

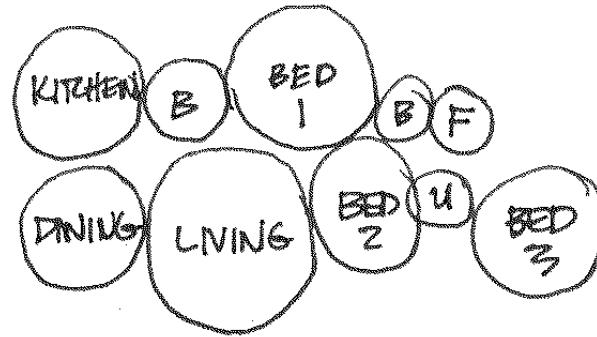
# **Example**

## Space Planning

# Space Planning Goals

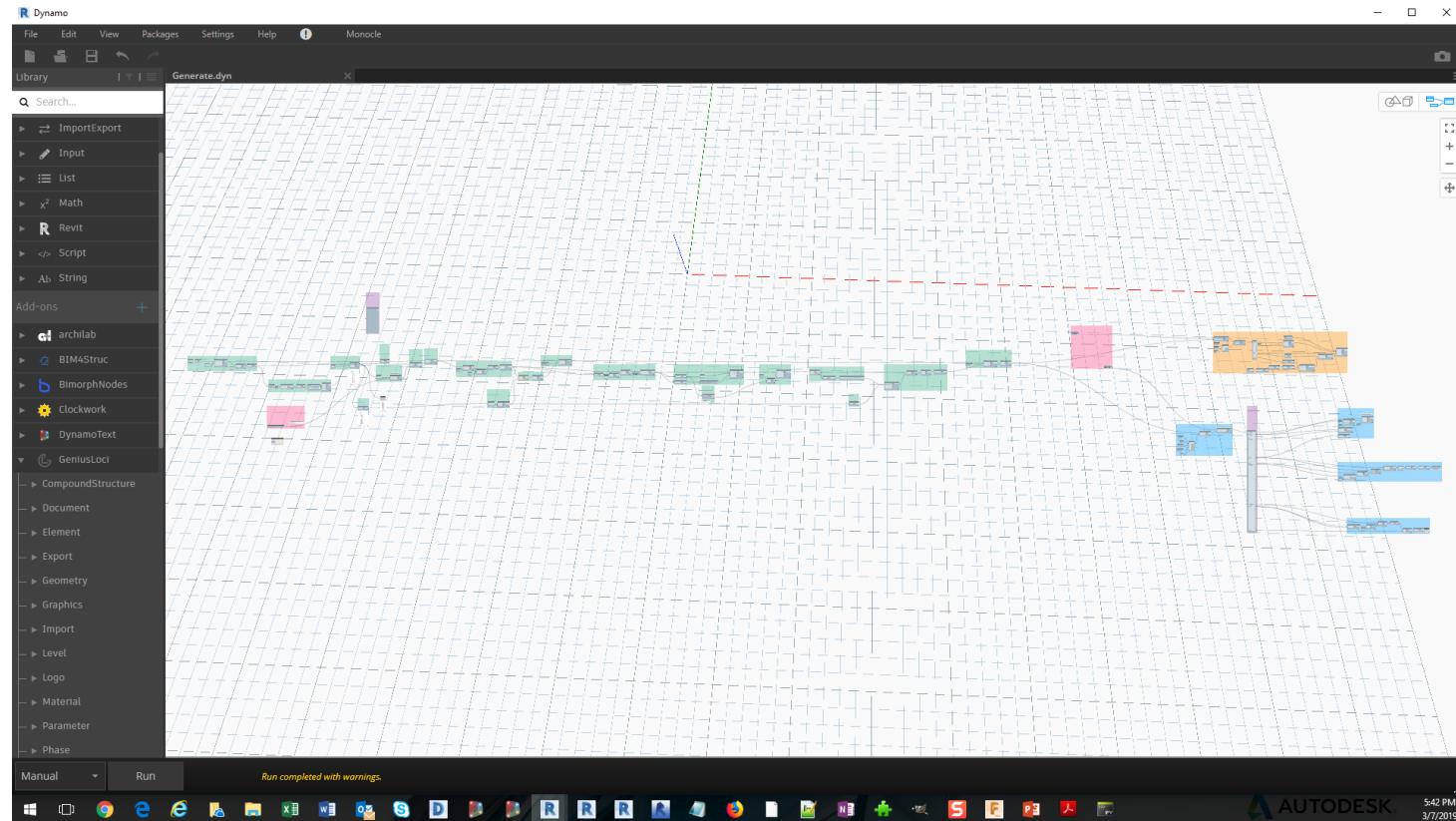
What are we solving for?

- Generate a bunch of apartment layouts
- Variable Inputs
  1. Permutation of program
- Fixed Inputs
  - Desired Adjacencies
  - Noise Score for each room
- Goals
  1. Minimize proximity score
  2. Minimize noise score



# Space Planning Process

## Overall View



# Space Planning Process

## Start with the Program



Simple.xlsx - Last Saved 3/13/2019 9:03 PM

Jacob Small

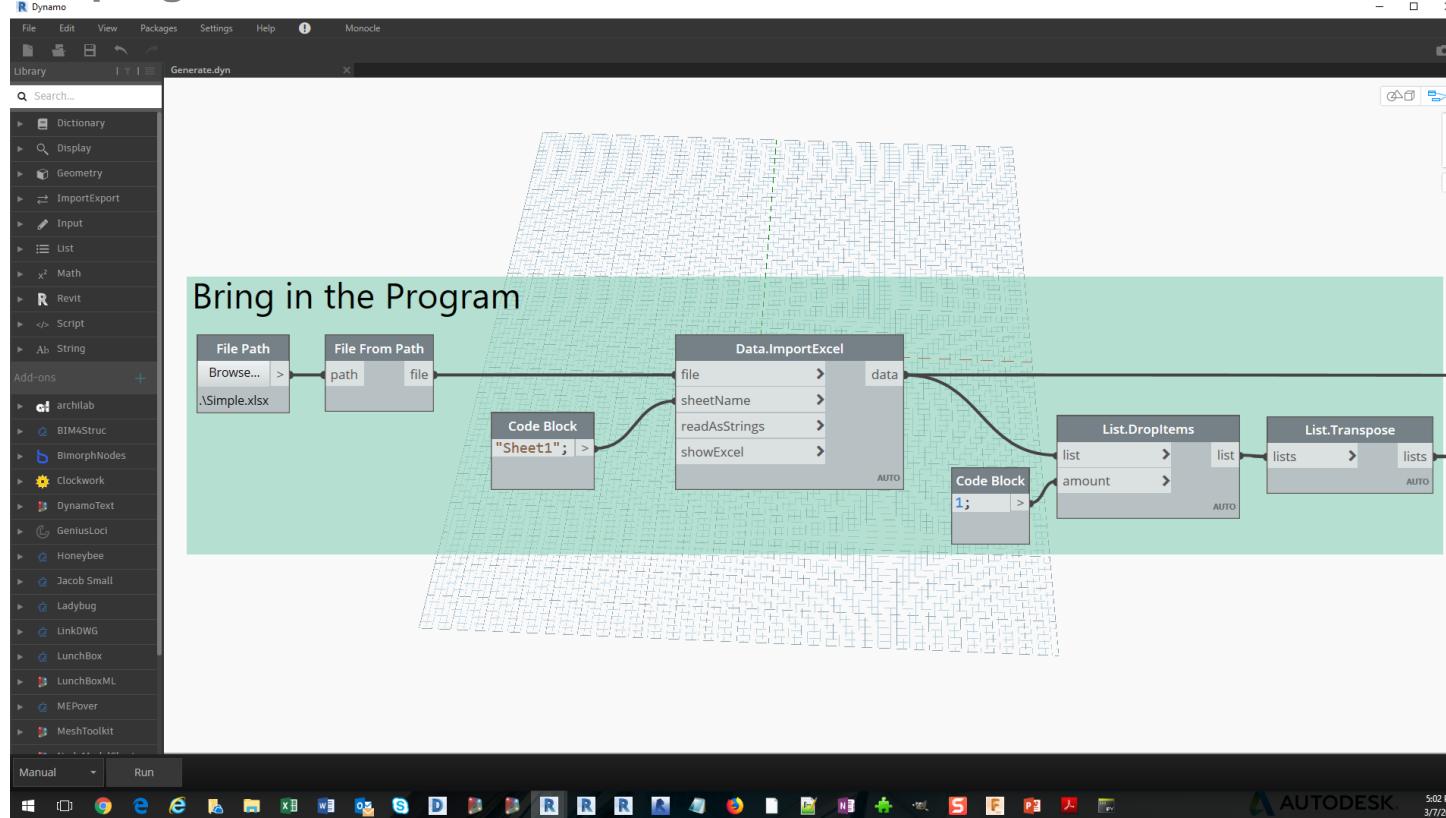
File Home Insert Draw Page Layout Formulas Data Review View Developer Results Connect Help Team Tell me what you want to do

C6

	A	B	C	D	E	F	G	H	I	J
1	SPACE ID	SPACE NAME	DEPARTMENT	AREA	COLOUR	ADJACENT SPACES	NOISE			
2	0	hall		150	128,128,128	util, living, bath 2, bed 1, bed 2, bed 3	3			
3	1	dining		175	255,0,0	living, kitchen	5			
4	2	kitchen		189	255,128,0	dining, living	5			
5	3	bath 1		99	255,0,128	bed 1	2			
6	4	bath 2		55	255,128,128	hall	2			
7	5	bed 1		238	255,255,0	hall, bath 1	1			
8	6	living		372	255,0,255	dining, kitchen	5			
9	7	util		37	128,0,128	hall	2			
10	8	foyer		38	0,255,0	living	5			
11	9	bed 2		150	0,255,128	hall	1			
12	10	bed 3		204	0,255,255	hall	1			
13										
14										
15										
16										
17										

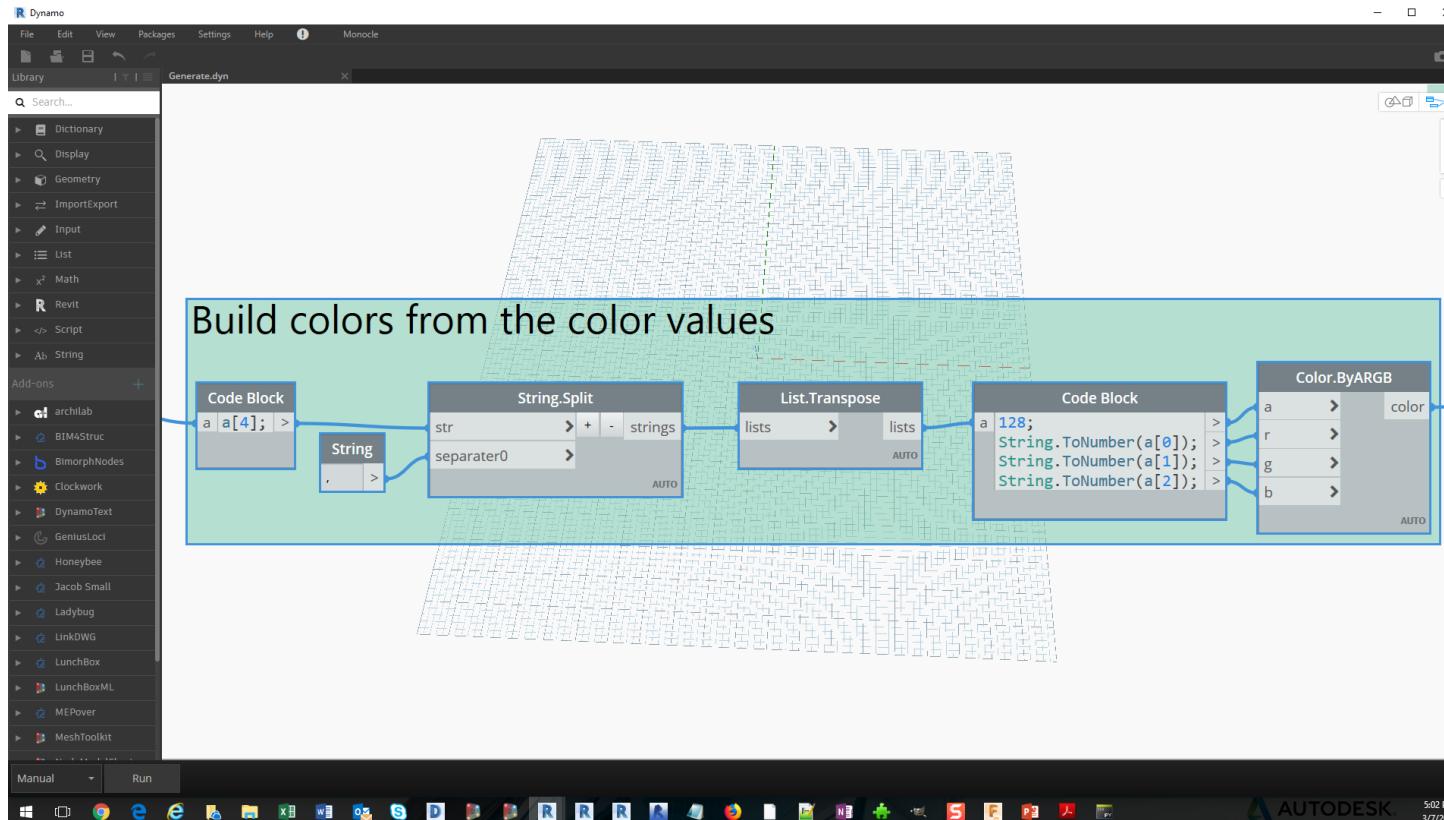
# Space Planning Process

## Bring in the program



# Space Planning Process

## Define the Colors



# Space Planning Process

## Replace the values



Replace our color values with the produced colors.

```
graph TD
    CB[Code Block: 4;] --> RI[Get Room]
    RI --> R[Get Room]
    R --> CR[Color]
    CR --> ST[Set Room Color]
    ST --> RT[Revit Transaction]
    RT --> CT[Commit Transaction]
    LI[List.ReplaceItemAtIndex]
    LI -- list --> RL[Revit List]
    RL --> GetRooms[Get Rooms]
    LI -- index --> GRI[Get Room Index]
    GRI --> RI
    LI -- item --> CR
```

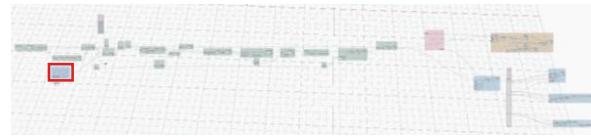
Dynamo Software Interface:

- File Edit View Packages Settings Help
- Library
- Search... Dictionary Display Geometry ImportExport Input List x<sup>2</sup> Math Revit Script String
- Add-ons archilab BIM4Struc BimorphNodes Clockwork DynamoText GenusLoc Honeybee Jacob Small Ladybug LinkDWG LunchBox LunchBoxML MEPover MeshToolkit
- Manual Run

AUTODESK 5:03 PM 3/7/2019

# Space Planning Process

## Define the input



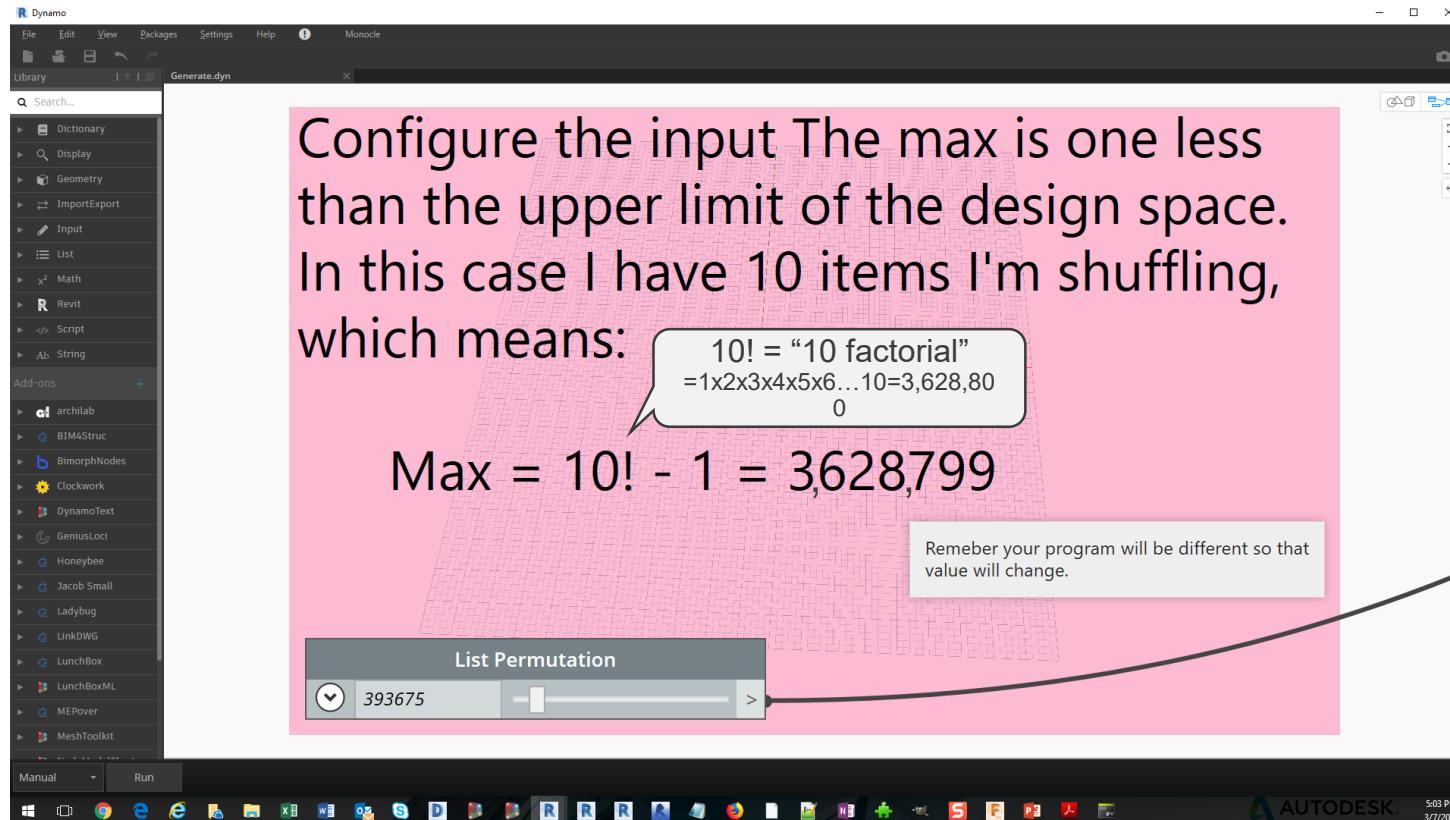
Configure the input The max is one less than the upper limit of the design space. In this case I have 10 items I'm shuffling, which means:

$10! = "10 factorial"$   
 $=1x2x3x4x5x6...10=3,628,800$

$\text{Max} = 10! - 1 = 3628799$

Remeber your program will be different so that value will change.

List Permutation  
393675 >



**~1 week to run  
3,628,800  
permutations!**

**(@ 1 sec per run on 6 cores)**



# Space Planning Process

## Get the permutation



Custom node to get the nth permutation lexocographical order.

Lexocographical [ **lex-i-co-graph-i-cal** ],  
adjective Related to the principles and  
procedures involved in writing, editing, or  
compiling dictionaries.

Custom Node: Jacob Small

Code Block

```
a a[0]; >
```

nth permutation

list Permutation AUTO

SPACE ID SPACE NAME

0	hall
1	dining
2	kitchen
3	bath 1
4	bath 2
5	bed 1
6	living
7	util
8	foyer
9	bed 2
10	bed 3

# Space Planning Process

This is the Python contained in that custom node (since I figured you'd ask)



This is the python contained in the "nth permutation" node.

```
String
import clr
from math import factorial

# The inputs to this node will be stored as a list in the IN variables.

def from_rank(rank, lst):
    my_lst = sorted(lst)
    ret = []
    while my_lst:
        fact = factorial(len(my_lst) - 1)
        idx, rank = divmod(rank, fact)
        ret.append(my_lst.pop(idx))
        assert rank == 0 # invalid rank
        if out of range
            return ret

    lst = IN[0]
    rank = IN[1]

    perm = from_rank(rank, lst)

    # Assign your output to the OUT variable.
    OUT = perm
```

The screenshot shows the Dynamo software interface. On the left is the library pane with categories like Dictionary, Display, Geometry, ImportExport, Input, List, Math, Revit, Script, and String. Below it is the add-ons pane listing archilab, BIM4Struc, BimorphNodes, Clockwork, DynamoText, GeniusLoc, Honeybee, Jacob Small, Ladybug, LinkDWG, LunchBox, LunchBoxML, MEPower, and MeshToolkit. The main workspace shows a complex 3D grid-based model. A node editor window is open, displaying the Python code for a custom node named "String". The code implements a function `from\_rank` that generates the nth permutation of a list. It uses sorting, division, and modulo operations to calculate the index and rank. The code also includes assertions and handling for out-of-range errors. The bottom status bar shows the Autodesk logo and the date/time (5/04 PM 3/7/2019).

