Improving the Interoperability of gbXML Data Model through Redefining Data Mapping Rules of HVAC Systems

Ruiji Sun
Student Member ASHRAE

Zhizhang Hu

Krishnan Gowri
Fellow ASHRAE

Weili Xu, PhD
Associate Member ASHRAE

ABSTRACT

Green Building XML (gbXML) is one of the most prevalent data models used for Building Information Modeling (BIM). It enables interoperability between BIM and Building Energy Modeling (BEM). BEM is often used in building performance analysis software tools. However, through interviewing twenty engineers and building energy modeling professionals in the industry, it turned out that most gbXML files are only used for importing and exporting building geometry information. Information such as Heating, Ventilation, and Ait-conditioning (HVAC) systems and internal loads are rarely handled due to the lack of functionality in current BIM software. Taking account of more than 15% of the total energy consumption in the US is used by HVAC systems (DOE 2011), it is crucial to enable seamless HVAC data exchange between BIM software and BEM software.

This paper researched the definition rules of HVAC systems in the current gbXML schema 6.01 version. Firstly, through a detailed data mapping of ASHRAE baseline variable air volume and reheat system between IDF data model (EnergyPlus version 9.0) to corresponding elements in gbXML schema (version 6.01), interoperability issues were discovered and were concluded into three catecgories: missing components, the difficulty of decoding performance curves, and complex data mapping rules. Secondly, through redefining data mapping rules in current gbXML schema in terms of HVAC systems, the ASHRAE baseline system type seven is coded as a gbXML file. Finally, the gbXML file is validated through a medium office building case study. Revit 2020.1 Architecture is used in this study as the BIM tool. OpenStudio 2.9.1 and EnergyPlus 9.2.0 are used as the BEM tool. They are open-source and cross-platform, being adopted by lots of mainstream building performance analysis software.

Based on the result of this study, current gbXML schema version 6.01 is capable of defining HVAC systems data, but the data mapping rules need to be documented and presented. Redefined data mapping rules and improvement suggestions are proposed to the current gbXML schema in terms of HVAC systems. The improved interoperability will eliminate the duplicate generation of HVAC data and allows a bidirectional information update between BIM and BEM software, supporting a more accurate and efficient building performance analysis process.

INTRODUCTION

BIM-based building performance analysis

Building information model (BIM) is a digital representation of a building. It serves as a shared knowledge resource for communication in architecture, engineering, construction (AEC) and facility management (FM) industries (Eastman, C. et al. 2011). The first generations of data exchange models only include building geometry data,

Ruiji Sun and **Zhizhang Hu** are Master of Science students in the Center for Building Performance and Diagnostics, Carnegie Mellon University, Pittsburgh, PA. **Weili Xu** is the co-founder of BuildSimHub, Inc., Pittsburgh, PA. **Krishnan Gowri** is a senior consultant at Intertek Building Science Solutions, Inc., Seattle, WA.

transferring traditional 2-D drawings to 3-D object-based models. However, with the development of sustainable architecture, material properties, HVAC equipment, and electrical product data have been considered in current BIM data models. Green Building XML (gbXML) and Industry Foundation Class (IFC) are the two prevalent ones. (Dong B. et al. 2007). The new BIM-based performance analysis workflow makes the analysis process less expensive and labor-intensive and increases the accuracy of the performance simulation results. Especially during design phases, the value of BIM and performance analysis can be maximized (Moon, H. J. et al. 2011).

Interoperability of gbXML between BIM and BEM

Building energy modeling (BEM) is a subset of BIM. It allows more information inputs for building energy simulation such as HVAC systems data, operation schedules, envelope materials. BEM is mainly used for sustainable architecture design, HVAC systems design and operation, and building performance rating. The interoperability of gbXML between BIM and BEM should be seamless model translations or data exchange among disparate building design and performance analysis software tools. However, by interviewing more than twenty energy modelers, engineers, software developers, and other stakeholders, the current gbXML workflow only enables geometry data transformation. HVAC systems have to be manually created in BEM tools. In addition to the interviews, eleven current mainstream BIM and BEM software and tools are investigated in the gbXML interoperability from different aspects, shown in Table1.

