Automation of software independent data interpretation between architectural and structural analysis models

Goran Sibenik[1*] and Iva Kovacic[1],
[1]Institute for Interdisciplinary Building Process Management, TU Wien, Austria
*email: goran.sibenik@tuwien.ac.at

Abstract

In AEC practices multifold digital models exist to support various design and analysis tasks (e.g. architectural or HVAC design, structural or thermal analysis). Information models need to satisfy the particular and varying domain-specific needs. Despite the great efforts to create open standards for data exchange, digital data exchange between domains is still burdened with numerous difficulties, such as loss or wrong interpretation of information. The seamless data exchange between actors in the process would reduce communication-based errors and planning time, and it is a requirement for achieving a full potential of BIM.

This work focuses on the automation of interpretation steps between architectural and structural analysis models with intermediary data available in software independent storage. Previously conducted research identified the need for standardisation of interpretation in order to improve the software-independent data exchange.

Thereby we first review relevant software tools, standards and data exchange processes, where the architectural and structural analysis representations were analysed and the relations between them were identified.

Next, a new framework for data handling in the AEC planning process was created. The data-handling framework is based on domain-specific classification, interpretation and automation procedures that can support interdisciplinary exchange. Classification is needed for systematic arrangement of data in groups or categories according to established criteria, so the data would remain machine readable, and it is based on the IFC taxonomy. Interpretation between architectural and structural analysis models was proposed based on the previously conducted state-of-the-art review. The proposed domain-specific classification and interpretation procedures are developed for the exchange between architecture and structural engineering domains. However, the proposed central data handling framework is generic, so it could be used for other domains in the AEC industry as well. Finally, in the automation procedure, the framework is implemented with the system architecture that can support it. The data is available on a software independent storage, and the domain specific model communicates with the software independent storage. The interpretation is automated with the help of a programming language.

The innovative contribution are primarily the procedural steps defining the interpretation, followed by the proposed framework for the data exchange involving domain-specific classification and interpretation procedures, and finally the automation where a semi-structured database is used for the data storage and interpretations are automated with a programming language. Automating the central storage-based interpretation steps is a crucial feature of a new framework allowing communication between various domain-specific models within software independent storage.

Keywords: data exchange, interpretation steps, automation, structural analysis, open standards

1. Introduction

The architecture, engineering and construction (AEC) industry aims on improving the planning processes with the digitalisation of workflows. One of the key features of a digital workflow is the model-based communication and the seamless data exchange. Structural analysis computes the behaviour of building elements in the most unfavourable situation of load cases. The computations take place with the idealized building elements: punctual, linear and planar elements. Information about building elements originate from the architectural design which is usually a preceding step for structural analysis. The data exchange between architectural and structural analysis models is still done via physical documents (paper), digitised documents (pdf files) or files containing 2D geometry (dwg drawings), while the 3D model-based exchange takes place only in isolated cases and intrafirm workflows. If a 3D building model is created from 2D documents, the geometry needs to be remodelled in a 3D structural analysis tools, which is an expensive, redundant and erroneous work. However, the industry still fails to achieve a model-based exchange which would be trustworthy enough for the end users.

The model-based exchange gains on popularity for its potentials in improving the planning workflows. Depending on the existence of an accessible and modifiable intermediate building data model, two types of data exchange workflows exist: open and closed BIM (Building Information Modelling). Closed BIM is referred to as a workflow where the data exchange takes place with software to software interfaces; Open BIM approach refers to a workflow where an open, non-proprietary format is used for the data exchange purposes. The most widely spread and implemented open standard within the software tools intended for this use is the Industry Foundation Classes (IFC) standard developed by buildingSMART (buildingSMART, 2019). Researchers investigating the data exchange between architecture and structural analysis are focused on the open exchange, especially the IFC standard (e.g. Romberg et al., 2004; Ramaji & Memari, 2018).

In our previous research, the analysis of the open data exchange in practice showed that a crucial requirement for a systematic improvement is interpretation rules between different domains (Sibenik & Kovacic, 2019). These rules are not straightforwardly defined in the standards. The open data exchange framework needs to be improved, so the interpretation rules would become a part it. The analysis showed that a single integrated model supporting various AEC domains does not suffice and the AEC industry is stepping towards multiple domain specific data models and automated or semi-automated interpretation between them. The interpretation steps need to describe the most repetitive tasks in such a way that they could be automated, and at the same time understandable and open to an end user. This paper aims at describing possible interpretation steps between the architectural and structural analysis models.

