ResStock Documentation

NREL

Feb 03, 2023

Contents:

Note: Most people who use ResStock for analysis do so by using the datasets published to the ResStock Data Viewer. These datasets are often the best choice because they are a verified run that can be referenced. Datasets with upgrade scenario results will be made available over time.

For more complicated analyses including custom upgrade scenarios, the source code of ResStock and this documentation page are available to run your own simulations. However, running ResStock can be a complicated affair and can require a lot of computing resources. Running on Amazon Web Services requires some technical knowledge of cloud computing to deploy. Unfortunately, we don't have the resources to provide technical support if you choose this route.

CHAPTER 1

Tutorial

1.1 Installation

1.1.1 Download ResStock

There are two options for downloading ResStock:

1. download a released version
2. clone the repository

For the first option, go to the releases page on GitHub and select a release. Note the OpenStudio version requirements associated with each version of ResStock. For example, ResStock v2.4.0 requires that you have OpenStudio v2.9.0 installed.

For the second option, you will need to have Git or some other Git-based tool installed. Cloning the ResStock repository gives you access to the `develop` branch of ResStock. The `develop` branch is under active development.

Note: If you are planning to perform large-scale runs on ResStock (greater than 1000 simulations) or analyze timeseries data, you will need to use *buildstockbatch* to run and manage batch simulations of ResStock. Buildstockbatch can be run locally via Docker, on AWS, or on an HPC like NREL's Eagle. Installation instructions can be found in buildstockbatch's installation documentation.

If you are planning to perform small-scale runs of ResStock (1000 simulations or fewer), you can use the Docker-less option of running locally via *run_analysis.rb*.

1.1.2 Install OpenStudio

Download the version of OpenStudio software (corresponding to the ResStock version that has been selected) from the OpenStudio developer website.

1.1.3 Developer instructions

If you will be developing residential measures and testing residential building models, see the *Advanced Tutorial*. If you are a developer, make sure that you have checked out the `develop` branch of the repository.

1.2 Architecture

The key ResStock workflow components are described below.

1.2.1 Projects

At the top level of the ResStock repository you just downloaded, you will see two analysis project folders:

- `project_national`
- `project_testing`

The national project contains inputs describing the existing residential building stock. The testing project contains inputs to test our OpenStudio workflows. Within each project folder are sample "baseline" and "upgrades" input files that may serve as examples for how to set up different types of ResStock analyses. The contents of the input file ultimately determines the set of workflow steps (i.e., OpenStudio measures) for each ResStock sample. See *Run the Project* for more information about running ResStock analyses.

1.2.2 Sampling

To run the sampling script yourself, from the command line execute, e.g. `ruby resources/run_sampling.rb -p project_national -n 10000 -o buildstock.csv`, and a file `buildstock.csv` will be created in the `resources` directory.

If a custom `buildstock.csv` file is located in a project's `housing_characteristics` directory when you run the project, it will automatically be used to generate simulations. If it's not found, the sampling will be run automatically to create one. For each datapoint, the measure will then look up its building description from the sampled csv.

You can use this manual sampling process to downselect which simulations you want to run. For example, you can use the command above to generate a `buildstock.csv` for the entire U.S. and then open up this file in Excel and delete all of the rows that you don't want to simulate (e.g., all rows that aren't in New York). Keep in mind that if you do this, you will need to re-enumerate the "Building" column as "1" through the number of rows.

1.2.3 Measures

ResStock uses a mixture of both OpenStudio Model and Reporting measures in its workflow. The following depicts the order in which workflow measure steps are applied:

Index	Measure	Measure Type	Optional	Notes	Source
1	BuildExistingModel	Model	No	Meta measure	ResStock
2	ApplyUpgrade	Model	Yes[1]	Meta measure	ResStock
3	HPXMLtoOpenStudio	Model	No		OS-HPXML[2]
4	*Other Model Measures*	Model	Yes		Any[3]
5	ReportSimulationOutput	Reporting	No		OS-HPXML
6	ReportHPXMLOutput	Reporting	No		ResStock
7	ReportUtilityBills	Reporting	No		OS-HPXML
8	UpgradeCosts	Reporting	No		ResStock
9	*Other Reporting Measures*	Reporting	Yes		Any[4]
10	ServerDirectoryCleanup	Reporting	No		ResStock

The BuildExistingModel and ApplyUpgrade meta measures call the following model measures:

Index	Measure	Measure Type	Optional	Notes	Source
1	ResStockArguments	Model	No		ResStock
2	BuildResidentialHPXML	Model	No		OS-HPXML
3	BuildResidentialScheduleFile	Model	No		OS-HPXML

Model Measures

Model measures are applied *before* the simulation is run. They contribute to the generation of the model.

BuildExistingModel

BuildExistingModel is a meta measure; meaning, it incrementally applies other measures (i.e., ResStockArguments, BuildResidentialHPXML, and BuildResidentialScheduleFile) to create "baseline" residential models.

Builds the OpenStudio Model for an existing building.

Builds the OpenStudio Model using the sampling csv file, which contains the specified parameters for each existing building. Based on the supplied building number, those parameters are used to run the OpenStudio measures with appropriate arguments and build up the OpenStudio model.

ResStockArguments

Measure that pre-processes the arguments passed to the BuildResidentialHPXML and BuildResidentialScheduleFile measures.

Passes in all arguments from the options lookup, processes them, and then registers values to the runner to be used by other measures.

BuildResidentialHPXML

Builds a residential HPXML file.

Note: OS-HPXML default values can be found in the OS-HPXML documentation or can be seen by using the 'apply_defaults' argument.

BuildResidentialScheduleFile

Builds a residential schedule file.

[1] Baseline models with no upgrades do not have the ApplyUpgrade measure applied.
[2] OS-HPXML refers to the OpenStudio-HPXML repository.
[3] *Other Model Measures* do not need to originate from ResStock, but it is up to the user to ensure they work within the ResStock workflow.
[4] *Other Reporting Measures* do not need to originate from ResStock, but it is up to the user to ensure they work within the ResStock workflow.

Generates a CSV of schedules at the specified file path, and inserts the CSV schedule file path into the output HPXML file (or overwrites it if one already exists). Stochastic schedules are generated using time-inhomogeneous Markov chains derived from American Time Use Survey data, and supplemented with sampling duration and power level from NEEA RBSA data as well as DHW draw duration and flow rate from Aquacraft/AWWA data.

ApplyUpgrade

This measure can be optionally applied to the workflow. Like the BuildExistingModel measure, ApplyUpgrade is a meta measure; it, too, incrementally applies other measures (i.e., ResStockArguments, BuildResidentialHPXML, and BuildResidentialScheduleFile) to create "upgraded" residential models.

Measure that applies an upgrade (one or more child measures) to a building model based on the specified logic.

Determines if the upgrade should apply to a given building model. If so, calls one or more child measures with the appropriate arguments.

