Topologic

(This document is formatted for double-sided printing)

TUTORIAL 04
Analysing Energy Performance

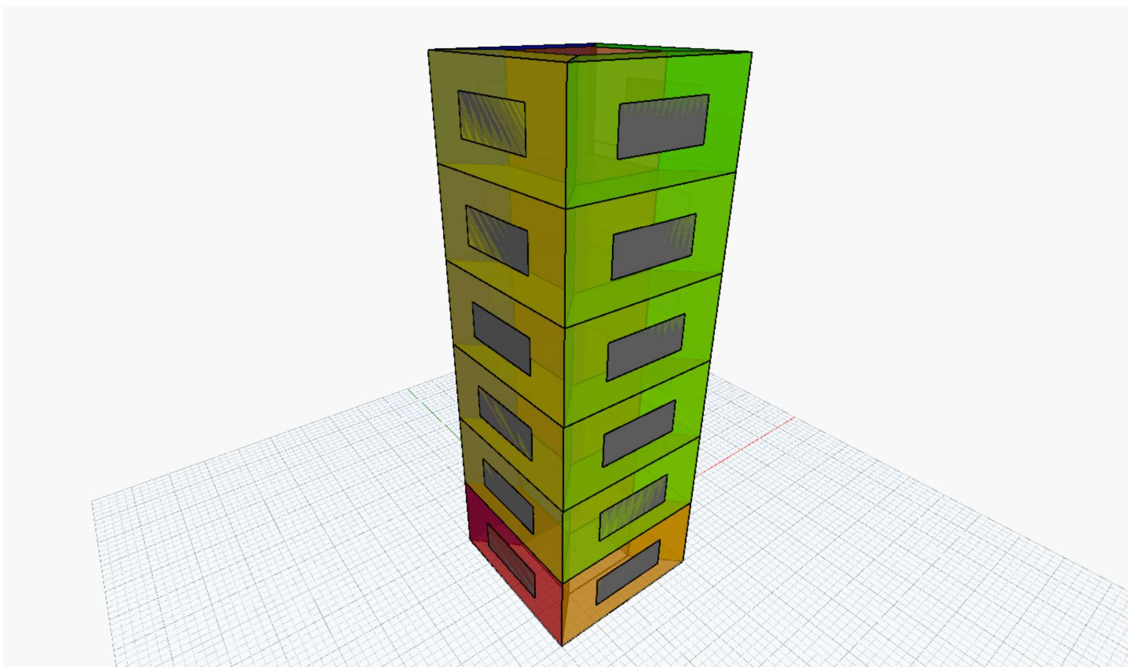
Version: 0.8.6

September 2019

TABLE OF CONTENTS

Introduction

Topologic enables the user to analyse the energy performance of a building with internal subdivisions. TopologicEnergy, which is part of Topologic, interfaces directly with Alliance's and NREL's OpenStudio and DOE EnergyPlus. TopologicEnergy is not designed to be a full energy simulation application, but as a demonstration of the power of non-manifold topology when combined with energy analysis. TopologicEnergy creates the necessary .OSM and .IDF files for your to conduct more sophisticated analysis in other applications. However, with TopologicEnergy, you can query the simulation results and display them as numbers, charts (using Excel or Mandrill), and coloured 3D geometry as you see below. To start, find and open the file called **Topologic-06a-Tutorial04.dyn**. This files already includes all the apertures. Please study how these apertures have been created and added to the CellComplex. Open this file and save it under a new name (e.g. **Topologic-06b-Tutorial04.dyn**).



Prerequisites

TopologicEnergy requires the following files to be installed:

1. OpenStudio v. 2.8.0. The topologic installer automatically detects and installs OpenStudio (which in turns includes EnergyPlus).
2. The minimal .OSM template. This template contains the minimal information needed to start building an EnergyModel. The file is called **OpenStudioTemplate280.osm**
3. Weather and DDY files. You can use your own weather (.EPW) files. In this tutorial, we will assume that you are using the files that we have included (Gatwick Airport in the UK).