This work is structured as following: in Section 2 the research background regarding the interpretations between the architectural and structural analysis domains will be explained. In Section 3 we will provide a review of the state of the art of existing software tools, standards and the data exchange workflow. In Section 4 a proposed framework that includes classification, interpretation and automation procedures will be presented. This paper addresses primarily the interpretations, which will be proposed in Section 5. Section 6 will describe a test system architecture, while we conclude with Section 7 with the summary of framework properties and the future outlook.

2. Background

The open model-based exchange between architecture and structural analysis rarely takes place in practice, since it still does not meet the end user requirements. There are several framework proposals using the IFC building data model as a starting point, which is interpreted to a structural analysis model (Deng & Chang, 2006; Hu et al., 2016; Liu et al., 2010; Qin et al., 2011; Ramaji & Memari, 2018; Wang et al., 2013; Zhang et al., 2014). They document significant differences between the architectural and structural analysis models, providing often software tool-related problem-solutions. The problems were addressed on a practical level and while the software-specific solutions give directions for the software-independent framework improvement, they still cannot be generally applied in the AEC industry as such. Various changes in the currently established practices have to be made in order to achieve a working data exchange. Interpretation steps are described based on the input model, focusing mostly on a specific IFC export and creating a new structural analysis model from it. The frameworks using the IFC building data model as a starting point are still software tool-specific because of the IFC schema redundancy. The conceptual differences between the models and the interpretation steps which take place between them are defined for the research purposes and not extensively investigated.

Some researchers question the use of the central integrated model, and pursue the use of multiple models. Rezgui et al. (2011) and Rezgui et al. (2010) propose a process driven framework, which is web-based and supported by ontologies; eventually leading to knowledge value creation from knowledge sharing. Lee et al. (2014) describe a web-based framework involving domain-specific filters as intermediary step between the complex central model and the domain-specific models; stepping away from the concept of integrated model. In our previous research, the necessity to separate multiple domain-specific building data models has been recognized, whereby the relations between multiple models need to be retained and originate from the inter-domain conceptualisation (Sibenik & Kovacic, 2019).

Another problem that we analysed is the data structure in the open central storage: the necessity to separately deal with the geometrical and non-geometrical information was recognised. In most of the proposed frameworks (Deng & Chang, 2006; Lee & Jeong, 2012; Liu et al., 2010; Ramaji & Memari, 2018) both types of information are treated similarly, often focusing on one and applying the same framework to the other, sometimes reflecting the storage structure of structural analysis software tools. Wan et al. (2004) define geometry, loads, materials, member sections and other as separate information gaps, Hu et al. (2016) geometrical, property and control information as parts of the data management layer, and Lee & Jeong (2012) distinct geometric and non-geometric information in the object level of their filter mechanism, but not focusing on geometric interpretations. On the other hand, the ontology, taxonomy and classification related work, is mostly dealing with the non-geometrical information (Pauwels et al., 2017). In order to achieve a seamless communication we propose to use two different approaches for managing geometrical and non-geometrical information. Exchange of geometrical information is dependent on geometrical kernels in the interoperating software tools, interpretations steps and the geometrical definitions supported by the mediatory open standard. The interpretation of the semantic and syntactic description of the building information models does not have much to do with the geometrical data interpretations. Besides that, geometric transformations of building models involve complex geometrical methods, and some researchers already suggested to introduce a geometry kernel in the intermediary translational steps (e.g. Mora et al., 2008; Romberg et al., 2004). However the integral approach which can process both geometrical and non-geometrical information is necessary in the AEC industry.

