

INTERPRETED OPEN DATA EXCHANGE BETWEEN ARCHITECTURAL DESIGN AND STRUCTURAL ANALYSIS MODELS

SUBMITTED: December 2019

REVISED: October 2020

PUBLISHED: February 2021

GUEST EDITORS: Francis Siu, Hung-Lin Chi, Hsi-hsien Wei & Minkoo Kim

DOI: [10.36680/j.itcon.2021.004](https://doi.org/10.36680/j.itcon.2021.004)

Goran Sibenik

Institute for Interdisciplinary Building Process Management, TU Wien, Austria
goran.sibenik@tuwien.ac.at

Iva Kovacic, Univ. Prof. Dr.techn.

Institute for Interdisciplinary Building Process Management, TU Wien, Austria
iva.kovacic@tuwien.ac.at

SUMMARY: *The heterogeneity of the architecture, engineering and construction (AEC) industry reflects on digital building models, which differ across domains and planning phases. Data exchange between architectural design and structural analysis models poses a particular challenge because of dramatically different representations of building elements. Existing software tools and standards have not been able to deal with these differences. The research on inter-domain building information modelling (BIM) frameworks does not consider the geometry interpretations for data exchange. Analysis of geometry interpretations is mostly project-specific and is seldom reflected in general data exchange frameworks. By defining a data exchange framework that engages with varying requirements and representations of architectural design and structural analysis in terms of geometry, which is open to other domains, we aim to close the identified gap. Existing classification systems in software tools and standards were reviewed in order to understand architectural design and structural analysis representations and to identify the relationships between them. Following the analysis, a novel data management framework based on classification, interpretation and automation was proposed, implemented and tested. Classification is a model specification including domain-specific terms and relationships between them. Interpretations consist of inter-domain procedures necessary to generate domain-specific models from a provided model. Automation represents the connection between open domain-specific models and proprietary models in software tools. Practical implementation with a test case demonstrated a possible realization of the proposed framework. The innovative contribution of the research is a novel framework based on the system of open domain-specific classifications and procedures for the inter-domain interpretation, which can prepare domain-specific models on central storage. The main benefit is a centrally prepared domain-specific model, relieving software developers from so-far-unsuccessful implementation of complex inter-domain interpretations in each software tool, and providing end users with control over the data exchange. Although the framework is based on the exchange between architectural design and structural analysis, the proposed central data management framework can be used for other exchange processes involving different model representations.*

KEYWORDS: *data exchange, classification, interpretation, architectural design, structural analysis, BIM*

REFERENCE: *Goran Sibenik, Iva Kovacic (2021). Interpreted open data exchange between architectural design and structural analysis models. Journal of Information Technology in Construction (ITcon), Special issue: 'CIB World Building Congress 2019: Constructing Smart Cities', Vol. 26, pg. 39-57, DOI: 10.36680/j.itcon.2021.004*

COPYRIGHT: © 2021 The author(s). This is an open access article distributed under the terms of the Creative Commons Attribution 4.0 International (<https://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.



1. INTRODUCTION

The AEC industry aims to improve efficiency and quality through digitalization of planning process workflows. One of the digitalization goals is to create a model-based communication and seamless data exchange between professional domains. The structural analysis domain is responsible for dimensioning the load-bearing building elements. These elements usually originate from architectural design, which tends to be a step preceding structural analysis. Data exchange between architectural design and structural analysis is one of the most important and common exchanges during the planning process, and needs to be included in the data exchange framework (Kepplin et al, 2017, Kovacic et al, 2015).

Data exchange between architectural design and structural analysis models is still conducted via physical documents (paper), digitized documents (pdf files) or files containing two-dimensional (2D) geometry (dwg drawings), while three-dimensional (3D) model-based exchange currently takes place only in isolated cases and intrafirm workflows. If a 3D building model is created from 2D documents, geometry needs to be remodeled in 3D structural analysis tools, which is an expensive, redundant and error-prone process (Kepplin et al, 2017). Despite these problems the industry has so far failed to achieve a model-based exchange that displays a sufficient level of usability as well as trust for the end users (Sibenik and Kovacic, 2020).

To support the exchange between architectural design and structural analysis, researchers have proposed adapting non-suitable or creating new domain-specific frameworks (Ramaji and Memari, 2020, Hu et al, 2016). However, a lack of professional knowledge means that such data exchanges are often implemented incorrectly in the various software solutions. Practitioners, however, have established various workarounds to overcome the shortcomings of software tools (Holzer, 2016). Still, geometry transfers are very common in AEC practice, and usually require geometry interpretation (Sanguinetti et al, 2012, Venugopal et al, 2012), especially the transfer to structural analysis.

In our previous research (Sibenik and Kovacic, 2020), the analysis of data exchange in practice showed that crucial preconditions for systematic improvement are: (1) the interpretation rules between different domains, and (2) data models supporting domain-specific requirements. These preconditions are not clearly defined in the existing standards or frameworks. A single integrated model supporting various AEC domains does not suffice: the AEC industry is stepping towards multiple domain-specific data models (Scherer and Schapke, 2011, Solihin et al, 2016), as well as the automated or semi-automated interpretation rules between them (Sanguinetti et al, 2012). Data exchange between architectural design and structural analysis requires extensive geometry interpretations, which are not considered in an open exchange framework such as industry foundation classes (IFC). In this paper we underline the need for paradigm shift from a product-model-centric to a process-model-centric information space, as did Scherer and Schapke (2011). The new paradigm needs to be reflected in the data exchange framework, and should consider the exchange to structural analysis.

Therefore the main research question this paper addresses is: *how to efficiently support a model-based exchange between architectural design and structural analysis tools.*

In order to support the model-based exchange, we will propose a new framework that overcomes the shortcomings of existing data exchange frameworks. The strategies to overcome the shortcomings have been identified in our previous research (Sibenik and Kovacic, 2020). Thus, we will introduce domain-specific classification and inter-domain interpretations, based on a review of existing classification standards, software tools and data exchange standards. In addition to that, the relationships between the systems will be proposed as interpretation procedures. Automating the exchange will be achieved by establishing the connections with proprietary building models. Finally, the framework will be implemented and demonstrated with a new system architecture. The aim of the research presented in this paper is to close the knowledge gap by including the complex geometry interpretations needed for data exchange between architectural design and structural analysis in the data exchange framework. By relocating the interpretation procedures from the importing software tools to the open central model, end users gain insight into the processes, thereby reducing the need for customized, individual development for each software tool.

This paper is structured as follows: Section 2 reviews the literature related to the data exchange framework; Section 3 presents the methodology and background on existing classification standards in the industry, the analysis of classification systems in software tools and the IFC data exchange standard; Section 4 provides an overview of the

novel framework that consists of classification, interpretation and automation; classification and interpretation are presented in Sections 5 and 6, respectively; Section 7 describes a framework implementation and test system architecture, while Section 8 discusses the proposal; and Section 9 concludes with the significance of the framework, wider implications and the future outlook.

2. LITERATURE REVIEW

2.1 Data exchange between architectural design and structural analysis

There are significant differences between the architectural design and structural analysis models, hence interpretations are needed. Ramaji and Memari (2016) list three possibilities for creating a structural analysis model from native BIM tools: (1) a structural analysis model is created in a native software tool; (2) a structural analysis model is created from a native model by a structural analysis tool; (3) an additional software tool (or a plug-in) is used to generate a structural analysis model. The focus of this paper is on open data exchange between architectural design and structural analysis, where IFC is the most frequently used standard. Specifically, this paper emphasizes that the end user needs to be able to access and change not only the building data model, but also the interpretations taking place during the creation of structural models. Existing model interpretations are not open to an end user and generally work as a 'black-box' scenario (Holzer et al, 2007).

The literature review identified several framework proposals using IFC building data models as an exchange starting point, which is interpreted to a structural analysis model. Deng and Chang (2006) describe a framework which uses the IFC 2x2 schema; a schema version not supporting the structural analysis concepts. As the IFC schema did not sufficiently support the structural analysis representations of building elements, they developed an additional XML-based file format suitable for importing to several structural analysis tools. Deng and Chang converted an architectural IFC model to a 'general format of unified finite element model'. Liu et al (2010) developed an integration tool to convert building models between IFC and the PKPM structural analysis formats, with a 3D viewer supporting the exchange. Qin et al (2011) improve the exchange using a similar approach to Deng and Chang (2006); that is, describing the IFC building model as an architectural physical model and creating a complex system that prepares it as an XML-based unified finite element method (FEM) model, and can then further prepare the information for several structural analysis tools. Zhang et al (2014) also deal with the conversion of IFC models to multiple structural analysis formats. They additionally propose a bidirectional conversion as well as the conversion between structural analysis models on their web-based platform. The same platform was used by Hu et al (2016), where further case studies are also described.

