BIM SERVER REQUIREMENTS TO SUPPORT THE ENERGY EFFICIENT BUILDING LIFECYCLE

Y. Jiang[1], J. Ming[1], D. Wu[1], J. Yen[1], P. Mitra[1], J. I. Messner[2], and R. Leicht[2]

[1] College of Information Sciences and Technology, The Pennsylvania State University, University Park, PA 16802. Email: {yzj107, jum310, dwu, jyen, pmitra}@ist.psu.edu

[2] Department of Architectural Engineering, The Pennsylvania State University, University Park, PA 16802. Email: {jmessner, rmleicht}@engr.psu.edu

ABSTRACT

Energy efficient building design, construction, and operations require the development and sharing of building information among different individuals, organizations, and computer applications. Building Information Modeling (BIM) servers are tools used to enable an effective exchange of data. This paper describes an investigation into the core BIM server requirements needed to effectively support information sharing related to energy efficient retrofit projects. The requirements have been developed through an analysis of existing functional capabilities combined with a case study analysis. The set of requirements identified includes fine-grained queries such as selective model queries, information queries (e.g. weather, building description), and operational information queries (by building parts, proximity, and context). A set of RESTful programming interfaces for building tools to access and exchange data, including security and data privacy issues, is being explored to provide a server-centric building information model exchange and interoperability to facilitate energy efficient retrofit.

INTRODUCTION

The topic of facilitating data exchange between different design and simulation tools in the Architecture, Engineering, and Construction industry (AEC industry) is gaining increasing interest. One notable fact is that AEC has its own tradition of ways to adopt the assistance from computer and information science in the whole building design life cycle (Watson, 2010). Instead of taking advantage of a server-centric solution, the workstation-based CAD approach, which dates back to 1980s, is still widely used by the AEC industry (Watson, 2010). In the current AEC

work flows, the simulation processes are not fully integrated into the design and modeling process (Hirsch et al., 2011). In most cases, engineers need to manually build models for simulation, which duplicates work that has already been performed. Approximately 80% of the effort needed to run a simulation is spent in building models (Bazjanac, 2001). With Building Information Modeling (BIM), attributes and data can be attached to a model, which potentially allows integrated analysis and simulation, especially for energy efficient retrofit projects (Beetz et al., 2010, Beetz et al., 2011). However, most of the analysis and simulation software have been developed by experts in different domains, and their formats are different in nature, which limits model reuse and data sharing among tools. To solve this problem, the requirements for the Building Information Modeling server are explored based on the existing open source platforms (e.g., BIMserver.org).

In this paper, "BIM Server" is a general concept that describes the server used for sharing building design data, and "BIMserver.org" refers to the specific platform implemented by bimserver.org. We investigate a set of core requirements for energy efficient building lifecycles and identify the gap between the current BIMserver.org implementation and the requirements. We give an outlook on the future work needed to fill in the gap. First, we introduce previous efforts on combining BIM with energy simulation tools leveraging the open source BIMserver.org. Then we present the details of generic requirements for building information exchanges in the Section "Requirements", and discuss the current status of BIMserver.org and future work in the Section "Discussion."

RELATED WORK

BIM based energy simulation. Currently BIM technologies have been widely adopted in the AEC industry for 3D-rendering, drawing extraction, estimation of cost, and clash detection (Eastman et al. 2008). Several previous efforts have tried to integrate building information models with energy simulation tools. McGraw-Hill (2009) surveyed that the lack of software interoperability and functionality is rated as one of the greatest obstacles to improving business value of BIM. Moon, et al. (2011) conducted case studies to evaluate the interoperability of a BIM based architecture model and performance simulation programs (e.g., EnergyPlus). In this paper, we identify a set of key requirements, among which some are new, that are needed for BIM-based energy efficient building lifecycle.

Previous work using BIMserver.org. The open source BIMserver.org (version 1.1 beta, 2012) enables users to host their own BIM server to centralize the information of any construction or other building related projects. Weiming et al. (2010) showed

an agent-based, serviced-oriented approach to integrating data, information, and knowledge captured and accumulated during the entire facility lifecycle with BIMserver.org. Recently, Singh et al. (2011) developed a theoretical framework of technical requirements for using BIMserver.org as a multi-disciplinary collaboration platform.