As already mentioned, the trends in the academic community supports multiple domain-specific models and the interpretation between them, sometimes referred to as filters or pre-processing steps. Although in the analysed papers the interpretations are part of the workflows, the data interpretation steps are simplified and not sufficiently documented; focus either on geometrical or non-geometrical interpretation. The interpretation steps are usually described as the generally accepted truth, although they are based on the intuitive, and commonly software tool or regionally specific workflows. For the automation purposes it is necessary to define the tolerances, borderline cases and all interpretation steps in such a way that the models can be machine processable, and the interpretations understandable by the end users. Because our reasoning brings us to the unavoidable automation of repetitive tasks during the model exchange, with this research we aim to contribute to the body of knowledge that bridges the interdisciplinary differences between architectural and structural analysis models, focusing on the geometric interpretations, but also considering the non-geometric interpretations.

The main research question this paper addresses is: **how to efficiently support the end user needs within the exchange between architectural and structural analysis tools**. We aim to answer this question by providing a data exchange framework which focuses on two procedures: domain-specific classification and interpretation between the domains, implemented using a software independent central storage and processing. In this paper we will focus on the interpretation steps.

3. State of the art

3.1 Software tools

The taxonomy and classes present in software tools are compared in Sibenik & Kovacic (2019). The paper concludes that the objects available in software tools show significant similarities if the tools support a same domain. For example, architectural and structural design software tools (Revit and Allplan) as well as the software for architectural design (Archicad) provide similar classes for the description of column, wall and slab. On the other hand, structural analysis software tools represent these elements as linear (column) or planar (wall and slab). These classes represent same real-world building elements, but they differ between the domains to which they belong. In the planning workflow it is necessary to transfer this information between the domains.

Ramaji & Memari (2018) list three possibilities to create a structural analysis model from a native BIM model: 1. Structural analysis model is created in a native software tool; 2. Structural analysis model is created from a native model in a structural analysis tool; 3. An additional software tool (or a plug-in) is used to generate the structural analysis model. This classification is based on the primary role of the software tool that generates the structural analysis model: architectural tool, structural analysis tool, or a third-party data exchange tool. The interpretations taking place are not open to the end user and generally work as a "black-box" scenario (Holzer et al., 2007). This means that the processes defined within the software tools are not explained to or editable by the end user, and the interventions are limited to the methods defined by the software industry. The lack of interpretation transparency and the lack of support for all necessary user interventions during the exchange process discourages the end users to allow for the automatic exchange.

3.2 Standards

The three building elements (column, wall, slab) are not sufficiently precisely described in the standards so the conversion between their architectural and structural classes could be automated. In fact, the majority of building elements are not sufficiently standardised due to the heterogeneity of the industry and the lack of consideration of automation possibilities. However, there are recommendations for structural engineers (Pech & Kolbitsch, 2005; Schuler, 2016) which suggest how to treat the architectural elements when interpreting them for structural analysis. Standards provide little or no information about the rules of how the elements are interpreted between the domains (e.g. obsolete DIN, 2002).

The most widely implemented standard for the digital exchange between the two domains is the IFC schema. Early works on the schema (Weise et al., 2003) introduced the majority of the structural analysis concepts and their corresponding properties. Current version of IFC schema supports all three analysed elements, both as architectural and structural analysis domain entities. Therefore, a building element instance can occur in a single model multiple times - as structural and architectural element, without any interdependency between the two representations. Relations between the element representations do not exist and they are not included on other levels of the IFC-based data exchange framework like Information Delivery Manuals (IDM) or multiple Model View Definitions (MVD). The main problem of the standard for an interdisciplinary exchange is identified as the missing relations between the elements (Figure 1).

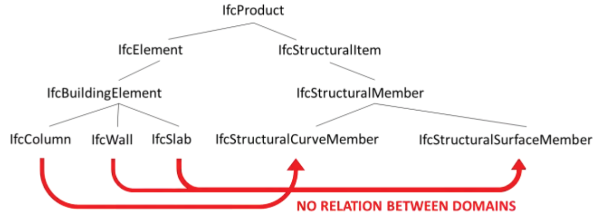


Figure 1 Extract from the IFC 2x3 schema

3.3 Data exchange

In practice, the data exchange process taking place between the architectural and structural analysis models takes place primarily with 2D architectural plans, by remodelling them in the structural analysis software tools. In this work we focus on the 3D model-based open data exchange which still has multiple problems in practice (Sibenik, 2016). We will focus on the end-user need to access and edit not only the building data model, but also the interpretation processes that take place before the import to the structural analysis software.

