

Building Information Modelling (BIM) – Versioning for collaborative design

A. JalyZada¹, W. Tizani² and A. H. Oti³

¹ PhD Candidate, Department of Civil Engineering, University of Nottingham, United Kingdom; email: evxaj2@nottingham.ac.uk

² Associate Professor, Department of Civil Engineering, University of Nottingham, United Kingdom; email: walid.tizani@nottingham.ac.uk

³ PhD Candidate, Department of Civil Engineering, University of Nottingham, United Kingdom; email: evxaho@nottingham.ac.uk

ABSTRACT

The engineering design process is a complicated activity. It is often characterized by multi-disciplinary teams in multiple places working together, on a single project, using different models and software tools. In addition, such activities generate a large amount of data that require exchange among designers and stages of the work. In industry, the current collaboration approaches often focus on integrating and managing multiple models from multi-designers. Building Information Modelling (BIM) is playing a major role in facilitating collaboration. BIM provides an opportunity to electronically model and manage the vast amount of information embedded in a building project, from conception to completion. In the building design process, changes and modifications are inevitable even in the contemporary BIM approach. Such changes need to be well managed to keep track of changes to ensure that designers have an up-to-date version of the BIM model. The main goal of this research is to develop a collaborative BIM platform that tackles the challenges of integrating object versioning, as a change management approach, and an IFC model, as data representation of BIM. This has been done through suggesting new IFC extensions to add further concepts representing the history of changing to any object of the model. It also explores possibilities of adding or merging object-based change information to existing BIM models to enable the representation of design intentions, identification of affected changes numerically and visually. A prototype system is implemented in C#, using .NET framework and Revit API platform. This paper concludes that the proposed system can contribute to improving collaboration – in terms of tracking and management of affected changes during multi-disciplinary design process.

INTRODUCTION

Collaborative building design is a comprehensive, distributed, and continuous iterative process. It is a complicated activity involving multi-disciplinary designers and experts in the AEC industry. It generates large amounts of data that require transfer among construction stakeholders and the various stages of the design process to achieve effective collaboration. Building Information Modelling (BIM) is envisaged to play an important role in the process of collaborative design. BIM entails the integral digital representation of all building information for the different stages of project development in the form of objects (Shafiq et al. 2013).

Each disciplinary designer such as the architect, structural engineer, electrical engineer, HVAC engineer, etc. usually works on a different BIM model depending on their speciality. Clearly, it is difficult to anticipate that a single application would meet all the requirements in the AEC industry. The BIM models that are used among designers are often too large, complex and shared in certain areas. Some common elements among different disciplinary designers require good management to use them properly. For example, structural elements such as beams, columns, slabs and walls are also shared by other disciplinary teams. Changing the properties of these elements by any designer need be done with the consent of others to ensure consistency (Macdonald 2013). Unavoidably, the building model keeps changing and increasing in size/complexity as accumulated information is being added to it. The process of comparing two large-scale BIM models is time-consuming and cumbersome when performed manually. It is even more difficult when the BIM models for multi-disciplinary designers are being updated frequently to meet the project requirements. Koch and Firmenich (2011) proposed a new processing-oriented building model through integrating version-oriented building information with change-oriented descriptions. This approach provides information about both states and changes in the model, but does not tackle the interoperability and sharing changes among applications. Nour (2012) addressed the problem of creating a BIM workspace for the virtual organization that consists of several stakeholders. He proposed information management system at the object level for dependency management and access rights allocation.

The challenges associated with different disciplinary BIM applications not effectively communicating with each other and the difficulties in handling successive changes in information models have warranted the need to find a solution. For the purposes of collaboration, it is essential to seamlessly transfer data between different AEC applications effectively and allow other applications to utilize the transferred data (Wang et al. 2009). Hence, interoperability of design software and standardization of building data is a major requirement for successful collaboration (Tizani and Mawdesley 2011). One of the open standards available exploitation in tackling the existing challenges of interoperability is the Industry Foundation Classes (IFC). It was specifically developed as a means to exchange model-based data among model-based applications (Steel et al. 2012); these kinds of data models may not be the best interoperability key under all circumstances and domains. The IFC EXPRESS schema is intended to describe the current information of the building (Gökçe et al. 2013). IFC has been undergoing improvements and revisions to reach a relatively advanced level to derive the most benefits for all industry stakeholders. The associated existing challenges led to a lack of adequately handling with the following:

- Changes made in the BIM and associated information
- Information sharing and changes effected among different BIM models
- Records of the history of objects in the BIM models
- Records relating to earlier versions of the BIM models.

