



## EXCHANGE REQUIREMENT-BASED DELIVERY METHOD OF STRUCTURAL DESIGN INFORMATION FOR COLLABORATIVE DESIGN USING INDUSTRY FOUNDATION CLASSES

Huahui LAI<sup>1, 2</sup>, Cheng ZHOU<sup>3</sup>, Xueyuan DENG<sup>1\*</sup>

<sup>1</sup>*Department of Civil Engineering, School of Naval Architecture, Ocean and Civil Engineering,  
Shanghai Jiao Tong University, Shanghai, China*

<sup>2</sup>*Shenzhen Municipal Design & Research Institute Co., Ltd., Shenzhen, China*

<sup>3</sup>*Arup International Consultants (Shanghai), Shanghai, China*

Received 25 March 2018; accepted 06 March 2019

**Abstract.** In Architecture, engineering, and construction (AEC) collaboration, exchange requirements (ERs) vary in different projects with different platforms. In order to ensure the completeness and accuracy of data sharing and exchange for structural engineering in collaborative design, an ER-based delivery method was proposed to improve the delivery of structural design information. First, a process map of structural design was developed based on Information Delivery Manual (IDM). Within this process map, an ER Matrix of structural design was proposed to define information required by other disciplines at different stages. This matrix was composed of a set of required structural objects and their attributes, which were mapped to related Industry Foundation Classes (IFC) data. The mapping between the ER Matrix and IFC-based structural model data was implemented by an exchange model generation algorithm. Furthermore, a delivery tool was developed to define the ER Matrix in two ways, including user interface and XML-based language. A practical project was used to illustrate the utility of the proposed method. The results show that the proposed method using IFC is beneficial for structural information delivery.

**Keywords:** BIM, IDM, IFC, exchange requirement, structural engineering, collaborative design.

### Introduction

As an information technology revolution in the architecture, engineering, and construction (AEC) industry, Building Information Modeling (BIM) has been widely applied in collaborative design to improve the quality and productivity of building projects (Gilkinson, Kiviniemi, Raju, & Chapman, 2015; Juan, Lai, & Shih, 2017). During collaborative design, the information is continuously created, accumulated, and exchanged between heterogeneous software tools. To enhance collaborative design, it is essential to support information delivery between software tools with different internal data structures (National Institute of Building Sciences [NIBS], 2015). Some efforts have been made. However, when delivering information during collaborative design, interoperability issues still exist (Muller, Garbers, Esmanioto, Huber, Loures, & Canciglieri, 2017).

Some researchers have tried to establish frameworks and methodologies to improve information delivery in

specific domains. Lucas, Bulbul, Thabet, and Anumba (2013) conducted a case analysis to identify information links in a hospital setting, such as phase, format and granularity, and then proposed a framework to capture required delivery information for facility management. Cavka, Staub-French, and Poirier (2017) conducted a longitudinal research project for the process of developing and formulating BIM requirements to support project delivery and asset management, and developed a conceptual framework related to digital and physical products. Hu, Tian, Li, and Zhang (2017) adopted several BIM-based delivery technologies to digitalize and integrate MEP information at the operation and maintenance stage, such as the logic chain for MEP systems and the transformation from BIM to Geographic Information System (GIS). These studies provided several effective methods for delivering diverse building information, such as BIM files, 2D drawings, and monitoring records.

\*Corresponding author. E-mail: [dengxy@sjtu.edu.cn](mailto:dengxy@sjtu.edu.cn)

However, a large quantity of building information is stored in heterogeneous software tools. Using the open and neutral data schemas is commonly recognized as a need in practice (NIBS, 2015). For this purpose, Industry Foundation Classes (IFC), a comprehensive object-oriented data schema for BIM interoperability, was developed by buildingSMART. IFC covers all stages of building projects, and meets all business needs (Liebich, 2010). IFC schema is an open and neutral data format supported by various BIM software tools. As the version is updated from IFC1.0 to IFC4, it has become an international data standard in the AEC industry. Generally, BIM software tools require partial model data from other software tools for particular tasks (Oh, Lee, Hong, & Jeong, 2015). For example, to create a thermal model, a designer needs to extract specific information from the architectural model, such as building layout, spatial parameters, and building usage (Mitchell, Wong, & Plume, 2007). From the view of whole lifecycle, IFC schema is highly redundant for software tools (Won, Lee, & Cho, 2013). The method of information delivery should be standardized.

To address this issue, buildingSMART proposed Information Delivery Manual (IDM) and Model View Definition (MVD). IDM defines a set of information that must be contained in the exchange models for specific purposes, and MVD provides a guideline for specifying how to map exchange requirements (ERs) to particular IFC data (See, Karlshøj, & Davis, 2012). To define explicit concepts in the process of information delivery, Eastman, Jeong, Sacks, and Kaner (2010) defined specific procedures for developing IDM, and described in detail exchange models and exchange objects for use cases. This effort provided a guideline for IDM development. As the publication of IDM (International Organization for Standardization [ISO], 2010), IDM has been used in various projects for information delivery, such as precast concrete (Panushev, Eastman, Sacks, Venugopal, & Aram, 2010), geotechnical design and analysis (Obergrösser & Borrmann, 2012), and heating, ventilating and air conditioning (HVAC) design (Liu, Leicht, & Messner, 2012; Liu, Akinci, Berges, & Garrett, Jr., 2013). There are 101 IDM documents under the development by national organizations and company groups, and four IDMs have been approved (buildingSMART, 2013).

Some researchers have also tried to implement information exchange by integrating IDM and MVD. Lee, Park, and Ham (2013) proposed an extended Process to Product Modeling (xPPM) method to define IDM and MVD. The xPPM method could produce the IDM by using existing ERs and Functional Parts (FPs), and resolve the mapping between IDM and MVD. Lee, Eastman, and Solihin (2016) proposed an ontology-based approach to define accurate data modules for model views, and translated ontology-based IDM from OWL/XML to mvdXML. Some MVDs related to specific domains are being developed. Ramaji, Memari, and Messner (2017) extended current available MVDs to support information delivery of multi-story

modular building projects. Pinheiro et al. (2018) proposed an MVD-based method of information exchanges between BIM models and building energy performance simulation information. In the AEC industry, some existing commercial software tools can be used for BIM collaboration, but they may not be able to support specific user-defined ERs of information delivery (Rezgui, Beach, & Rana, 2013). To date, 29 MVDs have been submitted to IFC Solutions Factory (BLIS, 2018), such as architectural design to structural design and structural design to structural analysis, but partially supported by commercial software tools. For example, Revit 2018 is capable of exporting IFC models based on a set of MVDs, such as IFC2x3 Coordination View, IFC2x3 Basic FM Handover View, and IFC4 Reference View. In the structural domain, the CIMsteel Integration Standards (CIS/2) is the product data model for structural steel projects. To import structural steel information to other BIM applications, CIS/2 models need to be translated into IFC models (Lipman, 2009). It shows that the MVD related to the delivery of structural design for collaborative design is not yet defined in practice.

Although some methodologies were developed for IDM and MVD, they have not been easily implemented as expected (Kiviniemi, Tarandi, Karlshøj, Bell, & Karud, 2008). Some challenges of the implementation of IDM and MVD are summarized as follows:

- IDM and MVD seem to be complementary to each other. However, the boundary between IDM and MVD is blurred (Aram, Eastman, Sacks, Panushev, & Venugopal, 2010), easily leading to duplication efforts in IDM and MVD (Lee et al., 2013).
- Due to the tight connection between IDM and MVD, either IDM or MVD changes, and the other is required to update accordingly (Lee et al., 2013).
- When developing IDM and/or MVD, it is difficult for project users to handle the complex mapping between IDM and MVD elements, which generally needs the effort from industry experts.
- Due to the lack of automatic mapping between IDM and MVD, ERs are manually defined in some documents (Lee et al., 2016), and then MVD developers translate these ERs to model views. It is a time-consuming and ineffective task.
- There is no robust standardized method for specifying IDM ERs to MVD, which potentially results in different mappings (Lee et al., 2016).

In addition, according to the authors' cooperation experience with design companies, the required information for a designed business task may slightly vary in different projects. Consequently, users need to build exchange models with different objects and attributes for specific projects. Given these challenges, it is necessary to develop a common method for the automatic mapping between required information and original model data according to user-defined exchange requirements. Taking structural design of civil buildings as an example, an improved deliv-

ery method was proposed to define the required structural design information during collaborative design. First of all, to standardize the delivery of structural design information in collaborative design, the IDM-based process map (PM) was used to design the BIM-based delivery process. Subsequently, an ER Matrix, composed of user-defined exchange requirements, was developed for different exchange models defined in the proposed PM. The proposed matrix could be defined by using user interface (UI) and Extensible Markup Language (XML)-based language, in order to accommodate different projects. Furthermore, an exchange model generation algorithm was developed to support the delivery of structural design information. Finally, a practical project was presented to demonstrate the feasibility and validity of the proposed method.

### 1. Overview of the ER-based delivery method of structural design information

Figure 1 summarizes the current information delivery process based on IDM and MVD. IDM and MVD are important parts of the Information Exchange Framework. The first step is to define key elements in IDM according to business tasks, such as PM, ER, and FP (Wix & Karlshøj, 2010). A PM uses Business Process Modeling Notation (BPMN) to define a process flow for a particular task, including activities, participants and exchange models; ERs are a set of information needed for exchange models in non-technical terms; FPs provide a detailed technical specification for ERs, which is generally specified by IFC entities and attributes. These IDM elements defined by users are independent of any software tool (Aram et al., 2010). In Step 2, MVD developers define related entities and attributes according to IDM elements to form model views for software tools. According to these MVDs, software vendors develop corresponding IFC interfaces to extract the required information from the original model.

In summary, the current information delivery process is mainly divided into two parts: an IDM that identifies required information in a human-readable form for the particular task, and an MVD that integrates and translates IDM elements to corresponding model views, which can be supported by software tools. It is difficult for project users to deliver required information from a complete model because of additional translators based on IDM and MVD. To integrate ERs and software implementation, a method that allows users to define ERs and forms the required model is proposed to improve information delivery. Figure 2 illustrates the technical architecture of the proposed method for information delivery.