The current version of IFC supports the majority of structural analysis requirements. Ramaji and Memari (2016) develop a framework describing direct and interpreted exchange between architectural and structural analysis models, meaning that some of the information provided by the architects can be directly imported while some needs to be interpreted. They later implement this framework with the IFC (Ramaji and Memari, 2020), where their workflow involves two variants of IFC files: architectural design and structural analysis.

In the literature above the interpretation steps are based on the input model; creating a new structural analysis model from a specific IFC export. However, because of the IFC schema redundancy, solutions using IFC are software tool specific (Sibenik and Kovacic, 2020); that is, the problem-solving approaches proposed above do not reflect the needs of the various current design practices. The conceptual differences between the models and relevant interpretation steps are defined for the particular research needs, rather than being extensively investigated on a more generic level. Thus there is a lack of investigation of how domain-specific requirements could be incorporated into existing exchange frameworks.

2.2 Data exchange frameworks

Some researchers question the use of the central integrated model, and instead pursue the use of multiple models. Lee and Jeong (2012) report that the integrated model is not feasible for technical and procedural reasons, because the model quickly becomes too large and does not support dynamic design processes. Scherer and Schapke (2011) move from all-in-one models and focus on a multi-model concept. They propose link models to relate multi-models, focusing on semantical interoperability. Solihin et al (2016) separate a single model into multiple domain-specific federated models; however, complex geometry interoperations are not part of their framework. Lee et al (2014) describe a web-based framework involving domain-specific filters as an intermediary step between a

complex central model and domain-specific models; thus stepping away from the concept of integrated model. Sanguinetti et al (2012) address the need for multiple geometric models and describe two types of geometric operations for model processing – union and split operations – that can be used for models that adopt a similar approach to building element representation. Our previous research (Sibenik and Kovacic, 2020) recognized the need to separate multiple domain-specific building data models such that the relationships between multiple models are retained and originate from the inter-domain conceptualization.

The need to separately deal with the geometrical and non-geometrical information within the central storage was recognized in our previous analysis (Sibenik and Kovacic, 2018). Both types of information are usually treated similarly in data exchange frameworks, where researchers often focus on one type only. However, the interpretation of the semantic and syntactic description of the building information models do not have many similarities with the geometrical data interpretations. Lee and Jeong (2012) distinguish between geometric and non-geometric information in their work, but do not focus on geometric interpretations. Andrade et al (2008) state that ontology is only concerned with static and not dynamic knowledge. Work relating to ontology, taxonomy and classifications mostly deals with the non-geometrical information (e.g. Pauwels et al, 2017); whereas our view is that the geometrical information during the exchange is dependent on geometrical kernels used by the interoperating software tools, interpretations taking place and the geometrical definitions supported by the mediatory model such as IFC (if any). The exchange between geometrical kernels is usually separately researched (Zhu et al, 2019), especially if the semantic information differences are manually or easily overcome. In addition, the geometric transformation of building models involves complex geometrical methods, and some researchers have already suggested introducing a geometry kernel to perform interpretations (e.g. Romberg et al, 2004; Mora et al, 2008).

Introduction of business processes, procedural or expert knowledge in data exchange frameworks is discussed in work dealing with interoperability issues in the AEC industry. Rezgui et al (2011) emphasize the existence of tacit and explicit knowledge in the AEC industry and propose a process-driven framework, which is web-based and supported by ontologies. Rezgui et al shift the focus from standardization to automation of processes, using service-oriented IT architecture. A framework for collaboration platform presented in Singh et al (2011) describes the need for organizations to develop their own data management practices to suit their business models, with flexible and user-friendly user interfaces. More recently, Lindgren and Widen (2018) stress that academic research focuses on technical interoperability while ignoring business processes. Although their work deals with dissemination of building information management, they regard knowledge as domain-specific, procedural and general, instead of tacit and explicit. Amann and Borrmann (2016) introduce ‘procedural knowledge’ to the IFC schema through IFC Procedural Language (IFCPL). IFCPL displays numerous advantages compared with the limited IFC EXPRESS schema; however, it does not provide a solution to complex interpretations of geometry, which is still based on the IFC that lacks predefined geometry manipulation methods. A framework created by Sacks et al (2017) incorporates expert knowledge to semantically enrich the models originating from point clouds (a set of data points in space). The rules are based on element geometries and their relationships, which produce new non-geometric model information. Ladenhauf et al (2016) improve the data exchange process specifically for energy performance simulation. Their interpretations are mainly geometrical, and are called geometric simplification. Ladenhauf et al introduce several algorithms to manage IFC models, based on expert knowledge to achieve automatic generation of new building models for energy performance simulation.

Thus data exchange frameworks are moving towards multiple domain-specific models and introducing interpretation processes between the models. An integrated approach that can process both geometrical and non-geometrical information is necessary in the AEC industry. In summary, the proposed frameworks consider expert knowledge and interpretation of semantic information, while geometry interpretations are limited to filtering of same geometric representations. Complex interpretations of geometry are still lacking in the context of BIM data exchange frameworks, despite the above-mentioned proposals.

2.3 Geometry interpretations in other domains

Geometry interpretations can be found across the AEC industry, as well as beyond. One such example is the computational design approach (Caetano et al, 2020). It deals with design, analysis and implementation of algorithms for solving geometrical problems and it is present in domains such as geographic information systems (GIS), computer graphics or computer aided design (CAD). The generative design approach belongs to computational design and uses algorithms to generate designs (Caetano et al, 2020). Singh and Gu (2012) explore

five different generative design approaches and create a framework which can combine them. Use of generative design in AEC is often related to evolutionary processes and multiple solutions, which is especially desirable in the concept design phase (Hofmeyer and Davila Delgado, 2013); in the developed design phase the term ‘algorithmic design’ would be more suitable for generation of structural analysis models because it is characterized by an identifiable correlation between the algorithm and its outcome (Caetano et al, 2020).

Zhu et al (2019) integrate BIM and GIS, mainly by converting the geometry defined using IFC building data model into shapefile geometry which is then used in GIS. The geometry remains the same: its underlying geometry definitions are changed. Davis and Laender (1999) describe standard geometry transformation methods for creating different resolution, detail level and representation styles in GIS. They use unification, derivation and replication to avoid consistency problems caused by redundant information. Unification holds a single representation in the database, and all the following representations are obtained from it. Derivation stores a primary representation and secondary representations derived from it, but only the primary one may be modified, while the other ones are updated. Replication holds multiple representations that need to be synchronized between each other. In the AEC industry multiple stakeholders change and define building models with complex CRUD (create, read, update, delete) rights that often change between projects, making only replication a feasible solution.

Vangenot et al (2002) use two complementary approaches – integrated and inter-relationship – to deal with consistency of different viewpoints and resolutions in representing geographic data. Borrmann et al (2015) deal with a specific problem of tunnel design, which belongs both to BIM and GIS domains. Their approach supports tracing the geometry development and establishing procedural geometry descriptions because the design process and data modification are highly dynamic compared with conventional geographic data.

Manufacturing industry is more digitized than the AEC industry: the processes are more repetitive and therefore easier to capture and automate. Learning processes and development of knowledge is enabled by capturing experiences (de Giorgio et al, 2020). In manufacturing the knowledge-based methods are based on expert knowledge that is codified in advance. De Giorgio et al define ‘declarative’ and ‘procedural’ knowledge, where declarative knowledge is systemic information of one system related to information about another system at an instant of time, and procedural knowledge is related to the variation of information about another system between two instants of time.

Research in the industries beyond AEC also deals with multiple model representations and interpretations taking place between the models. The geometry interpretations can be summarized as the interpretation of (a) same geometry and similar representation (e.g. different geometry kernels), (b) similar geometry and different representation (change of level of details, especially in cartography) and (c) different geometry and different representation (e.g. architectural design to structural or energy analysis). Fig. 1 shows possible model states in a three-axial coordinate system (similar to Scherer and Schapke, 2011). The lessons from other industries are useful, but the peculiarities of the AEC industry, where the model is changed in all three directions and the relationships are crucial, makes the automation of all possible exchange variants far more complex. Contriving all of the possible changes in the state of the model would quickly result in scalability issues, thus requiring clear descriptions of the process and the focuses.

