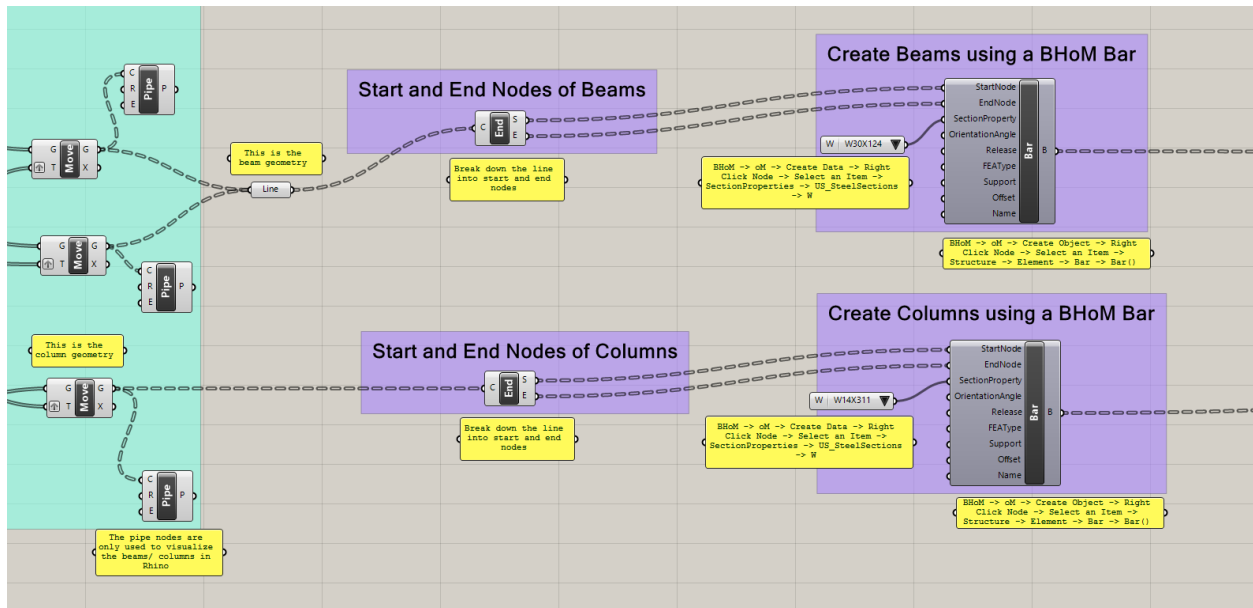
This script may seem intimidating to those just starting in Grasshopper (or a messy script to Grasshopper purists), but it is a good starting point for those wanting to learn more on creating lines, series of points, moving geometry, etc. For the purpose of this example, no further changes are required outside of adjusting the sliders to the building size and stories you desire. Note that the “pipe” nodes are created at the end of the script. These are only placed to create a simple visual element to the Rhino window. Within Rhino, you should see the following:



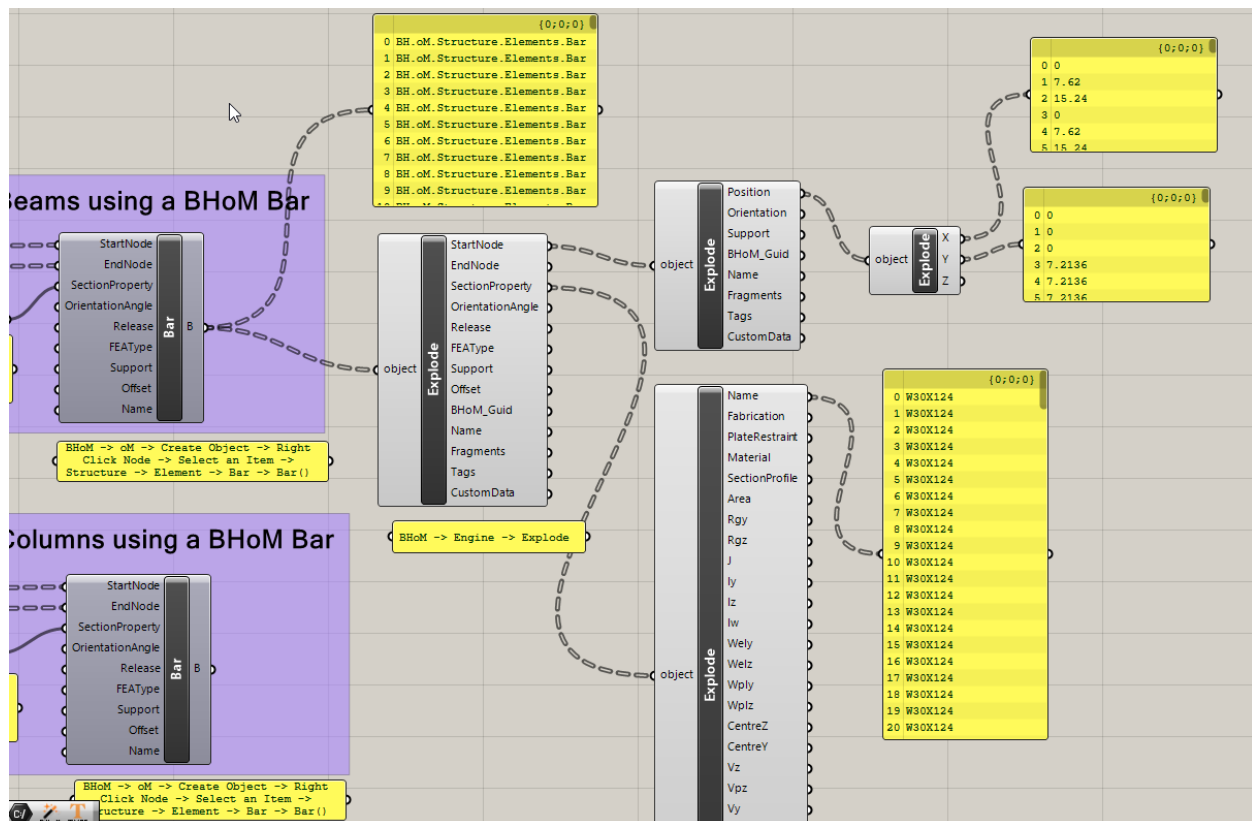
Step 2: Create BHoM Bars



Our next step is to create BHoM Bars using the geometry we created within Grasshopper. Recall that we will do this within Grasshopper, as BHoM is as a plug-in within the Grasshopper environment. Note the sequence of getting a BHoM node. Typically, you select a node from the toolbar then right click on the node to define it. Another option is to hold “Control + Shift + B” and start typing the node name. This will only search BHoM nodes and is very helpful.

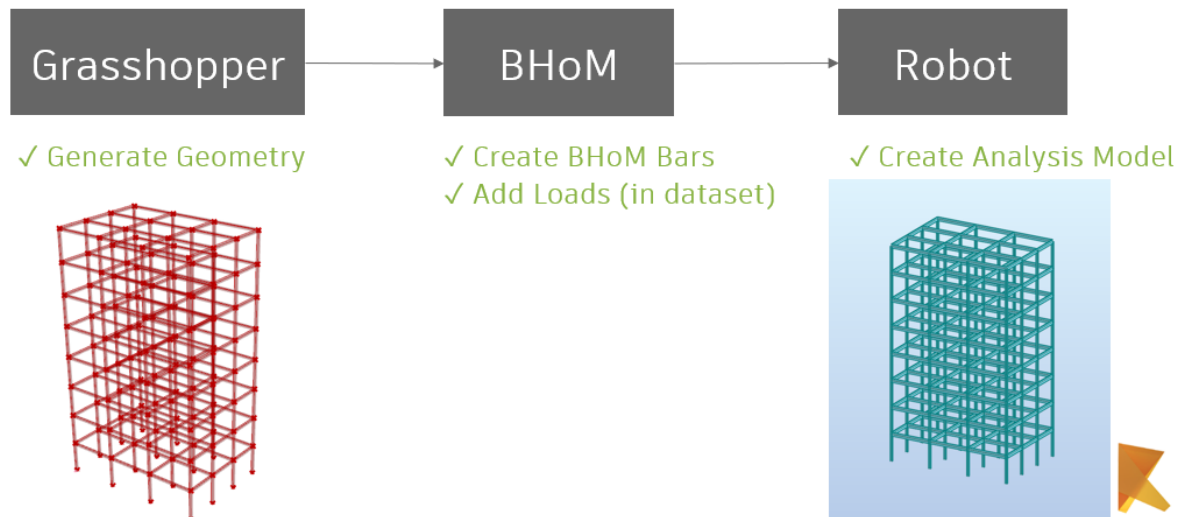


Now, we have created our BHoM Bars. We will send them to analysis later, but at this point, we have created BHoM Bars and assigned the beams and columns with wide flange steel sizes. A good check is to use the explode node. We will use the explode node several times to “explode” the data to another level (think boxes within boxes). It’s good to experiment with the explode nodes anytime you create BHoM elements to verify the objects were created correctly.

