ONTOLOGY- AND FREEWARE-BASED PLATFORM FOR RAPID DEVELOPMENT OF BIM APPLICATIONS WITH REASONING SUPPORT

Zhiliang Ma[1], and Zhe Liu[2]

Abstract: In the Architecture, Engineering and Construction (AEC) area, a remarkable tendency is to use the Building Information Modeling (BIM) data to carry out analysis and calculation based on rules specified in regulations or standards so that the BIM applications with reasoning support (BIM-R applications) are necessary. The current way to develop BIM-R applications separately and represent rules by coding or in proprietary formats has the problems of both cost and efficiency. To solve the problems, a new way is proposed, i.e. to use a platform for rapid development of BIM-R applications (BIM-R platform) implemented based on ontology and freeware components. The research establishes 1) the functional requirements of the BIM-R platform, 2) the mechanism to transform BIM data into ontology data, and 3) the architecture of the BIM-R platform, and selects freeware components to be used. A platform is implemented accordingly and applied to develop a prototype BIM-R application for as-bid cost estimation of buildings for verification. The proposed platform can help reduce cost and improve efficiency for the development of BIM-R applications and can be used by both researchers and developers.

Keywords: BIM (Building Information Modeling), reasoning, rapid development platform, ontology, freeware.

1 INTRODUCTION

In the past decade, Building Information Modeling (BIM) has begun to be widely applied in the AEC (Architecture, Engineering and Construction) area and a remarkable tendency is to use the BIM data to carry out various analysis and calculation, such as design check, cost estimation, etc. Since such analysis and calculation always need to be based on rules specified in regulations or standards, the BIM applications with reasoning support (BIM-R applications) to execute rules are necessary. Up to now, a number of BIM-R applications already exist commercially, e.g., Navisworks, Vico quantities, etc. However, most of the existing BIM-R applications are developed separately although they share many similarities on development and related rules are represented by coding or in proprietary formats in them so that it is difficult for the users to maintain, update and reuse the rules. It means that the current way to develop BIM-R applications has the problems of both cost and efficiency. On one hand, if there is a rapid development platform which provides common components to facilitate the development of BIM-R applications, the speed and quality of development can be greatly improved and the cost can be reduced. On the other hand, ontology is one of the semantic web technologies to represent, exchange and reuse

---

[1] Professor, Department of Civil Engineering, Tsinghua University, Beijing, China, mazl@tsinghua.edu.cn.

[2] PhD Candidate, Department of Civil Engineering, Tsinghua University, Beijing, China, zhe.liu.thu@gmail.com.

domain concepts, relations between concepts, and rules, and has started to be utilized in the AEC area for knowledge representation, information interoperation and rule-based reasoning (Zhou et al 2016). Hence, it is expected that a platform for rapid development of BIM-R applications (BIM-R platform), in which ontology is used for the representation of rules for easy manipulation and reuse, is promising. In addition, if such a BIM-R platform can make use of the freeware components that are available nowadays, it can further reduce the cost of development.

At present, Solibri Model Checker (Solibri) and FORNAX are the only two available BIM-R platforms. The courthouse design circulation check system for GSA (Lee 2010) was implemented based on Solibri Model Checker, while the e-PlanCheck system in CORENET project (Khemlani 2005) was implemented based on FORNAX. In both cases, instead of ontology, proprietary formats were used to represent the rules.

Although researches on ontology-based BIM-R platforms have not been reported, this is not the case for ontology-based BIM-R applications for specific domain problems. Hyunjoo and Francois (2009) proposed a method to analyze possible conflicts of design BIM data by transforming the data into ontology data and checking the consistency of ontology with the support of existing reasoning machine. The prototype tool in the research showed the basic processes of BIM-R applications, but the data transformation was achieved by coding, which was hard to reuse. Pauwels et al (2011) proposed a method to analyze the sound performance of buildings by translating the design BIM data into ontology data and carrying out reasoning according to ontology-based rules. The data translation in the research was generalized, but it had no post-process for the reasoning results represented in text with building component IDs. Considering there are hundreds of building components with complex spatial relations in real projects, the post-process for reasoning results, such as visualization, statistic, etc., is necessary. Zhang el at (2013) developed a BIM-R application based on a commercial BIM modeling tool, Tekla Structures, to analyze design BIM data for construction safety and suggest proper safety measures in a visualized way. But the application was limited to the projects where Tekla Structures is used. Lee et al (2014) developed a BIM-R application to infer cost items for building components consistently by reasoning, but the rules were limited to tiling work, which hampered the application from practical use.

