Using Open Standards Based Building Information Modelling to Simulate Actual Design and Construction Processes

Igor Svetel[1], Milica Pejanović[2], Nenad Ivanišević[3]
[1]Innovation Center, Faculty of Mechanical Engineering, Serbia, [2]Faculty of Architecture, University of Belgrade, Serbia, [3]Faculty of Civil Engineering, University of Belgrade, Serbia.
[1]isvetel@mas.bg.ac.rs, [2]pmilica@arh.bg.ac.rs, [3]nesa@grf.bg.ac.rs

Abstract. *The paper describes pilot project conducted to achieve first understanding of the IFC standard and BIM process in Serbia. During the project a research team have developed information model of the actual building using IFC standard and BIM technology and used that model to simulate an actual construction processes. The experience from this project shows that BIM principles and the way IFC standard is incorporated in applications are still a set of recommendations that each software developer interprets separately. At the end of the paper a possibility of further development that would bring BIM and related ICT standards to expected functionality is discussed.*
Keywords. *BIM; IFC; BIMserver; design; construction.*

GENERAL

Recently a new wave of technical advances has swept the architecture, engineering and construction (AEC) industry. The term Building Information Modelling (BIM) is in use to denote a spectrum of both technological and methodological changes in the way the AEC profession is conducted. The objective of promoting BIM generated models as the international standard that will replace traditional AEC project documents spurred worldwide interest in the necessary adaptations of national AEC standards. The project described in the paper was commenced as the mean to achieve a first experience with BIM technologies for the AEC industry in Serbia.

The project goal was to create a BIM model based on the data from the actual building project and to evaluate the model by comparing it with the actual design and construction processes. The project team decided to skip thorough learning of available commercial BIM applications and to start modelling after brief acknowledgment with applications using available demos. This approach simulates the method that most AEC teams in Serbia use when they introduce new technology in their existing business process, so we expected to experience all problems that AEC industry in Serbia will face in the future. Second, as we have already advocated approach to computer-based design that is similar with BIM, we wanted to compare functionality of existing BIM applications with our expectations what BIM ought to provide.

OBJECT ORIENTED MODELLING AND BIM

The object oriented modelling principle was developed in computer sciences during the 1970s as an answer to growing software complexity and the need to invent a method that will enable rapid and reliable development of complex systems. Today, object oriented modelling is de facto standard in the software industry and is becoming an essential standard for modelling all other branches of human endeavour.

It is logical to expect that BIM should embrace at least following object oriented modelling concepts: class, object, inheritance, and abstraction (Rumbaugh 1990). A class is the blueprint (specification) defining abstract properties of the group of similar items, their relations to other classes, and the rules of their behaviour. Using a class, properties common to all items pertaining to that class are enumerated and their type and range are defined. An object is an instance of the class i.e. a specific sample of the class. An object is created when all abstract parameters of the class acquire their actual values, and this process is termed as instantiation. An object must belong to only one class as an instance of that class. An inheritance is a process of deriving new class (subclass) from actual class (superclass). Subclass inherits attributes and methods of superclass, and may have extra attributes and methods defining specialization of the class. An abstraction is a mechanism to reduce and factor out details so that one can focus on a model using as much concepts as needed on particular level of analysis.

If we see BIM as the exercise in object oriented modelling then the process should start from defining all classes representing all building elements. Those classes should be implemented in the BIM application as the library of elements. Actual modelling turns into a process of choosing classes and instantiating them into objects depicting actual building elements (Figure 1).

OPEN STANDARDS FOR AEC

Based on years of research on a general data model for AEC (Eastman et al. 1991; Björk 1989) the term Building Information Modelling (BIM) denotes a process of using Information Technology (IT) to model

