
Lecture 10. BEM: Simulation Control

“Simulation is cheaper than constructing the wrong building!”

— Dru Crawley, in a 2014 presentation to the Utah ASHRAE Chapter

1 Using the OpenStudio interface

When you work with the OpenStudio application, all of the tabs on the left side will affect the building energy model or the way you interact with it. The OpenStudio Introductory Tutorial [5] provides a PDF of the OpenStudio Quick Start Guide, which contains a Basic Workflow Guide that is a visual map of the graphical user interface (GUI). The tab with an arrow is where the action happens—when you click Run, OpenStudio reads the instructions you’ve given it, calls the EnergyPlus engine, and starts simulating.

The rest of this lecture will focus on key pieces of information provided to the EnergyPlus engine that determine how the simulation will be conducted, concluding with a description of measures that act on OpenStudio or EnergyPlus models.

2 Version

Each IDF has an object (see Lecture 8) called **Version** that indicates the version of EnergyPlus that the IDF file was created for. An error will result if you are simulating the file in a different version of EnergyPlus, but will not necessarily stop the simulation. The simulation may crash if, for instance, the version of EnergyPlus you’re using can’t find something in the IDF file that it is looking for.

Note that an EnergyPlus version is different from an OpenStudio version. The EnergyPlus software is much older, and we’re currently on Version 9.0, which runs under the hood of OpenStudio Version 2.7.

3 Run Period

An IDF has an object called **RunPeriod** that tells EnergyPlus how to match up the simulation with the weather file. It contains information like the month, day, and year when the simulation will begin and it determines how long the simulation will run for (assuming a fatal error doesn’t stop it along the way). If the weather file contains information about Daylights Savings time or holidays, the RunPeriod object will tell EnergyPlus how to use that data.

4 Time Step

The basic time step for a simulation is provided in the number of timesteps per hour. You may choose any integer that divides into 60, from 1 (meaning an hourly timestep) to 60 (meaning a timestep of 1 minute). A value of 6 (timestep of 10 minutes) is commonly used, as it settles in the middle of the trade-off between quick simulations (longer timesteps) and capturing more of the building’s dynamic response (smaller timesteps).

Remember that the weather data are typically provided in hourly timesteps (8760 rows of entries). This is not a problem for simulations because EnergyPlus will automatically interpolate the weather data between the data points it has and, as explained in the Input-Output Reference:

Many aspects of a model have time scales that differ from the that of the weather data. A goal of the modeling is to predict how the building will respond to the weather. However, the building’s response is not governed by the time scale that the weather data are available at, but rather the time scales of the dynamic performance of the thermal envelope as well as things like schedules for internal gains, thermostats, and equipment availability. [3]

Technically, EnergyPlus is working with two important timestep values that are not necessarily the same:

- **Zone timestep**—the basic timestep for solving general heat transfer equations, for example. It dictates the smallest time scale on which results would be available. This is consistent throughout the simulation.
- **System timestep**—the timestep used internally for solving an HVAC system model, for example. Its maximum value will be the zone timestep value, but it may be smaller as needed for EnergyPlus to reach convergence. It does not affect the time scale on which results would be available. This can change throughout the simulation.

As end users, we will only modify the zone timestep, although we could indirectly force changes in the system timestep by adjusting the convergence limits.

5 Simulation Control

Each IDF has an object called **SimulationControl** that tells EnergyPlus what simulations it will actually perform. We have several options, including:

- Sizing calculations for the zone and system design loads (defaults to No)
- Sizing calculations for plants, such as a central chiller loop or cooling tower (defaults to No)
- Simulation for the weather file run period, as in Section 3 (defaults to Yes)
This is what we typically think of as an EnergyPlus ‘run’.
- Simulation for specific advanced sizing calculations for HVAC equipment (optional field; default is to ignore it, which is essentially a No)

6 Design Day

A **design day** simulation is chosen to provide conditions that will result in load calculations demonstrating loads that the equipment needs to be able to meet. That is, it is a representative day used for design (e.g. equipment sizing). Often more than one design day is used in the process of sizing and selecting equipment for a building—for example, one summer design day representing extreme hot weather conditions and one winter design day representing extreme cold weather conditions are frequently used. For more about the concept of design days or their implementation in EnergyPlus, refer to the ASHRAE Handbook [2] or the EnergyPlus Input-Output Reference [3].

An IDF has an object called **SizingPeriod:DesignDay** that provides information about what design day simulation(s) to conduct. Information includes the month and day, maximum (dry bulb) temperature, daily temperature range (difference between high and low dry bulb), humidity conditions, and more.

