2020 Building Performance Analysis Conference and
SimBuild co-organized by ASHRAE and IBPSA-USA

A MODELING FRAMEWORK FOR ENGINE-NEUTRAL AUTOMATION OF BUILDING ANALYSIS AND COMPLIANCE REPORTING

Eric Niemeyer[1] and Sagar Rao[1]
[1]Affiliated Engineers, Inc., Madison, WI, USA

ABSTRACT

Building energy modeling (BEM) can be a powerful tool in the design of high performance buildings (HPBs). However, as code requirements become more stringent and energy efficiency targets more ambitious, modelers are struggling to keep up with tightening project timelines and increasingly complex design features. New technologies can require significant time for research, prototyping, and development of modeling solutions. Additionally, code compliance documentation can be time-consuming but provides limited value to the design process. The time spent on these tasks can end up being a significant component of project soft costs, thus diminishing the business case for BEM. Previous attempts have been made to simplify the process of analyzing groups of Energy Saving Measures (ESMs), but they are limited in customizability and programmability for complex measures.

This paper proposes an automation workflow capable of optimizing large scale parametrics to inform building design and generating code and beyond-code compliance documentation. The authors developed a Python module to share and automate numerous repeatable BEM tasks using predefined data models for simulation data exchange using the EnergyPlus engine. The combined use of a programming interface with EnergyPlus allows for the repeatable deployment of complex design measures and custom control sequences that have historically been significant pain-points for BEM practitioners. The Python module can also generate reporting documentation for code and beyond code compliance. A case study demonstrates how using this automation workflow allows design teams to maximize value to their clients and improve the overall value proposition for BEM.

INTRODUCTION

Buildings account for nearly 40% of energy related CO_2 emissions worldwide (Global Alliance for Buildings and Construction 2018), and they will play a key role in meeting the emissions reduction goals of the 21st century. Building designers, engineers, and analysts will play a key role in achieving these energy reduction goals, yet, they face many of the same fundamental problems they did at the start of the last decade. Tasks like input data entry, seed model creation, parametrics modeling, and results processing make up a significant share of a project's billable hours but are a vital component of a workflow to delivering improved building performance. Available analysis tools have struggled to achieve the conflicting need for accuracy, flexibility, and speed. These issues are compounded for large buildings, where BEM stands to make the greatest overall impact.

In 2011, a post-report to the Rocky Mountain Institute Building Energy Modeling Innovation Summit concluded that two of the biggest challenges with the energy modeling process are—i) time constraints and ii) modelers constantly having to redevelop methods for transforming data inputs and model systems with workarounds (Rocky Mountain Institute 2011). Because these workarounds are typically developed for a specific project, they frequently require additional re-work to be used for other projects. Thus, the process repeats itself, and each new project requires the same manual steps, leaving limited time for critical thinking and research of new technologies and practices. In this framework, modelers are not only working in their own silos but building new ones for each project.

DESIGN, ANALYSIS & COMPLIANCE

The Design Process

While BEM can be utilized at any point in a building project, it is generally the case that earlier and less settled design phases offer greater opportunities for improving energy performance through design-assist modeling (Hemsath 2013). In particular, the pre-design and schematic design (SD) phases generally offer the only realistic window for BEM to meaningfully inform some of the most impactful overarching decisions for the architectural design. These may include orientation, building form, and window-to-wall ratios on each face, which can be highly dependent on program layout. In the absence of detailed energy modeling in these early stages, opportunities for energy savings with marginal

© 2020 ASHRAE (www.ashrae.org) and IBPSA-USA (www.ibpsa.us).
For personal use only. Additional reproduction, distribution, or transmission in either print or digital form is not permitted without ASHRAE or IBPSA-USA's prior written permission.

increases in design cost can be missed. Instead these design decisions can be made arbitrarily.

In attempts to bridge this gap, a number of software-based solutions are being proposed to help inform architects with early phase modeling by simplifying inputs for project data like HVAC and internal loads. The shortcomings of such simplified early stage modeling platforms are twofold: i) they fail to accurately represent the complex interactions between building HVAC, architectural design, and building program that are necessary to inform an integrated whole-building design approach, and ii) they necessitate a design break when the project's level of detail surpasses that of the software. Tools designed exclusively for simplified early stage modeling cannot be adapted to handle the increasd complexity required for, say, code compliance models. Maintaining modeling continuity throughout the entirety of a project is vital for consistent reporting of results, as well as avoiding labor-intensive data re-entry each time a new software is utilized.