Step 1: Build an Energy Model

From TopologicEnergy, drag and drop an **EnergyModel.ByCellComplex** node. Connect the output of **Topology.AddApertures** (i.e. the final **CellComplex**) to the ***building*** input parameter. Next, create a Code block and type the following lines into it:

floorLevels = 0..60..10;

buildingName = "Multi-Storey Building";

buildingType = "Commercial";

defaultSpaceType = "ASHRAE 189.1-2009 ClimateZone 4-8 MediumOffice";

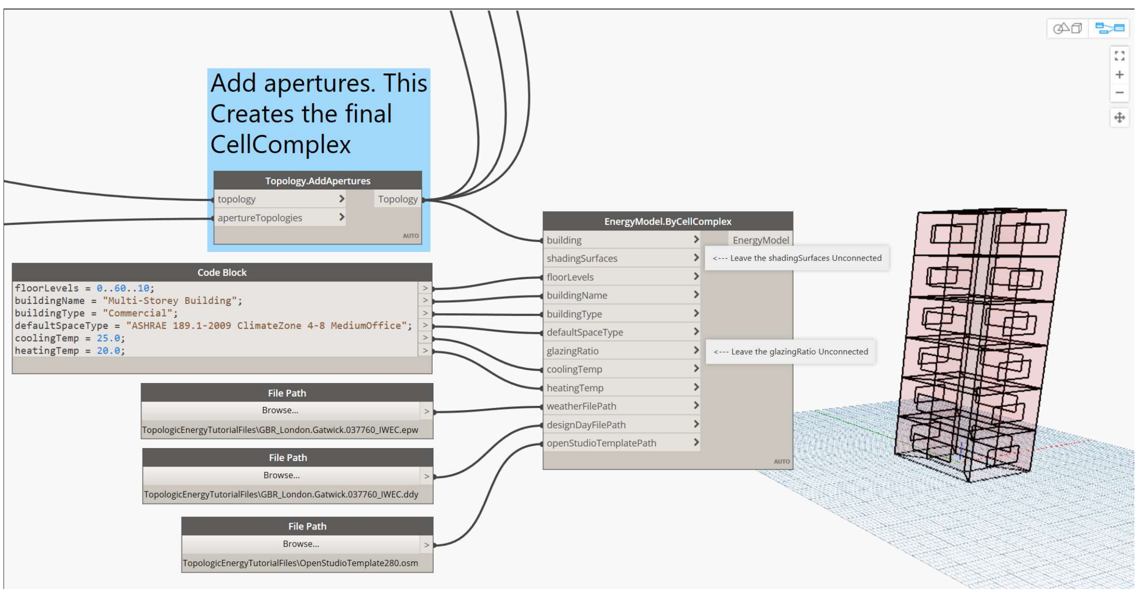
coolingTemp = 25.0;

heatingTemp = 20.0;

Next, create three Dynamo **File Path** nodes. Connect each node to the following files:

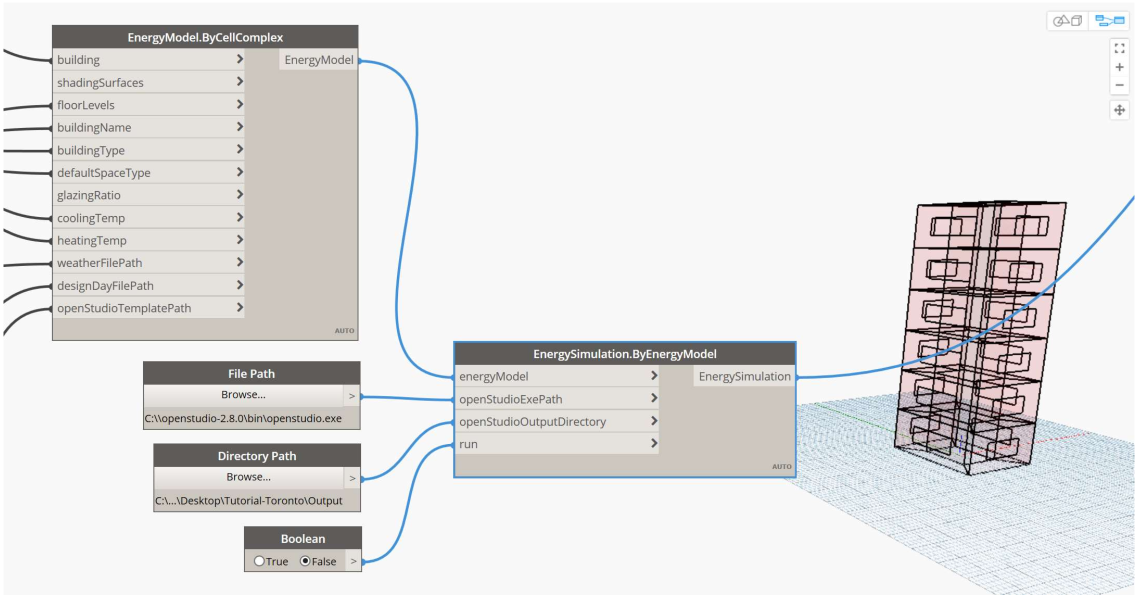
1. The weather file: GBR_London.Gatwick.037760_IWEC.epw
2. The design day file: GBR_London.Gatwick.037760_IWEC.ddy
3. The OpenStudio template file: OpenStudioTemplate280.osm

The energy model node should now look as the figure below.



