ADVANCES IN THE DEVELOPMENT AND APPLICATION OF AN OPEN SOURCE MODEL SERVER FOR BUILDING INFORMATION

Jakob Beetz, PhD / Assistant professor, j.beetz@tue.nl
Eindhoven University of technology, Eindhoven, The Netherlands
Leon van Berlo, M.Sc. / senior researcher, leon.vanberlo@tno.nl
Ruben de Laat, B.Sc. / senior developer, ruben.delaat@tno.nl
Peter Bonsma, M.Sc. / senior researcher, peter.bonsma@tno.nl
Nederlandse Organisatie voor Toegepast Natuurwetenschappelijk Onderzoek (TNO), Delft, The Netherlands

ABSTRACT

The need for Building Information Model (BIM) servers to facilitate collaboration has been repeatedly reported in literature and stated by industry practitioners. To date, only a few commercial implementations of model servers are available. However, these applications are either limited to vendor-specific, proprietary data models or their complexity and pricing structure imposes a considerable threshold to their adoption by SMEs. The academic, standardization and development communities suffer a lack of open platforms that allows the integration and evaluation of research results and prototypes in a larger context. Although promising efforts aiming the creation of free and open platforms for a larger audience have been made in the past (Kiviniemi et al, 2005), none of these initiatives is still in existence to date.

In this paper we report on the progress of the bimserver.org project, a collective effort to create a software framework that serves both as an open, low-cost entry into collaboration using shared building models for SMEs and as a test bed for research and development. We introduce conceptual, methodological and implementation specific advancements that have been made compared to earlier publications. Based on a series of interviews and a survey we report on user experiences that have been made by a diverse audience of architectural, engineering, project management companies as well as educational projects in academic environments and third party developers.

A discussion of future research and development directions concludes this paper. Among these prospective additions to the framework are customizations and tailored solutions of external parties that use the open nature of the platform to suit their specific needs.

Keywords: Model Server, IFC, BIM, Collaboration, MDA, REST, MOF

1. INTRODUCTION

Collaboration in the building industry comprises complex workflows in which information of a multitude of stakeholders has to be incorporated into a common pool of information over an extensive period of time. During the lifecycle of a building – from the initial clients brief to the recycling of its dismantled components – a diverse spectrum of information sources needs to be tightly integrated. Among these sources, technical drawings act as a crucial information conveyor. To keep affected project participants in the loop, frequent exchanges of sets of such drawings have traditionally played an important role in building and construction workflows. A number of project collaboration methods and tools exist, that allow the controlled spread and integration of information among stakeholders. In contemporary project environments, storage and access of CAD drawings and other digital documents in electronic collaboration platform has become commonplace.

With the advent of object oriented product modeling approaches in building design, construction and operation however, the requirements for the exchange of information changed. Instead of swapping complete sets semantically poor geometric information transported in two or three dimensional drawings, changes in single building components and their effects on related aspects can now be isolated and communicated separately. A prerequisite for such information exchange is the existence of both a common data model and a data base allowing individual CRUD (create, read, update, delete) operations on the level of semantically meaningful objects or even attributes. Such model servers have been identified as a crucial requirement for an increase in productivity and efficiency in the building sector time and again (Hannus et al 2003, Eastman et al 2008). Model servers fall into two categories: *a) Persistency tools for proprietary, native application models enhanced with versioning and multi-user capabilities*. Examples for such tools can be found in recent editions of commercial software packages such as ArchiCAD™ and Revit™. *b) Persistency tools for open, vendor-neutral models stemming from heterogeneous applications.* In heterogeneous software environments, where a number of domain specific applications need to exchange and integrate data, the need for an open standard has been emphasized by many authors (Tolman, 1999; Howared and Björk, 2007; Cerovsek, 2011). The dominating model covering aspects of several domains involved in building and construction has been the IFC (Industry Foundation Classes), which is currently in the process of becoming an ISO standard. At present only a only few commercial tools exist that allow the native storage of instance models of this standard. In the past, the Sable initiative has aimed at an open platform for purposes of experimenting with central and federated storage of BIMs (Kiviniemi et al, 2005). The initiative resulted in a prototype system based on object-relational mapping into. Unfortunately this project is discontinued, though many valuable conclusions can be drawn from experiences made during its execution. Among them are conceptual as well as performance and usability issues some of which have been attributed to the overhead in the API exposing hardwired model views over a SOAP interface.

