A New BIM to BEM Framework: The Development and Verfication of an Open-Source gbXML to EnergyPlus Translator for Supporting Building Life Cycle Performance Analysis

Building Simulation 2019 Conference

Weili Xu[1], Adrian Chong[2], Khee Poh Lam[2], Haopeng Wang[1]

[1]BuildSimHub Inc, Pittsburgh, U.S.A

[2]Department of Building, School of Design and Environment
National University of Singapore, Singapore

Abstract

gbXML is an open schema that supports data interoperability between BIM applications and different building design software tools (gbXML, 2018). Its capability to bring in geometric and construction information can help reduce the time and uncertainty of the energy modeling process (Ham and Golparvar-Fard, 2015). However, during the design cycle, when both geometric and system data are updated continuously, the lack of methods to quickly insert system information in the BIM-exported gbXML has a significant impact on the modeling efficiency as well as the model integrity. This project aims to develop an open-source gbXML to EnergyPlus translator, the gbEplus, which provides flexibility for user-defined system modules plug-in. It adopts the OpenStudio reverse translator to ensure a quality geometry and construction data conversion. A plug-in interface is designed as the gateway for importing data from external resources to gbXML-EnergyPlus translation process in the run-time, and all the previously developed system modules can be stored in a user-defined library for reuse in the future. The gbEplus was developed and validated with models from Phase II gbXML geometry benchmark test documents. Two applications, for parametric design and model calibration, were developed to demonstrate the adaptability of this new BIM-BEM tool.

Introduction

The ability to import data from building information models (BIM) has the potential to significantly reduce the time and effort needed in the building energy modeling (BEM) process. The green building XML (gbXML), a data schema developed to share building information primarily with building energy modeling tools, is widely adopted in many BEM applications. However, the BIM-BEM data interoperability issues were reported in various research (Steel et al., 2012; Cemesova et al., 2015).

A typical workflow starts from a BIM model in Revit, ArchiCAD or any other BIM tool. Almost all BIM tools have a conversion function for creating a gbXML model. However, energy modelers have little control over this function, thus, the conversion misses essential data for energy models. Therefore, modelers often have to spend a significant amount of time to prepare the data before executing simulations. The whole process is labor-intensive. What's worse, building designs are continuously changing during the early and schematic design phases. In each iteration, modelers need to repeat the same workflow to calculate the energy impact of the new design. Such workflow significantly affects the turnover time for energy models and eventually decreases the value of modeling.

Tools such as DesignBuilder introduces the concept of template to semi-automate this process. The template stores a set of model assumptions so the same inputs can be reused in different designs. However, the technique still requires creating and linking templates to zones manually in each design iteration. Similarly, a recent study tries to address this issue by introducing an intermediate data schema idfXML as a template for keeping internal loads and HVAC assumptions. The proposed idfXML schema is an XML-based data structure that enables scripting. Such a framework has the potential of reducing the time for BIM to BEM conversion (Dimitriou et al., 2016). However, this function was designed to be part of a model calibration framework rather than a stand-alone component.

This paper presents the development of a stand-alone BIM-BEM translator that can directly convert a BIM exported gbXML model into a ready-to-simulate EnergyPlus model. The application named gbEplus is maintained on GITHUB[1]. gbEplus is a JAVA-based application designed to be a light-weight component. It implements an innovative data plugin interface, which allows software tools and web applications to benefit from its core functions. The data plugin interface allows designers to customize detail building system configurations, modularizes the building systems into the defined system categories, and store and manage them in data libraries. More importantly, the data that is registered on the plugin interface will be imported during the translation run-time. This feature implies that all the converting processes can be automated with no human intervention. Through the data plugin interface, gbEplus has the potential to improve the efficiency of BIM-BEM translation significantly.

Methods

Inputs to BEM are often obtained from multiple sources including drawings, specifications or other resources that are available. Although the gbXML schema include detailed data tags that have the potential to cover the majority of this information, the BIM exported gbXML model often includes the information of geometry and construction only. Furthermore, the data format from the multiple resources can vary significantly. Therefore, the primary focus of the gbEplus development is to provide the freedom for inserting data to multiple breakpoints in the gbXML-EnergyPlus translation process with a user-specified data format. Figure 2 shows the gbEplus workflow. Each component is color-coded to indicate its functions. The "BIM exported gbXML" is the gbXML model exported from a BIM tool such as Autodesk Revit® or GraphiSoft ArchiCAD®. Before converting, energy modelers need to evaluate the inputs to determine whether to supply a user-defined data template. For instance, if energy modelers have the lighting system specification from the manufacturer, these data can be easily defined in a preferred data format and inserted to the break-point when translating the lighting system. The break-points, which are colored in orange in Figure 2, are the places where energy modelers can insert collected data into the model during the translation process. Currently, there are eight break-points defined, which correspondent to eight categories: opaque construction and fenestration, lighting, outdoor air, equipment and occupant schedules as well as the HVAC systems. The user-defined data has the highest priority among all the data sources to be used in the translated model, followed by the gbXML data and the default library. Therefore, once each break-point is reached, gbEplus will examine the availability of user-defined data first, and then check both data in the gbXML model and default library in order to ensure that all the essential modeling inputs are in place for simulation.