Table 1. Data exchange as gbXML in BIM and BEM Software Tools

Software Tools	Geometry	Material Properties	HVAC Systems	Internal Loads
BuildSimHub	Yes	Yes	No	No
Design Builder 6.1	Yes	Yes	No	No
IES VE 2019	Yes	Yes	Yes	Yes
OpenStuido 2.9.0	Yes	Yes	Yes	Yes
HVAC Solution 9.5.1	No	No	Yes	Yes
TRACE 700 6.3.4	Yes	Yes	No	Yes
Revit 2020.1 Architecture	Yes	Yes	No	No
Revit 2020.1 MEP	Yes	Yes	No	No
Revit 2020.1 System Analysis	Yes	Yes	Yes	Yes
Spider gbXML Viewer 0.17	Yes	Yes	No	No
Talece BIMPort	Yes	No	No	No

Same as the interview results, most BEM software tools in the industry facilitate the gbXML as a data exchange model. However, compared with geometry, material properties, and internal loads information, HVAC systems data was turned out to be the least handled one. Therefore, to improve the interoperability of gbXML, HVAC systems data exchange is supposed to be enabled. A survey has reported that 78% of contractors ranked interoperability as the top way to increase the business value of BIM (McGraw Hill, 2009). Currently, some researchers are working on developing translator tools from gbXML directly to EnergyPlus, which is critical to the gbXML interoperability improvement (Xu, W. et al. 2019).

gbXML as an XML-based green building schema

gbXML is an XML-based data schema representing building information. XML is an extensible markup language. It is already a mature and powerful data model. Elements are the key objects of this language. They start with the opening tag "<tag>" and end with the closing tag "</ tag>". The attributes included in open tags are used to distinguish elements "<tag attribute = 'something' />". The content between the opening and closing tags may be text, other elements, or a mixture of them (W3Schools, 2017). Due to the extensibility of the language, XML has many derived forms. For example, based on XML data structures, Hypertext Markup Language (HTML) has developed a set of predefined tags and attributes to help display content on a website. Similarly, gbXML has the same XML data structure. It also has predefined tags and attributes that are specifically designed for green building information modeling (gbXML, 2017). Therefore, only XML files that follow the gbXML schema can be called gbXML. However, confusion occurs when the BIM model follows part of the gbXML schema and customizes some other elements and attributes. For example, in the Revit 2020.1 system analysis, the exported file follows the gbXML schema in the geometric data, but for HVAC system data, it adds two custom elements. For clarity, the customized gbXML is referred to as an XML file in this article.

METHODOLOGY

The objective of this research is improving the interoperability of gbXML. The study scope is limited to the ASHRAE baseline system seven, which is a Variable Air Volume (VAV) and reheat fan system. The validation method is a case study. Firstly, a research of current gbXML data schema was conducted through mapping the baseline HVAC system data one by one from the IDF data model to the gbXML schema. Three issues were found and analyzed. Secondly, based on the research result, the gbXML data mapping rules in terms of HVAC systems were redefined. A gbXML example file was created and was parsed to OpenStudio and EnergyPlus environment. Finally, the simulation result validated the effectiveness of the redefined data mapping rules. The workflow of this study is shown in Figure 1.