The aim of the bimserver.org initiative is to provide an open platform for the research, development and deployment of model-based collaboration. While the platform itself offers a range of re-usable basic functionalities such as versioned persistency on the object level, its modular and open architecture is designed to be extended and improved by a growing number of researchers and developers.

2. ARCHITECTURE AND METHODS

As partially reported upon in earlier publications, the basic conceptual decision of the bimserver.org platform is its foundation on a strictly model based architecture. At its center, an Ecore model generated from the EXPRESS representation of the IFC schema is used. Every entity in the model is augmented with versioning information. Upon import of and IFC instance model through one of the importers provided, instances of these objects are created and a unique combination of identifiers for the project, the object itself and the revision is assigned to each entity. New revisions of a model can then be added without overwriting previous versions, even if the object remains the same and only a single attribute changes, thus preserving the complete history of a project over time. For more information on the versioning strategies see (Beetz et al, 2010). Physically, the object instances are stored in one of the possible database backends. At present, a key-value store based on the open source BerkeleyDB with one table per entity class is the main implementation of the abstract database interfaces, although different database backends can be added by developers in future.

In order to integrate partial domain models from different building project contributors, a project can freely be subdivided in cascading subprojects. For example, a model coordinator might create separate subprojects for architectural, structural and mechanical equipment models. In an initial revision, an architect checks in a model. A structural engineer downloads this model as an overlay, adds structural components and uploads the resulting model into the structural subproject. A clash detection (van den Helm et al 2010) might be carried out on the sever itself, using the IFCEngine (Bonsma 2007) as a geometric kernel. The complete model including both sub models can be downloaded and will be merged together on the fly by the server. Access rights can be assigned per subproject, which allows a fine grained compartmentalization of a complex project eliminating the risks of interference and corruption of other projects participants' data.

Queries for particular objects or entity types can be carried out on arbitrary revisions and advanced queries allow the creation of detailed inquiries. This is achieved by Java code that can be uploaded to the server and compiled at runtime.

All of this functionality is wrapped in an API (the bimserver.org 'actions') and exposed through different interfaces on the web either by an external servlet container such as Tomcat or a built-in Jetty server. Based on these interfaces exemplary client end-user interfaces have been created. Among them, a browser-based user interface is provided that allows the immediate use of the platform without prior technical knowledge by end users.

3. IMPROVEMENTS AND NEW FEATURES

Next to several smaller performance improvements and a considerable amount of bug fixes, a number of features have been added to the platform since its introduction that are reported upon in this section.

As an alternative to the existing SOAP and REST interfaces which expose detailed aspects of the platform, a new protocol buffers (Google Inc., 2008) interface has been added. As its implementation is not fully completed yet, it is not possible at this point to quantify performance gains, yet for certain scenarios, we expect a considerable increase in communication throughput. In particular, the exclusion of the overhead of encoding and decoding XML based SOAP messages between client and server by replacing it with the light-weight binary format is expected to gain performance in scenarios where large amounts of messages have to be exchanged between a client and a bimserver.

A new lazy loading feature allows the delay of the in-memory instantiation of data from the database until it is needed. This keeps the initial and overall memory footprint on the server side small since only data that has to be operated upon has to be loaded. That way, queries on the model which for example only involve a limited amount of building stories will be carried out more efficiently since all unaffected model areas will not undergo expensive retrieval an comparison operations.

Often, when heterogeneous applications are involved in the creation of the overall common model, different measuring units (millimeter, meter etc.) have to be dealt with during the import of partial models. To overcome mismatching setups, the merging of sub models following the sub project approach described earlier has been enhanced with automatic conversions between these units: An architectural sub model that has been created in meters will be merged with a construction model by adapting a common unit system and converting them during checkout time. The originally uploaded model however remains unmodified, enabling future reversions within a single domain.

One of the performance bottlenecks detected in earlier tests and use cases was the creation of temporary files for the communication with the underlying geometry kernel in the IFCEngine. Now, instead of encoding an intermediate IFC file, loading and parsing it on the geometry kernel side again, data is streamed into the integrated subcomponent directly.

As the system evolves and matures, crucial parts of the underlying components change. One of these elements is the data base schema itself. In the recent versions of the platform this is taken into account through a programmatic update process that applies cascading updates to existing databases for downward compatibility. This has become a necessity due to the rising deployment of the platform in mission critical commercial scenarios, as reported in the following section.

One of immediate reactions during the presentation of the bimserver.org platform at the CIB W78 conference in 2010 was the request of the addition to handle COBie (East and Brodt 2007) data. An initial interface has been added and is currently in a testing phase.