[1]gbEplus Repo: https://github.com/weilixu/gbEplus

To support such a workflow, the team has developed three modules in the gbEplus, namely the EnergyPlus script engine, gbXML translator, and data plugin interface. Figure 1 illustrates the overall workflow between all modules. The process started with a BIM exported gbXML model. gbXML translator module manages the overall translation workflow. It communicates with the data plugins, which contain a group of user-defined data, through the plugin interface, manages all model meta-data such as number of floors and space areas, and transfer the organized data into EnergyPlus script engine. The EnergyPlus script engine will reform the data into the IDF format by following the input data dictionary (IDD).

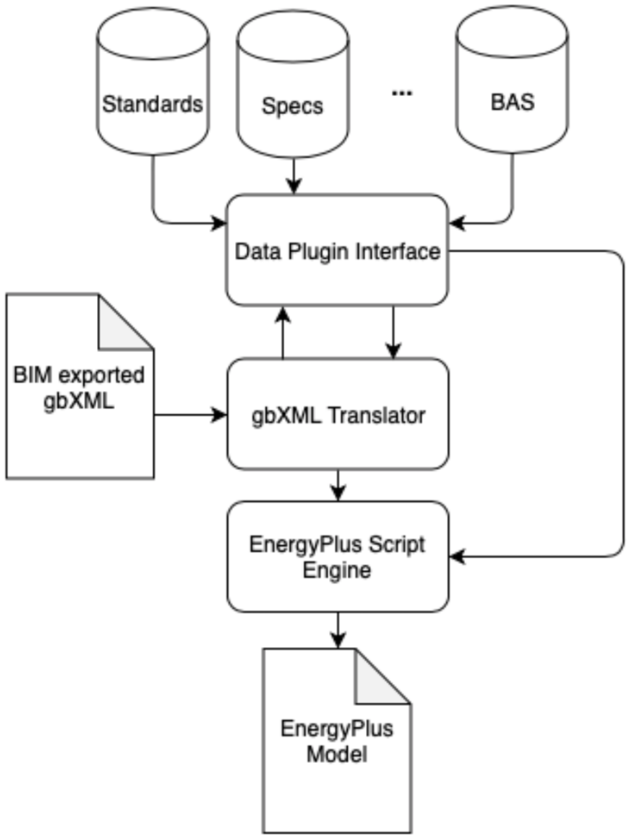


Figure 1: gbEplus system modules

EnergyPlus script engine

The EnergyPlus script engine provides the capabilities of writing, modifying and removing data in the EnergyPlus model using scripts. The engine supports three API requests. They are addEnergyPlusObject, modifyAField and deleteEnergyPlusObject. Once requests are initialized, EnergyPlus validation will check the data integrity against the EnergyPlus IDD. If the data is validated, it will be inserted into EnergyPlus model. Otherwise, error messages will be prompted in the command line to indicate the location of erroneous data. Figure 3 depicts the EnergyPlus script engine framework. In addition, several features were implemented by the team to allow a flexible EnergyPlus input/output (I/O) using this