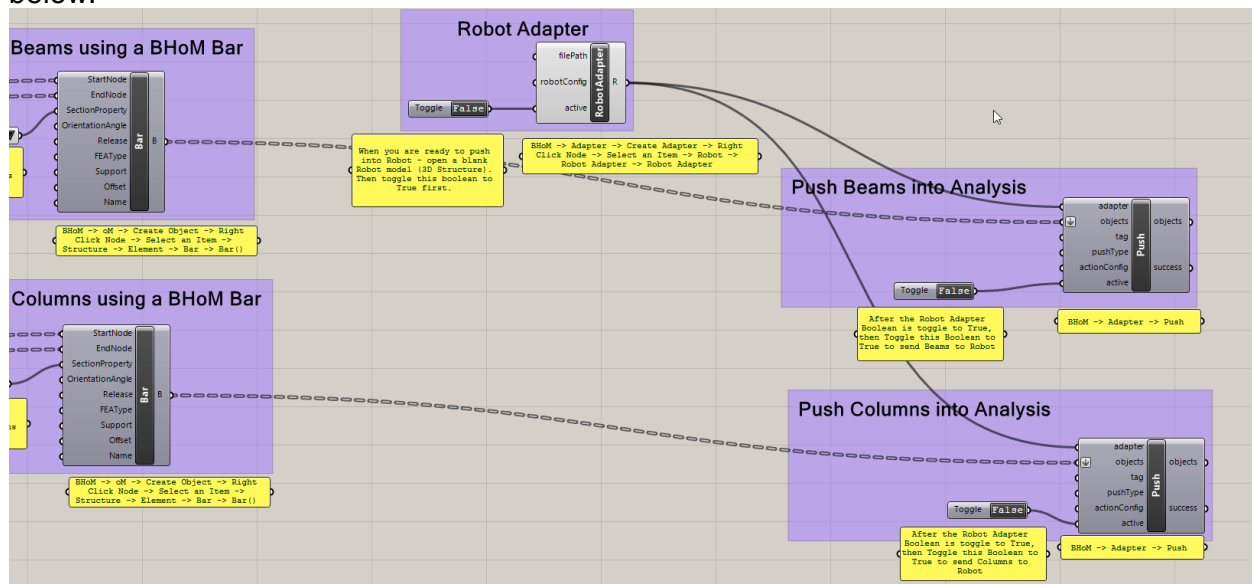


Step 3: Create Analysis Model

BHoM – Create Robot from Grasshopper

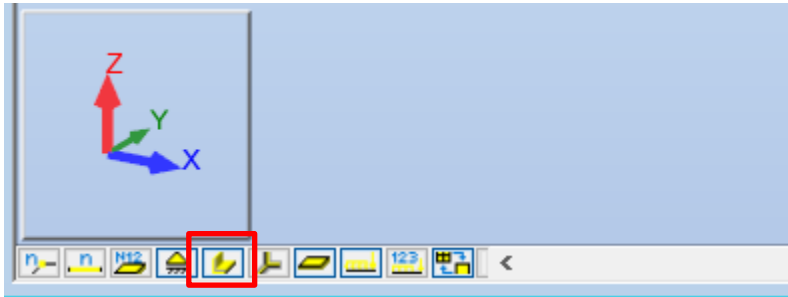


Our next step is to create a Robot analysis model. You might be asking – “What about loads”? – we will first create the Robot analysis model and then assign the loads. Open a blank Robot model (most versions of Robot will work) and create a blank 3D Structure Model. See the nodes below.

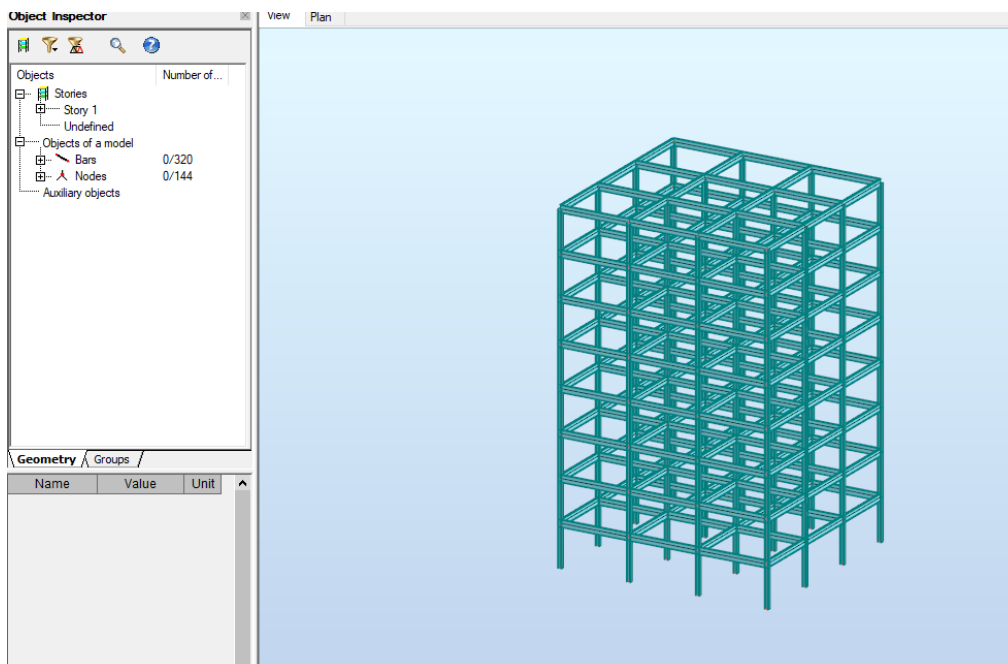


When we are ready, we start double clicking the Boolean toggle at the Robot Adapter (to set it to “True”). Next, we double click the toggles at the “Push Beam” and “Push Columns” to send it

to Robot. Once all the toggles turn to “True”, open Robot and go to a 3D View. In Robot, click the “Section Shapes” Icon at the bottom left of the screen to turn on the Rendering of the members.



Your 3D View in Robot should look similar to below. Great! We now have created our analysis in Robot from Grasshopper. You can right click on any member in Robot to see that the steel wide flanges sizes were assigned correctly.

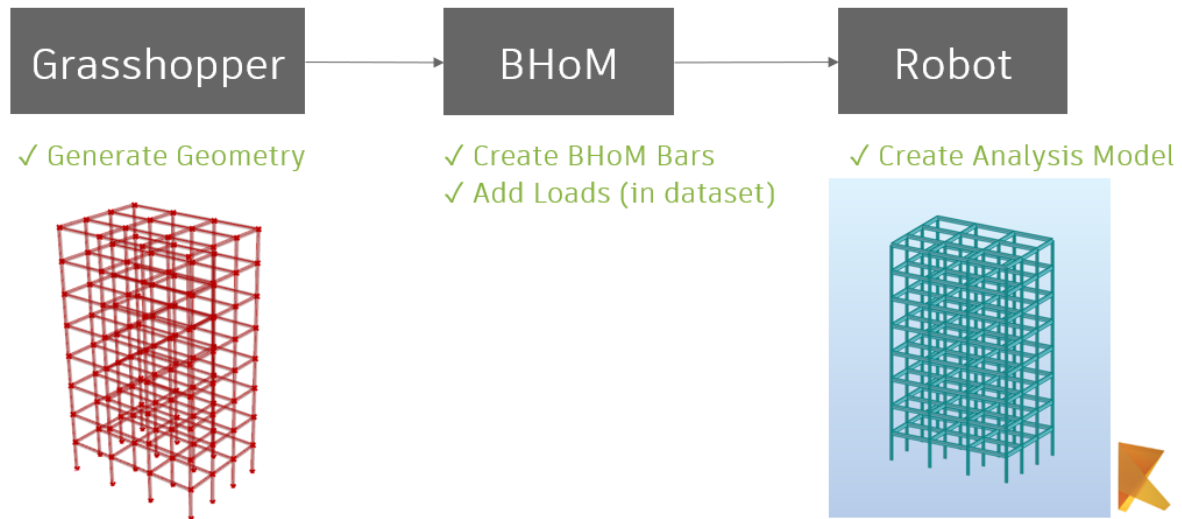


Our next step is to create loads and send them to Robot using BHoM.

Helpful Hint: If we wanted to send our Grasshopper Geometry to a different structural analysis package, we would simply change the adapter from a Robot Adapter to a SAP2000, ETABS, or RAM Adapter. There are several other options within BHoM. Check out the BHoM Wiki!