# Space Planning Process

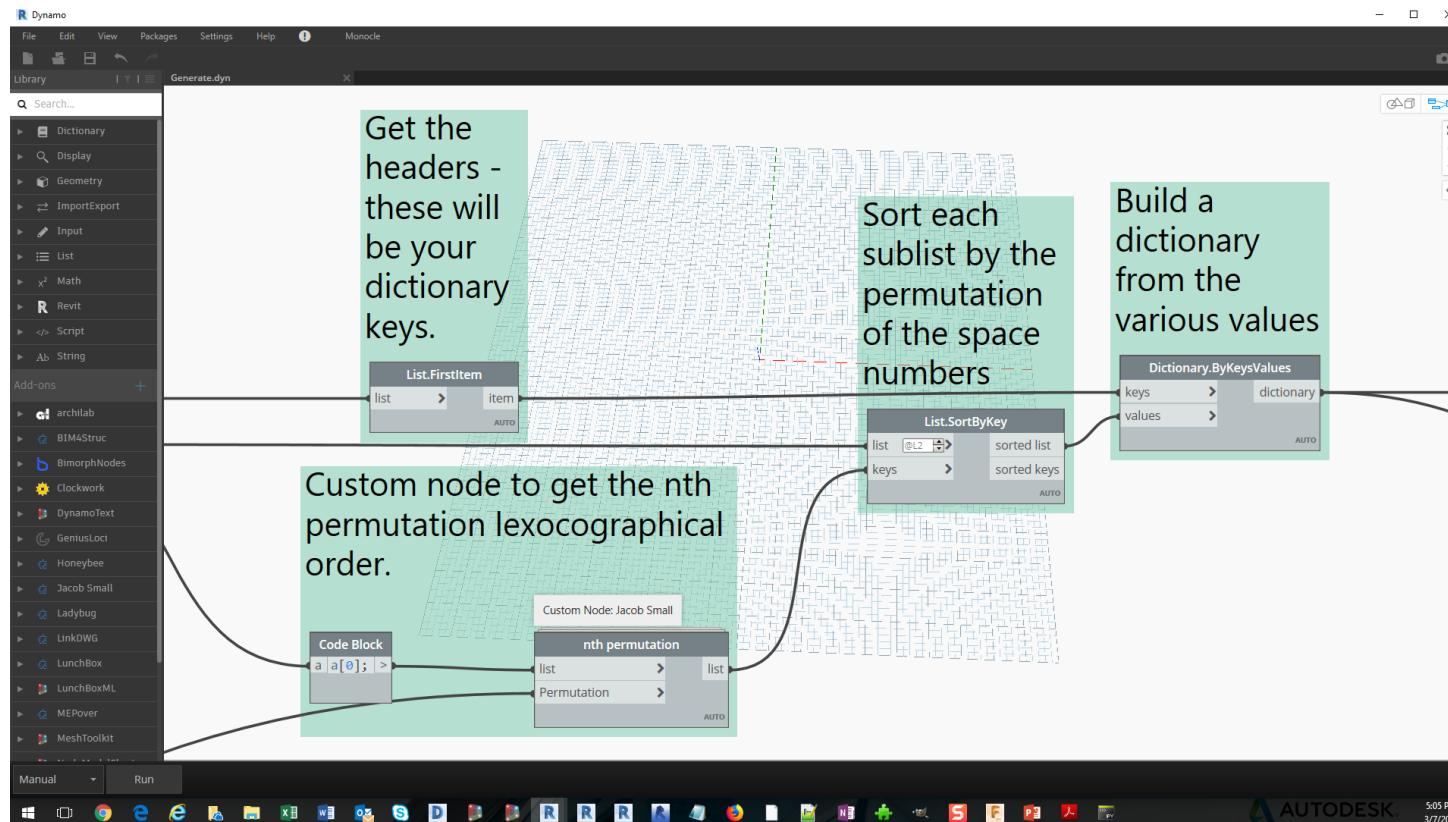
## Room list ordering pattern

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
hall																				
dining																				
kitchen																				
bath 1																				
bath 2																				
bed 1																				
living																				
util	util	util	util	util	util	foyer	foyer	bed 2	bed 3	bed 2	bed 3	bed 2	bed 3	foyer	foyer	bed 2	bed 3	bed 2	bed 3	foyer
foyer	foyer	bed 2	bed 3	bed 2	bed 3	util	bed 2	bed 3	foyer	bed 3	bed 2	bed 3	foyer							
bed 2	bed 3	foyer	foyer	bed 3	bed 2	bed 2	bed 3	foyer	foyer	bed 3	bed 2	util	util	util	util	util	util	bed 3	bed 2	bed 3
bed 3	bed 2	bed 3	bed 2	foyer	foyer	bed 3	bed 2	bed 3	bed 2	foyer	foyer	bed 3	bed 2	bed 3	bed 2	foyer	util	util	util	util



# Space Planning Process

Combine the data into a dictionary



# Space Planning Process

Build a shape from your data (except the hall).



Build a rectangle for all spaces (except the hallway).

The screenshot shows the Dynamo interface with a node graph. The graph consists of several nodes connected by arrows:

- A **Code Block** node with the value `"AREA";` connects to a **Dictionary.ValueAtKey** node. This node has two outputs: `dictionary` and `value`.
- The `value` output connects to a **List.DropItems** node.
- The **List.DropItems** node has two outputs: `list` and `amount`.
- The `list` output connects to a **Math.Sqrt** node. This node has two outputs: `number` and `sqrt`.
- The `number` output connects to a **Rectangle.ByWidthLength** node.
- The `sqrt` output connects to another **Rectangle.ByWidthLength** node.
- The second **Rectangle.ByWidthLength** node has two outputs: `width` and `length`.
- The `width` and `length` outputs both connect to a **Rectangle** node.

The background of the workspace shows a 3D wireframe model of a building interior, representing the space being planned.

# Space Planning Process

Build a shape for the hall



Gets the hallway data (I kept that first in the list) and pulls the area and divides by 3 (assuming 3' wide hallway)

The screenshot shows the Dynamo interface with a node graph. A 'Code Block' node contains the following script:

```
a a[3][0]/3;  
3;
```

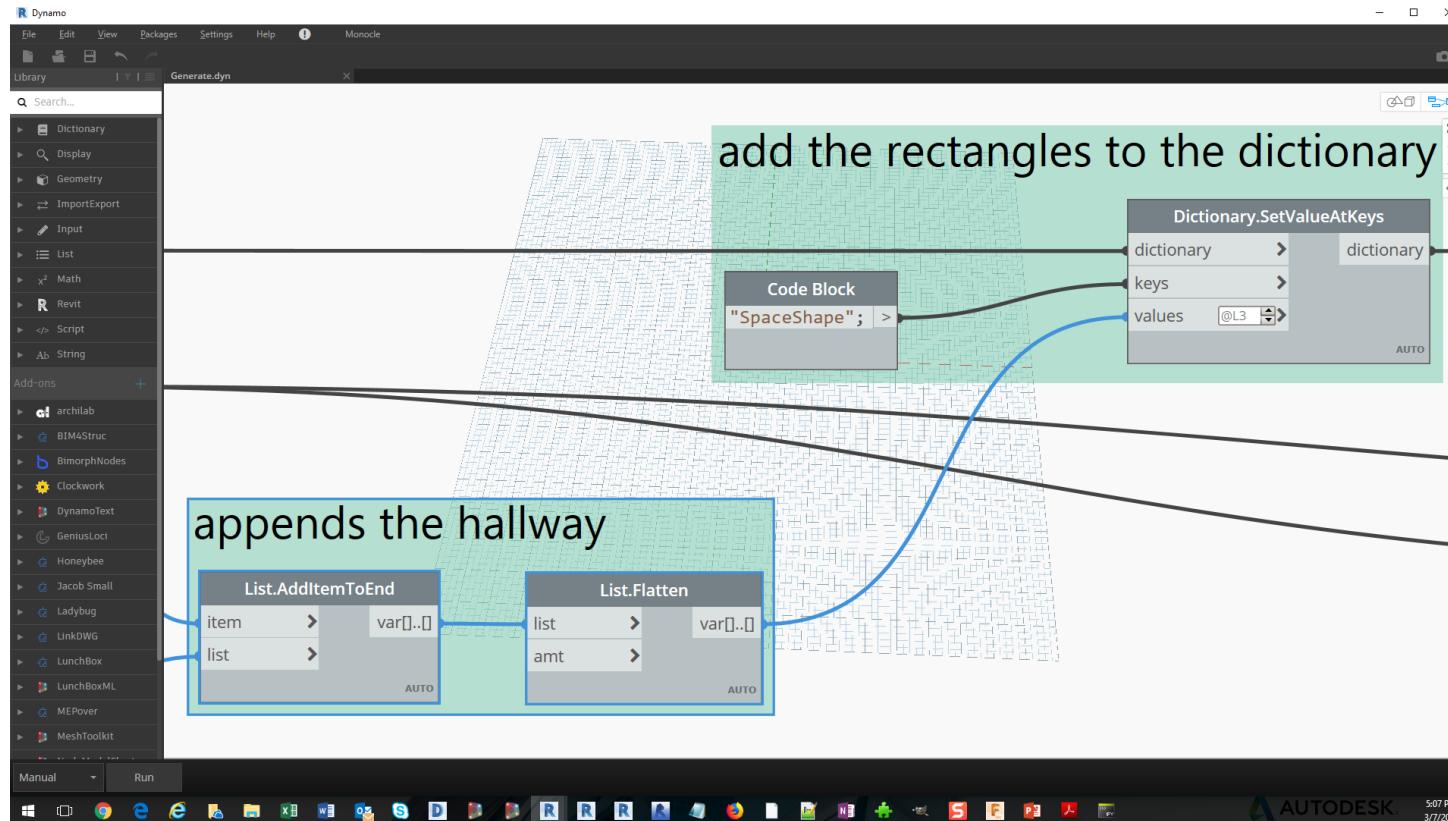
The output of this node connects to a 'Rectangle.ByWidthLength' node. This node has two outputs: 'width' and 'length'. These outputs connect to a 'Rectangle' node, which is set to 'AUTO' mode. The 'Rectangle' node is shown as a black U-shaped curve on a grid background.

The left sidebar shows the 'Library' tab with categories like ImportExport, Input, List, Math, Revit, Script, String, and Add-ons. The 'Generate.dyn\*' file is open in the main workspace.

# Space Planning Process

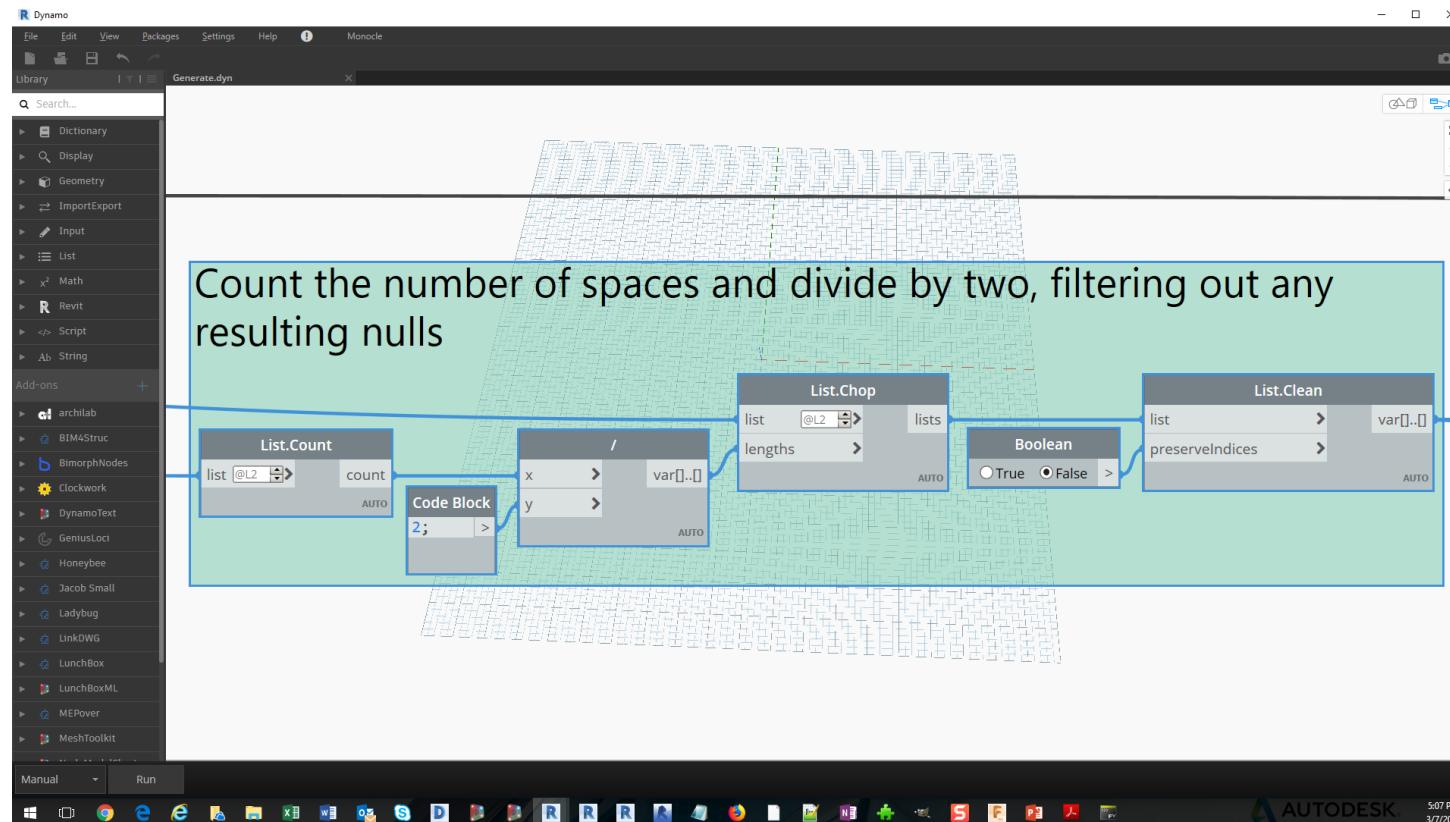


Adds the hall shape to the list of shapes and then adds the shapes to the dictionary



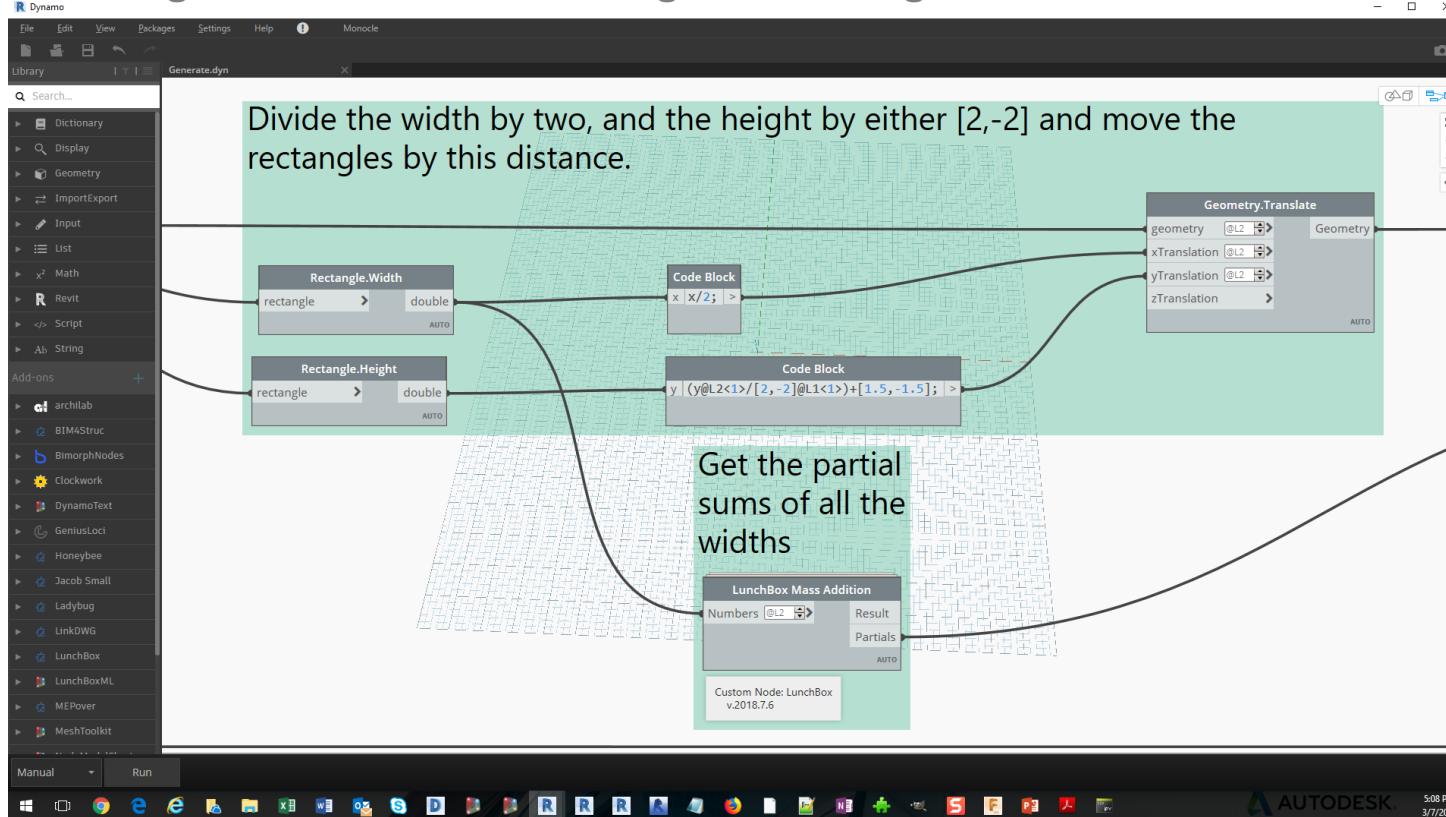
# Space Planning Process

Split the list in two – one list for either side of the hall



# Space Planning Process

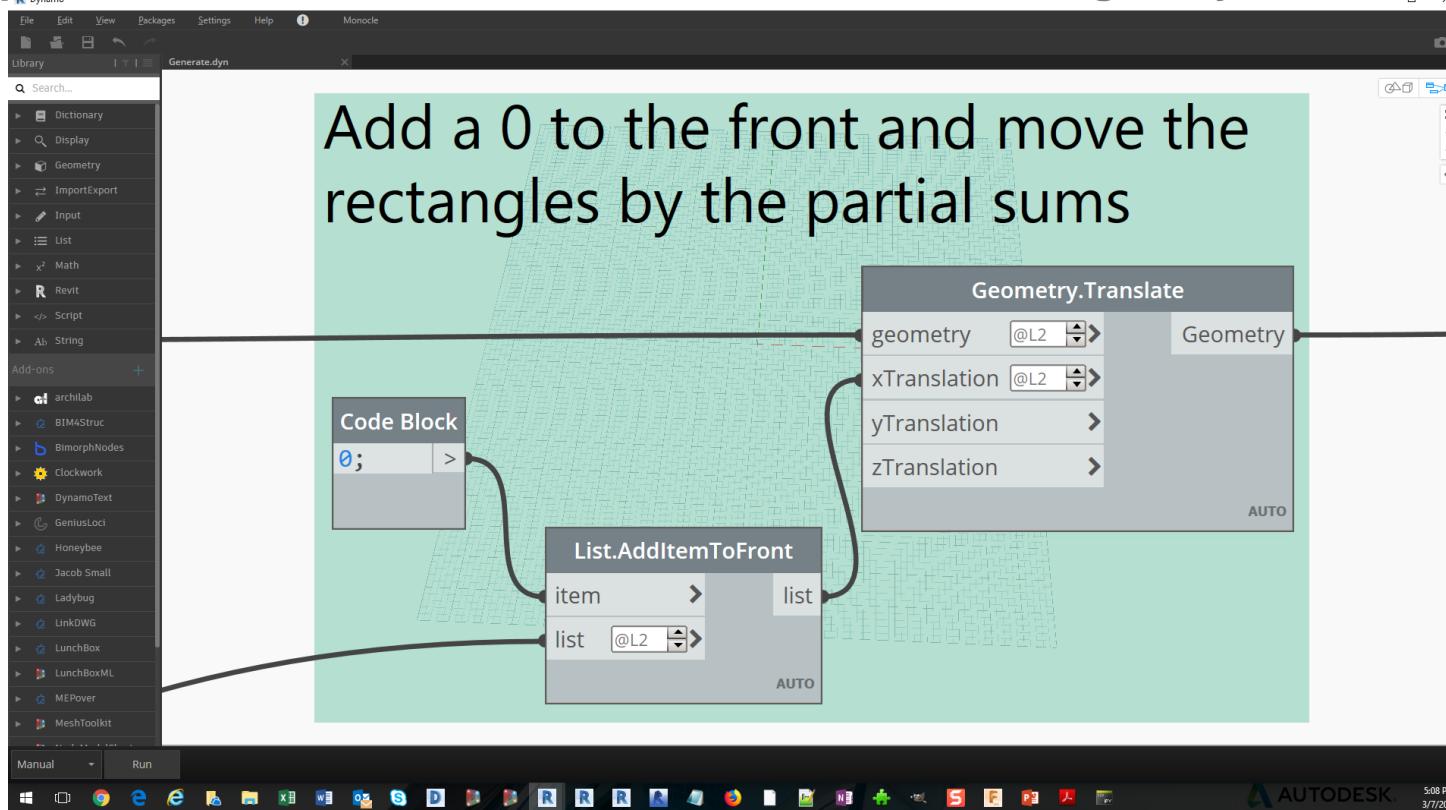
Move the rectangle's so their corners align to the origin