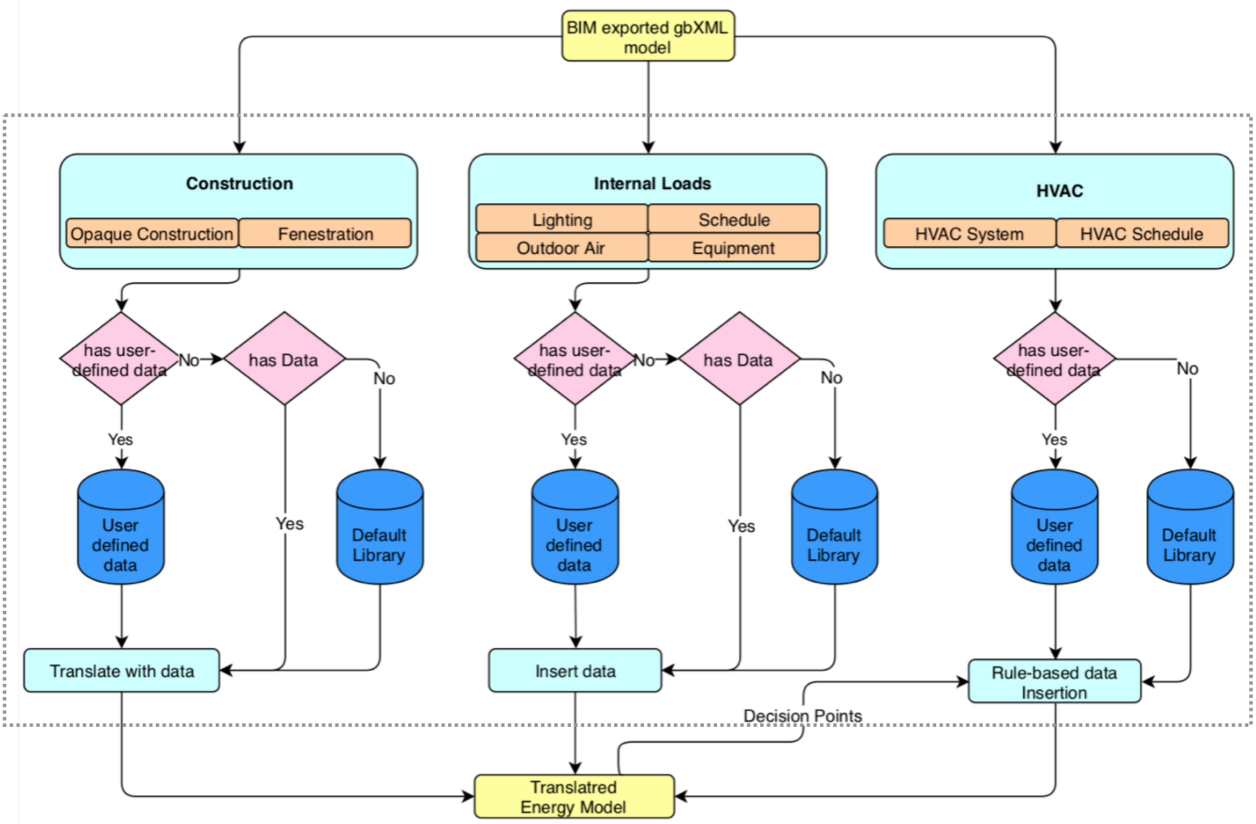


Figure 2: gbEplus workflow overview

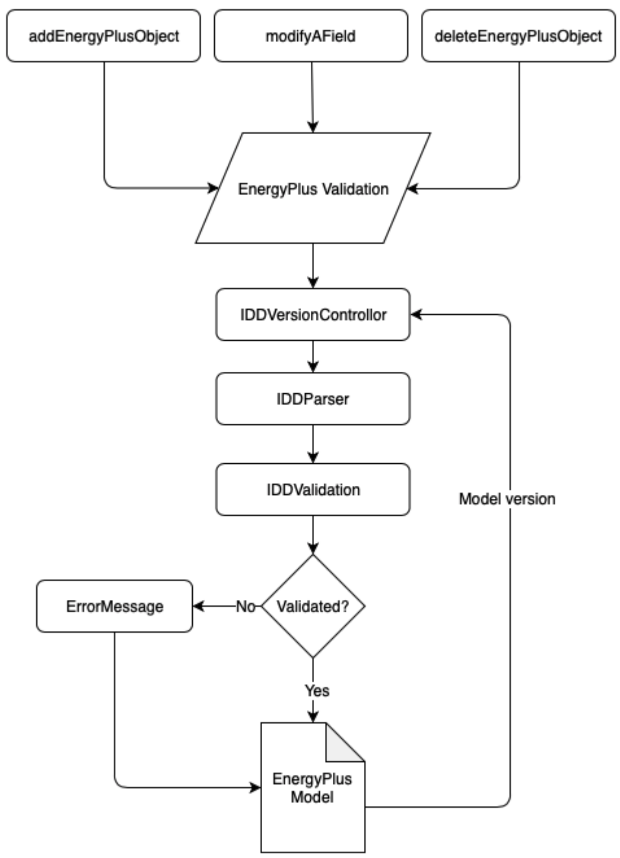


Figure 3: EnergyPlus script engine operation workflow

engine.

- This workflow supports batch API functions so that it can save the computer processing cost.
- The IDD validation supports extension fields so that modelers will not receive an error message even when the EnergyPlus object reaches or go beyond the maximum number of fields.
- Error requests will still be inserted into the EnergyPlus model because those operations could be intentional such as inputs for EnergyPlus parametric module.

The API requests typically need the EnergyPlus key and value pairs. Figure 4 shows a sample code on how to add a new EnergyPlus object using scripting. Listing 1 displays the EnergyPlus data after processing by the script engine.

Listing 1: EnergyPlus data

```
! - This is a sample
Schedule:Constant,
  Sample Schedule, !- Name
  On/Off, !- Schedule Type Limits Name
  2; !- Hourly Value
```

gbXML translator

The gbXML translator implements the translation logic following the OpenStudio gbXML reverse trans-

```java
private void addObjectExample() {
    IDFObject idfObject = new IDFObject("Schedule:Constant", 4);
    idfObject.setTopComments(new String[] { "!- This is an sample" });

    idfObject.setIndexedStandardComment(0, "Name");
    idfObject.setIndexedData(0, "Sample schedule");

    idfObject.setIndexedStandardComment(1, "Schedule Type Limits Name");
    idfObject.setIndexedData(1, "On/Off");

    idfObject.setIndexedStandardComment(2, "Hourly Value");
    idfObject.setIndexedData(2, value);
    idfFileObject.addObject(idfObject);
}
```

Figure 4: Java addEnergyPlus API code sample

lator[2]. However, the OpenStudio translator utilizes its OSM data structure, which is not available for gbEplus. Therefore, migrating the logic from OSM to IDF with the use of EnergyPlus script engine was performed to ensure the integrity of the program. A set of five validation tests were performed to ensure the accuracy of the translation (Chong et al., 2019). The test cases were built in Revit according to the instructions provided in Phase II gbXML geometry benchmark test documentation [3] (PNNL, 2014). Table 1 presents a visual comparison of the geometry between the native BIM (Revit) and the energy model (EnergyPlus). The image shows that the gbXML translator is capable of identifying the spaces and surfaces in the BIM models.