In practice, the model-based exchange workflows established within the company (intracompany) sometimes have positive outcomes. On the other side, projects based on one-time stakeholder setups rarely achieve a successful model-based exchange. The success is commonly determined by the experience of practitioners with the software tools, and the software tool interoperability. Closed, underlying interpretations within software tools are understood with time, and the successful exchange workflows are developed by adjusting the original workflows to the software performance. However, achieving a successful exchange in this way does not satisfy the majority of end users because it requires adapting to non-intuitional design modelling practices and uncontrolled exchange processes. End users need a fully transparent exchange framework, with transparent model data and interpretation steps to satisfy their design needs.

4. Framework

The proposed framework defines three mandatory procedures of the data exchange process: classification, interpretation and automation (Figure 2).

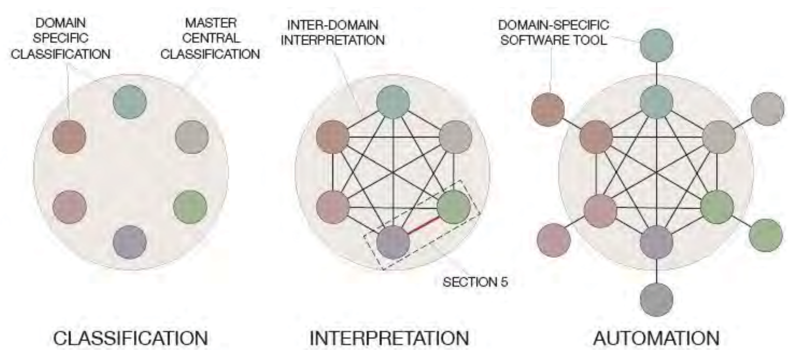


Figure 2 Framework overview

These terms will be explained more thoroughly in the following:

Classification: a data exchange procedure where the systems of terms are developed for all partaking domains and master central classification as their union. These systems include all building elements, attributes, properties and containers used in the exchange processes. Terms originate from the concept system that encompasses the whole AEC industry. The concept system is the knowledge domain that is not limited to a single domain. Different terms may represent the same element, however not within the same domain. The relations between the elements need to be based on the interdisciplinary knowledge. Eventually, the exchange requirements of all domains partaking in the planning process should be supported.

Technological solution on how the classification systems are created and maintained is not the main topic of this research. In this work, the classification systems are defined with the semi-structured database MongoDB, leaving sufficient flexibility for extensions. Current strivings in technological implementation of the AEC classification systems and the relationships between the terms are focused on the ontologies. Proposals involve transition from the monolithic integrated building data models to multiple ontology-based graph data models (Lee et al., 2014; Pauwels et al., 2017; Rezgui et al., 2011).

This approach will be further investigated; however, it is still necessary to facilitate the interpretation of both geometrical and non-geometrical data. The non-geometrical data defined through ontologies has advantages that many interpretations could be defined within the ontology. However, the geometrical interpretations still cannot be defined solely with the ontological relations.

We propose a consistent and clearly structured geometrical representation on the central storage defining every element having the geometrical properties. This has not been possible so far since the practices with the IFC are defined with the STEP (STandard for the Exchange of Product model data), where the geometry parameters are highly interrelated within a schema and the geometry manipulations by the end users are not supported. Therefore, we propose the use of geometry kernels on the central storage to define and store geometrical information required for the exchange. Achieving the interoperability through geometry kernels would ease the development of intermediary software tools.

Interpretation: in this data exchange procedure, the differences between the centrally available information and the domain-specific models are overcome; the domain-specific model is prepared for the import in the software tool. The interpretations might differ depending on the domain and the central model (model which is used as a source of information). The standards are heterogeneous on national and even company levels, and they do not suffice for the automation of interpretation steps between the domains. In the next section we will describe a framework that could be used for the open interdisciplinary interpretation. The proposed interpretation framework consists of validation, filtering, non-geometrical interpretations, geometrical operations, enrichment and reasoning operations.