# Space Planning Process

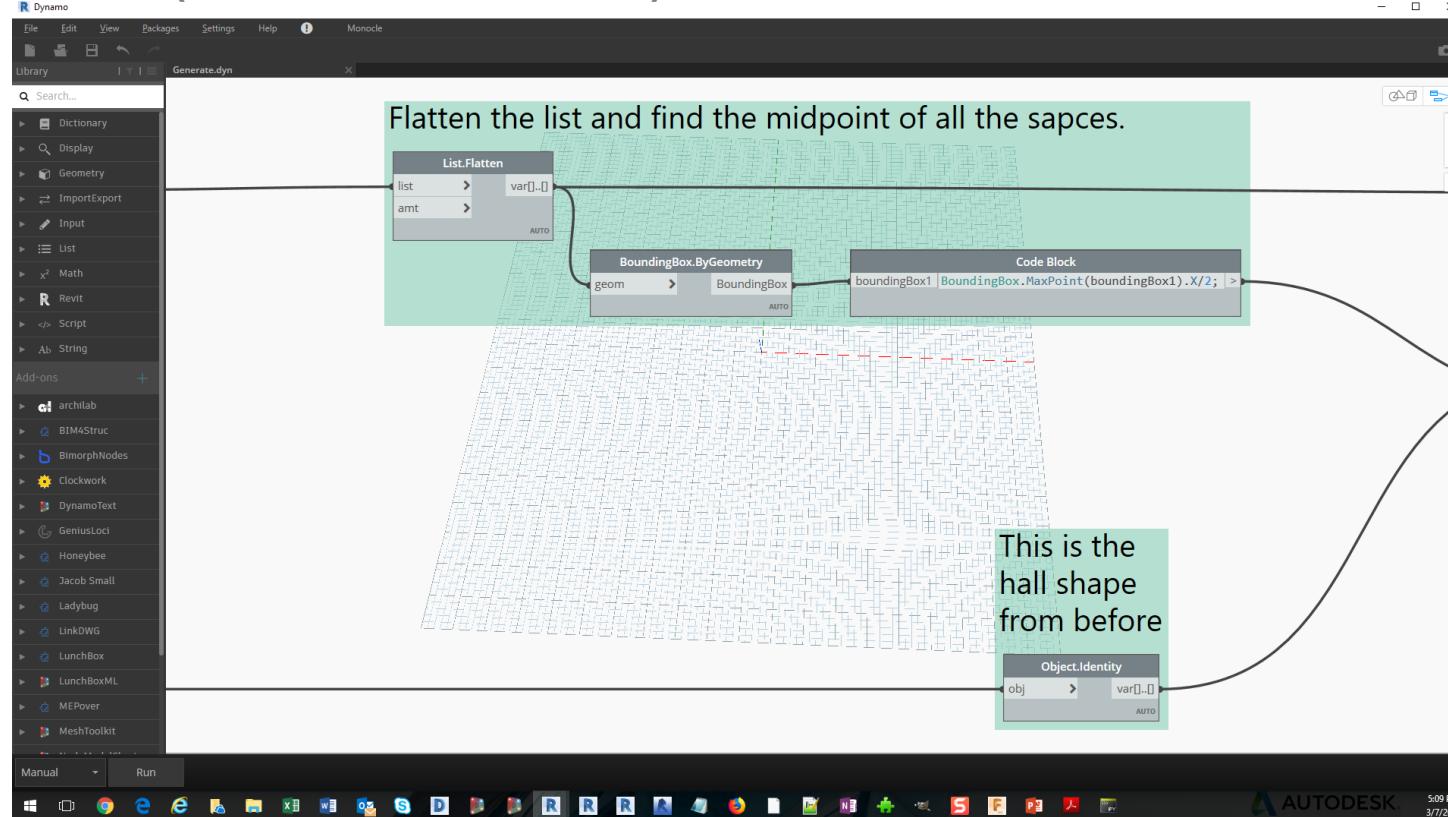


Get the partial sums of the list of widths, and move the rectangles by these distances



# Space Planning Process

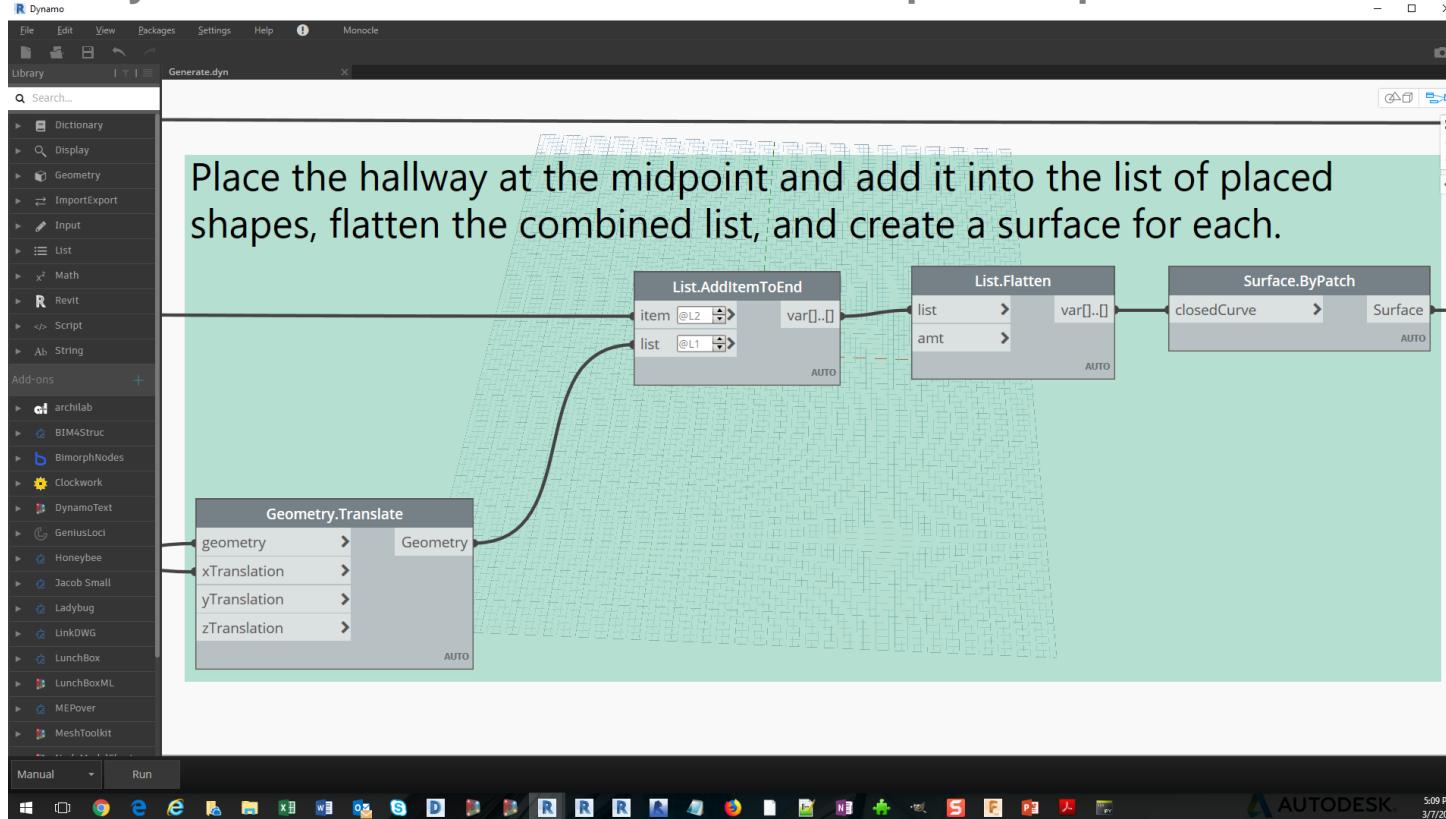
Find the middle (the hall should be here)



# Space Planning Process



Put the hallway in the middle and add it to the list of placed spaces



# Space Planning Process

Add the spaces to the dictionary and flatten it



Add the space location (surface) to the overall dictionary and flatten it.

Generation is done time to evaluate.

```
graph LR; SL[SpaceLocation] --> DS[Dictionary.SetValueAtKeys]; DS --> LF[List.Flatten]; DS --> L[amt]; DS --> V[values];
```

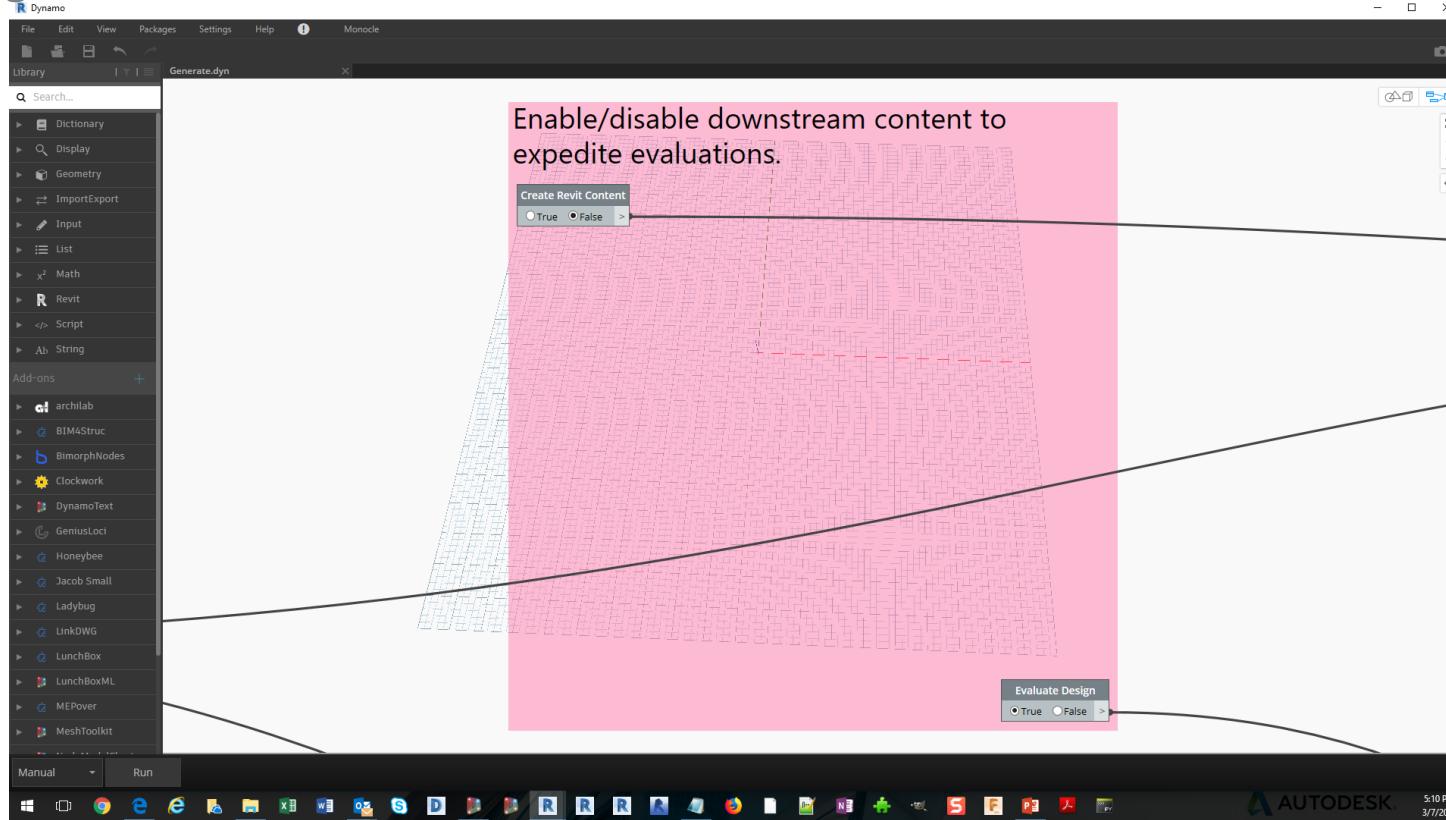
The screenshot shows the Dynamo interface with a script open. The script consists of three main nodes:

- A **String** node labeled **SpaceLocation**.
- A **Dictionary.SetValueAtKeys** node with inputs for **dictionary**, **keys** (set to **@L3**), and **values**.
- A **List.Flatten** node with inputs for **list** and **amt**.

The **SpaceLocation** node has a connection to the **values** input of the **Dictionary.SetValueAtKeys** node. The **Dictionary.SetValueAtKeys** node has connections to both the **list** and **amt** inputs of the **List.Flatten** node. The **Dictionary.SetValueAtKeys** node also has a connection to its own **dictionary** output.

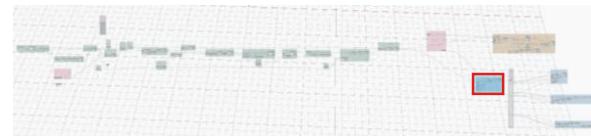
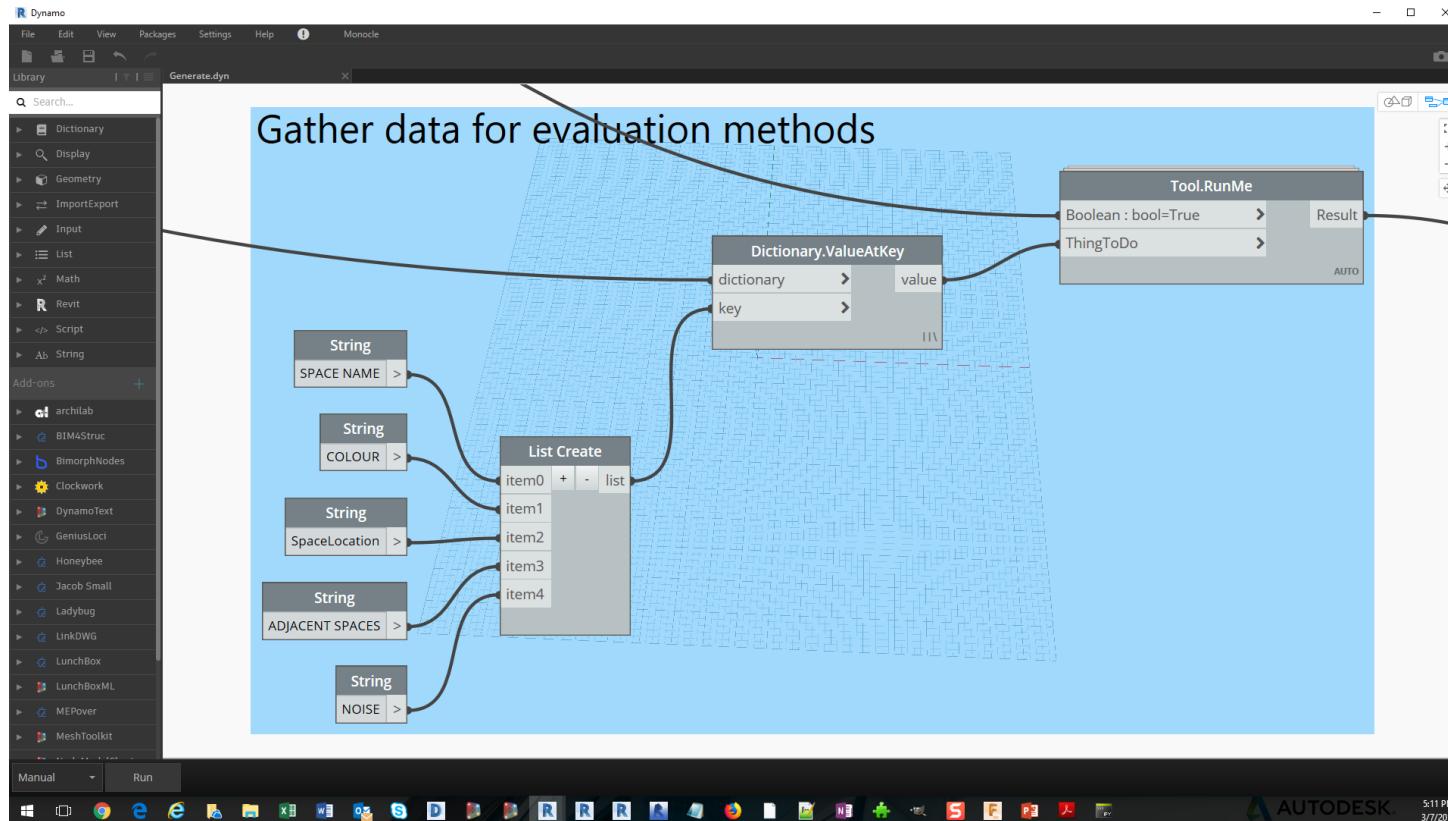
# Space Planning Process

## Add triggers



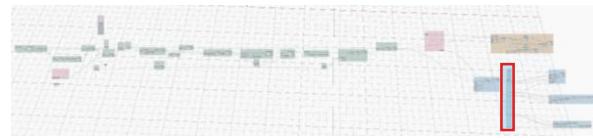
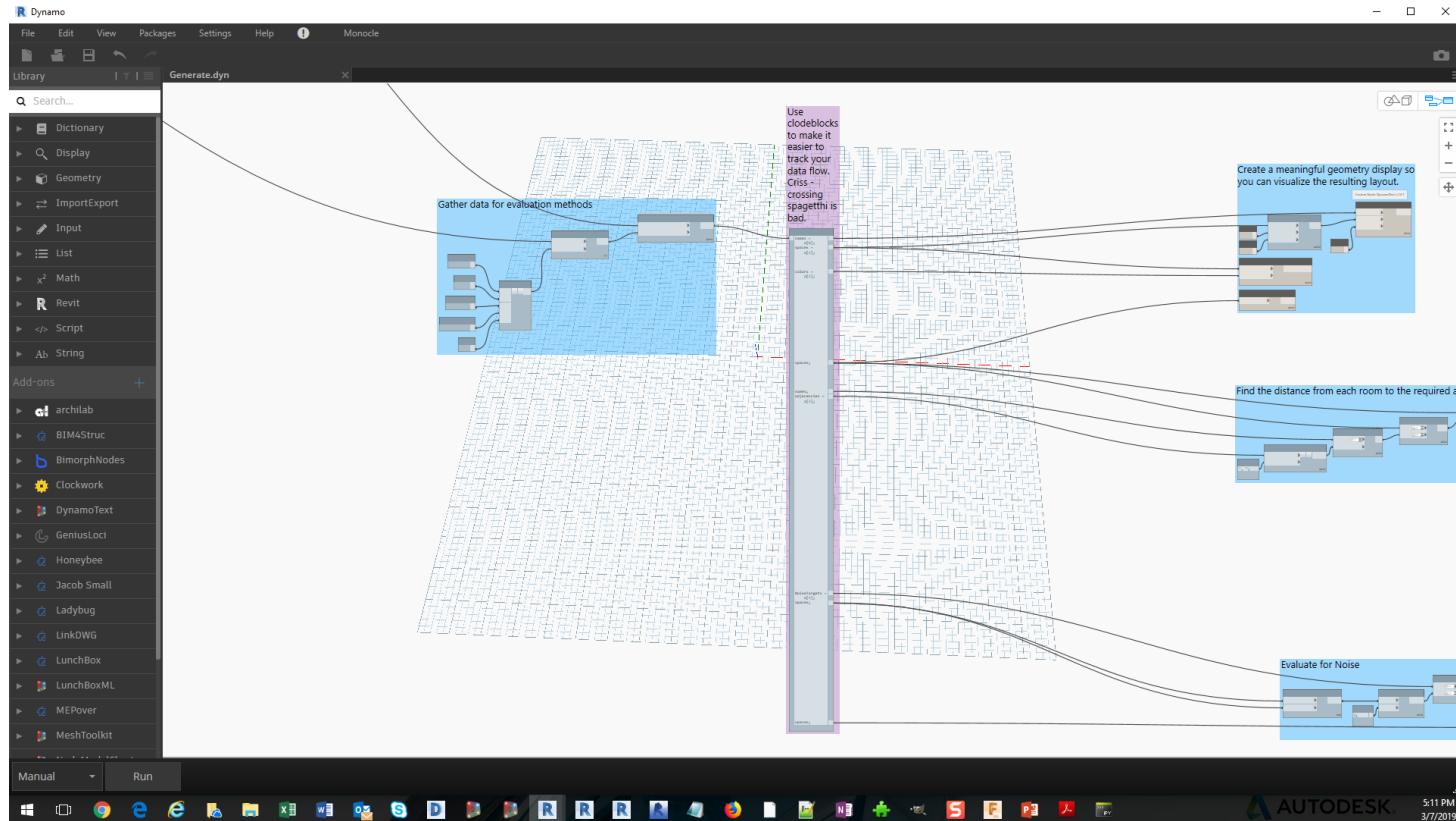
# Space Planning Process

## Gather data for evaluations



# Space Planning Process

## Maintain Readability



# Space Planning Process

## Set up a visualization



Create a meaningful geometry display so you can visualize the resulting layout.

