TOWARDS STANDARDIZING DATA EXCHANGE FOR PRECAST CONCRETE BRIDGES

Najaf Ali, Dept. of Civil, Structural and Environmental Engineering, University at Buffalo, Buffalo, New York
Hanjin Hu, Dept. of Civil, Structural and Environmental Engineering, University at Buffalo, Buffalo, New York
Stuart S. Chen, Dept. of Civil, Structural and Environmental Engineering, University at Buffalo, Buffalo, New York

ABSTRACT

During the bridge life-cycle, a variety of information is created, modified, exchanged, and discarded. Bridge Information Modeling (BrIM) is being investigated to maintain the information in an accessible repository to leverage its use for multiple purposes among different software applications while avoiding error-prone re-entries of data manually. However, the lack of standardized data exchange protocols hinders the development of BrIM. Building Information Modeling (BIM) is being adopted increasingly by the building industry earlier than BrIM in the bridge industry. Typical BIM data models include the Precast NBIMS (National Building Information Modeling Standard) based on Industry Foundation Classes (IFC), CIMsteel Integration Standard Version 2 (CIS/2), and ISM (Integrated Structural Model). These data models, although not bridge-specific, provide at least a starting point for standardized bridge-oriented product data models to support data exchanges throughout the bridge life-cycle. This paper describes ongoing investigation into gap-analyzing and adapting earlier data models for the unique requirements of precast concrete bridge life-cycle data. The eventual goal of this ongoing investigation is the development of a recommended standard to facilitate precast concrete bridge data exchange and interoperability. Potential benefits of such a standard are illustrated in the context of a typical "workhorse" concrete girder bridge.

Keywords: BIM, BrIM, data exchange, interoperability, IFC, ISM

INTRODUCTION

Bridge Information Modeling (BrIM) is a key to understanding and managing the realities and complexities of the practice of design, construction and maintenance of a bridge. It utilizes data models that aim to fit together (interoperate) to describe and thus assist in managing every aspect of the design, construction and maintenance of bridges. BrIM helps in making possible the correlation of Idealized and Realistic bridge descriptions in models starting with the geometric and functional requirements.

BrIM is a powerful visualization tool as well. It allows us to visualize not only the finished product but also the construction sequence and thus not just rely on our imagination based on interpretation of 2D drawings. With BrIM, like BIM, we essentially construct and manage a bridge project digitally in a virtual environment. The structure can be spatially located in the precise global coordinates and orientation consistent with the roadway alignment so that its accurate geometry can be represented in the model in its context and a parametrically sensible manner.

In current practice, multiple applications with overlapping data requirements support various tasks of design and construction. Interoperability is the ability to exchange data between such applications, thereby smoothing workflows (e.g., by avoiding time-consuming error-prone manual data re-entry) and in some cases facilitating their automation[1].

Building Information Modeling (BIM) is being adopted increasingly by the building industry earlier than BrIM in the bridge industry. Key concerns of BIM have included interoperability and visualization, whereby consistent and accurate information can be communicated across the lifecycle process[1]. Industry Foundation Classes (IFCs), a file format developed by the International Alliance for Interoperability (IAI), remains a popular approach in supporting the exchange and use of data across technological platforms[2], although in the Transportation enterprise (where bridges find their “home”) information exchanges based on XML (eXtensible Markup Language) are of interest as well. Although shortcomings have been highlighted by various individuals and organizations regarding implications of IFC use on information technology systems[3], the IAI continues to improve the IFC framework. Therefore the possibilities associated with the application and use of a single file format across a global scale, is a goal that may be attainable[2]. The IFC specifications are currently administered by the buildingSMART[4] alliance.

From a technical point of view, IFC is defined using the ISO 10303[5] suite of specifications for data modeling and exchange, otherwise known as STEP (Standard for the Exchange of Product Data). STEP consists of a range of specifications, most notably a language for specifying data schemata (STEP/Express[6], in which the IFC language is defined), a mapping (Part-21[7]) for text-file representations of models conforming to that schema, a mapping (StepXML[8]) for XML file representation of models, and mappings to APIs for accessing models programmatically (notably Part-22[9], Standard Data Access Interface, or SDAI). Of these technology mappings, the most significant in terms of interoperability is currently the Part-21 mapping, which effectively defines the IFC's file format[4]. Although designers, builders, and maintainers of constructed facilities (like concrete bridges) may be unfamiliar with these, they are in fact quite familiar with the information needed to define interoperability requirements; that information is typically captured in an IDM.

An Information Delivery Manual (IDM) defines one or more Exchanges of BIM information in the context of reference industry processes. IDMs are defined by end users and practicing professionals and serve as the requirements definition for such BIM exchanges. A Model View Definition (MVD) is defined by the buildingSMART organization as "a subset of the IFC schema that is needed to satisfy one or many Exchange Requirements of the AEC industry."[10] A relevant precedent IDM in the concrete structures industry is Eastman et al. (2009), "Information Delivery Manual for Precast Concrete." This document defines a number of data exchange functional requirements and workflow scenarios for exchanges between architects, engineers, general contractors and precast fabricators in the building industry. That prior work, however, explicitly excluded bridges from the scope of its considerations.

The main purpose of developing IDMs and MVDs is to define the specifications for mapping the information exchange with the IFC model objects for orderly and standardized implementation in software interfaces. Then business rules are defined for the business processes involved in those exchanges. IFC implementations typically include deciding about use cases that should be supported in a specific project. As such, they need substantial knowledge about the BIM tools that will be used in the projects and their current IFC capabilities[11]. IFC development efforts incorporate requirements defined by industry experts[12].

The non-bridge precast concrete data exchange efforts[10] were carried out in order to establish a data schema for precast concrete in industries (e.g., architectural facades, parking garages, etc) that are distinct from bridges. The bridge industry has yet to define such a data schema in order to carry out the interoperability tasks between the disciplines involved in designing, constructing, operating and maintaining bridges. Some principal features that clearly distinguish bridges from buildings include the fact that unlike in the buildings industry where the geometry of a building is decided by an architect, the basic geometry of a bridge is established by highway designers. Instead of grid lines for columns, as used in the buildings industry, we have survey stationing of the highway along the highway grade line (HGL) in bridges. These are just a few examples of the differences.