Step 4: Create Loads in Robot from Grasshopper

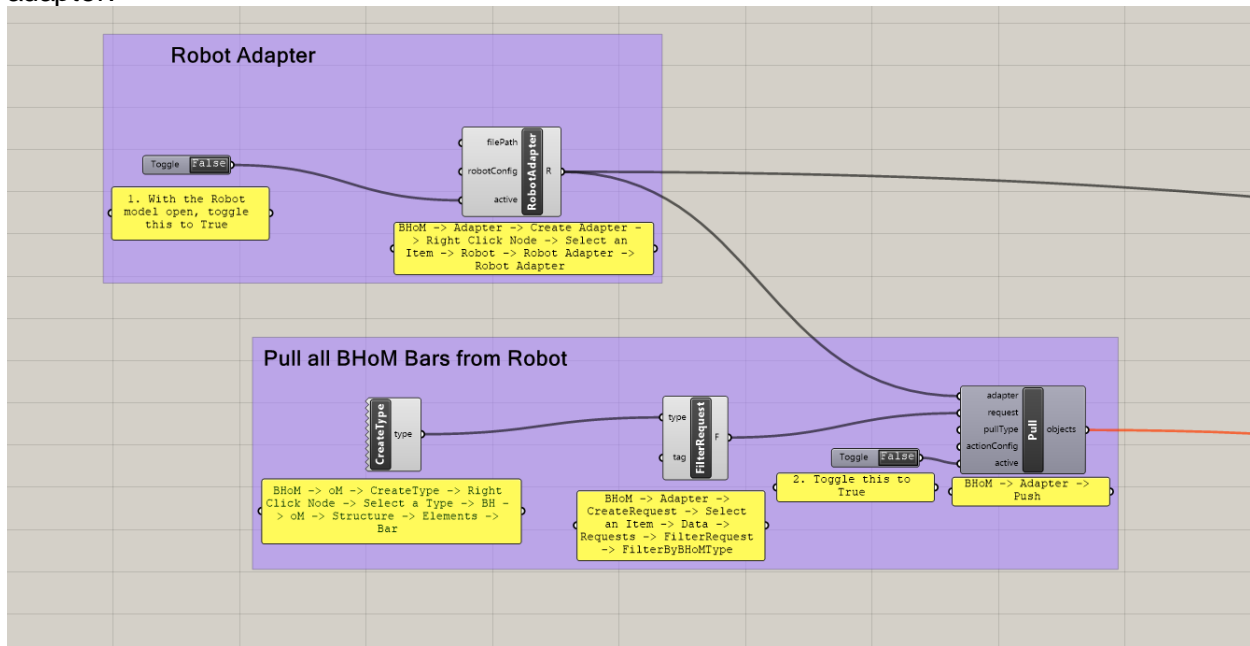
BHoM – Create Robot from Grasshopper



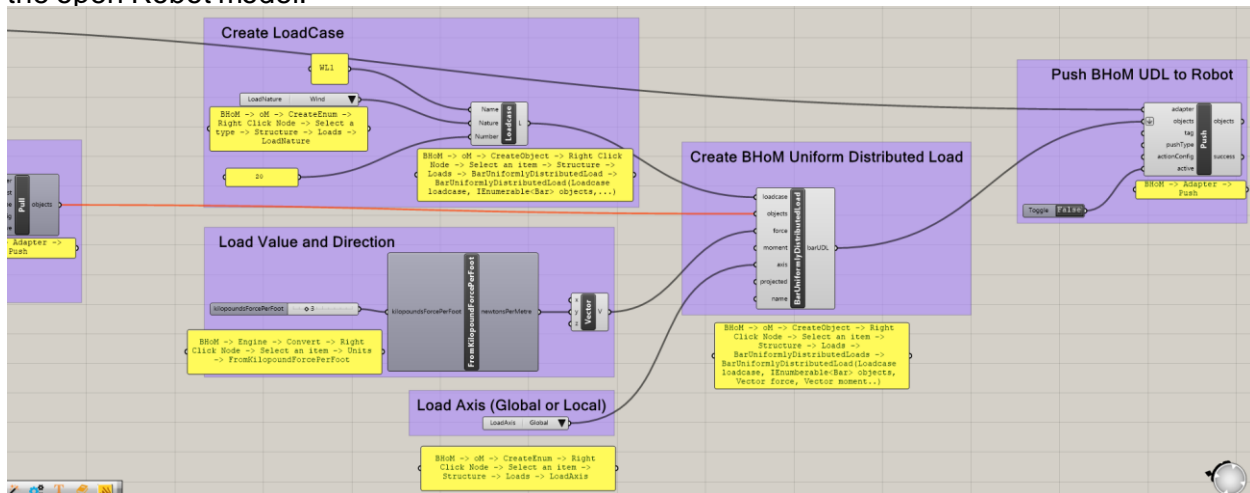
Our last step in this example is to add loads to the Robot model. This will require a few new concepts. The first is “Pull” – this allows us to grab all the data from the model. The second is “Filter Request” – one which we create a request and filter by a certain object type.

Let’s get started, with the Robot model open (that we just created), we will now add loads. If you were having trouble with the model, go ahead and open the Robot model in the dataset- “Example1-BeamsandColumns-NoLoads”.

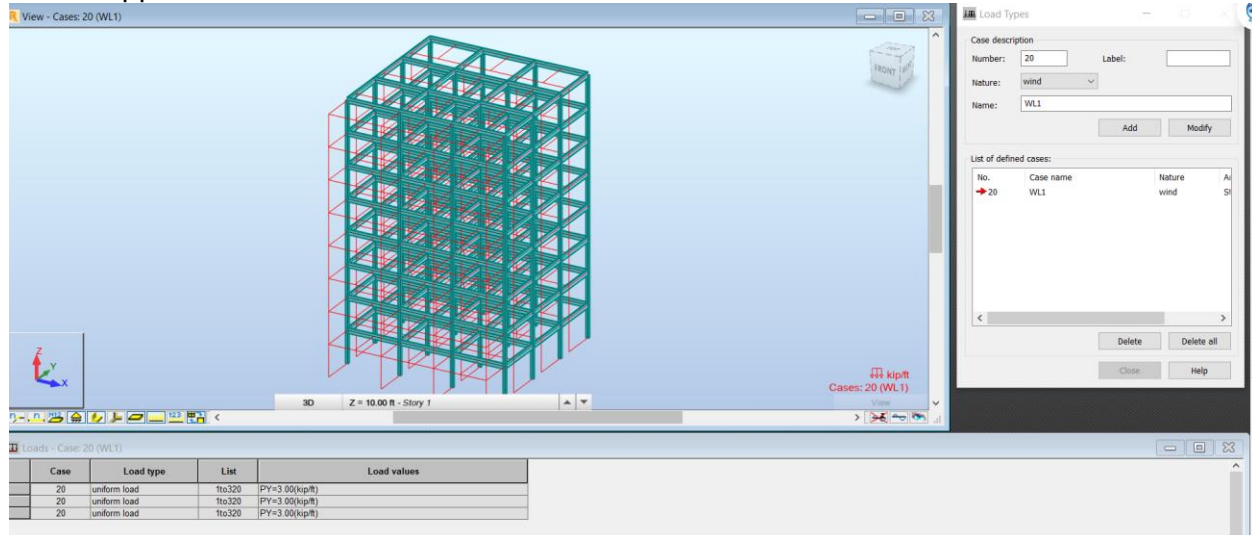
The first part of this script will pull all the data from the model. This is done using the “Pull” node. This will pull all the data from the model (in this case Robot) which could include elements, loads, restraints, panels, etc. The pull node requires a “request” – which is a request that BHoM requires to initiate a pull. We will filter by BHoMType – and we will use a Bar type. This pull node will then pull all the analytical bars in the Robot model. When ready to run, first double click the Boolean toggle at the Robot Adapter. Then double click the Boolean toggle at the Pull adapter.



The next step is to create a uniform load object. This requires a load case, load value and load direction. Note the unit conversion for the load value. Once all the values are set to the desired type, double click the Push node at the end of the script. This will push the distributed load into the open Robot model.