Each instance of the ApplyUpgrade measure defines an upgrade scenario. An upgrade scenario is a collection of options exercised with some logic and costs applied. In the simplest case, we apply the new option to all housing units. The available upgrade options are in `resources/options_lookup.tsv` in your git repository. For this example, we will upgrade all windows by applying the `Windows|Triple, Low-E, Non-metal, Air, L-Gain` option to all houses across the country. We do this by entering that in the **Option 1** box on the Apply Upgrade measure. Also, we'll give the upgrade scenario a name: "Triple-Pane Windows" and a cost of \$40/ft^2 of window area by entering the number in **Option 1 Cost Value** and selecting "Window Area (ft^2)" for **Option 1 Cost Multiplier**. Like the **downselect logic**, excluded datapoints (i.e., datapoints for which the upgrade does not apply) will result in "completed invalid workflow". Note that using no downselect logic will apply the option to all housing units. For a full explanation of how to set up the options and logic surrounding them, see *Upgrade Scenario Configuration*.

HPXMLtoOpenStudio

Translates HPXML file to OpenStudio Model

See also OpenStudio-HPXML Workflow Inputs for documentation on workflow inputs.

Other Model Measures

Additional model measures can be optionally applied to the workflow. They are applied following generation of the model, but before any reporting measures.

Reporting Measures

Reporting measures are applied *after* the simulation is run. They process and report simulation output.

ReportSimulationOutput

Reports simulation outputs for residential HPXML-based models.

Processes EnergyPlus simulation outputs in order to generate an annual output file and an optional timeseries output file.

ReportHPXMLOutput

Reports HPXML outputs for residential HPXML-based models.

Parses the HPXML file and reports pre-defined outputs.

ReportUtilityBills

Calculates and reports utility bills for residential HPXML-based models.

Calculate electric/gas utility bills based on monthly fixed charges and marginal rates. Calculate other utility bills based on marginal rates for oil, propane, wood cord, wood pellets, and coal. User can specify PV compensation types of 'Net-Metering' or 'Feed-In Tariff', along with corresponding rates and connection fees.

UpgradeCosts

Measure that calculates upgrade costs.

Multiplies cost value by cost multiplier.

Other Reporting Measures

Additional reporting measures (e.g., QOIReport) can be optionally applied to the workflow. They are applied following all standard reporting measures, but before the ServerDirectoryCleanup measure.

ServerDirectoryCleanup

Optionally removes a significant portion of the saved results from each run, helping to alleviate memory problems.

1.3 Run the Project

Both buildstockbatch and run_analysis.rb can used to run ResStock analyses. They use a common project definition, the YAML file, to provide the details of the analysis. See the Residential HPXML Workflow Generator documentation page for more information.

1.3.1 Using buildstockbatch

See the BuildStock Batch documentation for information on running projects (large-scale).

1.3.2 Using run_analysis.rb

You also have the option of running (small-scale) projects using the OpenStudio Command Line Interface (CLI) with buildstockbatch yml input files. This method needs only the OpenStudio CLI.

The primary differences relative to buildstockbatch are:

1. `Debugging`: Individual, or sets of (e.g., 1, 2, 4), building ID(s) from the entire sampled space can be run. (See the related `--building_id` argument below.)
2. `Convenience`: Simulation input files (both OSW and HPXML) are collected and stored on-the-fly. (See the related `--debug` argument below.)
3. `Size`: Folders containing intermediate files can be either preserved or successively overwritten. (See the related `--keep_run_folders` argument below.)
4. `Accessibility`: Simulation input and output files are organized and stored differently, and are not tarred or compressed.

Call the OpenStudio CLI with the provided `workflow/run_analysis.rb` script. For example: `openstudio workflow/run_analysis.rb -y project_testing/testing_baseline.yml` The previous command samples from `project_testing` and runs simulations using baseline workflows generated from the specified yml file. An "output directory" (as specified in the yml file) is created with all input (OSW and HPXML) files and simulation results.

Note: If the `openstudio` command is not found, it's because the executable is not in your PATH. Either add the executable to your PATH or point directly to the executable found in the openstudio-X.X.X/bin directory.

You can also request that only measures are applied (i.e., no simulations are run) using the `--measures_only` flag. For example: `openstudio workflow/run_analysis.rb -y project_testing/testing_baseline.yml -m`

Run `openstudio workflow/run_analysis.rb -h` to see all available commands/arguments:

```
$ openstudio workflow/run_analysis.rb -h
Usage: run_analysis.rb -y buildstockbatch.yml
 e.g., run_analysis.rb -y national_baseline.yml
    -y, --yml <FILE>                 YML file
    -n, --threads N                  Number of parallel simulations (defaults to processor␣
↪count)
    -m, --measures_only              Only run the OpenStudio and EnergyPlus measures
    -i, --building_id ID             Only run this building ID; can be called multiple times
    -k, --keep_run_folders           Preserve run folder for all datapoints
    -s, --samplingonly               Run the sampling only
    -d, --debug                      Preserve lib folder and "existing" xml/osw files
    -o, --overwrite                  Overwrite existing project directory
    -v, --version                    Display version
    -h, --help                       Display help
```

Note: At this time the `residential_quota_downselect` sampler with `resample` is not supported.

1.4 Conclusion

Congratulations, you have now completed your first ResStock analysis. See the other sections in this documentation for more advanced topics.

CHAPTER 2

Advanced Tutorial

This advanced tutorial describes the process for developing residential measures and testing residential building models. Reasons for wanting to develop residential measures include: customizing any of the existing residential modeling algorithms or adding new technology models.

At this point in the tutorial, it is assumed that you have checked out a new branch that is up-to-date with the `develop` of the ResStock repository. Optionally, you may have created a new project folder (i.e., copied an existing project folder) and modified the set of tsv files in its `housing_characteristics` folder.

If your changes are intended to be merged into the `develop` branch of the ResStock repository, a pull request review is required.

2.1 Modifying Probability Distributions

This section provides a description of the housing characteristics and their dependencies and options.

A particular building within the building stock has a set of characteristics (e.g., level of wall insulation, type of lighting, vintage, and a variety of different schedules). Each housing characteristic corresponds to a tab-separated value (tsv) file with the extension *.tsv*. These housing characteristics files are found in the `<project_folder>/housing_characteristics` directory. A housing characteristic defines the probability mass function (PMF) of that characteristic in the building stock.

$Pr(X = A_i) = P(A_i) > 0 \quad and \quad \sum_{A_i \in S_A} P(A_i) = 1 \quad for \quad i = 1 : n$

When sampling a discrete random variable X to create a representative building, X takes a particular **Option** A_i. All possible options are collected in the set $S_A = \{A_0, A_1, ..., A_n\}$ and is size n. Since these are probabilities, the entries $P(A_i)$ must be greater than 0 and the probability of all possible options must sum to 1.

For example, a set of options for a building's vintage (when the building was built) may be the following:

$S_A = < 1950, 1950s, 1960s, 1970s, 1980s, 1990s, 2000s.$

Then the probability mass function may look like the following:

A_i	<1950	1950s	1960s	1970s	1980s	1990s	2000s
$P(X = A_i)$	0.020	0.060	0.090	0.230	0.370	0.130	0.090

Where the probability of a building having a given vintage in this example is

- 2% built before 1950,
- 6% in the 1950s,
- 9% in the 1960s,
- 23% in the 1970s,
- 37% in the 1980s,
- 13% in the 1900s, and
- 9% in the 2000s.