An increasing number of users request the addition of further specific exporters and serializers. In reaction to these requests the serialization component of the platform has been restructured in a more generic manner. The aim is to keep only a limited number of essential exporters integrated into the core and provide mechanisms for the development of additional, pluggable extensions by the community.

A similar approach has been taken for GUI and visualization components. By moving both from the platform kernel into separate spaces we would like to encourage developers to add their own user interfaces and viewers.

4. USER FEEDBACK AND SURVEY

In order to augment the feedback gathered in informal face-to-face interviews, the bimserer.org mailing lists and public forums, an online-survey has been created that gives some more insight into the backgrounds, experiences and wishes of end-users and developers.

Even though the spectrum of bimserver.org users spans 15 countries, the biggest user base is found in the Netherlands (45%), followed by the US (20%) and Norway (10%). This Dutch predominance can be explained by a number of local events such as workshops ("bimserver.org breakfast") and bimserver.org presentations on trade fairs, business events and the networks of the project team within the Dutch building industry. In fact, the survey shows that most respondents have learned about the project by word of mouth. A third of the respondents are commercial end-users from architectural, engineering or consultant SMEs, followed by software developers (23%) and academic researchers (16%). Service providers and academic lecturers using the platform for teaching purposes complete the spectrum of users. The platform is actively used by 5 people on average per server instance, with reported peaks of 60 parallel users during a collaborative pilot test on the public "bimcaseweek" workshop. Additional informal feedback for this event showed the capability of the platform to handle large, diverse models, even though powerful hardware was mandatory and collaboration was often slowed down by mismatching sub model coordinates and other management issues. The platform is mostly used in the design development stage of a project (32%) with pre-design (20%) and construction (15%) as the runners up and occasional applications during schematic design, procurement and operation. Revit™ (26%), ArchiCad™ (25%) and Solibri™ (12%) are the dominant applications used as source or target applications by the users, and a majority (57%) of the respondents made earlier use of collaboration platforms, tools and model servers, among which are dedicated IFC model servers (14%) and proprietary model servers (24%).

The model merging capabilities of the platform are both the most used (26%) and the most important (46 % ranked them as "very important") features indicated by the respondents. Revision management (22%), simple queries and filters (16%) as well as advanced queries and rules (12%) are also frequently used.
Much to our surprise, the biggest obstacles for an effective use of the platform were neither the memory use (18%), nor the operation speed (12%) but the usability of the provided client user interface (24%) and the visualization (18%). Only a small number of respondents were unhappy with the stability of the system. By far, the most important external factor limiting the usefulness of the system is the lacking quality and compliance of IFC models exported from the source applications (42%), followed by the import capabilities of target applications (21%) and the absence of interoperability with specialized domain specific tools (18%). Concerning the lack of quality of the source models being exported from the various tools, two main categories of difficulties can be identified: a) Model-inherent obstacles and b) implementation issues. Among the challenges inherent to the IFC model schema itself and its architecture are the lack of GUIDs and owner histories for important parts of the model such as the geometrical representations. Their absence makes it difficult to determine subtle changes of an object whose changes are only implicitly 'hidden' in geometric reconfigurations without being described explicitly. Other issues are the extensions of semantic meanings via the proxy mechanism, the lack of parametric coupling between entities and their attributes (direct or via PropertySets) with their respective geometric representations and the flexibility of the model with respect to optional/mandatory information. It can also be argued that the lack of schematic rigidness which allows to model information in more than one way and which is often only limited by informal "implementers agreements" documents is an inherent shortcoming for a standardized data exchange format designed to address interoperability. Implementation issues become apparent when examining instance models stemming from various packages found 'in the wild' range from precision issues to lost property sets, changed GUIDs and simple syntactical errors. Detailed studies on these issues can be found in (Lipman et al 2010) and (Ma et al 2006). Promising developments such as the automated conformance testing (GTDS 2011), increased user-demand for high quality instance files as well as local regulations that embrace open data standards will eventually lead to implementations of higher quality.
A tighter integration of the overall workflow e.g. by means of IDM or the coupling with document management systems and the capabilities of the IFC model schema itself are considered only minor shortcomings in comparison to the import and export.
Visualization capabilities in the form of bitmap snapshots or interactive WebGL representations are highest on the wish lists of users (41%) with a number of additional export formats (Excel 18%, gbXML 6%) and the export of standardized views according to an MVD as follow-ups on this whish list.