The screenshot shows the Dynamo software interface with a node graph for generating a visualization. The graph consists of the following nodes and connections:

- Surface.PointAtParameter**:
  - Inputs: surface, u (Number 0.2500), v (Number 0.5000).
  - Outputs: Point.
- Text.FromStringOriginAndScale**:
  - Inputs: text (Curve[]), origin, scale (Number 2.0000).
  - Outputs: Custom Node: DynamoText v.2.0.1.
- GeometryColor.ByGeometryColor**:
  - Inputs: geometry (GeometryColor), color.
  - Outputs: AUTO.
- Surface.PerimeterCurves**:
  - Inputs: surface.
  - Outputs: Curve[].

The software interface includes a toolbar at the top with File, Edit, View, Packages, Settings, Help, and Monocle buttons. The left sidebar is the Library with sections like Dictionary, Display, Geometry, ImportExport, Input, List, Math, Revit, Script, String, and Add-ons. The bottom toolbar has Manual and Run buttons, and the taskbar shows various application icons.



# Space Planning Process

## Measure Proximity Score



Dynamo

File Edit View Packages Settings Help Monocle

Library Search... Dictionary Display Geometry ImportExport Input List Math Revit Script String

Add-ons archilab BIM4Struc BimorphNodes Clockwork DynamoText GeniusLoci Honeybee Jacob Small Ladybug LinkDWG LunchBox LunchBoxML MEPower MeshToolkit

Generate.dyn\*

Find the distance from each room to the required adjacent room per the excel spreadsheet.

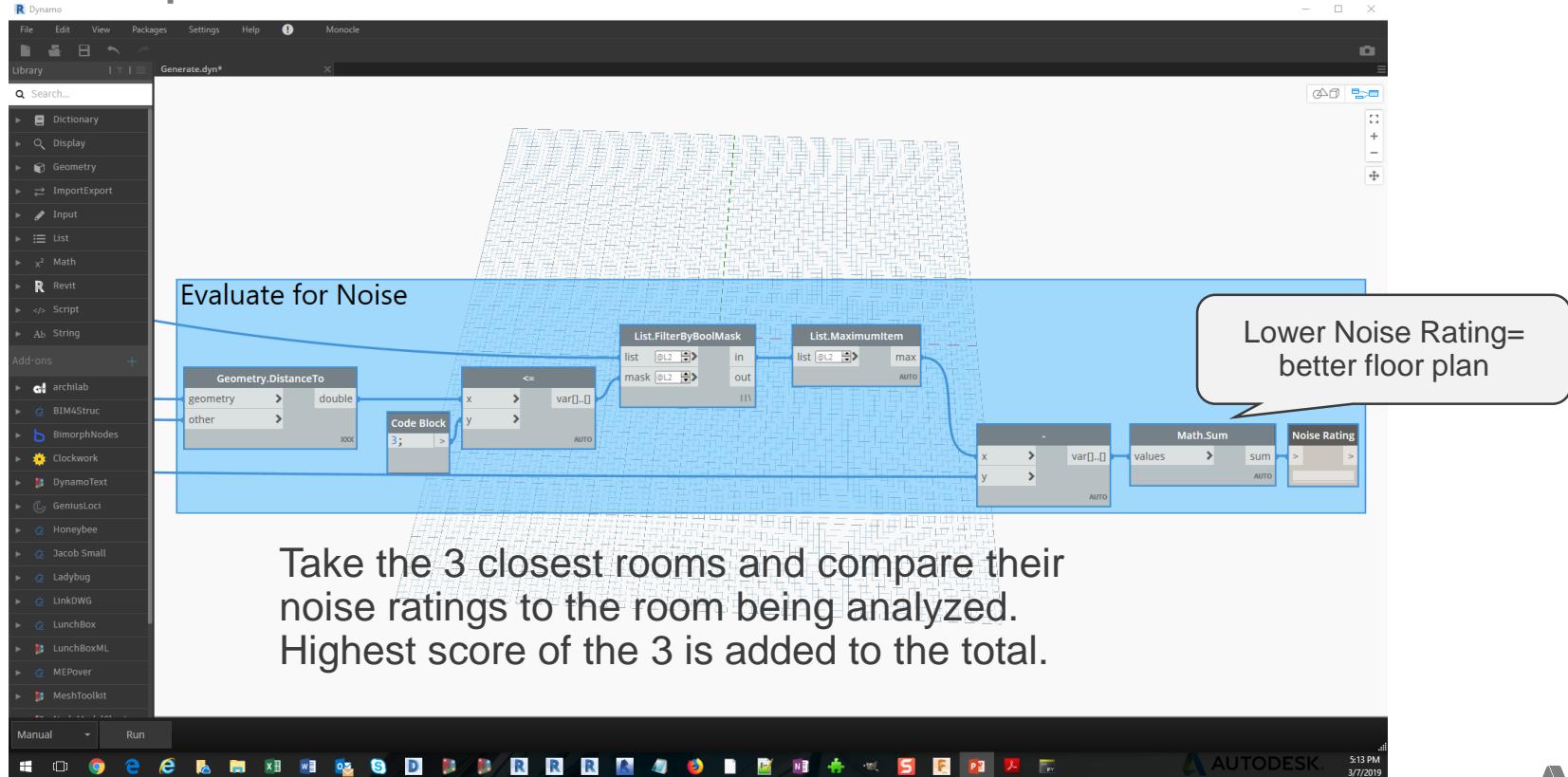
```
graph LR; StringSplit[Code Block: String.Split] --> ListGetIndex[List.GetItemAtIndex]; ListGetIndex --> ListIndexOf[List.IndexOf]; ListIndexOf --> GeometryDistanceTo[Geometry.DistanceTo]; GeometryDistanceTo --> CodeBlock1[Code Block]; CodeBlock1 --> MathSum1[Math.Sum]; MathSum1 --> MathSum2[Math.Sum]; MathSum2 --> ProximityScore[Proximity Score (minimize)]
```

Lower proximity score = better floor plan

Autodesk 5:12 PM 3/7/2019

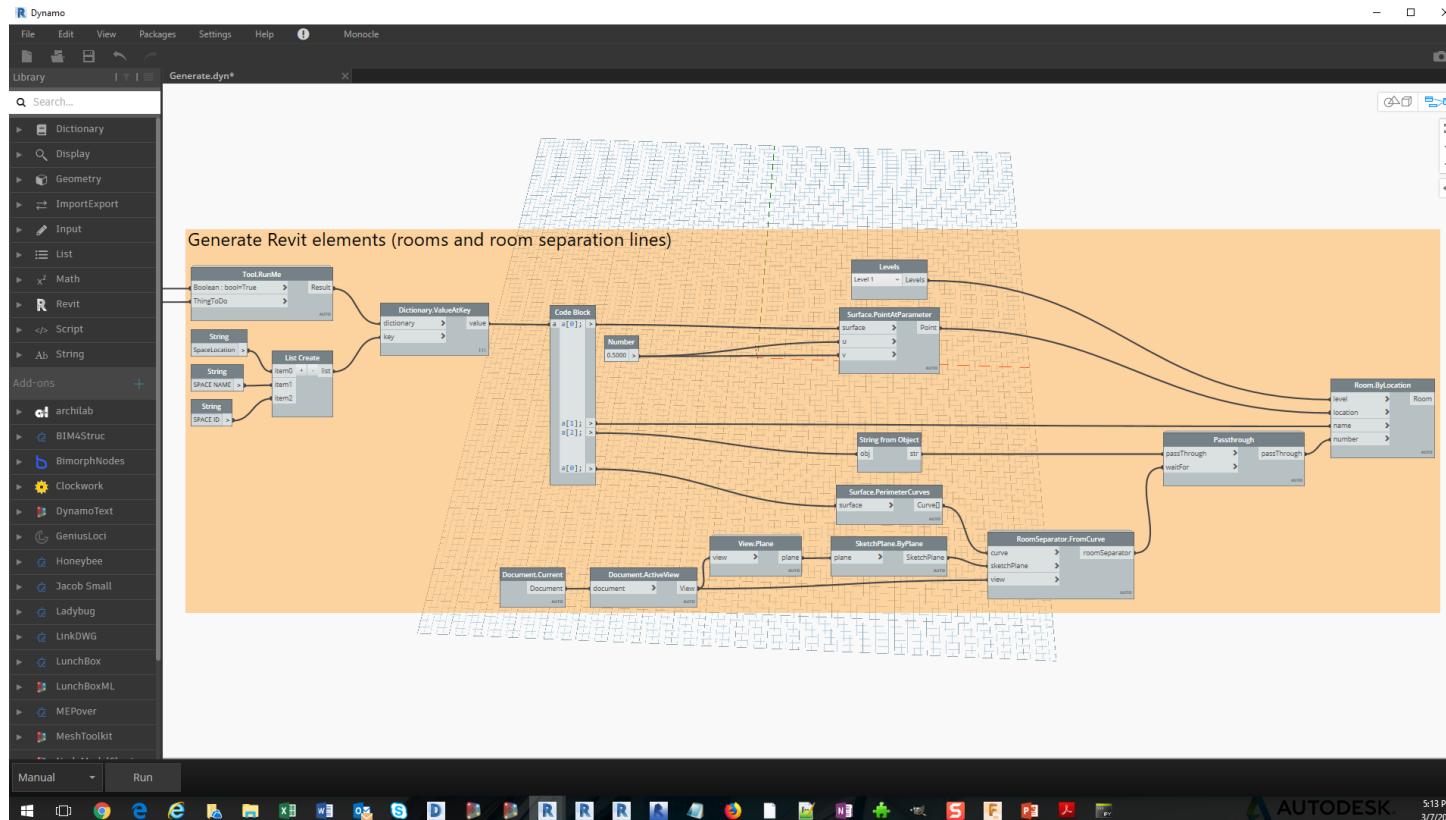
# Space Planning Process

## Evaluate noise spill



# Space Planning Process

## Set up for Revit Elements



# Space Planning Process

Test all aspects, flex the file and save the graph.



Dynamo

File Edit View Packages Settings Help Monocle

Library

Generate.dyn

Search...

- ImportExport
- Input
- List
- x<sup>2</sup> Math
- Revit
- Script
- String

Add-ons +

- archilab
- BIM4Struc
- BimorphNodes
- Clockwork
- DynamoText

GeniusLoc

- CompoundStructure
- Document
- Element
- Export
- Geometry
- Graphics
- Import
- Level
- Logo
- Material
- Parameter
- Phase

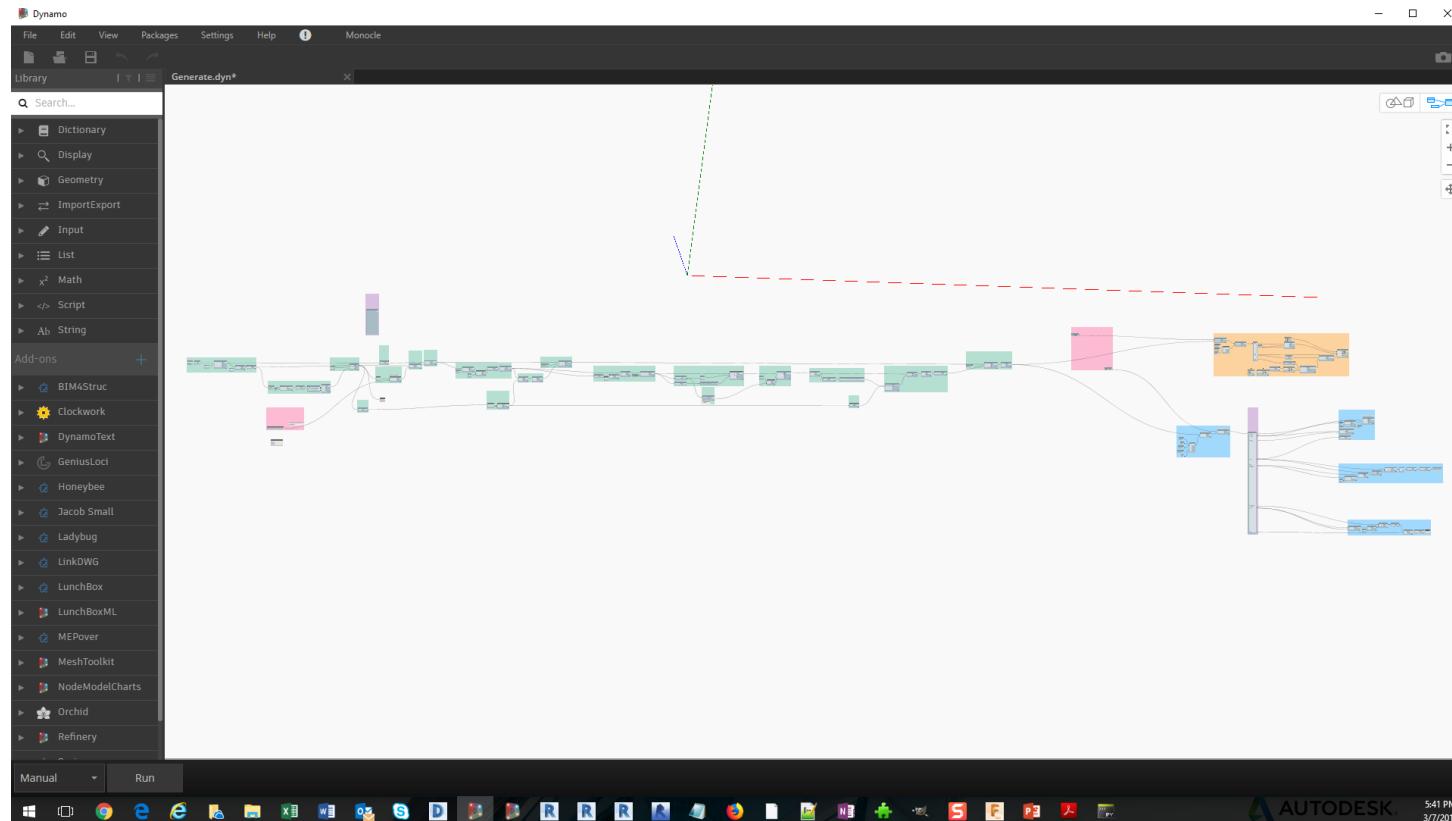
Manual Run

Run completed with warnings.

A screenshot of the Dynamo interface, a node-based programming environment for Revit. The main workspace shows a complex graph of nodes and connections on a 3D grid. The graph includes various colored nodes (green, pink, orange, blue) and dashed lines representing paths or boundaries. The interface has a dark theme with a sidebar for library and add-ons. The status bar at the bottom indicates "Run completed with warnings." The taskbar at the bottom shows various application icons, and the system tray on the right shows the Autodesk logo and the date/time.

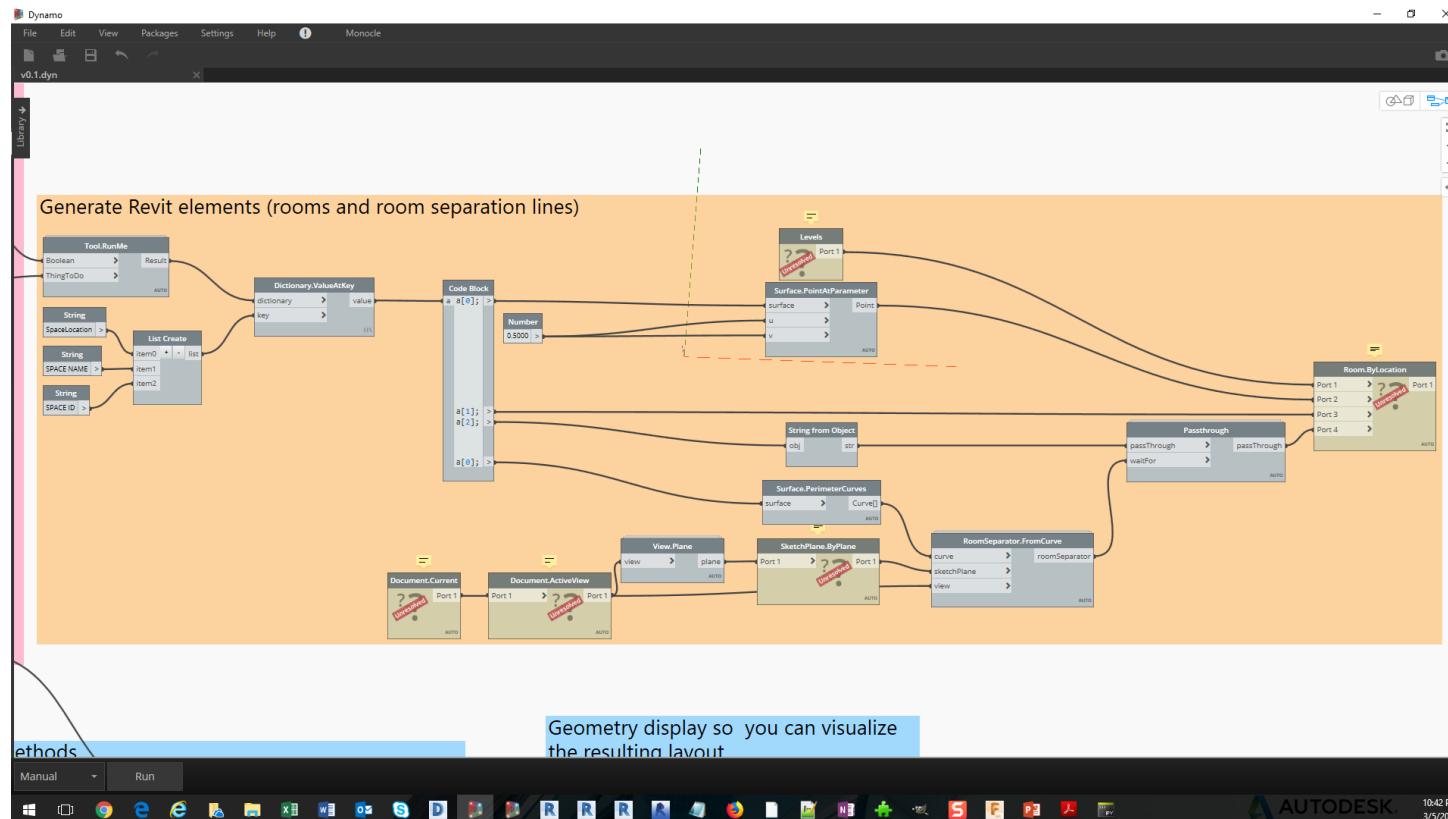
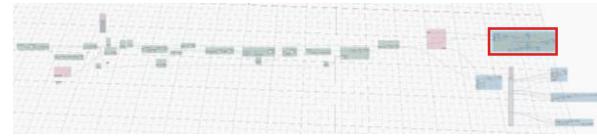
# Space Planning Process

Optionally, move to Dynamo Sandbox



# Space Planning Process

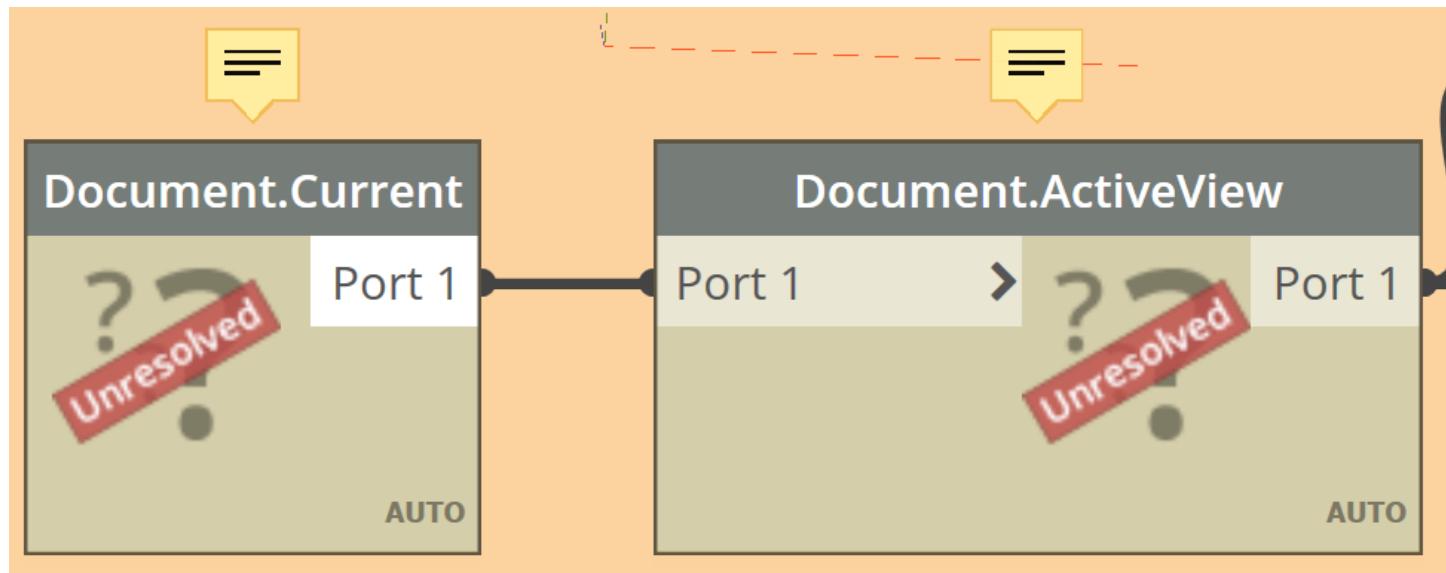
Notice the Revit generation has many “unresolved” warnings.