A non-bridge data exchange approach should not be assumed to be applicable, "as is," to bridges, even though both are for concrete structures. Design of bridges involves a number of different approaches than design of buildings. There are also significant differences between the construction methods adopted by buildings and bridges industries. The interoperable items in the buildings industry, whether related to geometry or analysis and designand facility management, are often substantially different from those needed in the bridges industry. Furthermore, IFCs are not the only possible means of defining data exchange standards; ISM (Integrated Structural Model) provides an alternative approach. A careful gap analysis of IFC and ISM data schemata developed for the buildings industry reveals that some of the data items developed for buildings can also be used for bridges due to the common features between both the industries. However, these data items are not sufficient to eliminate the need for a separate data schema for the bridge industry[13].

Questions about legal aspects inevitably arise. The ultimate source of such questions is that now there is a model to manage, whereas pre-BIM there was no such model to manage (and thus no provision for such in typical pre-BIM contract language). The bridge industry can gain some insights regarding associated legal issues from related industries. For example, when IPD (Integrated Project Delivery) is used on a design and construction project, model AIA contract documents[14] presume the use of BIM and the changes it brings to the ways the design and construction portions of the project happen. The ConsensusDocs approach[15], on the other hand, is somewhat different (e.g., in allowing the parties to define who manages the model information). In both cases, revisions to default contract language can be expected as experiences are accumulated with BIM-enabled projects. Additional contract language adjustments will be needed moving forward to define or recognize the role(s) and usage of the BIM/BrIM model on a particular bridge design and construction project and the differences between bridge and building projects. Making these adjustments will facilitate the maximal exploitation of the potential advantages of BIM/BrIM. It should not be assumed that pre-BIM contract language and its underlying assumptions can effectively be used in a BIM/BrIM – enabled project without such adjustments.

PROCESS MAPPING

A process map, analogous to the process maps developed for precast concrete by Eastman et. al. (2009), is developed to characterize the workflow and data exchanges for precast concrete bridges. The horizontal rows in the process map containing data exchange models are called "swim lanes." The rows identify the "Disciplines" involved in the exchanges. These identify, organize and group data exchanges between Disciplines. The vertical columns identify project Phases such as TS&L (type, size and location) and Preliminary Structural Design. Within the cells created by swim lanes, white rectangles with rounded corners signify Activities. The purple box (DATA 1) identifies data hand-off practiced for a specific case study whereas the yellow box (B3) identifies the exchange of comments regarding a specific hand-off between two Disciplines. The appropriate Discipline's row and project Phase column identifies the context of the exchange[1].

As shown in a representative portion of the process map in Fig. 1, green blocks in the Exchange lanes designate Information Exchanges and are called Exchange Models (EMs). The process map shows several different kinds of exchanges. For each of the EMs, the working group (of human participants representing relevant disciplines or stakeholders) provides detailed specifications for the content of each exchange. This functional specification must list the types and extent of entities, their geometry, attributes, level of detail, material or processes that are needed for passing (exchanging) from one application to another[1].