Automation: this procedure is the technological implementation of the two previously described procedures with the appropriate system architecture. In this way, classification and interpretation are implemented with the software tools, and the central open system is coupled with the AEC software tools (mapping). The classification and interpretation are transparent, in order to support the necessary changes and the heterogeneous workflows, and also to leave the insight and control in the whole exchange process to the end user. The interfaces and the automated methods need to provide a certain amount of flexibility because of the varying workflows and processes in the industry. The main advantage of the centrally interpreted domain-specific models is that the mapping processes between a domain-specific open model and a native building model are reduced to establishing relationships with already prepared domain specific data, and free software developers from complex or reasoning tasks.

In the next section we will focus on the interpretation procedural steps and geometric interpretation methods that are part of the interpretation framework.

5. Interpretation Framework

5.1 Procedural Steps

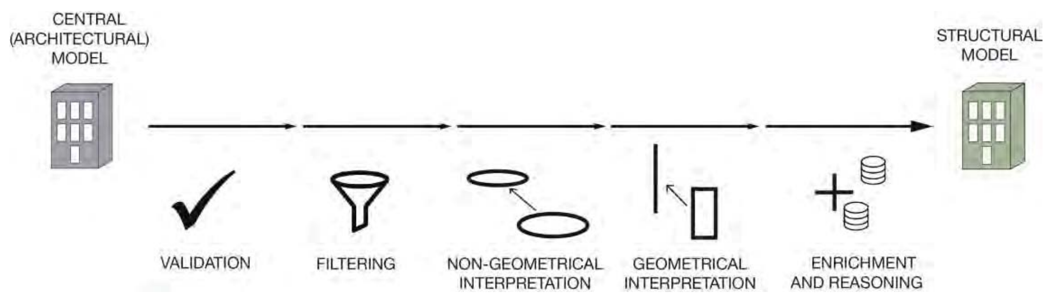


Figure 3 Procedural steps for interpretation

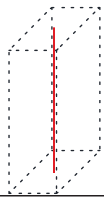
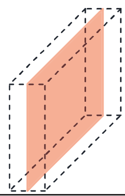
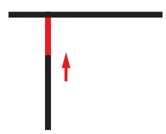
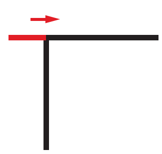
The interpretation procedure to define a structural model from a central data model involves the following procedural steps (Figure 3):

- **Validation:** Identifying the terms available in the central data model, if the elements of the model contain all necessary properties, relations, etc. to allow for the creation of the structural model. The validation is performed on the specific elements or the group of elements within the central model. The validation steps can be so implemented that they automatically report the inconsistencies to the stakeholder who generated specific information. A geometrical element validation can also be introduced. In the case of linear and planar elements, linear elements are considered the ones where two dimensions are considerably larger than the third one ($w<<l$) and ($h<<l$) (Schuler 2016). Further on, a planar element has one dimension considerably smaller than the other two ($t<<l_1$, l_2). The difference between the column and the wall is described as the ($h/b>5$) (Pech & Kolbitsch, 2005) in the case of the concrete and reinforced concrete walls.
- **Filtering:** the elements and properties identified in the validation step on the central storage are reduced to the information required for the next domain-specific planning task. In the case of structural analysis, the load-bearing properties of building elements have been analysed to isolate only the building elements relevant for structural analysis.
- **Non-geometrical interpretation:** Mostly ontological data interpretation. We observed three building elements: wall, column and slab. Since the original model is the architectural model, the 3D physical representation was considered as a starting definition of the element. The structural analysis building elements classes can be identified solely based on the available architectural classification: column is a linear element; wall is a planar element and slab is a planar element.
- **Geometrical interpretation:** geometrical interpretation for this data exchange are complex and described in the next part of this section.
- **Enrichment and reasoning:** adding new information based on the existing ones by using the external databases or methods which can enrich the existing elements with additional knowledge, for instance recognition of the architectural and assigning a structural material.

5.2 Geometric Interpretation Methods

The following geometry interpretation methods are proposed based on the practices employed by structural engineers, modelling instructions and the state-of-the-art review.