Step 1. In Figure 1, the elements including PMs, ERs, and FPs need to be defined by project users. Only the PM needs to be defined in this new method, and other elements can be implemented in the ER Matrix. Since too many notations exist in the BPMN standard, the proposed PM of structural design for collaboration is defined by using simplified BPMN notations, in order to improve the

accuracy and efficiency of PM definition. Subsection 1.1 presents the details of this PM definition.

Step 2. In the traditional method shown in Figure 1, MVD developers have to define IFC entities/attributes one by one for forming a corresponding MVD. To improve the efficiency of ER definitions, an ER Matrix of structural design was proposed. Users can define required objects and attributes according to different ERs of building projects. Through the exchange model generation algorithm, these required objects and their attributes can be mapped to the designated IFC data, avoiding manual data processing. Subsection 1.2 describes the proposed ER Matrix in detail.

Step 3. According to the defined MVD, software vendors have to develop corresponding algorithms to extract required model data from native models. However, the algorithms may vary depending on data formats of software. Unlike this step in Figure 1, the ERs in the proposed matrix can be mapped to related IFC entities and their attributes through an IFC-based algorithm, and then the target model is generated. A delivery tool has been developed to automatically generate the target model according to the ER Matrix. Subsection 1.3 introduces the mapping between the ER Matrix and IFC data.

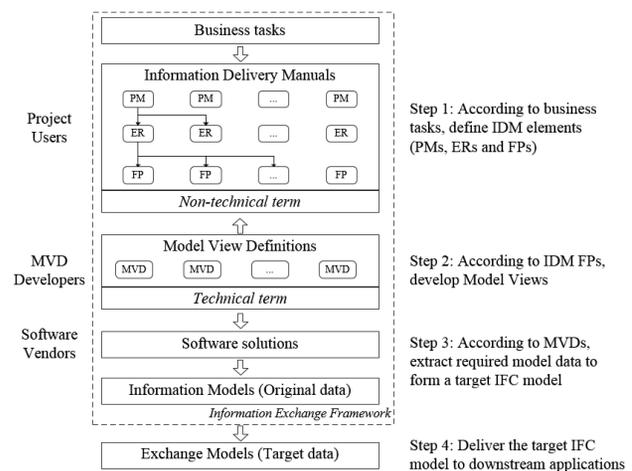


Figure 1. Current information delivery process based on IDM and MVD

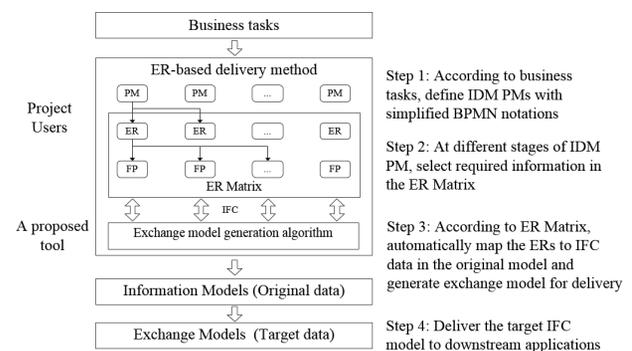


Figure 2. Improved information delivery process based on user-defined exchange requirements

Step 4. Similar to the last step in Figure 1, the target IFC model generated through the previous steps is delivered to downstream applications.

### 1.1. Defining the PM of structural design for collaborative design with simplified BPMN notations

As the first step in the proposed delivery process, the PM describes the relationships between activities, participants, and ERs. One of the methods used to define PM is the BPMN, which is recommended as a PM modeling notation in ISO 29481-1 (ISO, 2016). BPMN provides over 160 graphical notations for illustrating numerous meanings in the process map (Object Management Group, 2013), which is difficult to use in practice (Ko, S. S. G. Lee, & E. W. Lee, 2009; Recker, 2010). An analysis of 54 BPMN-based processes (Lee et al., 2013) shows that some misuses of BPMN notations do exist in practice. Hence, to ensure the accuracy of the PM, 22 frequently used notations (e.g., Activity, Intermediate Event (Message/Catching), and Gateway (Exclusive)) were suggested by Lee et al. (2013) to define the PM. In this study, these simplified BPMN notations were used to define the PM of structural design for collaboration, ensuring a human-readable and correct PM. These 22 BPMN notations are recommended for the use in PM, but not mandatory.

To ensure the validity and applicability of the PM in practice, the proposed PM of structural design related to civil buildings was proposed through literature review and interviews with several structural engineers who are experienced in civil engineering projects. Furthermore, some practical projects were employed to verify and improve the proposed PM. Figure 3 illustrates a portion of the PMs of structural design for civil engineering projects. The PMs are illustrated in several separate but interconnected diagrams. Figure 3(a) shows the overall PM of structural design for collaboration, and an example of the expansion of preliminary structural design in Figure 3(a) is illustrated in Figure 3(b).

In Figure 3, activities and ERs in the PM of structural design are located in different swimlanes, such as discipline-specific swimlane and ER swimlane. The activities which are represented by round-corner boxes are conducted by structural engineers at different stages. The plus sign in some activities (called Sub-process (Collapsed)) means that the detailed activities can be elaborated by another process map, such as the preliminary structural design in Figure 3(b). Some activities need to import the required information for getting started, and others may export specific deliverables for downstream activities. The imported and exported information is organized into different types of ERs. These ERs and activities are connected by the circle sign with an email symbol. The white-colored circle with an email (called Intermediate Event (Message/Catching)) is used to import ERs, and the black one (called Intermediate Event (Message/Throwing)) for exporting ERs.

For better recognition, an activity code is formulated in the form of “Discipline.Step”, where the Discipline means discipline codes for different disciplines (that is, A for architecture, S for structure, M for mechanical, E for electrical, and P for plumbing), and the Step uses digits to represent project stages in the PM. The S.1, for example, means the conceptual structural design. In some cases, one activity may contain several sub-activities, so newly added digits need to follow the Step. As shown in Figure 3(b), the S.3 contains S.3.1, S.3.2, S.3.3, etc.

In order to enable users to conduct activities in a standardized way, a table template for specifying each activity was designed. Based on the interviews with structural engineers, the tasks for each activity in the PM of structural design are documented in Table 1. In addition, Table 2 presents an example of the process specification in detailed structural design (S.5), including task, analysis, and ER. Because of the paramount importance of the analysis in structural design, the description about structural analysis is added into this table template (as shown in Table 2).

### 1.2. Developing ER Matrix for the delivery of structural design information

After defining the PM of structural design, various ERs which are imported to or exported from some activities need to be identified to form a target exchange model. For example, at the preliminary design stage, a structural engineer builds the preliminary structural model, and then generates some information required by the architect to form an exchange model. The exchange model includes several objects and attributes for architectural design, that is, the ERs. Thus, the exchange requirement is a key factor in information delivery.

As shown in Figure 3, the ERs are represented by color-coded files, and those at the same stage follow the same color. To standardize ER names and distinguish model versions, a coding mechanism was proposed with the form “Discipline\_ER.Version-Discipline\_ER.Version”. For example, the S\_ER.1-A\_ER.2 means that an exchange model is extracted from a structural model (version 1) to build an architectural model (version 2). The form “Discipline\_ER.Version” means a complete data model exported from the activity. Given the fact that data models from different disciplines need to be integrated for BIM collaboration, the form “Discipline-BIM\_Co” is used to represent these models.

Through a series of interviews with experienced structural engineers, structural objects of civil buildings are divided into three types: project, exchange, and discipline-specific elements (as shown in Figure 4).

- The project elements include the project and site. These elements are fundamental to create building models from different disciplines, and they remain the same within different models of one building project. Hence, these elements do not need to be delivered to other disciplines.
- The exchange elements contain the objects that need to be exchanged. Structural exchange elements

Table 1. Tasks for different activities in the PM of structural design

	Activity	Task
S.1	Conceptual structural design	<ul style="list-style-type: none"> <li>– Determine structural type (such as steel structure, concrete structure, and composite structure) according to the description and scope of the building project;</li> <li>– Determine structural system (such as frame structure and shear wall structure) according to the height, function, and structural type of the building project;</li> <li>– Determine spatial arrangement of beams, columns, walls, slabs, and supports according to the structural system of the building project.</li> </ul>
S.2	Conceptual structural design review	– Check the structural type determined at the conceptual design stage, and ensure its validity.
S.3	Preliminary structural design	<ul style="list-style-type: none"> <li>– Determine antiseismic requirements;</li> <li>– Determine structural loads;</li> <li>– Determine strength grade of materials;</li> <li>– Determine cross-sectional dimensions of structural components;</li> <li>– Complete the basic selection and preliminary settings;</li> <li>– Build the preliminary design model.</li> </ul>
S.4	Preliminary structural design review	<ul style="list-style-type: none"> <li>– Check the rationality of antiseismic information, load information, and member section at the preliminary design stage;</li> <li>– Check structural layout at the preliminary design stage.</li> </ul>
S.5	Detailed structural design	<p>Based on the preliminary design model, the structural engineer builds the detailed model in terms of detailed information at this stage:</p> <ul style="list-style-type: none"> <li>– For the steel structure, structural joints between components and local stiffening measures are designed in detail after selecting the section of stressed members;</li> <li>– For the concrete structure, when determining the section of members, member reinforcements and local structure measures need to be identified. A large number of structural joints in the building project are designed in detail.</li> </ul>
S.6	Detailed structural design review	– Check the detailed structural model and related drawings, and review the validity of reinforcements and structural joints.
S.7	Structural model for BIM collaboration	– Deliver the structural model for BIM collaboration between multiple disciplines.

Table 2. Process specification in detailed structural design

Detailed structural design (S.5)	
Task	<p>Based on the preliminary design model, the structural engineer builds the detailed model in terms of detailed information at this stage:</p> <ul style="list-style-type: none"> <li>– For the steel structure, structural joints between components and local stiffening measures are designed in detail after selecting the section of stressed members;</li> <li>– For the concrete structure, when determining the section of members, member reinforcements and local structure measures need to be identified. A large number of structural joints in the building project are designed in detail.</li> </ul>
Analysis	Structural analysis, such as the analysis for reinforcements and structural joints.
ER	Build a detailed structural model (S_ER.2), including the foundation, reinforcements, and structural joints.

include the beam, column, foundation, slab, stair, and wall. In the information delivery process, some designated exchange elements are integrated to form an exchange model for other disciplines.