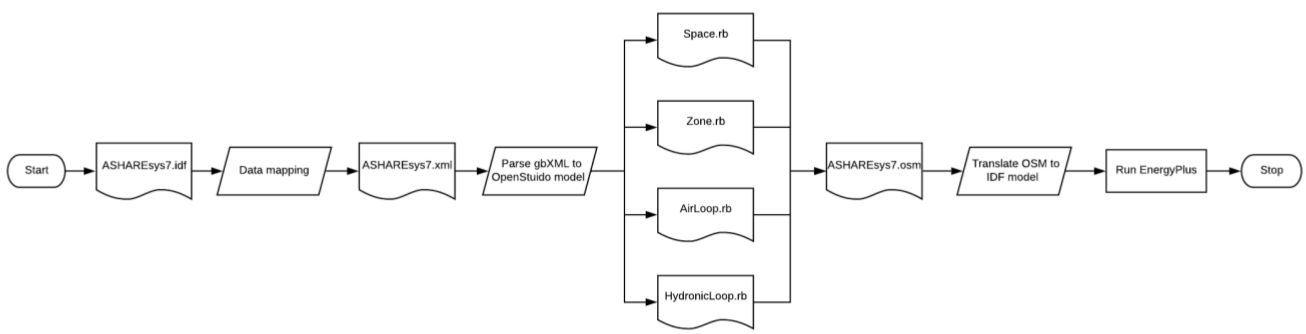

Figure 1. gbXML interoperability improvement workflow. Through a data mapping research of the ASHRAE baseline system type seven from IDF to gbXML, interoperability issues of current gbXML data schema (Version 6.01) have been figured out. The data mapping rules were redefined, and a gbXML model with HVAC systems data was created. For validating the redefined data mapping rules, the gbXML model was parsed and imported into OpenStudio through four customized OpenStudio measures. It was then translated to the IDF model for energy simulation using EnergyPlus.

DATA MAPPING

HVAC systems data mapping from IDF to gbXML

Compared with the XML, the IDF data structure is flattened. There are only three levels in the IDF model: “class”, “field”, and “objective”. The "field" is used to define the characteristics of a "class". The "objective" is for different objects, belonging to the same class. In the IDF data model of the ASHRAE baseline system type seven, HVAC related classes are “PlantLoop”, “CondenserLoop”, “ZoneHVAC: AirDistributionUnit”, “Boiler: HotWater”, etc. These classes have predefined IDF fields and each respective object has its customized value, following corresponding fields. Take the "Boiler: HotWater" class as an example to explain the data mapping process from IDF to gbXML in detail. IDF representation of the “Boiler: HotWater” is shown in Table 2.

Table 2. IDF Representation of a Boiler

Class	Field	Units	Object1
Boiler: HotWater	Name	-	Boiler1
Boiler: HotWater	Fuel Type	-	NaturalGas
Boiler: HotWater	Nominal Capacity	W	autosize
Boiler: HotWater	Nominal Thermal Efficiency	-	0.8
Boiler: HotWater	Efficiency Curve Temperature Evaluation Variable	-	LeavingBoiler
Boiler: HotWater	Normalized Boiler Efficiency Curve Name	-	Boiler Efficiency Curve
Boiler: HotWater	Design Water Flow Rate	m3/s	autosize
Boiler: HotWater	Minimum Part Load Ratio	-	0
Boiler: HotWater	Maximum Part Load Ratio	-	1.1
Boiler: HotWater	Optimum Part Load Ratio	-	1
Boiler: HotWater	Boiler Water Inlet Node Name	-	Boiler1 HW Inlet
Boiler: HotWater	Boiler Water Outlet Node Name	-	Boiler1 HW Outlet
Boiler: HotWater	Water Outlet Upper Temperature Limit	C	100
Boiler: HotWater	Boiler Flow Mode	-	LeavingSetpointModulated
Boiler: HotWater	Parasitic Electric Load	W	0
Boiler: HotWater	Sizing Factor	-	1.25
Boiler: HotWater	End-Use Subcategory	-	-

All fields of the “Boiler: HotWater” class in IDF are mapped to elements and attributes in gbXML data schema. To define the boiler as a “HydronicloopEquipemt”, its parent element “HydronicLoop” needs to be defined. In Table 3, gbXML defines HVAC systems in a hierarchy way. “AirLoop” and “HydronicLoop” are two main elements for primary system data. “AirLoopEquipemt” and “HydronicloopEquipemt” are used to define the secondary system data or equipment data. “Zone” and “Space” are used to assign HVAC systems to specific areas.