7 Location

An IDF has an object called **Site:Location** that provides relevant information about the building site, such as latitude, longitude, local time zone, and elevation. If information given here conflicts with any location data provided in the weather file, the weather file takes precedence.

8 Building

An IDF has an object called **Building** that provides basic information about the building itself. The includes information like the building's orientation (defined as an offset of the building's north axis from true North) and the type of terrain surrounding it (such as 'city' or 'suburbs')

It also contains a field for the name of the building, which doesn't affect the simulation itself, but will be used with output from the simulation. This will come in handy when you're digging through output files days or even months later.

9 Variables

EnergyPlus holds on to information about the simulation as internal variables. These could be items like the air temperature in a certain zone (in Celsius), the electricity used for lighting (in J), or the gas used for the whole facility (also in J). The specific variable names that are available for a given building simulation are shown in two "eplusout" output files. That means EnergyPlus doesn't know all of the variables it might be possible to report on before it's simulated anything. An experienced user who puts together their own IDF may know on the front end what variables they will be able to ask for; as for us, we will sometimes need to run an input file to get a list of possible variables. We'll talk more about controlling what output you get from a simulation (i.e. "Input for Output" [3]) in the next lecture.

10 Measures

Within the OpenStudio platform, we can leverage scripts to automate the process of altering our model.

A **measure** is a set of programmatic instructions that makes changes to an energy model to reflect its application. [For] example, the measure might find the default construction used by roof surfaces in the model, copy this construction and add insulation material to the outside, then set the new construction with added insulation as the default construction to be used by roof surfaces. Measures can be written specifically for an individual model, or they may be more generic to work on a wide range of possible models. [1]

The underlying code is written in Ruby, but by leveraging the Building Component Library [4] as described in the OpenStudio Introductory Tutorial [5], we can use measures without having to create or alter the code ourselves. I do recommend taking a look at the script for a measure (look for 'measure.rb') just to get a sense of what it looks like. Try to alter it and see what happens when you run it again if you are already familiar with Ruby or just feel adventurous [6].

When you reach the Measures interface in the OpenStudio GUI, you will see options for OpenStudio measures, EnergyPlus measures, and reporting Measures. The first two affect input files, as described below, and the last affects your output, as discussed in the next lecture.

10.1 OpenStudio

An OpenStudio measure is written to alter the *.osm* file, the OpenStudio model. It could change the way the currently open model will run, or could change the entire simulation workflow of the OpenStudio Application.

10.2 EnergyPlus

An EnergyPlus measure is written to alter the *.idf* file, the input data file for the EnergyPlus engine. It might change each object of a given type in a consistent way, for example.

RESTRICTED PUBLIC LICENSE -- READ BEFORE SHARING. This is a draft version made available by Amanda D. Smith under a Creative Commons Attribution-NonCommercial-ShareAlike license. CC BY-NC-SA 4.0

References

- [1] *About Measures - OpenStudio[®] User Docs*. http://nrel.github.io/OpenStudio-user-documentation/getting_started/about_measures/. Accessed: 2019-1-29. URL: http://nrel.github.io/OpenStudio-user-documentation/getting_started/about_measures/.
- [2] *ASHRAE Handbook Online*. ASHRAE. URL: <http://handbook.ashrae.org/Handbook.aspx>.
- [3] Big Ladder Software LLC. *EnergyPlus Version 9.0 Documents*. <https://bigladdersoftware.com/epx/docs/9-0/index.html>. Accessed: 2019-1-23. URL: <https://bigladdersoftware.com/epx/docs/9-0/index.html>.
- [4] *Building Component Library*. <https://bcl.nrel.gov/>. Accessed: 2016-6-23. URL: <https://bcl.nrel.gov/>.
- [5] *Installation and Introductory Tutorial - OpenStudio[®] User Docs*. http://nrel.github.io/OpenStudio-user-documentation/getting_started/getting_started/. Accessed: 2019-1-29. URL: http://nrel.github.io/OpenStudio-user-documentation/getting_started/getting_started/.
- [6] Amanda D. Smith. *Quick guide to getting up and running with Ruby*. <https://medium.com/@amandasmith/quick-guide-to-getting-up-and-running-with-ruby-bl1ab148dcff4>. Accessed: 2019-1-29. Mar. 2018. URL: <https://medium.com/@amandasmith/quick-guide-to-getting-up-and-running-with-ruby-bl1ab148dcff4>.