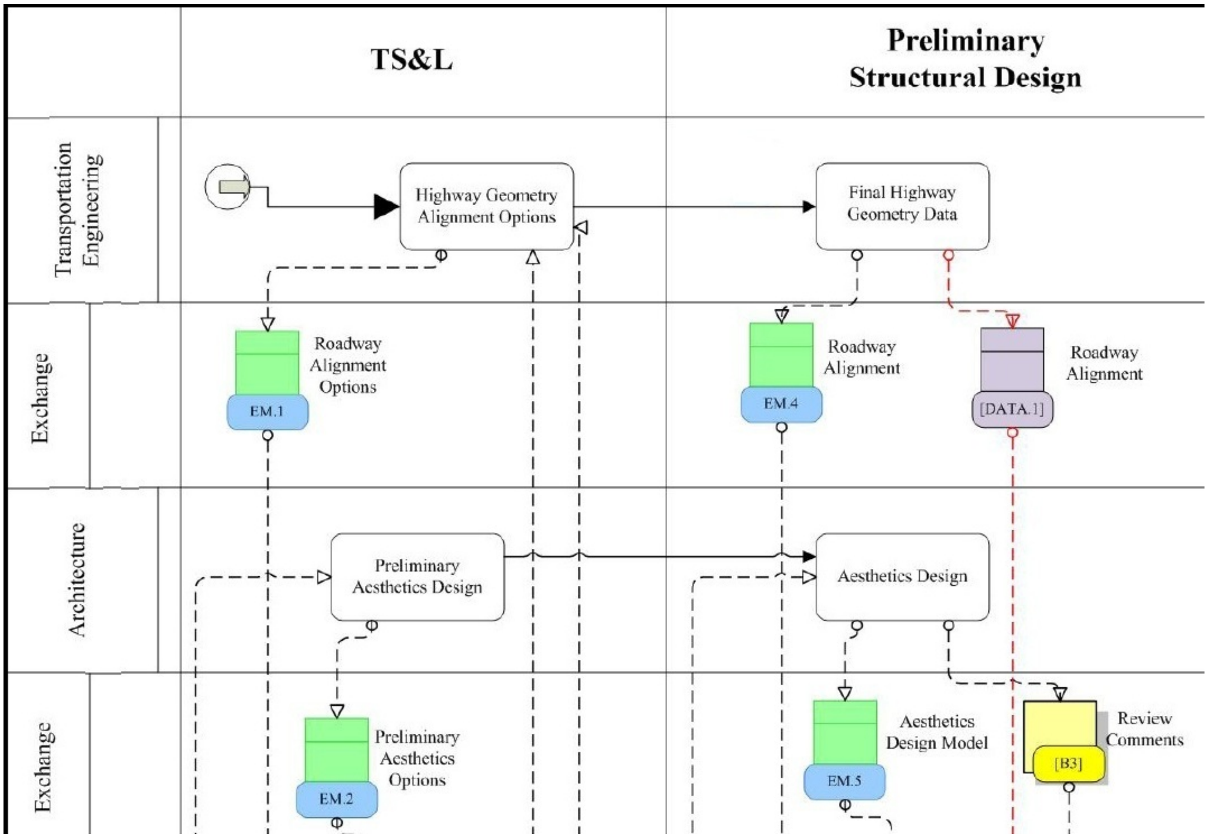

Fig. 1 Representative portion of Process Map for Precast Concrete Bridges

ILLUSTRATIVE EXAMPLES OF DATA EXCHANGE

The Process Map facilitates imagining a well-oiled machine in which the following occur. All of the individual stakeholders (disciplines) in all of the phases of a facility's lifecycle work in familiar ways within their own specialty areas. They are able to gather information, explore options, assemble, test, and perfect the elements of their work within a computer-based model before committing their work to be shared with or passed on to others, to be built, or to be operated, maintained etc. Imagining further that when it becomes necessary to share or pass a bundle of information to another organization, which may or may not be using the same tools, or to move it on to another phase of work, it is possible to safely and almost instantaneously (through a computer-to-computer communication) share or move just the right bundle of information without loss or error and without giving up appropriate control. In this envisioned future world the exchange is standardized across the entire (bridge) industry such that each item is recognized and understood without the parties having to create their own set of standards for that project team or for their individual organizations. Finally in this envisioned integrated scenario, for the life of the facility every important aspect, regardless of how, when, or by whom it was created or revised, could be readily captured, stored, queried, and recalled as needed to support real property acquisition and management, inspection and maintenance, operations including load rating, routing, and OS/OW (oversize/overweight) vehicle permitting, rehabilitation, new construction, and analytics[16]. In the bridge world such a scenario would presumably include updates to NBI (National Bridge Inventory) and other related reporting and programmatic efforts.

Software interoperability is clearly needed to facilitate the brave new world envisioned above. Interoperability requires seamless data exchange at the software level among diverse applications, each of which may have its own internal data structure. Interoperability is achieved by mapping parts of each participating application's internal data structure to a universal data model and vice versa. If the employed universal data model is open, any application can participate in the mapping process and thus become interoperable with any other application that also participated in the mapping. A standards-based approach to nteroperability would eliminate the costly practice of custom-integrating every individual software application (and version thereof) with every other individual software application (and version thereof)[16].

Such an exercise is carried out in the following section in order to achieve interoperability between different bridge related software applications and provide a glimpse into some of the mechanics of alternate approaches to achieving interoperability. Fig. 2 shows a portion of the process map where a data exchange takes place between the “Structural Engineering" and “Detailing” phase. Using two particular data exchange models for illustration, i.e. EM.8 and EM.9 between the design and detailing phase, an iteration of the structural system could easily take place in the process of generating a final detailed design. Such an exchange between the two mentioned phases is illustrated in detail in the examples, utilizing 3 distinct approaches to implement that exchange.

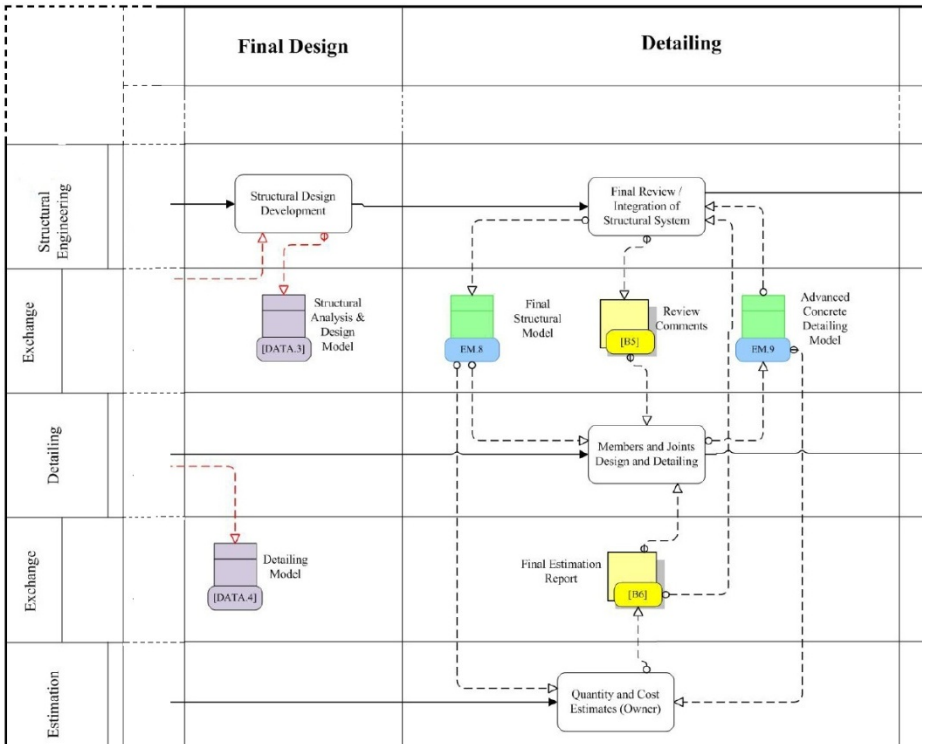

Fig. 2 Data Exchange Models

Table 1 summarizes key aspects of 3 distinct approaches explored herein to illustrate, compare, and contrast data exchange approaches for concrete bridges. The particular exchange used for illustrative purposes is between a concrete bridge analysis and design software application and a detailing application. As such, the interoperability issues arising between analysis models and physical models are quite relevant to the discussion.

Table 1 **Concrete Bridge Data Exchange Approaches**

Aspect	**Ad-Hoc Hard-Coded**	**Standards Based**		**Remarks**
		ISM	IFC	
ID	1	2a	2b	
models	Parametric to physical	Analytical & physical combined	Analytical & physical separated	
Analysis & Design App	MathCad (design checks)	LEAP Bridge	SAP2000	>1 provider
Detailing App	Tekla Structures	ProConcrete	Tekla Structures	>1 provider
Round-Tripping Possible?	No (uni-directional only)	Yes	Yes*	Need this for data integrity
# Translators	n*(n-1)	2n	2n	

Req'd for n Apps				

Note:

*IFC mechanisms provide support for round-tripping, but user verification of software application implementations thereof are ongoing.