Step 2: Run an Energy Simulation

From **EnergySimulation**, create an **EnergySimulation.ByEnergyModel** node. Connect the ***EnergyModel*** output parameter from **EnergyModel.ByCellComplex** to its ***energyModel*** input parameter. Create a Dynamo **File Path** node and Browse for the OpenStudio.exe file. This is usually found at **C:\\openstudio-2.8.0\bin\openstudio.exe**. Next create a Dynamo **Directory Path** node. Browse for a folder of your choosing in your home folder to save the output energy files. Finally, create a Dynamo **Boolean** node and make sure it is set to ***False***. Connect these nodes to their respective input parameters on the **EnergySimulation.ByEnergyModel** node. Your definition should now look like the figure below.



Step 3: Query the Simulation Result

From **SimulationResult**, create a **SimulationResult.ByEnergySimulation** node. Connect the **EnergySimulation** output parameter of the **EnergySimulation.ByEnergyModel** to the **energySimulation** input parameter of this node. Next, create a Dynamo **Code Block** node and type the following 5-line text into it. This is the SQL query to retrieve the Calculated Design Load of each thermal zone:

EPReportName = "HVACSizingSummary";

EPReportForString = "Entire Facility";

EPTableName = "Zone Sensible Cooling";

EPColumnName = "Calculated Design Load";

EPUnits = "W";

Or, if you wish to get the thermal comfort data, try:

EPReportName = "SystemSummary";

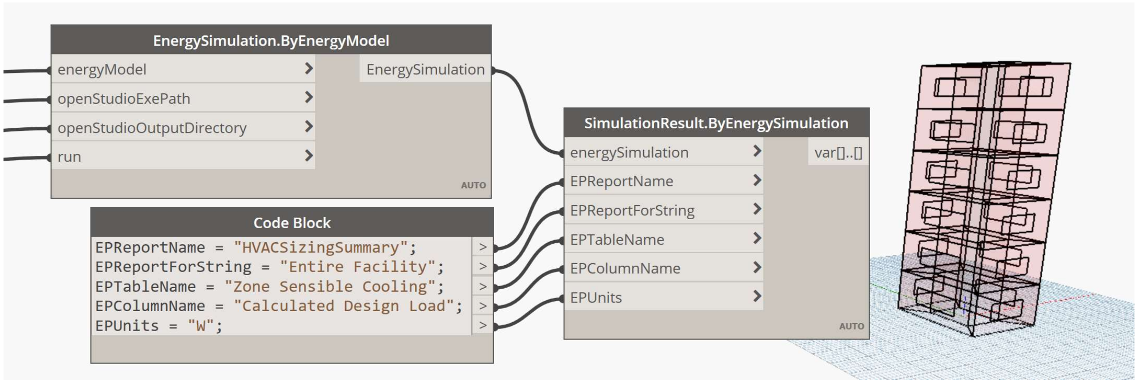
EPReportForString = "Entire Facility";

EPTableName = "Time Not Comfortable Based on Simple ASHRAE 55-2004";

EPColumnName = "Summer or Winter Clothes";