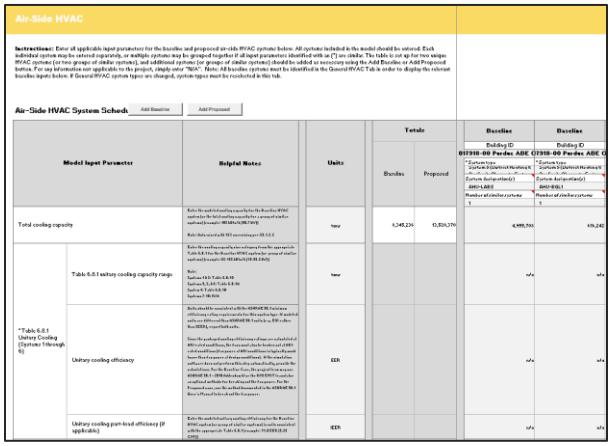

Figure 1 LEED Minimum Energy Performance Inputs

Compliance Reporting

While BEM can have a greater impact with early-phase design assistance, it is often instead used for post-design validation, or to confirm a design complies with code or beyond-code requirements. A design team may decide early in a project to achieve energy code compliance through either a prescriptive or performance-based path. Prescriptive code compliance documentation can be completed with relatively low effort, while performance-based paths require an energy model with more extensive documentation. This encourages design teams to forego an energy model if budget is the primary concern and all prescriptive requirements are being met. A growing number of high-performance and green building certifications has arisen in recent years, but each carries a different cost add to the design. For some, a non-trivial part of that cost lies in the documentation efforts for energy performance metrics.

Analysis Tools & Delivery Methods

Extensive BEM tools exist for practitioners to choose from. Typically, an energy simulation tool and engine are determined early in a project, based on a combination of modeler experience and the unique modeling needs of a project. However, certain tools may be better for handling a particular HVAC system, while others may be incapable of modeling them without workarounds. This discourages thorough testing of HVAC system options if something falls out of the boundaries of the selected tool. Doing so would mean another break in model continuity and data re-entry efforts.

SCRIPTING, AUTOMATION, & ENGINE-NEUTRALITY

Many solutions have been proposed in the last decade to reduce the labor-intense efforts modelers are facing. EnergyPlus was developed largely out of a need for increased flexibility and modularity for modeling new technologies (U.S. DOE 2019), and the OpenStudio software development kit (SDK) provides an open source platform for users to programmatically manipulate energy models and develop additional functionalities (Guglielmetti, Macumber, and Long 2011; Weaver et al. 2012). This has made EnergyPlus a natural choice for modelers who require higher degrees of flexibility and programmability. Consequently, most of these efforts have been focused on assisting EnergyPlus users.

The Building Component Library was introduced to make it easier to manage data inputs for consistent and streamlined model articulation (Fleming, Long, and Swindler 2012). Shortly thereafter, the Parametric Analysis Tool was developed to help bring down the entry barrier for running computationally intense large scale parametrics (Macumber, Ball, and Long 2014). jEPlus also provides another pathway for simulating complex parametrics in parallel (Zhang and Korolija 2010).

A number of more extensive scripting frameworks have been developed to help users automate additional steps of their modeling workflows (Roth et al. 2018). OpenStudio Measures allows manipulation of the OpenStudio objects for translation to EnergyPlus input files (Roth, Goldwasser, and Parker 2016). Eppy gives users the ability to parse EnergyPlus input and output files and to manipulate inputs for through Python (Philip 2019). Modelkit is an open-source framework for parametric modeling that includes a variety of templates and toolkits used for model generation (Ellis 2015). Methods have also been developed to automate 90.1

© 2020 ASHRAE (www.ashrae.org) and IBPSA-USA (www.ibpsa.us).
For personal use only. Additional reproduction, distribution, or transmission in either print or digital form is not permitted without ASHRAE or IBPSA-USA's prior written permission.

baseline models from a proposed design using the OpenStudio SDK and Measures (Goldwasser 2018; Parker et al. 2017).

```
from eppy import modeleditor
from eppy.modeleditor import IDF

iddfile = r"C:\EnergyPlusV8-8-0\Energy+.idd"
try:
    IDF.setiddname(iddfile)
except modeleditor.IDDAlreadySetError as e:
    pass

fname1 = r"C:\Users\ENIEMEYER\Desktop\5-Zone Model.idf"
idf1 = IDF(fname1)
surfaces = idf1.idfobjects['BUILDINGSURFACE:DETAILED']

s_names = [surface.Name for surface in surfaces]
print s_names[:5]

vertical_walls = [sf for sf in surfaces if sf.tilt == 90.0]
print [sf.Name for sf in vertical_walls]
```

Figure 2 Example Script Using Eppy

While these tools vary in approach and intended scope, they are all designed specifically to manipulate files for and around EnergyPlus or OpenStudio. As such, each derives its data dictionary from EnergyPlus or OpenStudio (and hence, indirectly EnergyPlus). Automating certain tasks can be effective within these frameworks, but establishing workflows entirely within these boundaries presents inherent structural limitations to what can be included in the scope of an automated workflow.

