

Topologic

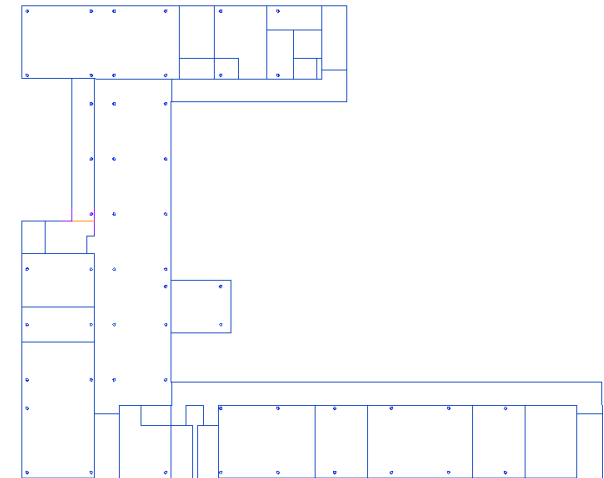
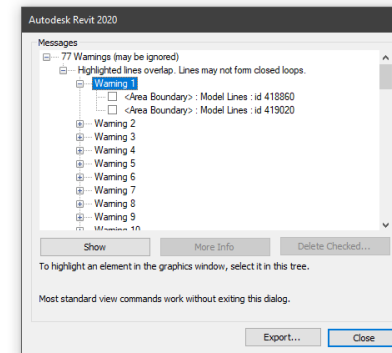


Merge Overlapping Lines
Using Dynamo!

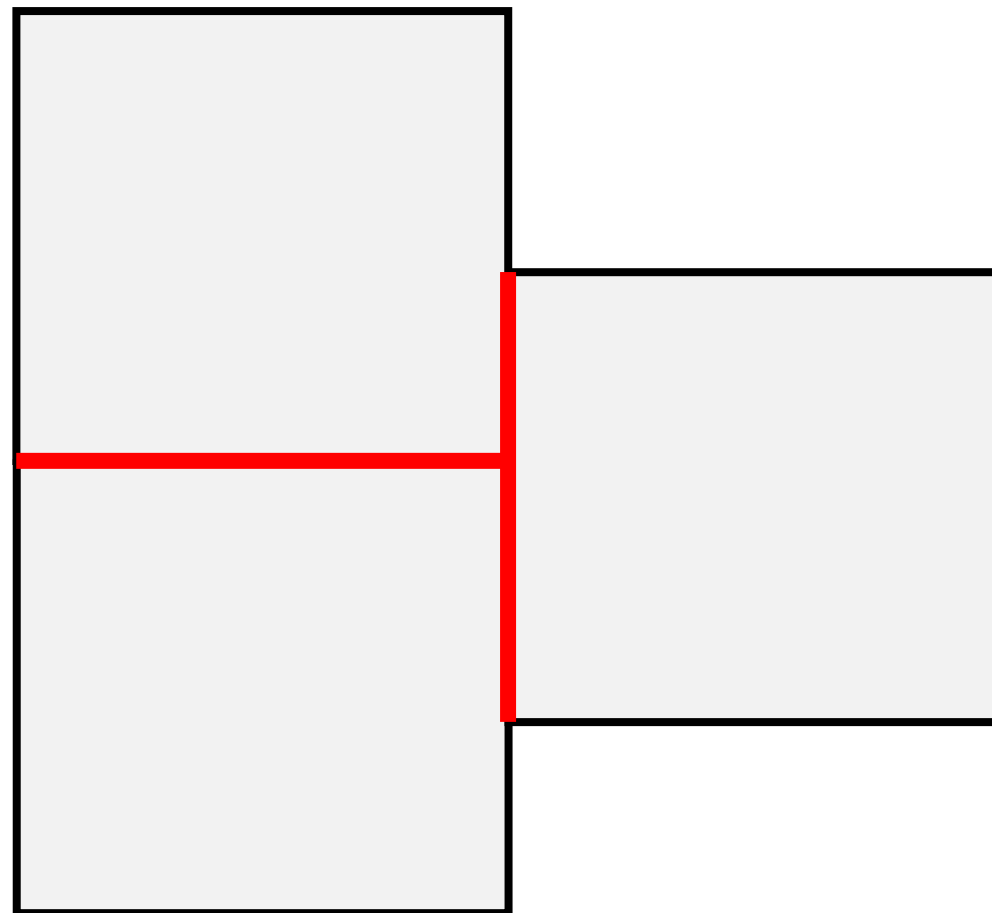
Previously



Merge Overlapping Lines
Using Rhino Inside!