- The discipline-specific elements refer to some elements which are only used in the specific task, and not required by other disciplines. The reinforcement is one of discipline-specific elements in the structural model at the detailed design stage, so it is contained in the S\_ER.2.

Figure 4 illustrates the required structural objects for different ERs. For example, the beams are required by all exchange models. It means that the beams in the structural model must be delivered to architects and MEP engineers for collaborative design.

In general, the ERs in IDM are manually defined one by one in an Excel table (Lee et al., 2016). It is difficult to reuse these ERs in other scenarios. Based on the required objects in Figure 4, the ERs of structural design information were documented in a matrix, which was called the ER Matrix for the delivery of structural design information. A portion of an example of the ER matrix is illustrated in Figure 5.

The ERs in Figure 5 follow the same color-coded as those in Figure 3. The cell with a minus sign (–) means that the object doesn't have the corresponding attribute. To determine whether the attribute is required or not in the exchange model, three types of options are designed: “R” for the required information, “O” for the optional, and the empty cell for the ones without any requirement.

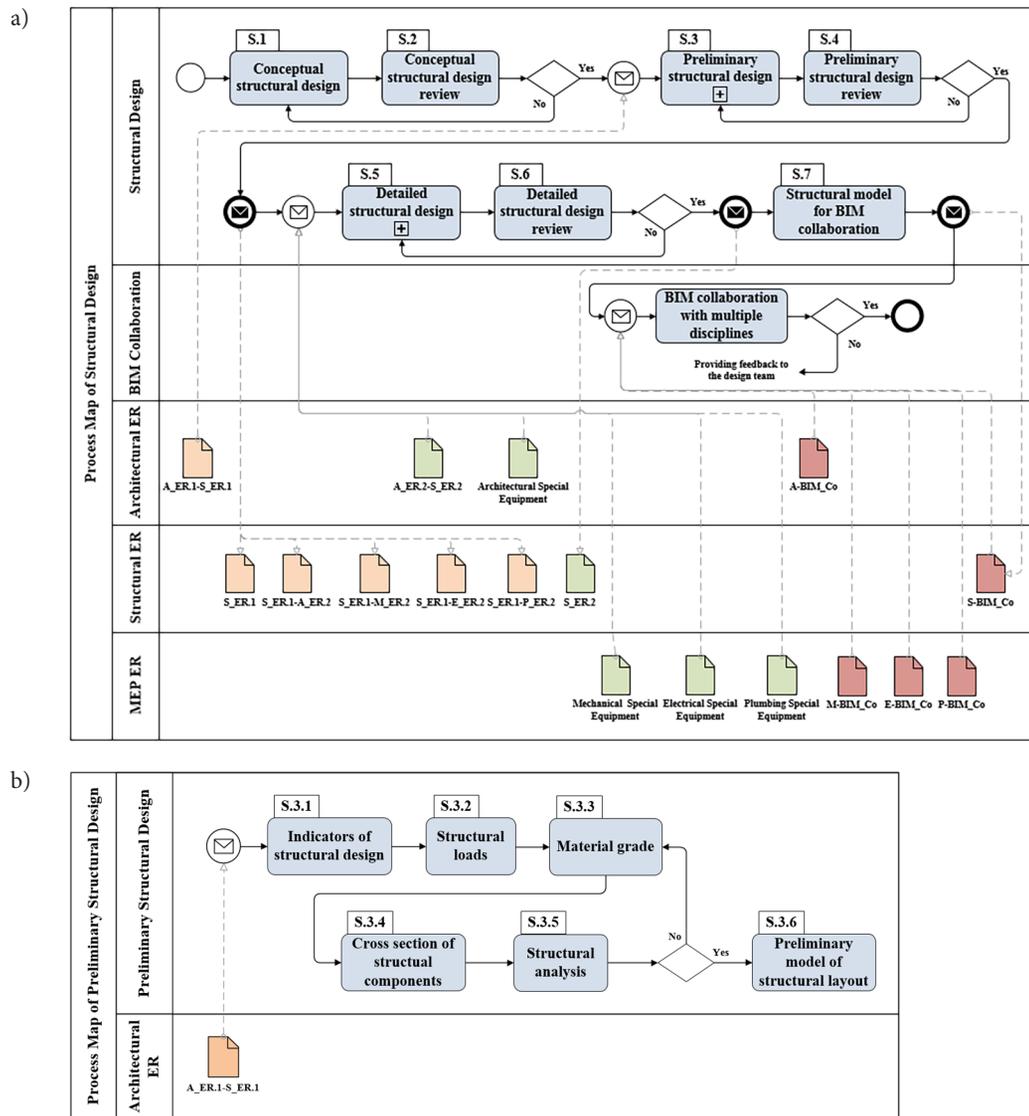


Figure 3. Process map of structural design for collaborative design: (a) Overall process map of structural design; (b) Process map of preliminary structural design

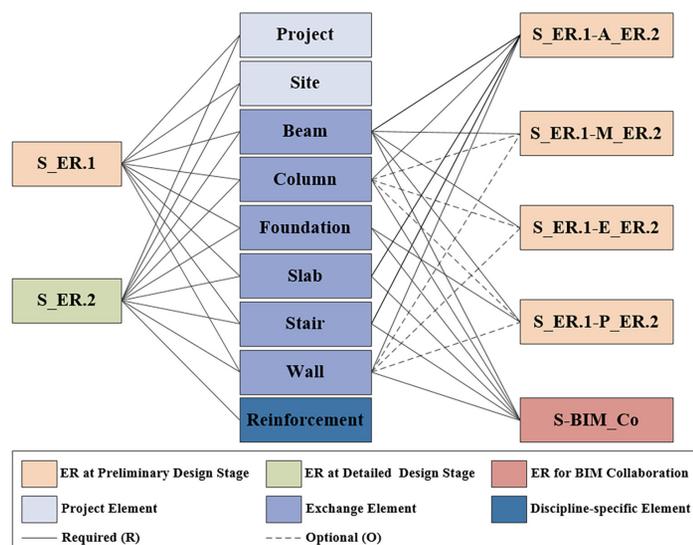


Figure 4. Structural objects for different exchange requirements

	S_ER.1								S_ER.1-A_ER.2								S_ER.2								S-BIM.Co										
	Project	Site	Beam	Column	Foundation	Slab	Stair	Wall	Reinforcement	Project	Site	Beam	Column	Foundation	Slab	Stair	Wall	Reinforcement	Project	Site	Beam	Column	Foundation	Slab	Stair	Wall	Reinforcement	Project	Site	Beam	Column	Foundation	Slab	Stair	Wall
Designer	R	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	R	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Discipline code	R	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	R	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Geographical location	-	R	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	R	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Site type	-	R	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	R	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Investigation info.	-	R	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	R	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
ID	-	R	R	R	R	R	R	-	-	-	R	R	R	R	R	R	-	-	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	
Object type	-	R	R	R	R	R	R	-	-	-	R	R	R	R	R	R	-	-	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	
Location	-	R	R	R	R	R	R	-	-	-	R	R	R	R	R	R	-	-	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	
Shape	-	R	R	R	R	R	R	-	-	-	R	R	R	R	R	R	-	-	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	
Cross section	-	R	R	R	R	R	R	-	-	-	R	R	R	R	R	R	-	-	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	
Length/ Height/ Thickness	-	R	R	R	R	R	R	-	-	-	R	R	R	R	R	R	-	-	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	
Material	-	R	R	R	R	R	R	-	-	-	R	R	R	R	R	R	-	-	R	R	R	R	R	R	R	R	-	O	O	O	O	O	O	O	
Strength grade	-	R	R	R	R	R	R	-	-	-	O	O	O	O	O	O	-	-	R	R	R	R	R	R	R	R	-	-	-	-	-	-	-	-	
Connection	-	R	R	R	R	R	R	-	-	-	O	O	O	O	R	-	-	-	R	R	R	R	R	R	R	R	-	-	-	-	-	-	-	-	
Reinforcement info.	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	R	R	R	R	R	R	R	R	-	-	-	-	-	-	-	-	

Figure 5. A portion of an example of ER Matrix for the delivery of structural design information

A detailed description of these options is listed as follows:

- Required (R): The information must be included in the ER, which forms an exchange model for delivering to downstream applications.
- Optional (O): The information is not mandatory in the ER. It means that the exchange model without this information still fulfills the requirements of downstream applications.
- Empty: The ER doesn't need the information for delivery. In this case, the exchange model without redundant information will be rapidly delivered to downstream applications, and processed by other software in a more effective way.

With efforts and contribution from structural engineers, an ER Matrix with these options (R/O/Empty) was developed as a default template for the delivery of structural design information. These options allow users to have a quick review of required information for collaborative design. On the other hand, after several practical applications, the users demand that the ER Matrix should be able to be modified for different projects, because structural ERs may vary in different exchange scenarios. Therefore, the ER Matrix was designed to be modifiable by project users according to requirements of different projects. The users do not have to “start from scratch” to design an ER Matrix, because they can modify the existing matrix template for other building projects.

In summary, this matrix provides an easier and more efficient way to define ERs, and the existing ER Matrix of structural design information can be modified for accommodating other projects. It is noted that (a) collaborative design mainly focuses on physical models of different disciplines, so there are no parameters of structural analysis (such as loads and forces) in this ER Matrix; (b) the current ER Matrix focuses on the delivery of structural design information on civil engineering projects, and the undefined ERs for other domains or projects need to be further analyzed to form the new ER Matrixes.

### 1.3. Mapping user-defined ERs to IFC data of the structural model

The purpose of an MVD is to translate the ERs to a model view, which can be supported by software tools. MVD definitions vary in manual processes defined by different users, and these different definitions may cause confusion between software developers when developing their IFC interfaces. It requires considerable effort and time from the industry experts and MVD developers to develop the transformation from IDM to MVD. Additionally, one MVD is designed specifically for a specific business task, and it is difficult for users to modify for meeting different data exchange requirements. To solve these problems, instead of a complete MVD, this study mainly focuses on IFC entities/attributes related to structural design information. Furthermore, the mapping between ERs in the proposed ER Matrix and corresponding IFC data was developed according to the user-defined options (R/O/Empty). It avoids a complex process to define MVD, and enables users to extract exchange model according to user-defined requirements.