Current Deficiencies

Each version of EnergyPlus includes a unique Input Data Dictionary (IDD) that defines the EnergyPlus data model for that version. It is possible to make minor edits to the IDD to, say, increase the allowed capacity for certain extensible fields. However, making more significant edits to other input fields or adding new objects definitions is not possible, as they will not be recognized by the simulation engine. Several data constructs available in the OpenStudio Model file (OSM) is, such as space types and spaces as distinct from thermal zones, do not exist in the IDD. For the most part, however, the structure of the OSM is a clone of the EnergyPlus data model, so it is subject to the same shortcomings.

Therefore, scripting within the OpenStudio SDK schema will limit the lifespan of use for those efforts to the life of that version of OpenStudio, or at most, to OpenStudio itself. Changes in the structure of future versions may require significant re-work of any such measures. Furthermore, the OpenStudio graphical user interface has been deprecated, so further investment in that platform limits availability of model interactions through the interface.

Additionally, there is no way to add custom definitions in the OSM. This presents serious limitations for automating tasks and analyses that require considerations outside the scope of the OSM. If a user wishes to assign additional data tags to an object, such as the Integrated Part Load Value (IPLV) of a chiller, there is no way to bake that information into the model data itself and, consequently, no way to read that information from simulation results. If a user has specifications for a renewable energy system that cannot be modeled sufficiently in EnergyPlus, there would be a need for post-processing of results to accurately account for it. This disconnect from model inputs to model outputs makes the OSM insufficient in fully defining a project's data model.

This speaks to several key issues at the heart of the OSM that limit its ability to write fully automated workflows around it: i) it is not extensive enough to handle the ever-growing range of object types necessary for accurate and wholistic building data representation, and ii) it is not an extensible file format, in that users cannot introduce their own custom object definitions with unique rulesets to meet project-specific analysis needs. Therefore, any proposed solution which limits itself to the pre-defined scope of OpenStudio and the OSM will fall short of providing full start-to-end workflow automation, instead requiring additional workarounds to handle incompatible data flows.

Achieving workflow automation that is sufficiently scalable, modular, and flexible to confront the full range of challenges a modeler might face requires a standardized project data model that is both extensive and extensible. Extensibility is central to giving modelers the ability to freely define new attributes and objects in their data model. It allows users to write rules about how data are transformed, processed, and presented. Translations to the input language of a simulation engine are then independent from the data model, allowing for additional data definitions to be accounted for outside the simulation engine. This allows for the data model to be engine-neutral, meaning the project data definition can be used to run simulations with any tool for which data translations are written.

A NEW MODELING FRAMEWORK

A modeling framework was developed to demonstrate and assess the use of a full start-to-end BEM workflow automation. It needed to be capable of carrying a fully defined project data model from project Pre-SDs and early phase parametrics through construction phase modeling and documentation for code compliance and beyond-code building certification. The framework, is comprised of three distinct workflow components—i) a user interface (UI) to simplify and standardize user inputs and front-end customizability, ii) a data model with sufficient definition to run detailed energy

© 2020 ASHRAE (www.ashrae.org) and IBPSA-USA (www.ibpsa.us).
For personal use only. Additional reproduction, distribution, or transmission in either print or digital form is not permitted without ASHRAE or IBPSA-USA's prior written permission.

simulations, and iii) a Python library to facilitate model manipulation and results processing. Compiling the optimal tool set for the workflow involved a modular selection that allows increased flexibility of data flows for future versions.