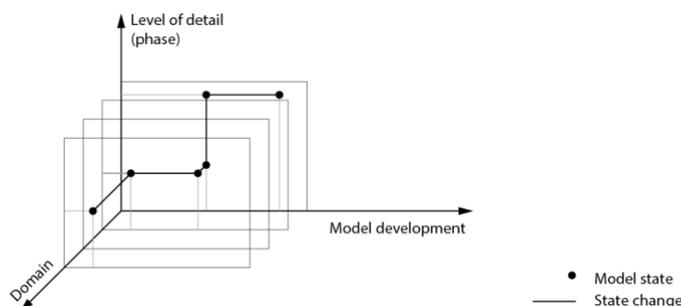


Fig. 1 Model in three-axial model space

3. METHODOLOGY AND BACKGROUND

Existing interpretation methods within software applications lack transparency, are not exhaustive and, in some cases, are incorrectly implemented, discouraging designers and planners from using automated data exchange. Automated or semi-automated interpretations need to describe the most repetitive tasks during the design process in such a way that they remain understandable and open, as well as reliable to the end users.

On the basis of the above literature review and our previous research (Sibenik and Kovacic 2020), we have identified:

- The deficits in the data exchange frameworks; and
- The requirements of data exchange between architectural design and structural analysis models.

This has enabled us to develop a proposal for a novel data exchange framework.

The framework is based on three parts:

- *Classification* – Classification in the AEC industry is a well-established topic, and the classification standards, classification systems and implementation in software tools will be analyzed in Section 5.
- *Interpretation* – Existing standards will be analyzed for the presence of interpretation rules in Section 6.
- *Automation* – Automation is software-specific (i.e. proprietary tools are used) and is closely related to the application programming interface (API) of a specific software tool. It is not discussed in detail in this paper, but will be explored at a later date.

In Section 7, the framework proposal will be demonstrated and tested using a test building model, for which the supportive system architecture will be developed and implemented. The building model consists of three building element types: wall, column and slab. These have been chosen because they are the basic, most common elements in building models suitable for testing a building model. Geometries of building elements are simple: rectangular walls and slabs with constant thickness and columns with square cross-section. Classification, interpretation and automation will be applied to all three elements as follows:

- The proposal of the classification system used in the test building is primarily based on the analysis of the IFC standard.
- Interpretations are developed by comparing the existing physical model with the required analytical model of the test building, and by developing the algorithms that can overcome the model differences.
- Automation is achieved by mapping the information from the test building model in IFC format to the central architectural model, and by mapping the central structural analysis to the RFEM model (see Section 7).

The development of detailed methodology to define and test geometry interpretation rules will be discussed in future work.

The rest of this section discusses the state-of-the-art regarding classification and interpretation. First, a review of classification-related standards is presented with terminology and rules in creating a classification system. Further on, an analysis of software tools provides an insight in the implementation of classification systems. The analysis is extended in Sub-section 3.3 to the IFC data exchange standard.

3.1 Classification standards

The most significant standards describing the approaches for classification systems and terminology are ISO 1087-1 (ISO, 2000a), ISO 860 (ISO, 2007b), ISO 704 (ISO, 2000b) and ISO 22274 (ISO, 2013). These standards define most common terms in classification systems, such as objects, classes, concepts, and classification and concept systems. A 'concept system' establishes the relationships between specific concepts in three different ways (ISO, 2000b):

- *Hierarchical generic* – the intention of the subordinate concept is included in the intention of the superordinate concept (the subordinate concept is a specific, whereas the superordinate is generic).
- *Hierarchical partitive* – superordinate concept represents the whole and subordinate concept a part.
- *Associative* – the non-hierarchical associative relationships can be of various types, and are a 'thematic connection by virtue of experience'.

The basis of each classification system is an appropriate concept system. Recommendations on how to use a concept system and build a classification system are given in ISO 22274 (ISO, 2013). Consistent terminology is a mandatory condition for achieving unambiguous communication. Users must be provided with definitions for the class in order to reduce ambiguities. A concept system reflects the knowledge about the concepts in a specific domain, and the relationships between them. However, multiple classifications can be a product of a single concept system. ISO 22274 (ISO, 2013) defines three types of classification systems: (1) the enumerative system, which represents all the classes through the hierarchic system; (2) the faceted system, which lists multiple classifications of the object, where an object can belong to any combination of classes from the facets (e.g. material and building element for a concrete wall); and (3) the combined system which, according to ISO 22274, is recommended as enumerative on the higher levels and faceted on the lower levels.

ISO 860 (ISO, 2007b) suggests that when there are multiple classification systems with non-minor differences, even if the concept systems are from the same subject field, they should not be harmonized. But architectural design and structural analysis are domains with major conceptual model differences; therefore harmonizing the terms present in architectural and structural analysis software is considered unfeasible. Although both domains take part in the building design process, the two classification systems have different purposes.

Classification systems in the AEC industry are mainly created at the national level. Examples include Uniclass 2015 (United Kingdom); Omniclass, MasterFormat and Uniformat (United States of America); CoClass (Sweden); CCS (Denmark); building 2000 (Finland); STABU LexiCon (Netherlands); and NS 3451 (Norway) (Dikbas and Ercoskun, 2006, Lou and Goulding, 2011). Significant attempts to classify building information in the so-called DACH region (Germany, Austria and Switzerland) can be found in the classifications for costs in construction; namely, DIN 276, ÖNORM 1801-1 and SIA 112, respectively. The common procurement vocabulary (CPV) developed by the European Union defines a classification system for all matters pertaining to public contracts.

There are two ISO standards related to classification systems in the AEC industry: ISO 12006-2 (ISO, 2015) and ISO 12006-3 (ISO, 2007a). ISO 12006-2 provides several classification tables regarding all stages in the construction project lifecycle, with definitions. Classification systems such as CCS and Uniclass 2015 are based on this standard. ISO 12006-3 specifies the language-independent model defined using the EXPRESS data modelling language. The model identifies the way in which the objects, groups and relationships within them are created to represent the knowledge. STABU LexiCon and buildingSMART Data Dictionary (also known as bSDD) are based on this standard.

Ekholm and Haggstrom (2011) criticize the classification standard ISO 12006-2 for failing to bring together the existing national systems and for not being suitable for specific domains in the construction sector (e.g. cost calculations using the Swedish and Danish systems). Ekholm (2005) describes the difference between the ISO 12006-2 and IFC building data model, identifying some issues in the ISO 12006-3-based IFC building data model, such as elements being defined without reference to their spatial, functional or compositional view. Ekholm (2005) also suggests several steps necessary for harmonizing the existing systems based on the two standards.

The classification systems aim to encompass the whole AEC industry at the international level, eventually harmonizing the term codes, but at the moment they do not recognize the heterogeneity of the various domains. Classification systems are a prerequisite for organizing building product libraries so that building data models with different purposes could communicate. The first critical step in creating a classification system is defining the purpose of classification (Ekholm, 1996, Jorgensen, 2011). The purpose of the classification defines various exchange requirements that will need to be supported, depending on the software and domain.

Most of the existing classification systems are not used in the products developed by the international software industry for architectural design and structural analysis. Generally, terminology is not harmonized among the software tools supporting the AEC industry. ACCA software (2018) identified the following reasons for the low implementation of classification systems:

1. Local – limited by their geographical and linguistic character;
2. Incomplete – they do not encompass all the relevant knowledge;
3. Technically behind – there are more advanced ways to classify information that can be supported with current technology; and
4. Specific – they are mainly created for cost calculations; differences in representations are not addressed; and interpretations would make data management more complicated in some cases.

3.2 Classification in software tools

The conceptualization of objects in software tools is often software-solution sensitive and follows the software development logic, so lacks understanding of the professional domain (Andrade et al, 2008). The software industry provides the AEC industry with the tools for architectural and structural design, analysis and optimization (i.e. BIM tools). BIM authoring tools support object-based modelling and use different classification systems. The objects represent the concepts of a specific domain. ‘Objects in the domain of interest are modelled as software objects with properties that represent domain object properties’ (Ekholm, 2005). For the software industry, the property is known as an ‘attribute’. By using the classification, software developers are able to use the same code to deal with multiple object properties, methods, relations, etc. The need to accommodate the classification systems in the AEC sector to work effectively with the advances in the software industry was recognized more than 20 years ago, in order to develop a classification system which will be used for both AEC and software industry. According to Ekholm (1996), product modelling needs to be classified within a series of different classifications, and Ekholm refers to various project stages as ‘purposes’. More recently, Jorgensen (2011) has emphasized the need to explore new opportunities in the software industry that can be used in classification systems, and describes multiple misconceptions in existing classification systems that need to be reconsidered because software tools are increasing their stake in construction projects and could provide new possibilities if suitably designed for the AEC industry.