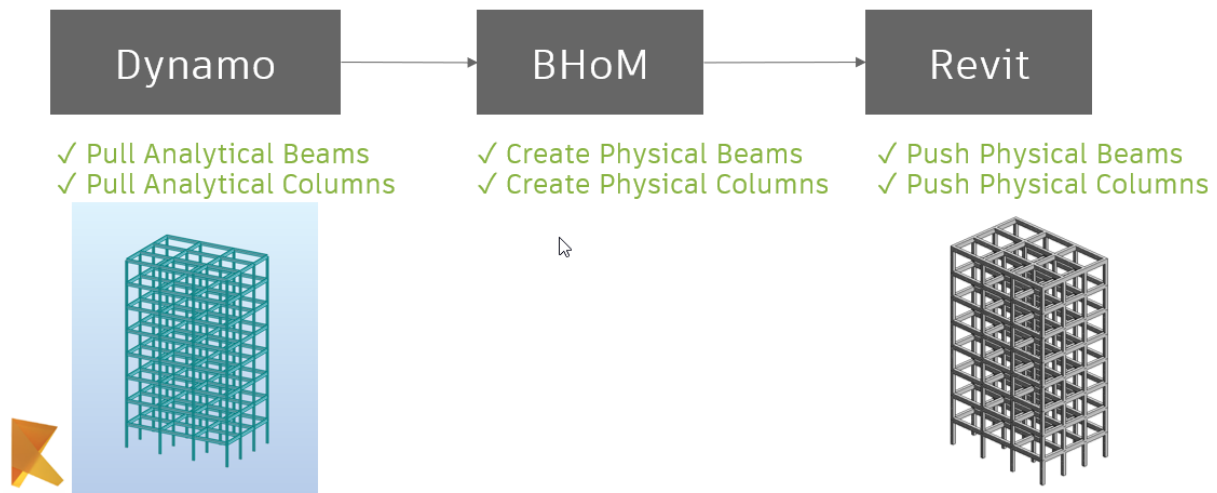
Once complete, you should see distributed loads applied to all bars in the open Robot model. If we wanted to only apply this to beams (or columns), we would need to filter the items in Grasshopper via other means. Your Robot model should look similar to below:



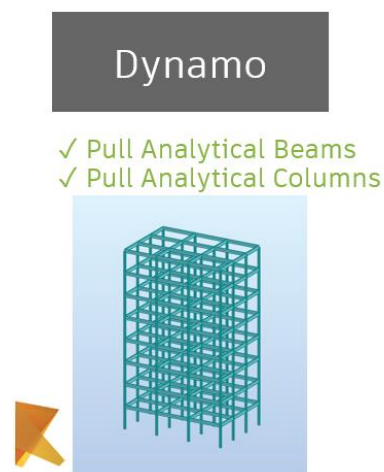
Using the BHoM: Robot to Revit

In this next example, we will pull all the beams and columns from the Robot analysis model we created, then we will use Dynamo to pull all the beams and columns and create BHoM Physical objects, and finally we will use BHoM to push all the physical objects into native Revit families.

BHoM – Create Revit Model from Dynamo

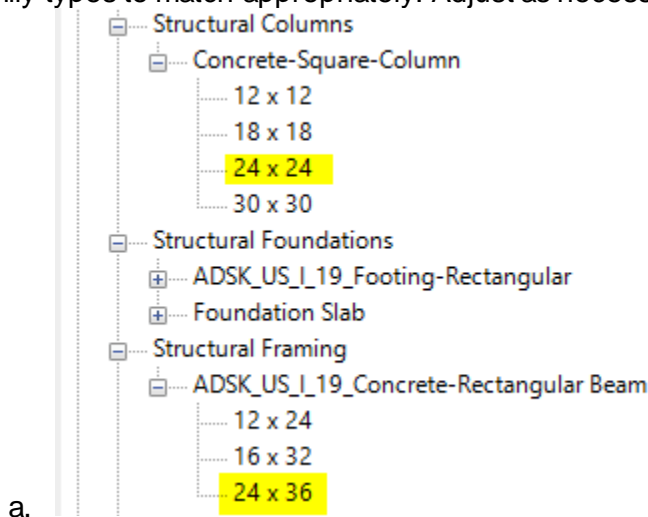


Step 1: Pull all elements from Dynamo



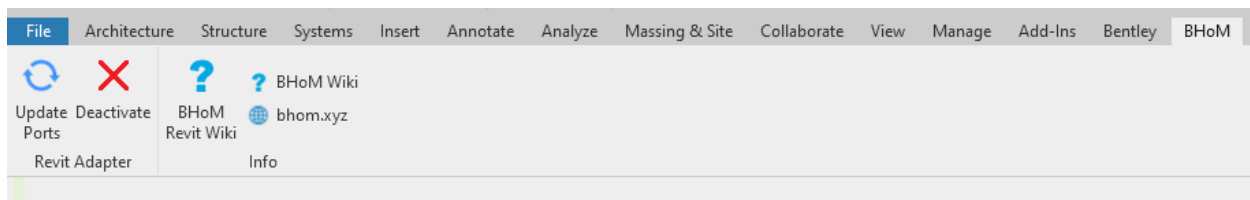
Our first step will be to grab all the information in the Robot from Dynamo. Note that the beam and column types have been changed to concrete in the Robot model.

1. Open Revit 2020, and open the Example2-Revit2020-ImportAnalysis-START.rvt Revit file.
2. Adjust the levels per the settings in your grasshopper model. Verify the levels are in alignment with your desired floor to floor height. (Note: BHoM could be used to create Revit levels, see the BHoM Github – Revit Toolkit – Examples)
3. In Revit – click manage tab – Dynamo – Open the “Example2-Dynamo-ImportRobotToRevit.dyn” file. Go ahead and minimize the Dynamo screen.
4. In this example, we have set up two Robot section types, “24x36-BEAM” and “24x24-COL” to represent concrete columns and beams. We have used the OOTB Revit families to create the following Revit families. Note this is required so that the Revit family types to match appropriately. Adjust as necessary.



- a.
5. Now, we need to turn on the BHoM

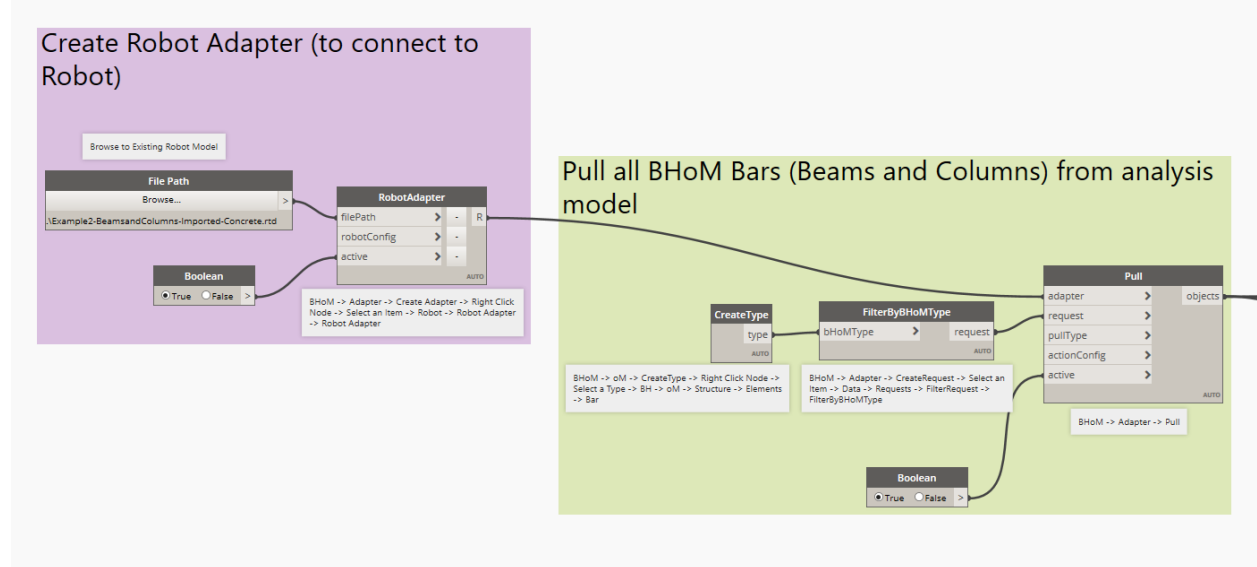
Next, we need to turn ON the BHoM Adapter. In Revit, Click the BHoM Tab, select “Activate” at the Revit Adapter. Your BHoM Tab should now look like this:



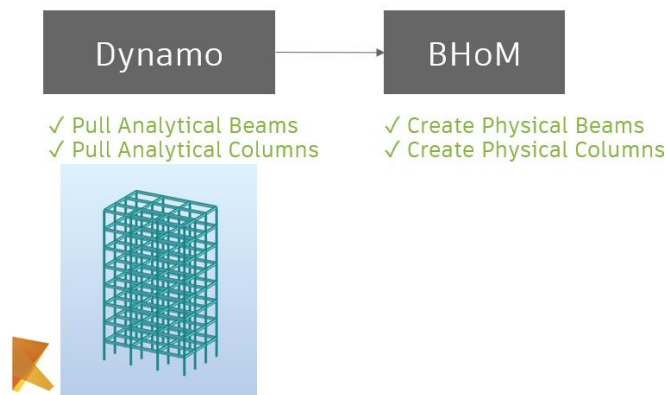
This will “Activate” the Revit Adapter so that Revit is now ready to receive Data from any source.

Let's move back to Dynamo. We will first reference the Robot model which has the concrete beams and columns. I recommend setting the Dynamo Script to run in Manual. Only select Run when ready to send the model information into Revit.