However, housing characteristics can have a **Dependency**, B_i, to another housing characteristic. All possible values of the dependency are collected in the set $S_B = B_0, B_1, ...B_m$ which is size m. If the **Option** of interest A_j and the **Dependency** B_i is known to have occurred when sampling X in the creation of a representative building, then conditional probability of A_j given B_i is usually written $P(A_j|B_i) = P_{B_i}(A_j)$.

Using the example from before, the PMF of the vintage depends on location of the particular building stock (which is represented by EPW weather files). In this example the vintage housing characteristic is examined. The first three lines in the `<project_folder>/housing_characteristics/Vintage.tsv` are shown in the table below.

	Location EPW (S_B)	<1950	1950s	1960s	1970s	1980s	1990s	2000s
$P(B_0\|A_j)$	USA_FL_Key.West.Intl.AP.722010_TMY3.epw	0.02	0.06	0.09	0.23	0.37	0.13	0.09
$P(B_1\|A_j)$	USA_FL_Miami.Intl.AP.722020_TMY3.epw	0.05	0.13	0.13	0.18	0.17	0.18	0.16

The vintage is dependent on the EPW location. The vintage discrete PMF that uses the Key West International Airport weather file, B_0, is defined by the following distribution:

- 2% built before 1950,
- 6% in the 1950s,
- 9% in the 1960s,
- 23% in the 1970s,
- 37% in the 1980s,
- 13% in the 1900s, and
- 9% in the 2000s.

While the vintage PMF that uses the Miami International Airport weather file, B_1 is defined by the following distribution:

- 5% built before 1950,
- 13% in the 1950s,
- 9% in the 1960s,
- 13% in the 1970s,
- 18% in the 1980s,
- 17% in the 1900s, and
- 18% in the 2000s.

The **Options** can correspond to a Measure in OpenStudio or can be used as a **Dependency** for other housing characteristics. For the list of available options for a given housing characteristic, see the `resources/options_lookup.tsv` file. In this file the "Parameter Name" corresponds to the housing characteristic, the "Option Name" corresponds to an available option for the housing characteristic, the "Measure Dir" corresponds to the OpenStudio Measure being used, and the following columns

correspond to different arguments needed by the OpenStudio Measure. Each option used in the housing characteristics tsv files must be in this `resources/options_lookup.tsv`. These options can be modified by the user to model their particular building stock.

2.2 Writing Housing Characteristics

This section provides a description of the standard format for the housing characteristics. In order to stop recommitting entire files and to keep differences easy to read during the review process, a standard format for writing housing characteristics has been created. All housing characteristics shall follow these format guidelines:

2.2.1 Guidelines

1. All lines have the line ending characters '\r\n' (i.e., "crlf" or "carriage return, line feed").
2. Comment lines in the housing characteristics file, indicated by the "#" symbol, can be added.

2.3 Installer Setup

After you have downloaded the OpenStudio installer, you may want to optionally install Ruby (2.7.2). This will allow you to execute rake tasks contained in the Rakefile. Follow the instructions below for *Windows Setup* or *Mac Setup*.

2.3.1 Windows Setup

1. Install Ruby (2.7.2). Follow the installation instructions here ("Optional - Install Ruby").
2. Run `gem install bundler -v 1.17.1`.

Note: If you get an error, you may have to issue the following: `gem sources -r https://rubygems.org/` followed by `gem sources -a http://rubygems.org/`. If you still get an error, manually update your gem sources list by including a config file named ".gemrc" in your home directory (.e.g, /c/Users/<USERNAME>) with the following contents:

```
---
:backtrace: false
:bulk_threshold: 1000
:sources:
- http://rubygems.org
:update_sources: true
:verbose: true
```

3. Download the DevKit at http://rubyinstaller.org/downloads/ (e.g., DevKit-mingw64-64-4.7.2-20130224-1432-sfx.exe). Choose either the 32-bit or 64-bit version depending on which version of Ruby you installed. Run the installer and extract to a directory (e.g., C:\RubyDevKit). Go to this directory, run `ruby dk.rb init`, modify the config.yml file as needed, and finally run `ruby dk.rb install`.
4. Run `bundle install` from the resstock directory. (If you get an error, check that `git` is in your `PATH` and that you are using the correct version of Ruby (2.7.2).)

2.3.2 Mac Setup

Install Homebrew if you don't have it already.

Run `brew doctor`. It should give you, among other issues, a list of unexpected dylibs that you'll need to move for this to work such as:

```
Unexpected dylibs:
  /usr/local/lib/libcrypto.0.9.8.dylib
  /usr/local/lib/libcrypto.1.0.0.dylib
  /usr/local/lib/libcrypto.dylib
  /usr/local/lib/libklcsagt.dylib
  /usr/local/lib/libklcskca.dylib
  /usr/local/lib/libklcsnagt.dylib
  /usr/local/lib/libklcsrt.dylib
  /usr/local/lib/libklcsstd.dylib
  /usr/local/lib/libklcstr.dylib
  /usr/local/lib/libklmspack.0.1.0.dylib
  /usr/local/lib/libklmspack.0.dylib
  /usr/local/lib/libklmspack.dylib
  /usr/local/lib/libssl.0.9.8.dylib
  /usr/local/lib/libssl.1.0.0.dylib
  /usr/local/lib/libssl.dylib
  /usr/local/lib/libz.1.2.5.dylib
  /usr/local/lib/libz.1.2.6.dylib
  /usr/local/lib/libz.1.dylib
  /usr/local/lib/libz.dylib
```

Highlight and copy the list (without the header "Unexpected dylibs:"). Run the following commands to move them to another location where they won't interfere.

```
mkdir ~/unused_dylibs
pbpaste | xargs -t -I % mv % ~/unused_dylibs
```

Install `rbenv` and required dependencies.

```
brew install openssl libyaml libffi rbenv
```

Initialize `rbenv` by running the command below and following the instructions to add the appropriate things to your `~/.bash_profile`.

```
rbenv init
```

Install the appropriate ruby version.

```
cd path/to/repo
rbenv install `cat .ruby-version`
```

Add the path to the install ruby libraries top the bottom of your `~/.bash_profile`

```
echo "export RUBYLIB=/Applications/OpenStudio-3.2.1/Ruby" >> ~/.bash_profile
echo "export ENERGYPLUS_EXE_PATH=\"/Applications/OpenStudio-3.2.1/EnergyPlus/energyplus-9.5.0\"
↪"
```

Install bundler and the libraries that bundler installs.

```
gem install bundler -v 1.17.1
bundle install
```

2.4 Tasks

Run `openstudio tasks.rb` to see available task commands:

```
$ openstudio tasks.rb
ERROR: Missing command.
Usage: openstudio tasks.rb [COMMAND]
Commands:
  update_measures
  integrity_check_national
  integrity_check_testing
  download_weather
```

2.4.1 Update Measures

Use `openstudio tasks.rb update_measures` to apply rubocop auto-correct to measures, and to update measure.xml files:

```
$ openstudio tasks.rb update_measures
Applying rubocop auto-correct to measures...
Running RuboCop...

91 files inspected, no offenses detected
Updating measure.xmls...
Done.
```

2.4.2 Integrity Checks

Run `openstudio tasks.rb integrity_check_<project_name>`, where `<project_name>` matches the project you are working with. If no task exists for the project you are working with, extend the list of integrity check tasks to accommodate your project by modifying the `tasks.rb` file. An example for running a project's integrity checks is given below:

```
$ openstudio tasks.rb integrity_check_national
Checking for issues with project_national/Location Region...
Checking for issues with project_national/Location EPW...
Checking for issues with project_national/Vintage...
Checking for issues with project_national/Heating Fuel...
Checking for issues with project_national/Usage Level...
...
```

If the integrity check for a given project fails, you will need to update either your tsv files and/or the `resources/options_lookup.tsv` file. See *Options Lookup* for information about the `options_lookup.tsv` file.