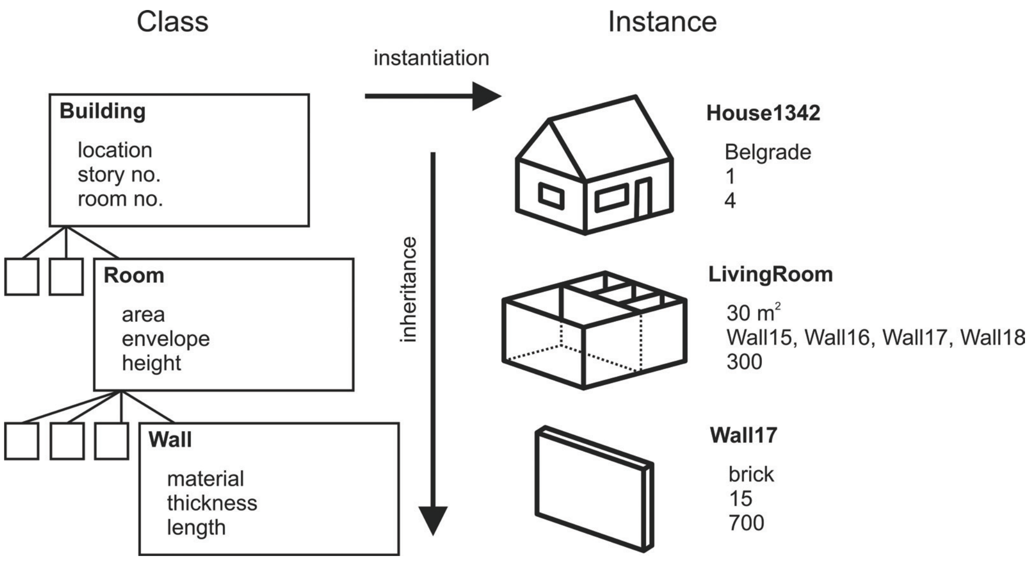

Figure 1
Main object oriented modelling concepts and their relation to BIM

and manage data encompassing the whole facility lifecycle (Lee et al. 2006). The BIM concept means to build a facility virtually, prior to building it physically, in order to work out problems and simulate and analyze potential impacts. It is easier to fix a problem by moving element with a mouse than to demolish and rebuild elements on a construction site. The commercially developed BIM applications support creation of the computer-based facility model using parametric three-dimensional (3D) components with attached descriptive parameters that are necessary to fully identify particular elements. Still, those applications typically use proprietary data formats to represent facility models thus keeping all information locked in distinct software.

The need to establish interoperability among applications dealing with different phases of the facility lifecycle, such as architectural design, civil engineering, HVAC design, building construction, and facility management (FM), was met with the development of the Industry Foundation Classes (IFC) standard [1]. The currently available model is Version 2, Revision 3 and is also registered as the ISO/PAS 16739:2005 standard. IFC is a neutral and open model whose development is conducted by the International Alliance for Interoperability (AIA), which has 550 member organizations in 24 countries. The standard provides the following basic functionality:

- Data interchange without information loss among all AEC and facility management (FM) applications.
- Unified model-based description of all building components.
- Information on the graphical representation of components.
- Description of relationships with other components and their location in the whole structure.
- Link to property and classification data, and access to external libraries.

The open specification of the IFC data model allows commercial software developers to write interfaces for their software that enable exchange and sharing of the same data in the same format with other software applications, regardless of the internal data structure of any individual software application. All leading software companies like Autodesk, Bentley System, Graphisoft, Nemetschek, Data Design System, Solibri, Tekla, Archimen Group, Vector–Works, etc. support IFC in their applications.

Being an object-oriented data model, the IFC standard is comprised of class definitions representing all things and events occurring in the facility lifecycle. At the top of the hierarchy is a domain layer that describes classes related to basic functional units: building controls, plumbing and fire protection, structural elements, structural analysis, heating, ventilation and air conditioning, electrical circuits, architecture, construction and facilities management. Below that layer rests the interoperability layer that defines all classes essential for connection and cooperation among disciplines. Next is the core layer, containing basic model classes depicting controls, products, and processes. The resource layer is at the bottom, embodying classes that represent all building elements. Elements encompass not only physical components, as traditional models, but also actors and their roles, time, price, approval, etc.

The IFC standard does not produce one monolithic data model encompassing the whole lifecycle. Instead, many separate models are generated. In the context of IFC, a View is a defined as a subset of the IFC Object Model that a number of implementers have agreed to support in their implementations. The software certification process is conducted according to IFC Views. Depending on agreement many IFC Views can exist with partially overlapping content or with entirely different contents. The data exchange between applications should occur within the scope of a specific View. The entire facility lifecycle is represented across multiple Views.