# Space Planning Process



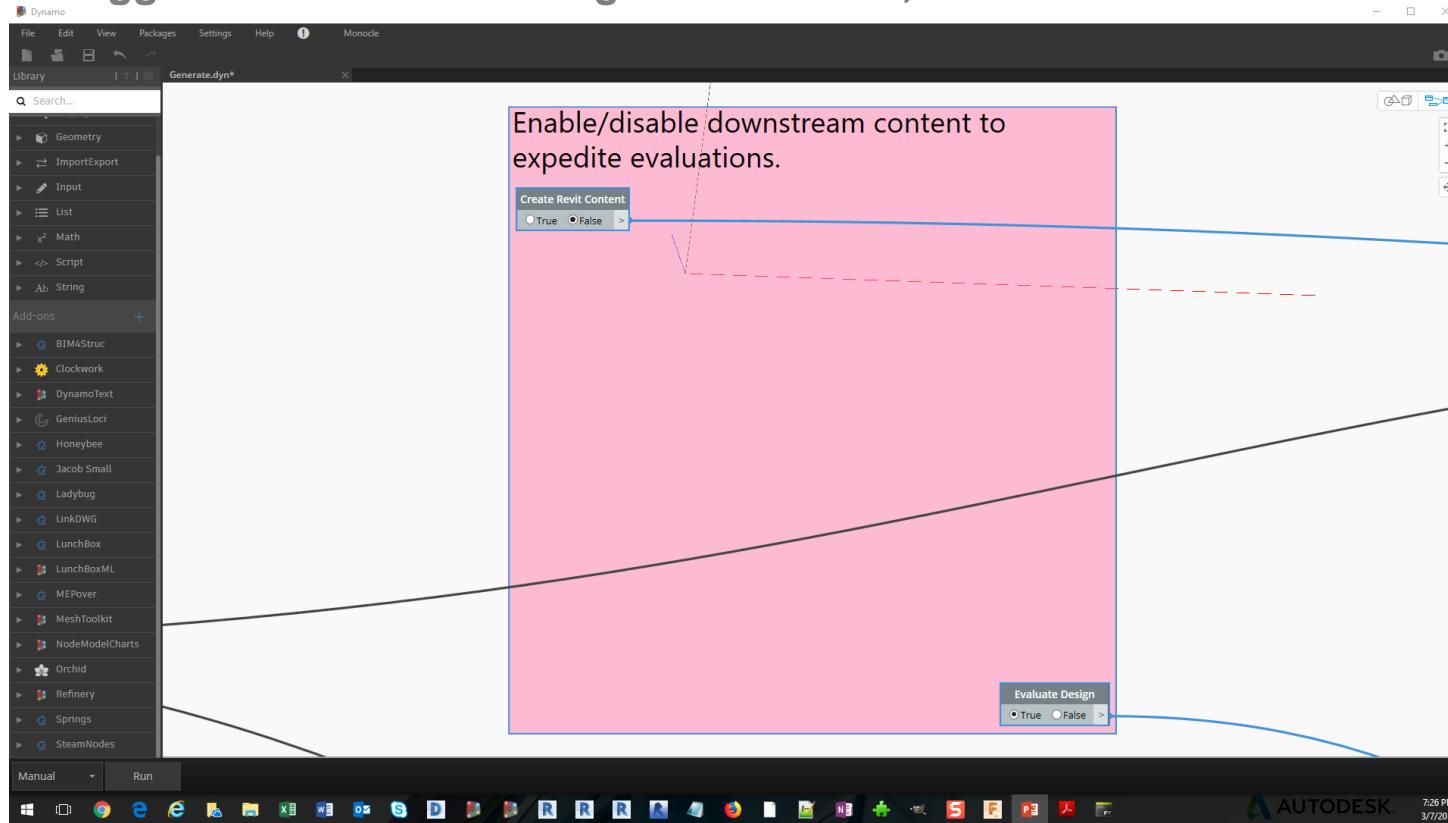
Unresolved nodes are OK here. In fact it's a situation where it's intentional – you'll still want to check to make sure the evaluations work. Warnings result from not running in Dynamo for Revit.



# Space Planning Process

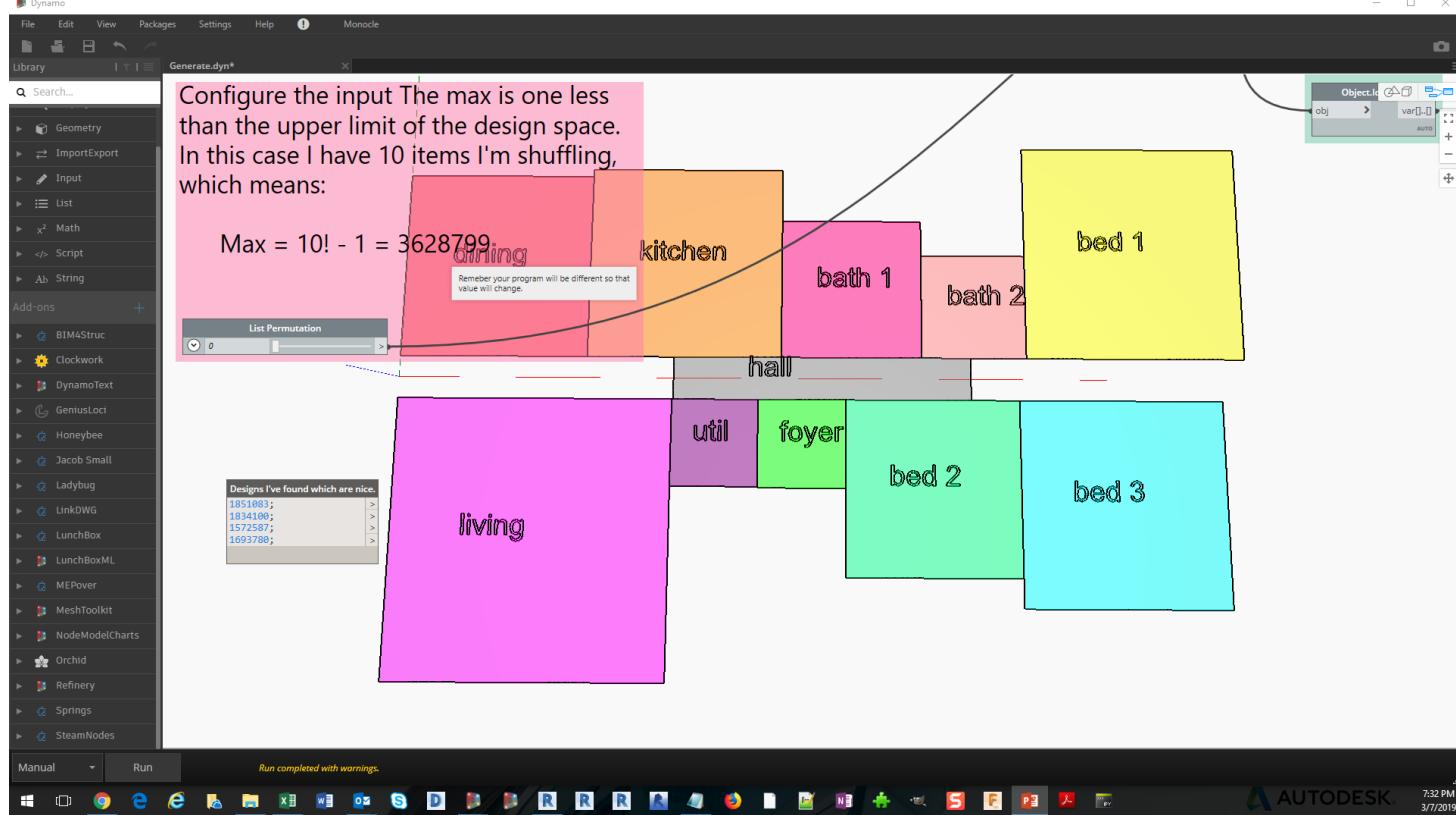


Toggle the triggers – we're not making Revit content, but we want to evaluate things first.



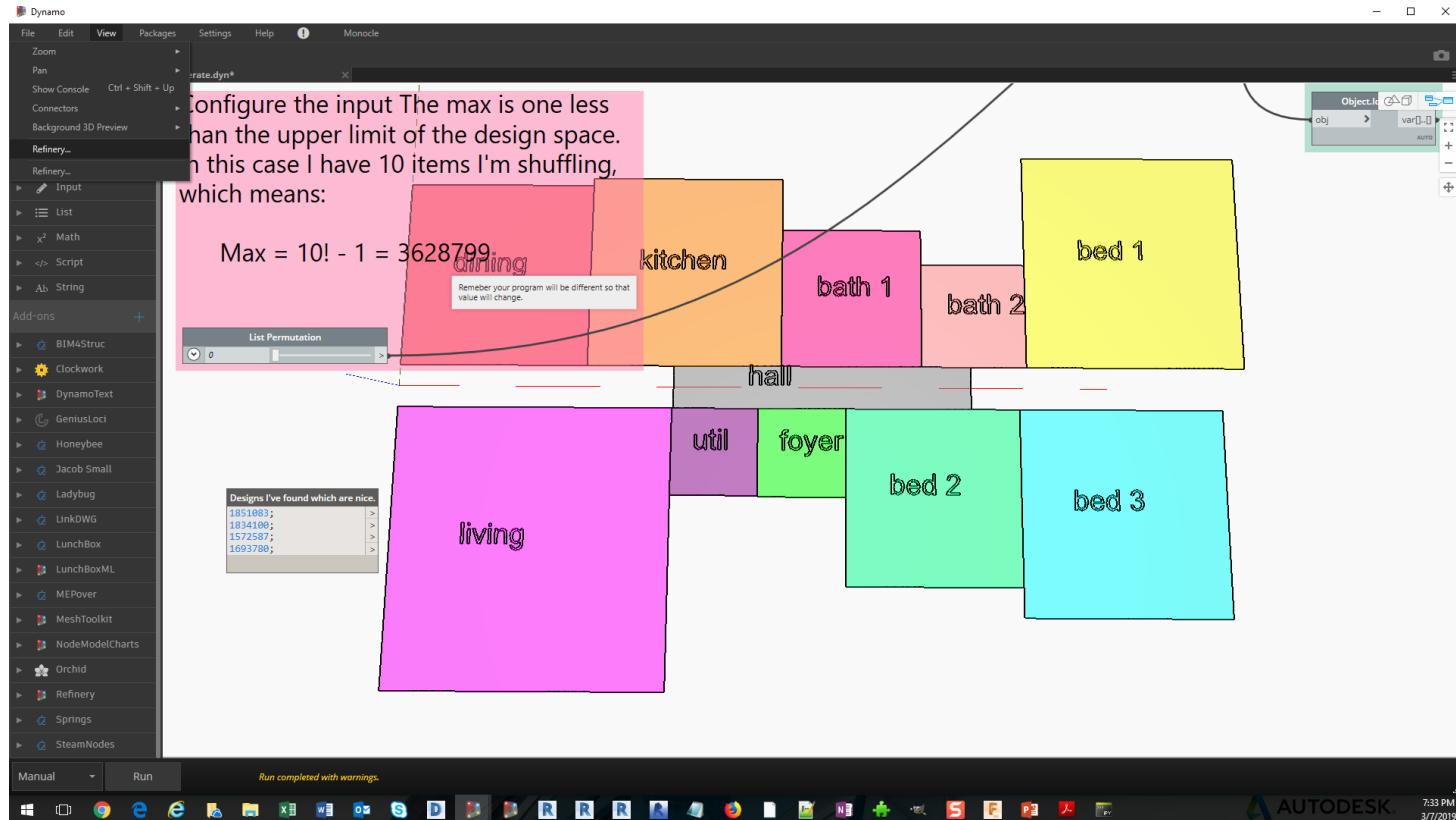
# Space Planning Process

Run the graph, and make sure your background preview is well centered and legible.



# Space Planning Process

## Launch Refinery



# Space Planning Process

## Flex the graph in Dynamo

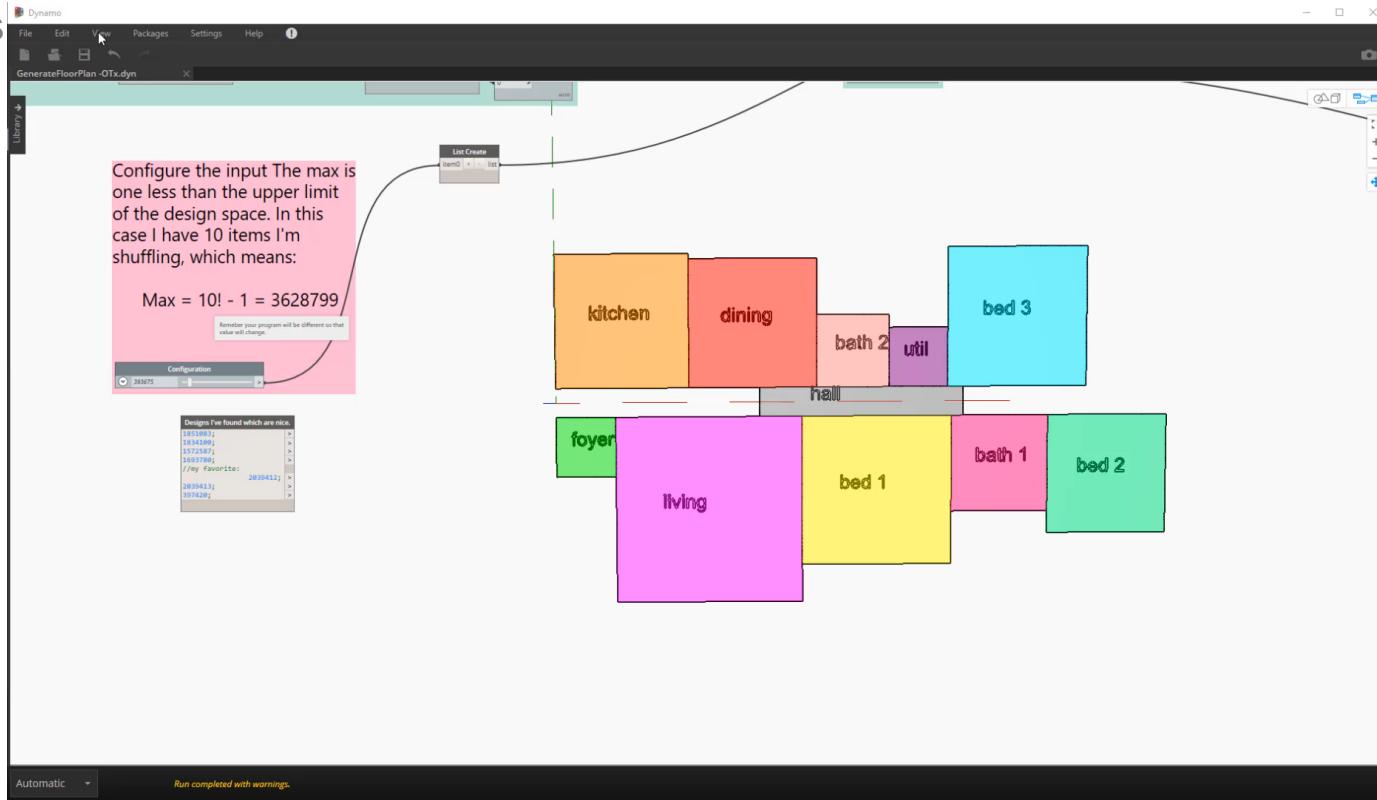
The screenshot shows the Dynamo interface with a graph-based workflow for generating a floor plan. The graph consists of nodes and connections:

- A "List Create" node (item0 + - list) is connected to a curved line that leads to the floor plan.
- The floor plan is a 2D layout of rooms: living (pink), dining (red), kitchen (orange), foyer (green), hall (grey), bed 1 (yellow), bed 2 (teal), bed 3 (cyan), bath 1 (pink), and bath 2 (purple).
- A "Configuration" node contains a dropdown menu set to 2039412 and a slider.
- A text box provides configuration instructions: "Configure the input The max is one less than the upper limit of the design space. In this case I have 10 items I'm shuffling, which means: Max = 10! - 1 = 3628799". It also notes: "Remember your program will be different so that value will change."
- A "Designs I've found which are nice." panel lists several design IDs:

```
1851083;  
1834100;  
1572587;  
1693780;  
//my favorite:  
2039412;  
397420;
```
- At the bottom, status bars indicate "Automatic" and "Run completed with warnings."
- Performance metrics are displayed on the right: "Noise Rating: 14.000000" and "Proximity Score: 0.000000".

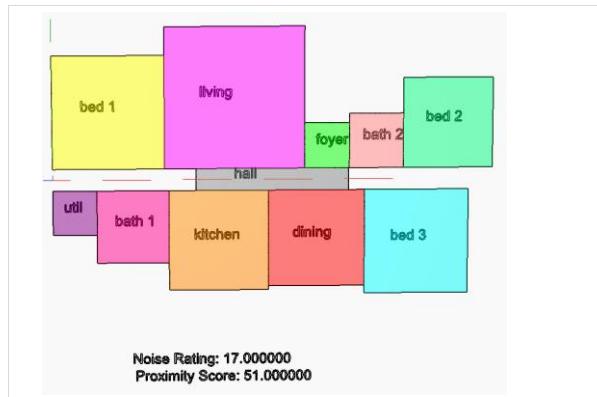
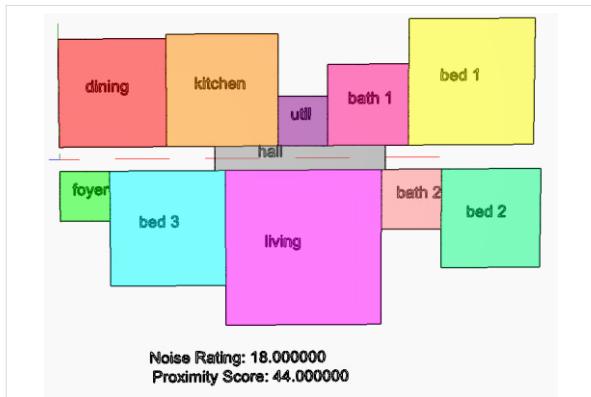
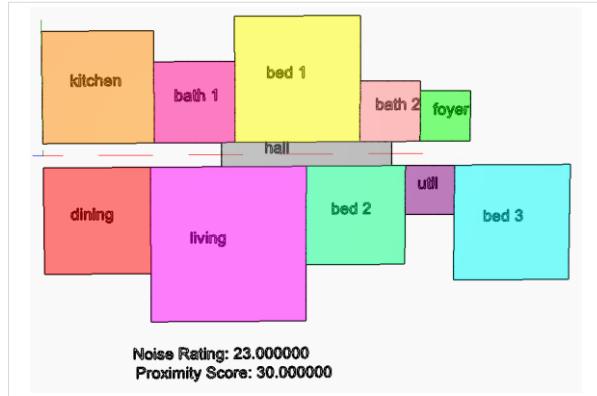
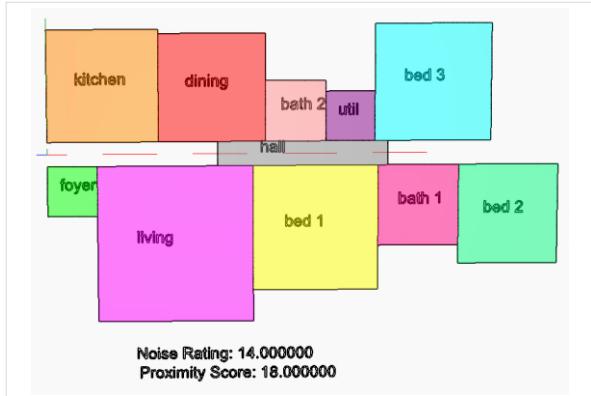
# Space Planning Process