2.4.3 Download Weather

Run `openstudio tasks.rb download_weather` to download available EPW weather files:

```
$ /c/openstudio-3.4.0/bin/openstudio.exe tasks.rb download_weather
Downloading /files/156/BuildStock_TMY3_FIPS.zip (  1%)
Downloading /files/156/BuildStock_TMY3_FIPS.zip (  2%)
Downloading /files/156/BuildStock_TMY3_FIPS.zip (  3%)
...
```

2.4.4 Rakefile

Once you have completed instructions found in *Installer Setup*, you can then use the Rakefile contained at the top level of this repository. You can run rake task(s) for *performing integrity checks on project inputs* as well as executing various tests.

Run `rake -T` to see the list of possible rake tasks. The `-T` is replaced with the chosen task.

```
$ rake -T
rake unit_tests:integrity_check_tests     # Run tests for integrity_check_t...
rake unit_tests:measure_tests             # Run tests for measure_tests
rake unit_tests:project_integrity_checks  # Run tests for project_integrity...
rake workflow:analysis_tests              # Run tests for analysis_tests
```

2.5 Options Lookup

The `options_lookup.tsv` file, found in the `resources` folder, specifies mappings from sampled options into measure arguments. For example, if the distribution of cooling system types in `HVAC System Cooling.tsv` has `Option=AC, SEER 13` and `Option=AC, SEER 15`, but you want to include a `Option=AC, SEER 17` option, you would add that option as a column in `HVAC System Cooling.tsv` and then create a corresponding row in `options_lookup.tsv`. Updates to this file will allow you to avoid hitting the following types of integrity check errors:

- *Could not find parameter and option*
- *Required argument not provided*

2.5.1 Could not find parameter and option

You do not have a row in `options_lookup.tsv` for a particular option that is sampled.

An example of this error is given below:

```
$ openstudio tasks.rb integrity_check_testing
...
Error executing argv: ["integrity_check_testing"]
Error: ERROR: Could not find parameter 'Insulation Wall' and option 'Wood Stud, Uninsulated'␣
↪in C:/OpenStudio/resstock/test/../resources/options_lookup.tsv.
```

2.5.2 Required argument not provided

For the particular option that is sampled, your corresponding measure is missing an argument value assignment.

An example of this error is given below:

```
$ openstudio tasks.rb integrity_check_testing
...
Error executing argv: ["integrity_check_testing"]
Error: ERROR: Required argument 'wall_assembly_r' not provided in C:/OpenStudio/resstock/test/.
↪./resources/options_lookup.tsv for measure 'ResStockArguments'.
```

2.6 Increasing Upgrade Options

To allow more options per upgrade, increase the value returned by the following method defined in `measures/ApplyUpgrade/resources/constants.rb`:

```
def self.NumApplyUpgradeOptions
  return 25
end
```

Then run `openstudio tasks.rb update_measures`. See *Tasks* for instructions on how to run tasks.

CHAPTER 3

Upgrade Scenario Configuration

There is quite a bit more flexibility and capability in defining an upgrade scenario than was discussed in the *tutorial*. Here we will go through each field in the **Apply Upgrade** measure and discuss how it can be used to build more complicated real-life scenarios for upgrades.

3.1 Upgrade Name

This is a human readable name for the upgrade scenario. Something like, "Replace electric furnaces with Energy Star heat pumps" or "Insulate attics to R-49".

3.2 Option <#>

In this field we enter the parameter and option combination to be applied. In the upgrade scenario simulations, this option will replace the option for the corresponding parameter in the baseline run. These can be found and referenced in the `resources/options_lookup.tsv` file in your local git repository. You can see the most updated version on github here, but it's recommended to use your local version as it will be synchronized with your project. The file can be opened in a spreadsheet editor like Excel for viewing.

The text to enter in the field will be the Parameter Name followed by the Option Name separated by a pipe character.

```
Insulation Wall|Wood Stud, R-36
```

3.3 Option <#> Apply Logic

The apply logic field specifies the conditions under which the option will apply based on the baseline building's options. To specify the condition(s) include one or more `parameter|option` pairs from `options_lookup.tsv`. Multiple option conditions can be joined using the following logical operators. Parentheses may be used as necessary as well.

`||`	logical OR
`&&`	logical AND
!	logical NOT

A few examples will illustrate. First, lets say we want the apply the option `Water Heater|Gas Tankless`, but only for water heaters that are worse and also use gas. We would use the following apply logic:

```
Water Heater|Gas Standard||Water Heater|Gas Benchmark
```

Or say we want to apply the upgrade only to houses with 3 car garages that aren't in New England.

```
(!Location Census Division|New England)&&(Geometry Garage|3 Car)
```

Currently, you can enter up to 25 options per upgrade. To allow additional options per upgrade you would need to update a method defined in a resource file, run a rake task, and update the outputs section for all PAT projects. See *Increasing Upgrade Options* for more information.

3.4 Option <#> Cost <#>

This is the cost of the upgrade. Multiple costs can be entered and each is multiplied by a cost multiplier, described below.

3.5 Option <#> Cost <#> Multiplier

The cost above is multiplied by this value, which is a function of the building. Since there can be multiple costs (currently 2), this permits both fixed and variable costs for upgrades that depend on the properties of the baseline house.

See the *Upgrade Costs* workflow outputs for a list of all available multiplier types.

3.6 Package Apply Logic

This is where to specifiy logic to determine whether the whole package of upgrades is applied (all of the options together). It uses the same format as *Option <#> Apply Logic*.

CHAPTER 4

Available Outputs

ResStock generates a default set of housing characteristics, simulation output, and upgrade cost outputs. ResStock optionally generates outputs related to timeseries data, component loads, emissions, utility bills, and quantities of interest.

4.1 Default Outputs

The default set of outputs include housing characteristics, annual simulation outputs, and upgrade cost information.