To date, IFC2x3 has been widely supported by BIM software tools, while IFC4 is the newest official IFC release. Hence, IFC2x3 was adopted as the data format for the delivery of structural design information, ensuring the exchange model available in most BIM software tools. In addition, to be compatible with IFC4 for future extension, the mapping between structural ERs and IFC4 data was also studied in this paper. According to the proposed ER Matrix, the required information is divided into two types: overall information and object information.

#### (1) Overall information

The overall information includes general project information in the ER Matrix (e.g., designer, discipline code, geographical location, site type, and investigation information). They can be queried from the attributes of

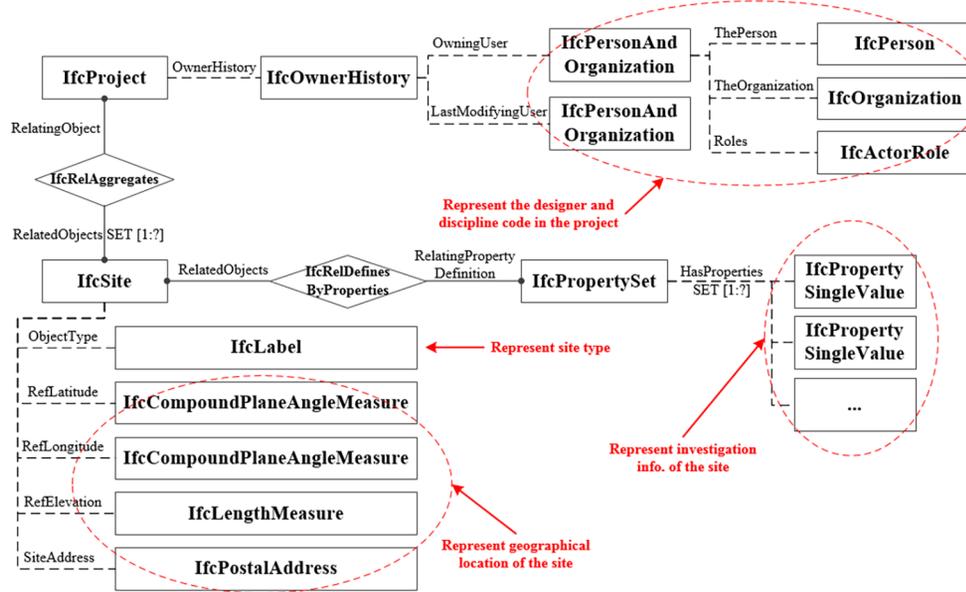


Figure 6. Required attributes in the IfcProject and IfcSite entities

“Project” and “Site”. According to the IFC schema, the “Project” and “Site” are represented as IfcProject and IfcSite, respectively. Figure 6 presents the entities IfcProject, IfcSite, and their attributes related to the ERs in the ER Matrix. Through reference relations, overall information can be mapped into the target IFC data. For example, the OwningUser attribute, which is represented by the IfcPersonAndOrganization entity, can describe designer information. When design information needs to be extracted, the IfcOwnerHistory entity will be queried through the IfcProject entity.

**(2) Object information**

As a rich schema, IFC can be used to represent structural objects and their attributes defined in the ER Matrix. According to structural elements in IFC schema, Table 3 shows the mappings between the required objects in the ER Matrix and IFC object entities. As mentioned above, the required objects were also mapped to the relevant IFC object entities in IFC4.

Each object has numerous attributes for business tasks. The ER Matrix of structural design provides the required attributes in the delivery of structural design information. According to the representation methods of diverse attributes in IFC schema, the required attributes were mapped to IFC attributes or referenced to other IFC entities, as shown in Table 4. The object location information, for example, is mapped into the attribute ObjectPlacement of IfcProduct. In IFC schema, each IFC object entity inherits this attribute from the IfcProduct entity. In an IFC model, the IfcShapeRepresentation entity is used to represent this attribute. Thus, the IfcShapeRepresentation entity will be queried from the target IFC object entity when extracting object location information.

Taking the IfcBeam entity as an example, Figure 7 illustrates the structure of its attributes by using EXPRESS-G language. The required attributes defined in the ER Matrix are also presented in Figure 7. The attributes of IfcBeam can be divided into two types: explicit attributes and those defined by inverse attributes. The explicit attributes are

Table 3. Definitions of required objects in the ER Matrix

Required object	IFC class (IFC2x3)	IFC class (IFC4)
Beam	IfcBeam.	IfcBeam, IfcBeamStandardCase.
Column	IfcColumn.	IfcColumn, IfcColumnStandardCase.
Foundation	IfcFooting, IfcPile.	IfcFooting, IfcPile.
Slab	IfcSlab, IfcRoof.	IfcSlab, IfcRoof, IfcSlabStandardCase, IfcSlabElementedCase.
Stair	IfcStair, IfcStairFlight.	IfcStair, IfcStairFlight.
Wall	IfcWall, IfcWallStandardCase.	IfcWall, IfcWallStandardCase, IfcWallElementedCase.
Reinforcement	IfcReinforcingBar, IfcReinforcingMesh, IfcTendonAnchor, IfcTendon.	IfcReinforcingBar, IfcReinforcingMesh, IfcTendonAnchor, IfcTendon.

Table 4. Definitions of required attributes in the ER Matrix

Required attribute	Target IFC data	Attribute value	IFC relation entity
ID	IfcRoot.GlobalId	String (22) Fixed	–
Object type	IfcObject.ObjectType	IfcLabel	–
Location	IfcProduct.ObjectPlacement	IfcLocalPlacement	–
Shape	IfcProduct.Representation	IfcShapeRepresentation	–
Cross section	IfcProfileDef	IfcPositiveLengthMeasure	–
Length/Height/Thickness	IfcExtrudedAreaSolid.Depth	IfcPositiveLengthMeasure	–
Material	IfcMaterial	IfcLabel	IfcRelAssociatesMaterial
Strength grade	IfcPropertySingleValue	IfcValue	IfcRelDefinesByProperties
Connection	IfcElement.ConnectedTo	IfcElement	IfcRelConnectsElements
Reinforcement info.	IfcElementAssembly	IfcElement	IfcRelAggregates

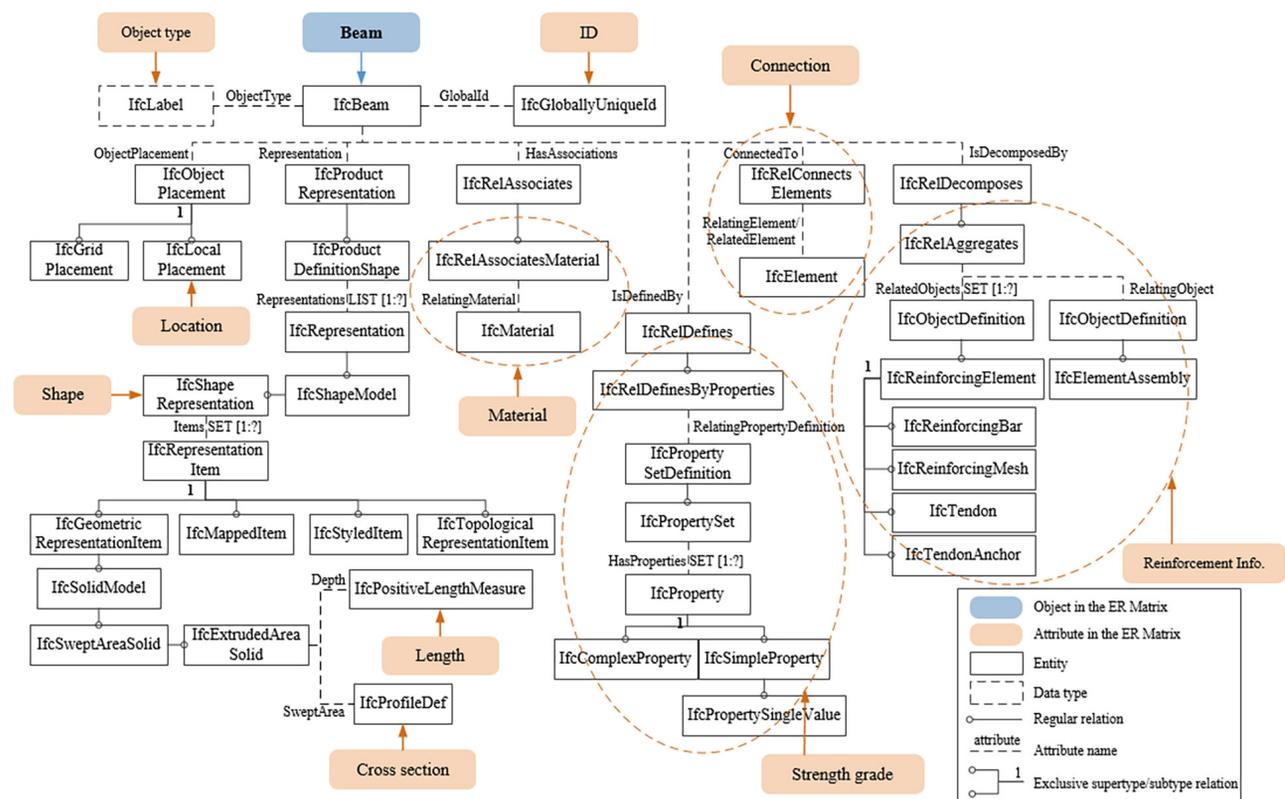


Figure 7. Required attributes of IfcBeam in EXPRESS-G language (IFC2x3)

directly queried from IFC instances referenced by IfcBeam, including ID, object type, location, shape, cross section, and length; and inverse attributes (including material, strength grade, connection, and reinforcement information) need to be represented by other IFC entities which are associated through IFC relation entities. For example, the beam material in the ER Matrix can be represented by the IfcMaterial entity, which is associated with IfcBeam through IfcRelAssociatesMaterial (as shown in Figure 7). The attributes of other structural objects can be extracted from the IFC model through the similar mapping mechanism.

