Student Project Activity #2: Energy Efficiency Measures

Overview

In this activity, students propose and evaluate their own combination of building energy efficiency measures for the class project community. The addition of energy efficiency measures demonstrates how the community can achieve its energy goals.

Tasks

1. (10 mins) Select an energy-related objective for your district, based on the results from Activity #1.
2. (20 mins) Choose Energy Conservation Measures (ECM): Browse the measures in the list below and select 3 that you believe will help achieve your energy objective.

> **Info:** An ECM is a proposed upgrade to a building that is expected to save energy, improve IEQ, and/or enhance grid interconnectivity. URBANopt uses OpenStudio's term of "measure" to implement an ECM in the modeling context of a script.

- Add Window Overhangs
- Enable Demand Controlled Ventilation
- Enable Economizer Control
- Improve Fan Belt Efficiency
- Improve Motor Efficiency
- Increase Insulation for Walls
- Increase Insulation for Roofs
- Reduce Electrical Equipment Loads
- Reduce Lighting Loads
- Reduce Space Infiltration

Further detail on each of these measures is available here: https://github.com/NREL/openstudio-ee-gem/tree/develop/lib/measures

3. (30 mins) Research the approximate cost to implement your selected measures. This information will be needed for your evaluation and final report.

4. **(60 mins) Write a Proposal:** Write a 1-page proposal describing your objective and the selected measures you will examine, with a brief justification of why you selected them. Clearly state the evaluation criteria upon which you will base your analysis in the subsequent portions of the project.

5. **(30 mins) Enable Measures**

   The linked document in the callout below fully documents how to add measures to your project. The steps below can be used for this project since the example project is prepopulated with measures already.

   <div align="center">

   [Details on how to add measures](#)

   </div>

   Add measures to your project model:

   > **Info:** The Mapper File is used to programmatically assign the appropriate values to measure arguments based on inputs defined in the GeoJSON Feature File.

   a. Open the ClassProject.rb mapper file in a text editor. The default location will be in <working>/[coincident|diverse]/mappers/ClassProject.rb

   b. In the opened ClassProject.rb file, find the measures that you would like "enabled" and change the "SKIP" value of your chosen measures from "true" to "false" (i.e., do not skip the measure), shown below:

   ```
   OpenStudio::Extension.set_measure_argument(osw, 'ExampleMeasure',
   '__SKIP__', false)
   ```

   c. Most measures require arguments to be passed in (e.g., to reduce internal lighting loads, the user must pass the percentage to reduce the loads by). To edit a measure argument, add the following template code to the mapper and update the argument as desired:

   ```
   OpenStudio::Extension.set_measure_argument(osw, 'ExampleMeasure',
   'MeasureArgument', <set argument value here>)
   ```

   d. Save and close the mapper file

6. Create a new scenario CSV file using the modified mapper created in Task #5. The video below is an overview on how the scenario CSVs work and the steps below provide an example on accomplishing this for the class project.

Creating Scenario CSV Files

a. In the project directory (e.g., <working>/[coincident|diverse]/) make a copy of the baseline_scenario.csv file and rename to a name of your choice (e.g., custom_example_scenario.csv)

b. Open the CSV file and substitute all the string inputs under the header "Mapper Class" to from ` URBANopt::Scenario::BaselineMapper ` to `URBANopt::Scenario::ClassProjectMapper`

> 💡 **Info:** Upon creation of the coincident and diverse project files, an example `classproject_scenario.csv` is automatically created for reference purposes.

7. (60 mins) Run Command: Run this project model with your new measures added by using the path for your new scenario CSV file (e.g., custom_example_scenario.csv):

```
$ cd <working>
$ uo run -f coincident/class_project_coincident.json -s
coincident/custom_example_scenario.csv

>>> Simulating features of 'class_project_coincident.json' as
directed by 'custom_example_scenario.csv'...

>>> Lots of simulation output
>>> DONE, result = true
>>> Done Running Scenario
```

8. (5 mins) Process Command: Post-process general results for the project scenario:

```
$ cd <working>
$ uo process -d -f coincident/class_project_coincident.json -s
coincident/custom_example_scenario.csv

>>> Done

# Also run the visualize command to compare baseline with
class_project_coincidennt.json

$ uo visualize -f coincident/class_project_coincident.json

>>> Done
```
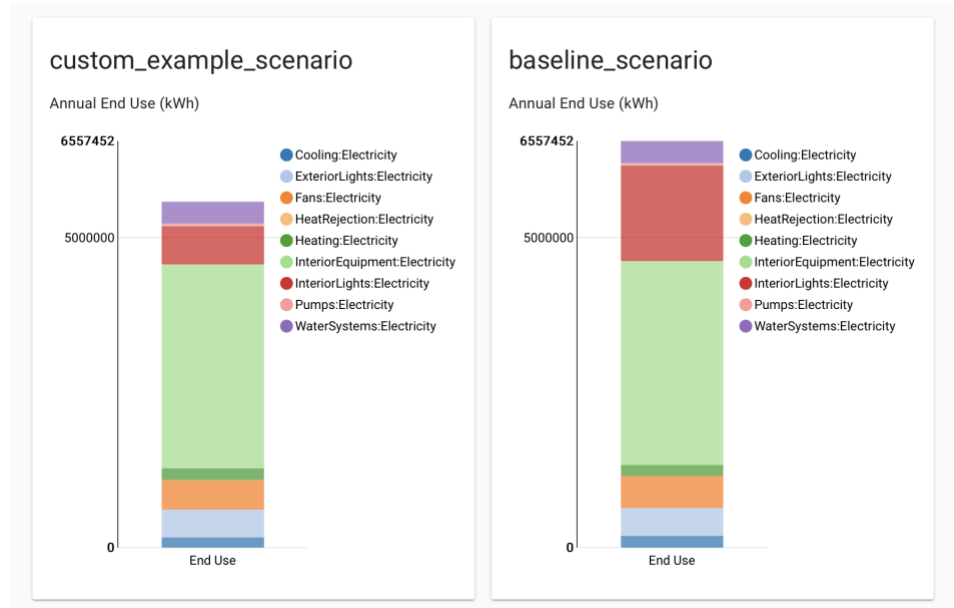
Below is an example of changing Lighting Power Density and the graph created in the scenario_comparison.html. It is recommended that custom scripts be generated to post process the results for more flexible and customized plotting. This can be done in Microsoft Excel or Python/Jupyter Notebooks.

9. (60 mins) Assess your Measures: Analyze the effectiveness of your added measures for the coincident and diverse projects by examining the results located in the Project "run" folder:
10. (120 mins) Modify and Iterate: If your chosen measures were not very effective in achieving your goals, choose other measures (or modify existing measure arguments) that you think may have a better chance of success.
    a. Present a comparison of the cumulative annual electricity consumption and a chosen 3-day period between the baseline scenario and class measures scenario results to display the benefits of your added measures.
11. (60 mins) Write a Report: Summarize your methodology and the effects of the measures chosen for the coincident and diverse projects (including net annual energy and daily power profile). Make final recommendations on which energy flexibility or efficiency measures to implement for the district based on your modeling and optimization results. Your analysis must include energy and installation upgrades cost comparisons, regardless of your specific evaluation criteria. Use of an executive summary is recommended.

Deliverables (Week 4-5)

- Your final ClassProject.rb mapper files used in this project
- A 1-page proposal as requested in Task #4
- A report as requested in Task #11