4.1.1 Housing Characteristics

Default characteristics include sampled properties for each dwelling unit.

build_existing_model.ahs_region
build_existing_model.aiannh_area
build_existing_model.area_median_income
build_existing_model.ashrae_iecc_climate_zone_2004
build_existing_model.ashrae_iecc_climate_zone_2004_2_a_split
build_existing_model.bathroom_spot_vent_hour
build_existing_model.bedrooms
build_existing_model.building_america_climate_zone
build_existing_model.cec_climate_zone
build_existing_model.ceiling_fan
build_existing_model.census_division
build_existing_model.census_division_recs
build_existing_model.census_region
build_existing_model.city
build_existing_model.clothes_dryer
build_existing_model.clothes_washer
build_existing_model.clothes_washer_presence

Continued on next page

Table 1 – continued from previous page

build_existing_model.cooking_range
build_existing_model.cooling_setpoint
build_existing_model.cooling_setpoint_has_offset
build_existing_model.cooling_setpoint_offset_magnitude
build_existing_model.cooling_setpoint_offset_period
build_existing_model.corridor
build_existing_model.county
build_existing_model.county_and_puma
build_existing_model.dehumidifier
build_existing_model.dishwasher
build_existing_model.door_area
build_existing_model.doors
build_existing_model.ducts
build_existing_model.eaves
build_existing_model.electric_vehicle
build_existing_model.emissions_electricity_folders
build_existing_model.emissions_electricity_units
build_existing_model.emissions_electricity_values_or_filepaths
build_existing_model.emissions_fossil_fuel_units
build_existing_model.emissions_fuel_oil_values
build_existing_model.emissions_natural_gas_values
build_existing_model.emissions_propane_values
build_existing_model.emissions_scenario_names
build_existing_model.emissions_types
build_existing_model.emissions_wood_values
build_existing_model.federal_poverty_level
build_existing_model.generation_and_emissions_assessment_region
build_existing_model.geometry_attic_type
build_existing_model.geometry_building_horizontal_location_mf
build_existing_model.geometry_building_horizontal_location_sfa
build_existing_model.geometry_building_level_mf
build_existing_model.geometry_building_number_units_mf
build_existing_model.geometry_building_number_units_sfa
build_existing_model.geometry_building_type_acs
build_existing_model.geometry_building_type_height
build_existing_model.geometry_building_type_recs
build_existing_model.geometry_floor_area
build_existing_model.geometry_floor_area_bin
build_existing_model.geometry_foundation_type
build_existing_model.geometry_garage
build_existing_model.geometry_stories
build_existing_model.geometry_stories_low_rise
build_existing_model.geometry_story_bin
build_existing_model.geometry_wall_exterior_finish
build_existing_model.geometry_wall_type
build_existing_model.has_pv
build_existing_model.heating_fuel
build_existing_model.heating_setpoint
build_existing_model.heating_setpoint_has_offset
build_existing_model.heating_setpoint_offset_magnitude

Continued on next page

Table 1 – continued from previous page

build_existing_model.heating_setpoint_offset_period
build_existing_model.holiday_lighting
build_existing_model.hot_water_distribution
build_existing_model.hot_water_fixtures
build_existing_model.household_has_tribal_persons
build_existing_model.hvac_cooling_efficiency
build_existing_model.hvac_cooling_partial_space_conditioning
build_existing_model.hvac_cooling_type
build_existing_model.hvac_has_ducts
build_existing_model.hvac_has_shared_system
build_existing_model.hvac_has_zonal_electric_heating
build_existing_model.hvac_heating_efficiency
build_existing_model.hvac_heating_type
build_existing_model.hvac_heating_type_and_fuel
build_existing_model.hvac_secondary_heating_efficiency
build_existing_model.hvac_secondary_heating_type_and_fuel
build_existing_model.hvac_shared_efficiencies
build_existing_model.hvac_system_is_faulted
build_existing_model.hvac_system_single_speed_ac_airflow
build_existing_model.hvac_system_single_speed_ac_charge
build_existing_model.hvac_system_single_speed_ashp_airflow
build_existing_model.hvac_system_single_speed_ashp_charge
build_existing_model.income
build_existing_model.income_recs_2015
build_existing_model.income_recs_2020
build_existing_model.infiltration
build_existing_model.insulation_ceiling
build_existing_model.insulation_floor
build_existing_model.insulation_foundation_wall
build_existing_model.insulation_rim_joist
build_existing_model.insulation_roof
build_existing_model.insulation_slab
build_existing_model.insulation_wall
build_existing_model.interior_shading
build_existing_model.iso_rto_region
build_existing_model.lighting
build_existing_model.lighting_interior_use
build_existing_model.lighting_other_use
build_existing_model.location_region
build_existing_model.mechanical_ventilation
build_existing_model.misc_extra_refrigerator
build_existing_model.misc_freezer
build_existing_model.misc_gas_fireplace
build_existing_model.misc_gas_grill
build_existing_model.misc_gas_lighting
build_existing_model.misc_hot_tub_spa
build_existing_model.misc_pool
build_existing_model.misc_pool_heater
build_existing_model.misc_pool_pump
build_existing_model.misc_well_pump

Continued on next page

Table 1 – continued from previous page

build_existing_model.natural_ventilation
build_existing_model.neighbors
build_existing_model.occupants
build_existing_model.orientation
build_existing_model.overhangs
build_existing_model.plug_load_diversity
build_existing_model.plug_loads
build_existing_model.puma
build_existing_model.puma_metro_status
build_existing_model.pv_orientation
build_existing_model.pv_system_size
build_existing_model.radiant_barrier
build_existing_model.range_spot_vent_hour
build_existing_model.reeds_balancing_area
build_existing_model.refrigerator
build_existing_model.roof_material
build_existing_model.sample_weight
build_existing_model.simulation_control_run_period_begin_day_of_month
build_existing_model.simulation_control_run_period_begin_month
build_existing_model.simulation_control_run_period_calendar_year
build_existing_model.simulation_control_run_period_end_day_of_month
build_existing_model.simulation_control_run_period_end_month
build_existing_model.simulation_control_timestep
build_existing_model.solar_hot_water
build_existing_model.state
build_existing_model.tenure
build_existing_model.units_represented
build_existing_model.usage_level
build_existing_model.utility_bill_electricity_fixed_charges
build_existing_model.utility_bill_electricity_marginal_rates
build_existing_model.utility_bill_fuel_oil_fixed_charges
build_existing_model.utility_bill_fuel_oil_marginal_rates
build_existing_model.utility_bill_natural_gas_fixed_charges
build_existing_model.utility_bill_natural_gas_marginal_rates
build_existing_model.utility_bill_propane_fixed_charges
build_existing_model.utility_bill_propane_marginal_rates
build_existing_model.utility_bill_pv_compensation_types
build_existing_model.utility_bill_pv_feed_in_tariff_rates
build_existing_model.utility_bill_pv_monthly_grid_connection_fee_units
build_existing_model.utility_bill_pv_monthly_grid_connection_fees
build_existing_model.utility_bill_pv_net_metering_annual_excess_sellback_rate_types
build_existing_model.utility_bill_pv_net_metering_annual_excess_sellback_rates
build_existing_model.utility_bill_scenario_names
build_existing_model.utility_bill_wood_fixed_charges
build_existing_model.utility_bill_wood_marginal_rates
build_existing_model.vacancy_status
build_existing_model.vintage
build_existing_model.vintage_acs
build_existing_model.water_heater_efficiency
build_existing_model.water_heater_fuel

Continued on next page

Table 1 – continued from previous page

build_existing_model.water_heater_in_unit
build_existing_model.weather_file_city
build_existing_model.weather_file_latitude
build_existing_model.weather_file_longitude
build_existing_model.window_areas
build_existing_model.windows

4.1.2 Simulation Output

Default annual simulation outputs include energy consumptions (total, by fuel, and by end use), hot water uses, building loads, unmet hours, and peak building electricity/loads.