According to the proposed mapping between ERs and IFC-based structural information, the required information can be automatically queried and exported from IFC

models. This data-processing process helps users avoid wasting time to understand the complex data structure in IFC schema.

## 2. Development of the delivery tool of structural design information

To automatically deliver the required objects and their attributes of structural models, this study developed an IFC-based delivery tool with four modules: IfcReader, Er2StruIfc, IfcWriter, and ReportWriter. As depicted in Figure 8, this method includes the following steps to generate exchange model of structural design:

1. Define the ER Matrix: the structural engineer can directly use or modify the existing ER Matrix to

meet the need of building project. The ER Matrix includes structural ERs and other ERs for collaborative design. This study provides two ways to define the ERs, that is, a UI and an XML-based language. Subsection 2.1 describes these two ways in detail.

2. Import an IFC data model: the structural engineer can use a structural software tool to build an original structural model according to structural ERs defined in the ER Matrix, and then store this native model using IFC format. This IFC data model will be imported into the proposed tool for generating the required model data.
3. Judge the IFC release: an IfcReader module was developed to interpret IFC data model. When the structural model is imported, its IFC release is judged to be IFC2x3 or IFC4, and then the IfcReader interprets different kinds of objects and their attributes from the original model.
4. Map the ERs to IFC data: according to the mapping between the ERs and IFC data, the Er2StruIfc module queries required information in the original IFC model. If the required information cannot be obtained, error messages will be collected in an error report. The exchange model generation algorithm in the Er2StruIfc module is elaborated in Subsection 2.2.
5. Export the exchange model or error report: through the IfcWriter module, the target IFC data will be integrated to form an IFC model. If the exchange model fulfills requirements in the ER Matrix, it will be delivered to downstream applications; if not, an error report generated by the ReportWriter module will be returned to the structural engineer for modification.

While the last three steps are conducted by corresponding modules, designers are only required to complete the first two steps. Within these two steps, designers are able to define the required information according to various civil engineering projects through the proposed ER Matrix, and it avoids a manual mapping between the ERs and IFC data. In the light of providing better support in the exchange model generation based on the user-defined ERs, the following subsections introduce (1) two ways to define the ER Matrix (that is, user interface and XML-based language), and (2) an exchange model generation algorithm in the Er2StruIfc module, which enables the mapping between the ERs and IFC data.

### 2.1. ER Matrix definition

This study mainly targets the definition of the ER Matrix in two interchangeable forms: an intuitive UI with effective interaction, and an XML-based language to define the ERs.

As mentioned above, a default ER Matrix is provided for the delivery of structural design information for collaboration. Through the proposed UI, the users can directly browse and use the default matrix to deliver target model data. As shown in Figure 9, the options (R/O/Empty) for all attributes of structural objects are presented when importing the ER Matrix. The ERs in various projects may be different from those in the default ER Matrix, so the UI can be used to modify the options of several ERs to meet the demand of the specific project. For the reuse in similar buildings, these user-defined ER Matrixes can be stored as templates, avoiding a repeated definition. Through this method, users have a better understanding of which information should be created in the structural model, and which information should be integrated into

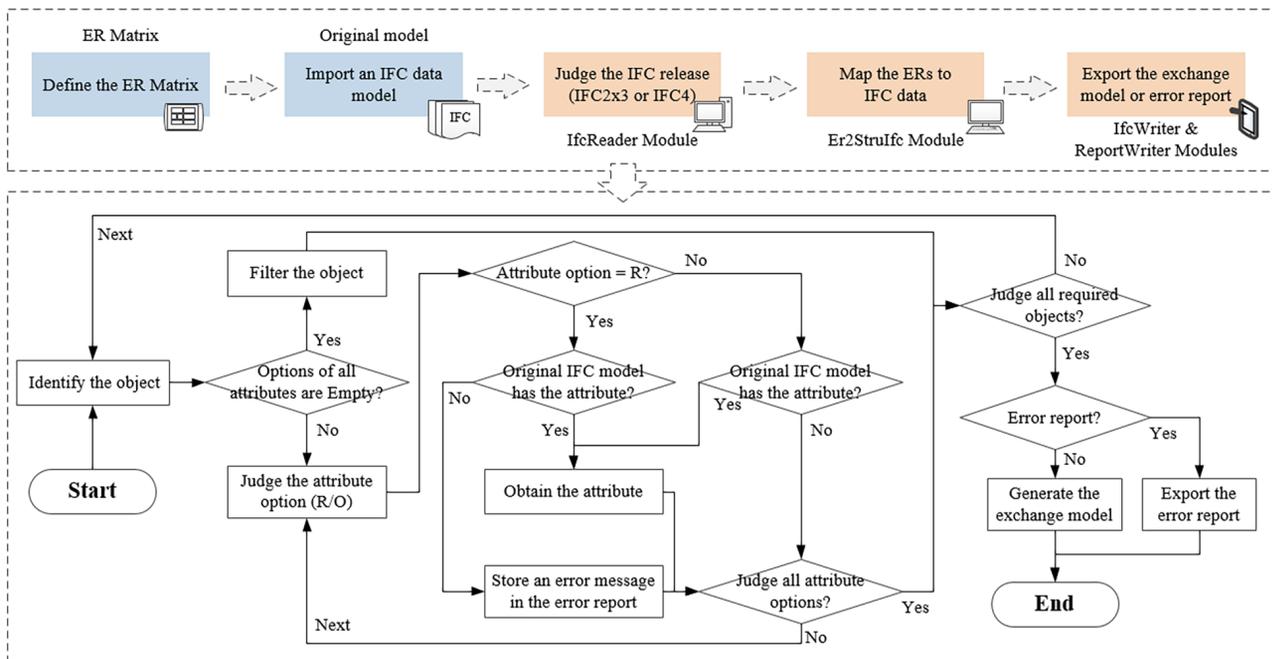


Figure 8. Flow diagram of the exchange model generation

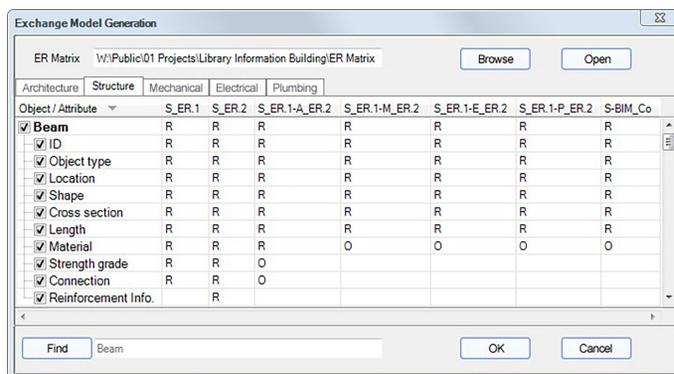


Figure 9. The user interface of exchange requirements of structural design information

exchange models for other disciplines. This UI has been embedded in the IFC-based platform developed by the authors.

In the future, more ERs may need to be added in the matrix for other cases, such as additional exchange requirements, the information from other disciplines (e.g., architecture), and other types of projects (e.g., infrastructure projects). There are mainly two methods to solve this problem: exhaustive matrix and extension mechanism of the matrix. It is difficult to establish a complete matrix for all ERs in the AEC industry, so the domain-specific matrixes have the potential to support specific cases, e.g., the delivery of structural design information in this study. Given the extension of the proposed matrix in future work, an XML-based query language has been developed to support the representation of semantic meanings in the AEC industry. XML is a searchable format, and a great many tools are available to read, write, and transform it. The proposed query language for all domains in the AEC industry is beyond the scope of this study, and will be described in detail in a separate paper. Thus, the XML-based language developed in this study focuses on structural ERs only.

Three types of elements are contained in the proposed ER Matrix, that is, object, attribute, and option. The “item” with three attributes (“Type”, “Match” and “Op”) was designed to represent these elements.

- “Type”-rule: an IFC model is comprised of plenty of information. The “Type” was designed to specify whether the target information belongs to the object or attribute. For example, if the “Type” is “Object”, the proposed algorithm only queries corresponding IFC object entities, which improves the efficiency of data query.
- “Match”-rule: after the type of the required information is determined, the name of the required object or attribute needs to be listed in the “Match”. In the original IFC model, only the IFC data which name matches the value of “Match” will be extracted to form exchange model.
- “Op”-rule: the attribute “Op” is an acronym for “Option”, the values of which are “R” and “O”. As mentioned above, the “R” and “O” are marked for



Figure 10. A portion of an XML-based file for the S\_ER.1-A\_ER.2

whether the object/attribute is required in the exchange model.

Furthermore, the “cascade” (this element is inspired by the PMQL (Adachi, 2002)) is used to bridge the relationship between the target object and required attributes, and its structure inherits from the “item”. The first step is to use the “Type” to define the target object, and then to define its required attributes through the “cascade”. Figure 10 illustrates a portion of an XML-based file which defines the ERs in the S\_ER.1-A\_ER.2.

## 2.2. Exchange model generation algorithm from the ERs to IFC data

After defining the ERs in the UI or the XML-based file, the next key step is to map these ERs to corresponding IFC data. The mappings from the user-defined ERs to IFC entities/attributes have been established in Subsection 1.3. The exchange model generation algorithm in the

Er2StruIfc module was implemented in C++. The proposed algorithm can automatically extract the required IFC data according to the mapping relationship in Figure 6 and Figure 7, and then generate the required exchange model. The workflow (as shown at the bottom of Figure 8) is summarized as follows:

1. According to the ER Matrix, this algorithm firstly identifies which object is required. If options for all attributes of the target object are “Empty”, this object will not be contained in the exchange model. The reinforcement, for example, is filtered in the S-BIM.Co according to the ER Matrix in Figure 5.
2. When the target object has an attribute which is marked with the R-symbol, this attribute of the IFC object entity in the original model will be obtained for the later exchange model generation. If the IFC object entity in the original model doesn't have the required attribute, an error message will be generated and stored in the error report.
3. If the option of the attribute is “O”, this attribute is not necessarily mandatory in the target object. For example, the material of the beam in S-BIM.Co. If the IfcBeam in the original model has material information, the algorithm will obtain corresponding IFC instances, such as IfcMaterial. If no material information is contained in the IfcBeam, the data processing flow will go to the next step without any error message.
4. After all attributes in one object have been queried and processed, the same process will repeat in the next target object, until all objects are processed. Finally, the target objects and their attributes are extracted from the original model to form an exchange model. The exchange model which completely fulfills the requirements in the ER Matrix will be delivered to downstream applications. If error messages are stored in the error report, only the error report is exported and returned to the original designer.