Table 1 Geometrical interpretation methods

Method	Description	
Linearisation		Architectural columns are interpreted as linear elements. In this method the central points defining lowest and highest surface of the element are connected and the linear element is created. The cross section information becomes a property of the linear analytical element.
Planarisation		Filtered planar elements are identified, the largest surfaces are identified and placed in the central position between them. The thickness information becomes a non-geometrical property of the planar analytical element.
Vertical connectivity adjustment		Vertical connectivity adjustment: in order to be able to transfer the forces, the horizontal elements must be connected to the vertical ones transferring the forces to the ground. Therefore, the linear and planar elements are tested for their connectivity with the native horizontal 3D elements. Tolerances may be introduced as well. The connected elements are extended to the corresponding interpreted horizontal elements.
Horizontal connectivity adjustment		a) Vertical planar elements need to be checked for horizontal connections in the physical 3D model, then connected in the analytical model. Horizontal connections of vertical linear elements are important for the next step of horizontal element adjustment. b) The edges of the planar horizontal elements need to be aligned with the vertical linear and planar elements. First the vertical elements (or connections of linear vertical elements) located under the horizontal element closest to the edges need to be identified. After testing if the distances are in the tolerance scale the edges are aligned to the vertical elements.

6. System Architecture Proposal

The above described framework has not yet been implemented in any available software tool or the combination of software tools in such a way that the interpretations are open to the end user. Therefore, the implementation of the proposed framework took place by developing a new system architecture. In order to implement the previously defined procedures, a combination of software tools has been established. First, a central storage database was chosen, which replaced the file-based building data, so a synchronous exchange could be achieved. For the data storage a semi-structured database MongoDB was chosen. The advantage of the semi-structured database to a relational database is the flexibility it provides to support unexpected information (Lee et al., 2014; Rasys et al., 2014). The unpredictable behaviour of the end users, the domains that are part of the planning process, as well as the software tools which can be involved differ from project to project (Holzer, 2007) make the flexibility an important factor when choosing a database.

The interpretation framework was implemented with C#, but the chosen semi-structured database can be accessed with other programing languages. The validation, filtering, geometrical and non-geometrical interpretation and enrichment procedural steps can all be practically realized with C#. Mapping processes depend on the API (Application Programming Interface) of the importing software tool, and in the case of Dlubal RFEM, the mapping processes are also supported by C#. For the geometrical interpretation an open geometry kernel OpenCascade was used. In this way, the complex geometry editing methods are simplified with the predefined options provided by the OpenCascade kernel. OpenCascade was used with C# Wrapper, because it is a kernel whose methods are C++ based.

A test workflow (Figure 4) showed promising results for the wider implementation. The starting point was the prepared architectural model on the semi-structured database. The architectural model is created from the Revit IFC export, and converted to JSON (JavaScript Object Notation) format to serve the new workflow. The validation step was not performed since the IFC model was converted and filtered to the desired model in JSON format, providing a valid starting open architectural model for further actions on central database. The interpretation followed the creation of the central open model and the structural analysis model was created on the central database. The process is fully automated, involving geometric and non-geometric information, and enrichment and reasoning. However, it is necessary to support more user-friendly end-user interface in the future. The created model is then mapped to the structural analysis software tool for test purposes.

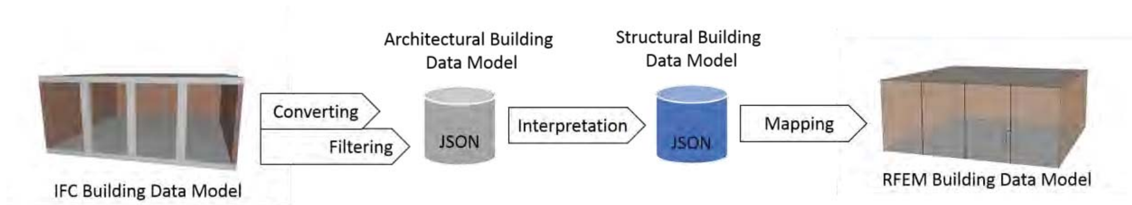


Figure 4 Interpretation framework testing methodology with screenshots

7. Conclusion

This paper describes the proposal of a new framework for overcoming the semantic and geometric differences between the architectural and structural analysis models. Three procedures are recognised as the requirements of an automated exchange framework: classification, interpretation and automation. The focus of the paper is the interpretation procedure taking place on the central data storage.