See the OpenStudio-HPXML Workflow Outputs section on Annual Outputs for more information about annual outputs.

report_simulation_output.end_use_coal_clothes_dryer_m_btu
report_simulation_output.end_use_coal_fireplace_m_btu
report_simulation_output.end_use_coal_generator_m_btu
report_simulation_output.end_use_coal_grill_m_btu
report_simulation_output.end_use_coal_heating_heat_pump_backup_m_btu
report_simulation_output.end_use_coal_heating_m_btu
report_simulation_output.end_use_coal_hot_water_m_btu
report_simulation_output.end_use_coal_lighting_m_btu
report_simulation_output.end_use_coal_mech_vent_preheating_m_btu
report_simulation_output.end_use_coal_range_oven_m_btu
report_simulation_output.end_use_electricity_battery_m_btu
report_simulation_output.end_use_electricity_ceiling_fan_m_btu
report_simulation_output.end_use_electricity_clothes_dryer_m_btu
report_simulation_output.end_use_electricity_clothes_washer_m_btu
report_simulation_output.end_use_electricity_cooling_fans_pumps_m_btu
report_simulation_output.end_use_electricity_cooling_m_btu
report_simulation_output.end_use_electricity_dehumidifier_m_btu
report_simulation_output.end_use_electricity_dishwasher_m_btu
report_simulation_output.end_use_electricity_electric_vehicle_charging_m_btu
report_simulation_output.end_use_electricity_freezer_m_btu
report_simulation_output.end_use_electricity_generator_m_btu
report_simulation_output.end_use_electricity_heating_fans_pumps_m_btu
report_simulation_output.end_use_electricity_heating_heat_pump_backup_m_btu
report_simulation_output.end_use_electricity_heating_m_btu
report_simulation_output.end_use_electricity_hot_tub_heater_m_btu
report_simulation_output.end_use_electricity_hot_tub_pump_m_btu
report_simulation_output.end_use_electricity_hot_water_m_btu
report_simulation_output.end_use_electricity_hot_water_recirc_pump_m_btu
report_simulation_output.end_use_electricity_hot_water_solar_thermal_pump_m_btu
report_simulation_output.end_use_electricity_lighting_exterior_m_btu
report_simulation_output.end_use_electricity_lighting_garage_m_btu
report_simulation_output.end_use_electricity_lighting_interior_m_btu
report_simulation_output.end_use_electricity_mech_vent_m_btu
report_simulation_output.end_use_electricity_mech_vent_precooling_m_btu
report_simulation_output.end_use_electricity_mech_vent_preheating_m_btu
report_simulation_output.end_use_electricity_plug_loads_m_btu

Continued on next page

Table 2 – continued from previous page

report_simulation_output.end_use_electricity_pool_heater_m_btu
report_simulation_output.end_use_electricity_pool_pump_m_btu
report_simulation_output.end_use_electricity_pv_m_btu
report_simulation_output.end_use_electricity_range_oven_m_btu
report_simulation_output.end_use_electricity_refrigerator_m_btu
report_simulation_output.end_use_electricity_television_m_btu
report_simulation_output.end_use_electricity_well_pump_m_btu
report_simulation_output.end_use_electricity_whole_house_fan_m_btu
report_simulation_output.end_use_fuel_oil_clothes_dryer_m_btu
report_simulation_output.end_use_fuel_oil_fireplace_m_btu
report_simulation_output.end_use_fuel_oil_generator_m_btu
report_simulation_output.end_use_fuel_oil_grill_m_btu
report_simulation_output.end_use_fuel_oil_heating_heat_pump_backup_m_btu
report_simulation_output.end_use_fuel_oil_heating_m_btu
report_simulation_output.end_use_fuel_oil_hot_water_m_btu
report_simulation_output.end_use_fuel_oil_lighting_m_btu
report_simulation_output.end_use_fuel_oil_mech_vent_preheating_m_btu
report_simulation_output.end_use_fuel_oil_range_oven_m_btu
report_simulation_output.end_use_natural_gas_clothes_dryer_m_btu
report_simulation_output.end_use_natural_gas_fireplace_m_btu
report_simulation_output.end_use_natural_gas_generator_m_btu
report_simulation_output.end_use_natural_gas_grill_m_btu
report_simulation_output.end_use_natural_gas_heating_heat_pump_backup_m_btu
report_simulation_output.end_use_natural_gas_heating_m_btu
report_simulation_output.end_use_natural_gas_hot_tub_heater_m_btu
report_simulation_output.end_use_natural_gas_hot_water_m_btu
report_simulation_output.end_use_natural_gas_lighting_m_btu
report_simulation_output.end_use_natural_gas_mech_vent_preheating_m_btu
report_simulation_output.end_use_natural_gas_pool_heater_m_btu
report_simulation_output.end_use_natural_gas_range_oven_m_btu
report_simulation_output.end_use_propane_clothes_dryer_m_btu
report_simulation_output.end_use_propane_fireplace_m_btu
report_simulation_output.end_use_propane_generator_m_btu
report_simulation_output.end_use_propane_grill_m_btu
report_simulation_output.end_use_propane_heating_heat_pump_backup_m_btu
report_simulation_output.end_use_propane_heating_m_btu
report_simulation_output.end_use_propane_hot_water_m_btu
report_simulation_output.end_use_propane_lighting_m_btu
report_simulation_output.end_use_propane_mech_vent_preheating_m_btu
report_simulation_output.end_use_propane_range_oven_m_btu
report_simulation_output.end_use_wood_cord_clothes_dryer_m_btu
report_simulation_output.end_use_wood_cord_fireplace_m_btu
report_simulation_output.end_use_wood_cord_generator_m_btu
report_simulation_output.end_use_wood_cord_grill_m_btu
report_simulation_output.end_use_wood_cord_heating_heat_pump_backup_m_btu
report_simulation_output.end_use_wood_cord_heating_m_btu
report_simulation_output.end_use_wood_cord_hot_water_m_btu
report_simulation_output.end_use_wood_cord_lighting_m_btu
report_simulation_output.end_use_wood_cord_mech_vent_preheating_m_btu
report_simulation_output.end_use_wood_cord_range_oven_m_btu