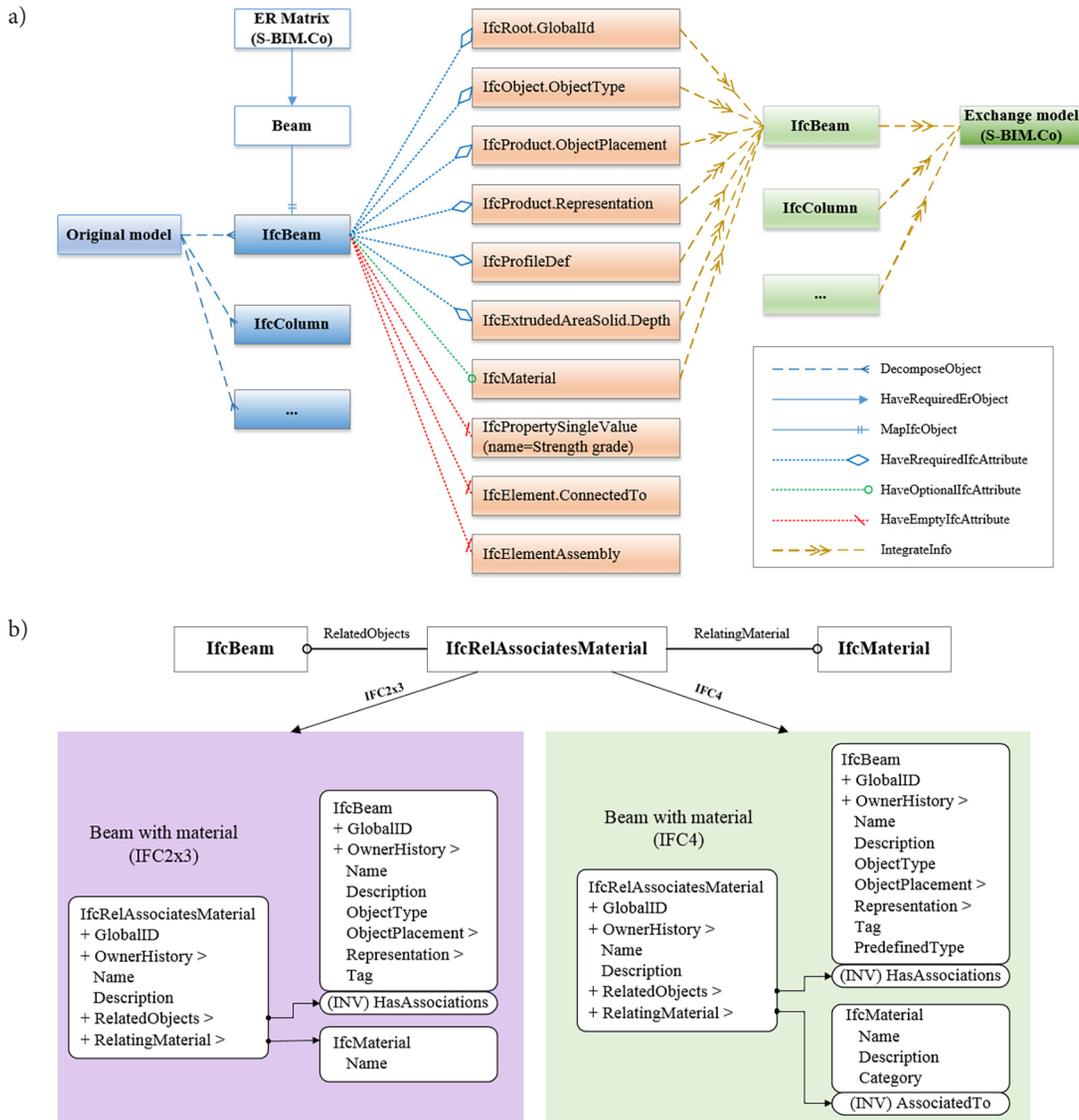


Figure 11. Data processing from the beam and its attributes to related IFC data through exchange model generation algorithm: (a) Data processing of the IfcBeam and its attributes in the S-BIM.Co; (b) Interrelationships between IfcBeam and IfcMaterial

As an example, Figure 11(a) illustrates the data process of an IFCBeam entity and its attributes in S-BIM.Co. First, the original model is parsed by the IFCReader module, and then the Er2StruIfc module reads the defined ERs of S-BIM.Co. Subsequently, the IFCBeam entity and its attributes are queried through the interrelationships, as depicted in Figure 7. Some attributes can be directly queried from IFCBeam. For example, the ID can be obtained from the GlobalID attribute of IFCBeam, an attribute inherited from IFCRoot. Other attributes are “identified relationally across multiple entities” (Lee et al., 2016). Figure 11(b) depicts the interrelationships between IFCBeam and IFCMaterial in two IFC releases. The IFCRelAssociatesMaterial entity specifies the relationship between the beam and the material. Through this IFC relation entity, the required material information can be obtained from the original model. Finally, all IFC object entities with required attributes will be integrated to form the S-BIM.Co by using the IFCWriter module.

For a better understanding of the Er2StruIfc module, Figure 12 shows the mapping between the ERs and IFC model data in terms of Table 3 and Table 4. The proposed algorithm has three steps for generating the exchange model. In the first step, the Er2StruIfc module interprets the object type. In Figure 12, the IFCBeam entity (that is, the instance #499) can be identified according to the mapping relationship in Table 3. The second step is to query the required IFC entities/attributes according to the defined ERs in the matrix. Taking the material as an example, its option is “O” in Figure 12. It means that the material information will be extracted, if it is defined within the beam. Through parsing the IFC model, the Er2StruIfc module obtains the instance #171 IFCMaterial through IFCRelAssociatesMaterial (#256). In the third step, the target IFC instances are integrated, and redundant information

(such as the instances #508 and #815) is removed. The remained IFC data can be exported as an IFC model.

### 3. Case study

In this section, a library building was selected to illustrate the utility of the proposed method. This project with an area of 47,293 m<sup>2</sup> includes a 9-story main building and three 4-story reading rooms. The main building is a concrete frame-shear wall structure, and reading rooms are concrete frame structures. Three discipline-specific models were built by ArchiCAD (architecture), Tekla Structures (structure), and MagiCAD (MEP). These software tools are certified for IFC2x3 Coordination View Version 2.0.

In current practices, the structural engineer builds a structural BIM model based on the architectural model, and then delivers the model to other designers (such as the architect and MEP engineer) along with some comments for design changes. Subsequently, other designers further build their discipline-specific BIM models according to the delivered structural model and comments, and return their own models and comments to related designers. This process is an iterative redesign for an agreed-upon solution. To ensure a more effective way to develop their own models, designers need to manually extract required objects and attributes from other models, which is time-consuming and error-prone. The proposed method was applied to this project, because no IDM/MVD related to the delivery of structural design information was released. Following the procedure in Figure 3(a), the following steps elaborate the delivery of structural design information for collaboration in this project.

1. After the structural type was defined in the conceptual design, the structural engineer received the exchange model A\_ER.1-S\_ER.1 from the architect for

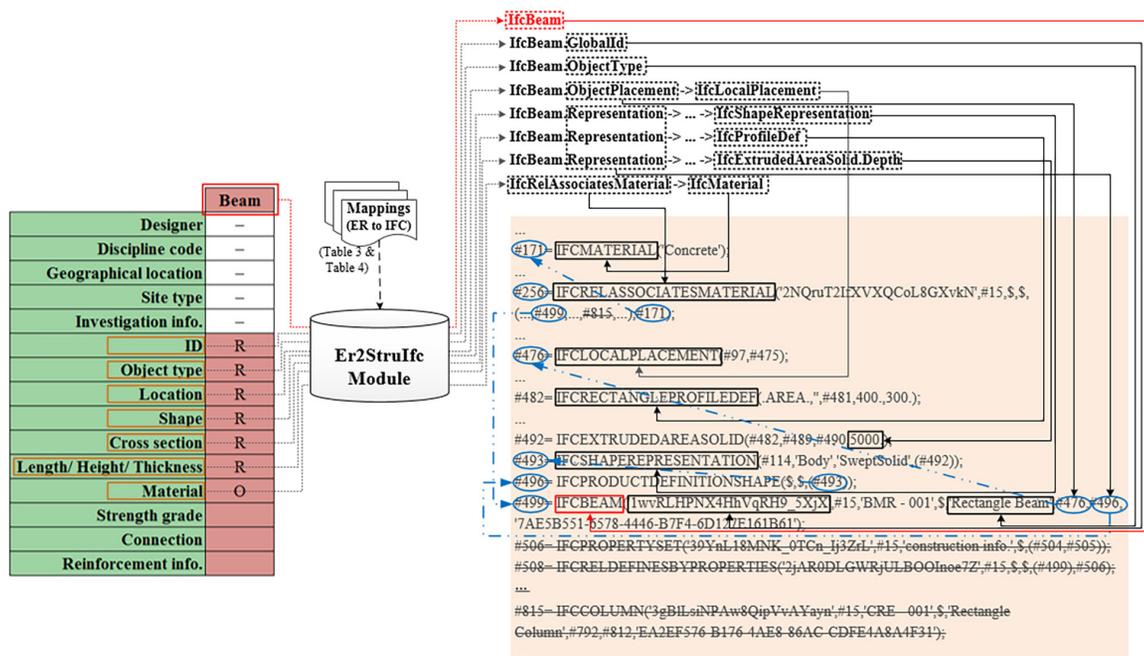


Figure 12. An example of the deployment of the Er2StruIfc module

preliminary structural design. The A\_ER.1-S\_ER.1 was automatically extracted from the architectural model A\_ER.1. According to the ER Matrix of preliminary structural design (Figure 5), the structural model S\_ER.1 was built with beams, columns, piles foundations, slabs, and shear walls.