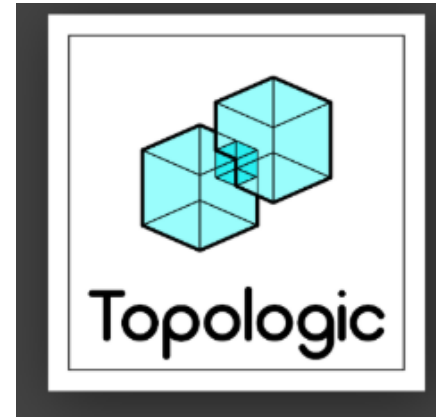


The Problem



Topologic

<https://topologic.app/>



Dr. Wassim Jabl
Project Director
Welsh School of Architecture
Cardiff University



Prof. Robert Aish
Co-Investigator
Bartlett School of Architecture
University College London



Dr. Simon Lannon
Co-Investigator
Welsh School of Architecture
Cardiff University



Dr. Aikaterini Chatzivasileladi
Research Associate
Welsh School of Architecture
Cardiff University



Dr. Nicholas Mario Wardhana
Research Associate
Welsh School of Architecture
Cardiff University

Thanks for
collaborating!



Wassim Jabi · 1st



(Cardiff UK)

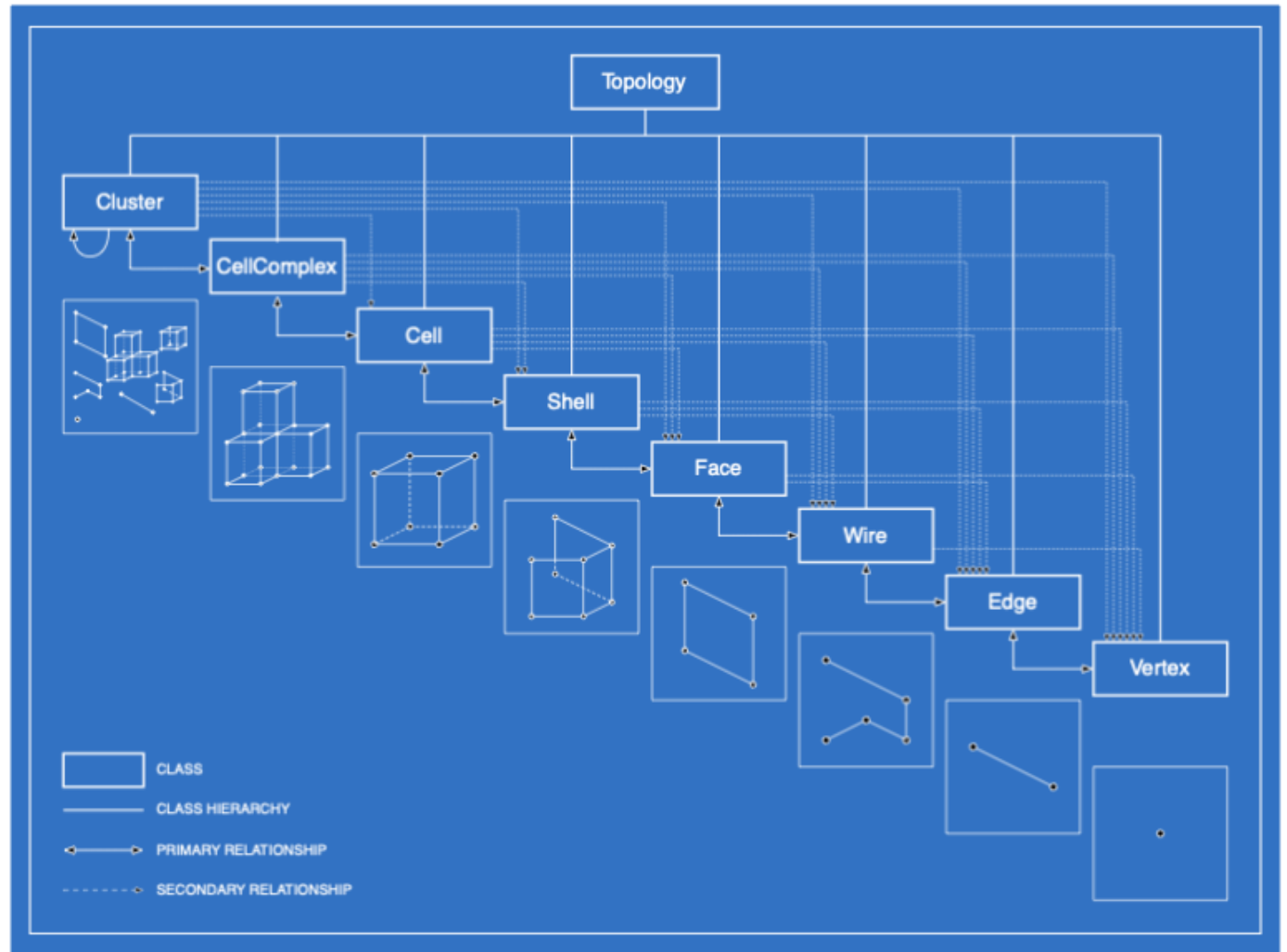


Wassim Jabi · 5:13 PM

Hi Gavin. Was watching your video on getting room boundaries from Revit. I think I know the answer to the following, but hoping to be proven wrong: Is there any way you can think of to create room boundaries from Revit that always share lines (similar to Testfit plans?). Obviously this would allow me to build Topologic CellComplexes and run energy analysis

Topologic

<https://topologic.app/>



Resource files and package

<https://topologic.app/>

LEARNING

Documentation for Users

[Introduction](#) [\[PDF\]](#)

[Software Installation](#) [\[PDF\]](#)

[User Documentation](#) [\[LINK\]](#)

[Tutorial 1: Getting Started](#) [\[PDF\]](#) [\[DYN\]](#)

[Tutorial 2: Creating a Multi-Storey Building](#) [\[PDF\]](#) [\[DYN\]](#)

[Tutorial 3: Topological Queries](#) [\[PDF\]](#) [\[DYN\]](#)

[Tutorial 4: Analysing Energy Performance](#) [\[PDF\]](#) [\[DYN A\]](#), [\[DYN B\]](#), [\[ZIP\]](#)

Other Sample Definition Files

Wire Closed Circuits. This custom node returns the exclusive closed wires found in the input wire. [\[DYF\]](#)