In order to improve efficiency and reduce cost for the development of BIM-R applications, a new way is proposed, i.e., to use a BIM-R platform implemented based on ontology and freeware components. This paper presents the development of such a platform. Firstly, by analyzing existing BIM-R applications, the functional requirements of the BIM-R platform are summarized and a mechanism to transform BIM data into ontology data is designed. Secondly, based on the result, freeware components are surveyed and selected not only for the import, calculation and visualization of BIM data, but also for the reasoning and query of ontology data, then the architecture of the BIM-R platform is established. Finally, an ontology- and freeware-based BIM-R platform is implemented accordingly, and it is used to develop a prototype BIM-R application for as-bid cost estimation of buildings to verify the applicability of the platform.

2 FUNCTIONAL REQUIREMENTS OF THE BIM-R PLATFORM

As a premise, the rules on the BIM-R platform are assumed to be represented in terms of a domain ontology. Besides, for better extensibility and reusability, the BIM data are assumed to be stored in files conforming to the Industry Foundation Classes (IFC) standards (IFC files). Obviously the BIM-R platform should provide the common functions

that the BIM-R applications share. Hence, by analyzing the existing ontology-based BIM-R applications for different domain problems (Hyunjoo and Francois 2009; Pauwels et al 2011; Zhang et al 2013; Lee et al 2014), five functional requirements are summarized as follows:

1. Import of domain ontology. The platform should read, parse and write domain ontology.
2. Loading of BIM data. The platform should read, parse and write BIM data with the support of mainstream format of IFC files.
3. Transformation from BIM data into ontology data. The platform should transform both geometric and non-geometric BIM data into ontology data and filter unnecessary BIM data.
4. Reasoning and query of ontology data. The platform should support expressive rules and queries with acceptable speed of execution.
5. Visualization of query result. The platform should support basic 3D interactions including Highlighting and hiding building components.

It should be noted that, creation and modification of the domain ontology and the BIM data can be done in existing third-party tools and is unnecessary to be included in the BIM-R platform for simplicity.

3 MECHANISM TO TRANSFORM BIM DATA INTO ONTOLOGY DATA BASED ON IFCOWL

When ontology is used to represent rules, to answer a certain domain question, implicit data will be inferred from explicit data in the domain ontology by reasoning according to ontology-based rules, where both the explicit data and the implicit data are represented as instances of concepts and relations between instances. The purpose to transform BIM data into ontology data is to obtain the explicit data in order to prepare for the reasoning, as shown in Figure 1**Error! Reference source not found.**.

When BIM data are transformed into explicit ontology data, since not all BIM data are required for reasoning, the unnecessary data should be filtered out (e.g. data A) or changed into another format to represent the same information (e.g. data B). It should be noted that, not all implicit data need to be inferred by reasoning. Instead, some data, such as the volume for complex building components, collision between building components, etc., should be calculated by coding for the transformation (e.g. data C).