We have analyzed the classification systems of five of the software tools mostly used in the Central European region and conducted a comparison of these systems with existing standards and literature. The software tools are for architectural design (Archicad 21), architectural and structural design (Allplan 2018 and Autodesk Revit 2019), and structural design and analysis (Dlubal RFEM 5.17 and SCIA Engineer 18.1). Classes representing building elements found in almost every high-rise building project were reviewed: namely, column, wall and slab. All the analyzed software tools claim to be suitable for BIM workflows and are IFC certified. One of the characteristics of BIM models is that they consist of parametric 3D objects containing additional information, alongside geometrical information. All of the examined tools are object-oriented modelling, design and analysis tools. The software classification systems were analyzed primarily through software tool interfaces, but also in regard to corresponding APIs and the behavior of resulting objects. Fig. 2 presents the existing classification systems graphically based on ISO 704 (ISO, 2000b), and shows the differences in the classifications used by these tools.

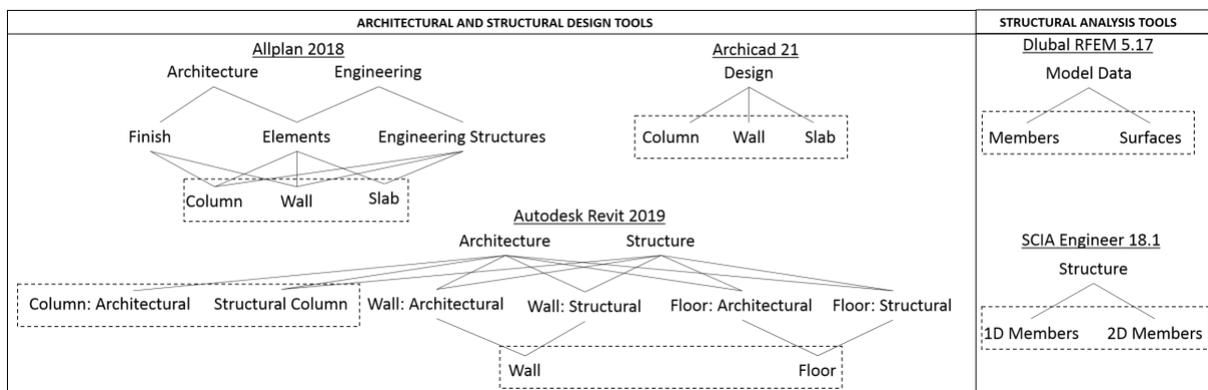


Fig. 2 Classification systems in software tools

Our analysis found that the relationships between classes are hierarchical generic. On the one hand, we found that terms generally differ and the definitions describing the terms are missing; however, they can be compared by analyzing the object behavior. Among the tools mainly used for architectural design and architectural and structural design, the main difference is the possibility to access the same class (except for columns in Autodesk Revit, which are divided into two classes). The terms are similar in all three software tools, but the way they are structured differs. On the other hand, the structural design and analysis software solutions show similarities and the three building elements are represented in only two classes. This results from the reduced dimensionality of analytical models. In structural models, columns are defined as linear elements, where the cross-section is non-geometrical information (i.e. not modelled), and walls and slabs are planar elements where thickness is non-geometrical information. The column class suggests that a linear element is vertical, the wall class that the planar element is

vertical and the slab class that a planar element is horizontal. These classes are not necessarily part of the structural analysis software tools (Dlubal RFEM), or the class is not limited to the characteristic of the building element (e.g. direction), meaning that it can be changed during the design and still remain in the same class (SCIA Engineer). Therefore, the classes ‘column’, ‘wall’ or ‘slab’ are considered irrelevant for structural analysis models, and the main classification is on a higher level, involving ‘linear’ and ‘planar’ elements.

3.3 Classification in data exchange standard

End users are required to deliver information that conforms to the relevant national standards, using the templates and defining the data exchange properties. This process is often not sufficiently defined and software tools do not guarantee that the information will retain the definition after the data exchange. The process of information mapping during the exchange processes is left to software developers. This research primarily addresses structural analysis tools, and the data originally created in architectural design software tools. The classification systems mentioned in Sub-section 3.1 do not recognize the building elements in their structural analysis representation (linear and planar elements).

Early work on the IFC building data model (Weise et al, 2003) introduced the majority of the structural representations and their corresponding properties. The IFC standard conforms to the buildingSMART data dictionary, which is based on ISO 12006-3. The IFC is the most widely accepted standard for data exchange in practice and it defines the building elements’ classification in an object-oriented manner. It tends to grow and aims to encompass all the elements present in the AEC industry. The standard recognizes all three building elements considered in this research in their architectural as well as their structural representation.

This means that the same building element can occur in the same model multiple times – as a structural and an architectural element, without any interdependency between them. Our previous work (Sibenik and Kovacic, 2020) concluded that the crucial data-exchange problem is that the relationships between the element representations do not exist and are not defined in an IFC-based data exchange framework. The relationships are supposed to be described using information delivery manuals (IDMs) or varying model view definitions (MVDs), but this concept has not been widely accepted in practice (buildingSmart, 2011). Fig. 3 shows an extract of IFC schema representing the classes from the analysis for this paper and emphasizes the crucial problem within the standard that we identified; namely, the missing relationship between the representations of the same building element.

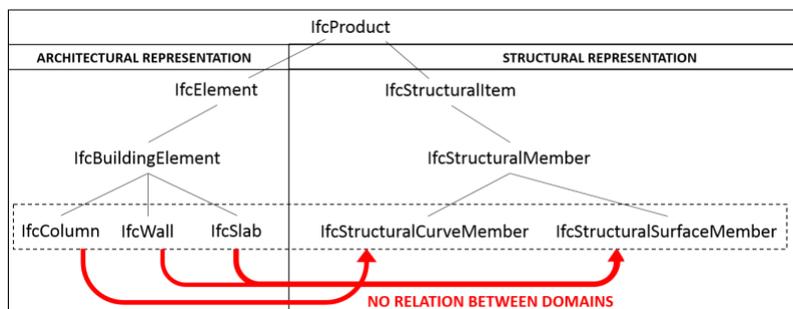


Fig. 3 Classification used in the IFC standard

The relationships between the IfcElement and IfcStructuralItem are not part of the standard. Ekholm (1996) states that it is necessary to disregard the properties that are not of interest and create a definite and exhaustive system, where ‘definite’ means that an object may not belong to more than one class of the same rank, and ‘exhaustive’ means that every object must be assigned to a class. A unique encompassing schema in the AEC industry becomes exhaustive by supporting each domain-specific representation, software tool and corresponding classes. The industry heterogeneity makes this task highly complex and results in a constantly changing redundant schema; achieving ‘exhaustiveness’ is more feasible by focusing on multiple domain-specific schemas. Thus, the IFC exchange standard does not manage to be definite because the elements may occur in multiple classes simultaneously.

The software tools mostly deal with a single representation of elements and support domain-specific concept systems. The interfaces do not necessarily follow the underlying classification system, and allow multiple terms and ways to access the same resulting class. Each object contains a number of additional properties, addressed in

different ways by each software tool, requiring additional analysis and harmonization; in order to reduce the complexity our analysis does not include properties.

4. NOVEL FRAMEWORK PROPOSAL

Having analyzed the state-of-the-art in terms of standards defining classification, existing domain-specific classification systems, lack of inter-domain relations and other approaches to data exchange frameworks, a new framework has been developed that introduces inter-domain relationships as a mandatory part of the data exchange process. The three mandatory parts in the proposed framework are: classification, interpretation and automation (Fig. 4).

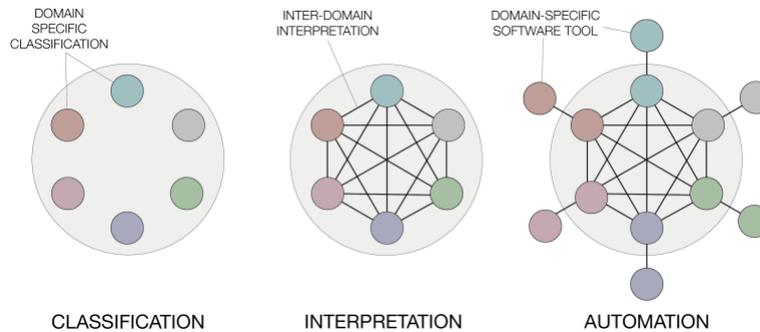


Fig. 4 Framework overview

Classification (see Section 5) – This is the part of the framework where classification systems defining domain-specific terms are developed. The systems can include building elements, properties, relationships, and the containers used to interrelate multiple element, of all domains partaking in the data exchange processes; eventually creating a master central classification as a loosely coupled system of domain-specific classification systems, supporting all exchange requirements of all domains partaking in the planning process. The relationships between the terms must be derived from the inter-domain concept system. The concept system is the existing knowledge that is not limited to a single domain.