Besides the visual comparison, geometry metrics, including gross wall area, window opening area, roof area, and zone volume, were used to evaluate the translation errors. The weighted absolute percentage error (WAPE) was used to quantify the differences when comparing the metrics between BIM and BEM. The WAPE is calculated based on Equation 1, where n is the number of surfaces or zones; A_i is the area of the surface or volume of the zone in the BIM (calculated in Revit), and F_i is the area of the surface or volume of the zone in the BEM.

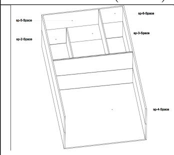
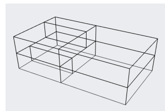
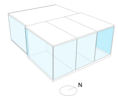
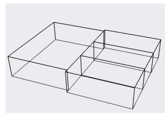
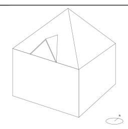
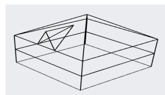
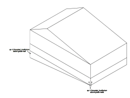
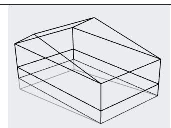
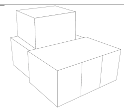
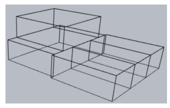
$$WAPE[\%] = 100 \cdot \frac{\sum_{i=1}^{n} |A_i - F_i|}{\sum_{i=1}^{n} A_i} \qquad (1)$$

Table 2 illustrates the comparison between the calculated WAPE for the translated energy model and the BIM model. Majority of the WAPEs were less than 1%, which implied the translations were nearly identical. The only exception was the gross wall area in Case 3 ($WAPE = 3.720\%$) and Case 4 ($WAPE = 1.716\%$). A detailed investigation on these two cases revealed that the error was due to the different definition of wall surface boundary in BIM and gbXML with a tilted roof. Figure 5 displays the results of the detailed investigation. During the translation, the inner surface of the wall was used to represent this surface in the BEM, resulting in walls with a slightly

[2]OpenStudio gbXML Reverse Translator: `https://github.com/NREL/OpenStudio/blob/develop/openstudiocore/src/gbxml/ReverseTranslator.cpp`

[3]gbXML Test Case Documentation: `http://www.gbxml.org/downloads/gbxmltestcasedocumentation_PhaseII.zip`

Table 1: Comparison of 3D view for Revit model as compared to BIM-based EnergyPlus model (PNNL, 2014; Chong et al., 2019)

	gbXML Test	Native BIM (Revit)	Energy Model (EnergyPlus)
Case 1	Second level space boundary		
Case 2	Multiple adjacent overlapping		
Case 3	Single gabled roof		
Case 4	Sloping slab on grade		
Case 5	Multiple adjacent overlapping		

larger wall area since the wall would be extended to the roof. Further details of the evaluation of these test cases can be found in (Chong et al., 2019).

Table 2: Weighted absolute percentage error (WAPE) for gross wall area, window opening area, and zone volume of the BIM translated BEM (Chong et al., 2019).

	Gross Wall Area [%]	Roof Area [%]	Zone Volume [%]
Case 1	0.004	0.001	0.000
Case 2	0.0146	0.000	0.000
Case 3	3.720	0.002	0.001
Case 4	1.716	0.001	0.000
Case 5	0.907	0.000	0.000

Data plugin interface

The data plugin interface allows energy modelers to add preferred modeling data in the translation process. The data plugin is categorized into eight breakpoints as shown in Figure 2. A plugin is activated when the gbXML translator reaches a break-point and matches the break-point's category. For example, when the translator is translating lighting system, it will examine whether a lighting data plugin is registered in the gbEplus. Once the data plugin is identified, gbEplus will execute this data plugin and bypass the BIM data or data in the default library.