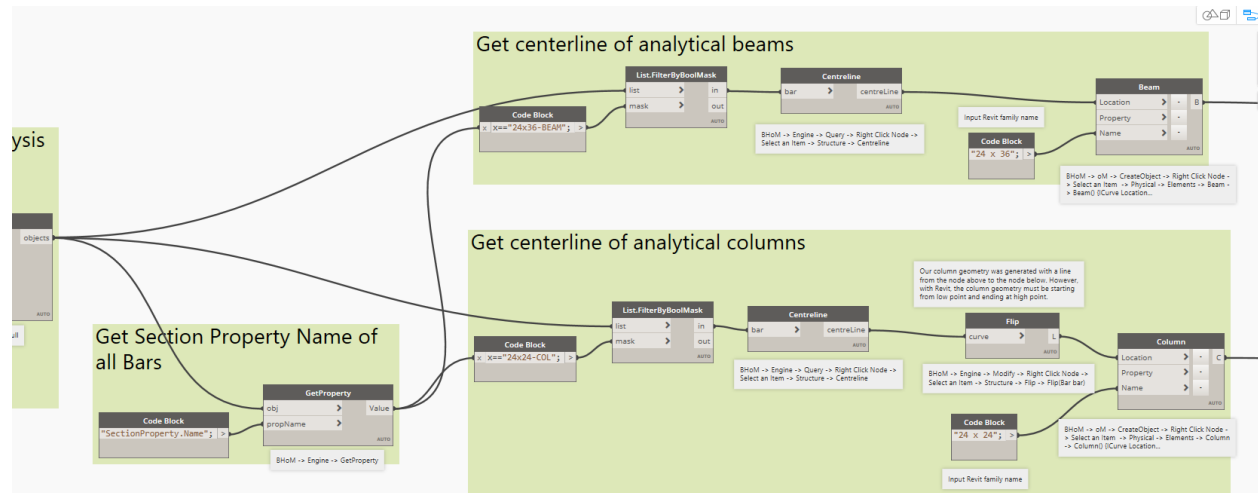
As before, we are creating a Robot Adapter. We will then Pull data from the Robot model using the Pull component. In this case, we need filter by BHoMType using Bar Types. This portion of the script pulls all of the BHoM Bars from the analysis model.



Step 2: Create Physical Beams and Column BHoM Objects

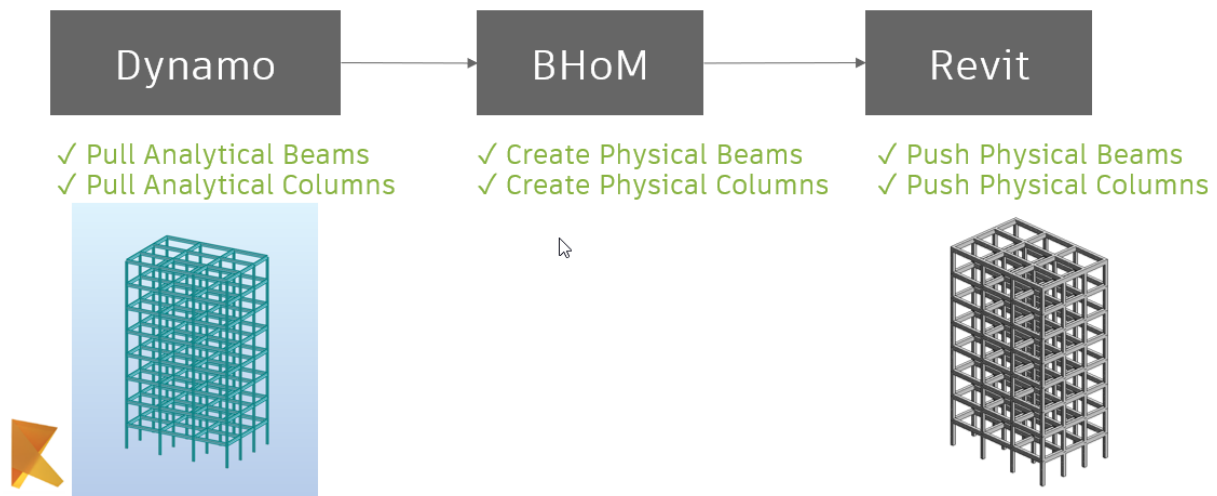


We now need to create BHoM Physical Objects from the analytical bar underlying geometry. In order to do this, we will filter BHoM Objects by their Section Property Name. We will first use a GetProperty component to get the name of every beam and column in the Robot analysis model, in this case "24x36-BEAM" and "24x24-COL". We will then use a filter by Boolean mask node to split the list into two. We will then use a centerline BHoM component to get the underlying geometry of the analytical bars. For the columns, we must flip the bar orientation as Revit Structural Columns must have the start point at the lower elevation.

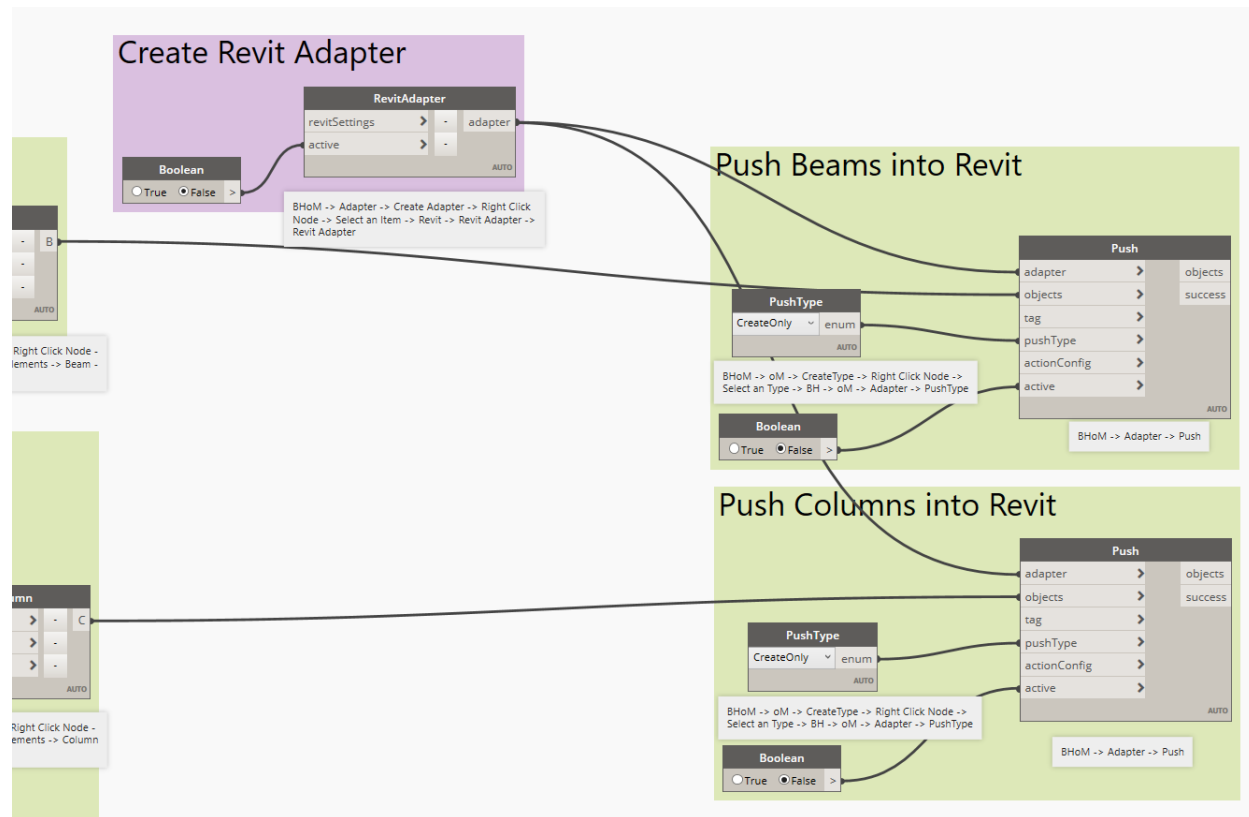


Last, we will create physical beams and physical columns. Note these BHoM objects are not analytical bars, but Physical elements. This is required to create Structural Beams and Structural Columns in Revit. We are inputting the Revit family name which we created in the Revit project at the beginning of the script.

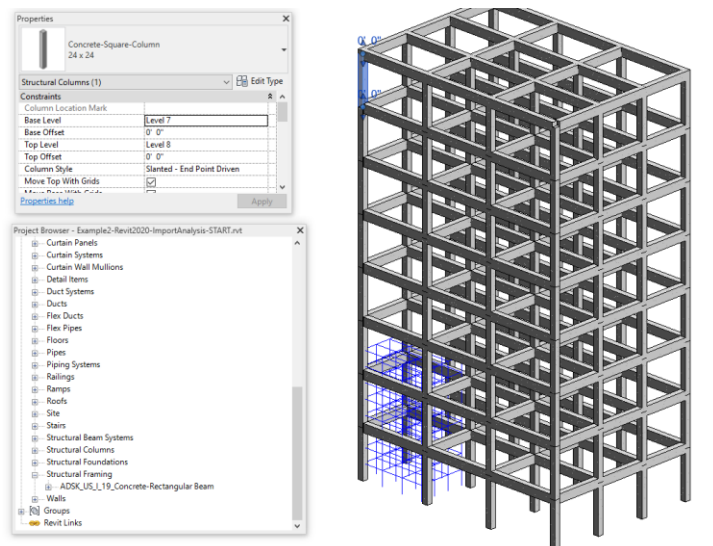
Step 3: Push Physical Beams and Columns into Revit



Our next step involves creating a Revit adapter and pushing these physical elements into Revit. Create a Revit Adapter as shown and then create Push nodes. In order for the elements to be pushed into Revit, it requires a "PushType". A PushType has 4 Options – Create, Read, Update and Delete – which are the four basic functions that models typically need to function. At the time of this writing, BHoM only supports "CreateOnly", "UpdateOnly" and "DeleteThenCreate". More can be found [here](#).