Space Type Name	Total Area [sf]	90.1 LPD [W/sf]	Design LPD [W/sf]	EPD Sensible [W/sf]	EPD Latent [W/sf]	Occupant Density [sf/person]	Modeled Heat Gain [W/person]
Private Office	15,736	0.93	0.84	1.5	0.0	120	126
Open Office	30,868	0.81	0.73	1.5	0.0	80	126
Restrooms	8,734	0.85	0.77	0.2	0.0	500	180
Conference Room	10,201	1.07	0.96	2.0	0.0	40	117
Collaboration Space	19,453	1.07	0.96	1.5	0.0	25	126
Auditorium	4,338	0.63	0.57	0.2	0.0	20	108
Kitchen - servery, prep	2,606	1.06	0.95	10.0	2.0	350	189
Lobby, Prefunction	18,801	1.00	0.90	0.2	0.0	225	126
Corridor	27,775	0.66	0.59	0.1	0.0	1,000	180
Classroom	14,285	0.92	0.83	1.0	0.0	40	126
Stairway	6,154	0.58	0.52	0.1	0.0	1,000	270
Electrical and Mechanical	24,728	0.43	0.39	2.0	0.1	5,000	207
Teaching Lab - Dry	2,842	1.20	1.08	6.0	0.5	200	144
Maker Space	8,549	1.14	1.03	4.0	0.5	400	180
Storage	5,538	0.46	0.41	0.1	0.0	1,000	207
Lockers	4,929	0.48	0.43	0.2	0.0	80	207
Dining Areas - Café	9,326	0.63	0.57	0.5	0.2	120	144
Unconditioned-general	24,831	0.46	0.41	0.0	0.0	5,000	72
Reference Library	1,875	0.82	0.74	0.2	0.0	250	99

Figure 3 UI Inputs Example

User Interface

To facilitate the front-end user experience for quickly developing data models with input entry, a number of options were available. While a Python- or web-based UI would offer a high degree of complexity and customizability, an Excel-based spreadsheet UI was selected for prototyping. Excel allows for rapid structural and format changes to the UI, while allowing more complex user inputs like HVAC system assignments and database lookups to be partially automated with VBA scripting. An in-house spreadsheet tool previously used for the direct generation of scalable and complex EnergyPlus input files was adapted for this use (Rao et al. 2018). The back-end components used for file generation were removed to leave only the user-facing inputs for the data model.

The UI is designed to pull in EnergyPlus geometry files generated using the OpenStudio plugin for Sketchup. Within Sketchup, building space types are assigned, and Ruby scripts are used to set interior ceiling heights and to name spaces, thermal zones, and surfaces by standardized naming conventions. As the UI imports a geometry file, it reads for specific objects and calculates zone areas to be available for editing and model articulation directly from the UI. Consolidating all relevant inputs into the Excel UI introduces a variety of possibilities for early quality control checks to model inputs, and it allows for new tables and input definitions to be added freely and easily. Removing the other functional programmatic elements of the data model from the spreadsheet UI allows the user to focus only on the relevant model inputs that affect results to more easily track changes made throughout the different phases of a project. This allows for start-to-end continuity for the data model as it evolves alongside the project design. The UI includes a comprehensive, automated data mapping of all model inputs, so that the data model can automatically read the inputs while maintaining flexibility of data formats and locations.

The UI is capable of running batches of parametric runs based on different ESM bundles specified by a modeler. It can also import simulation results for single runs or parametric bundles so as to keep track of changes made to model inputs throughout the project. The UI template also serves to record modeling conventions and best practices with the use of default values and input restrictions, which contributes to improved consistency of results. In the long-term, the Excel UI could potentially be replaced with a web client that replicates its functionalities and provides the users a seamless evolution from desktop apps to web apps for managing their data models.

Data Model

As previously discussed, input files to both EnergyPlus and OpenStudio are modular and flexible options for describing the data model for energy simulations. However, they fall short in that they are not sufficiently extensive or extensible. The framework utilizes bsim, a standardized BEM data definition language, for its data model. This enables the framework to be completely engine-neutral, though it is currently designed to utilize EnergyPlus as the calculation method.

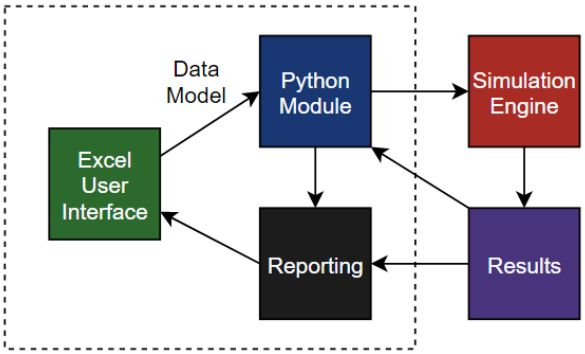

Figure 4 New Modeling Workflow Diagram

The data model is made up entirely of inputs automatically gathered using the dynamic data mapping structure set up within the UI. The data conform to the standardized data definitions in bsim, which can also convert the exported data into an EnergyPlus Input File (IDF) on demand for EnergyPlus simulations. As new inputs and tables are introduced to the UI, bsim is

© 2020 ASHRAE (www.ashrae.org) and IBPSA-USA (www.ibpsa.us).
For personal use only. Additional reproduction, distribution, or transmission in either print or digital form is not permitted without ASHRAE or IBPSA-USA's prior written permission.

capable of incorporating new definitions to account for them as needed, and vice versa.