Continued on next page

Table 2 – continued from previous page

report_simulation_output.end_use_wood_pellets_clothes_dryer_m_btu
report_simulation_output.end_use_wood_pellets_fireplace_m_btu
report_simulation_output.end_use_wood_pellets_generator_m_btu
report_simulation_output.end_use_wood_pellets_grill_m_btu
report_simulation_output.end_use_wood_pellets_heating_heat_pump_backup_m_btu
report_simulation_output.end_use_wood_pellets_heating_m_btu
report_simulation_output.end_use_wood_pellets_hot_water_m_btu
report_simulation_output.end_use_wood_pellets_lighting_m_btu
report_simulation_output.end_use_wood_pellets_mech_vent_preheating_m_btu
report_simulation_output.end_use_wood_pellets_range_oven_m_btu
report_simulation_output.energy_use_net_m_btu
report_simulation_output.energy_use_total_m_btu
report_simulation_output.fuel_use_coal_total_m_btu
report_simulation_output.fuel_use_electricity_net_m_btu
report_simulation_output.fuel_use_electricity_total_m_btu
report_simulation_output.fuel_use_fuel_oil_total_m_btu
report_simulation_output.fuel_use_natural_gas_total_m_btu
report_simulation_output.fuel_use_propane_total_m_btu
report_simulation_output.fuel_use_wood_cord_total_m_btu
report_simulation_output.fuel_use_wood_pellets_total_m_btu
report_simulation_output.hot_water_clothes_washer_gal
report_simulation_output.hot_water_dishwasher_gal
report_simulation_output.hot_water_distribution_waste_gal
report_simulation_output.hot_water_fixtures_gal
report_simulation_output.hvac_capacity_cooling_btu_h
report_simulation_output.hvac_capacity_heat_pump_backup_btu_h
report_simulation_output.hvac_capacity_heating_btu_h
report_simulation_output.hvac_design_load_cooling_latent_ducts_btu_h
report_simulation_output.hvac_design_load_cooling_latent_infiltration_ventilation_btu_h
report_simulation_output.hvac_design_load_cooling_latent_internal_gains_btu_h
report_simulation_output.hvac_design_load_cooling_latent_total_btu_h
report_simulation_output.hvac_design_load_cooling_sensible_ceilings_btu_h
report_simulation_output.hvac_design_load_cooling_sensible_doors_btu_h
report_simulation_output.hvac_design_load_cooling_sensible_ducts_btu_h
report_simulation_output.hvac_design_load_cooling_sensible_floors_btu_h
report_simulation_output.hvac_design_load_cooling_sensible_infiltration_ventilation_btu_h
report_simulation_output.hvac_design_load_cooling_sensible_internal_gains_btu_h
report_simulation_output.hvac_design_load_cooling_sensible_roofs_btu_h
report_simulation_output.hvac_design_load_cooling_sensible_skylights_btu_h
report_simulation_output.hvac_design_load_cooling_sensible_slabs_btu_h
report_simulation_output.hvac_design_load_cooling_sensible_total_btu_h
report_simulation_output.hvac_design_load_cooling_sensible_walls_btu_h
report_simulation_output.hvac_design_load_cooling_sensible_windows_btu_h
report_simulation_output.hvac_design_load_heating_ceilings_btu_h
report_simulation_output.hvac_design_load_heating_doors_btu_h
report_simulation_output.hvac_design_load_heating_ducts_btu_h
report_simulation_output.hvac_design_load_heating_floors_btu_h
report_simulation_output.hvac_design_load_heating_infiltration_ventilation_btu_h
report_simulation_output.hvac_design_load_heating_roofs_btu_h
report_simulation_output.hvac_design_load_heating_skylights_btu_h

Continued on next page

Table 2 – continued from previous page

report_simulation_output.hvac_design_load_heating_slabs_btu_h
report_simulation_output.hvac_design_load_heating_total_btu_h
report_simulation_output.hvac_design_load_heating_walls_btu_h
report_simulation_output.hvac_design_load_heating_windows_btu_h
report_simulation_output.hvac_design_temperature_cooling_f
report_simulation_output.hvac_design_temperature_heating_f
report_simulation_output.load_cooling_delivered_m_btu
report_simulation_output.load_heating_delivered_m_btu
report_simulation_output.load_hot_water_delivered_m_btu
report_simulation_output.load_hot_water_desuperheater_m_btu
report_simulation_output.load_hot_water_solar_thermal_m_btu
report_simulation_output.load_hot_water_tank_losses_m_btu
report_simulation_output.peak_electricity_summer_total_w
report_simulation_output.peak_electricity_winter_total_w
report_simulation_output.peak_load_cooling_delivered_k_btu_hr
report_simulation_output.peak_load_heating_delivered_k_btu_hr
report_simulation_output.unmet_hours_cooling_hr
report_simulation_output.unmet_hours_heating_hr

4.1.3 Upgrade Costs

Upgrade cost multipliers include:

upgrade_costs.door_area_ft_2
upgrade_costs.duct_unconditioned_surface_area_ft_2
upgrade_costs.floor_area_attic_ft_2
upgrade_costs.floor_area_attic_insulation_increase_ft_2_delta_r_value
upgrade_costs.floor_area_conditioned_ft_2
upgrade_costs.floor_area_conditioned_infiltration_reduction_ft_2_delta_ach_50
upgrade_costs.floor_area_foundation_ft_2
upgrade_costs.floor_area_lighting_ft_2
upgrade_costs.flow_rate_mechanical_ventilation_cfm
upgrade_costs.rim_joist_area_above_grade_exterior_ft_2
upgrade_costs.roof_area_ft_2
upgrade_costs.size_cooling_system_primary_k_btu_h
upgrade_costs.size_heat_pump_backup_primary_k_btu_h
upgrade_costs.size_heating_system_primary_k_btu_h
upgrade_costs.size_heating_system_secondary_k_btu_h
upgrade_costs.size_water_heater_gal
upgrade_costs.slab_perimeter_exposed_conditioned_ft
upgrade_costs.wall_area_above_grade_conditioned_ft_2
upgrade_costs.wall_area_above_grade_exterior_ft_2
upgrade_costs.wall_area_below_grade_ft_2
upgrade_costs.window_area_ft_2

Other upgrade cost information includes:

upgrade_costs.option_<#>_name
upgrade_costs.option_<#>_cost_usd
upgrade_costs.option_<#>_lifetime_yrs
upgrade_costs.upgrade_cost_usd

where <#> represents any of the defined option numbers. See *Upgrade Scenario Configuration* for more information.

Note that the name, cost, and lifetime information will only be populated when applicable for a given upgrade option (i.e., when the apply logic evaluates as true).

4.2 Optional Outputs

Optional outputs include annual component loads, emissions, utility bills, and quantities of interest. Additionally, timeseries outputs can be requested.

To be generated, some optional outputs need only a single switch enabled (e.g., component loads). Others need additional input arguments specified (e.g., emissions, utility bills).

Note that enabling a set of optional outputs will by default generate the entire set. For example, requesting emissions outputs will result in columns for totals by fuel *and* by end use. There is currently no way to request totals by fuel while excluding totals by end use.