Table 3. gbXML Representation of a Boiler

1st Element	2nd Element	3rd Element	4th Element	Attributes	Object1
gbXML	HydronicLoop	-	-	loopType	HotWater
gbXML	HydronicLoop	-	-	fluidType	Water
gbXML	HydronicLoop	-	-	id	DHW1
gbXML	HydronicLoop	HydronicLoopEquipment	-	id	DHW1-B1
gbXML	HydronicLoop	HydronicLoopEquipment	-	equipmentType	Boiler
gbXML	HydronicLoop	HydronicLoopEquipment	Name	-	Boiler1
gbXML	HydronicLoop	HydronicLoopEquipment	Temp	unit	C
gbXML	HydronicLoop	HydronicLoopEquipment	Temp	tempType	Max
gbXML	HydronicLoop	HydronicLoopEquipment	Power	unit	Watt
gbXML	HydronicLoop	HydronicLoopEquipment	Power	useType	Heating
gbXML	HydronicLoop	HydronicLoopEquipment	Power	powerType	NaturalGas
gbXML	HydronicLoop	HydronicLoopEquipment	Efficiency	efficiencyType	BoilerEff
gbXML	HydronicLoop	HydronicLoopEquipment	Control	controlType	Boiler
gbXML	HydronicLoop	HydronicLoopEquipment	Control	minPowerRatio	0
gbXML	HydronicLoop	FlowControl	DesignFlow	unit	LPerSec

Issue one: missing components

During the data mapping process from the IDF data model to the gbXML schema. It turned out that some IDF fields do not have corresponding elements or attributes in gbXML. Take the “Boiler: HotWater” IDF class as an example. In the IDF data model, the “Maximum Part Load Ratio”, “Optimum Part Load Ratio”, “Boiler Flow Mode”, and “Sizing Factor” are clearly defined and they critical to boiler performance as well as building energy simulation. But in current gbXML schema, they can not be defined using existing elements or attributes, unless custom elements or attributes are created. Some other missing components are unnecessary for gbXML data exchange due to the hierarchy structure, such as “Boiler Water Inlet Node Name”, “Boiler Water Outlet Node Name” and “End-Use Subcategory”. They can be defined by order of “HydronicLoopEquipment” elements.

Issue two: the difficulty of decoding performance curves

The performance of HVAC equipment is essential to the acoustic quality, thermal quality, and energy efficiency of a building. However, in the gbXML schema, it is challenging to express performance curves information. Take a pump as an example, in “Pump: VariableSpeed” IDF class, a pump’s performance curve is defined by values at different coefficients of the part-load performance curve, respectively. It is clearly defined as linear, quad linear or quadratic equation, etc. Take biquadratic curves as an example, and the IDF representation is shown in Table 4. The biquadratic equation is:

$$z = a + bx + cx^2 + dy + ey^2 + fxy \qquad (1)$$

Table 4. IDF Representation of a quadratic curve

Class	Field	Units	Object1
Curve:Biquadratic	Name	-	CoolCapFTExample
Curve:Biquadratic	Coefficient1 Constant	-	0.757382
Curve:Biquadratic	Coefficient2 x	-	0.014666
Curve:Biquadratic	Coefficient3 x**2	-	0.000459
Curve:Biquadratic	Coefficient4 y	-	-0.00095
Curve:Biquadratic	Coefficient5 y**2	-	-0.00067
Curve:Biquadratic	Coefficient6 x*y	-	-0.00015
Curve:Biquadratic	Minimum Value of x	-	17.22222
Curve:Biquadratic	Maximum Value of x	-	21.66667
Curve:Biquadratic	Minimum Value of y	-	18.33333
Curve:Biquadratic	Maximum Value of y	-	46.11111
Curve:Biquadratic	Minimum Curve Output	-	-
Curve:Biquadratic	Maximum Curve Output	-	-
Curve:Biquadratic	Input Unit Type for X	-	Temperature
Curve:Biquadratic	Input Unit Type for U	-	Temperature
Curve:Biquadratic	Output Unit Type	-	Dimensionless

However, in gbXML, the whole equation is written as one expression. Though the independent and dependent variables are defined, it is still difficult to differentiate the curve class and to extract the coefficients of each independent variable using current schema elements. Extra work is needed. There is another method called “PointData”, using multiple “data” elements to represent points on a performance curve in order. This method is neither accurate nor efficient. Two gbXML representation schema is shown in Figure 2.