The principal disadvantage of the Ad-Hoc Hard-Coded approach is the number of translators required. Consider, for example, the 6 software applications A through F shown in Fig. 3 each typically with its own proprietary internal format. The number of import/export translators is (6*(6-1)) = 30. Adding merely one more application, assuming interoperability needs among all applications, requires another 12 translators ((7*(7-1) = 42)! Clearly, there is a need for information exchange standards in a context as diverse as the entire life cycle of a concrete bridge structure.

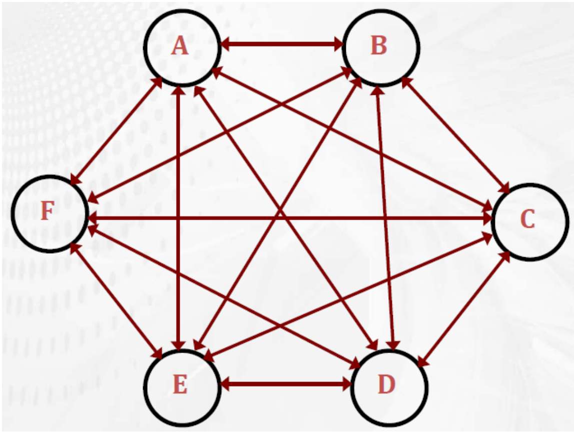

Fig. 3 Data Exchanges among Six Software Applications (courtesy of R. Lipman)

DATA TRANSFER FROM MATHCAD TO TEKLA (APPROACH 1)

In efforts to streamline the design to detailing BrIM model process, a C# code has been developed to export desired parameters and members from MathCAD to Tekla Structures using the Tekla Application Programming Interface (API). C# (pronounced "C-sharp") is an object-oriented programming language that aims to combine the computing power of C++ with the programming ease of Visual Basic. C# is based on C++ and contains features similar to those of Java[17]. As shown in Fig. 4, a TXT file that includes size, shape, and location in 3D space of key members and parameters is extracted from MathCAD's worksheets being used for design purposes. This TXT file is then read by the C# code and is fed to Tekla Structures using the Tekla API. Within the Tekla Structures 3D environment, the data transported from the MathCAD worksheets is read and an intelligent 3D model is created in space that contains information such as member cross-section, length, concrete strength, finished surface and material quantity. Once the 3D model is generated, the information related to the structure is readily accessible at any point onwards in the project; quantity takeoffs and shop drawings are simply reports extracted from this model.

An example of the C# code that generates 3D-model of a beam from the TXT file extracted from MathCAD is shown in Appendix A. This approach is clearly not generalizable and is clearly unidirectional (i.e., no “round-tripping”).

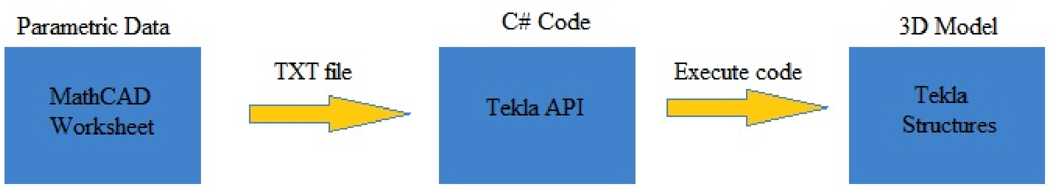

Fig. 4 Data transfer from MathCAD to Tekla

Fig. 5 shows a 3D-model of a bulb tee girder (BT-72), like that used in the subject bridge superstructure created in Tekla Structures using C# code from a TXT file provided in Appendix A along with the C# code.

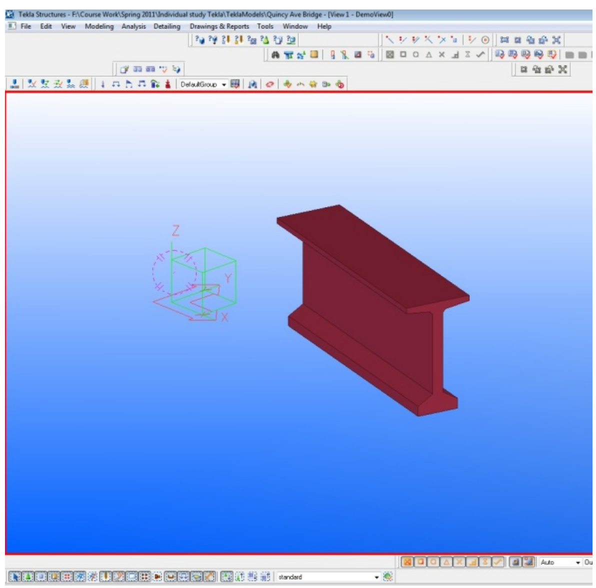

Fig. 5 BT-72 3D-model generated using C# code

INTRODUCTION TO ISM

The Integrated Structural Modeling (ISM) approach is a largely building-oriented application developed by a major software solution provider for data exchange and software interoperability. It consists of two parts, the ISM schema and the ISM application called the Structural Synchronizer.

The ISM schema is created to share structural engineering project information through the project life cycle, including structural modeling, analysis, design, drafting, and detailing phases[1]. It provides interoperability not only between the Bentley software applications themselves, but also between the Bentley products and other AEC software products or other schemas. It can be accessed through ISM Application Programming Interface (API), which has 4 classes, over 100 interfaces and about 40 enumerations (a complete, ordered listing of items in a collection).

The ISM enabled applications, the Structural Synchronizer, is used for two purposes: coordination of the information between applications and model viewing. It provides the capabilities of version control, which can detect differences between ISM models[18].

The ISM has been chosen as a methodology of interest in this research because it is an active and extensible schema for data exchange and software interoperability. It supports multiple software products, some of which are widely used in the bridge industry. Furthermore, its development is being significantly influenced to incorporate features needed for bridges that were not in the original, largely buildings-oriented, release of ISM.

DATA EXCHANGE USING ISM (APPROACH 2a)

The data exchange of this same use case, bridge design to bridge detailing, can be implemented by using two software applications and one data schema,. They are LEAP Bridge Enterprise V8i, ProConcrete V8i and the ISM V2. In this deployment, LEAP Bridge and ProConcrete are used for bridge analysis and design and bridge detailing, respectively.