Interpretation (see Section 6) – This part of framework overcomes the differences between centrally available information and a domain-specific model required in an importing software tool. The model is prepared for import in the software tool, in terms of its semantical and geometrical requirements, based on the available information. Interpretation comprises the multiple procedures required to prepare the model, such as validation, filtering, non-geometrical interpretations, geometrical operations, enrichment and reasoning.

Automation – This part of the framework puts the created central classification and interpretation parts in context with other software tools. In this way, classification and interpretation are related with proprietary software tools by information mapping. Mapping processes between a domain-specific open model and the native building model are simplified to establishing relationships with already prepared data; no complex interpretation or reasoning tasks are required for automation. To connect the central systems with software tools, APIs provided by the software tool producer are used. The quality of the API governs the level of access to information and influences exchange to a certain extent.

5. CLASSIFICATION

The analysis of software tool and data exchange classification systems implies that the relationships between the architectural and structural representations are missing. Nevertheless, structural engineers are able to distinguish and interpret the elements without significant problems. We consider that it is necessary to understand the concepts from both perspectives and unite them in a single concept system. This approach allows for definition of interpretations between the domains. The first recommendation from ISO 12006-2 (ISO, 2015) for developing a classification system is defining its purpose. As already discussed, the harmonization of classification systems for architecture and structural analysis domains is not a feasible solution. Thus, our recommendation are *two classification systems* with architectural and structural purposes, based on a *unique concept system*. The analysis of classification systems and the standards responsible for the classification discussed in Sections 3.2 and 3.3 above implied the classification systems for the examined building elements illustrated in Fig. 5.

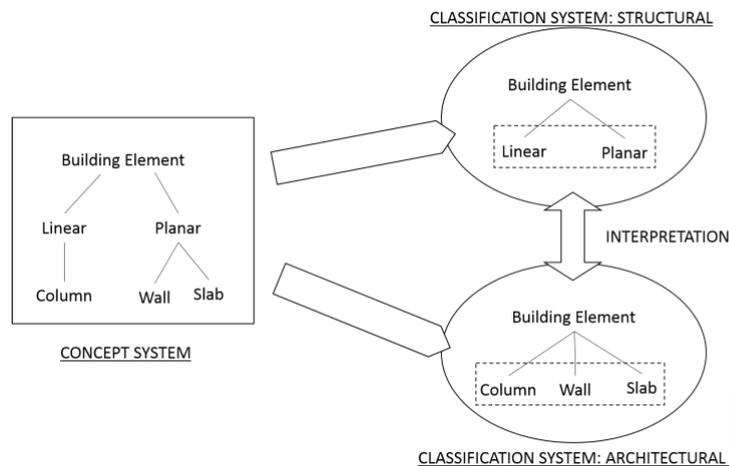


Fig. 5 Proposal for domain-specific classification systems based on the concept system

An approach involving multiple classification systems must retain the interpretation rules existing in the concept system. Without the interpretation rules, the systems lose their conceptual origin relationships and the data exchange between two classification systems becomes impossible. The proposal can be defined as:

concept system = classification systems + interpretations

Each concept is separated from the other concepts by its intension: the unique set of its characteristics. Each terminological entry should be composed of a statement explaining what the concept is (ISO, 2000b), but the software tools usually do not provide the end users with the clear definition. From the analyzed classification systems, only the IFC standard describes the terms textually. However, the definitions from IAI (IAI International Council Limited, 2019) define columns as ‘structural member of slender form, usually vertical’, walls as ‘usually vertical, or nearly vertical, planar elements’ and slabs as ‘the construction that normally encloses a space vertically’, and these definitions are too vague for automatization of the interpretation rules without introducing clearly defined tolerances. The AEC industry lacks definitions that would facilitate the automatization of the interpretation rules. This leads to over-differentiating of classes in software tools and excludes the possibility of automating interpretations. To facilitate the seamless data exchange, definitions of elements are required that will make automated interpretation possible and reflect the underlying concept system. Therefore the three building elements compared in this study are regarded strictly as horizontal or vertical. The following are examples of building element definitions that allow for the automated interpretation:

- *Column*
A linear vertical building element.
- *Wall*
A vertical planar building element.
- *Slab*
A horizontal planar building element.
- *Linear element*
Building element where the largest dimension of its bounding box equals or is greater than 10 times next greatest dimension.
- *Planar element*
Building element where the largest dimension of its bounding box is smaller than 10 times next greatest dimension.

The definitions are used as an example only and have no roots in any reviewed standard; rather, they are used to describe a necessary amount of information that could be used to clearly distinguish building elements and automate the processes between them. The differences are based on the element’s geometrical information that can be automatically recognized. Proposed definitions serve for the classification proposal testing in Section 7 below; however, the comprehensive automation-suitable definitions need to be made with mutual consent in the industry and need to consider more building elements.

The interpretation of elements from the architectural to the structural classification is straightforward, because the

structural classes are superordinate concepts towards the architectural ones. In this case, columns are linear elements in the structural classification system, and the walls and slabs are planar. The interpretations in the reverse direction are more complex: the easiest method is to address the object by its ID if possible and update it, if it originates from the central database. If not, the elements can be interpreted by analyzing the geometrical information. Linear and planar elements contain information about cross-section and thickness, respectively, as non-geometrical information. For the three element types, it is possible to identify whether the element is vertical or horizontal: a horizontal planar element is a slab; a vertical planar is a wall; and a vertical linear element is a column.

6. INTERPRETATION

The three building element types (column, wall, slab) are not sufficiently distinct in standards regarding their domain-specific representations, so it would be possible to automate the conversion between architectural and structural classes. In fact, the majority of building elements are not sufficiently standardized regarding the relationships between multifold representations, and automation steps are not considered. However, there are recommendations about some relationships (e.g. Pech and Kolbitsch, 2005, Schuler, 2016) which suggest how to treat architectural elements when interpreting them for structural analysis manually. In this paper interpretations are defined by finding the existing knowledge about the elements in the standards, by instructions provided for structural analysis software tools, and by capturing the intuitive knowledge in a form suitable for automation from our experience. We focus on the basic interpretation concept in order to construct the framework, while in future work we aim to investigate the methodology to define the interpretations more thoroughly.

The framework presented here proposes interpretation of a centrally available architectural design model to a structural analysis model and it consists of five procedures.

6.1 Interpretation procedures

Interpretation to define a structural model from a central data model involves the following procedures (illustrated in Fig. 6):

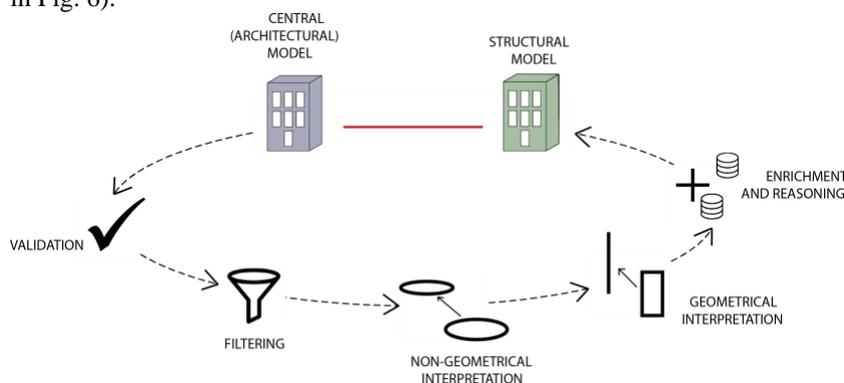


Fig. 6 Interpretation procedures

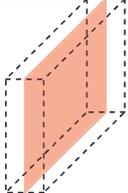
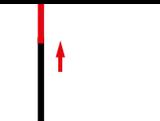
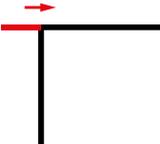
- *Validation* – Identifying the terms available in the central data model, if the elements of the model contain all necessary properties, relationships, and so on to allow for the creation of a structural model. The validation is performed on specific classes or a group of classes within the central model. The validation can be implemented in a way that automatically reports the inconsistencies to the stakeholder who generated specific information. Additionally, validation of geometrical properties can be introduced. For instance, linear elements are considered the ones where two dimensions are considerably larger than the third one ($w \ll 1$) and ($h \ll 1$) (Schuler, 2016). Further on, a planar element has one dimension considerably smaller than the other two ($t \ll 1_1, 1_2$). The difference between the column and the wall is described as the ($h/b > 5$) (Pech and Kolbitsch, 2005) in the case of the concrete and reinforced concrete walls.
- *Filtering* – The elements and properties identified in the validation procedure are reduced to those necessary for further interpretation procedures. For structural analysis the load-bearing properties are considered to isolate only the building elements relevant for structural analysis.

