An MEP Engineer’s Guide to Dynamo

Andrew Duncan
Associate | BIM Development Manager
@AGDunc

Andrei Capraru
Mechanical Engineer
People often associate Dynamo software with designing complex parametric geometry; but Dynamo software is not just a tool for creating funky shapes, it’s a whiz at processing all kinds of data. This class will demonstrate various examples of how Dynamo software has been applied to common MEP (mechanical, electrical, and plumbing) engineering tasks to make them more efficient or more accurate. We will cover linking Revit Space naming utility information to Revit software families, using Dynamo software as an engine for MEP calculations, using Dynamo software to give you visual feedback on how hard equipment is being asked to work in your design, and using Microsoft Excel as a source of data for much of the above. You also needn't worry if you are completely new to Dynamo software (most MEP folk will be), as we will cover enough basics to get you going before we dive into the advanced stuff.
Key learning objectives

At the end of this class, you will:

- Know enough about Dynamo to get started on your own scripts
- Learn how to pull data from Excel into Revit via Dynamo to inform calculations
- Learn how to use Dynamo as an engine for performing MEP calculations such as determining heating loads or flow rates
- Learn how to link Revit Space information to Revit families for use in calculations
A little info about me …

- Joined Arup in 2005
- BIM Development Manager, based in London
- MEP by background (mainly M)
- IEng professionally qualified Engineer
- Recently finished an MSc in *BIM and Integrated Design*
- 2 x Autodesk Revit Gunslinger
- … General technology nerd
A little info about me …

- Joined Arup in 2011
- Mechanical Engineer, based in London
- Experience on mission critical, healthcare, commercial and custodial projects.
- BIM Project of the Year for UK Government Project (HMYOI Cookham Wood)
- Specialised in implementation of BIM as a design tool for engineers.
Arup’s creative spark and intellectual independence has been there from the very beginning. These shared values, like the firm’s name, are essentially derived from the beliefs and convictions of the firm’s founder, the engineer and philosopher, Sir Ove Arup.

ARUP
... this time last year
What is Dynamo and How does it work?
Efficiency
Quality
Reduced Risk

"Reduce the effort to do the usual / necessary and increase the time to be creative"

Tristram Carfrae Global BIM strategy
Today’s Menu…

**Starter**
Populate a Model with Data from Excel

**Main**
Use that data for Calculations

**Dessert**
Link the Space calculations to Families for further analysis
Pushing Excel data into Revit *Spaces* using Dynamo
BEFORE WE START …

MODEL POPULATED WITH SPACES AND PARAMETERS REQUIRED FOR THE DYNAMO SCRIPT. ALL SPACES SET TO A SPACE TYPE (Using a specific parameter) TO MAP THEM TO THE SPACE DATA CONTAINED IN THE SPREADSHEET.
SELECT SPACES IN REVIT MODEL
READ DATA FROM REVIT
SELECT SPREAD-SHEET AND READ DATA FROM EXCEL
COMPARE SPACE TYPES IN EXCEL AGAINST SPACE TYPES IN REVIT
COMPARE SPACE TYPES IN EXCEL AGAINST SPACE TYPES IN REVIT
FORMATS ALL OF THE DATA IN FOR SPACE TYPES FOUND TO BE PRESENT IN BOTH REVIT AND IN EXCEL INTO THE NECESSARY STRUCTURE TO UPDATE THE REVIT MODEL
READ DATA FROM REVIT
UPDATES THE SPACE PARAMS IN REVIT MODEL

AUTODESK UNIVERSITY 2015
Excel List containing all room types with parameters to be set for each room listed on the first row, starting with Space_Type.