REQUIREMENTS

Internal data storage requirements. Internal data handling can be addressed using file-based and model-based approaches. These two approaches have different strengths and shortcomings. A file server's structure could be quite clear and simple, which is easy to implement. The underlying database could be a classical relational database. Adopting such a file server will not impact current energy-efficient retrofit workflows. The file server does not necessarily need a unified file format or a format translator. It stores files and facilitates the collaboration of sections of design and simulation teams that share the same file format. The basic unit of file server is a file, while sometimes users have to query data distributed in different files. For example, a simulation tool may need the height and width of all windows. However, a file server may not able to parse the information it stores semantically and cannot support such data export or other advanced functions such as model merging and partial model query (Beetz et al., 2010).

A model-driven server could compensate for the file server's drawbacks. A model-driven server doesn't save the files submitted by users into database directly. It parses the data in the file first and uses these data to construct its own pre-defined data structure and maintain a comprehensive model. The advantage of this method is that some advanced functionalities (e.g. clash detection, model merging) could be performed on the model easily (BIMserver 1.1 beta, 2012). Currently, BIMserver.org is a model-driven server which adopts Eclipse Model Framework (EMF, 2012) to represent the data model that is parsed from IFC files.

Web service interface. There are three kinds of popular web service interfaces: remote procedure call (RPC), service-oriented architecture (SOA), and representation state transfer (REST) interface (Fielding, 2000). RPC allows two distributed remote heterogeneous systems to call the functions or methods of each other. But their interaction and function calls deeply relate to the specific implementation of specific programming languages, which violates the principle of loose coupling in software engineering. SOA is widely adopted by many web service providers. Instead of relying on the specific implementation, SOA is driven by messages (events). The third option, the RESTful interface, is most suitable for

BIM servers. The strength of the RESTful interface is its simplicity. In this architecture, the representation of all resources would be abstracted as a Uniform Resource Identifier (URI). The resources here do not just refer to the files or other tangible objects; it also could be a composition of several files, a table, a query, a result set of queries, and any concepts. The web interface is essential to the usability of a server. As previously mentioned, simulations for different scenes and different purposes are highly domain-specific. Even equipped with a web-page or client-based query tool, the server cannot support all possible workflows and needs of specific data model composition, let alone the possible appearance of new simulation scenes and tools. The domain experts are the ones who know the specific data model request best in a simulation scenario. They can use the RESTful interface to manipulate the data model and build clients directly for one specific domain. A dedicated client could achieve partial model query from the model pool of the server for its own simulation. The easy-to-build clients based on a RESTful interface are loosely coupled with the server and make the methods of data selective model queries, information queries, and operational information queries flexible.

Automated query generation from model view definitions (MVD). IFC is an open source data format for facilitating information exchange in the whole building construction lifecycle which is widely used for BIM. Instead of using full IFC schema, an MVD would be defined for a subset of the IFC schema to satisfy one or several specific data exchange requirements. Whereas an MVD is independent of a particular IFC release, its realization is implementation dependent. It is critical to automate the data exchange process by generating queries from MVDs automatically. The automated generation of queries from MVDs will facilitate, simplify, and streamline information exchange between tools built around BIM servers, and enable flexible queries to BIM servers. This functionality will also enable a BIM server to selectively export relevant information for tools.

Query language. One of the limitations of the current BIMserver.org implementation is that users have to write a chunk of tedious Java code to perform sophisticated queries, which is not user friendly. The previous survey showed that the biggest obstacle for an effective use of BIMserver.org is client interface usability (Beetz et al., 2011). We recommend that the Java code should be hidden from the normal users. Figure 1 shows a code snippet from the BIMserver.org advanced query demo. Only the code in **bold** has to be specified by user. Other code requires extensive knowledge of the underlying programming language to construct and understand, and should be encapsulated by a higher-level declarative query language and made transparent to normal users.

```
@Override
public void query(IfcModelInterface model, PrintWriter out) {
    out.println("Running doors example");
    List<IfcBuildingStorey> stories = model.getAll(IfcBuildingStorey.class);
    Map<Double, IfcBuildingStorey> orderedStories =
        new TreeMap<Double, IfcBuildingStorey>();
    for (IfcBuildingStorey storey : stories) {
        orderedStories.put(storey.getElevation(), storey);
    }
    if (orderedStories.size() > 1) {
        IfcBuildingStorey firstFloor = stories.get(1);
        for (IfcRelContainedInSpatialStructure rel :
                    firstFloor.getContainsElements()) {
            for (IfcProduct product : rel.getRelatedElements()) {
                    if (product instanceof IfcDoor) {
                    ifcDoor ifcDoor = (IfcDoor)product;
                                if (ifcDoor.getOverallHeight() > 2) {
                                    out.println(ifcDoor.getName() + " "
                                        + ifcDoor.getOverallHeight());
}}}}}}
```