The method to translate BIM data into ontology data is affected by format and structure of both the BIM data and the ontology data. As stated in Section 2, IFC format is selected for inputting BIM data. Concerning ontology, Resource Description Framework (RDF) is a well-adopted framework that provides necessary data model, syntax and basic vocabulary for semantic web, and Web Ontology Language (OWL) is a recommended language that follows the RDF syntax and extends the data model and vocabulary to represent concepts, relations and instances of concepts in ontology. The ontology represented in OWL (OWL ontology) is compatible for RDF syntax and can be stored in various RDF formats, such as Turtle (TTL). Thus, OWL is selected as the language used in the BIM-R platform to represent ontology data. The selection of the languages for represent rules and queries will be discussed in the next section.
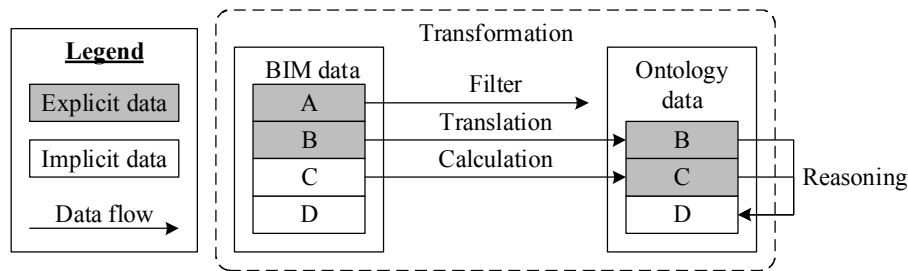
Figure 1: Transformation from BIM data to ontology data

One of the principles for building domain ontology is to reuse existing basic ontologies. Although there is only a few basic ontologies in the AEC area, IfcOWL (Beetz et al 2009) is the only one that is recommended by buildingSMART International Ltd. It has the same structure with IFC, which makes it possible to translate between them directly by changing syntax. For simplicity, it is also assumed that the domain ontology on the BIM-R platform is built on IfcOWL so that an existing open source freeware component, IFC-to-RDF-converter (Pauswels and Oraskari), to translate BIM data stored as IFC file into ontology data stored as RDF file (Beetz et al 2009) can be used. Thus, the mechanism of transformation based on IfcOWL consists of three steps as follows.

1. Calculate necessary implicit data according to the requirements of rules in the domain ontology. The results are written back into the IFC files.
2. Filter unnecessary BIM data and export the rest as a temporary IFC file.
3. Translate the temporary IFC file into an OWL file in TTL format by invoking the IFC-to-RDF-converter. The generated OWL file contains the required data for reasoning.

4   COMPONENTS AND ARCHITECTURE OF THE BIM-R PLATFORM

Based on the functional requirements and the mechanism to transform BIM data into ontology data, freeware components are selected for the BIM-R platform.

4.1   Components for import, calculation and visualization of BIM data

In order to implement the functional requirements related to BIM data stated in Section 2, the functions to calculate BIM data, especially the geometric data, are required according to the mechanism of transformation stated in Section 3.

A survey on existing freeware components based on IFC (IFC components) was carried out and the results showed that only a few IFC components support the calculation of geometric data. Among them, IFC Engine DLL (RDF Ltd.), IfcOpenShell (IfcOpenShell Team) and xBIM Toolkit (xBIM Team) are still active. Their functions are compared as shown in Table 1. All of them support data import from IFC files. The function to calculate geometric data in the IFC Engine DLL is limited. Especially, complex geometric operations such Boolean operation, profiling, etc., are unavailable in the free version of IFC Engine DLL. Both IfcOpenShell and xBIM Toolkit support complex geometric operations. IfcOpenShell is designed as an engine to process BIM data, and it can extract and convert geometric data into many other formats, which can be imported in mainstream general 3D viewers. But the development based on IfcOpenShell is relatively difficult with much configuration and setup. By contrast, xBIM Toolkit is a platform with little configuration and setup and it provides optimized 3D BIM viewers for visualization and interaction on windows and web browsers, and the powerful InsertCopy function to filter BIM data,

which can separate geometric data and non-geometric data within lines of coding. Thus, for efficiency, xBIM Toolkit is selected for the import, calculation and visualization of BIM data in the BIM-R platform.

Table 1: Functions of free IFC components (+: supported; *: partially supported)

No.	Name	Data import	Visualization	Calculation of geometric data
1	IFC Engine DLL	+	*	*
2	IfcOpenShell	+	*	+
3	xBIM Toolkit	+	+	+

4.2 Components for reasoning and query of ontology data

The selection of components for reasoning and query of ontology data is affected by the language to represent rules (rule language) and the language to represent queries (query language) of reasoning results in the domain ontology. As stated in Section 3, OWL is selected to represent ontology data. Hence, the compatibility of OWL is important for the selection of the rule language and the query language. Semantic Web Rule Language (SWRL) and Notation3 (N3) are two mainstream OWL-compatible rule languages used in the existing BIM-R applications. SWRL natively supports the vocabulary of OWL, while N3 requires extra rules to do so. Additionally, the rules represented in SWRL can be added in the OWL ontology and stored as RDF files. For simplicity, SWRL is selected as the rule language used in the BIM-R platform.