The logic on how data plugin interacts with the gbXML translator and EnergyPlus script engine is

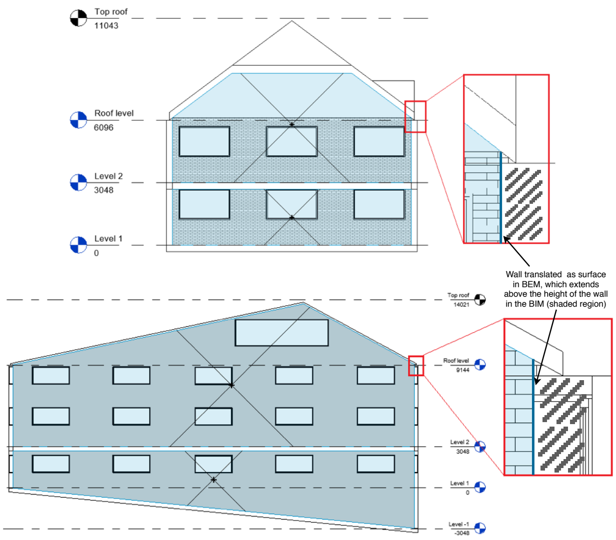


Figure 5: Illustration of the wall surface boundary with tilted roof in Case 3 and Case 4 (Chong et al., 2019).

depicted in Figure 6. When the translation process reaches a break-point, the translator requests for the availability of data plugin. As soon as it receives positive feedback, the translator asks the data plugin to provide data for the model. Inside a data plugin, two types of data can be specified, namely key-value mapped data and system data. The key-value mapped data can be a lighting power density value that links to a set of specific spaces, or occupant density mapped with space type. On the other hand, system data is a set of data that describes a system for instance, the HVAC system. Based on the data type, the data plugin can determine whether to map the data through the gbXML translator or insert the data into EnergyPlus directly. The data plugin interface offers flexibility for adding user-defined model inputs through multiple resources including standards, manufacturer submittals, and building management systems (BMS). It add the potential for performing BIM-based iterative designs or model calibrations using gbEplus.

Application: Parametric design modeling

In the early design phase, exporting a BIM model to a ready-to-use baseline energy model is often time-consuming and labor-intensive. A set of baseline data plugins were built in the gbEplus repository [4]. The data includes construction, lighting system, outdoor air, and HVAC systems and these data were extracted from ASHRAE standard 90.1 (ASHRAE, 2010a) and ASHRAE standard 62.1 (ASHRAE, 2010b).

Figure 7 shows the test case adopted for the BIM-BEM baseline application. The test case was con-

[4]gbEplus repository: `https://github.com/weilixu/gbEplus`

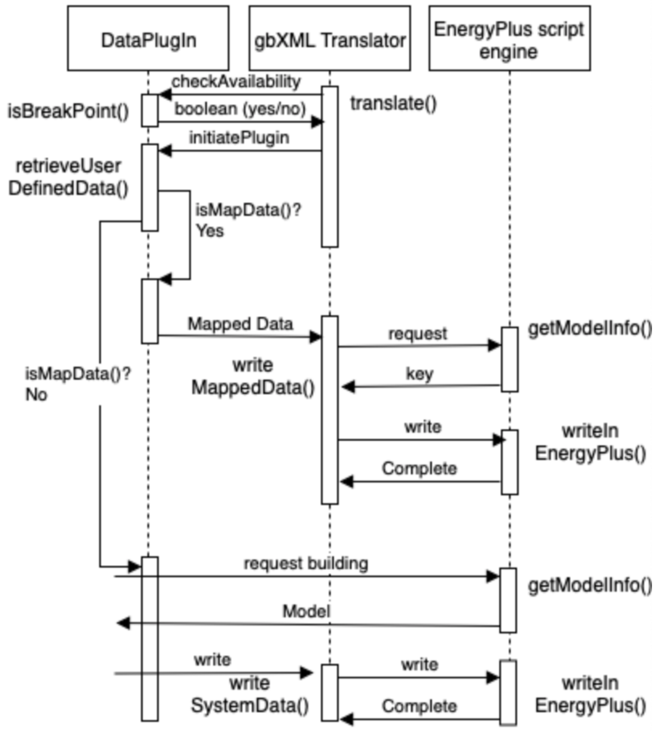


Figure 6: Illustration of data plugin workflow

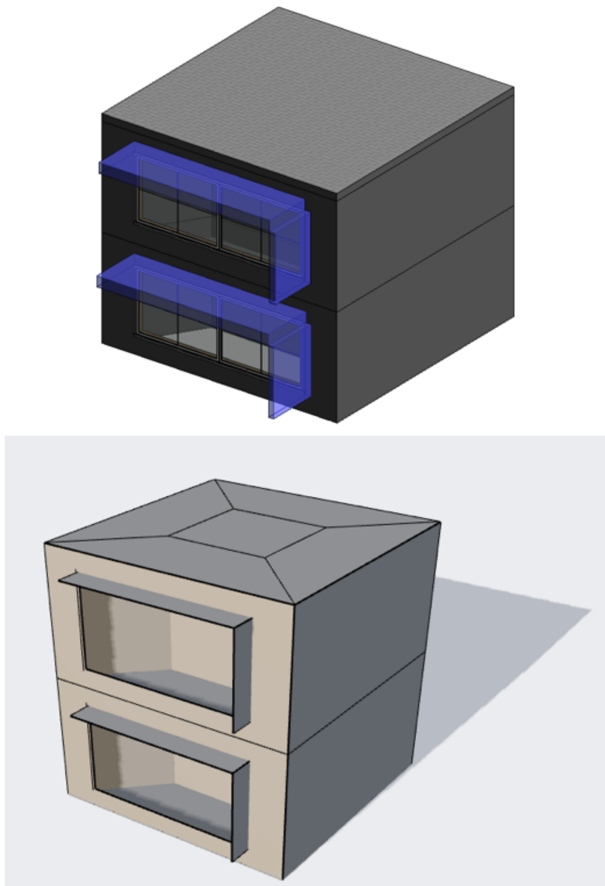


Figure 7: Test model in BIM (Upper) and in BEM (Lower)