File Path
Browse...
`\d:\Dynamo Scripts\Space Types Parameters\Space Types.xlsx`

File.FromPath
path → file

Excel.ReadFile
file → sheetName, readAsStrings → data

String
Output →>

Name of worksheet in Excel file containing parameter values

List.Deconstruct
list: first, rest

LISTS
List.Deconstruct

Separates a list into two new lists, one showing just the first item in the original list and one for all of the other items.
<table>
<thead>
<tr>
<th>Space_Type</th>
<th>Design Heating Load</th>
<th>Design Cooling Load</th>
<th>Specified Return Airflow</th>
<th>Specified Exhaust Airflow</th>
<th>Specified Supply Airflow</th>
<th>Space_Mechanical_Cooling</th>
</tr>
</thead>
<tbody>
<tr>
<td>Atrium</td>
<td>968752</td>
<td>861113</td>
<td>2472</td>
<td>2119</td>
<td>1766</td>
<td></td>
</tr>
<tr>
<td>Cellular Office</td>
<td>861113</td>
<td>753474</td>
<td>2119</td>
<td>1766</td>
<td>1413</td>
<td></td>
</tr>
<tr>
<td>Conference Center</td>
<td>1076391</td>
<td>1184030</td>
<td>4238</td>
<td>4591</td>
<td>5299</td>
<td></td>
</tr>
<tr>
<td>Corridor</td>
<td>753474</td>
<td>645835</td>
<td>1766</td>
<td>1413</td>
<td>1059</td>
<td></td>
</tr>
<tr>
<td>Gym</td>
<td>538196</td>
<td>430556</td>
<td>1059</td>
<td>706</td>
<td>353</td>
<td></td>
</tr>
<tr>
<td>Open Office</td>
<td>538196</td>
<td>430556</td>
<td>1059</td>
<td>706</td>
<td>353</td>
<td></td>
</tr>
<tr>
<td>List of Space Types to be updated:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[0] Space_Type</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[1] Design Heating Load</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[2] Design Cooling Load</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[3] Specified Return Airflow</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[4] Specified Exhaust Airflow</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[5] Specified Supply Airflow</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[6] Space_Mechanical_Cooling</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[7] Space_Mechanical_Ventilat</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[8] Space_Mechanical_Fresh_Ai</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[9] Space_Occupancy_HVAC_Nor</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[10] Number of People</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[11] Space_Occupancy_HVAC_Nor</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[12] null</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Reads data listed in Excel file
- List.Deconstruct
- Excel.ReadFile
- Space Types to be updated
Reads data listed in Excel file

FromPath

Excel.ReadFile

List.Deconstruct

SheetName

readAsString

List.FirstItem

List of Space_Types to be updated

[25] null
[26] null
[27] null
[28] null
[29] null
[30] null

[1] list

[0] List

[0] Atrium
[1] 968751.939
[2] 861112.835
[3] 2472.027
[4] 2118.880
[5] 1765.733
[6] 80.000
[7] 120.000
[8] 1.000
Reads data listed in Excel file

FromPath

Excel.ReadFile

list

file

sheetName

readAsStrings

List.Deconstruct

list

first

rest

List.FirstItem

list

item

List of Space_Types to be updated

[25] null
[26] null
[27] null
[28] null
[29] null
[30] null

[1] List

[0] list

[0] Atrium
[1] 968751.939
[2] 861112.835
[3] 2472.027
[4] 2118.880
[5] 1765.733
[6] 80.000

Excel spreadsheet:

<table>
<thead>
<tr>
<th>Space_Type</th>
<th>Design Heating Load</th>
<th>Design Cooling Load</th>
<th>Specified Return Airflow</th>
<th>Specified Exhaust Airflow</th>
<th>Specified Supply Airflow</th>
<th>Space_Mechanics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Atrium</td>
<td>968752</td>
<td>861112.835</td>
<td>2472</td>
<td>2119</td>
<td>1766</td>
<td>1413</td>
</tr>
<tr>
<td>Cellular Office</td>
<td>861113</td>
<td>753474</td>
<td>2119</td>
<td>1766</td>
<td>1413</td>
<td></td>
</tr>
<tr>
<td>Conference Cen</td>
<td>1076391</td>
<td>1184030</td>
<td>4238</td>
<td>5297</td>
<td>4591</td>
<td></td>
</tr>
<tr>
<td>Corridor</td>
<td>753474</td>
<td>645835</td>
<td>1766</td>
<td>1413</td>
<td>1059</td>
<td></td>
</tr>
<tr>
<td>Gym</td>
<td>645835</td>
<td>538196</td>
<td>1059</td>
<td>706</td>
<td>353</td>
<td></td>
</tr>
<tr>
<td>Open Office</td>
<td>538196</td>
<td>430556</td>
<td>1059</td>
<td>706</td>
<td>353</td>
<td></td>
</tr>
</tbody>
</table>
Why is this so important?
Data passes through the WHOLE graph in LISTS…
Reads data listed in Excel file