4.2.1 Component Loads

Component loads represent the estimated contribution of different building components to the annual heating/cooling building loads.

report_simulation_output.component_load_cooling_ceilings_m_btu
report_simulation_output.component_load_cooling_doors_m_btu
report_simulation_output.component_load_cooling_ducts_m_btu
report_simulation_output.component_load_cooling_floors_m_btu
report_simulation_output.component_load_cooling_foundation_walls_m_btu
report_simulation_output.component_load_cooling_infiltration_m_btu
report_simulation_output.component_load_cooling_internal_gains_m_btu
report_simulation_output.component_load_cooling_internal_mass_m_btu
report_simulation_output.component_load_cooling_lighting_m_btu
report_simulation_output.component_load_cooling_mechanical_ventilation_m_btu
report_simulation_output.component_load_cooling_natural_ventilation_m_btu
report_simulation_output.component_load_cooling_rim_joists_m_btu
report_simulation_output.component_load_cooling_roofs_m_btu
report_simulation_output.component_load_cooling_skylights_conduction_m_btu
report_simulation_output.component_load_cooling_skylights_solar_m_btu
report_simulation_output.component_load_cooling_slabs_m_btu
report_simulation_output.component_load_cooling_walls_m_btu
report_simulation_output.component_load_cooling_whole_house_fan_m_btu
report_simulation_output.component_load_cooling_windows_conduction_m_btu
report_simulation_output.component_load_cooling_windows_solar_m_btu
report_simulation_output.component_load_heating_ceilings_m_btu
report_simulation_output.component_load_heating_doors_m_btu
report_simulation_output.component_load_heating_ducts_m_btu

Continued on next page

Table 3 – continued from previous page

report_simulation_output.component_load_heating_floors_m_btu
report_simulation_output.component_load_heating_foundation_walls_m_btu
report_simulation_output.component_load_heating_infiltration_m_btu
report_simulation_output.component_load_heating_internal_gains_m_btu
report_simulation_output.component_load_heating_internal_mass_m_btu
report_simulation_output.component_load_heating_lighting_m_btu
report_simulation_output.component_load_heating_mechanical_ventilation_m_btu
report_simulation_output.component_load_heating_natural_ventilation_m_btu
report_simulation_output.component_load_heating_rim_joists_m_btu
report_simulation_output.component_load_heating_roofs_m_btu
report_simulation_output.component_load_heating_skylights_conduction_m_btu
report_simulation_output.component_load_heating_skylights_solar_m_btu
report_simulation_output.component_load_heating_slabs_m_btu
report_simulation_output.component_load_heating_walls_m_btu
report_simulation_output.component_load_heating_whole_house_fan_m_btu
report_simulation_output.component_load_heating_windows_conduction_m_btu
report_simulation_output.component_load_heating_windows_solar_m_btu

These outputs require that only a single switch is enabled. See the Residential HPXML Workflow Generator documentation page (i.e., `build_existing_model` section) for more information.

See the OpenStudio-HPXML Workflow Outputs section on Annual Component Building Loads for more information about component loads.

4.2.2 Emissions

Optional emissions outputs include annual total, by fuel, and by end use, for each emissions scenario requested. See the Residential HPXML Workflow Generator documentation page (i.e., `emissions` section) for more information on how to request emissions outputs by scenario.

See the OpenStudio-HPXML Workflow Outputs section on Annual Emissions for more information about annual emissions.

For the example below, the "LRMER_MidCase_15" emissions scenario was requested. See the list of available emissions scenario choices at https://github.com/NREL/resstock/tree/develop/resources/data/cambium.

report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_electricity_ceiling_fan_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_electricity_clothes_dryer_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_electricity_clothes_washer_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_electricity_cooling_fans_pumps_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_electricity_cooling_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_electricity_dehumidifier_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_electricity_dishwasher_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_electricity_electric_vehicle_charging_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_electricity_freezer_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_electricity_heating_fans_pumps_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_electricity_heating_heat_pump_backup_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_electricity_heating_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_electricity_hot_tub_heater_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_electricity_hot_tub_pump_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_electricity_hot_water_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_electricity_hot_water_recirc_pump_lb

Continued on next page

Table 4 – continued from previous page

report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_electricity_hot_water_solar_thermal_pump_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_electricity_lighting_exterior_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_electricity_lighting_garage_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_electricity_lighting_interior_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_electricity_mech_vent_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_electricity_plug_loads_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_electricity_pool_heater_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_electricity_pool_pump_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_electricity_pv_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_electricity_range_oven_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_electricity_refrigerator_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_electricity_total_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_electricity_well_pump_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_fuel_oil_heating_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_fuel_oil_hot_water_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_fuel_oil_total_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_natural_gas_clothes_dryer_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_natural_gas_fireplace_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_natural_gas_grill_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_natural_gas_heating_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_natural_gas_hot_tub_heater_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_natural_gas_hot_water_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_natural_gas_lighting_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_natural_gas_pool_heater_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_natural_gas_range_oven_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_natural_gas_total_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_propane_clothes_dryer_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_propane_heating_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_propane_hot_water_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_propane_range_oven_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_propane_total_lb
report_simulation_output.emissions_co_2_e_lrmer_mid_case_15_total_lb

4.2.3 Utility Bills

Optional utility bill outputs include annual fixed and marginal costs (total, by fuel type), and PV credits for electricity, for each utility bill scenario requested. See the Residential HPXML Workflow Generator documentation page (i.e., `utility_bills` section) for more information on how to request utility bill outputs by scenario.

See the OpenStudio-HPXML Workflow Outputs section on Utility Bill Outputs for more information about utility bill outputs.

report_utility_bills.bills_electricity_energy_usd
report_utility_bills.bills_electricity_fixed_usd
report_utility_bills.bills_electricity_pv_credit_usd
report_utility_bills.bills_electricity_total_usd
report_utility_bills.bills_fuel_oil_energy_usd
report_utility_bills.bills_fuel_oil_total_usd
report_utility_bills.bills_natural_gas_energy_usd
report_utility_bills.bills_natural_gas_fixed_usd
report_utility_bills.bills_natural_gas_total_usd
report_utility_bills.bills_propane_energy_usd
report_utility_bills.bills_propane_total_usd
report_utility_bills.bills_total_usd

4.2.4 Quantities of Interest

These outputs require that only a single switch is enabled. See the Residential HPXML Workflow Generator documentation page (i.e., `reporting_measures` section) for more information on how to request quantities of interest outputs.

qoi_report.qoi_average_maximum_daily_timing_cooling_hour
qoi_report.qoi_average_maximum_daily_timing_heating_hour
qoi_report.qoi_average_maximum_daily_timing_overlap_hour
qoi_report.qoi_average_maximum_daily_use_cooling_kw
qoi_report.qoi_average_maximum_daily_use_heating_kw
qoi_report.qoi_average_maximum_daily_use_overlap_kw
qoi_report.qoi_average_minimum_daily_use_cooling_kw
qoi_report.qoi_average_minimum_daily_use_heating_kw
qoi_report.qoi_average_minimum_daily_use_overlap_kw
qoi_report.qoi_average_of_top_ten_highest_peaks_timing_cooling_hour
qoi_report.qoi_average_of_top_ten_highest_peaks_timing_heating_hour
qoi_report.qoi_average_of_top_ten_highest_peaks_use_cooling_kw
qoi_report.qoi_average_of_top_ten_highest_peaks_use_heating_kw
qoi_report.qoi_peak_magnitude_timing_hour
qoi_report.qoi_peak_magnitude_use_kw

4.2.5 Timeseries

Specifying any timeseries frequency other than "none" results in, by default, end use consumptions and total loads timeseries output requests. See the Residential HPXML Workflow Generator documentation page (i.e., `simulation_output_report` section) for more information on how to request various timeseries outputs (or override default requests).

See the OpenStudio-HPXML Workflow Outputs section on Timeseries Outputs for more information on timeseries outputs.