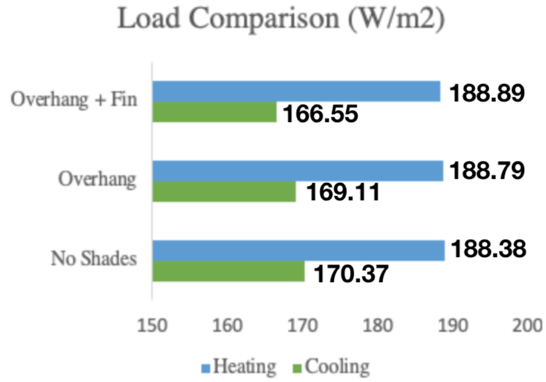


Figure 8: Cooling and heating loads comparison for the three shading configurations.

structed in Revit. The building had two stories with a total of 100 m^2 floor area. Windows were installed on the south and west facing walls. Each of the windows has an overhang and a single right-side fin.

Model data

In this study, the building was located in Chicago, IL. Therefore, the baseline data for climate zone 5 was programmed and registered in the gbEplus. The goal was to examine a better shading design strategy among three proposed solutions: no shades, overhangs only, and overhangs with right-side fins. The geometry modification was conducted in Revit, and the energy models were then generated for comparing the building loads. Baseline data were listed in Table 3.

Table 3: Baseline data for the case study

System	Value	Unit
Roofs U-Value	0.184	W/m2-K
Walls U-Value	0.513	W/m2-K
Floors U-Value	0.214	W/m2-K
Windows U-Value	2.38	W/m2-K
Lighting power density	8.8	W/m2
Outdoor air per area	0.6	L/s-m2
Outdoor air per people	2	m2/p
HVAC system	PSZ-AC	per zone

Results

In Revit, three configurations of the shades were created on the south and west facade of the building. All cases were exported through the gbXML export function. With the gbEplus and baseline data plugins, three energy models were automatically generated, and the designer ran the simulations for comparison. The comparison results of these three cases were obtained and displayed (Figure 8) in under one hour. The results indicated that adding shades to the south and west windows did not have a significant impact on both cooling and heating loads. The heating load was almost identical among the three cases. This observation suggested that adding shading devices such as overhangs and fins in this small building did not have a significant impact on the building loads in Chicago. The possibility of investigating energy and load performance of design options within the Revit environment were examined with the help of gbEplus through web APIs. This piece of work was published on Github (`https://github.com/bimlauncher/buildsimhub-demo`) and can be downloaded for testing.

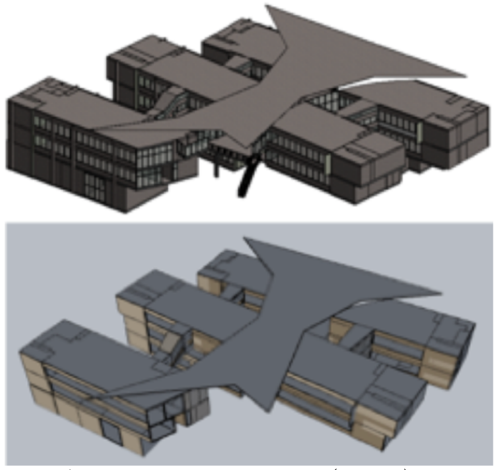


Figure 9: Actual building in BIM (upper) and in BEM (lower)

Application: model calibration

The gbEplus is developed to be adaptable to any energy modeling data sources. A model calibration study typically extracts modeling data from various resources, which makes it a perfect demonstration for gbEplus. Furthermore, a BIM-based energy calibration process eliminates the overhead of creating energy models. The results of the calibration can support operation optimization, fault detection (Dong et al., 2012) and retrofit analysis (Miller et al., 2014) throughout the life cycle of a building (Chong et al., 2019).

The case study building is an office building located on the campus of the National University of Singapore (NUS) in Singapore. The building is 3-story high with a total gross floor area of 5,445 m^2.