Programming Interface

A comprehensive scripting library was developed to easily and quickly manipulate the data model to run parametrics, test modeling scenarios for design-assist, and process results for different reporting needs and quality control. A variety of scripting languages would provide suitable methods to create such a library, including Ruby, Julia, and Python. Python was a suitable choice for this purpose, given its prevalence in the field and the availability of libraries focused on building simulations (Miller, Hersberger, and Jones 2013).

```
#modify model to create mutations
ep = EPModel(seed_file)
log = ''

szs = ep.get_all_objects('Sizing:Zone')
for sz in szs:
    for space_type in bsim['SpaceTypes']:
        if bsim['SpaceTypes'][space_type]['Space Tag'] == sz.Zone_or_ZoneList_Name[5:-4]:
            if bsim['SpaceTypes'][space_type]['Supply Airflow Occupied'] != '':
                sz.Cooling_Minimum_Air_Flow = convert((convert(ep.get_zone_area(sz.Zone_or_Zor

atus = ep.get_all_objects('AirTerminal:SingleDuct:VAV:Reheat')
for atu in atus:
    for space_type in bsim['SpaceTypes']:
        if bsim['SpaceTypes'][space_type]['Space Tag'] == atu.Name[5:-8]:
            if bsim['SpaceTypes'][space_type]['Supply Airflow Occupied'] != '':
                atu.Zone_Minimum_Air_Flow_Input_Method = 'FixedFlowRate'
                atu.Fixed_Minimum_Air_Flow_Rate = convert((convert(ep.get_zone_area(atu.Name[:
log = log + '_fixedAirflows'

fans = ep.get_all_objects('Fan:VariableVolume')
fans_delete = []
for fan in fans:
    if 'FCU' in fan.Name:
        cv = ep.new_object('Fan:ConstantVolume')
        cv.Name = fan.Name
        cv.Availability_Schedule_Name = fan.Availability_Schedule_Name
        cv.Fan_Total_Efficiency  = fan.Fan_Total_Efficiency
        cv.Pressure_Rise = fan.Pressure_Rise
        cv.Motor_Efficiency = fan.Motor_Efficiency
        cv.Motor_In_Airstream_Fraction = fan.Motor_In_Airstream_Fraction
        cv.Air_Inlet_Node_Name = fan.Air_Inlet_Node_Name
        cv.Air_Outlet_Node_Name = fan.Air_Outlet_Node_Name
        fans_delete.append(fan.Name)
```

Figure 5 aeiSIMPLE Script Example

aeiSIMPLE is a library developed in Python to automate tasks ranging from model manipulations to results processing and reporting. It utilizes buildSIMPLE, a Python-based SDK that for bsim. Certain pre-defined measures in buildSIMPLE are employed for use in the library. aeiSIMPLE includes packages of measures written using buildSIMPLE that allow for quick and easy manipulation of the bsim data model. It also includes central repositories of energy modeling data that can easily be retrieved for simulations. Individual modules are written for different ESMs for streamlined deployment in parametric simulations. These are written to be independent of project-specific conditions, so that they can be utilized freely without manual manipulation of model files. Measures can be written in customizable combinations depending on project needs.

aeiSIMPLE also includes modules written to automatically generate code compliance documentation. Extensive documentation requirements for building certification processes, like the LEED Minimum Energy Performance Calculator, are also available. These functions can be called to quickly produce all required documentation from an instance of the data model as described in the UI. Because aeiSIMPLE can work with extensible data definitions, it can process information outside of the simulation engine used. In the case of renewable energy systems, unique inputs can be defined in the UI, so that the Python module can read them and pre-process inputs or post-process results accordingly before they are displayed in the UI. In this way, the workflow can fully incorporate calculation methods in cases where OpenStudio would require an exceptional calculation.

CASE STUDIES

Two projects were selected for case studies utilizing the new workflow: i) An academic research laboratory building in Detroit, Michigan with zero net energy (ZNE) goals, and ii) A mental healthcare facility in Toronto, Ontario with an industry leading energy use intensity (EUI) target. The former was utilized for a large scale parametrics study, while the latter was used for automated code and beyond-code compliance documentation. Each was intended to demonstrate unique features and strengths of the framework.

Academic Research Laboratory Parametric Study

The University of Michigan Detroit Center for Innovation is an academic research laboratory building planned for its developing downtown Detroit campus. The project is been considering ambitious ESMs to meet its ZNE target, including heat recovery chillers, dedicated outdoor air systems serving active chilled beams, and a geo-exchange field in various combinations with more conventional systems like VAV with hot water reheat.