Therefore, it is important to extend the IFCs schema so that the history of elements can be implicit within the neutral data file. The proposed extension represents the related information above as additional data structure classes in the IFC data model.

A number of modelling software companies are now developing suites of standing-alone BIM modelling related software tools. Solibri Model Checker

(SMC), Navisworks, and Tekla BIMsight applications are information modelling integrated with data modelling to mainly detect clashes and resolve conflicts among different objects for different disciplinary BIM models (Volk et al. 2014). However, they do not address changing in the shared objects among different BIM models.

This paper is structured as follows. After reviewing the state of the art in building information modelling with focus on the collaborative design system, the paper highlights current challenges with sharing, exchanging, comparing, versioning, changing and merging different information models. The remaining aspect of the paper discusses the idea of object versioning and then suggests new entities for integration into the existing data-modelling schema. Requirements of a collaborative-BIM system have been proposed to draw up a modelling framework. Based on the established framework, an implementation of the proposed system in an object environment is ongoing to develop a prototype that will meet the described collaborative design improvements. The paper ends with a conclusion section.

OBJECT VERSIONING

The AEC design process is complex in nature. It involves multidisciplinary designers that can work sequentially or in parallel, synchronous or asynchronous. The amount of available information increases with time, in terms of quality and quantity. Thus, the amount of changes in the information needs to be managed. It might be required at some point to return to an earlier state of the design, in case designers fail to reach a consensus on the model changes in the development of the design or to keep track of effected changes. Since the information model is embedded with objects, it is possible to manage changes to the information model based on object versions as the design usually progresses through several versions to arrive at final solutions. Versioning can be divided into global and local version. The global version is among interdisciplinary designers when the changes of BIM model need to be shared with the other disciplinary designers. The local version is among intra-disciplinary designers when any of the designers in one of the engineering disciplines need to save the changes of his BIM model and share it with the same disciplinary designer as shown in Figure 1. These two types of versions are not exclusive of each other, but are interrelated, and a change in one of the version type can be shared with the other.

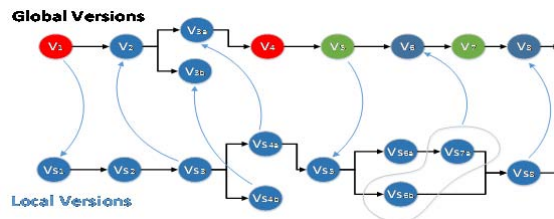


Figure 1: Proposed versioning process

Elements are represented as objects in a virtual building and building information is modelled based on object. The information and the behavior of each object differ from others in terms of the properties that represent the object. These properties can be classified according to the requirement of its object and the stability of its value. Object properties in the BIM come from a variety of

sources. It covers more than just geometrical part of the BIM object. It also includes specification, quantities, analytical parts, and some extra parts such as cost and sustainability information. Therefore, the properties of objects in a model can be demonstrated as an open matrix with variable numbers of objects and variable number of properties for each object, (for instance, the number of column's properties is different from slab's properties).

Versioning the object makes it possible to have multiple versions of the properties of each object. Any change in the object represents a change in the overall BIM model. The new model version can be saved by only selecting the changing with the properties of each object. A version history of the objects can be provided as shown in Figure 2a, to represent the development of design processes and to demonstrate the sequence of versioning process in terms of adding new objects and modifying or deleting existing objects. Each object in each version has a repository about the embedded data and information that is not necessary to be filled in each new version as illustrated in repository properties in figure 2b, which represent the properties of some objects in some versions extracted from figure 2a. The repository of each version represents all the changing in the properties of this object. From the figure, it is better to navigate through the details of the object from each version and acquire complete information about the properties of each object. For the modified objects, any new version is dealing with the property changes that only represented in the repository while full properties of the new object are presented in that version.