The IFC standard relates to the representation of a particular instance of the facility, its components, properties, and relationships. Using the vocabulary of object-oriented modelling it can be said that it deals with object instances. It does not allow representation of the object classes and their relationships (i.e., that part is covered by the International Framework for Dictionaries (IFD), registered as the ISO 12006-3:2007 standard [2]).

IFD is the classification system for all information in the AEC/FM field. It is an object-oriented framework that defines objects, collections and their relationships. It is intended to work as the overarching structure that will provide support for the development of the unified AEC/FM vocabularies at the national, regional or domain levels. Since all share the same structure it will be possible to translate terms between languages and domains, preferably using automated software agents. IDF identifies each object in the model and this provides the capability to define context within which a concept is going to be used. Each object can have multiple names providing for the definition of synonyms or usage in different languages. An object is related to a formal classification system using references. The standard supports the following types of objects: Subjects, Activities, Actors, Units, Measures with Units, and Properties. Relationships are divided into: Association, Collection, Specialization, Composition, Involvement, Property Assignment, Sequence and Measure Assignment. Using these mechanisms the user can create a model-based definition of all concepts in AEC/FM including facts about classification systems, information models, object models, and process models. In other words, IFD functions as the IFC metamodel. Also IFD provides a unique global reference for any AEC/FM concept. The mechanism that relates IFC and IFD standards is scheduled to be published in the IFC 2x4 standard revision. The actual realization of IFD is the IFD Library, an international initiative currently run by four nations: Canada, Netherlands, Norway and USA. The purpose of the library is to provide semantic knowledge to the construction industry in a global and uniform way.

In addition to the above described standards a second type of interoperability formats has been developed based on another open standard - eXtensible Markup Language (XML). XML is a general-purpose specification capable of describing published data (Harold and Means, 2004). The data description mechanism is based on the insertion of tags in the traditional text and the user can choose any term to define a particular tag. The language permits representation of arbitrary data arranged as a hierarchical tree with one element serving as the tree root (Harold and Means 2004). XML enables the structured representation of any kind of information but does not provide any mechanism to infer the meaning of the terms used in tags. One approach to the definition of a tag's meaning is the XML schema. It is a language that provides a description of a type of XML document, usually articulated in terms of constraints on the structure and content of related XML documents. Many schemas have been developed for the AEC/FM field. The gbXML (Green Building XML) schema is used for describing data relating to the building energy efficiency of the facility and its impact on the environment. The aecXML schema is used for depicting all building data in design, engineering and construction disciplines, and the CityGML schema is used for geo-spatial data representation. Also IFC data can be represented with the ifcXML schema. Since the IFD is an EXPRESS model, the EXPRESS to XML Schema Converter [3] can be used to obtain the XML schema for IFD.

Open standards developed for the AEC/FM industry relate to the problem of interoperability since this is the most obvious obstacle in the industry. These standards enable the highly structured representation of information about buildings but do not consider the problem of information reuse outside of the context of a particular facility lifecycle or the automatic creation of new information for later reuse.

THE SCOPE OF THE PROJECT

The first phase of the project consisted of the development of the electronic model of the office building based on the existing documentation. The traditional documentation consisting of the 2D CAD drawings was used as the outline to construct elements of the electronic model. The process is simplified as the existing drawings can be used as trace layouts enabling quick and precise location of the elements. The commercial BIM application ArchiCAD v12 was selected for the task because its routines for the modification of the element's parameters have visually most comprehensive interface. The construction starts by selecting design tool specific for some class of building elements, like wall, beam, column, slab, stair, window, door, etc. The user creates required object by providing values of the parameters that suites his/her design. If the supplied procedures or object library doesn't suit designer's needs the user can use the basic 3D modelling techniques or some external modelling application to create required element.

The reliance of the system on element libraries is the main characteristic of the BIM software. The user is forced to think about required elements in his/her design and to obtain in advance all necessary values required for the elements he/she wants to construct. Thus the greatest burden in the process is shifted toward initial design phases and to the preparation of necessary information required to construct building elements. That way system prevents inclusion of undefined elements yielding to unforeseen work and costs that in traditional practice can often be detected only during construction phase. Once created an element becomes part of the electronic building model and all future operations refer to that unique data structure. All graphical representations like layouts, sections, perspective view or virtual reality walkthrough use that single data structure to create required representation.