Figure 1. Advanced query sample from BIMserver.org (version 1.1 beta, 2012)

A core requirement for the query language is to support partial model query. For example, user A wants window objects and their related geometric objects, while user B requires the same window objects and their penetration property. Although some general purpose query language is available already (e.g. SQL), it is difficult to achieve the partial model query of IFC data models directly (Adachi, 2003). An AEC industry domain-specific query language could be designed for such purpose. For the area that has special requirements on query and design, adopting a domain-specific language (DSL) has been proven as a viable solution. One such example is VHDL, a hardware description language (Camposano et al., 1991). To achieve an AEC domain specific query language, one could start from the query language that could execute partial queries on an IFC compliant data model. Dating back to 2003, PMQL is an effort that tried to achieve partial model query of IFC files (Adachi, 2003). GMSD goes one step further to support better “partial model query on specific model view,” and less “request-response cycles” (Weise et al., 2003).

We believe a domain specific query language is needed and should have at least some of the following characteristics:

- Compatible with MVDs: Queries should be able to be (semi-)automatically composed based on the existing MVDs;
- Compatible with to the RESTful interface;

- Easy for normal users to construct queries;
- Hidden from the users behind GUI and support transaction queries with ACID (atomicity, consistency, isolation, and durability) (Gray et al., 1993).
- Able to reuse routine queries.

DISCUSSION

Current BIMserver.org implementation. The open source implementation BIMserver.org is a promising platform that facilitates information exchange in building projects. It could be a valuable platform for the energy efficient retrofit project too. In the latest version of BIMserver.org, some requirements containing revision control, authority control, and clash detection have been implemented.

BIMserver.org 1.1 was released in November 2011. In this version, there are 9 new features, among which client library, new protocol buffer interface, and new plugins mechanism are most attractive and make the architecture of BIMserver.org more flexible (BIMserver 1.1 beta, 2011). Especially, the change in the plugin mechanism is critical. In version 1.0, the instances of plugins are created in the core part, intertwined with the BIMserver.org kernel logic. The new plugin infrastructure is implemented in the sub-project called “Plugin,” one of the required projects on the build path of the BIMserver.org kernel. This brand new design allows the developers to add new functions to BIMserver.org easily and quickly.

Due to the complexity of constructing sophisticated queries as we mentioned in the section “Query language,” BIMserver.org should encapsulate the routine Java code into some drop-down menu and/or check box based graphical user interface. Taking the query on doors as an example, shown in Figure 1, users need to check the object they want and the range of data and other criteria, like the “OverallHeight” should be more than 2 meters in the example. The back end could generate such code according to the user input automatically. Currently, using Java code is an implementation tradeoff. The ideal solution is to introduce a new query language that fully supports partial model query. Inspired by PMQL (Adachi, 2003) and GMSD (Weise et al., 2003), Beetz et al. (2011) plan to design such a new query language in their future work.

Integration of OpenStudio and BIMserver. OpenStudio is another integrated energy simulation and analysis platform created by the NREL (Guglielmetti et al., 2011). It can be used for managing models and data and facilitating analysis for tools such as EnergyPlus and Radiance. The range of tools that are supported by OpenStuido is not that wide currently, but it works quite well in some dedicated integrated simulation scenes like day lighting simulation. It is potentially useful to

integrate OpenStudio and BIMserver. In such architecture, OpenStudio would focus on the parameter and analysis management, meanwhile, BIMserver could offer the transactional query and data persistence service. To achieve this, a translation between the .osm data (OpenStudio file format) model and IFC data model might be needed to combine the formats supported by the two platforms respectively into one big data model set. Currently both OpenStudio and BIMserver have opened their programmable interfaces, which make the integration of the two platforms easier. We believe it is necessary to build links between the two platforms and make them interoperable.