The BIM file was first created in Autodesk Revit, containing geometry and construction data, and exported into gbXML using the native gbXML export function. One year of electrical energy consumption data from 1 January 2014 to 31 December 2014 was collected and used to train the energy model. The weather file used for calibration was the Actual Meteorological Year (AMY) weather data from the Singapore Changi airport weather station (WMO 486980). Figure 10 depicts the workflow of the BIM-based energy calibration process. gbEplus translated the BIM exported gbXML into an EnergyPlus model with data from BIM, baseline data plugin and user-defined data. A sensitivity analysis was then performed on the translated model to screen critical calibration parameters. Combined with data collected

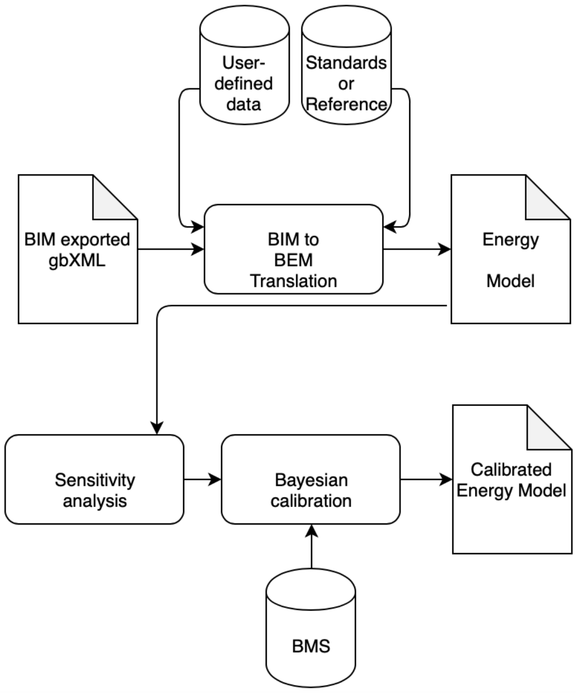


Figure 10: Actual building calibration workflow

from BMS, a Bayesian calibration was conducted to produce a calibrated energy model. The Bayesian calibration was formulated according to (Kennedy and O'Hagan, 2001). To reduce the computation time, a Gaussian process model was used as the emulator for energy models(Lim and Zhai, 2017). Furthermore, Hamiltonian Monte Carlo was employed as the sampling method as it was proved to be more efficient in energy model calibration (Chong and Lam, 2017). The overall calibration workflow and detailed algorithms can be found in Chong and Menberg (2018).

Model data

In this application, three data types were defined namely "BIM", "User-defined Data", and "Standards or References". The "BIM" data was translated through the gbXML translator directly, and the "User-defined Data" such as lighting power density and equipment power density were derived from the sensitivity analysis or BMS data. In addition to these two resources, a full set of baseline data (including construction layers, material properties, lighting requirements, outdoor air requirements, and HVAC systems) from Singapore Building and Construction Authority (BCA) Green Mark Scheme (BCA, 2016) was built into the "Standards or References" data plugins.

Results

The calibration results were assessed by both the coefficient of variance of the root mean square error (CVRMSE) and normalized mean biased error (NMBE). Equations 2 and 3 presents the formulation of these two coefficient. As specified in ASHRAE Guideline 14, an acceptable calibrated model should

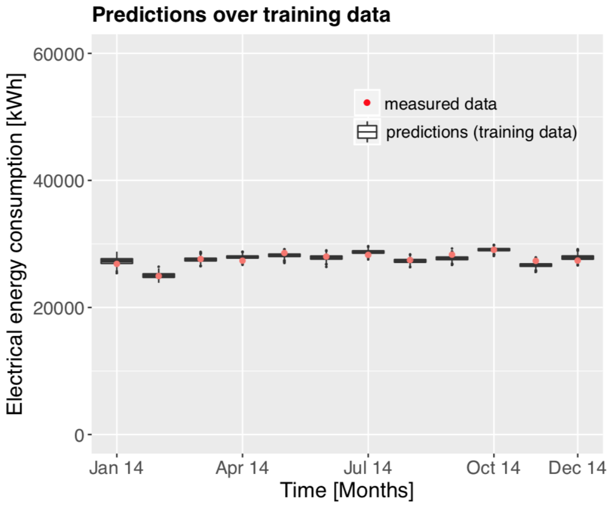


Figure 11: Box and whisker plots showing the comparison between predicted distribution of electricity consumption against measured data (Chong et al., 2019)

have a lower than 15% CVRMSE and 5% NMBE for monthly resolution (ASHRAE, 2002).

$$CVRMSE[\%] = 100 \times \frac{\sqrt{\sum_{i=1}^{n}(y_i - \hat{y_i})^2 / (n-1)}}{\bar{y}} \quad (2)$$

$$NMBE[\%] = 100 \times \frac{\sum_{i=1}^{n}(y_i - \hat{y_i})}{(n-1) \times \bar{y}} \quad (3)$$

where y_i = observed value at hour i, $\hat{y_i}$ = predicted value at hour i, $\bar{y}$ = mean energy consumption of n observations; and n = number data points.

Figure 11 shows the comparison between the measured monthly electricity data and the distribution of electricity consumption generated from calibration. The results indicate that the measured data falls within the 95% confidence interval in every month. The calculated CVRMSE is 1.04%, and NMBE is -0.01%, which are well below the 15% and 5% threshold.