The second phase of the project included simulations of the processes occurring during actual building construction like contracting, and cost, time, and resource management. The commercial BIM application that was used in the first project phase demonstrated few shortcomings in this phase. Its reliance on the library for the creation of new elements in the building model was perfect in the first phase where the goal was to create model from a scratch. But second phase required large amount of changes and modifications of existing data. The process of instantiating objects from the library that is implemented in the ArchiCAD does not allow definition of the types of objects. To select all objects in the model that belongs to same type a user have to use search procedure to find all objects having same set of common attributes. The second phase required too much effort to find smart ways to select types of object in the model in order to modify its parameters. Whole process had to be adapted to the application's behaviour instead of following natural sequence. Problems with ArchiCAD motivated project team to expand the scope of the project and to conduct experiments with Autodesk Revit application. The unique parametric change engine and support for management of data both on element and class levels provided environment that is fit for all requirements of the second phase of the project. First experiments were conducted using demo Revit files. Unfortunately, further experiences with IFC file format and with the Revit's object modelling mechanism, which will be further documented later in this paper, diminished our anticipations.

The third phase of the project investigated IFC based interoperability on three levels: 1) using solely one commercial software, 2) using IFC format to interchange data between two commercial application, but using their native formats to keep project data, and 3) using Open source BIMserver to keep whole project data in IFC format and using commercial applications do create and modify model's data. It was surprising to observe that even same application was unable to restore full project information from the exported IFC data. Even parts of geometry were missing, and parameters describing particular objects were significantly reduced, constraining parametric modelling of the imported project elements. Data interchange between different applications suffered from the same defects. To our surprise same IFC files, when browsed with free applications like Nemetschek IfcViewer [4], DDS-CAD Viewer [5], or Tekla BIMsight [6] demonstrated less errors and with more parametric information.

The third experiment with BIMserver revealed that crucial problem in interoperability is the commercial application developers' habit to treat modelling mechanism that lies in the core of their software as single representation method, and to adapt information available in IFC files to their needs.

DEDUCTIONS FROM THE PROJECT

The experience from this project shows that BIM principles really changes established practices in architectural design, and shifts focus of the architecture from the creation of design documentation to the real building modelling. Unfortunately the BIM principles are still a set of recommendations that each software developer interprets separately.

Even if object oriented modelling principles represent logical choice on which BIM software should be based, and that IFC standard is an object-oriented data model the core software mechanism in commercial applications was developed before the advent of IFC format. In the case of ArchiCAD that mechanism is based on procedural modelling language that doesn't follow principles of object oriented modelling, albeit in the version 14 of the software, which was released after the conclusion of this project, it is possible to distinguish information pertaining to types of building elements from information about instances. The newer software like Revit has implemented distinction of instances from types of elements in its core software mechanism enabling it to behave closer to principles of object oriented modelling. Unfortunately relations among elements are modelled in accordance to Revit's unique parametric change engine that treats relations as binary relations among elements. A user should explicitly attach each pair of elements one to another to model a relationship. While this mechanism represents advantage comparing to other commercial software it still lacks ability to model property common to group of elements as the property that elements inherit from the class higher in the hierarchy.

Related to the discussion about adherence to the object oriented modelling principles is the issue of element libraries. If we look at the IFC standard we see that it defines classes necessary to represent all concepts related to building during its lifecycle. Regrettably, at the time when development of the commercial software started IFC was still under development and could not provide inspiration for software developers. Instead element libraries closely follow core software architecture. As a result BIM applications are completely incompatible on the level of element libraries. It is possible to export element from one application, to import it to another, and with a moderate to a high effort to incorporate it in the library. But the element in the library loses majority of parameters, and behave like 3D block. In the case of Revit it looks possible to create fully parametric library element from imported object using Family Editor. But in the case of ArchiCAD it is impossible since parametric elements in its library can be constructed using only special purpose GDL language.