This project was suitable for generating large scale parametric runs using the new workflow. ESMs to be considered for parametric analyses can be selected in the UI. Available options include typical envelope parameters like wall and roof R-values, window insulation and shading coefficients, and window-to-wall ratios. Detailed envelope measures like electrochromic glazing can also be considered. Complex HVAC measures like a geo-exchange field and active chilled beams in certain space types were also applied in parallel to envelope measures to study the interactive effects of different system configurations and how many combinations could realistically lead to a ZNE building.

Parametric runs can also be conducted in series or parallel, by an order selected by the user. Results of the incremental measure analysis were used to inform simple payback analysis of several key ESMs being

© 2020 ASHRAE (www.ashrae.org) and IBPSA-USA (www.ibpsa.us).
For personal use only. Additional reproduction, distribution, or transmission in either print or digital form is not permitted without ASHRAE or IBPSA-USA's prior written permission.

considered. For smaller bundles of parametrics, runs were simulated on a local unit for instant feedback, but for larger permutations of ESM combinations, simulations were pushed to a separate server with accelerated hardware to minimize overall runtimes.

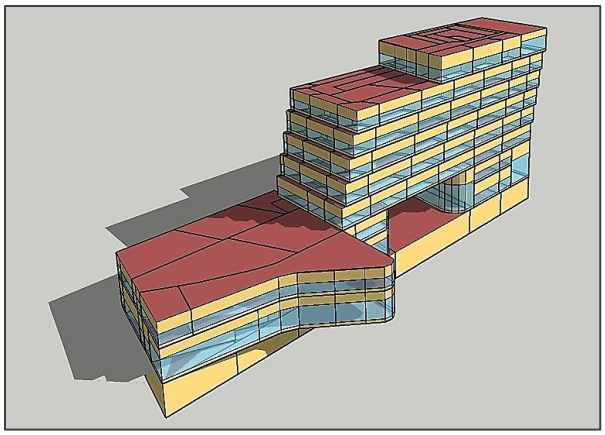

Figure 6 UM Detailed Building Geometry

This case study demonstrated how the new workflow was capable of producing complex parametric runs from a very simple set of inputs by the user. Separating all scripting efforts from the inputs allows the UI to be distilled down to the essential data inputs for parametric runs. Selecting "R-15", "R-20", and "R-25" for wall constructions or "6-pipe heat recovery chiller with geo-exchange loop" for a plant side HVAC selection are the only requirements for testing those ESMs in all applicable bundles.

aeiSIMPLE includes modules for complex HVAC system configurations that are capable of conforming to any project geometry, so that simulations conducted are based on detailed energy models, even at project phases when simplified models might ordinarily be employed. It also packages results in easily viewable formats when the runs are finished. Runtimes are still currently subject to the simulation speed of EnergyPlus, but total setup time is significantly reduced.

Healthcare Facility Compliance Documentation

The Centre for Addiction and Mental Health (CAMH) Research Building is a 7-story 330,000 square foot research laboratory building located near downtown Toronto, Ontario. The project is intended to be an internationally significant brain research facility. The building utilizes a 6-pipe heat recovery chiller with a geo-exchange field to provide heating and cooling, a double-skin façade, underfloor air distribution, and active chilled beams, among other ESMs.

The stated project goal was to reach an ambitious building EUI of 110 kBtu/ft^2-yr, inclusive of heavy equipment loads and air change requirements in many of the laboratory and vivarium spaces. In addition to the internal energy target, the project needed to demonstrate compliance with ASHRAE 90.1-2013 to satisfy 2025 Tier 3 Toronto Green Standards. The project was also planning for LEED certification under LEED v4, in part by achieving a maximum amount of points under the Optimize Energy Performance Credit. This project was a suitable candidate to demonstrate code and beyond code compliance documentation with the new workflow, due to its multi-faceted energy goals and the extensive reporting requirements resulting for the project.

Energy models for CAMH were developed using the UI, which is used to define not only data directly used for energy simulations but additional definitions used specifically for code compliance, such as exterior lighting categories and service hot water fixture information. The bsim data model exports all available data sets, which allow the Python library to auto-generate completed compliance documentation for upload. In this case the deliverables were data outputs for an energy report demonstrating the EUI performance of the building and LEED Minimum Energy Performance documentation. For this case study, graphics and appendix tables in the report were also generated directly from the data model using aeiSIMPLE.