The procedure of the data exchange is described as follows. LEAP Bridge is the data entry point for bridge design (including highway geometry import). Design parameters such as superstructure configuration, component sections and material properties can be input to LEAP Bridge to build an analytical model of the case study bridge. After analysis and design of the case study bridge, design results such as member sizes and reinforcement are transferred from LEAP Bridge to the ISM repository, which is a stand-alone ISM application. The ProConcrete detailing application can read the bridge data from the repository and create a physical model for detailing, as shown in Fig. 7.

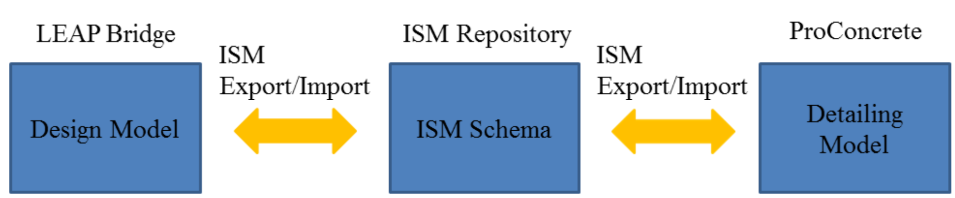

Fig. 6 Procedure of Data Exchange Through ISM

This data transfer through this process is done by using the ISM file and the ISM repository. There is, for all practical purposes from the user’s viewpoint, a direct link between the ISM and ProConcrete. The direct link between LEAP Bridge and the ISM is under development.

Fig. 8 shows the bridge model in LEAP Bridge, the bridge model in the ISM viewer and the bridge model in ProConcrete, respectively. Fig. 9 shows an enlarged image of a part of a detailed model in ProConcrete.

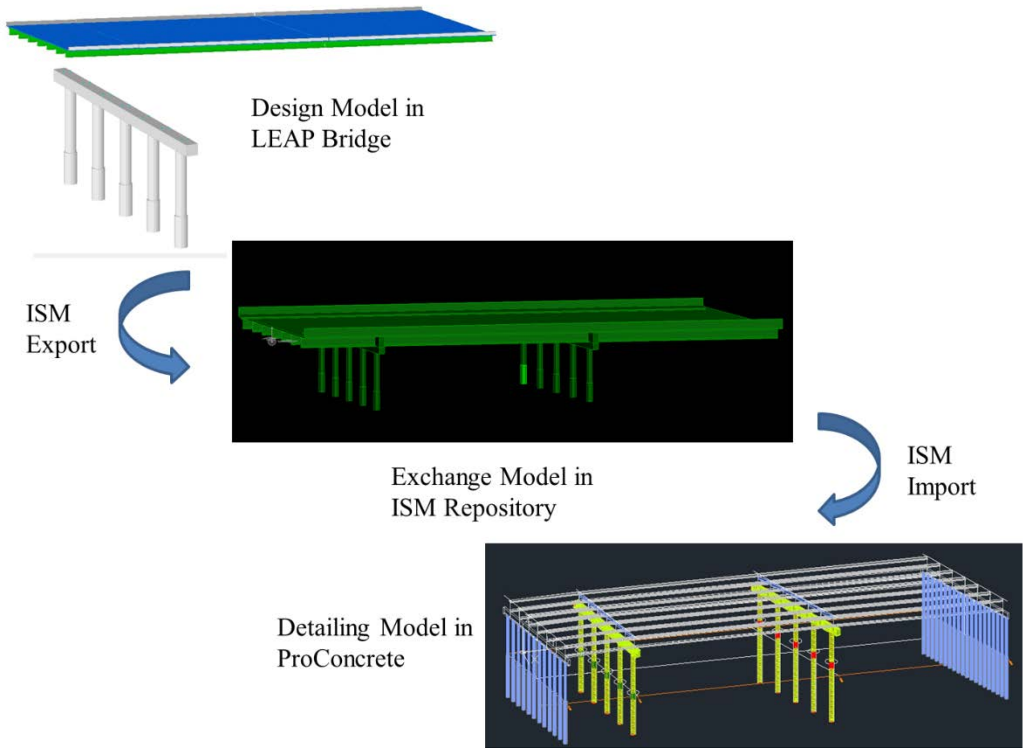

Fig. 7 A Case Study for the Use Case - Bridge Design to Bridge Detailing

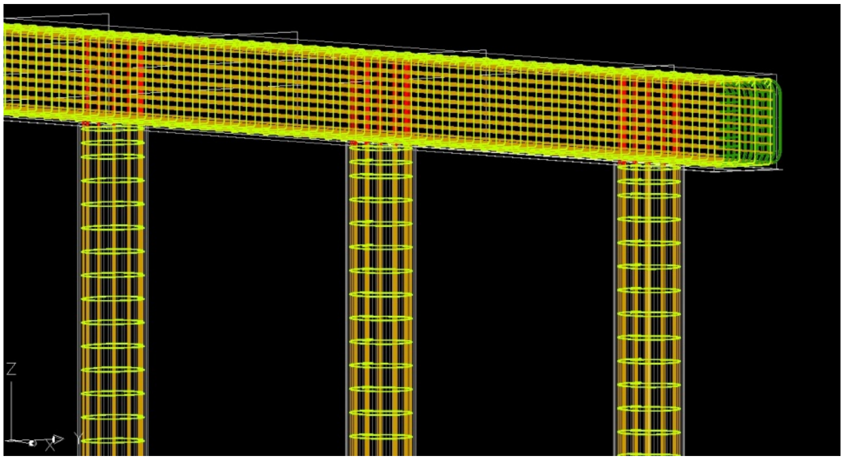

Fig. 8 Detailed model in ProConcrete

A current drawback of this approach is that at the time of writing, prestressing information is not yet within the scope of what can be transferred. This is because the current ISM schema does not support it. However, prestressing data is expected to be involved in the next version of the ISM schema.