On the level of modelling capability BIM applications also show disparity. On the first glance all applications look similar, providing to user modelling tools that cover all building elements. But the modelling knowledge acquired with one application can not be transferred to another application. ArchiCAD has rich modelling interface, which visually demonstrates all parameter modifications, enabling easier learning. On the other hand modelling principles differs from the element to the element, and it is necessary to master all peculiarities to master modelling techniques. It is regrettable that some functionality like solid modelling, that was a research topic some 40 years ago, still does not function as expected creating artefacts and preventing spontaneous creation of the building model. The Revit has different concept providing same modelling method to all elements, but the interface is less informative and requires a lot of learning effort. On the level of the change management ArchiCAD resembles old CAD systems with its association of elements to layers and need to search for elements that need to be changed. Also it is not possible to change all parameters independently of the view in which the user currently works. Revit on the other hand relies for its functionality on the unique parametric change engine that enables modification of any parameter in any view and automatically propagates that change to all affected elements. Because of described functionality the program is an example that other developers should follow.

The experience with the use of the IFC standard shows that seamless data transfer is still something to expect. Often some data is missing and some strange objects appear. Also most of IFC related data have to be entered manually instead of deduced form the building information model. The problem obviously arises from the fact that all commercial applications have their own data structures that represent building model and that in modelling IFC import and export developers try to adapt information available in the IFC standard to their data structures. During the project a valuable help was provided by free applications like Nemetschek IfcViewer, DDS-CAD Viewer, or Tekla BIMsight. Without them it was impossible to deduce if false transfer of data between commercial applications was to blame on improper export or import procedures. Similar function was provided by BIMServer [7] application. In addition BIMserver provides centralized storage and version control that simplify exchange of IFC files. Also, ability to decompose project to subprojects and to provide their clear management enables easier collaboration.

FUTURE DEVELOPMENT

BIM and IFC definitely represent great technological advancement in the AEC field. But, as it is a case with all technologies, its embracement is not a straightforward process. Users are reluctant to change their established habits of working, and to invest time or money into a new business process. Without a pressure from the users, software vendors will continue to expand existing software architectures, and to force users to accept their applications as proper BIM solutions. Based on the experience from our project we think that future advances in the BIM acceptance relies on building a greater interest in BIM related standards like IFC, IFD etc.

Since the IFC is the open and independent standard we see it as the primary candidate for keeping all project information. BIMserver application already fosters the view of the IFC format as the central project repository. At the moment it is hard to expect that users will readily accept this idea, since it is evident that native commercial software formats provide models that are richer in information, and that require less effort to construct. The solution is the development of more free applications that supports IFC based BIM. The BIMserver is the platform on which such development could be based. Since it is Open Source application, everyone can modify it to its purpose. In the first phase the development should be directed toward tools that enable modification of existing IFC models, since it is hard or impossible to do this in commercial applications because they transform IFC model to their native formats during import process. Further development could provide greater control of users over parameters in IFC models enabling creation of richer information content. Hopefully wider use of the IFC standard spurred by the existence of free IFC based BIM tools will persuade software developers do modify data structures in their applications to better fit IFC structures. This development direction could evolve toward distributed design environment that we envisioned long ago (Petrović and Svetel, 1999), where collection of design agents operates on the unique standardized building representation.

Second direction of development could be directed toward applications that manipulate information about IFC models (Svetel and Pejanović, 2010). The IFC standard is intended to represent only object instances, so it is valuable to keep information about particular projects that can be used as examples in the future. If we want to infer information about types of objects or to talk about experience gained on more projects or to create application independent libraries of objects we need other information standards.