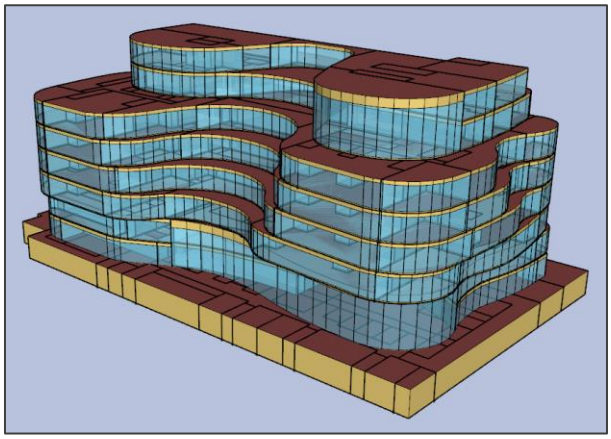

Figure 7 CAMH Detailed Building Geometry

To achieve LEED points under the Minimum Energy Performance Credit, specific model output tables must be uploaded to LEED online, in addition the LEED v4 Minimum Energy Performance Calculator, which must be filled out with appropriate model inputs and outputs for both the ASHRAE 90.1-2010 Appendix G baseline and proposed models. For a modeler new to LEED working on large or complex building, the

© 2020 ASHRAE (www.ashrae.org) and IBPSA-USA (www.ibpsa.us).
For personal use only. Additional reproduction, distribution, or transmission in either print or digital form is not permitted without ASHRAE or IBPSA-USA's prior written permission.

documentation process can be particularly time intensive. aeiSIMPLE can generate the required output tables from baseline and proposed models, as well as a unique spreadsheet designed to replicate the Minimum Energy Performance Calculator for LEED v4, shown in Figure 8. While the data must still be copied into the macro-enabled calculator with built-in checks that LEED provides, this method can reduce entry time from hours or even days to minutes.

Air-Side HVAC

Air-Side HVAC System Schedule

Model Input Parameters		Baseline Building ID 017231-00 IUH NCC CD System designation(s) AHU-L1 Number of similar systems 1	Baseline Building ID 017231-00 IUH NCC CD System designation(s) AHU-L2 Number of similar systems 1
Total cooling capacity		1,220,975	1,277,991
* Table 6.8.1Unitary Cooling (Systems 1through 6)	Table 6.8.1unitary cooling capacity range	≥ 760,000	≥ 760,000
	Unitary cooling efficiency	n/a	n/a
	Unitary cooling part-load efficiency (if applicable)	n/a	n/a
Total heating capacity		704,591	710,969
* Table 6.8.1Unitary Heating (Systems 2, 3, 4, and 9)	Table 6.8.1unitary heating capacity range	≥ 225,000	≥ 225,000
	Unitary heating efficiency	n/a	n/a
* Fan control		Variable Volume	Variable Volume
Supply airflow		25,064	26,247
Outdoor airflow		9,910	11,320
Demand control ventilation		No	No
* Economizer high-limit shutoff		55	55

Figure 8 Auto-generated LEED Minimum Energy Performance Inputs

This case study demonstrated how the new workflow can facilitate full automation of certain model compliance documentation. All relevant data definitions only needed to be included in the UI, which is capable of holding inputs that have no direct way of being expressed in an EnergyPlus or OpenStudio input file. The improved speed of compliance documentation file generation also allows modelers to focus on model improvements for upcoming deadlines, without having to set aside days to complete required code and beyond-code compliance documentation.

CONCLUSION

The workflow in this study was developed on the notion that achieving full start-to-end BEM workflow automation requires a data structure that acknowledges it cannot anticipate all future needs of a modeler. BEM is a deliberately open-ended field, which at its most effective can actively lead project conversations to more efficient and innovative HPBs. In order to stay ahead of design, rather than lag behind a design team's analysis needs, modelers need to be able to adapt quickly to react to changing project requirements and shortening project timelines. This study demonstrates how a scalable, flexible, and extensible workflow framework can allow modelers to freely define data types and easily send relevant data to analysis tools.

The ability to consolidate all relevant information in one data model prevents re-work for input entry and provides start-to-end continuity throughout a project. Such breaks, either from switching of modeling tools or the need for exceptional calculations or workarounds, can disrupt a modeler's sense of how a model behaves and reacts to various updates and changes. This improved continuity can lend additional confidence to model results.

Because aeiSIMPLE was written around a standardized data definition, it can function regardless of the type or version of the simulation engine used. While this study utilized EnergyPlus for its calculation method, the framework is built such that writing future input translations to another engine would allow for running simulations in different tools instantaneously with the same data model. This also ensures that automation efforts can last beyond the use of any one simulation tool.