Wire Remove Non-manifold Edges. This custom node removes all the non-manifold edges found in the input wire. [\[DYF\]](#)

DynaSpace/Topologic. This example workflow illustrates how to integrate DynaSpace and Topologic. [\[ZIP\]](#)

Minimum Spanning Tree. This custom node, based on a found python code, outputs a wire representing the Minimum Spanning Tree of the input graph. [\[ZIP\]](#)

Diversion Workflow. This example workflow diverts a straight line around an obstacle. This experiment diverts a straight line around an arbitrary obstacle (concave or convex polygon). The path always diverts in 90 degree increments. The path avoids the bounding box of the polygon as oriented perpendicular to the path. The path chooses to go left or right based on the shortest path. [\[DYN\]](#)

TestFit2Topologic Workflow. All files needed to illustrate a TestFit to Topologic workflow. This workflow takes as input a TestFit RSD file and outputs a Cluster of Cells categorised by their TestFit type. [\[ZIP\]](#)

CellComplex Utilities. A collection of Dynamo custom nodes to derive specific Faces from a CellComplex. These include: CellComplexBy3DArray.dyf, CellComplexExternalFaces.dyf and CellComplexInternalFaces.dyf. A demo definition file (CellComplexTower.dyn) is also included. [\[ZIP\]](#)

Topology Display. This custom node makes it far easier to display any topology with custom colors, transparency and with or without edges. [\[DYF\]](#)

Graph Utilities. A collection of Dynamo custom nodes to compute some Graph metrics. These include: GraphAdjacencyMatrix, GraphIsConnected, GraphIsRegular, GraphIsBipartite. [\[ZIP\]](#)