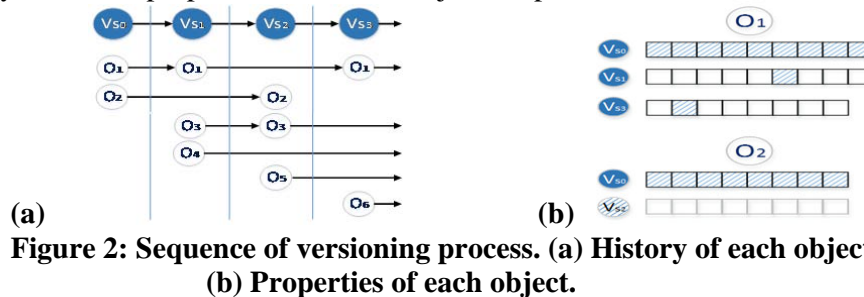


Figure 2: Sequence of versioning process. (a) History of each object. (b) Properties of each object.

Suggested data model Schema Extensions. The existing data model schema may not be the best representation of the building data from the conception to completion. A major challenge facing an IFC EXPRESS schema is oriented towards defining the existing objects and the relationships between objects of the information models (JalyZada and Tizani 2013). Therefore, an extension to the capability of the IFC schema has been suggested to add further concepts to the IFC standard. These new definitions represent the history of changes on any object of the model to get integration with all earlier versions of BIM models and for communication among multi-disciplinary designers. To classify the basic content of an IFC-based data model, all needed components clarified precisely, and the structures of all required and related entities, attributes, and relationships of objects in the IFC schema studied correctly to support the identified requirements. Six existing entities from the IFC standard have been suggested to be modified to represent as new entities within the IFC schema to support the idea of object versioning that holds the history of changes to objects of the BIM model. Three of these new entities represent detailed information about the changes that have occurred in the current BIM version and the other three represent the collector to all instances of entities. The data structure of each new

entity has been formulated based on the EXPRESS language definitions and classifications. The definition of the new suggested types of entities is:

- *IfcVersionPropertySingleValue*: defines a particular property type for the modified object.
- *IfcVersionPropertySet*: collects all modifying properties for an object.
- *IfcVersionBuildingElement*: comprises a modified object, this entity will be used when any of the properties of this object has been modified.
- *IfcVersionBuildingElementSet*: connects all modified objects for one version.
- *IfcVersionInformation*: refers to full information about any version of the BIM model.
- *IfcVersionSet*: collects all versions of the prototype model to the current version.

The architecture of each new entity has been suggested as a class following the same IFC Express definitions. The new data structure is needed to integrate and attach to the existing data-modelling schema to document all changing elements of that version and to manage all new classes that have been taken place to be represented in the IFC format. Each modified object could have a version class attach to the IFC standard. Whenever the object changes, even if it is a single property, a new version class representing that change could attach to the data model. Each discipline has his IFC file, which is different in terms of the representation of information. While, in terms of the structure of the IFC file, all IFC files are following the same data modelling language for its data definition and specification. The new classes for versioning should follow the same unified approach. Figure 3 shows the classification tree of the new entities suggested within the IFC standard. For each element, there are more than instances of *IfcVersionPropertySingleValue* entity to represent all modified properties of that object. Thus, for each version, there are more than instances of *IfcVersionBuildingElement* entity to represent all modified elements of that version. And so on for all versions, there are more than instances of *IfcVersionInformation* entity to represent all modified versions in the unique IFC file.

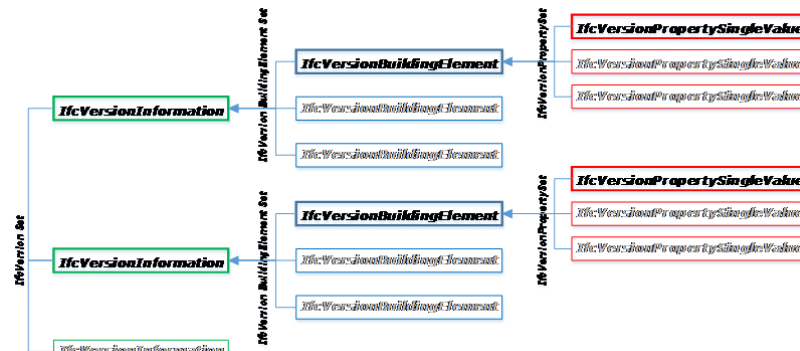


Figure 3: classification tree of the new entities

COLLABORATIVE BIM-FRAMEWORK

The proposed collaborative-BIM framework has been developed to reflect all the requirements for modelling a collaborative design process. It is essential that the proposed framework for collaborative multi-information models should, as much as possible, cover the economic, technological, and practical

aspects in the design process. It should also satisfy the requirement of inter, and intra-disciplinary designer teams, the compatible with BIM models, managing changes to BIM models and support the collaborative environment. The suggested collaborative framework solution can be represented in stages:

- Creating an extended data model file: This is possible through one of the disciplinary designers by considering the effected changes in object versioning management of the BIM model.
- Storing the extended data model file: This is related to change management in a centralized server to share, exchange and store BIM versioning information.
- Reading the extended data model file: This applies to all disciplinary designers to effect changes in building model. The designers can handle with object versioning and reflect with all object changes numerically and graphically.
- Communication among multi-disciplinary designers: This is related to managing the communication among all designers for the purposes of version control and to respond to changes made in the BIM model.

The proposed collaborative-BIM system supports and manages the processes associated with changes effected in the BIM data by the sender and receiver. It allows for the traceability of effected changes in the design process. Changes or modifications in BIM models are very common and likely to occur from different sources, for various reasons, at any stage of a project. This may have considerable negative impacts on the project delivery. The proposed approach can automate and coordinate the design changes and identify the effected changes in each BIM model numerically and visually.

IMPLEMENTATING THE COLLABORATIVE BIM-SYSTEM

A prototype collaborative-BIM system is being implemented using object-oriented programming (OOP) within the .NET framework environment in accordance with the proposed framework. The prototype is integrated as an external application into Revit 2013 through Revit Platform application programming interface (API) which allows interfacing with programming languages that are .NET compliant. The programming language used is C#. It is a multi-paradigm programming language encompassing object-oriented and component-oriented programming (Deitel et al., 2013). Although it is possible to link the collaborative model to any object-oriented building modelling platform, Revit is imbedded with an open software development kit (SDK) suitable for such implementation.

The main goal of the implementation is to develop a collaborative-BIM tool to share and exchange changing in BIM models among multi-disciplinary designers in a collaborative environment. Current aspects covered include:

- To read, analyze and compare the properties of each object in different BIM models and identify any new, deleted and modified elements in each building model
- To study the asynchronous collaboration in BIM models at different time and in different places.
- To study the synchronous collaboration for each object in the new BIM version with the current BIM model for different disciplinary designers at the same time.

- To convert all changes in the BIM model to a data modelling language and structure all suggested IFC schema extensions in a coding form.
- To attach and merge these extra new instances into the new IFC file.
- To locate what has been changed numerically, and show how the objects in BIM have been manipulated graphically.

The process begins by calling up the collaborative-BIM program from Revit 2013 environment where this application has been integrated through Revit Platform API. Managing all suggested entities and merging all new instances into IFC file have been carried out through developing a tool linked to the Collaborative-BIM program. The program captures relevant changes in the information model and represents this information as IFC entities. The representations are in the required code format that can create the proposed attachment to the IFC data schema. The process involves the creation of new instances that follow a consistent set of rules, complying with the format and behavior of the EXPRESS data modelling.

In a collaborative BIM system, multi-disciplinary designers are working on their different BIM models. Each designer is instantaneously notified every time changes are made. This is important as even very small changes can have an impact on their local model. In each BIM model, the system can identify and keep track of all changes from the extended IFC file. The system can recognize them numerically and graphically by highlighting the modified elements, drawing the new or identifying the delete elements as shown from the result output of the collaboration program (Figure 4). The tool can be called up in Revit in two ways depending on the user's intentions. Either, the user selects certain elements of the model to obtain their information or the whole structure could be selected and the collaborative tool filters the relevant elements for further action to obtain the changes associated with elements of the building. The operation of the collaborative model is continuously undergoing improvement. More details are being added to the model and tests are performed periodically to the program components in order to increase the level of maturity of the implementation.