Under this framework, adding capabilities to explore new analyses will hold up for future projects, eliminating the need for re-work and expanding possibilities for further design explorations without increasing project budgets for BEM. Developing new analysis methods, then, becomes a way of creating roadmaps for building data flows. It means that once a problem is solved one time, it is solved for any future projects, freeing up time for critical thinking and research of upcoming industry technologies and practices. This allows modelers to start asking bigger questions about their models.

FUTURE WORK

There are steps in the energy modeling workflow that do not lend themselves to full automation. However, this study establishes an engine-neutral framework through which modelers can decide which steps can and should be automated in their workflows.

An extensible data model removes constraints on the analyses that can be completed from a singular model definition. Therefore, additional analyses like fully integrated water balances can potentially be incorporated in the workflow without requiring duplicate inputs. Concepts like baseline model generation are also achievable within this framework. A more extensive framework that includes other areas of study in building performance modeling, including life cycle cost analysis, occupant experience, and wellness, is also possible. In this framework, each project brings with it new opportunities for creating analysis pathways to be explored in future projects, instilling a continued sense of progress in what BEM can accomplish.

© 2020 ASHRAE (www.ashrae.org) and IBPSA-USA (www.ibpsa.us).
For personal use only. Additional reproduction, distribution, or transmission in either print or digital form is not permitted without ASHRAE or IBPSA-USA's prior written permission.

REFERENCES

Ellis, Peter. 2015. “Parametric Modeling with Templates and Scripting.” in *2015 ASHRAE Energy Modeling Conference*.

Fleming, Katina, N. Long, and A. Swindler. 2012. “Building Component Library: An Online Repository to Facilitate Building Energy Model Creation; Preprint.” *NREL*.

Global Alliance for Buildings and Construction. 2018. *2018 Global Status Report*.

Goldwasser, David. 2018. “Create Baseline Building.” *NREL*. Retrieved (https://bcl.nrel.gov/node/83661).

Guglielmetti, R., D. Macumber, and N. Long. 2011. “OpenStudio: An Open Source Integrated Analysis Platform; Preprint.” *NREL*.

Hemsath, Timothy. 2013. “Conceptual Energy Modeling for Architecture, Planning and Design: Impact of Using Building Performance Simulation in Early Design Stages.” *Proceedings of BS 2013: 13th Conference of the International Building Performance Simulation Association* 376–84.

Macumber, D. L., Brian Ball, and N. Long. 2014. “A Graphical Tool for Cloud-Based Building Energy Simulation.” *2014 ASHRAE/IBPSA-USA Building Simulation Conference* 87–94.

Miller, Clayton, Christian Hersberger, and Marcus Jones. 2013. *Automation of Common Building Energy Simulation Workflows Using Python*.

Parker, Andrew, Philip Haves, Subhash Jegi, Vishal Garg, and Baptiste Ravache. 2017. “Development of Automated Procedures to Generate Reference Building Models for ASHRAE Standard 90.1 and India’s Building Energy Code and Implementation in OpenStudio.” in *Building Simulation 2017*. United States.

Philip, Santosh. 2019. “Eppy.”

Rao, Sagar, David Conant-Gilles, Yiyuan Jia, and Brittany Carl. 2018. “Rapid Modeling of Large and Complex High Performance Buildings Using EnergyPlus.” *SimBuild 2018*.

Rocky Mountain Institute. 2011. *Collaborate and Capitalize: Post-Report from the BEM Innovation Summit*.

Roth, Amir, Jamie Bull, Scott Criswell, Peter Ellis, Jason Glazer, David Goldwasser, and D. Reddy. 2018. “Scripting Frameworks For Enhancing EnergyPlus Modeling Productivity.” *SimBuild 2018* 312–19.

Roth, Amir, David Goldwasser, and Andrew Parker. 2016. “There’s a Measure for That!” *Energy and Buildings* 117:321–31.

U.S. DOE. 2019. “Getting Started.” *EnergyPlus Version 9.2.0 Documentation*.

Weaver, E., N. Long, K. Fleming, M. Schott, K. Benne, and E. Hale. 2012. “Rapid Application Development with OpenStudio: Preprint.” *NREL*.

Zhang, Yi and Ivan Korolija. 2010. *Performing Complex Parametric Simulations with JEPlus*.

© 2020 ASHRAE (www.ashrae.org) and IBPSA-USA (www.ibpsa.us).
For personal use only. Additional reproduction, distribution, or transmission in either print or digital form is not permitted without ASHRAE or IBPSA-USA's prior written permission.