The above described IFD standard provides a step in this direction but IFD only provides domain ontology, the structure of concepts and the relationships between them. The standard lacks power to express classes, aggregations and rules. Information standards related to those issues are developed under Semantic Web initiative. The goal of the endeavour is to build a unified information medium that is both understandable for people and computers, and that can be used for the automatic deduction of meaningful inferences. The Semantic Web knowledge representation is based on the layered structure of representation standards. The upper layer exploits functionality of lower layers and provides greater semantic expressiveness. At the bottom level resides XML. Meaning is expressed in the next layer containing the Resource Description Framework (RDF), a data model for representing information about entities on the Web. The next level of the semantic expressiveness is achieved with ontology. In the Semantic Web domain, ontology is identified as the formal representation that defines relationships among terms. The first level of ontological functionality is achieved with the RDF Schema (RDFS). Like other schema languages, RDFS provides information about basic RDF structures. It accomplishes this task by supplying constructs that allow the declaration of classes, subclasses, property, and subproperty relationships among resources. RDFS provides a limited set of inference rules that are restricted to the transitive closure of the hierarchies. The Web Ontology Language (OWL) currently provides the highest level of ontological functionality among Semantic Web technologies. It is a family of languages based on two semantics. OWL Lite and OWL DL are based on Description Logic semantics that guarantee completeness of reasoning. OWL Lite is a restricted version of OWL DL and is intended as a quick migration path for thesauri and other taxonomies. OWL Full provides maximum expressiveness and the syntactic freedom of RDF, but does not support complete or efficient reasoning. The language provides constructs like class, property, property restrictions, Boolean combinations, enumerations and instances. A wide range of services like reasoners and editing tools enable users to express and test knowledge using this formalism leading to the widespread acceptance of this technology. So far, few prototype systems that link Semantic Web and IFC standard are developed. The easier technique to extend existing AEC standard formats and enable knowledge management functionality is to add semantic annotations using RDF. The method is demonstrated in the system for conformity checking in construction. The norms are extracted from the electronic regulations and formalized as SPARQL queries in terms of the IFC model. The conformity checking process is based on matching an RDF representation of a project to a SPARQL conformity query. The project's RDF representation is extracted from the ifcXML schema and later manually enriched with domain knowledge. More projects are using OWL to add knowledge management functionality.

CONCLUSION

The initial goal of the project was to use solely BIM principles and IFC standard to model actual building. Project demonstrated that assertions on conformity to the BIM process and IFC standard given by majority of commercial BIM software developers are exaggerated and that functionality of their products substantially differs. The adoption of the BIM method and IFC standard on the current development level requires cooperation between AEC staff with the teams specialized in the application of information technologies in the AEC industry. Further development is necessary to achieve full potential of the BIM methodology, and it should be based on the available BIM related standards.

ACKNOWLEDGMENTS

This work was supported by the Ministry of Science and Technological Development - Republic of Serbia under grant TP - 16025. It is a part of the project 'Pilot Project: Application of the IFC (ISO/PAS 16739:2005) Standard in Drywall Systems.' The project director is Dr Igor Svetel.

REFERENCES

Björk, B-C 1989, 'Basic structure of a proposed building product model', *Computer-Aided Design*, vol. 21, no. 2, pp. 71-78.

Eastman, CM, Bond, A and Chase S 1991, 'A Data Model for Engineering Design Databases' in JS. Gero (ed) *Artificial Intelligence in Design '91*, Butterworth-Heinemann, Oxford, pp. 339-365.

Harold, ER, Means, WS 2004, *XML in a Nutshell, Third Edition,* O'Reilly Media Inc., Sebastopol, CA.

Lee, G, Sacks, R and Eastman CM 2006, 'Specifying parametric building object behavior (BOB) for a building information modeling system', *Automation in Construction*, vol. 15, no. 6, pp. 758-776.

Petrović, I. and Svetel, I. 1999, 'From Number Cruncher to Digital Being: The Changing Role of Computer in CAAD', *Architectural Computing from Turing to 2000 eCAADe-99 Conference Proceedings*, Liverpool, UK, pp. 33-39.

Rumbaugh JR et al. 1990, *Object-Oriented Modeling and Design,* Prentice Hall, Englewood Cliffs, N.J.

Svetel, I and Pejanovic, M 2010, 'The Role of the Semantic Web for Knowledge Management in the Construction Industry', *Informatica (Slovenia),* vol. 34, no. 3, pp. 331-336.

[1] http://buildingsmart-tech.org/specifications/ifc-releases/ifc2x3-tc1-release/summary

[2] http://www.ifd-library.org

[3] http://cic.vtt.fi/projects/ifcsvr/index_exc.html

[4] http://www.nemetschek.eu/solutions/ifc.html

[5] http://www.dds-cad.net/132x2x0.xhtml

[6] http://www.teklabimsight.com/

[7] http://bimserver.org/