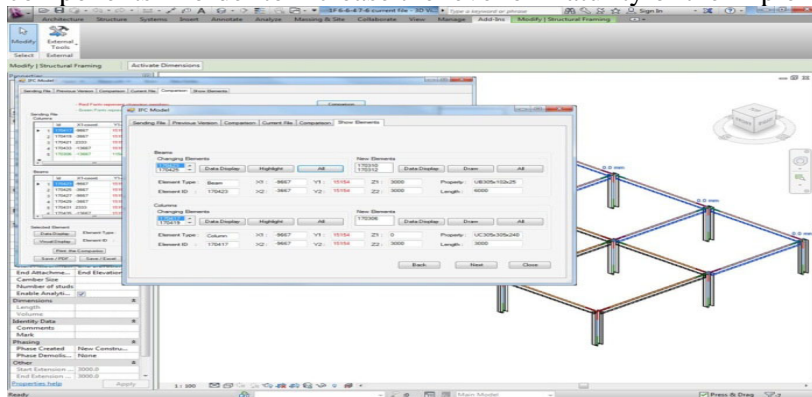


Figure 4: Output file of the Collaborative-BIM program

CONCLUSION

This paper presents the research on the investigation the collaborative design process concerning the processes of storing, managing, versioning, exchanging and sharing of building information in an interoperable and reusable way. It proposes on the backdrop that utilizing building information models to build an integrated and collaborative engineering information environment can contribute to supporting the design process. A number of new information

modelling entities are proposed as an extension to the current IFC standard. They provide consistency and robustness for all new changes, all earlier versions, and to the modified neutral data model. The new entities have been suggested can be read by other software products. Moreover, it supports versioning in file exchanges and visualization of changes. An automated approach for model matching has been presented. This approach leverages the collaborative work among BIM models and upgrades those matching when a new version of a target model is released. New dimensions to the synchronous designing process has been added to the adopted Collaborative –BIM system to support the designers with storing and sharing multi-versions while working simultaneously on the same or different information model.

ACKNOWLEDGEMENTS

This work is sponsored by the KRG-HCDP Scholarship program / Ministry of Higher Education and Scientific Research / Kurdistan Regional Government.

REFERENCES

- DEITEL, P. J., DEITEL, H. M. and DEITEL, A. 2013. *C# 2012 for Programmers*, Pearson Education.
- GÖKÇE, K. U., GÖKÇE, H. U. and KATRANUSCHKOV, P. 2013. IFC-Based Product Catalog Formalization for Software Interoperability in the Construction Management Domain. *Journal of computing in civil engineering*, 27, 36-50.
- JALYZADA, A. and TIZANI, W. 2013. Collaboration between Architectural and Structural Models. *11th. International Postgraduate Research Conference (IPGRC13)*. University of Salford / Manchester.
- KOCH, C. and FIRMENICH, B. 2011. An approach to distributed building modeling on the basis of versions and changes. *Advanced Engineering Informatics*, 25, 297-310.
- MACDONALD, J. A. G., MARKO 2013. Multidisciplinary AEC Education Utilising BIM / PLIM Tools and Processes. In: KTH, S. O. A. A. T. B. E. A. (ed.) *PLM International Conference (PLM IC), PLM 13*. Nantes, France, 6th-10th July 2013.
- NOUR, M. 2012. BIM support for design change management and workflow approval In: ISCCBE (ed.) *14th International Conference on Computing in civil and Building Engineering*. Moscow , Russia
- SHAFIQ, M. T., MATTHEWS, J. and LOCKLEY, S. 2013. A study of BIM collaboration requirements and available features in existing model collaboration systems. *Journal of Information Technology in Construction (ITcon)*, 18, 148-161.
- STEEL, J., DROGEMULLER, R. and TOTH, B. 2012. Model interoperability in building information modelling. *Software and Systems Modeling*, 11, 99-109.
- TIZANI, W. and MAWDESLEY, M. J. 2011. Advances and challenges in computing in civil and building engineering. *Advanced Engineering Informatics*, 25, 569-572.
- VOLK, R., STENGEL, J. and SCHULTMANN, F. 2014. Building Information Modeling (BIM) for existing buildings—Literature review and future needs. *Automation in Construction*, 38, 109-127.
- WANG, H., AKINCI, B., GARRETT JR, J. H., NYBERG, E. and REED, K. A. 2009. Semi-automated model matching using version difference. *Advanced Engineering Informatics*, 23, 1-11.