- *Non-geometrical interpretation* – This is mostly ontological data interpretation. Since the original model is the architectural design model, structural analysis classes of building elements 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* – The geometrical interpretation for this data exchange are complex and are described in Section 6.2 below.
- *Enrichment and reasoning* – This involves adding new information, by using external databases or methods that can enrich the existing elements with additional knowledge, for instance recognition of the architectural material and assigning a structural material.

6.2 Geometrical interpretations

The following geometrical interpretation workflow is proposed based on the test model and knowledge of practices employed by structural engineers.

Table 1 Geometrical interpretation steps

Step	Description
Linearization	 <p>Architectural columns are interpreted as linear elements. A linear element is created by connecting central points of lowest and highest face of a column, which represent the end points of a central axis. The cross-section information becomes a non-geometrical property of the linear analytical element.</p>
Planarization	 <p>Walls and slabs are converted to planar elements. The largest face is identified and placed in the central position towards the second largest face. The thickness information becomes a non-geometrical property of the planar analytical element.</p>
Vertical connectivity adjustment	 <p>In order to be able to transfer the forces, vertical elements must be connected to horizontal ones transferring the forces to the ground. Therefore, the vertical elements are tested for their connectivity with the horizontal 3D elements in their physical representation. Tolerances can be introduced as well. The elements connected in the physical model are then extended in the corresponding analytical model.</p>
Horizontal connectivity adjustment	 <p>a) Vertical planar elements need to be checked for connections in the physical 3D model, then connected in the analytical model. Horizontal connections of vertical linear elements are used in the next step: horizontal connectivity adjustment. b) Edges of planar horizontal elements need to be aligned with vertical linear and planar elements. First, the vertical elements located under the horizontal element closest to the edges need to be identified. After testing if the distances are in a tolerance scale, the edges are aligned to the vertical elements underneath.</p>

7. SYSTEM ARCHITECTURE PROPOSAL

The framework described in this paper has not yet been implemented in any available software tool or 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 (Fig. 7), which will be briefly presented in this section.

First, a central storage database was chosen, which replaced the building data stored in computer files, so the synchronous exchange can be achieved. Behavior of the end users is unpredictable: domains that could take part of the central storage as well as the software tools differ from project to project (Holzer, 2007). Therefore, the degree of flexibility available was an important factor for choosing a database, and was achieved by choosing open system architecture and the semi-structured database. For data storage, a semi-structured database (MongoDB) was chosen. The advantage of a semi-structured database rather than a relational database is the flexibility it provides to support unexpected information (Rasys et al, 2014, Lee et al, 2014).

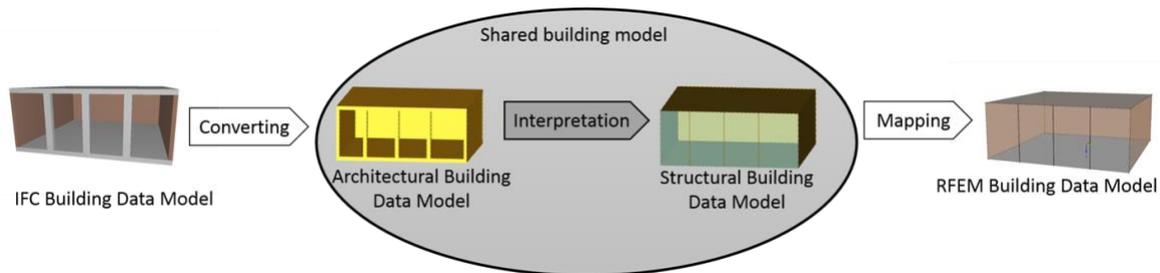


Fig. 7 Classification proposal testing workflow with screenshots

The interpretation, including validation, filtering, geometrical and non-geometrical interpretation and enrichment, can be realized using .NET Framework and C#. Automation was achieved by mapping the Revit model to the central storage, and the Dlubal RFEM model from the central storage. The Revit model was not directly mapped but its IFC 2x3-based export was used. However, in order to achieve a fully synchronous data exchange, a direct connection between Revit and the central storage needs to be realized through Revit API. The export from Revit was performed with the default setup for Coordination View 2.0. In order to map the architectural model available as an IFC file to the central storage, we created the IFCtoJSON parser, which converts the IFC file to Java Script Object Notation (JSON) format, keeping only the classes necessary for the data exchange. This JSON-based central data is then regarded as a starting point of the exchange.

Mapping processes depend on the API of the proprietary software tools, and in the case of Dlubal RFEM mapping methods are provided in C#. Since the starting point was the prepared architectural model edited to serve the new framework, the validation was not performed and filtering steps were performed during the conversion of the IFC model to JSON.

A consistent and clearly structured geometrical representation for every element with the geometrical representation was defined with the help of a geometry kernel. In order to implement geometrical interpretations the open geometry kernel OpenCascade was used. In this way, the complex geometry editing methods are simplified because of the predefined options provided by the OpenCascade kernel. OpenCascade was used with a C# wrapper, although the original version of OpenCascade is created for C++. In order to use this geometry kernel, IFC geometries are converted to OpenCascade with IFCtoJSON, and Dlubal RFEM reads the geometries from their OpenCascade definitions.

The central architectural building data model is converted to the central structural building data model, both available on MongoDB. All interpretation procedures take place before communication with the structural analysis software is initialized. The interpretation reduces dimensionality of a physical architectural model to an analytical structural model, retaining connections between elements and converting geometrical and non-geometrical data on the central storage. The process is fully automated, involving geometric and non-geometric interpretations, and enrichment and reasoning.

8. DISCUSSION

With the growth of BIM tools and digital building models in various domains, there is potential to reduce redundant remodeling tasks. A significant effort to achieve a seamless data exchange is recognized in the IFC standard. This standard encompasses multiple domains and supports varying representations of the same building element. The lack of relationships between different representations make seamless exchange between the representations impossible (Sibenik and Kovacic, 2020). Relationships between different geometrical representations are not considered in standard BIM-based data exchange frameworks. The work presented in this paper aims to improve the model-based exchange between architectural design and structural analysis tools by providing a novel data exchange framework that is able to overcome differences between model representations.

The novel framework is based on domain-specific classification, interpretation between the domains, and automation of communication with proprietary software tools. The concepts present in architectural design and structural analysis clearly require two classification systems (ISO, 2007b), and the academic community implies the use of multiple schemas (Lee and Jeong, 2012). The analysis of classification systems showed that software tools support domain-specific concept systems, but mandatory interpretation procedures between the domains are not standardized. The interpretation, representing expert knowledge as procedures or algorithms, is generally

missing in data exchange frameworks, although this lack is highlighted by the literature (Singh et al, 2011, Ladenhauf et al, 2016). In the novel framework the interpretation is centrally placed. The lack of clear interpretation rules represents a gap in standardization which, in this work, is addressed by proposing some basic interpretations. The interpretation rules are implemented in the data exchange frameworks between architectural design and structural design, but their complexity and heterogeneity is not considered, which delivers project-specific solutions (Ramaji and Memari, 2020, Hu et al, 2016). In the novel framework we introduce a geometry kernel to facilitate the complex geometry interpretations, as in Mora et al (2008). The main innovative contribution of the novel framework are the multiple open central domain-specific classifications supported by inter-domain interpretations.

A possible implementation of the framework using software independent central storage and processing has been developed with a test building model. The framework test shows that the process of importing building data models to structural analysis tools can be significantly simplified when the structural analysis tools import an already interpreted model. The interpretation procedures are taking place on the central data storage, based on the conceptual differences of domain-specific building models, and therefore they do not have to be implemented during the import step. If a structural analysis tool imports a structural analysis representation of a model, the importing process is mainly reduced to mapping of information. This is the main benefit of the novel framework: a substantial and expensive effort of defining and implementing interpretation rules from an architectural model to a structural model is relocated from an importing tool to central storage. Thus the process of data import is simplified and the origin of most of the problems in this particular exchange are bypassed.

The framework was implemented with a test building model. New definitions for each element were created before the interpretation process could be developed. Implementation is only possible with clearly defined algorithms for element interpretation. The lack of standards suitable for automation of data exchange processes is the main obstacle for the new framework. Scalability issues of the whole framework were not considered at this stage of research. However, the idea behind the multiple domain-specific classifications is to reduce the scope of a standard defining the classification, which would also reduce the existing scalability issues (Lee and Jeong, 2012). The automation part of the framework and its scalability are the same as for the frameworks with open standards, because the communication with central storage needs to be implemented in each software tool individually. Interpretations could raise scalability issues when defined for multiple workflows. Similarities between the workflows need to be investigated in future research with real-world projects, and eventually new solutions would need to be developed if interpretations become difficult to maintain.