To query the reasoning results, entailment and multi-variable query are necessary. Entailment can simplify the query intuitively. For example, with the support of entailment, the query "all doors" not only gets the building components directly tagged as "door" but also the ones tagged as other sub-type of door, such as "wood door" etc. Multi-variable query is useful to query instances with specific relations. For instance, it is difficult to express the meaning of query "wall X that intersects with wall Y" with only one variable.

Through literature review, SPARQL-DL (Sirin and Parsia 2007), and SQWRL (O'Connor and Das 2009) are two OWL-compatible query languages which supports both entailment and multi-variable query. The availability of the corresponding freeware components is considered in the selection between SPARQL-DL and SQWRL. The available freeware components that support reasoning in SWRL include Pellet (Complexible Inc.), HermiT (Information Systems Group), Jess (Friedman-Hill), and Drools (Red Hat, Inc.). Among them, Jess and Drools are general rule engines which can be customized to support SWRL but both of them are significantly slower than Pellet and HermiT in reasoning in SWRL according to the authors' performance tests. The available freeware components that support query in SPARQL-DL include SPARQL-DL API (derivo GmbH) and Pellet, while only one freeware component that supports query in SQWRL exists, i.e. SWRLAPI (SWRLAPI Team) implemented based on Drools. For large scale of data set, query of stardog, an enterprise semantic platform which embedded with Pellet, was fast in the performance benchmark (Pauwels et al 2016). Since Pellet supports both SWRL for reasoning and SPARQL-DL for query, for simplicity, Pellet is better to be selected as the reasoner and query engine in the BIM-R platform. Consequently, SPARQL-DL is selected as the query language.

4.3   Architecture of the BIM-R platform

Based on the selected freeware components, a browser-server architecture of the BIM-R platform is established, as shown in Figure 2. The grey rectangles represent customized components which require extra coding, i.e. "3D BIM viewer", "Parser", "IFC calculator", "IFC filter" and "Reasoning and query wrapper", while the white rectangles represent selected freeware components, i.e. the xBIM Toolkit, the IFC-to-RDF converter, Pellet, and Jena API. It deserves to explain that Jena API (Apache Software Foundation) is used for Pellet to access ontology data for query in SPARQL-DL.
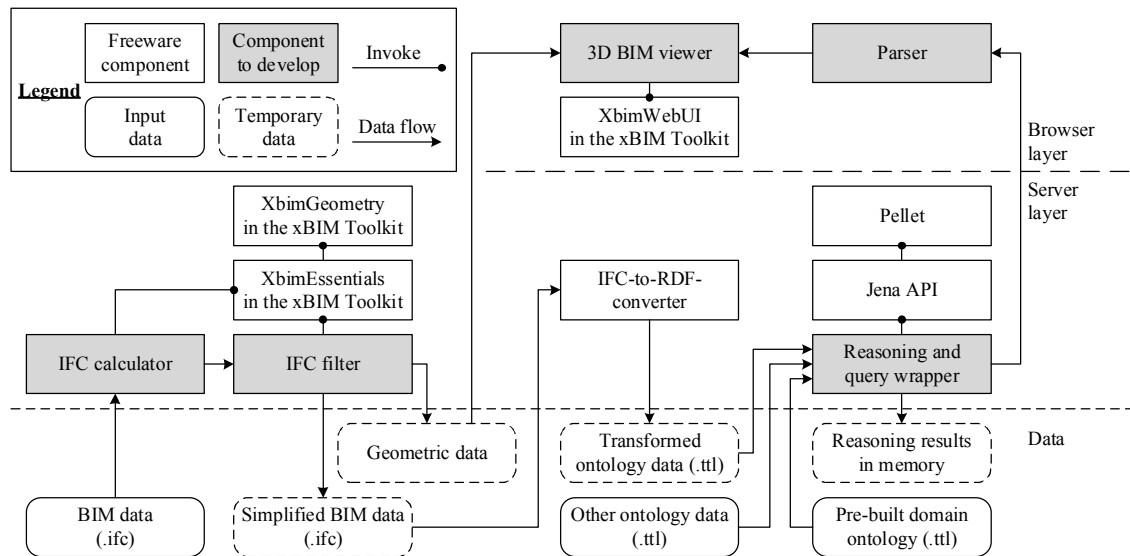