2. The model S\_ER.1 was imported into the delivery tool for generating the exchange models for other designers. According to the user-defined ER Matrix, the required exchange models such as S\_ER.1-A\_ER.2 and S\_ER.1-M\_ER.2 were generated and delivered for collaborative design. The exchange model S\_ER.1-A\_ER.2, for example, was made up of beams, columns, slabs, shear walls, and their required attributes, as shown in Figure 13. The required objects and their attributes comply with the ERs defined in Figure 5.
3. According to the PM of structural design in Figure 3, before building the model S\_ER.2, exchange models from other disciplines needed to be imported along with the special equipment. In this library building, the load of bookshelves full of books is one of crucial structural load information for structural design, so the specific information was delivered to structural engineers. Subsequently, the structural model S\_ER.2 was built in detail. As shown in Figure 5, the reinforcements were added into each structural object of the S\_ER.2 compared with the S\_ER.1. In addition, more information and details for representation of the objects were also required in the S\_ER.2 according to the ER Matrix.
4. After fulfillment of the S\_ER.2 in detailed design, the model S-BIM.Co was generated from the S\_ER.2 according to the defined matrix (as shown in Figure 5). In this project, the S-BIM.Co was used for BIM collaboration with multiple disciplines, and some conflicts and disagreements were found. As a result, the requests for design changes were submitted to related designers.

Through the proposed method for the delivery of structural design information, structural engineers in this project easily understood which information was included in the structural models, and which information should be delivered to other designers. The deliverables from structural models which completely satisfied the ER Matrix of this project were automatically generated without manual data processing. Taking the S\_ER.1-A\_ER.2 and the S-BIM\_Co as examples, the accuracy and efficiency of the delivery of structural design information in collaborative design are further analyzed in the following section.

According to the ER Matrix in Figure 5, the S\_ER.1-A\_ER.2 and the S-BIM\_Co were extracted from the S\_ER.1 and the S\_ER.2, respectively. By using the IFC File Analyzer (National Institute of Standards and Technology, 2011), the numbers of the extracted objects were listed in Table 5. The results show that the numbers of required objects in the S\_ER.1-A\_ER.2 and the S-BIM\_Co were correct, and the objects with the “Empty” symbol were not found in the S\_ER.1 and the S\_ER.2. By defining ERs in the proposed matrix, only required structural objects were extracted from the original structural models, which showed the accuracy of the proposed delivery method.

Moreover, the proposed method enhanced the efficiency of data interoperability between structural engineers and other designers. Figure 14 presents file sizes of some structural models in this project.

The S\_ER.1-A\_ER.2 extracted from S\_ER.1 had a 39.35% reduction in the physical file size, and the decreasing rate of S-BIM\_Co reached 92.17%. According to the ER Matrix in Figure 5, the S\_ER.2 of this project had numerous reinforcements within each structural object. Unlike the regular structural objects, the geometric shapes of the reinforcements were represented by numerous IFC instances, resulting in an obvious increase in IFC file size. In the S\_ER.2, the reinforcement was represented by the IfcReinforcingBar entity, the geometry of which was represented by IfcSweptDiskSolid with many composite curve segments (represented by IfcCompositeCurveSegment

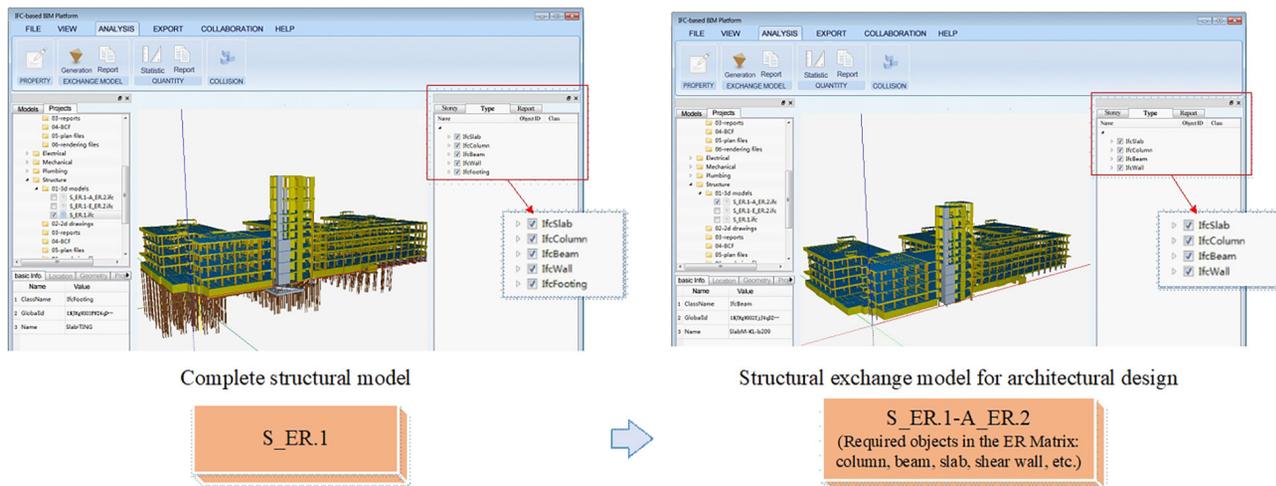


Figure 13. Information exchange requirements from structure to architecture

Table 5. Required objects in different exchange models during the delivery of structural design information

	S_ER.1	S_ER.1-A_ER.2		S_ER.2	S-BIM_Co	
	Number	Option	Number	Number	Option	Number
Beam	5,649	R	5,649	6,087	R	6,087
Column	1,299	R	1,299	1,594	R	1,594
Foundation	1,551		0	1,551	R	1,551
Slab	98	R	98	98	R	98
Wall	426	R	426	426	R	426
Reinforcement	0		0	11,337		0

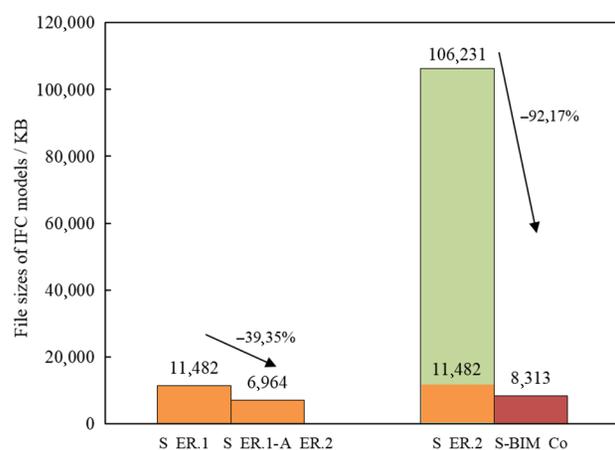


Figure 14. File sizes of IFC-based structural models

entity). However, the reinforcement was not required in BIM collaboration. As shown in Figure 5, the option of the reinforcement in S-BIM\_Co was “Empty”. Through the proposed delivery tool, numerous reinforcements were automatically filtered, and a lightweight model S-BIM\_Co was generated based on the S\_ER.2. After the filtration process, S-BIM\_Co had such a large decreasing in the file size. In contrast, the S\_ER.1-A\_ER.2 had almost the same structural objects as those in the S\_ER.1, except for the foundation. Consequently, the S\_ER.1-A\_ER.2 had a smaller decrease in the file size than the S-BIM\_Co did.

In this project, exchange models with required objects and their attributes were more easily used for collaborative design. The architect and MEP engineer could import the corresponding exchange models into ArchiCAD and MagiCAD for architectural and MEP design, respectively. These exchange models not only avoided manual extraction from complete structural models, but were also beneficial for data processing (such as data storage, management, sharing, and exchange) of software tools. If the model with a great deal of irrelevant information was delivered for collaborative design, it would lead to massive and non-value-adding work.

## Conclusions

In structural design, explicit exchange information required by other disciplines is of fundamental importance

in collaborative design. However, exchange requirements of structural design vary at different stages. This study proposes a delivery method with an ER Matrix, which allows users to define required information in the exchange models at different stages. The delivery method of structural design information has been applied to a case study of a library building, and the results demonstrate its feasibility and efficiency. Some conclusions are listed as follows.

1. The activities of structural engineering in collaborative design were summarized to design a PM of structural design. This proposed PM with 22 simplified BPMN notations is beneficial to be easily accepted. In order to resolve the confusion that BPMN notations are misused, these simplified notations are recommended to draw the PM for information exchange. In addition, a template for process specification was proposed to standardize the task, analysis, and ER in each activity of structural design.
2. The ER Matrix of structural design was proposed with three options (R, O, and Empty). This ER Matrix standardizes required objects and their attributes of structural design at different stages, avoiding inconsistent ER definitions. Through the non-technical term, exchange information of structural design can be easily defined. This matrix can also be extended to support information delivery for other disciplines.
3. To achieve the transformation from structural ERs to BIM models, the target objects and their attributes in the ER Matrix were mapped to corresponding IFC data. Unlike the mapping between IDM and MVD developed by software developers, the required BIM data can be automatically extracted from the original IFC model according to the ER Matrix. Besides IFC2x3, the proposed method is also capable of supporting the transformation between ERs and IFC4, thus ensures the delivery of exchange models based on IFC4 in the future.
4. A delivery tool with an exchange model generation algorithm was developed to automatically deliver structural design information extracted from original structural models. The tool provides two ways to define the ER Matrix, including user interface and XML-based language. The XML is a generic computer-interpretable language editable and

transformable by most software tools. In this study, the XML-based method supports the definition of structural design information.

The presented delivery method provides a user-defined way to mitigate the problem that there is no corresponding IDM/MVD for this delivery. Currently, its application is limited to structural design of civil engineering projects at different design stages. Furthermore, this study may have the following potential improvement:

1. The current ER Matrix still needs the user's identification or modification for the use in practice. To improve the efficiency of the proposed method, a query language is being developed to support an automatic extraction according to the user-defined semantic words. Through this language, the users only need to define several core words, and the algorithm can automatically query corresponding model data and export the required BIM model complying with the IFC.
2. This study can be extended to other disciplines for information delivery, such as architectural design and MEP design. Additionally, the proposed ER Matrix can be applied to other engineering domains in the AEC industry by extension, such as infrastructural and industrial engineering. It will improve information exchange in specific use cases where no relevant IDM/MVD exists.
3. The XML-based language developed for representing the ER Matrix can be supported by the XML-compliant software to extract the required information from their own models. The language can be transformed into the mvdXML, a generic format developed by buildingSMART to support MVD, which leads to broader support from software tools.