In summary, the automation of repetitive tasks during the model exchange is unavoidable, and it will significantly improve the data exchange between architectural design and structural analysis models (Ramaji and Memari, 2020). This research aims to contribute to the body of knowledge that bridges the interdisciplinary differences between architectural design and structural analysis models in a suitable framework.

9. CONCLUSION

This paper describes a new framework for overcoming the semantic and geometrical differences between the architectural design and structural analysis models. The novel framework encompasses three parts necessary for achieving automated exchange: classification, interpretation and automation. The proposed framework allows a significant simplification of the process of importing building elements to a structural analysis software tool because it is addressing a pre-processed architectural model with prepared geometries, avoiding the usually error-prone software tool-based interpretation process.

The research demonstrates the alternative approach to a data exchange framework and the results suggest that the complexity of information interpretation defined separately for each software tool combination could be avoided and moved to a central data management system. Multiple classification systems allow for the improvements in many aspects of the AEC industry by supporting domain-specific knowledge and allowing for more understandable central models for end users. Interpretations between different representations primarily serve for data exchange. Currently, the greatest challenge between the architectural and structural analysis data exchange are the geometrical interpretations which would have to be clearly defined for the new framework.

The proposal has several limitations: it was developed on the basis of only three building element types; the properties of elements were not included in the classification analysis; geometries and interpretations could

become very complex depending on the element's physical representation; and interpretations were defined only for architectural design and structural analysis representations. There are several possible technical improvements of the system architecture that could be considered: the semi-structured database could be replaced with a graph database; and other programming languages or the geometry kernel with additional methods and possibilities can be considered. However, the technical optimization is not the topic of this work: the system architecture is merely used for demonstration purposes. The framework implementation should eventually consider a more user-friendly interface, still leaving the methods understandable and open for end users.

The next step is the completion of the system for a proof-of-concept, involving more specific geometries and interpretation rules that need to take place in the central communication system. Implementing classification systems and interpretation rules into a framework will be further developed. The framework will be tested with multiple workflows and address the scalability issues. Implementation of system architecture needs to deal with the way to create, update and maintain interpretations, and on top of that offer a user-friendly interface. The approach discussed in this paper has implementation potential in various AEC domains, especially the ones using multiple concepts for the same building elements (e.g. energy performance analysis, structural analysis), but has potential even if only the filtering of centrally available data takes place (e.g. cost calculation, life cycle analysis).

ACKNOWLEDGEMENTS

This work was conducted as part of the Doctoral College 'Computational Design' at 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 part of Strabag AG.

REFERENCES

- ACCA software (2018). Klassifizierung und Kodierung von Informationen in BIM-Prozessen, *BibLus - Alles über BIM und Softwares für das Bauwesen* (Blog), 6 March. <http://biblus.accasoftware.com/de/klassifizierung-und-kodierung-von-informationen-in-bim-prozessen/>.
- Amann J. and Borrmann A. (2016). Embedding Procedural Knowledge into Building Information Models: The IFC Procedural Language and Its Application for Flexible Transition Curve Representation, *Journal of Computing in Civil Engineering*, Vol. 30, No.5. [https://doi.org/10.1061/\(ASCE\)CP.1943-5487.0000592](https://doi.org/10.1061/(ASCE)CP.1943-5487.0000592).
- Andrade J., Ares J., García R., Pazos J., Rodríguez S. and Silva A. (2008). Formal conceptualisation as a basis for a more procedural knowledge management, *Decision Support Systems*, Vol. 45, No. 1, 164-179. <https://doi.org/10.1016/j.dss.2007.12.015>.
- Borrmann A., Kolbe T., Donaubaauer A., Steuer H., Jubierre J. and Flurl, M. (2015). MultiScale Geometric-Semantic Modeling of Shield Tunnels for GIS and BIM Applications, *ComputerAided Civil and Infrastructure Engineering*, Vol. 30, 263-281. doi:10.1111/mice.12090.
- buildingSMART (2011). *Streamline IDM/MVD/Certification/Validation*. http://iug.buildingsmart.org/resources/abu-dhabi-iug-meeting/IDMC_016_2.pdf/view.
- Caetano I., Santos L. and Leitao A. (2020). Computational design in architecture: Defining parametric, generative, and algorithmic design, *Frontiers of Architectural Research*, Vol. 9, No. 2, 287-300. <https://doi.org/10.1016/j.foar.2019.12.008>.
- Davis C. A. and Laender A. H. F. (1999). Multiple representations in GIS: materialization through map generalization, geometric, and spatial analysis operations, *GIS '99: Proceedings of the 7th ACM international symposium on Advances in geographic information systems* (Medeiros C. B., editor), New York, NY: Association for Computing Machinery, 60-65. <https://doi.org/10.1145/320134.320151>.
- de Giorgio A., Lundgren M. and Wang L. (2020). Procedural knowledge and function blocks for smart process planning, *Procedia Manufacturing*, Vol. 48, 1079-1087. <https://doi.org/10.1016/j.promfg.2020.05.148>.
- Deng X. Y. and Chang, T.-Y. P. (2006). Creating Structural Model from IFC-based Architectural Model, *Joint International Conference on Computing and Decision Making in Civil and Building Engineering (ICCCBE)*,

- (Rivard H., Miresco E. and Cheung M.M.S., Eds.), Montreal, Canada, 3687-3695. <http://itc.scix.net/data/works/att/w78-2006-tf577.pdf>.
- Dikbas A. and Ercoskun K. (2006). Construction Information Classification: An Object Oriented Paradigm, *Proceedings of the 6th European Conference on Product and Process Modelling*, London, UK: Taylor & Francis Group plc, 317-326.
- Ekholm A. (1996). A Conceptual Framework for Classification of Construction Works, *Journal of Information Technology in Construction (ITcon)*, Vol. 1, 25-50. <https://www.itcon.org/papers/itcon.paper.1996-2.pdf>.
- Ekholm A. (2005). ISO 12006-2 and IFC – Prerequisites for Coordination of Standards for Classification and Interoperability, *Journal of Information Technology in Construction (ITcon)*, Vol. 10, 275-289. https://www.itcon.org/papers/2005_19.content.06433.pdf.
- Ekholm A. and Haggstrom L. (2011). Building Classification for BIM – Reconsidering the Framework, *Proceedings of CIB W78-W102 International Conference*. <http://portal.research.lu.se/ws/files/5731192/2201254.pdf>.
- Hofmeyer H. and Davila Delgado J.M. (2013). Automated design studies: Topology versus One-Step Evolutionary Structural Optimisation, *Advanced Engineering Informatics*, Vol. 27, No. 4, 427-443. <https://doi.org/10.1016/j.aei.2013.03.003>.
- Holzer, D. (2016). *The BIM Manager's Handbook: Guidance for Professionals in Architecture, Engineering, and Construction*, Chichester, England: Wiley.
- Holzer D., Tengono Y. and Downing S. (2007). Developing a Framework for Linking Design Intelligence from Multiple Professions in the AEC Industry, *Computer-Aided Architectural Design Futures (CAADFutures) 2007* (Dong A., Moere A.V., and Gero J.S., editors), Dordrecht, Netherlands: Springer, 303-316.
- Hu Z. Z., Zhang X. Y., Wang H. W. and Kassem, M. (2016). Improving interoperability between architectural and structural design models: An industry foundation classes based approach with web-based tools, *Automation in Construction*, Vol. 66, 29-42. <https://doi.org/10.1016/j.autcon.2016.02.001>.
- IAI International Council Limited (2019). *International Alliance for Interoperability, Industry Foundation Classes, IFC2x Edition 3*. <https://standards.buildingsmart.org/IFC/RELEASE/IFC2x3/FINAL/HTML/>.
- ISO (2000a). *1087-1: Terminology work - Theory and application*, Geneva, Switzerland: International Organization for Standardization.
- ISO (2000b). *ISO 704: Terminology work – Principles and methods*, Geneva, Switzerland: International Organization for Standardization.
- ISO (2007a). *12006-3: Building construction – Organization of Information about construction works – Part 3: Framework for object-oriented information*, Geneva, Switzerland: International Organization for Standardization.
- ISO (2007b). *860: Terminology work – Harmonization of concepts and terms*, Geneva, Switzerland: International Organization for Standardization.
- ISO (2013). *ISO 22274: Systems to manage terminology, knowledge and content – Concept-related aspects for developing and internationalizing classification systems*, Geneva, Switzerland: International Organization for Standardization.
- ISO (2015). *12006-2: Building construction – Organization of Information about construction works – Part 2: Framework for classification*, Geneva, Switzerland: International Organization for Standardization.
- Jorgensen K. A. (2011). Classification of Building Object Types: Misconceptions, challenges and opportunities. *Proceedings of the 28th International Conference of CIB W78*, Sophia Antipolis, France. <http://itc.scix.net/paper/w78-2011-Paper-24>.
- Kepplin R., Schnellenbach-Held M. and Held M. (2017). Building Information Modeling – Umsetzung in der Tragwerksplanung, *Bautechnik*, Vol. 94, 220-226. doi:10.1002/bate.201700003.