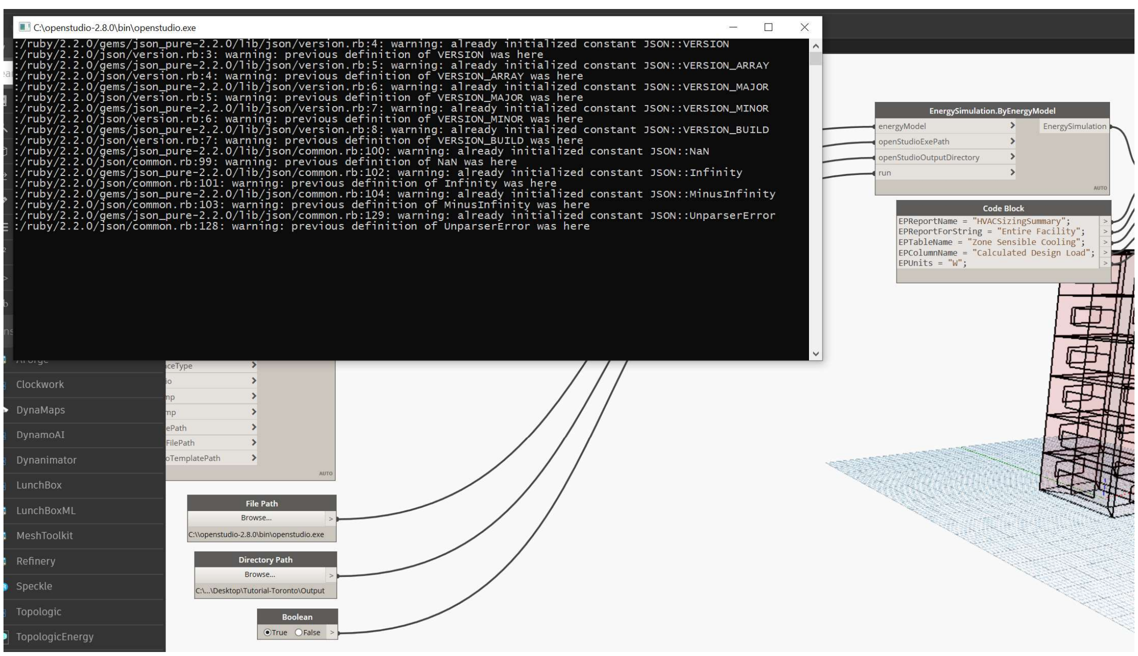
EPUnits = "hr";

Connect each line in order to the corresponding input parameter of the **SimulationResult.ByEnergySimulation** node. Your definition should now look like the figure below.



Step 4: Run the Simulation

Now is a good time to save your definition. Toggle to ***True*** the Dynamo **Boolean** node that is connected to the ***run*** input parameter of the **EnergySimulation.ByEnergyModel** node then press the Dynamo **Run** button in the lower left corner. After a few seconds, a black shell window will appear as you see below. This is the OpenStudio.exe application running the energy analysis. This window will close automatically after the completion of the analysis. Soon after the Dynamo run will complete as well. The model has now been analysed and the files written out to the output directory. In addition, the data has been read back into **TopologicEnergy** and we are now ready to display the results.



Step 5: Display the Simulation Result

From **SimulationResult**, create a **SimulationResult.RGB** node. Connect the ***SimulationResult*** output parameter of the **SimulationResult.ByEnergySimulation** node to its ***simulationResult*** input parameter. For now you can leave the ***minDomain*** and ***maxDomain*** input parameters unconnected. These control the colour range. If un-connected, it will use the minimum and maximum values from the simulation to determine the colour range to use. TopologicEnergy is designed to work with another Dynamo package called Mandrill to display bar charts and a colour legend. Please explore that on your own.

Next, create a Dynamo **List.Transpose** node. Connect the ***int[][]*** output parameter of the **SimulationResult.RGB** node to its ***lists*** input parameter.

Next, create a Dynamo Code Block node and type the following 4-line text into it:

a = 200;

r = rgblist[0];

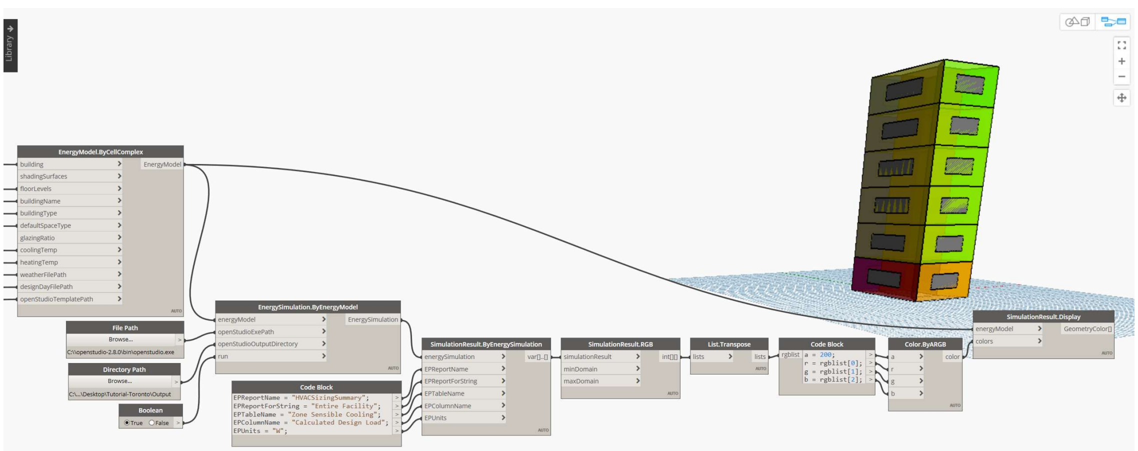
g = rgblist[1];