### Acknowledgements

This work was supported by the National Key Technologies Research and Development Program of China during the 13th Five-Year Plan Period under Grant 2016YFC0702001; and the China BIM Standard Research under Grant P-BIM01B00. This support is gratefully acknowledged.

### Author contributions

Xueyuan Deng and Huahui Lai conceived the study "Exchange requirement-based delivery method of structural design information for collaborative design using Industry Foundation Classes". Huahui Lai and Cheng Zhou were responsible for data collection and analysis. With the help of Xueyuan Deng, Huahui Lai developed the algorithm for the implementation of the proposed delivery method. Finally, Huahui Lai wrote the first draft of the article, and Xueyuan Deng revised it for publication.

### Funding

This work was supported by the National Key Technologies Research and Development Program of China during the 13th Five-Year Plan Period under Grant 2016YFC0702001; and the China BIM Standard Research under Grant P-BIM01B00.

### Disclosure statement

The authors declare that they do not have any competing financial, professional, or personal interests from other parties.

### References

- Adachi, Y. (2002). *Overview of partial model query language*. Retrieved from <http://cic.vtt.fi/projects/ifcsvr/>
- Aram, V., Eastman, C. M., Sacks, R., Panushev, I., & Venugopal, M. (2010). Introducing a new methodology to develop the information delivery manual for AEC projects. In *Proceedings of the CIB W78 2010: 27th International Conference*, 16–18 November 2010, Cairo, Egypt (pp. 49–59).
- BLIS. (2018). *The Model View Definition Site*. Retrieved from <http://www.blis-project.org/IAI-MVD/>
- BuildingSMART. (2013). *Overview of information delivery manuals independent of their status*. Retrieved from <http://iug.buildingsmart.org/idms/overview>
- Cavka, H. B., Staub-French, S., & Poirier, E. A. (2017). Developing owner information requirements for BIM-enabled project delivery and asset management. *Automation in Construction*, 83, 169–183. <https://doi.org/10.1016/j.autcon.2017.08.006>
- Eastman, C. M., Jeong, Y. S., Sacks, R., & Kaner, I. (2010). Exchange model and exchange object concepts for implementation of National BIM Standards. *Journal of Computing in Civil Engineering*, 24(1), 25–34. [https://doi.org/10.1061/\(ASCE\)0887-3801\(2010\)24:1\(25\)](https://doi.org/10.1061/(ASCE)0887-3801(2010)24:1(25))
- Gilkinson, N., Kiviniemi, A., Raju, P., & Chapman, C. (2015). Building Information Modelling: The tide is turning. *Structures and Buildings*, 168(2), 81–93. <https://doi.org/10.1680/stbu.12.00045>
- Hu, Z. Z., Tian, P. L., Li, S. W., & Zhang, J. P. (2017). BIM-based integrated delivery technologies for intelligent MEP management in the operation and maintenance phase. *Advances in Engineering Software*, 115, 1–16. <https://doi.org/10.1016/j.advengsoft.2017.08.007>
- International Organization for Standardization. (2010). *Building information models – Information delivery manual – Part 1: Methodology and format* (ISO 29481-1:2010). Geneva, Switzerland.
- International Organization for Standardization. (2016). *Building information models – Information delivery manual – Part 1: Methodology and format* (ISO 29481-1:2016). Geneva, Switzerland.
- Juan, Y. K., Lai, W. Y., & Shih, S. G. (2017). Building information modeling acceptance and readiness assessment in Taiwanese architectural firms. *Journal of Civil Engineering and Management*, 23(3), 356–367. <https://doi.org/10.3846/13923730.2015.1128480>
- Kiviniemi, A., Tarandi, V., Karlshøj, J., Bell, H., & Karud, O. J. (2008). *Review of the development and implementation of IFC compatible BIM*. Erabuild Project Report, Sintef S7716. Erabuild.

- Ko, R. K. L., Lee, S. S. G., & Lee, E. W. (2009). Business process management (BPM) standards: A survey. *Business Process Management Journal*, 15(5), 744-791. <https://doi.org/10.1108/14637150910987937>
- Lee, G., Park, Y. H., & Ham, S. (2013). Extended process to product modeling (xPPM) for integrated and seamless IDM and MVD development. *Advanced Engineering Informatics*, 27(4), 636-651. <https://doi.org/10.1016/j.aei.2013.08.004>
- Lee, Y. C., Eastman, C. M., & Solihin, W. (2016). An ontology-based approach for developing data exchange requirements and model views of building information modeling. *Advanced Engineering Informatics*, 30(3), 354-367. <https://doi.org/10.1016/j.aei.2016.04.008>
- Liebich, T. (2010). Unveiling ifc2x4 - the next generation of openBIM. In *Proceedings of the CIB W78 2010: 27th International Conference*, 16–18 November 2010, Cairo, Egypt (pp. 124-131).
- Lipman, R. R. (2009). Details of the mappings between the CIS/2 and IFC product data models for structural steel. *Electronic Journal of Information Technology in Construction*, 14, 1-13.
- Liu, Y., Leicht, R. M., & Messner, J. I. (2012). Identify information exchanges by mapping and analyzing the integrated heating, ventilating, and air conditioning (HVAC) design process. In *2012 ASCE International Conference on Computing in Civil Engineering*, 17–20 June 2012, Clearwater Beach, Florida, USA (pp. 618-625). <https://doi.org/10.1061/9780784412343.0078>
- Liu, X., Akinci, B., Berges, M., & Garrett, Jr., J. H. (2013). Extending the information delivery manual approach to identify information requirements for performance analysis of HVAC systems. *Advanced Engineering Informatics*, 27(4), 496-505. <https://doi.org/10.1016/j.aei.2013.05.003>
- Lucas, J., Bulbul, T., Thabet, W., & Anumba, C. (2013). Case analysis to identify links between facility management and healthcare delivery information in a hospital setting. *Journal of Architectural Engineering*, 19(2), 134-145. [https://doi.org/10.1061/\(ASCE\)AE.1943-5568.0000111](https://doi.org/10.1061/(ASCE)AE.1943-5568.0000111)
- Mitchell, J., Wong, J., & Plume, J. (2007). Design collaboration using IFC: A case study of thermal analysis. In *12th Computer-aided Architectural Design Futures Conference*, 11–13 July 2007, Sydney, Australia (pp. 317-329).
- Muller, M. F., Garbers, A., Esmanioto, F., Huber, N., Loures, E. R., & Canciglieri, O. (2017). Data interoperability assessment through IFC for BIM in structural design – a five-year gap analysis. *Journal of Civil Engineering and Management*, 23(7), 943-954. <https://doi.org/10.3846/13923730.2017.1341850>
- National Institute of Building Sciences. (2015). *National BIM standard - United States* (Version 3). Washington, D.C., USA.
- National Institute of Standards and Technology. (2011). *IFC file analyzer*. Retrieved from <https://www.nist.gov/services-resources/software/ifc-file-analyzer>
- Obergriesser, M., & Borrman, A. (2012). Infrastructural BIM standards-development of an information delivery manual for the geotechnical design and analysis process. In *eWork and eBusiness in Architecture, Engineering and Construction*, 25–27 July 2012, London, UK (pp. 581-587). <https://doi.org/10.1201/b12516-93>
- Object Management Group. (2013). *Business Process Model and Notation (BPMN)* (Version 2.0.2). Needham, MA, USA.
- Oh, M., Lee, J., Hong, S. W., & Jeong, Y. (2015). Integrated system for BIM-based collaborative design. *Automation in Construction*, 58, 196-206. <https://doi.org/10.1016/j.autcon.2015.07.015>
- Panushev, I., Eastman, C. M., Sacks, R., Venugopal, M., & Aram, V. (2010). Development of the national BIM standard (NBIMS) for precast/prestressed concrete. In *Proceedings of the CIB W78 2010: 27th International Conference*, 16–18 November 2010, Cairo, Egypt, paper 18.
- Pinheiro, S., Wimmer, R., O'Donnell, J., Muhic, S., Bazjanac, V., Maile, T., Frisch, J., & Van, Treeck, C. (2018). MVD based information exchange between BIM and building energy performance simulation. *Automation in Construction*, 90, 91-103. <https://doi.org/10.1016/j.autcon.2018.02.009>
- Ramaji, I. J., Memari, A. M., & Messner, J. I. (2017). Product-oriented information delivery framework for multistory modular building projects. *Journal of Computing in Civil Engineering*, 31(4), 04017001. [https://doi.org/10.1061/\(ASCE\)CP.1943-5487.0000649](https://doi.org/10.1061/(ASCE)CP.1943-5487.0000649)
- Recker, J. (2010). Opportunities and constraints: The current struggle with BPMN. *Business Process Management Journal*, 16(1), 181-201. <https://doi.org/10.1108/14637151011018001>
- Rezgui, Y., Beach, T., & Rana, O. (2013). A governance approach for BIM management across lifecycle and supply chains using mixed-modes of information delivery. *Journal of Civil Engineering and Management*, 19(2), 239-258. <https://doi.org/10.3846/13923730.2012.760480>
- See, R., Karlshøj, J., & Davis, D. (2012). *An integrated process for delivering IFC based data exchange*. Retrieved from [http://iug.buildingsmart.org/idms/methods-and-guides/Integrated\\_IDM-MVD\\_ProcessFormats\\_14.pdf/view](http://iug.buildingsmart.org/idms/methods-and-guides/Integrated_IDM-MVD_ProcessFormats_14.pdf/view)
- Wix, J., & Karlshøj, J. (2010). *Information delivery manual guide to components and development methods (IDMC 004)*. Retrieved from <http://iug.buildingsmart.org/idms/methods-and-guides>
- Won, J., Lee, G., & Cho, C. (2013). No-schema algorithm for extracting a partial model from an IFC instance model. *Journal of Computing in Civil Engineering*, 27(6), 585-592. [https://doi.org/10.1061/\(ASCE\)CP.1943-5487.0000320](https://doi.org/10.1061/(ASCE)CP.1943-5487.0000320)