Interpretation taking place between architectural and structural analysis models can be divided into five procedural steps: validation of the central model, filtering of required elements, interpretation of geometrical and non-geometrical elements and the enrichment and reasoning. The central building model needs to support all of these steps in order to achieve a seamless exchange. A precondition for it is a domain-specific classification.

The greatest challenge for the data exchange between architectural and structural analysis models are the geometrical interpretations. They consist of reduction of dimensionality of linear elements, reduction of dimensionality of planar element, adjustment of connectivity of vertical elements and adjustment of connectivity of horizontal elements. These proposed methods cover some of the most repetitive tasks when the structural model is created, however they do not suffice for all building elements, geometries or materials and would have to be extended for more complex building models.

The test case is limited to a simple building model. The interpretation tasks are implemented with the focus on geometry interpretation methods and used with a test model. System architecture consists of a semi-structured database for central storage, programing language and geometry kernel. The framework implementation should be improved with a more user-friendly interface and allow for more flexibility. The system architecture proposal leaves the interpretation as well as central model open for the end user. The following technical improvements could be considered in the future: the semi-structured database could be replaced with graph database, other programming languages or the geometry kernel with additional methods and geometry definitions could be used. The technical optimisation is not the topic of the work and will be considered in the future research.

Next steps will involve the proof of concept testing that will encompass additional building elements and extend the existing interpretations methods. The technical improvements of system architecture are going to be considered as well.

Acknowledgements

This work was conducted as part of the Doctoral College "Computational Design" on TU Wien (TUW) and the project DATAFILTER in cooperation with Strabag AG. The authors wish to express their appreciation to Dr. Wendelin Sprenger and Theodor Sansakrit Strohal for their support and contribution to the study on the side of Strabag AG.

References

buildingSMART (2019) *International home of openBIM*. Retrieved from http://www.buildingsmart-tech.org

Deng, X. Y., & Chang, T.-Y. P. (2006, June). *Creating Structural Model from IFC-based Architectural Model*. Paper presented at the Joint International Conference on Computing and Decision Making in Civil and Building Engineering, Montréal, Canada. Retrieved from http://itc.scix.net/data/works/att/w78-2006-tf577.pdf

DIN - Deutsches Institut für Normung. (2002). *Tragwerke aus Beton, Stahlbeton und Spannbeton - Teil 1: Bemessung und Konstruktion* (DIN 1045-1:2002-07). Retrieved from https://www.umwelt-online.de/recht/bau/din/1045_1ges.htm

Holzer D., Tengono Y., & Downing S. (2007). Developing a Framework for Linking Design Intelligence from Multiple Professions in the AEC Industry. In: Dong A., Moere A.V., & Gero J.S. (Eds.), *Computer-Aided Architectural Design Futures (CAADFutures) 2007* (pp. 303-316). Dordrecht, Netherlands: Springer.

Hu, Z. Z., Zhang, X. Y., Wang, H. W., & Kassem, M. (2016). Improving interoperability between architectural and structural design models: An industry foundation classes based approach with web-based tools. *Automation in Construction, 66*, 29-42. https://doi.org/10.1016/j.autcon.2016.02.001

Lee, J., & Jeong, Y. (2012). User-centric knowledge representations based on ontology for AEC design collaboration. *Computer-Aided Design, 44,* 735-748. doi:10.1016/j.cad.2012.03.011

Lee, J., Jeong, Y., Oh, M., & Hong, S.W. (2014). A Filter-Mediated Communication Model for Design Collaboration in Building Construction. *The Scientific World Journal, 2014*. http://dx.doi.org/10.1155/2014/808613

Liu, Z.-Q., Li, Y.-G., & Zhang, H.-I. (2010). IFC-based integration tool for supporting information exchange from architectural model to structural model, *Journal of Central South University of Technology, 17 (no. 6)*, 1344-1350. https://doi.org/10.1007/s11771-010-0640-z

Mora, R., Bedard, C., & Rivard, H. (2008). A geometric modelling framework for conceptual structural design from early digital architectural models. *Advanced Engineering Informatics, 22*, 254-270. doi:10.1016/j.aei.2007.03.003