40 random runs  
20 seconds



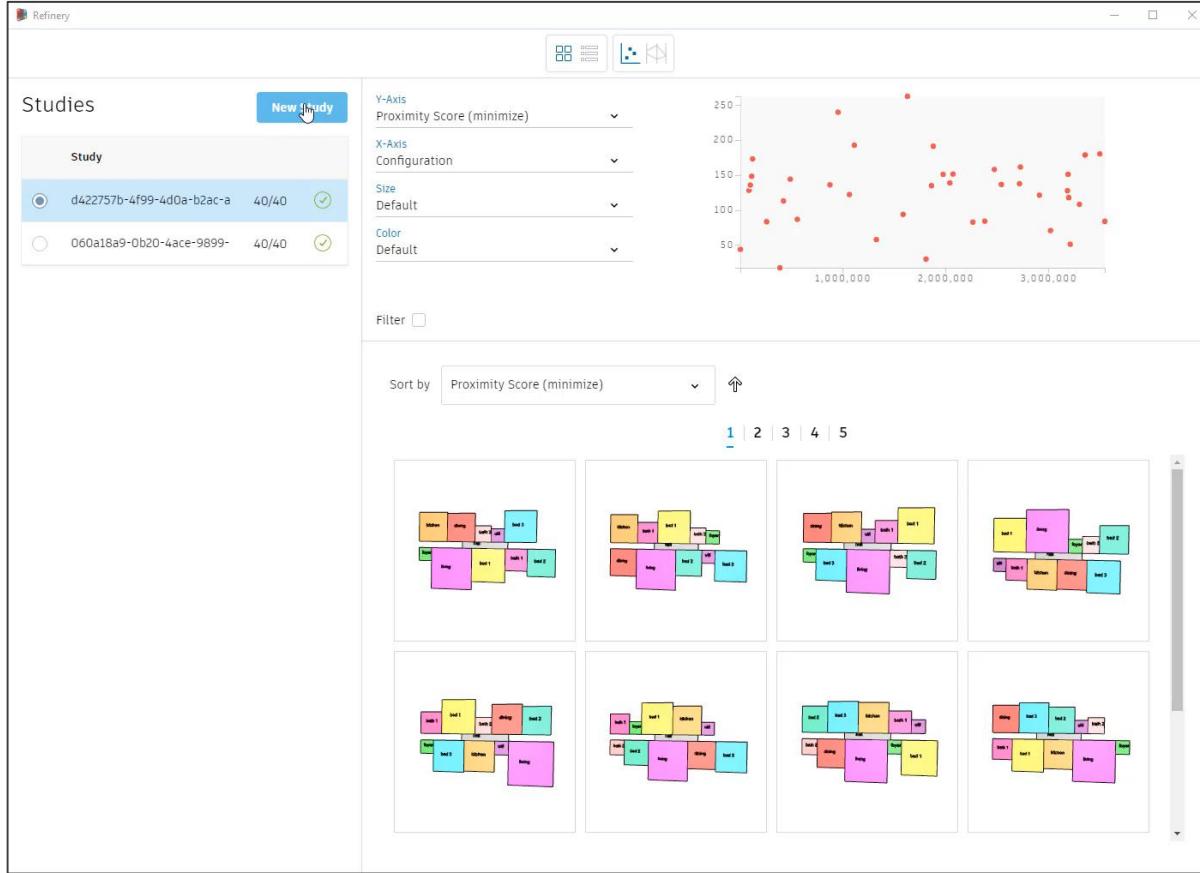
# Space Planning Process

Refinery: 4 best of the 40 random runs



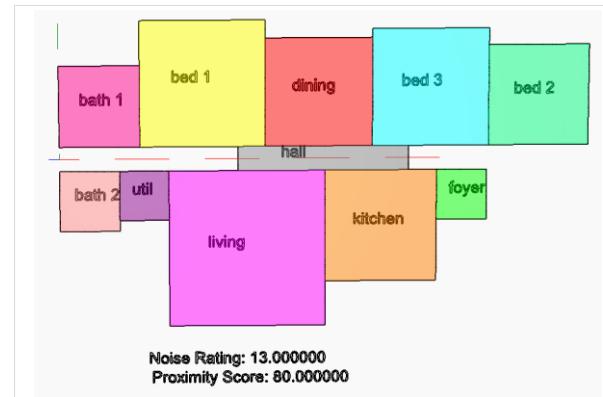
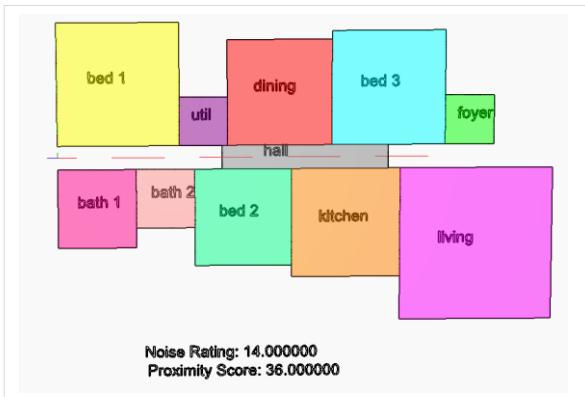
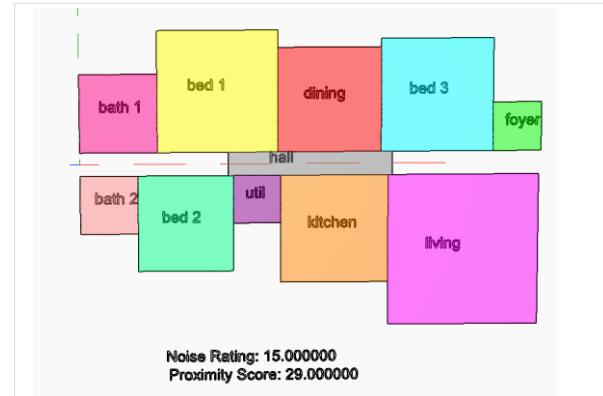
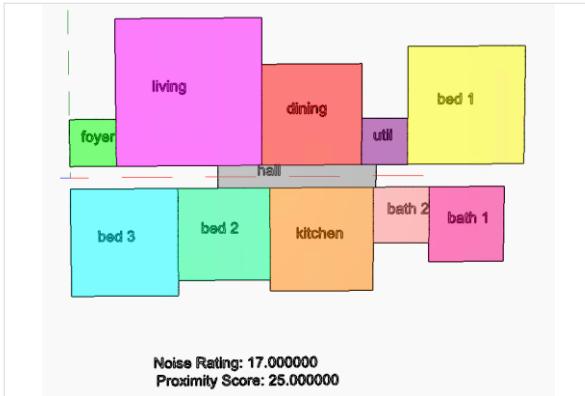
# Space Planning Process

20x100 optimization  
5 minutes



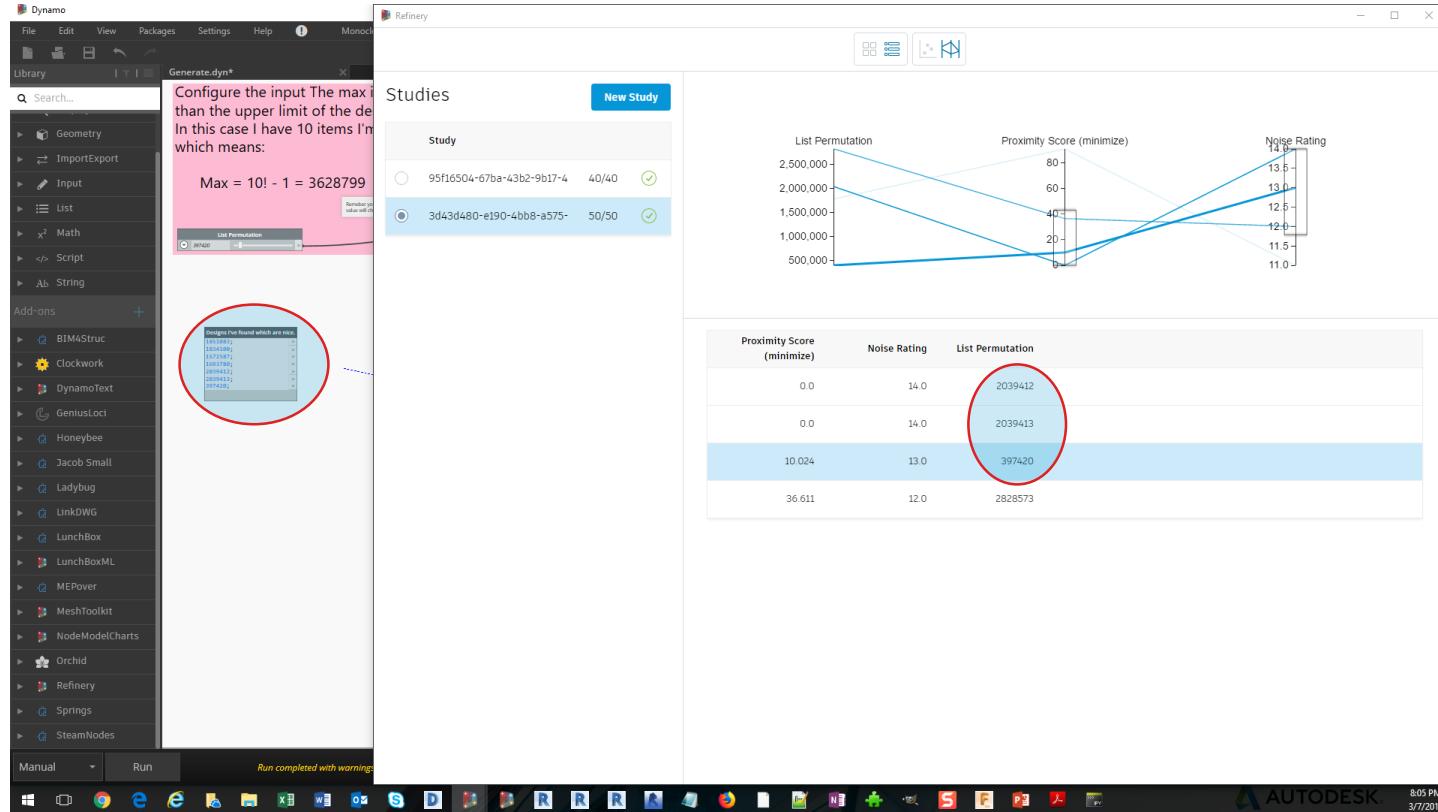
# Space Planning Process

Refinery: 4 best of the optimization runs



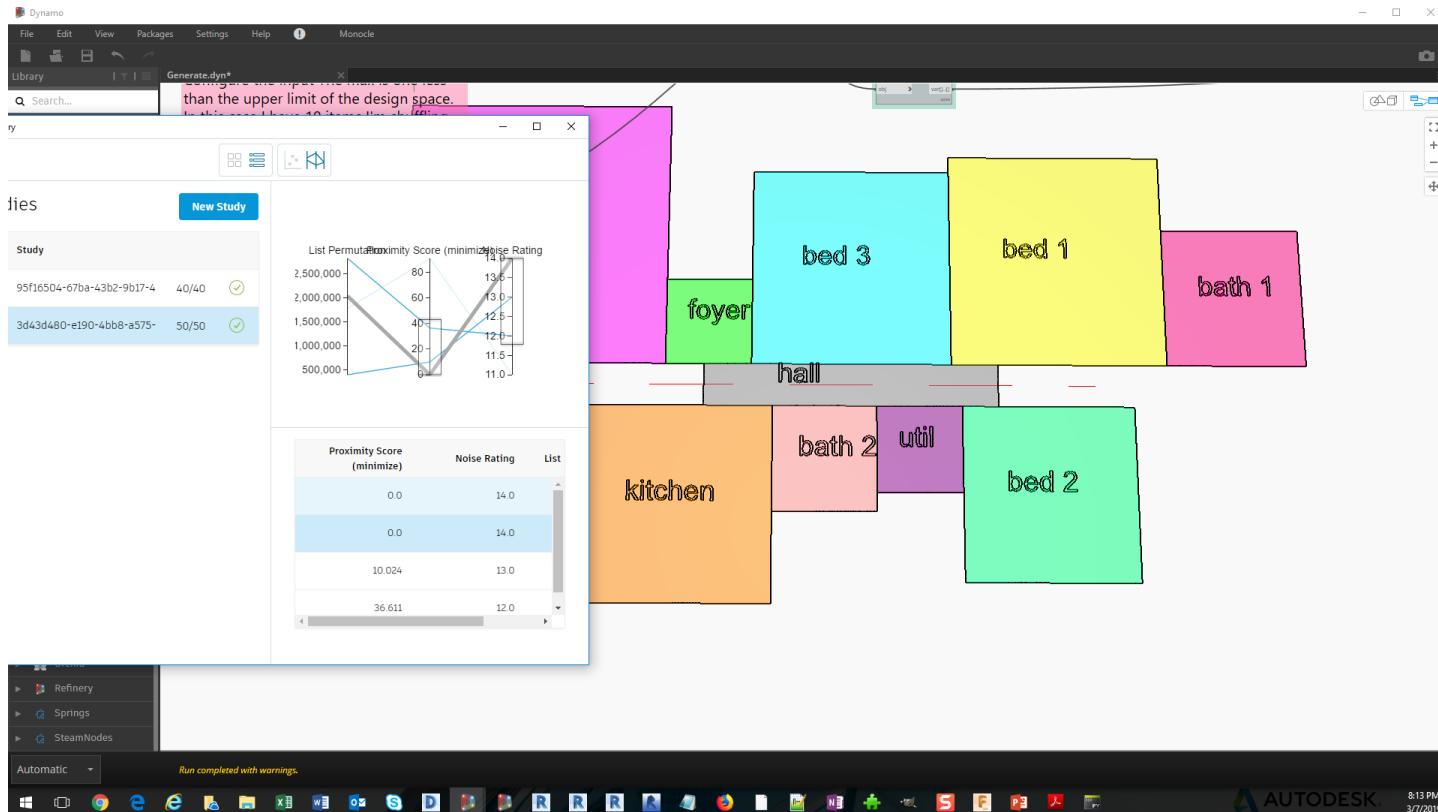
# Space Planning Process

Note the permutation numbers you like and document them in the graph.



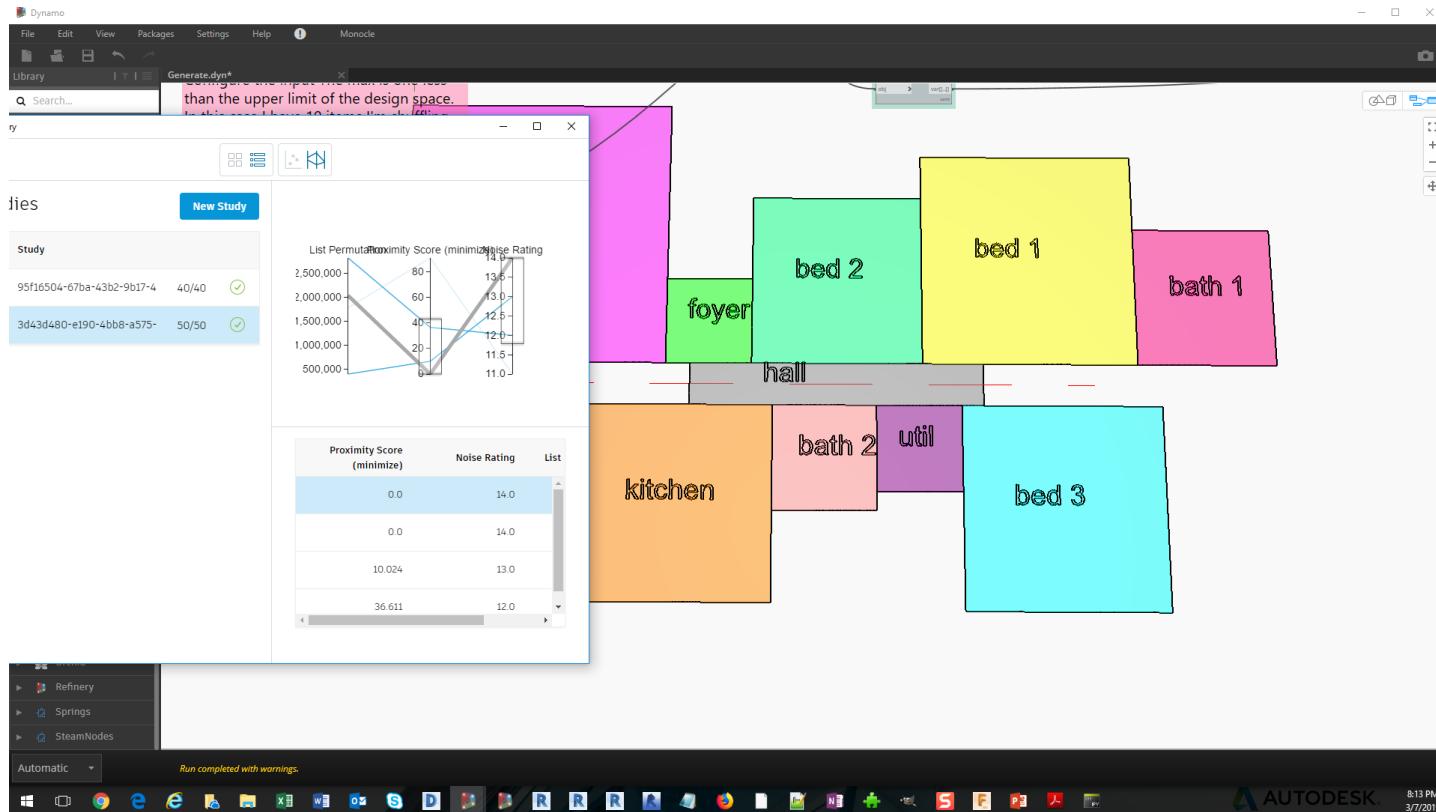
# Space Planning Process

Selecting a design in Refinery will change sliders in Dynamo (auto run mode can help)



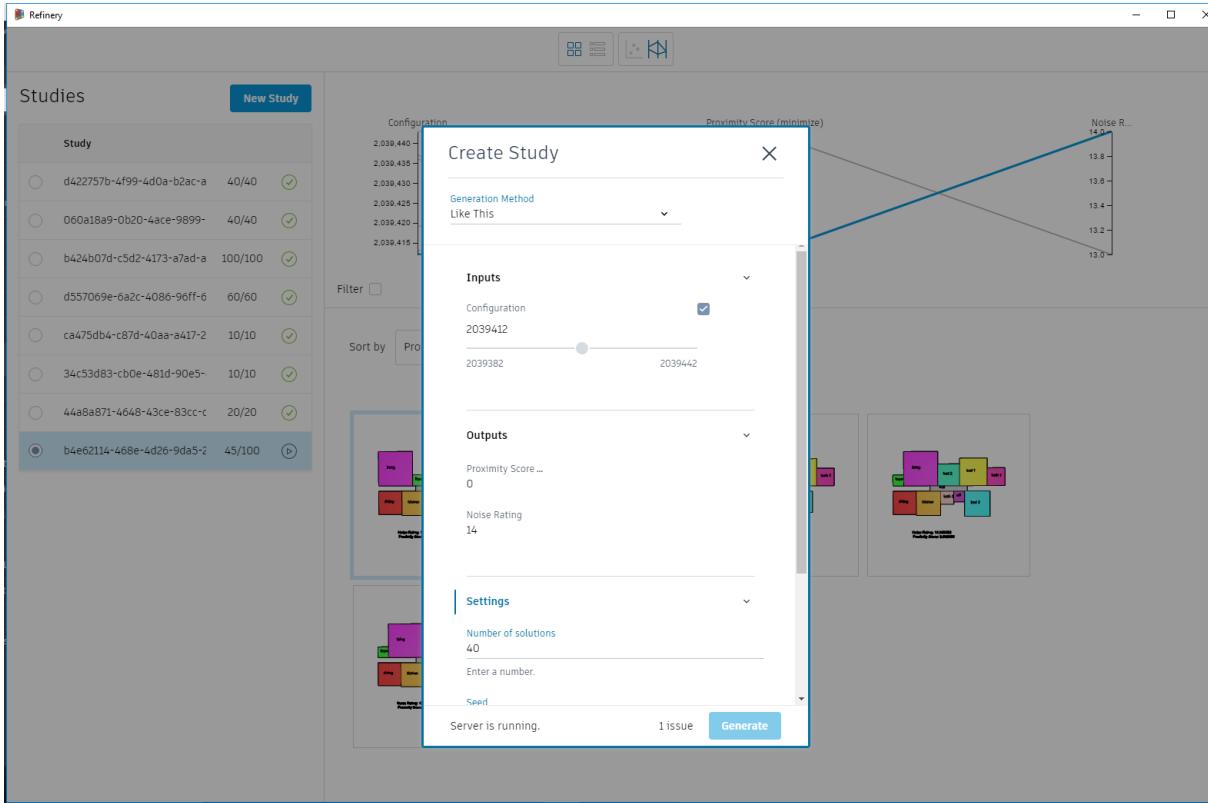
# Space Planning Process

Selecting a design in Refinery will change sliders in Dynamo (auto run mode can help)