Graph Shortcuts. This custom node and sample definition illustrate how to create optimized shortest paths through a graph by creating shortcuts around a certain set of vertices with a specified custom dictionary value. [\[DYN\]](#), [\[DYF\]](#)

Stanford Bunny. This definition illustrates how to convert a mesh into a Topologic entity. For this definition, you need the MeshToolkit package from the Package Manager and the following Stanford Bunny OBJ file. [\[OBJ\]](#) [\[DYN\]](#)

Optimising Location of Atrium. This definition illustrates how to use Dual Graph and topology to optimize the number of connected Cells and the remnant exterior wall area of an atrium Cell that is imposed on a CellComplex. This definition is meant to be used with Autodesk's Project Refinery or other optimisation engines. [\[DYN\]](#)

Voronoi 3D. This definition illustrates how to slice a Cuboid into a set of Voronoi Cells based on a list of points. [\[DYN\]](#)

Shortest Paths. This definition illustrates how to use the Graph class to find shortest paths in a CellComplex. [\[DYN\]](#)

Shortest Paths By Key. This definition illustrates how to use the Graph class to find shortest paths in a CellComplex by Key. [\[DYN\]](#)

Spread/Growth. This definition and custom node illustrate how to use the Graph class to simulate spread/growth within a CellComplex. [\[ZIP\]](#)

Content and Dictionary. This definition illustrates how to use the Content/Context system and Dictionary to store custom attributes in topologies. [\[DYN\]](#)

Topological Queries. This definition illustrates how to determine attributes based on topologic queries of custom dictionaries. [\[DYN\]](#)

Boolean By Dictionary. This definition illustrates how to perform a Boolean operation based on the Topology type as encoded in its custom dictionary. [\[DYN\]](#)

Video Recordings

- [Webinar: Introduction to Topologic - 25 February 2019](#) [\[LINK\]](#)
- [The Making of a Twisted Tower](#) [\[LINK\]](#)
- [Topologic Workshop: New Features \(11-10-2019\)](#) [\[LINK\]](#)

Documentation for Developers

[Software Development Kit Documentation](#) [\[Link\]](#).

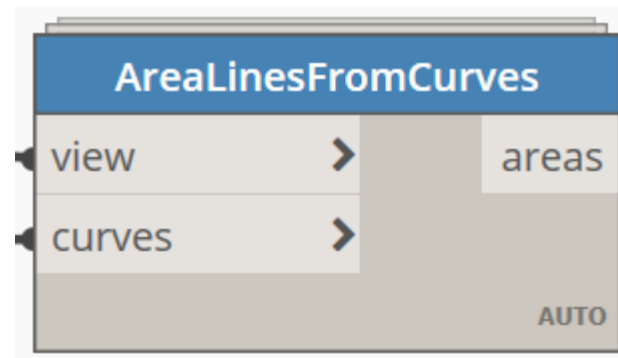
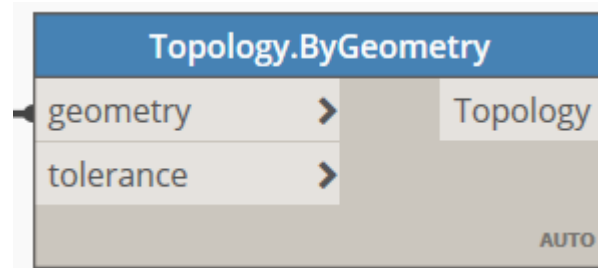
Frequently Asked Questions

▼ [Show Questions](#)

Custom packages

<https://topologic.app/>

Topologic (1.3.0 ideally)
Crumple (on my github)



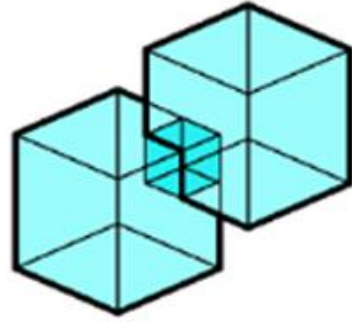
Anyway...

Let's explore
Wassim's
approach



Files are on
github

<https://github.com/aussieBIMguru>



Topologic



Merge Overlapping Lines
Using Dynamo!