Figure 2: Architecture of the BIM-R platform

In the server layer, the components "IFC calculator", "IFC filter" and the IFC-to-RDF converter are developed to implement the three steps to transform BIM data into ontology data separately, as stated in Section 3. The XbimEssentials and XbimGeometry modules contained in the component xBIM Toolkit are invoked in the transformation. The transformed ontology data, i.e., instances of concepts and relations between instances, as well as the pre-built domain ontology, i.e. concepts and relations between concepts are imported in the component "Reasoning and query wrapper" for reasoning in SWRL and query in SPAQRL-DL by invoking the component Pellet using the Jena API. In some cases, ontology data from sources other than BIM is also required and imported, e.g., recent material prices for cost estimation.

In the browser layer, the component "Parser" sends SPARQL-DL queries to the component "Reasoning and query wrapper" and parses the query results as visualization commands. The component "3D BIM viewer" invokes the XbimWebUI module in the component xBIM Toolkit to execute the visualization commands generated by the "Parser" to highlight or hide building components and display extra necessary information for the query results.

5   IMPLEMENTATION AND VERIFICATION OF THE BIM-R PLATFORM

An ontology- and freeware-based BIM-R platform was implemented according to the architecture stated in Section 4. Then it was used to develop a prototype BIM-R application for as-bid cost estimation of buildings. With reasoning support, about 90% of building

components in four concrete framework buildings were classified into correct bill of quantities in the application, which proved the usability of both the application and the platform. With the common functions provided by the BIM-R platform, the development of the application was simplified and researchers could focus on the establishment of ontology and rules for as-bid cost estimation. Thus, the efficiency for the development was improved by using the platform. Limited to the space of the paper, detailed information cannot be presented.

6   CONCLUSION

The current way to develop BIM-R applications separately and represent rules by coding or in proprietary formats has the problems of both cost and efficiency. To solve the problems, a new way was proposed, i.e. to use an ontology- and freeware-based BIM-R platform. According to the analysis of existing BIM-R applications, the functional requirements of the BIM-R platform were summarized and a mechanism to transform BIM data into ontology data was designed as three steps, i.e., calculation, filter and translation. OWL, SWRL and SPARQL-DL were selected as languages used to represent ontology data, rules and queries separately while the freeware components, IFC-to-RDF converter, xBIM Toolkit, Pellet and Jena API were selected to implement the functions of the platform. Accordingly, a browser-server architecture consisting of the selected freeware components as well as customized components was designed. The platform was implemented and used to develop a prototype BIM-R application for as-bid cost estimation of buildings for verification. As a conclusion, the platform can be used by both researchers and developers to reduce cost and improve efficiency for the development of BIM-R applications.

7   ACKNOWLEDGEMENT

8   REFERENCES

Apache Software Foundation. *Apache Jena*. Available at: https://jena.apache.org/ [Accessed 1 Feb. 2017].

Beetz, J., Van Leeuwen, J., and De Vries, B. (2009). *IfcOWL: A Case of Transforming EXPRESS Schemas into Ontologies*. Artif. Intell. Eng. Des. Anal. Manuf., 23, pp. 89-101, doi: 10.1017/S0890060409000122.

Complexible Inc. *Pellet*. Available at: https://github.com/stardog-union/pellet [Accessed 1 Feb 2017].

derivo GmbH. *SPARQL-DL API*. Available at: http://www.derivo.de/en/resources/sparql-dl-api.html [Accessed 1 Feb 2017].

Friedman-Hill, E. *Jess, the Rule Engine for the Java Platform*. Available at: http://herzberg.ca.sandia.gov/ [Accessed 1 Feb 2017].