`List.Deconstruct`
- `list`
- `first`
- `rest`

List of Space_Types to be updated

`List.FirstItem`
- `list`
- `item`
List.FirstItem

Gets the first item in a list, in this instance the Space Type

So why are there multiple items in the list?
LACING
List

1 + 10 = 11
2 + 20 = 22
3 discarded
Longest

1 + 10 = 11
2 + 20 = 22
3 + 20 = 23
Cross Product

List
(0) List
1 + 10 = 11
1 + 20 = 21

(1) List
2 + 10 = 12
2 + 20 = 22

(2) List
3 + 10 = 13
3 + 20 = 23
Good, but one of the best ways of working out which to *Lacing* method to use is trial and error (a lot of the time anyway)…
<p>| | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Atrium</td>
<td>968751.939</td>
<td>2</td>
<td>Cellular Office</td>
<td>861112.835</td>
<td>3</td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td>4</td>
<td>Conference Centre</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td>5</td>
<td>Gym</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td>6</td>
<td>Plantroom</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td>7</td>
<td>Reception</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td>8</td>
<td>Restaurant</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td>9</td>
<td>Storage</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td>10</td>
<td>Toilets</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td>11</td>
<td>null</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td>12</td>
<td>null</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td>13</td>
<td>null</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
<td>14</td>
<td>null</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td></td>
<td></td>
<td>15</td>
<td>null</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**List of Space Types to be updated**

- ListFirstItem
- ListCreate

Worksheet in Excel file containing values:

```
[28] null
[29] null
[30] null
```

List:

```
[Atrium 968751.939...]
```

`List.Create`
We’ve talked a lot about data, but what about the model?
Pick a category of Revit element to select from the active Revit model.

This actually selects the “MEPSpaces” in the active Revit model.

Gets the value of a specific parameter from each of the spaces...
...which creates a list.
ARE THE SPACES IN THE REVIT MODEL IN THE EXCEL LIST?
ARE THE SPACES IN THE EXCEL LIST IN THE REVIT MODEL?
COUNTS THE NUMBER OF EACH SPACE TYPE PRESENT IN THE MODEL

ADDS UP THE SUM TOTAL OF SPACES WITH SPACE TYPES PRESENT IN THE MODEL FOR USE LATER IN THE SCRIPT

FILTERS THE ORIGINAL LIST TO ONLY THOSE SPACE TYPES PRESENT IN THE MODEL

CONVERTS WHETHER A SPACE TYPE IS PRESENT IN THE MODEL INTO BOOLEAN LOGIC
FILTERS THE ORIGINAL LIST TO ONLY THOSE SPACE TYPES PRESENT IN THE MODEL
Space Types in Excel sheet with with Spaces set to those Space Types in the active Revit model

List.FilterByBoolMask

List
[0] List
  [0] Atrium
  [1] Cellular Office
  [2] Conference Centre
  [3] Corridor
  [5] Open Office
  [6] Plantroom
  [7] Reception
  [8] Restaurant
  [9] Storage
  [10] Toilets
[1] List
  [0] null
  [1] null
  [2] null
  [3] null

NOT IN THE MODEL

ARE IN THE MODEL
Space Types in Excel sheet with Spaces set to those Space Types in the active Revit model.
Space Types in Excel sheet with Spaces set to those Space Types in the active Revit model.
RESTRUCTURING THE LIST DATA IN ORDER TO PREPARE THE PARAMETER VALUES BEING PUSHED INTO THE SPACES THEMSELVES
Using *Space* parameters to inform calculations using Dynamo
SELECT THE SPACES IN THE MODEL

GET THE PARAMETER VALUES

CREATE A LIST OF THE PARAMETERS WE WANT FROM THESE SPACES

RE-ORDER THE LISTS CONTAINING THE PARAMETER VALUES FOR USE IN CALCULATIONS

USE PARAMETER VALUES FOR CALCULATIONS, AND SET NEW PARAMETER VALUES
SELECT THE SPACES IN THE MODEL
SELECT THE SPACES IN THE MODEL
Select Only Spaces for a Given Level