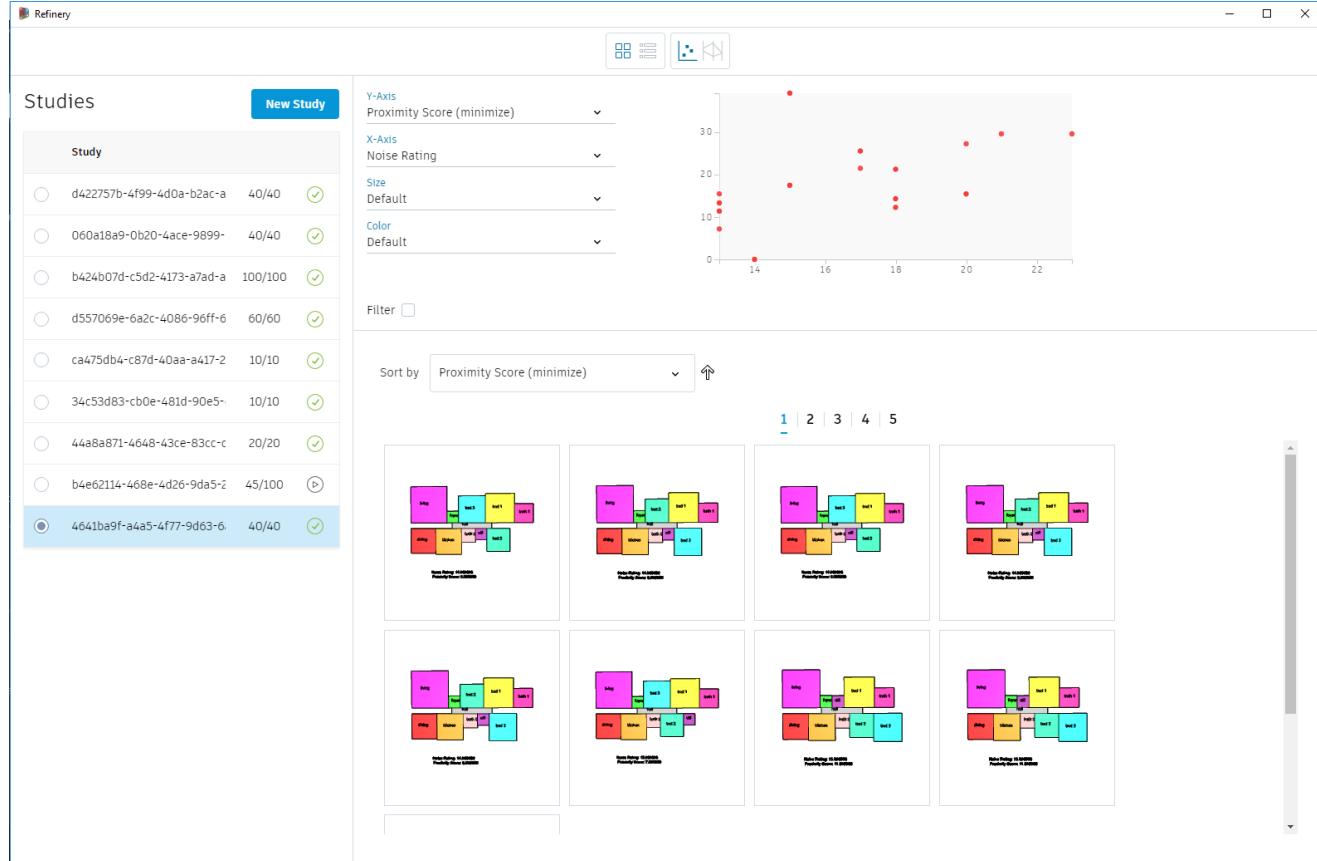
# Space Planning Process

## Refinery: Like this favorite



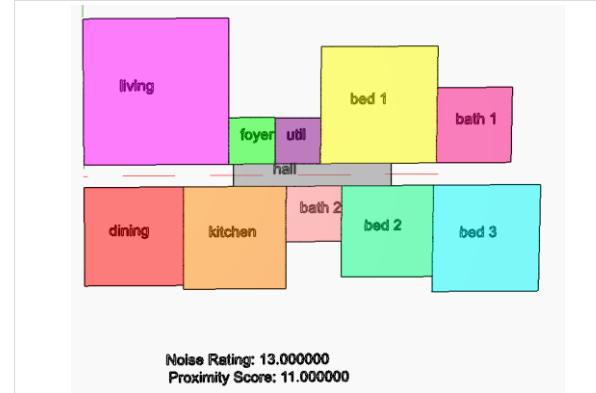
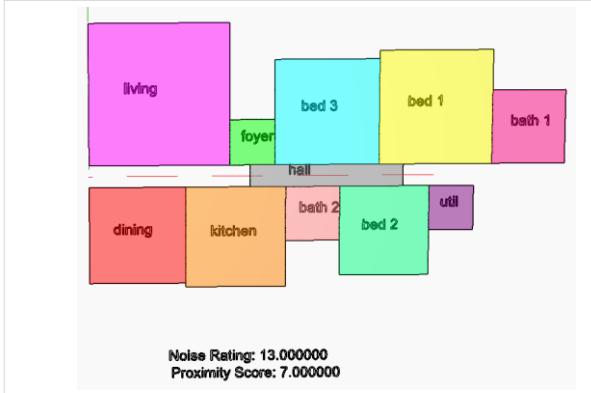
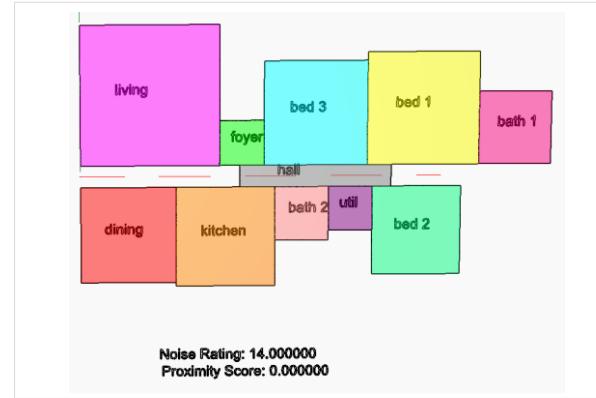
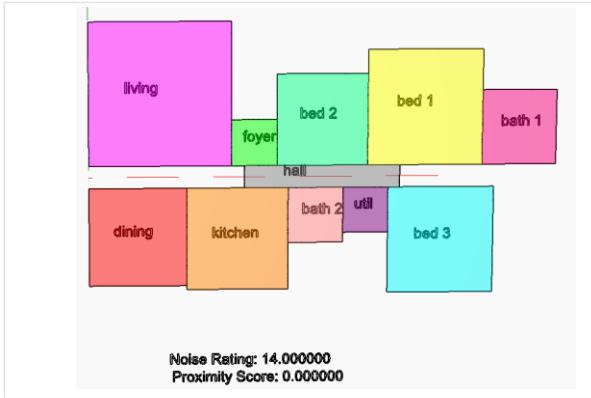
# Space Planning Process

Like This Results  
40 runs @ 20 sec



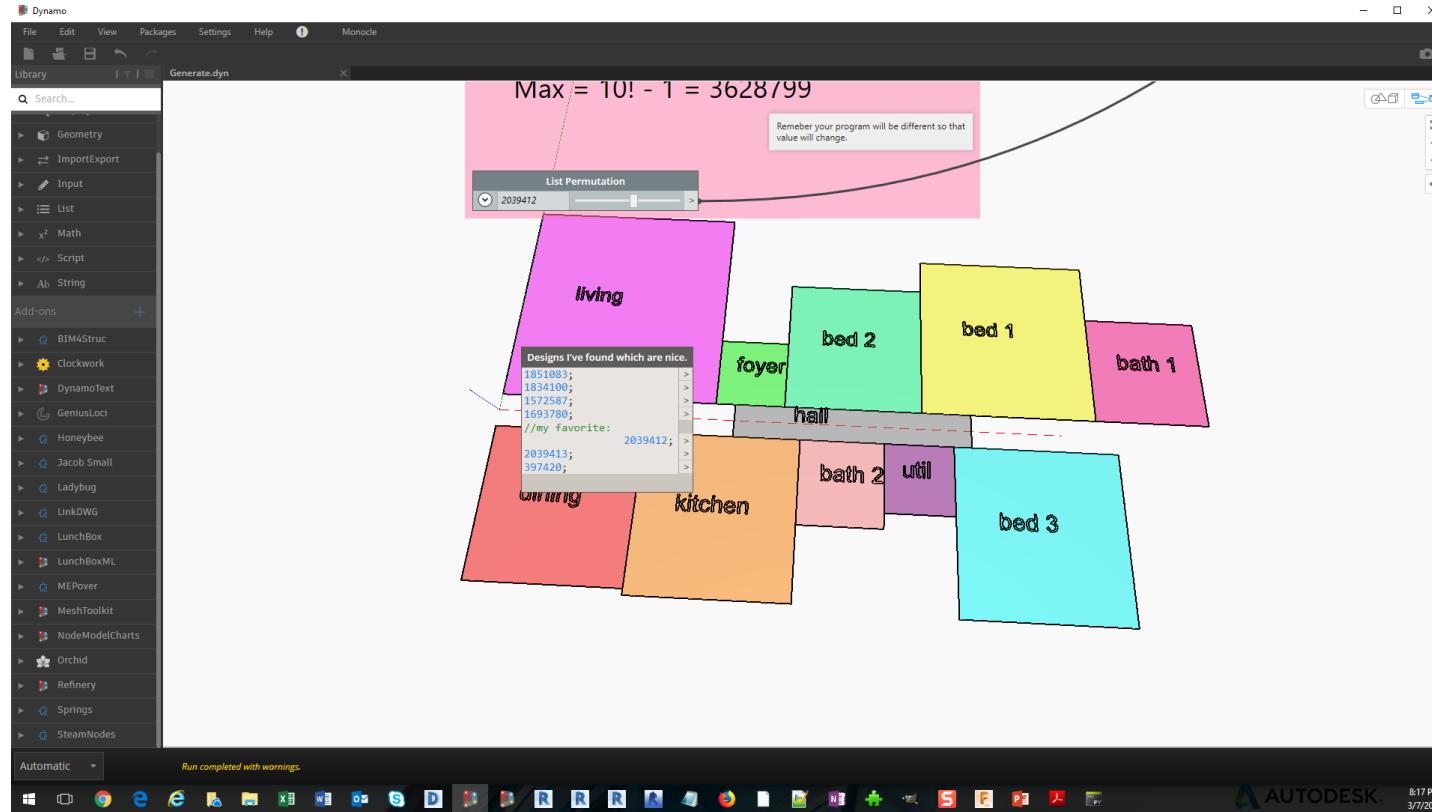
# Space Planning Process

Like This Results  
4 top results



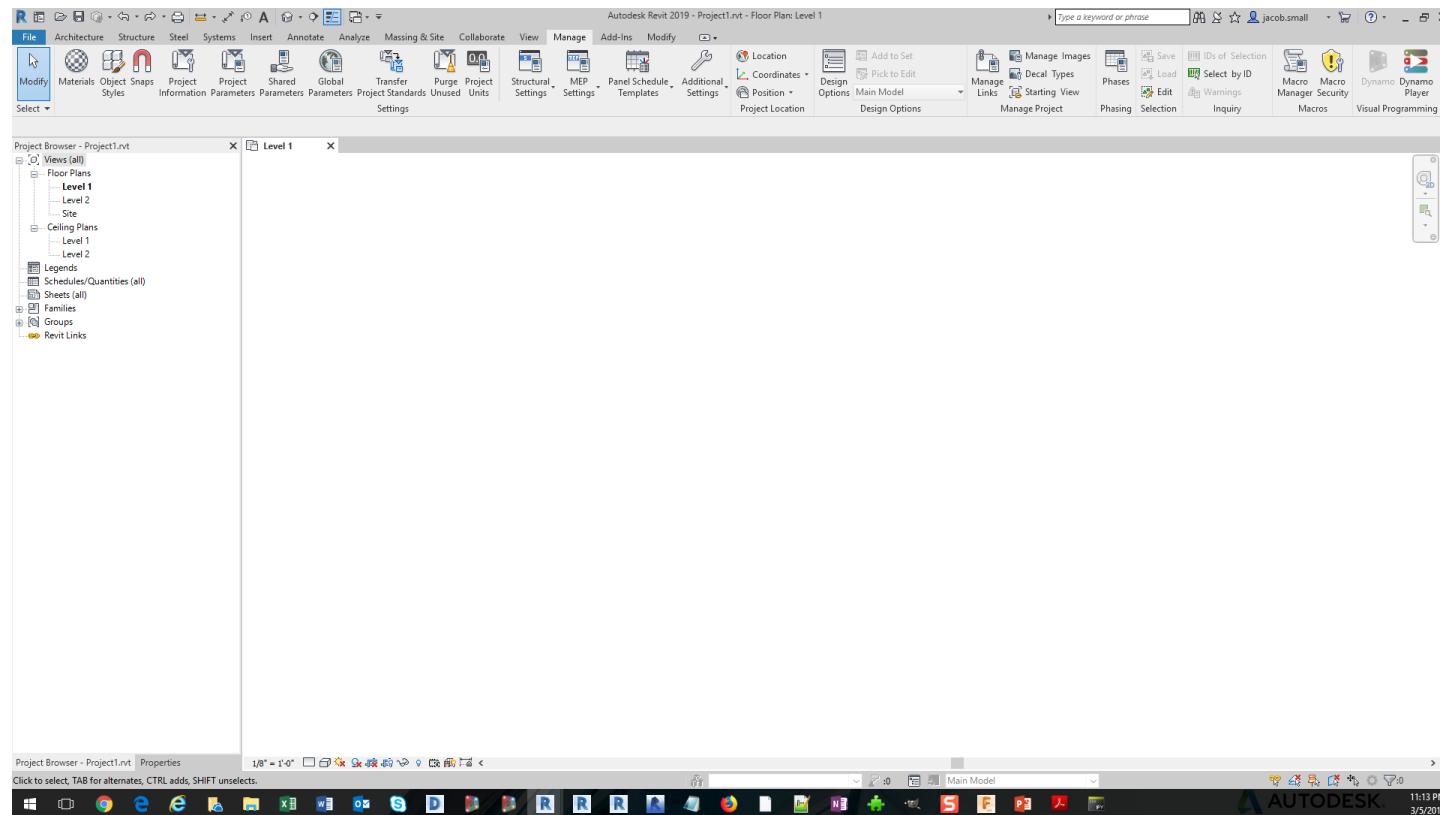
# Space Planning Process

Save the graph when you're happy with it



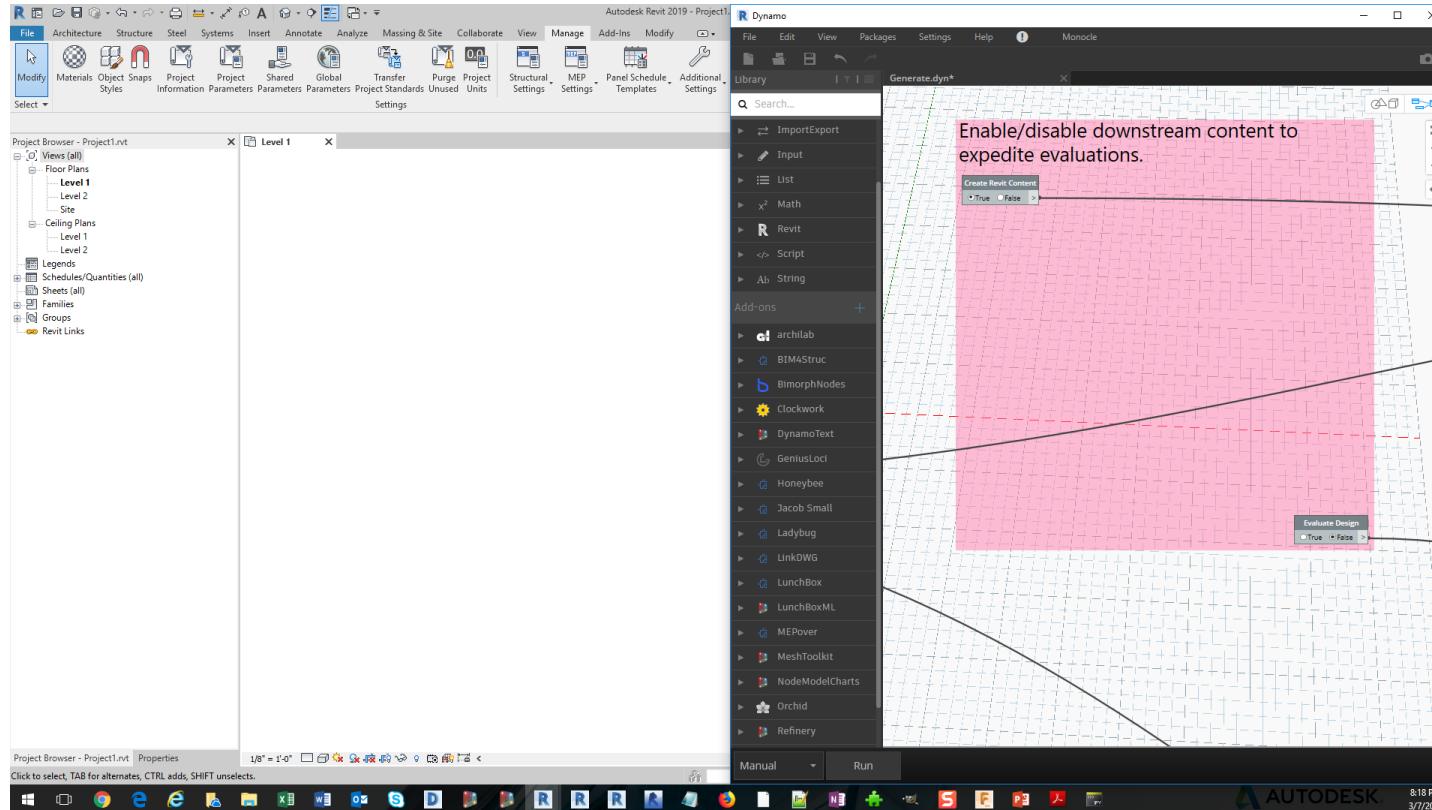
# Space Planning Process

## Move to Revit!



# Space Planning Process

Toggle the triggers – we want to create Revit content, but don't need to evaluate.



# Space Planning Process

Verify the favorite, and run the graph.

The screenshot shows the Autodesk Revit 2019 interface on the left and the Dynamo interface on the right, integrated within the Revit environment.

**Revit Project Browser:** Shows a hierarchy of views: Views (all), Floor Plans, Level 1, Level 2, Site; Ceiling Plans, Level 1, Level 2; Legends; Schedules/Quantities (all); Sheets (all); Families; Groups; Revit Links.

**Dynamo Script:** The script is titled "Generate.dyn". It includes the following code:

```
Max = 10! - 1 = 3628799
Remember your program will be different so that value will change.

List Permutation
2039412 >

Designs I've found which are nice.
1851083; >
1834108; >
1834174; >
1693788; >
//my favorite: 2039412; >
2039413; >
397428; >
```

A pink callout box highlights the text: "Configure the input The max is one less than the upper limit of the design space. In this case I have 10 items I'm shuffling, which means:

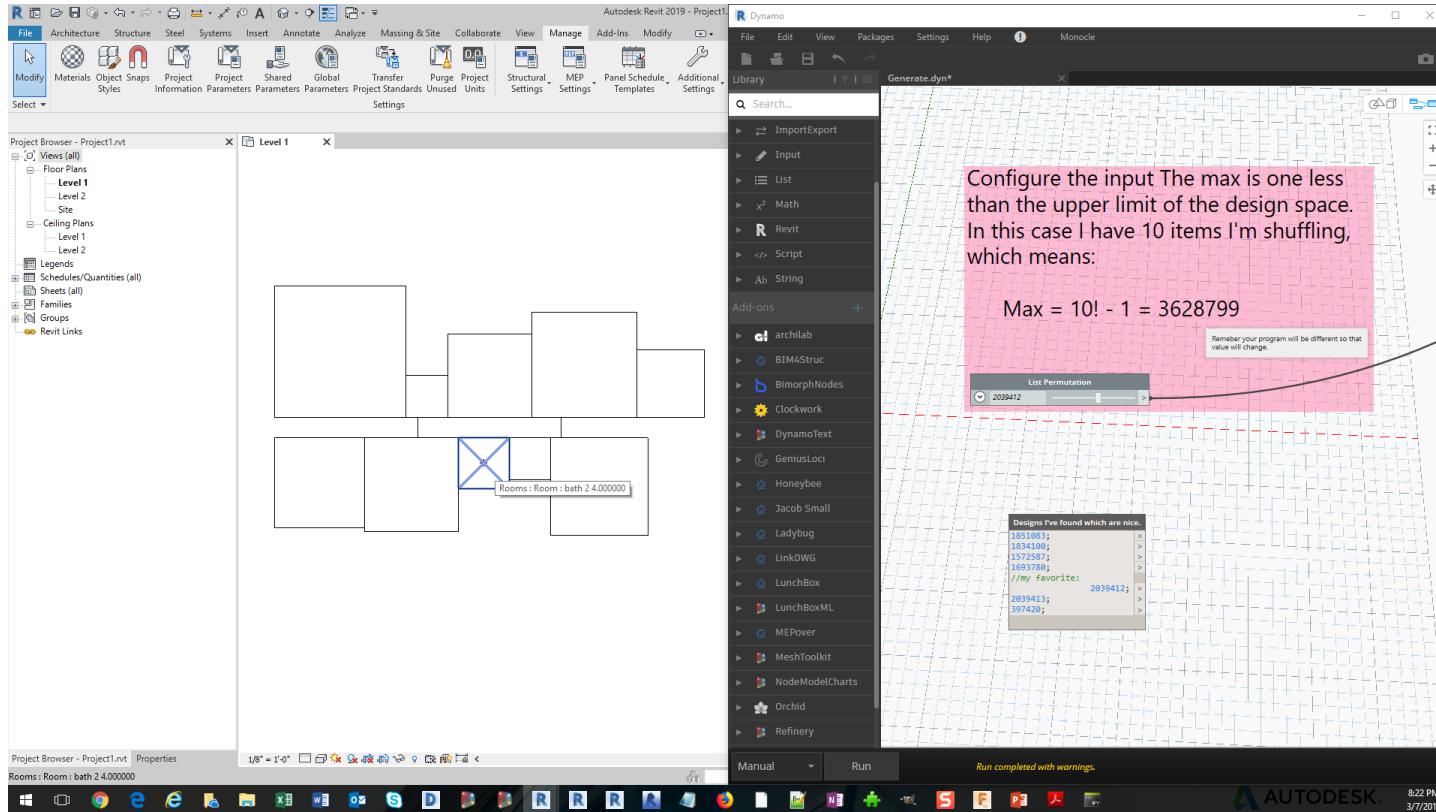
Max =  $10! - 1 = 3628799$

Remember your program will be different so that value will change.