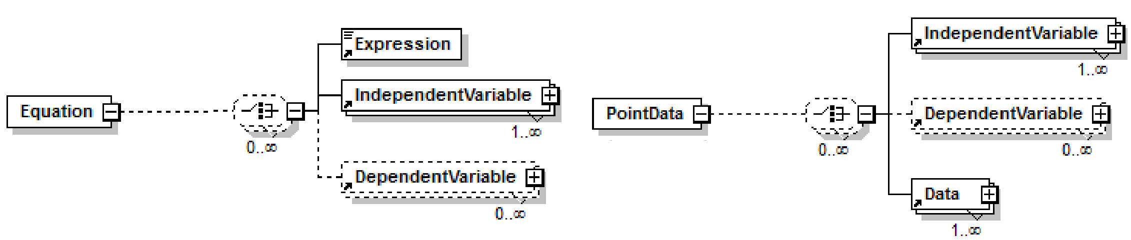

Figure 2. gbXML representation of a performance curve. Users can define points on the curve, the curve equation expression, independent variables, and dependent variables. But neither method is accurate or efficient.

Issue three: complex data mapping rules

In IDF, HVAC systems data are categorized into different classes. But in the gbXML schema, data is not independent of each other. Most elements have complicated reference relationships, and the gbXML documentation doesn't explain this well. For example, a zone is made up of one space or multiple spaces and is served by one HVAC system (McDowall, R. 2007). In the IDF, the "ZoneHVAC: AirDistributionUnit" class is named by zones. But in gbXML schema, HVAC systems and equipment are identified by "id" attributes. A primary system connects with a zone, the "AirLoop" and "HydronicLoop" elements have a "controlZoneIdRef" attribute. At the same time, the "Zone" element also has "AirLoopId" and "HydronicLoopId" children elements. However, for HVAC equipment, the "AirLoopEquipement" and "HydronicLoopEquipment" elements don't have a zone or space reference. Instead, "Space" elements in gbXML are connected with HVAC equipment. They have "AirLoopEquipementId" and "HydronicLoopEquipmentId" children elements. The complex relationships are shown in Figure 3. They would cause missed connections between zones or spaces with HVAC equipment, raising interoperability issues.