- Kovacic I., Vasilescu D., Filzmoser M., Suppin R. and Oberwinter L. (2015). BIM in teaching – lessons learned from exploratory study. *Organization, Technology & Management in Construction*, Vol. 3, No. 7, 1358-1366. doi:10.5592/otmcj.2015.3.3.
- Ladenhauf D., Battisti K., Berndt R., Eggeling E., Fellner D.W., Gratzl-Michlmair M. and Ullrich T. (2016). Computational geometry in the context of building information modeling, *Energy and Buildings*, Vol. 115, 78-84. <https://doi.org/10.1016/j.enbuild.2015.02.056>.
- Lee J. and Jeong Y. (2012). User-centric knowledge representations based on ontology for AEC design collaboration, *Computer-Aided Design*, Vol. 44, 735-748. <https://doi.org/10.1016/j.cad.2012.03.011>.
- Lee J., Jeong Y., Oh M. and Hong S.W. (2014). A Filter-Mediated Communication Model for Design Collaboration in Building Construction, *The Scientific World Journal*, Vol. 2014. <http://dx.doi.org/10.1155/2014/808613>.
- Lindgren J. and Widen K. (2018). Diffusing building information management – knowledge integration, mechanisms and knowledge development, *Architectural Engineering and Design Management*, Vol. 14, No. 5, 347-362. <https://doi.org/10.1080/17452007.2017.1394260>.
- Liu Z.-Q., Li Y.-G. and 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*, Vol. 17, No. 6, 1344-1350. <https://doi.org/10.1007/s11771-010-0640-z>.
- Lou E. C. W. and Goulding J. S. (2012). Building and Construction Classification Systems, *Architectural Engineering and Design Management*, Vol. 4, No. 3-4, 206-220. <https://doi.org/10.3763/aedm.2008.0079>.
- Mora R., Bedard C. and Rivard H. (2008). A geometric modelling framework for conceptual structural design from early digital architectural models, *Advanced Engineering Informatics*, Vol. 22, No. 2, 254-270. <https://doi.org/10.1016/j.aei.2007.03.003>.
- Pauwels P., Zhang S. and Lee Y.-C. (2017). Semantic web technologies in AEC industry: A literature overview, *Automation in Construction*, Vol. 73, 145-165. <http://dx.doi.org/10.1016/j.autcon.2016.10.003>.
- Pech A. and Kolbitsch A. (2005). *Wände*, Wien, Austria: Springer-Verlag Wien, ISBN 3-211-21498-4.
- Qin L., Deng X.-Y. and Liu X.-L. (2011). Industry foundation classes based integration of architectural design and structural analysis, *Journal of Shanghai Jiaotong University (Science)*, Vol. 16, No. 1, 83-90. <https://doi.org/10.1007/s12204-011-1099-2>.
- Ramaji I.J. and Memari A.M. (2016). Interpreted Information Exchange: Systematic Approach for BIM to Engineering Analysis Information Transformations, *Journal of Computing in Civil Engineering*, Vol. 30, No. 6, 230-246. [https://doi.org/10.1061/\(ASCE\)CP.1943-5487.0000591](https://doi.org/10.1061/(ASCE)CP.1943-5487.0000591).
- Ramaji I.J. and Memari A.M. (2020). Interpreted information exchange: implementation point of view, *Journal of Information Technology in Construction (ITcon)*, Vol. 25, 123-139. doi:10.36680/j.itcon.2020.008.
- Rasys E., Hodds M., Dawood N. and Kassem M. (2014). A Web3d Enabled Information Integration Framework for Facility Management. *Australasian Journal of Construction Economics and Building Conference Series*, Vol. 2, No. 1, 1-12. <https://doi.org/10.5130/ajceb-cs.v2i1.3763>.
- Rezgui Y., Boddy S., Wetherill M. and 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*, Vol. 43, 502-515. [10.1016/j.cad.2009.06.005](https://doi.org/10.1016/j.cad.2009.06.005).
- Romberg R., Niggl A., van Treeck C. and Rank E. (2004). Structural Analysis based on the Product Model Standard IFC, *Proceedings of International Conference on Computing in Civil and Building Engineering (ICCCBE)*, Weimar, Germany. https://e-pub.uni-weimar.de/opus4/frontdoor/deliver/index/docId/243/file/iccbe-x_028_pdfa.pdf.
- Sacks R., Ma L., Yosef R., Borrmann A., Daum S. and Kattel U. (2017). Semantic Enrichment for Building Information Modeling: Procedure for Compiling Inference Rules and Operators for Complex Geometry,

Journal of Computing in Civil Engineering, Vol. 31, No. 6. [https://doi.org/10.1061/\(asce\)cp.1943-5487.0000705](https://doi.org/10.1061/(asce)cp.1943-5487.0000705).

- Sanguinetti P., Abdelmohsen S., Lee J. M., Lee J. K., Sheward H. and Eastman C. (2012). General system architecture for BIM: An integrated approach for design and analysis, *Advanced Engineering Informatics*, Vol. 26, No. 2, 317-333, <https://doi.org/10.1016/j.aei.2011.12.001>.
- Scherer R.J. and Schapke S.-E. (2011). A distributed multi-model-based Management Information System for simulation and decision-making on construction projects, *Advanced Engineering Informatics*, Vol. 25, No. 4, 582-599. <https://doi.org/10.1016/j.aei.2011.08.007>.
- Schuler E. (2016). *Tragwerkslehre 3 – Tragwerkselemente*, Vaduz, Liechtenstein: Universität Liechtenstein. <http://www.gbd.at/skripten/TWL3.pdf>.
- Sibenik G. and Kovacic I. (2018). Proposal for a discipline-specific open exchange framework, *Proceedings of: 35th International Symposium on Automation and Robotics in Construction (ISARC)*, 570-577. <http://www.iaarc.org/publications/fulltext/ISARC2018-Paper128.pdf>.
- Sibenik G. and Kovacic I. (2020). Assessment of model-based data exchange between architectural design and structural analysis, *Journal of Building Engineering*, Vol. 32. <https://doi.org/10.1016/j.job.2020.101589>.
- Singh V. and Gu N. (2012). Towards an integrated generative design framework, *Design Studies*, Vol. 33, No. 2, 185-207. <https://doi.org/10.1016/j.destud.2011.06.001>.
- Singh V., Gu N. and Wang X. (2011) A theoretical framework of a BIM-based multi-disciplinary collaboration platform, *Automation in Construction*, Vol. 20, No. 2, 134-144. <https://doi.org/10.1016/j.autcon.2010.09.011>.
- Solihin W., Eastman C. and Lee Y. C. (2016). A framework for fully integrated building information models in a federated environment, *Advanced Engineering Informatics*, Vol. 30, No. 2, 168-189. <https://doi.org/10.1016/j.aei.2016.02.007>.
- Vangenot C., Parent C., and Spaccapietra S. (2002). Modelling and Manipulating Multiple Representations of Spatial Data. *Advances in Spatial Data Handling* (Richardson D.E. and van Oosterom P. Eds), Germany: Springer-Verlag Berlin Heidelberg. https://doi.org/10.1007/978-3-642-56094-1_7.
- Venugopal M., Eastman C.M., Sacks R. and Teizer J. (2012). Semantics of model views for information exchanges using the industry foundation class schema, *Advanced Engineering Informatics*, Vol. 26, No. 2, 411-428, <https://doi.org/10.1016/j.aei.2012.01.005>.
- Weise M., Katranuschkov P., Liebich T. and Scherer R. J. (2003). Structural analysis extension of the IFC modelling framework, *Journal of Information Technology in Construction (ITcon)*, Vol. 8, 75-95. <https://www.itcon.org/paper/2003/14>.
- Zhang X.-Y., Hu Z.-Z., Wang H.-W. and Kassem M. (2014). An Industry Foundation Classes (IFC) Web-Based Approach and Platform for Bi-Directional Conversion of Structural Analysis Models, *2014 International Conference on Computing in Civil and Building Engineering*, Orlando, USA: American Society of Civil Engineers, 390-397. <https://doi.org/10.1061/9780784413616.049>.
- Zhu J., Wang X., Chen M., Wu P. and Kim M.J. (2019). Integration of BIM and GIS: IFC geometry transformation to shapefile using enhanced open-source approach, *Automation in Construction*, Vol. 106. <https://doi.org/10.1016/j.autcon.2019.102859>.