SUMMARY

In this paper, we first investigated the key BIM server requirements for information exchange of energy efficient building retrofit projects, including automatic generation of queries from MVDs, the RESTful web interface, and domain specific query language. Such requirements would benefit partial model query and adoption of BIM server in energy efficient building lifecycle. The open source BIMserver.org implementation was used as the case study to discuss its current status and limitations. We then identified some future directions to fulfill these requirements based on the open source BIMserver.org.

Acknowledgments. This research is supported by Department of Energy under the project Greater Philadelphia Innovation Cluster (GPIC) for Energy-Efficient Buildings (EEB), a DOE Energy Innovation Hub.

REFERENCES

Adachi, Y., (2003). “Overview of partial model query language,” In *Proceedings of the 10th ISPE International Conference on Concurrent Engineering (ISPE CE 2003)*, 549–555.

Basarkar, M. and Swami, M., (2006). “IFC to CONTAM Translator,” In *Proceedings of SimBuild 2006 Conference*, Boston, August 2–4, 2006.

Bazjanac, V., (2001). “Acquisition of Building Geometry in the Simulation of Energy Performance,” In *Proceedings of the 2001 Building Simulation Conference*, Rio de Janeiro, Brazil, August 13–15, 2001.

Beetz, J., Berlo, L.-v., Laat, R.-d. and Helm, P.-v., (2010). “Bimserver.org – An Open Source IFC Model Server”. In *Proceedings of the CIP W78 conference*, Cairo.

Beetz, J., Berlo, L. v., Laat, R. d. and Bonsma, P., (2011). “Advances in the development and application of an open source model server for building

information." In *Proceedings of the CIP W78 conference*, Nice.

BIMserver 1.1 beta, available at http://bimserver.org/blog/2011/11/18/bimserver-1-1-beta-developers-release/, last reviewed, 01/04/2012.

Camposano, R., Saunders, L. F. and Tabet, R. M. (1991) "IEEE Design & Test of Computers - IEEE D&ToC", vol. 8, no. 1, pp. 43-49

Eastman, C., Teicholz, P., Sacks, R. and Liston, K., (2008). BIM Handbook, Wiley & Sons.

EMF-Eclipse Model Framework Project, available at http://www.eclipse.org/modeling/emf/, last reviewed 01/09/2012.

Gray, J. and Reuter, A., (1993). Transaction Processing: Concepts and Techniques, Morgan Kaufman.

Hirsch, A., Pless, S., Guglielmetti, R. and Torcellini, P.A. (2011). "The Role of Modeling When Designing for Absolute Energy Use Intensity Requirements in a Design-Build Framework". Golden, CO: National Renewable Energy Laboratory, NREL/CP-5500-49067.

Fielding. R., (2000). "Architectural Styles and The Design of Network-based Software Architectures." PhD thesis, University of California, Irvine, 2000.

McGraw Hill Construction. (2009). The Business value of BIM: Getting Building Information Modeling to the Bottom Line.

Moon, H.-J., Choi, M.-S., Kim, S.-K. and 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*, Sydney, November, 2011.

Singh, V., Gu, N. and Wang, X., (2011). "A theoretical framework of a BIM-based multi-disciplinary collaboration platform," *Automation in Construction*, 20(2), pages 134–144.

Watson, A., (2010). "BIM—a driver for change," In *Proceedings of the International Conference on Computing in Civil and Building Engineering*, Nottingham, UK, June 30–2, 2010.

Weiming, S., Qi, H. and Yunjiao, X., (2010). "An Agent-based service-oriented approach for facility lifecycle information integration and decision supports," In *Proceedings of the 2010 IEEE International Conference on Systems, Man, and Cybernetics*, Istanbul, Turkey.

Weise, M., Katranuschkov, P. and Scherer, R. J., (2003). "Generalised Model Subset Definition Schema," In *Proceedings of the CIB-W78 Workshop*, Waiheke Island, NZ, 2003.