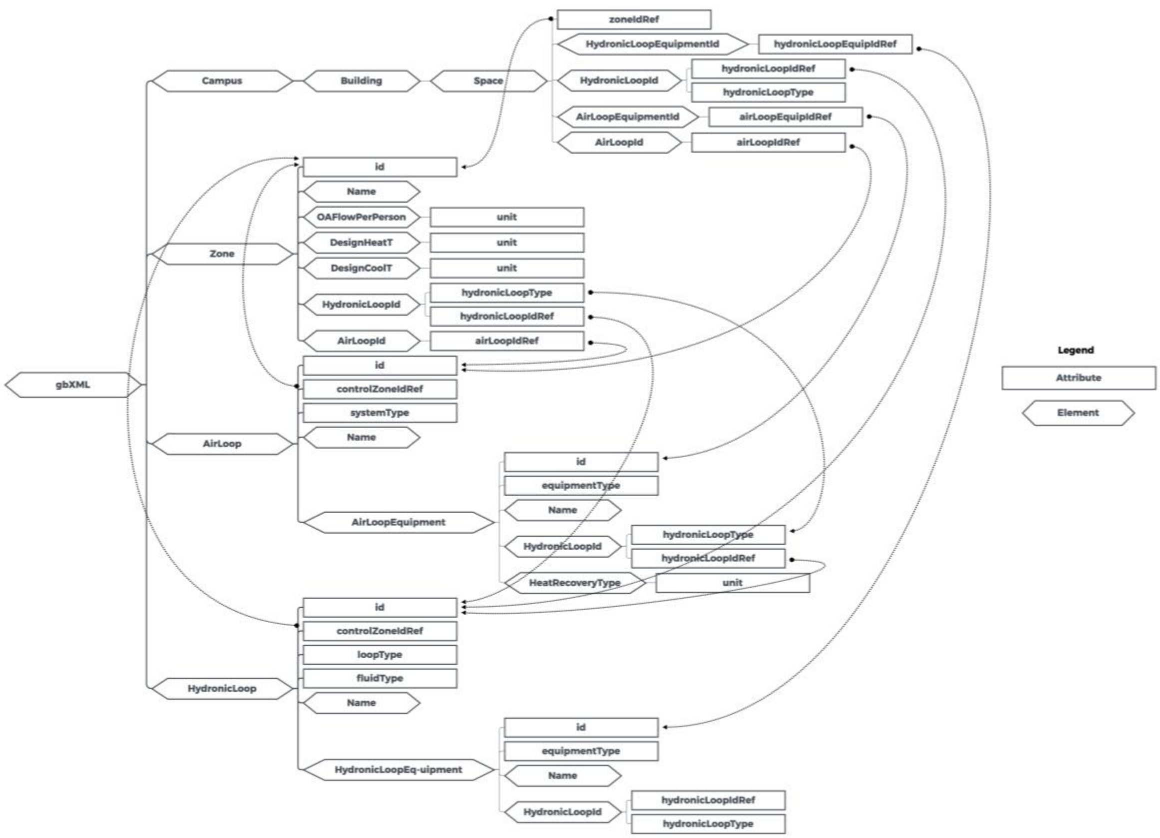

Figure 3. Complex gbXML data mapping rules of HVAC systems

REDEFINING DATA MAPPING RULES

Compared with missing components, decoding performance curves, the problem of complex data mapping rules is the most critical issue and needs to be redefined. The rule of thumb for defining HVAC systems in gbXML schema is bottom-up. First of all, defining "HydronicLoopEquipment", "AirLoopEquipment", "AirLoop" and "HydronicLoop" in order. Secondly, connecting spaces and HVAC equipment by "HydronicLoopEquipmentId" and "AirLoopEquipmentId" in "Space" element; connecting zones and primary system by "HydronicLoopid" and "AirLoopId" in "Zone" element. Additionally, connecting spaces and zones by "zoneIdRef" in the "Space" element. In this way, zones are connected with their spaces' equipment indirectly. For example, there are three spaces in a zone, but only one of them has a VAV box. Then this zone is connected with the VAV box. Figure 4 shows redefined data mapping rules, which are more effective and efficient than the relationships shown in Figure 3.