Hyunjoo, K., and Francois, G. (2009). Design Coordination in Building Information Modeling (BIM) Using Ontological Consistency Checking. In: Proc. International Workshop on Computing in Civil Engineering, June 24-27, Austin, Texas, USA, doi: 10.1016/j.autcon.2009.07.002.

IfcOpenShell Team. *IfcOpenShell*. Available at: http://ifcopenshell.org/ [Accessed 1 Feb. 2017].

Information Systems Group. *HermiT Reasoner*. Available at: http://www.hermit-reasoner.com/ [Accessed 1 Feb. 2017].

Khemlani, L. (2005). *CORENET e-PlanCheck: Singapore's Automated Code Checking System*. Available at: http://www.novacitynets.com/pdf/aecbytes_20052610.pdf [Accessed 1 Feb. 2017].

Lee, J.M. (2010). *Automated Checking of Building Requirements on Circulation over a Range of Design Phases*. PhD Diss., Georgia Institute of Technology, Atlanta, USA.

Lee, S.K., Kim, K.R., and Yu, J.H. (2014). *BIM and Ontology-based Approach for Building Cost Estimation*. Automation in Construction, 41, pp. 96-105, doi: 10.1016/j.autcon.2013.10.020.

O'Connor, M., Das, A., (2009). *SQWRL: A Query Language for OWL*. In: Proc. 6th International Conference on OWL: Experiences and Directions, 529, pp. 208–215.

Pauwels, P., and Oraskari, J. *IFC-to-RDF-converter*. Available at: https://github.com/mmlab/IFC-to-RDF-converter [Accessed 1 Feb 2017].

Pauwels, P., Van Deursen, D., Verstraeten, R., De Roo, J., De Meyer, R., Van de Walle, R., and Van Campenhout, J. (2011). *A Semantic Rule Checking Environment for Building Performance Checking*. Automation in Construction, 20, pp. 506–518, doi:10.1016/j.autcon.2010.11.017.

Pauwels, P., de Farias, T. M., Zhang, C., Roxin, A., Beetz, J., & De Roo, J. (2016). Querying and reasoning over large scale building data sets: an outline of a performance benchmark. In Proceedings of the International Workshop on Semantic Big Data. ACM. doi:10.1145/2928294.2928303.

Red Hat, Inc. *Drools - Business Rules Management System*. Available at: https://www.drools.org/ [Accessed 1 Feb. 2017].

RDF Ltd. *IFC Engine DLL*. Available at: http://rdf.bg/ifc-engine-dll.html [Accessed 1 Feb. 2017].

Sirin, E., and Parsia, B. (2007). *SPARQL-DL: SPARQL Query for OWL-DL*. In: Proc. the OWLED 2007 Workshop on OWL: Experiences and Directions, Innsbruck, Austria.

Solibri Inc. *Solibri Model Checker*. Available at: https://www.solibri.com/products/solibri-model-checker/ [Accessed 1 Feb. 2017].

SWRLAPI Team. *SWRLAPI*. Available at https://github.com/protegeproject/swrlapi [Accessed 1 Feb 2017].

xBIM Team. *xBIM Toolkit*. Available at http://docs.xbim.net/ [Accessed 1 Feb 2017].

Trimble Solutions Corporation. *Tekla Structures*. Available at: https://www.tekla.com/us/products/tekla-structures [Accessed 1 Feb. 2017].

Vico Software, Inc. *Vico Quantities*. Available at: http://www.vicosoftware.com/resources/vico-quantities [Accessed 1 Feb. 2017].

Zhang, S., Teizer, J., Lee, J.-K., Eastman, C.M., and Venugopal, M. (2013). *Building Information Modeling (BIM) and Safety: Automatic Safety Checking of Construction Models and Schedules*. Automation in Construction, 29, pp. 183–195, doi:10.1016/j.autcon.2012.05.006.

Zhou, Z., Goh, Y.M., and Shen, L. (2016). *Overview and Analysis of Ontology Studies Supporting Development of the Construction Industry*. J. Comput. Civ. Eng., 30(6), doi: 10.1061/(ASCE)CP.1943-5487.0000594.