DATA EXCHANGE USING IFC (APPROACH 2b)

IFC handles the data exchange for precast structures between different software applications through a file type called "ifc." For a data exchange between a designer and detailer, using exchange model A_EM.7, a data exchange model used in IDM, the designed structure can be exported from a design and analysis software application such as SAP2000 using file type "ifc" to a detailing software application like Tekla Structures. Such an exercise was carried out, as shown schematically in Fig. 6, to explore and illustrate the data exchange capabilities between design and detailing software using IFC file type and exchange mechanisms. The model can further be detailed in Tekla Structures and thus successfully avoids multiple re-entry of data that can be transferred using "ifc" file type. Like ISM, round-tripping in general is supported.

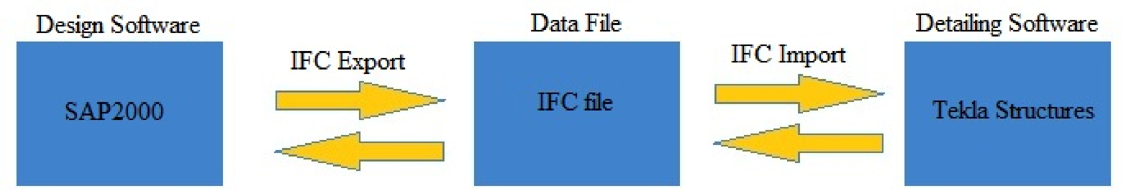

Fig. 9 Data exchange using IFC file

SIMILARITIES AND DIFFERENCES BETWEEN ISM AND IFC

Table 2 summaries some of the differences between ISM and IFC. ISM shares some common features with IFC. For example, they are both building-oriented and extensible. However, they are different in terms of the following characteristics.

1) The ISM focuses on sharing structural information, while the IFC concerns a broader scope about building information in all aspects. Because ISM was created for sharing structural engineering information, building information such as energy analysis data and cost estimation data, although included in the ISM client applications, cannot be exchanged by using the ISM[1].
2) The ISM schema was created in eXtensible Markup Language (XML), while the IFC was originally developed in the ISO-STEP EXPRESS language, and then extended to XML. Data modeling and processing with XML recently has been adopted by AEC/FM industry as core technology for data transfer[19] and is also of keen interest in the transportation industry, wherein bridges reside.
3) The ISM stores a combined analytical and physical model, while the IFC keeps distinct but parallel analytical and physical models. Because the ISM is mostly for

exchange of the structural data, it is easier to use one model to represent analytical and physical data. This eliminates the problem of synchronization of two parallel models[20].

4) The ISM merely allows users to access its API, while the IFC makes its whole schema accessible to users.
5) The ISM has an official viewer created and managed by its develolper, Bentley Systems, while the IFC has several viewers developed by multiple organizations other than its inventor, the buildingSMART alliance.

Table 2 Comparison of IFC and ISM (Bommer, 2012)

IFC	ISM
True BIM	Simplified BIM
All exchanges	Structural only
For entire project team	Design team
Parallel physical/analytical models	Combined physical/analytical models
Open standard	Open API

ONTOLOGY

Although the IFC and the ISM have been implemented and deployed in a large amount of AEC/FM software products and have enabled successful data exchange, it still requires manual coordination to make interoperability efficient[3]. This is caused by the nature of the schemas. The IFC was developed as a vast collection of tools for users to build a model in multiple views. Therefore, it acts as a weakly typed system. The ISM, which largely abbreviates the ambiguous manner of model representation, still cannot reflect semantic interrelationships of data model very well. To improve the interoperability of BrIM tools, Semantic Exchange Modules (SEM) needs to be defined using engineering ontologies. An SEM is a modular subset of a data schema which encloses relationships required in BrIM exchange model definitions. Using it can ensure the consistency and correctness of the data transfer, so as to reduce the manual work involved in interoperability[20,21,22,23,24].

GAP ANALYSIS OF ISM AND IFC2X4 FOR BRIDGES

Although IFC contains an extensive library of data fields used in the building industry, it still lacks important data fields used in the bridge industry. Thankfully, IFC being an extensible data schema provides a window to incorporate bridge data fields. In order to include bridge data fields in IFC, a gap analysis of the latest version of IFC2x4 has been carried out to identify missing data fields required in the bridge industry. To facilitate this analysis, a concrete bridge analysis and design software application and the well-developed bridge analysis schema for Opis/Virtis is also brought into this discussion.

The following observations have been made while conducting the gap analysis of IFC2x4 Release Candidate 3. It should be noted that ongoing efforts in Open-INFRA (previously IFC-BRIDGE) are attempting to address such gaps within an IFC development framework.

SIMILAR ENTITIES (SYNONYMS)

While conducting the gap analysis of IFC2x4 Release Candidate 3, it has been observed that the schema contains entities that have similar function in both buildings and bridges, but they are known by different names in these industries.

Table 3 Similar Entities

Mapping Entities			
LEAP Bridge (Bridges)	AASHTOWare Opis/Virtis (Bridges)	ISM	IFC2x4 RC3
Deck	abw_deck_panel, abw_deck_stl_panel, abw_deck_timber_panel	IsmSurfaceMember	IfcSlab
Girder	abw_lib_ps_box_beam, abw_lib_ps_ubeam, abw_lib_ps_tee_beam, abw_lib_ps_ibeam, abw_lib_stl_ishape, abw_lib_stl_angle, abw_lib_stl_channel, abw_lib_stl_struct_tee	IsmCurveMember	IfcBeamTypeEnum, IfcShapeProfileDef

In Table 3, a comparison of IFC2x4 RC3 has been done with two data schemas available in the bridge industry. It is illustrated in Table 2 that instead of “Slab” as in buildings, the term “Deck” is used in the bridge industry. Keeping in mind the similar function of the terms in both the industries, it would appear that this type of already existing entities in the IFC2x4 RC 3 can be used for bridges as well.

ENTITIES WITH DIFFERENT MEANING BUT SAME NAME (HOMONYMS)

Gap analysis of IFC2x4 RC 3 reveals that some of the entities related to building and bridge industry might have same name but these entities might represent something entirely different in both fields. This has been illustrated briefly in Table 4.