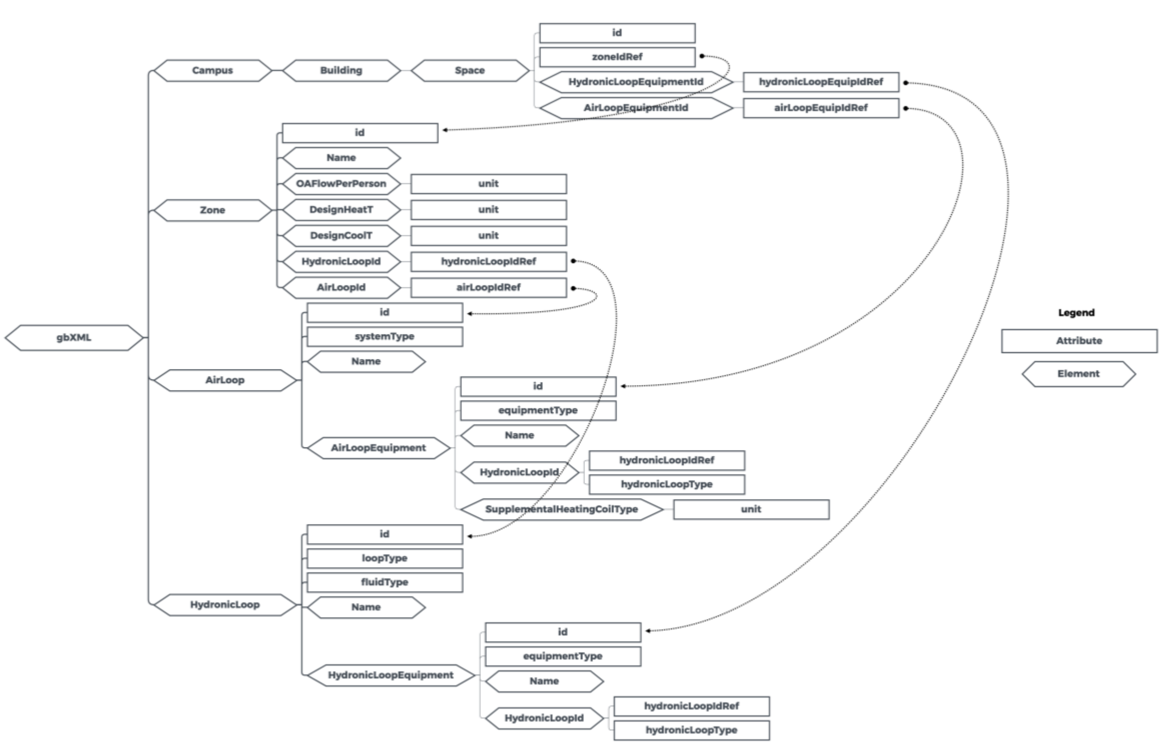

Figure 4. Redefined gbXML data mapping rules of HVAC systems. “AirLoop” and “HydronicLoop” are primary HVAC systems data. “AirLoopEquipment” and “HydronicLoopEquipment” are HVAC equipment data. “Space” and “Zone” are connected to the HVAC equipment and primary system respectively, through id reference.

VALIDATION

gbXML file preparation

Using the redefined data mapping rules of HVAC systems, the ASHRAE baseline system seven can be represented as a gbXML file. The corresponding building model was created in Revit 2020.1. Zones were assigned by the core-perimeter method. Geometry data and material properties were exported as gbXML file format, using energy settings. Then, HVAC systems data is added to this gbXML file through the redefined data mapping rules, using current gbXML schema. Figure 5 shows a part of the gbXML file, which represents the boiler information.

```xml
<!-- Whole Building, hot water loop -->
<HydronicLoop id="aim1111" loopType="HotWater" fluidType="Water">
  <Name>HW1</Name>
  <HydronicLoopEquipment id="aim1112" equipmentType="Boiler">
    <Name>HW1-Boiler1</Name>
    <Temp tempType="HighTempLockout">100</Temp>
    <Temp tempType="HeatDesign">82</Temp>
    <Temp tempType="Range">28</Temp>
    <Power unit="Watt" powerType="NaturalGas" useType="Heating" />
    <Efficiency efficiencyType="ThermalEff">0.8</Efficiency>
    <Control controlType="Boiler" minPowerRatio=0.0 />
  </HydronicLoopEquipment>
</HydronicLoop>
```
Figure 5. Boiler information added in gbXML

OpenStudio manipulation

The gbXML file can be imported into OpenStudio by applying customized measures. It becomes an OSM file. Figure 6 shows an example Ruby code that converts the boiler data in gbXML to OpenStudio. In this way, the HVAC system was successfully imported into OpenStuido, but weather files, operation schedules, and internal loads still need to be assigned manually. Finally, Openstuido uses a built-in ForwardTranslator to translate the OSM model to the IDF model for simulation using EnergyPlus.

```
def add_boiler
  boiler = OpenStudio::Model::BoilerHotWater.new(self.model)
  boiler.setName("#{self.name} Boiler")
  boiler.setEfficiencyCurveTemperatureEvaluationVariable('LeavingBoiler')
  boiler.setFuelType(self.powerType)
  boiler.setDesignWaterOutletTemperature(self.design_loop_exit_temp)
  boiler.setNominalThermalEfficiency(self.thermal_efficiency)
  boiler.setMinimumPartLoadRatio(0)
  boiler.setMaximumPartLoadRatio(1.1)
  boiler.setOptimumPartLoadRatio(1)
  boiler.setWaterOutletUpperTemperatureLimit(self.upper_temp_limit)
  boiler.setBoilerFlowMode('LeavingSetpointModulated')
  boiler
end
```