Now we are ready to run the script. Change all of the Boolean toggles to True and then run the script. Once complete, you should see this in Revit:



All done! If you have errors or need to run the script again, I'd recommend the following:

1. Set all Boolean toggles to false.
2. Run the script
3. Set all Boolean toggles to true.
4. Run the script

We have successfully referenced an existing Robot model with concrete beams and columns and used BHoM and Dynamo to create native Revit Structural Beams and Columns into an open Revit model.

Getting Started with LCA and BHoM

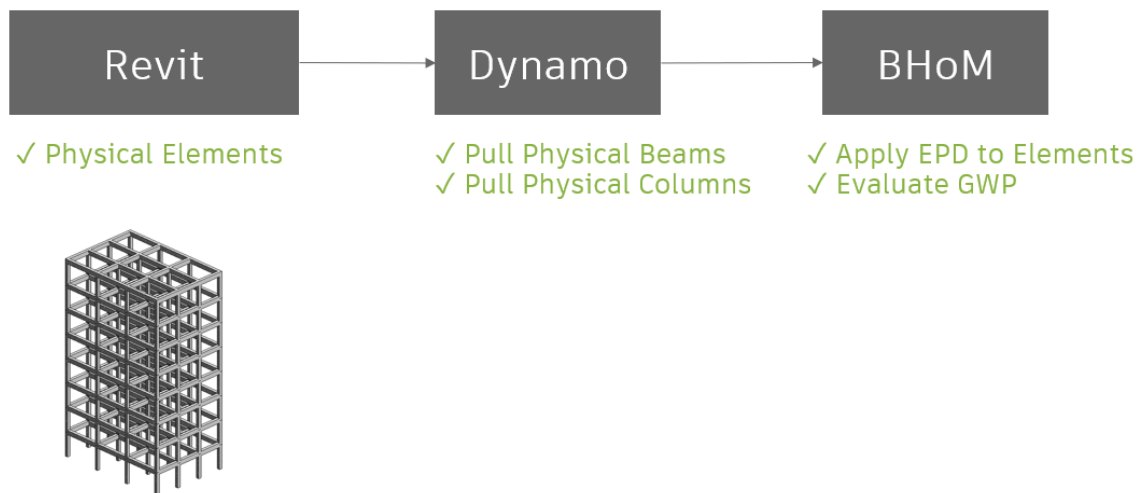
Our last example involved Lifecycle analysis (LCA) using BHoM. For a more thorough discussion about Lifecycle analysis, check out the Resources at the end of this document. To perform a Lifecycle Analysis on a structure, the Global Warming Potential (GWP) must be determined. This is calculated using a quantity and an Environmental Product Declaration (EPD) with the following formula:

$$\begin{array}{|c|} \hline \text{Quantity} \\ \hline \text{(Volume)} \\ \hline \end{array} \times \begin{array}{|c|} \hline \text{EPD} \\ \hline \text{(GWP/Volume)} \\ \hline \end{array} = \begin{array}{|c|} \hline \text{GWP} \\ \hline \text{(kg CO2-eq)} \\ \hline \end{array}$$

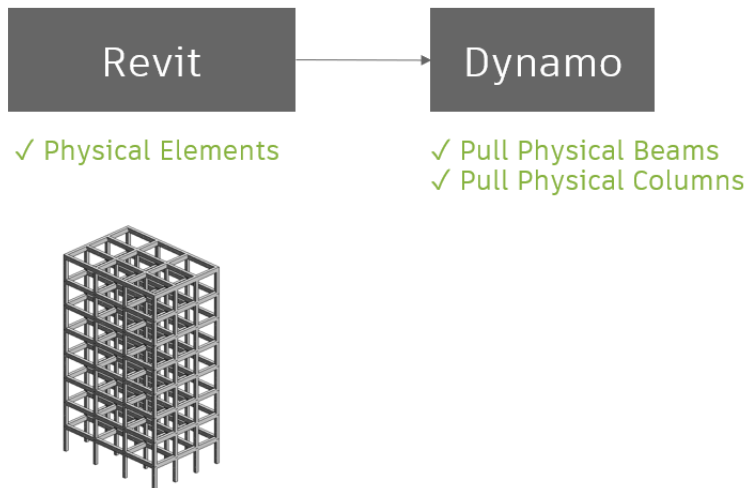
In some cases, the EPD units are GWP/Mass instead of GWP/Volume. Pay close attention when selecting an EPD to verify you are using the correct units. In these examples, EPDs are selected that use GWP/Volume.

We will first open Revit 2020 and the “Example3-Revit2020-LCA.rvt” file. We will perform a Lifecycle analysis using BHoM and the EC3 datasets.

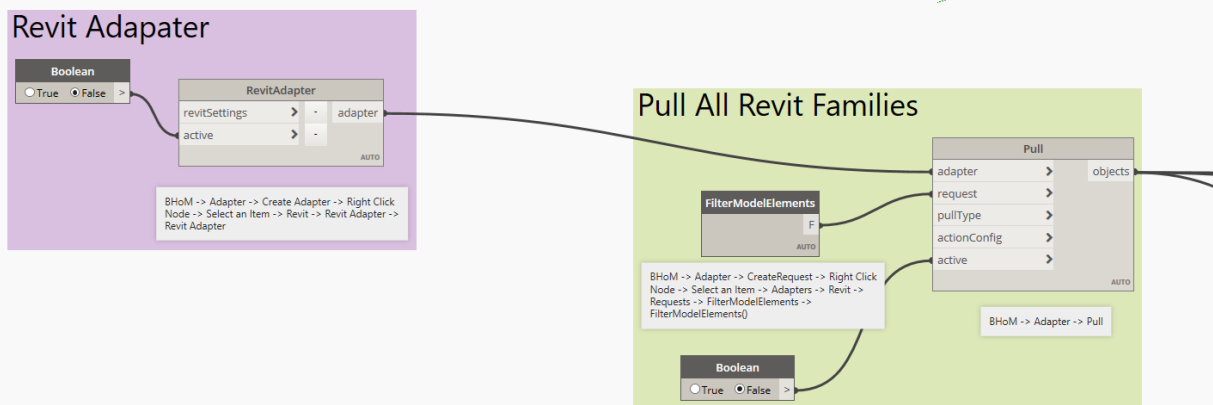
BHoM – Getting Started with LCA



Step 1: Use Dynamo to pull Revit data

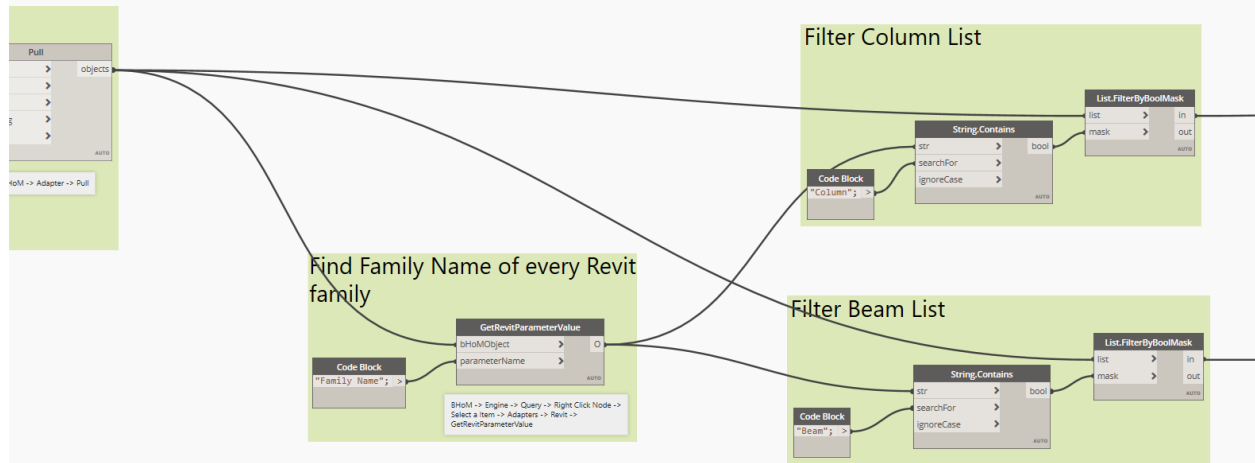


With the Dynamo script open, we will first use a Revit adapter and Pull component to pull all the model elements from Revit. Note we used FilterModelElements as our request – this allows us to pull all model elements from the Revit model.