Table 4 Homonyms

Mapping Entities			
LEAP Bridge (Bridges)	AASHTOWare Opis/Virtis (Bridges)	ISM	IFC2x4 RC3

Diaphragm	abw_ps_box_int_diaph_loc, abw_ps_ubeam_int_diaph_loc, abw_mbr_alt_diaph_loc	IsmCurveMember, IsmSurfaceMember	NA

Table 4 shows an example of the same word being used in both the building and bridge industry but with different meaning. In this example, diaphragm may refer to the floor or roof system. On the other hand, in the bridge industry a diaphragm is a member that spans between the girders and assists in the distribution of loads. To avoid confusion, such types of entities will have to be clearly distinguished on the basis of their functionality and use in their respective industries, presumably utilizing principled ontologies.

MISSING ENTITIES

Gap analysis of IFC2x4 RC3 shows that there are numerous entities specifically related to bridges which are not included in the schema. Such entities are given in Table 5.

Table 5 Missing Structural Items

Mapping Entities			
LEAP Bridge (Bridges)	AASHTOWare Opis/Virtis (Bridges)	ISM	IFC2x4 RC3
Project Information	abw_bridge, bridge	NA	NA
Alignment	abw_bridge_ref_line, abw_bridge_ref_line_horz, abw_bridge_ref_line_vert, abw_struct_def_ref_line, abw_struct_def_ref_line_horz, abw_struct_def_ref_line_vert	NA	NA
Bridge Superstructure	abw_super_struct, abw_super_struct_alt, abw_super_struct_def	NA	NA
Curb	abw_conc_curb_sidewalk	IsmCurveMember	NA
Haunch	abw_haunch_range, abw_bmdef_haunch_range	NA	NA
Diaphragm	abw_ps_box_int_diaph_loc, abw_ps_ubeam_int_diaph_loc, abw_mbr_alt_diaph_loc	IsmCurveMember, IsmSurfaceMember	NA
Continuity Diaphragm		NA	NA
Box Beam Interior Diaphragm	abw_ps_box_int_diaph_loc	NA	NA

U Beam Interior Diaphragm	abw_ps_ubeam_int_diaph_loc	NA	NA
LRFD Live Load Distribution Factors		NA	NA

Quantity extraction is a straightforward byproduct of implementing BIM/BrIM. However, in addition to items listed in Table 5, some nuances of structural modeling/analysis (e.g., composite action between girder and slab) and of General Contractor interest (e.g., bid item costs, percent complete in support of monthly invoicing, etc) will also be of interest to include in schemas to recommend to the bridge industry.

SUMMARY, CONCLUSIONS AND RECOMMENDATIONS

This paper has initiated examination of options and associated issues arising in grappling with the need to develop standardized methodologies for supporting electronic data exchange in support of various activities involved throughout the concrete bridge lifecycle. Our efforts include the following:

- Process mapping for precast concrete bridges,
- Gap analysis of bridge data schemas, and
- Identification of items in exchange models (EM's) by obtaining opinions from industry personnel, and
- Hands-on investigation of existing means of implementing data exchanges

From a slightly more generalized viewpoint:

- Data schemata and associated interoperability functionality for the buildings industry is available and being used increasingly, and
- Since hard-coded approaches are too brittle (insufficiently general) and do not scale well, there is a need for well-defined data schema for bridges that is sufficiently comprehensive and robust to support bridge lifecycle operations, and
- Illustrated example(s) shows the need, advantages and possibility to create similar data schema(s) for bridges, and
- A fresh start may not be ideal. Existing data schemas for buildings may be used as a starting point (as illustrated in the IFC discussion above).

Eventual widespread deployment of standards-based electronic information exchanges (in support of interoperability and thus accelerated processes) in the concrete bridge industry will likely require both top-down and bottom-up initiatives including at least the following:

- Achieving widespread bridge industry consensus that bridge lifecycle data exchange standards are needed, and
- Software translators to and from those standards need to be written (etc) by commercial software solution providers, and

- Appropriate adjustments need to be made to QA/QC procedures and regulations, and
- Owners will need to specify BIM/BrIM data (file) format deliverables compliant with such standards, and
- Industry-appropriate means of institutional support mechanisms including but not necessarily limited to updating, standards-issuance, software compliance certification, etc.

ACKNOWLEDGEMENTS

Support and assistance of the following are gratefully acknowledged: NWFP University of Engineering & Technology, Federal Highway Administration, Bentley Systems, and the University at Buffalo.

VERSIONS OF THE SOFTWARE USED

SAP2000 Advanced 14.2.2
Tekla Structures 14.0 (Build 402844)
Microsoft Visual C# 2010 Express
Solibri Model Viewer Version 7.1
ProStructures V8i (SELECT series 4) – Version 8.11.4.42
MathCAD 14.0
ISM Structural Synchronizer V2
AASHTOWare Opis/Virtis V6.3

REFERENCES

[1] Eastman, C., Teicholz, P., Sacks, R., and Liston, K., BIM Handbook: A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers, and Contractors, 2nd ed., 2011, pp. 99

[2] Dawood, N., and Iqbal, N., "Building Information Modelling: Scope for Innovation in the AEC Industry," 10th International Conference on Construction Applications of Virtual Reality, 2010

[3] Palzar, T., and Turk, Z., "interoperability in practice: geometric data exchange using the IFC standard," Electronic Journal of Information Technology in Construction, V. 13, 2008, pp 315-33

[4] Steel, J., Drogemuller, R., and Toth, B., "Model Interoperability in Building Information Modelling," Software and Systems Modeling, V.11, No.1, 2012

[5] International Standards Organisation (ISO): Industrial automation systems and integration - product data representation and exchange - part 1: Overview and fundamental principles. ISO Standard 10303-1:1994, 1994

[6] International Standards Organisation (ISO): Industrial automation systems and integration - product data representation and exchange - part 11: Description methods: The express language reference manual. ISO Standard 10303 - 11:2004, 2004

[7] International Standards Organisation (ISO): Industrial automation systems and integration - product data representation and exchange - part 21: Implementation methods: Clear text encoding of the exchange structure. ISO Standard 10303-21:2002, 2002

[8] International Standards Organisation (ISO): Industrial automation systems and integration - product data representation and exchange - part 28: Implementation methods: Xml representations of express schemas and data, using xml schemas. ISO Standard 10303-8:2007, 2007