5. CONCLUSIONS AND FUTURE WORK

In this paper we have presented advancements in the development of the open source bimserver.org software platform. We have shown how it can be used as a versioning system with enhanced sub-project merging in real world applications and how features such as queries and filters can be used to extract desired subsets of larger model populations. We have reported upon new developments since the initial launch and have gathered and analyzed feedback from a user survey.
Future enhancements and extensions of the project will address issues brought forward by end-users and fellow researchers. In the near future however, we concentrate on the stability of the core system and will continue to improve the robustness and openness of the programmable interfaces. We consider this to be

the only feasible approach for the bimserver.org initiative to expand beyond the resource restrictions of the current, small development team

Currently, prototypical real-time visualizations based on the open WebGL standard is nearing completion by members of the larger bimserver community. We will investigate offline bitmap rendering and automated snapshots per revision which are included into the client interfaces. One of the potential directions here is the implementation of the Building Collaboration Format (BCF) (buildingSMART BCF, 2011) which allows a neutral description of views on particular building elements and verbalization of related issues. It is feasible to follow design revisions automatically by carrying over particular views from on revision to revision in an automated manner. This would be a significant progress towards a tighter integration into the work processes, since the format has already been implemented by a number of packages. Similarly, the addition of Model View Definitions and Exchange Requirements as part of the Information Manual initiative in the form of filters and query library is expected to enhance efficiency. We are aiming at the design and implementation of a new query language, and will take promising existing efforts such as the mvdXML initiative, Adachis' PMQL (Adachi, 2003) and the GMSD (Weise et al, 2003) into account and hope on the collaboration with the respective authors. In a more distant future, we hope to integrate spatial reasoning operators into this query language. Ongoing developments of external bimserver.org contributors also include adaptions for the use of the server as IFD concept repository, as well as the object-based integration of document management systems.

In closing we would like to thank the bimserver.org community for their valuable contributions and feedback, and hope to see a future increase in collaborative research and development.

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*

Beetz, J. (2010) "bimserver.org – An Open Source IFC Model Server" *In Proceedings of the CIB W78 2010: 27th International Conference –Cairo, Egypt*

Bonsma, P. (2007). IFC Engine Viewer specification, *available at web site*: http://www.ifcbrowser.com/ *last accessed on 01.05.2011*

buidlingSMART BCF (2011) "BIM Collaboration Format" Specification. *Available online at http://87.106.252.103/apex/f?p=101:120:8129684495133126::::SELECTED_NODE_P120:359405, last accessed 01.05.2011*

Cerovsek, T. (2011) "A review and outlook for a 'Building Information Model' (BIM): A multi-standpoint framework for technological development" *In ADVEI 25 (2011), 224-244*

East, W. E. and Brodt, W. (2007) "BIM for Construction Handover," Journal of Building Information Modeling (2007) 28–35.

Eastman, C. et al (2008) "BIM Handbook", John Wiley & Sons, Hoboken, New Jersey

Google Inc (2008) "protobuf. Protocol Buffers - Google's data interchange format" *specification available online at* http://code.google.com/p/protobuf/, *last accessed 01.05.2011*

GTDS (2011) "Global Testing Documentation Server", available online at http://gtds.buildingsmart.com *last accessed 01.05.2011*

Hannus, M. et al (2003) "Construction ICT Roadmap" *Report D52 of the ROADCON: IST-2001-37278 project*

Helm, P. van den, et al (2010) "IFC-based clash detection for the open-source BIMserver". *In Proceedings of the International Conference on Computing in Civil and Building Engineering ICCCBE 2010*

Howard C. and Bjork, B.C. (2007) "Building Information Models–Experts' view on BIM/IFC developments," *In Proceedings of the 24th CIB-W78 Conference, Maribor, 2007*.

Kiviniemi A, et al (2005) “Integration of Multiple Product Models: IFC Model Servers as a Potential Solution”. *In Proceedings of the 22nd CIB-W78 Conference on Information Technology in Construction*

Lipman, R. et al (2010) “Assessment of conformance and interoperability testing methods used for construction industry product models” Automation in Construction 20 (2011) 418–428

Ma, H. et al (2006), “Testing Semantic Interoperability” *In Proceedings of the International Conference on Computing in Civil and Building Engineering, Montreal, Canada, June 2006.*

Tolman,F.P. (1999) “Product modeling standards for the building and construction industry: past, present and future,” *In Automation in Construction 8, no. 3 (1999) 227-235*.

Weise, M. et al (2003) “Generalised Model Subset Definition Schema”, *in: Construction IT: Bridging the Distance, Proceedings of the CIB-W78 Workshop, Waiheke Island, NZ*