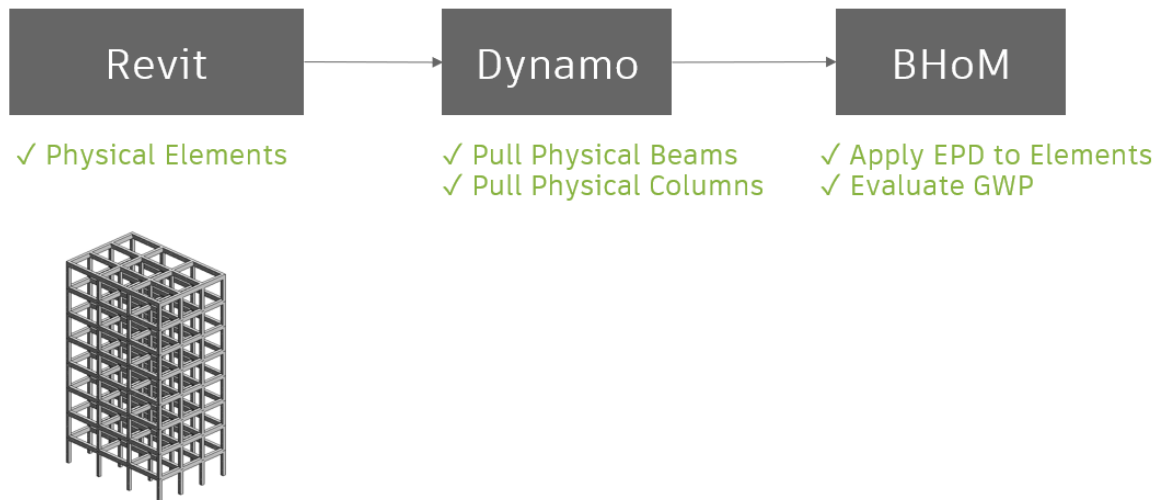
Next, we need to sort through all of the model elements to find the beams and columns. We will use a GetRevitParameterValue node to find the family name of every model element. Then, we will use a filter by Boolean mask to put all columns in one list and all beams in another list. Note if there were slabs, footings, or foundations, one would just need to copy these nodes

accordingly.

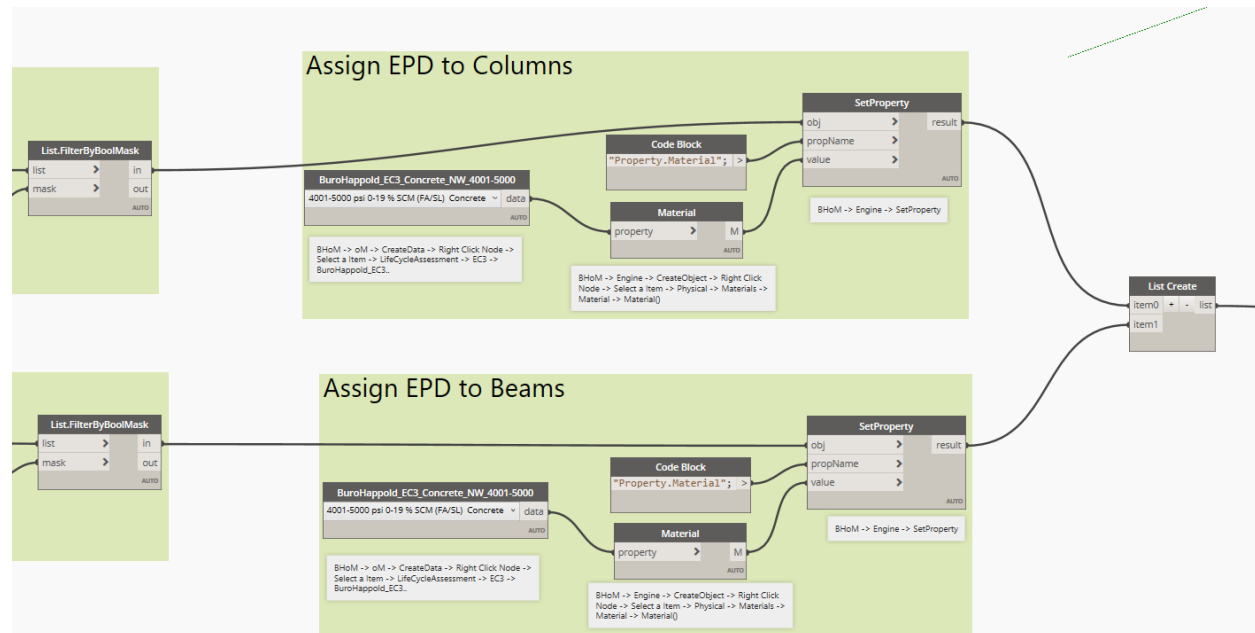


We have now pulled all Revit structural columns into one list and all Revit structural beams into another list. Let's move to the next step.

Step 2: Apply EPDs to Elements using BHoM and Evaluate GWP

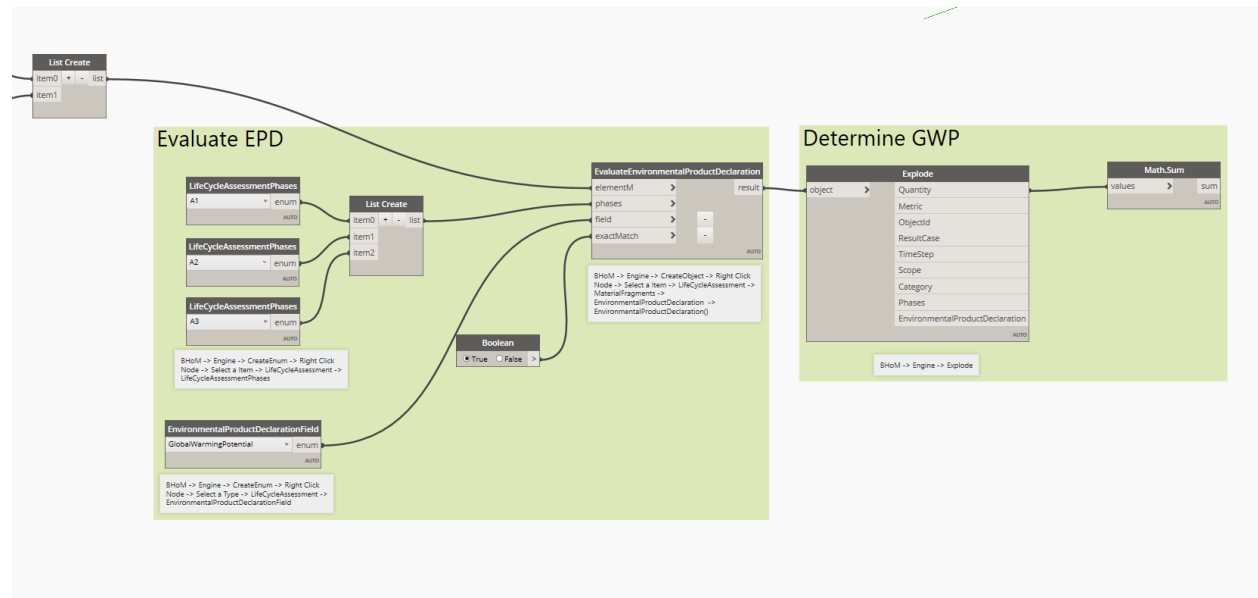


Now we will use BHoM to assign Environmental Product Declarations (EPDs) to the beams and columns. We do this by using the SetProperty components. For the SetProperty component, we want to assign the material property of the BHoM object. Note this is different than a Revit material. For the value, we want to use the BHoM datasets in this example. There are more datasets for EPDs available, and users can also create their own EPDs using BHoM. For this example, we will use the EC3 dataset and a normal weight concrete mix design. Once we have set all the properties, we will create a list of this data.

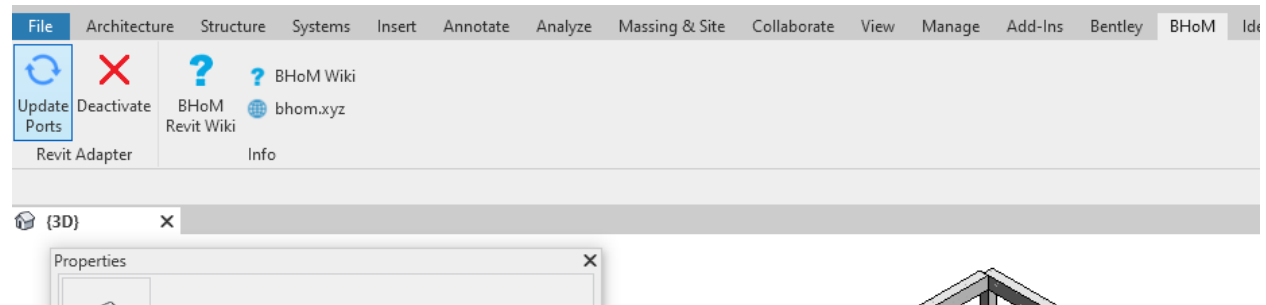


The last step is to evaluate the total GWP of all the elements we assigned EPDs. BHoM requires a few pieces shown below. This may seem excessive as we are only evaluating beams in columns, however, this becomes more powerful as more components are added to perform a complete Lifecycle analysis – i.e., mechanical scope, electrical scope, and structural scope.