Figure 6. Rudy code for converting boiler data in gbXML

Simulation results

The energy simulation result of the redefined gbXML file is supposed to be the same as the energy simulation result of the ASHARE system seven IDF model. Due to time limitations, the customized measures didn't cover all HVAC equipment. However, the successful import of boiler data and primary HVAC systems is enough to prove the effectiveness of the redefined data mapping rules and the capability of the gbXML schema. In addition to the gbXML data imported, detailed modification was conducted in OpenStudio, taking the parameters in the original IDF model as a reference. Finally, the simulated Energy Usage Intensity (EUI) achieves 80% accuracy of the original file.

CONCLUSION

The current gbXML schema (Version 6.01) is rarely used for HVAC systems representation due to complex data mapping rules. It has been one of the most critical obstacles to gbXML interoperability improvement. Through detailed data mapping of the ASHRAE standard system type seven from the IDF model (Version 9.0) to current gbXML schema, gbXML should be able to define HVAC systems though three interoperability issues were figured out. The complex data mapping rules have been redefined, using four elements: “Space”, “Zone”, “AirLoop” and “HydronicLoop”. The “AirLoop” and “HydronicLoop” elements are used for primary HVAC system data. Their children elements “AirLoopEquipment” and “HydronicLoopEquipment” are used for HVAC equipment data. “Space” can be connected to the HVAC equipment by its children elements “HydronicLoopEquipmentId” and “AirLoopEquipmentId”. “Zone” can be connected to the HVAC primary system by its children elements “HydronicLoopId” and “AirLoopId”. In terms of the other two interoperability problems: missing components and the difficulty of decoding performance curves. They could be handled as an updated schema in the future.

ACKNOWLEDGEMENT

This study was funded through the research project 1810 of the American Society of Heating, Refrigerating, and Air-Conditioning Engineers (ASHRAE). The authors acknowledge the support of employees at BuildSimHub. Inc. and faculties at Carnegie Mellon University for this research.

REFERENCES

DOE, U. (2011). 2011 Building energy data book. *US Department of Energy*.

Dong, B., Lam, K. P., Huang, Y. C., & Dobbs, G. M. 2007. A comparative study of the IFC and gbXML informational infrastructures for data exchange in computational design support environments. In *Tenth International IBPSA Conference* (pp. 1530-1537).

Eastman, C., Teicholz, P., Sacks, R., & Liston, K. 2011. *BIM handbook: A guide to building information modeling for owners, managers, designers, engineers and contractors*. Chapter. John Wiley & Sons.

gbXML Current Schema 2017. http://www.gbxml.org/Schema_Current_GreenBuildingXML_gbXML

McDowall, R. 2007. *Fundamentals of HVAC systems*, SI edition. Chapter 5, Zones. Academic Press.

Seyam, Shaimaa. 2018. Types of HVAC Systems. In *HVAC System*. IntechOpen.

McGraw Hill. 2009. The business value of BIM: Getting building information modeling to the bottom line. *Smart Market Report* (2009): 1-50.

Moon, H. J., Choi, M. S., Kim, S. K., & Ryu, S. H. 2011. Case studies for the evaluation of interoperability between a BIM based architectural model and building performance analysis programs. In *Proceedings of 12th conference of international building performance simulation association* (Vol. 2011).

W3Schools Online Web Tutorials. 2017. https://www.w3schools.com/

Xu, W., Chong, A. & Lam, K. P. 2019. A new BIM to BEM framework: the development and verification of an open-Source gbXML to EnergyPlus translator for supporting building life cycle performance analysis. In *building simulation 2019 conference*.

Zhang, Z., Chong, A., Pan, Y., Zhang, C., Lu, S., & Lam, K. P. (2018, September). A deep reinforcement learning approach to using whole building energy model for hvac optimal control. In *2018 Building Performance Analysis Conference and SimBuild*.