The bottom status bar shows: Project Browser - Project1.rvt | Properties | 1/8' = 1'-0" | Click to select: TAB for alternates, CTRL adds, SHIFT unselects. The taskbar at the bottom includes icons for various applications like Microsoft Edge, File Explorer, and Autodesk products.

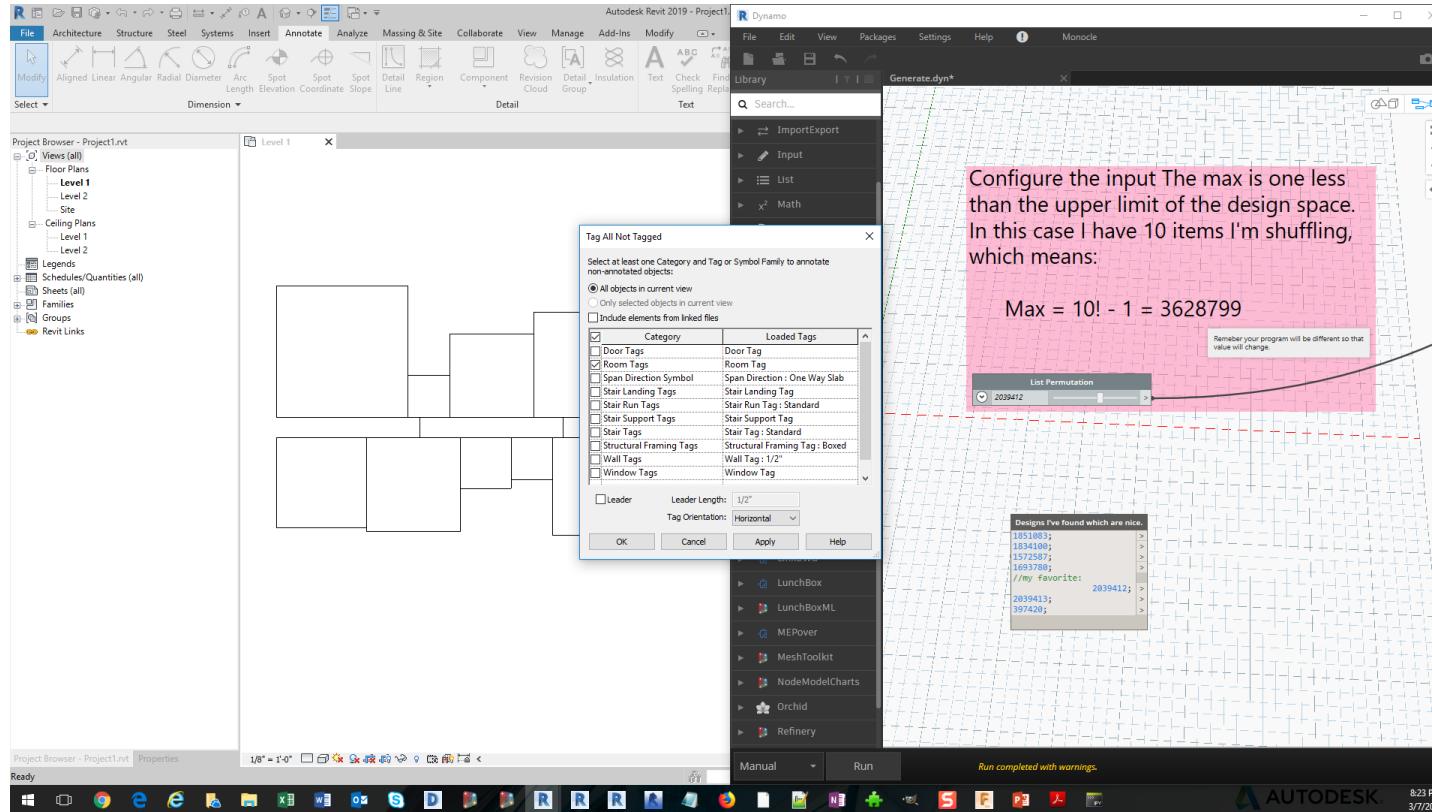
# Space Planning Process

Revit content is created as it would had you done it by hand.



# Space Planning Process

Update and document as you usually would – change at will – you can always go back



# Space Planning Process

Update and document as you usually would – change at will – you can always go back

The screenshot shows a dual-pane interface. On the left is the Autodesk Revit 2019 interface, displaying a floor plan of a house with rooms labeled: living (6), foyer (8), hall, dining (1), kitchen (2), bath 2 (4), util (7), bed 1 (5), bed 2 (9), bed 3 (10), and path 1 (3). The Revit ribbon menu is visible at the top. On the right is the Autodesk Dynamo interface, showing a grid-based workspace. A pink callout box contains the text: "Configure the input The max is one less than the upper limit of the design space. In this case I have 10 items I'm shuffling, which means: Max = 10! - 1 = 3628799". Below this is a code editor window titled "List Permutation" with the value "2039412" highlighted. A note says "Remember your program will be different so that value will change." At the bottom of the workspace, a list box displays "Designs I've found which are nice." with several IDs listed: 1851083; 1834108; 1834107; 1693780; //my favorite: 2039412; 2039413; 397420;.

# Space Planning Process

Update and document as you usually would – change at will – you can always go back

The image shows a dual-screen setup. On the left, the Autodesk Revit 2019 interface displays a floor plan for 'Level 1'. The plan includes rooms labeled 'living' (area 6), 'foyer' (area 8), 'hall' (area 0), 'dining' (area 1), 'kitchen' (area 2), 'bed 1' (area 5), 'bed 2' (area 9), 'bed 3' (area 10), 'bath 1' (area 3), 'bath 2' (area 4), and 'util' (area 7). On the right, the Autodesk Dynamo interface shows a 'List Permutation' node with the value '2039412'. A pink callout box contains the text: 'Configure the input The max is one less than the upper limit of the design space. In this case I have 10 items I'm shuffling, which means: Max = 10! - 1 = 3628799'. Below the node, a message says 'Remember your program will be different so that value will change.' A scrollable list titled 'Designs I've Found which are nice.' shows several item IDs: 1851083; 1834108; 1834107; 1693780; //my favorite: 2039412; 2039413; 397428;.

# Space Planning Process

Once you're done with the updates, consider evaluating your changes

The screenshot shows the Autodesk Revit 2019 interface with a floor plan of a house. The floor plan includes rooms labeled living (13), dining (1), foyer (6), kitchen (2), hall (12), bath 2 (4), util (7), bed 1 (5), bed 2 (9), bed 3 (11), and bath 1 (3). The Revit ribbon is visible at the top, showing tabs like File, Architecture, Structure, Steel, Systems, Insert, Annotate, Analyze, Massing & Site, Collaborate, View, Manage, Add-Ins, and Modify.

The right side of the screen shows the Autodesk Dynamo interface. A pink callout box highlights a note: "Configure the input The max is one less than the upper limit of the design space. In this case I have 10 items I'm shuffling, which means: Max = 10! - 1 = 3628799". Below this, a message says "Remember your program will be different so that value will change." A slider labeled "List Permutation" has the value "2039412" selected. A list titled "Designs I've found which are nice." shows several item IDs: 1851083; 1834108; 1834107; 1693780; //my favorite: 2039412; 2039413; 397420;.

At the bottom, the Windows taskbar is visible with various application icons, and the Autodesk logo is in the bottom right corner.

# Space Planning Process

Once you're done with the updates, consider evaluating your changes

The image shows a dual-screen setup for architectural design and data analysis.

**Autodesk Revit 2019 - Project1.rvt:** The left screen displays a floor plan for "Level 1". The rooms and their assigned room numbers are:

- living (6)
- foyer (8)
- hall (0)
- bed 2 (9)
- bed 1 (5)
- bath 1 (3)
- dining (1)
- kitchen (2)
- bath 2 (4)
- util (7)
- bed 3 (10)

**Dynamo - Evaluate Modifications.dyn\***: The right screen shows a complex data flow diagram. It starts with a "Get the content from Revit" node, which feeds into a "Gather data for evaluation methods" node. This is followed by a "Find the distance from each room to the required adjacent room per the excel spreadsheet" node. The diagram also includes nodes for "Bring in the Program", "Evaluate any area averages", and "Evaluate for Noise". A legend on the left lists various add-ons used in the script, such as archilab, BIM4Struct, BimorphNodes, Clockwork, DynamoText, GeniusLoc, Honeybee, Jacob Small, Ladybug, LinkDWG, LunchBox, LunchBoxML, MEPower, MeshToolkit, NodeModelCharts, Orchid, and Refinery.

Project Browser - Project1.rvt | Properties      1/8 = 1'0"      Click to select; TAB for alternates, CTRL adds; SHIFT unselects.

File Architecture Structure Steel Systems Insert Annotate Analyze Massing & Site Collaborate View Manage Add-Ins Modify

Autodesk Revit 2019 - Project1.rvt

File Edit View Packages Settings Help Monocle

Library Search... ImportExport Input List Math Revit Script String

Add-ons archilab BIM4Struct BimorphNodes Clockwork DynamoText GeniusLoc Honeybee Jacob Small Ladybug LinkDWG LunchBox LunchBoxML MEPower MeshToolkit NodeModelCharts Orchid Refinery

Manual Run

AUTODESK 8:34 PM 3/7/2019

# Space Planning Process

Evaluate the previous results to ensure it's not 'worse than it was before'

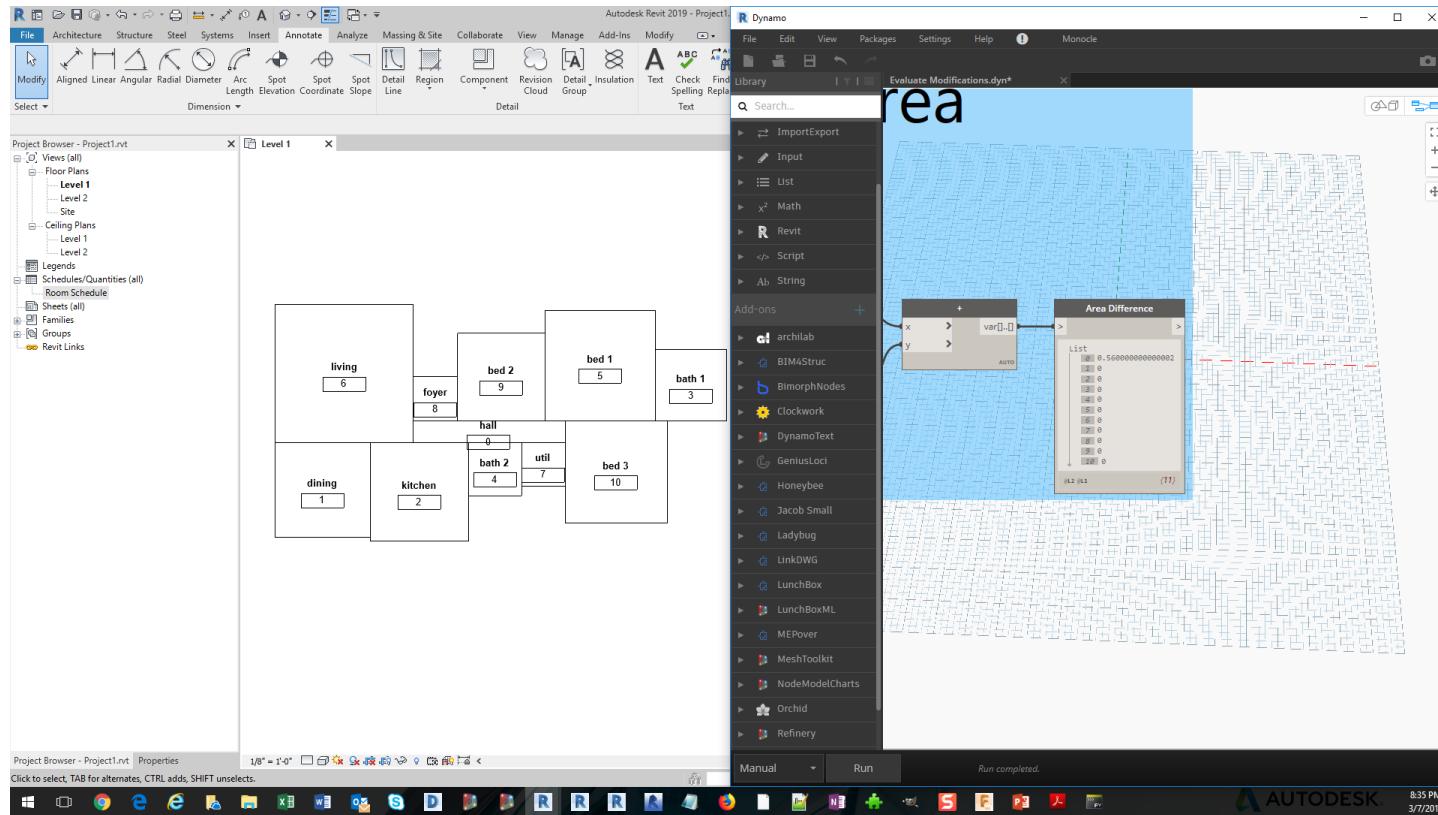
The image shows a dual-screen setup for architectural space planning. On the left screen, the Autodesk Revit 2019 interface is displayed, showing a floor plan of a house with rooms labeled and numbered (living: 6, foyer: 8, hall: 0, dining: 1, kitchen: 2, bed 1: 5, bed 2: 9, bed 3: 10, bath 1: 3, bath 2: 4, util: 7). The Revit ribbon menu is visible at the top. On the right screen, the Dynamo software interface is shown, which is a visual programming environment for BIM. A blue rectangular area on the floor plan represents a specific region being analyzed. A proximity analysis workflow is built in Dynamo, consisting of the following steps:

```
graph LR; CB["Code Block<br>X: 0"] --> MS1["Math Sum<br>values: 0<br>sum: 0<br>Auto"]; MS1 --> MS2["Math Sum<br>values: 0<br>sum: 0<br>Auto"]; MS2 --> PS["Proximity Score (minimum)<br>values: 0<br>sum: 0<br>Auto"]; subgraph Region [ ]; direction TB; R1["values: 0<br>sum: 0<br>Auto"] --- R2["values: 0<br>sum: 0<br>Auto"]; R2 --- R3["values: 0<br>sum: 0<br>Auto"]; end; R1 --> MS1; R2 --> MS1; R3 --> MS1; R1 --> MS2; R2 --> MS2; R3 --> MS2; R1 --> PS; R2 --> PS; R3 --> PS;
```

The proximity score is calculated by summing the values of the rooms within the analyzed region and then finding the minimum value among them.

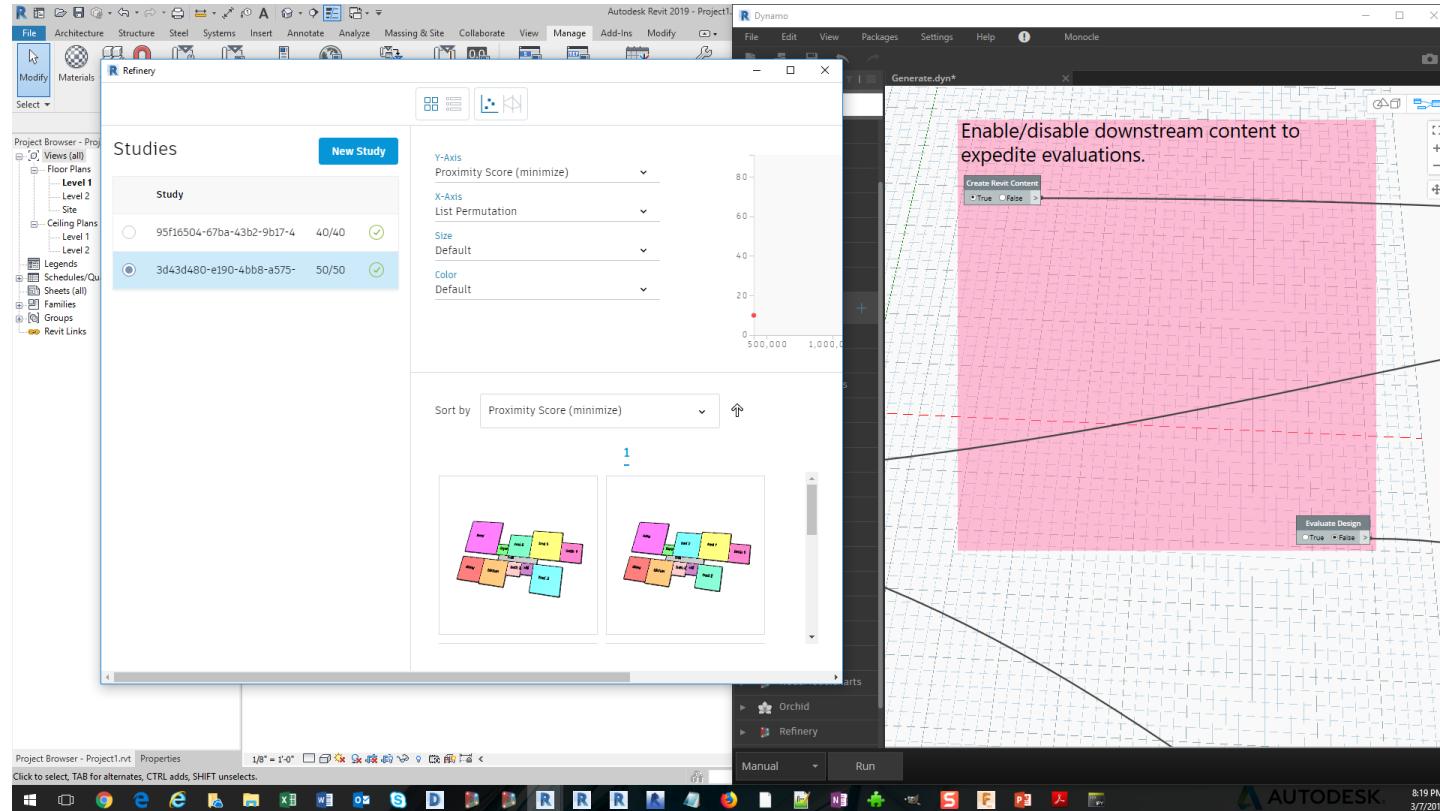
# Space Planning Process

Check for non-compliant geometry alterations as well (areas are a good thing to review)



# Space Planning Process

Still not sure? If you launch Refinery, you'll see that the studies are ready to reuse.



# Additional Learning Resources

- Getting Started with Dynamo:
  - <https://primer.dynamobim.org/>
- Dynamo Questions, inspiration:
  - <https://forum.dynamobim.com/>
- Design Script:
  - [https://dynamobim.org/wp-content/uploads/forum-assets/colin-mccroneautodesk-com/07/10/Dynamo\\_language\\_guide\\_version\\_1.pdf](https://dynamobim.org/wp-content/uploads/forum-assets/colin-mccroneautodesk-com/07/10/Dynamo_language_guide_version_1.pdf)
  - [http://designscript.io/DesignScript\\_user\\_manual\\_0.1.pdf](http://designscript.io/DesignScript_user_manual_0.1.pdf)
  - <https://dynamobim.org/wp-content/links/DesignScriptDocumentation.pdf>
  - <https://github.com/Amoursol/dynamoDesignScript>
- Refinery:
  - <https://www.autodesk.com/solutions/refinery-beta>
- Generative Design education:
  - <https://medium.com/generative-design>





AUTODESK®

Make anything™