Pauwels, P., Zhang, S., & Lee, Y.-C. (2017) Semantic web technologies in AEC industry: A literatur overview. *Automation in Construction, 73*, 145-165. http://dx.doi.org/10.1016/j.autcon.2016.10.003

Pech, A., & Kolbitsch, A. (2005). *Wände*. Wien, Austria: Springer-Verlag Wien

Qin, L., Deng, X.-Y., & Liu, X.-L. (2011). Industry foundation classes based integration of architectural design and structural analysis. *Journal of Shanghai Jiaotong University (Science)*, *16 (no. 1)*, 83-90. https://doi.org/10.1007/s12204-011-1099-2

Ramaji, I.J. & Memari, A.M. (2018). Interpretation of structural analytical models from the coordination view in building information models. *Automation in Construction, 90*, 117-133 https://doi.org/10.1016/j.autcon.2018.02.025

Rasys, E., Hodds, M., Dawood, N. & Kassem, M. (2014). A Web3d Enabled Information Integration Framework for Facility Management, *Australasian Journal of Construction Economics and Building Conference Series, 2(no. 1)*, 1-12. https://doi.org/10.5130/ajceb-cs.v2i1.3763

Rezgui, Y., Boddy, S., Wetherill, M., & Cooper, G. (2011). Past, present and future of information and knowledge sharing in the construction industry: Towards semantic service-based e-construction?. *Computer-Aided Design, 43*, 502-515. doi:10.1016/j.cad.2009.06.005

Rezgui, Y., Hopfe, C.J., & Vorakulpipat, C. (2010). Generations of knowledge management in the architecture, engineering and construction industry: An evolutionary perspective. *Advanced Engineering Informatics, 24*, 219-228. doi:10.1016/j.aei.2009.12.001

Romberg, R., Niggl, A., van Treeck, C., & Rank, E. (2004, June). *Structural Analysis based on the Product Model Standard IFC*. Paper presented at the 10. International Conference on Computing in Civil and Building Engineering (ICCCBE), Weimar, Germany. https://doi.org/10.25643/bauhaus-universitaet.243

Schuler, E. (2016). *Tragwerkslehre 3 - Tragwerkselemente*. Vaduz, Liechtenstein: Universität Liechtenstein. Retrieved from http://www.gbd.at/skripten/TWL3.pdf

Sibenik G. (2016). Building information modelling based interdisciplinary data exchange: a case study. In J. Underwood, B. Kumar, & F. Khosrowshahi (Eds.), *Proceedings of 1st International UK BIM Academic Forum Conference* (pp. 379-90). Glasgow, UK: Glasgow Caledonian University.

Sibenik G. & Kovacic I. (2019, June). *Proposal for a classification system supporting interdisciplinary model exchange between architecture and structural analysis*. Paper presented at the conference CIB World Building Congress 2019 ‘Constructing Smart Cities’, Hong Kong, China.

Wan, C., Chen, P.-H., & Tiong, R. L. K. (2004). Assessment of IFCs for Structural Analysis Domain. *ITcon, 9*, 75-95. Retrieved from https://www.itcon.org/paper/2004/5

Wang, X., Cui, Z. P., Zhang Q. L., & Yang, H. Z. (2013). Creating Structural Analysis Model from IFC-Based Structural Model, *Advanced Materials Research*, *Vols. 712-715*, 901-904. https://doi.org/10.4028/www.scientific.net/AMR.712-715.901

Weise, M., Katranuschkov, P., Liebich, T., & Scherer, R. J. (2003). Structural analysis extension of the IFC modelling framework. *Electronic Journal of Information Technology in Construction, 8*, 75-95. Retrieved from https://www.itcon.org/paper/2003/14

Zhang, X.-Y., Hu, Z.-Z., Wang, H.-W., & Kassem, M. (2014). An Industry Foundation Classes (IFC) Web-Based Approach and Platform for Bi-Directional Conversion of Structural Analysis Models. In I. I. Raymond, & I. Flood (Eds.), *Proceedings of the 2014 International Conference on Computing in Civil and Building Engineering* (pp. 390-397). Orlando, USA: American Society of Civil Engineers (ASCE). https://doi.org/10.1061/9780784413616.049