b = rgblist[2];

Connect the ***lists*** output parameter of the **List.Transpose** node to its ***rgblist*** input parameter.

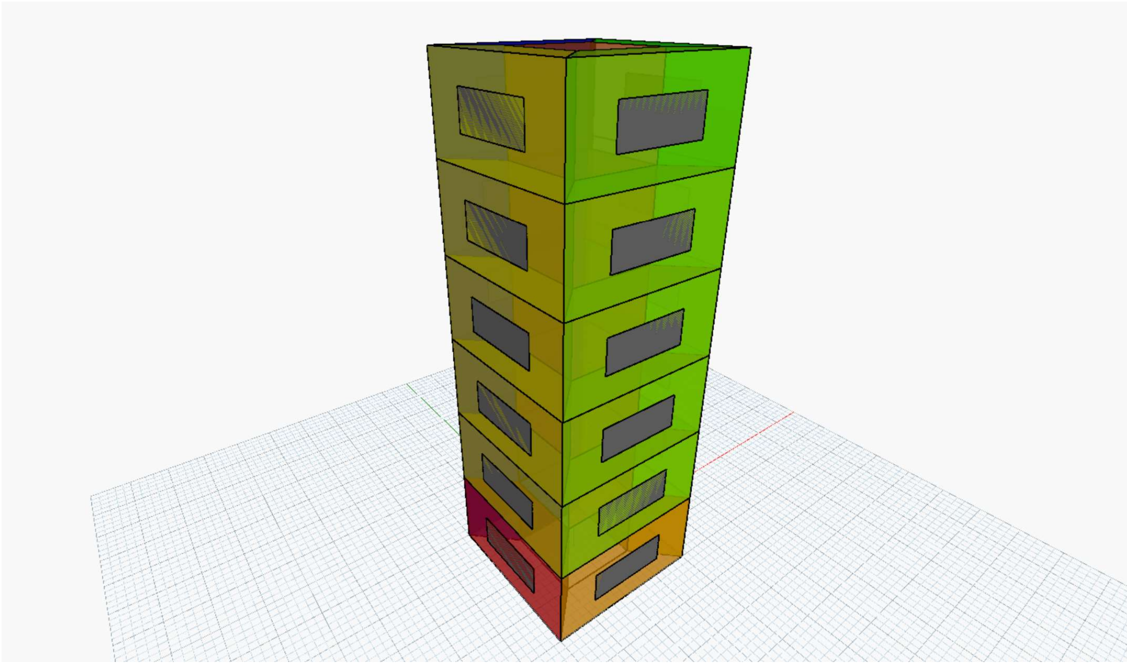
Next, create a Dynamo **Color.ByARGB** node and connect each of the output parameters of the Code Block to their respective input parameters on that node.

Next, from **SimulationResult**, create a **SimulationResult.Display** node. Connect the ***EnergySimulation*** output parameter of the **EnergySimulation.ByEnergyModel** node to its ***energySimulation*** input parameter. Connect the ***color*** output parameter of the **Color.ByARGB** node to its ***colors*** input parameter.



Hide the **CellComplex** by right-clicking on the **GeometryColor.ByGeometryColor** node in the “Display Cell Complex” group and un-ticking ***Preview***.

Finally, run the definition. You should see the colour of each cell according to its calculated cooling load as below.



The Leverhulme Trust

This project is funded by a Leverhulme Trust Research Project Grant
(Grant No. RPG-2016-016)

CONTACT US

Web: topologic.app

Email: info@topologic.app

Twitter: @topologicBIM

Phone: +44 (0) 29 2087 5981

Welsh School of Architecture
Cardiff University
Cardiff CF10 3NB
United Kingdom