Conclusion

This study introduced a newly developed gbXML to EnergyPlus translator that could directly convert a BIM exported gbXML model into a ready-to-simulate EnergyPlus model. The translator featured a core gbXML-EnergyPlus translation module, an EnergyPlus script engine, and a Data Plugin Interface. Through the interactions among these three modules, modelers could plug and play with a various set of data for different energy modeling projects using BIM.

Two applications integrated with gbEplus were introduced in this study as well. The first application plugs a set of ASHRAE standard data into gbEplus to support shading design comparison. The framework suggested a possibility for developing an energy modeling workflow inside the Revit framework.

The second application relates to the energy model calibration. The case study not only calibrated the model at monthly resolution by integrating gbEplus with Bayesian calibration framework but also demonstrated the possibility of establishing a life-cycle calibration framework using BIM.

However, a few challenges remain in the translation process. First of all, it is challenging to validate the gbXML models. gbEplus does not check the validity of a gbXML model. So a gbXML model could crash the translation process if it does not follow the latest gbXML version. One solution the team is working on is to develop a web-based validation tool that can validate the gbXML before the translation process starts. Another key challenge for using gbEplus is to translate gbXML models with geometry defects. The current gbEplus workflow begins with a validated gbXML model. It does not have control over the process of exporting the gbXML model from a BIM model. One piece of future work will involve investigating and summarizing commonly seen defects, particularly the building zoning in a curved shape block, in the gbXML and implementing plausible solutions under the gbEplus framework to resolve those geometric issues.

Besides the challenges, the team is also developing a logic engine which supports multiple data plugins under the same category. Based on user-defined logic, the logic engine will extract the model meta-data to select one of the data plugins to be applied in the translation. This will allow complex model generations such as ASHRAE baseline automation.

Acknowledgment

This research is funded by the Republic of Singapore National Research Foundation (NRF) through the Building and Construction Authority (BCA) Green Buildings Innovation Cluster (GBIC) R&D Grant. The authors also thank the U.S. Department of Energy (DOE) SBIR program for supporting the development of gbEplus as part of BuildSimHub Cloud.

References

ASHRAE (2002). Guideline 14-2002, measurement of energy and demand savings. *American Society of Heating, Ventilating, and Air Conditioning Engineers, Atlanta, Georgia*.

ASHRAE (2010a). ANSI/ASHRAE/IES Standard 90.1 Energy Standard for Buildings Except Low-Rise Residential Buildings. *American Society of Heating, Ventilating, and Air Conditioning Engineers, Atlanta, Georgia*.

ASHRAE (2010b). ASHRAE 62.1-2010 Ventilation for Acceptable Indoor Air Quality. *American Society of Heating, Ventilating, and Air Conditioning Engineers, Atlanta, Georgia*.

BCA (2016). BCA Green Mark for non-residential buildings NRB: 2015. *Singapore Building and Construction Authority*.

Cemesova, A., C. J. Hopfe, and R. S. Mcleod (2015). Passivbim: Enhancing interoperability between bim and low energy design software. *Automation in Construction 57*, 17–32.

Chong, A. and K. P. Lam (2017). A comparison of MCMC algorithms for the Bayesian calibration of building energy models. In *Proceedings of the 15th IBPSA Building Simulation Conference*.

Chong, A. and K. Menberg (2018). Guidelines for the Bayesian calibration of building energy models. *Energy and Buildings 174*, 527–547.

Chong, A., W. Xu, S. Chao, and N.-T. Ngo (2019). Continuous-time bayesian calibration of energy models using BIM and energy data. *Energy and Buildings 194*, 177 – 190.

Dimitriou, V., S. K. Firth, T. M. Hassan, and F. Fouchal (2016). Bim enabled building energy modelling: development and verification of a gbxml to idf conversion method.

Dong, B., Z. O'Neill, Z. Li, D. Luo, S. Madhusudana, S. Ahuja, and T. Bailey (2012). An integrated infrastructure for real-time building energy modeling and fault detection and diagnostics. *Proceedings of SimBuild 5*(1), 448–455.

gbXML (2018). gbxml-about. Last accessed 20 Jan 2019.

Ham, Y. and M. Golparvar-Fard (2015). Mapping actual thermal properties to building elements in gbxml-based bim for reliable building energy performance modeling. *Automation in Construction 49*, 214–224.

Kennedy, M. C. and A. O'Hagan (2001). Bayesian calibration of computer models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 63*(3), 425–464.

Lim, H. and Z. J. Zhai (2017). Comprehensive evaluation of the influence of meta-models on Bayesian calibration. *Energy and Buildings 155*, 66–75.

Miller, C., D. Thomas, S. D. Irigoyen, C. Hersberger, Z. Nagy, D. Rossi, and A. Schlueter (2014). Bim-extracted energyplus model calibration for retrofit analysis of a historically listed building in switzerland. *Proceedings of SimBuild 2014*.

PNNL (2014). Phase ii gbxml test case documentation. Last accessed 20 Jan 2019.

Steel, J., R. Drogemuller, and B. Toth (2012). Model interoperability in building information modelling. *Software & Systems Modeling 11*(1), 99–109.