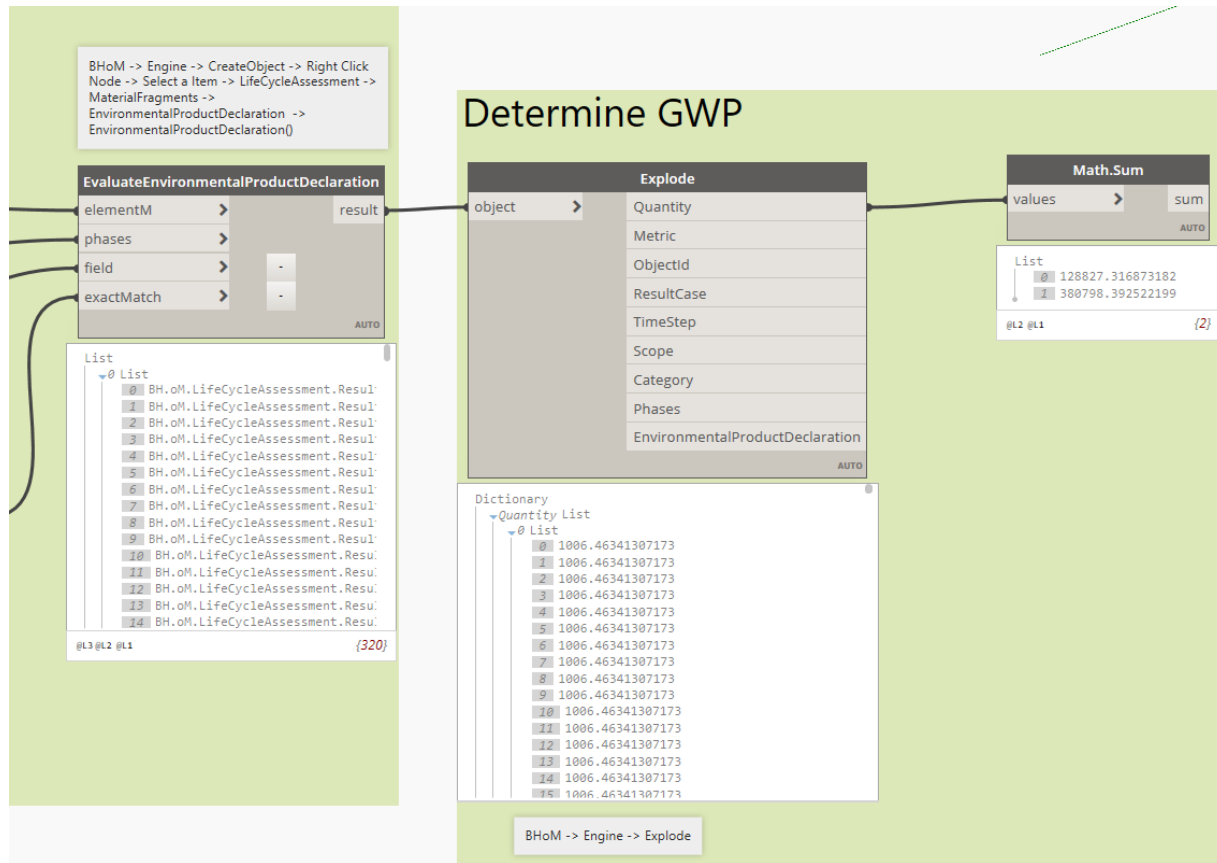
BHoM requires phases as shown and they are combined into a list. BHoM requires an EPD field, and in this case, we will be evaluating the Global Warming Potential (GWP). Once we have added all the necessary nodes to evaluate the EPD, we then need to explode the object. When we explode any object, it breaks down that piece of data into its components. For this example, we want the quantity of the GWP of the beams and columns. Our last step is to sum the values to see the total GWP.



To run the script, remember to go into Revit and the BHoM tab, and click activate, the tab should look below:

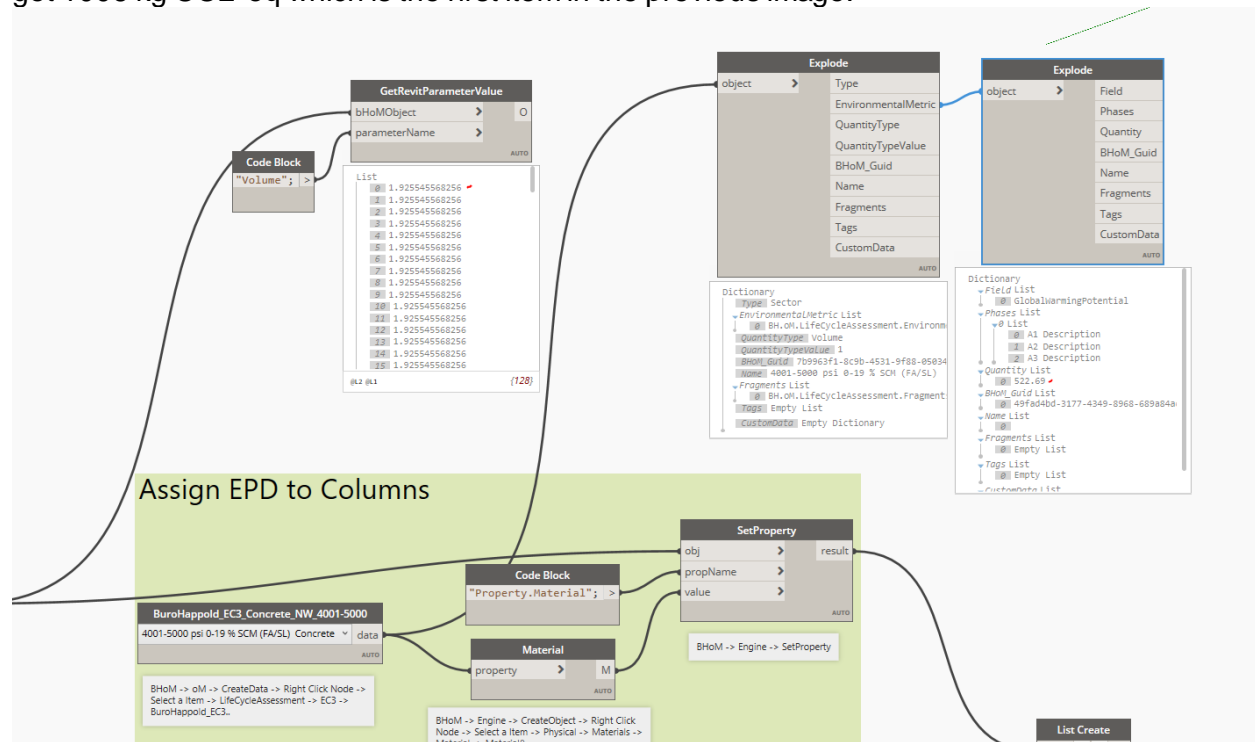


In Dynamo, set all the boolean toggles to True and then click Run at the bottom of the Dynamo window to run the script. Your script should look similar to below:



So, in simple terms, the total global warming potential (GWP) of this structure is 128,827 kg CO2-eq for the beams and 380,798 kg CO2-eq for the columns. One could take the individual GWP of each element and assign it to each individual Revit element, send it to Excel or other platforms to perform more analysis, graphs, etc. Now, we have performed a Lifecycle analysis on this Revit model using Dynamo and BHoM.

If you're interested on the math to get the results, the volume of the first beam in the list is 1.9255 m3 (see below). If we explode the EC3 concrete component a few times, we see that the EPD is 522.69 GWP/Volume (see below). When we multiply these two numbers together, we get 1006 kg CO2-eq which is the first item in the previous image.



Helpful Hint: If for any reason the script did not work, set all the toggles to false, run the script, then set all the toggles to true and run the script again. If the explode node is not working, right click the node and select "Update Outputs". Sometimes, this helps the script run properly.

AUTODESK UNIVERSITY

Resources

Computational Engineering:

[Home Page - The Computational Engineer](#)

[The Computational Engineer - YouTube](#)

Dynamo Learning Tools:

[Learn - Dynamo BIM](#)

Lifecycle Analysis:

<https://www.autodesk.com/autodesk-university/article/Using-Revit-and-Dynamo-Assess-Embodied-Carbon-2021>