OpenStudio Release Notes - 3.8.0

National Renewable Energy Laboratory

OpenStudio Version 3.8.0

Release Notes - 2024-05-08

These release notes describe version 3.8.0 of the OpenStudio SDK developed by the National Renewable Energy Laboratory (NREL), Buildings and Thermal Sciences Center, Commercial Buildings Research Group, Tools Development Section, and associated collaborators. The notes are organized into the following sections:

- Overview
- Where to Find OpenStudio Documentation
- Installation Notes
- OpenStudio SDK: Changelog

Overview

As of April 2020, development and distribution of the OpenStudioApplication and the SketchUp plugin have transitioned to the OpenStudio Coalition, who is independently managing and distributing the software through its own openstudiocoalition/OpenStudioApplication repository. The OpenStudio SDK is continuing to be actively developed and distributed by NREL and is released two times per year, through a spring and a fall release.

Below is the list of components that is included in this SDK installer:

OpenStudio SDK 3.8.0 - EnergyPlus - Command Line Interface (CLI) - Radiance - Ruby API - Python API - C++ SDK

Note that PAT is not included in either the SDK or the OpenStudio Coalition's Application installers. You will need to install PAT separately which is distributed on the OpenStudio-PAT GitHub page.

Where to Find OpenStudio SDK Documentation

- OpenStudio SDK release documentation, including these release notes, tutorials, and other user documentation, is available at https://www.openstudio.net/
- C++ API documentation is available at OpenStudio SDK Documentation
- Measure development documentation is available at OpenStudio Measure Writer's Reference Guide
- A roadmap for planned features is available at Roadmap

Installation Notes

OpenStudio SDK 3.8.0 is supported on:

- 64-bit Windows 7 – 11
- macOS: 11.6+ x86_64, 12.1+ arm64
- Ubuntu: 20.04 x86_64, 22.04 x86_64, 22.04 arm64
- Centos7

OpenStudio SDK 3.8.0 supports EnergyPlus Release 24.1.0, which is bundled with the OpenStudio installer. It is no longer necessary to download and install EnergyPlus separately. Other builds of EnergyPlus are not supported by OpenStudio SDK 3.8.0.

OpenStudio SDK 3.8.0 supports Radiance 5.0.a.12, which is bundled with the OpenStudio installer; users no longer must install Radiance separately, and OpenStudio will use the included Radiance version regardless of any other versions that may be installed on the system. Other builds of Radiance are not supported by OpenStudio SDK 3.8.0.

As usual, you can refer to the **OpenStudio SDK Compatibility Matrix** for more information.

Installation Steps

- Download and install OpenStudio SDK and/or any third party tool that embeds the OpenStudio SDK into their software such as openstudiocoalition/OpenStudioApplication depending on your needs. Select components for installation. Note that OpenStudio Application is a standalone app and does not require you to install OpenStudio SDK.
- A BCL AUth Key is no longer needed to access content on Building Component Library (BCL) BCL is typically accessed through third party OpenStudio applications to provide modelers access to building components and OpenStudio measures; however you an also browse it and download content using the BCL link above.

For help with common installation problems please visit Getting Started.

OpenStudio SDK:

The 3.8.0 is a **major** release. This update includes several new features, performance improvements, and bug fixes.

C++ Workflow code

As of OpenStudio SDK 3.7.0 a re-written workflow written in C++ is used by default in place of the Ruby based Workflow Gem that had been used in the past. This enhancement is in support of Python measures being used in an OpenStudio workflow, including mixed language workflows that include both Ruby Measures and Python Measures. If you need to use the older Workflow Gem implementation, you can do that by using the `classic` subcommand after `openstudio`. `classic` will be deprecated in a future version of OpenStudio.