Select Only a Single Space
Selecting Elements in the model using Dynamo

<table>
<thead>
<tr>
<th>Category</th>
<th>Elements</th>
</tr>
</thead>
<tbody>
<tr>
<td>MEPSpaces</td>
<td>Category</td>
</tr>
<tr>
<td></td>
<td>[53] Space</td>
</tr>
<tr>
<td></td>
<td>[54] Space</td>
</tr>
<tr>
<td></td>
<td>[55] Space</td>
</tr>
<tr>
<td></td>
<td>[56] Space</td>
</tr>
<tr>
<td></td>
<td>[57] Space</td>
</tr>
<tr>
<td></td>
<td>[58] Space</td>
</tr>
<tr>
<td></td>
<td>[59] Space</td>
</tr>
<tr>
<td></td>
<td>[60] Space</td>
</tr>
<tr>
<td></td>
<td>[61] Space</td>
</tr>
<tr>
<td></td>
<td>[62] Space</td>
</tr>
<tr>
<td></td>
<td>[63] Space</td>
</tr>
<tr>
<td></td>
<td>[64] Space</td>
</tr>
</tbody>
</table>
CREATE A LIST OF THE PARAMETERS WE WANT FROM THE SPACES
List
0. Volume
1. Space_Mechanical_Ventilation_Air_Change_Rate
2. Space_Mechanical_Fresh_Air_Flow_Rate_Per_Person
3. Space_Occupancy_HVAC_Normal
4. Area
5. Space_Mechanical_Cooling_Load_Per_Area
6. Space_Mechanical_Heating_Load_Per_Area
GET THE PARAMETER VALUES
RE-ORDER THE LISTS CONTAINING THE PARAMETER VALUES
List.Transpose

Rearranges the list structure FROM the parameter values from each space TO groups of like parameter values.
List
[0] Volume
[1] Space_Mechanical_Ventilation_Air_Change_Rate
[2] Space_Mechanical_Fresh_Air_Flow_Rate_Per_Person
[3] Space_Occupancy_HVAC_Normal
[4] Area
[5] Space_Mechanical_Cooling_Load_Per_Area
[6] Space_Mechanical_Heating_Load_Per_Area

Cooling Load W/m²
5

List.GetItemAtIndex
list
index

List.GetItemAtIndex
list
index
Cooling Load $W/m^2 \times \text{Area} \times 10.764$?
ARP-19
Open Office
Occupancy Density: 10 m²/person
Fresh Air Per Person: 10 L/s/person
Air Changes Per Hour: 1 ACH
Specified Exhaust Air: 0.0 L/s
TIP:
Use Code Block instead of Number for greater accuracy when required
Linking *Space* parameters to *Family* Parameters
DIVIDE THE HEATING LOAD OF A SPACE OVER THE NUMBER OF PIECES OF EQUIPMENT IN THAT SPACE ABLE TO PROVIDE HEATING

COLOURS IN THE HEATING EQUIPMENT BASED ON HOW HARD IT IS WORKING ACCORDING TO ITS MAXIMUM CAPACITY
SELECT THE FAN COIL UNITS, BASED ON THEM BEING MECHANICAL EQUIPMENT AND THAT THEY CONTAIN THE PARAMETER “Performance_Capacity_Heating”

SELECTS ALL OF THE SPACES IN THE MODEL, AS PER THE PREVIOUS SCRIPTS
WHICH FCUs ARE IN WHICH SPACES?
This only give the center point of a family, which may not be in a space (as the family may be in a void above the space)
WHY IS THIS SO DIFFERENT
Why is this so different

Room Calculation Point
DIVIDE THE HEATING LOAD OF A SPACE OVER THE NUMBER OF PIECES OF EQUIPMENT IN THAT SPACE ABLE TO PROVIDE HEATING
SELECT THE SPACES IN THE MODEL GET THEIR HEATING LOADS

SELECT THE FCUs IN THE MODEL

DIVIDE THE SPACE HEATING LOADS BETWEEN THE FCUs IN A GIVEN SPACE

DETERMINES WHICH FCUs ARE IN EACH OF THE SPACES

SET THE FCU HEATING LOADS
Automating calculations within Families using formulas

See my other class, ES11064 - Creating Smarter Revit MEP Families

**TIP:** Don’t model pipe sizes anymore, just model everything at largest anticipated pipe size (e.g. approximately 50Ø for a FCU system) and let Revit do the pipe sizing for you. This will save you loads of time.
Efficiency

Quality

Reduced Risk

“Reduce the effort to do the usual / necessary and increase the time to be creative”

Tristram Carfrae Global BIM strategy
… and as an added bonus (come back next year for more)
That’s your lot … Thank you very much for attending!

- Via the Survey Stations, email or mobile device
- AU 2015 passes given out each day!
- Best to do it right after the session
- Instructors see results in real-time