Python Bindings

As of OpenStudio SDK 3.2.0, Python bindings are officially supported and distributed through Python Package Index (PyPI). To install, users will need to have Python3 installed along with pip and simply run the following command in a terminal window.

```
pip install openstudio==3.8.0
```

Please see openstudio on PyPi for further instructions on how to install. Users can also visit the test channel at openstudio on TestPyPi to install development bindings.

You can also refer to the OpenStudio SDK Python Binding Version Compatibility Matrix to see the list of supported platforms and python versions.

New Features, Major Fixes and API-breaking changes

For a list of deprecated and removed methods, please refer to deprecated_methods.csv

- #5094 - Update to EnergyPlus 24.1.0

- #5137 - Update from Ruby 2.7.2 to Ruby 3.2.2. This required a significant effort in other dependencies including Conan, and significant work in the continuous integration (CI) system.

- Fix logging related issues. Openstudio version 3.7 introduced a number of new issues related to logging and log files. These issues have been addressed.

  - #5030
  - #4847
  - #5037
  - #4918

- Refactoring and relocating model methods that has been in the openstudio-extensio-gem Gem to be in the openstudio-standards Gem or in OpenStudio C++ SDK. A few methods will be moved directly into measures. The result of this is that with OpenStudio 3.8.0 no measures should have to require 'openstudio-extension' within a measure's Ruby code.

  - #5142 - Add native support for measure argument methods in OsLib_HelperMethods

- Enhancements to the Building Component Library (BCL)

  - #5126 - Add versionModified, repo, org, releaseTag to BCLSearchResult
  - Support for OpenStudio Python Measures
  - Updating Measure Details page to include measure language
  - Link to GitHub Repo from Measure Details page
  - Access to prior versions of a measure from the Measure Details page

- #5099 - HeatExchangerAirToAirSensibleAndLatent changes for effectiveness

  - To conform to E+ 24.1.0, where "Sensible/Latent Effectiveness at 75% Heating/Cooling" numeric fields have been replaced with a curve, the numeric getters / setters are now deprecated in favor of the new curves getters / setters: `sensibleEffectivenessofHeatingAirFlowCurve`, `latentEffectivenessofHeatingAirFlowCurve`, `sensibleEffectivenessofCoolingAirFlowCurve`, `latentEffectivenessofCoolingAirFlowCurve`
  - The constructor will no longer assign an effectiveness at 75% and assume a constant effectiveness. A helper method `bool assignHistoricalEffectivenessCurves()` is provided to create and assign 4 `TableLookups` that will match the pre E+ 24.1.0 defaults for Sensible/Latent Effectiveness at 75% Heating/Cooling airflow

- #5105 - ElectricEquipmentITEAirCooled / ElectricEquipmentITEAirCooledDefinition - align objects with other SpaceLoads

  - A number of methods have been renamed (and the old ones deprecated) to conform to the API for other `SpaceLoadInstance` / `SpaceLoadDefinition` objects
  - Mostly `getWattsperUnit` is changed to `getDesignLevel` and `getWattsperZoneFloorArea` is changed to `getPowerPerFloorArea`
  - Refer to deprecated_methods.csv for the complete list

- #5111 - ScheduleDay: new timeseries method and interpolation options

  - The `ScheduleDay` has API-breaking changes related to setters and getters for its `Interpolate To Timestep` field. They now use `string` rather than `bool` to conform to the IDD type `\choice`
  - The forward translator for `ScheduleDay` replaces always setting "Average" with the interpolation method stored in `Interpolate to Timestep` field
  - A new `timeseries()` method is introduced for returning vectors of times and (interpolated) values for the given number of timesteps per hour established by the `Timestep` object; `getValue` references this vector for an interpolated value at a given time

Minor changes and bug fixes

Refer to the changelog on the release page at v3.8.0

Full Changelog: https://github.com/NREL/OpenStudio/compare/v3.7.0. . . v3.8.0

New Contributors:

OpenStudio Standards v0.6.1

Update the openstudio-standards gem to version 0.6.1 In addition to some refactoring, this release also included conversion of 90.1 data to formal database.