[9] International Standards Organisation (ISO): Industrial automation systems and integration - product data representation and exchange - part 22: Implementation methods: Standard data access interface. ISO Standard 10303-22:1998, 1998

[10] Eastman, C., Panushev, I., Sacks, R., Venugopal, M., Aram, S., See, R., and Yagmur, E., "A Guide for Development and Preparation of a National BIM Exchange Standard," V. 2.0, 2011

[11] Bazjanac, V.,"Early Lessons from Deployment of IFC Compatible Software" Proceedings of the 4th ECPPM, Z. & Scherer R.J, (ed.); Balkema A.A., 2002

[12] Aram, S., Eastman, C., Sacks, R., Panushev, I., Vanugopal, M., "Introducing a New Methodology to Develop the Information Delivery Manual for AEC Projects," *Proceedings of the CIB W78 2010:27th*, 2010

[13] Ziering, E., Harrison, F., Scarponcini, P., "XML Schemas for Exchange of Transportation Data," NCHRP Report 576, *Transportation Research Board*, 2007

[14] AIA E202-2008 Building Information Modeling Protocol Exhibit, 2008

[15] ConsensusDOCS 301 Building Information Modeling Addendum, 2008

[16] BuildingSMARTalliance, "*National Building Information Modeling Standard,*" Version 1- Part 1 : Overview, Principles, and Methodologies, 2007

[17] SearchWinDevelopment, http://searchwindevelopment.techtarget.com/definition/C

[18] Bentley System, Inc., "Integrated Structural Modeling - Programmer's Introduction to ISM: 161," 2011

[19] Beetz, J., van Leeuwen, J. P., de Vries, B., "An Ontology Web Language Notation of the Industry Foundation Classes," "*The 22nd CIB W78 Conference on Information Technology in Construction*," Dresden, Germany: 6., 2005

[20] Bommer, A., "Zen of ISM," 2012

[21] Katranuschkov, P., Gehre, A., Scherer, R., J., "An ontology framework to access IFC model data." *Journal of Information Technology in Construction 8(eWork and eBusiness):* 413-437., 2003

[22] Venugopal, M., "Formal Specification of Industry Foundation Class Concepts Using Engineering Ontologies," *Civil and Environmental Engineering, Georgia Institute of Technology.* Doctor of Philosophy: 261, 2011

[23] Venugopal, M., Eastman, C., Sacks, R., Panushev, I., Aram, V., "Engineering semantics of model views for building information model exchanges using IFC," *CIB W-78 Conference on Applications of IT to Construction*, Cairo, Egypt. 2010

[24] Venugopal, M., Eastman, C., Sacks, R., Teizer, J., "Semantics of model views for information exchanges using the industry foundation class schema." *Advanced Engineering Informatics* 26(2): 411-428, 2012

ACRONYMS

AASHTO	American Association of State Highway Transportation Officials
AEC	Architecture, Engineering and Construction
API	Application Programming Interface
BPMN	Business Process Modeling Notation
CAD	Computer-Aided Design
CIS/2	CIMsteel Integration Standard Version 2
COBie	Construction Operations Building Information Exchange
EM	Exchange Model
FM	Facilities Management
IAI	International Alliance for Interoperability
IDM	Information Delivery Manual
IFC	Industry Foundation Classes
ISM	Integrated Structural Modeling
MVD	Model View Definition
NBI	National Bridge Inventory
NCHRP	National Cooperative Highway Research Program
PDM	Project Delivery Manual
SEM	Semantic Exchange Module
SQL	Structured Query Language
UML	Unified Modeling Language
XML	eXtensible Markup Language
XSD	XML Schema Documentation

GLOSSARY

Enumeration — An enumeration of a collection of items is a complete, ordered listing of all of the items in that collection.

Ontology	In computer science and information science, ontology formally represents knowledge as a set of concepts within a domain and the relationships among those concepts.
Round-tripping	It is a functionality of software development tools that synchronizes two or more related software artifacts, such as, source code, models, configuration files, and other documents.
Semantic	In programming language theory, semantics is the field concerned with the rigorous mathematical study of the meaning of programming languages.

APPENDIX A

// Representative C# code developed to model a beam using the parameters obtained from MathCAD design worksheet

```
//names of the reference system files
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

// Tekla structures specific namespace abbreviation
using Tekla.Structures;
using Tekla.Structures.Model;
using TSM = Tekla.Structures.Model;
using T3D = Tekla.Structures.Geometry3d;

// other user added namespace abbreviations
using System.Collections;
using System.IO;
namespace beam_from_txt_file
{
   public partial class Form1 : Form
   {
      public Form1()
      {
         InitializeComponent();
      }
      private void buttonexit_Click(object sender, EventArgs e)
      {
```

```
            Application.Exit();
        }
        private void buttongettext_Click_1(object sender, EventArgs e)
        {

// retrieving TXT file from hard disk, reading the data from the file and assigning each data
to a text box on the program for visual inspection of the transferred data
            StreamReader objstream = new StreamReader("C:\\MathCAD\\Girder.txt");
            textBox1.Text = objstream.ReadLine();
            textBox2.Text = objstream.ReadLine();
            textBoxsection.Text = objstream.ReadLine();
            textBoxstrength.Text = objstream.ReadLine();

           //Converts the strings of Textboxes to double
            double startx;
            startx = double.Parse(textboxstartx.Text);
            double endx;
            endx = double.Parse(textboxendx.Text);

            // Connects to the open model
            TSM.Model MyModel = new Model();

            // Creates Beam
            TSM.Beam MyBeam = new Beam();
            MyBeam.Name = "Beam";
            MyBeam.Profile.ProfileString = textBoxsection.Text;
            MyBeam.Material.MaterialString = textBoxstrength.Text;
            MyBeam.Class = textBoxclass.Text;
            MyBeam.StartPoint = new T3D.Point(startx, starty, startz);
            MyBeam.EndPoint = new T3D.Point(endx, endy, endz);
            MyBeam.Position.Depth = Position.DepthEnum.BEHIND;
            MyBeam.Position.Plane = Position.PlaneEnum.MIDDLE;
            MyBeam.Position.Rotation = Position.RotationEnum.TOP;
            MyBeam.Insert();
            MyModel.CommitChanges();
        }
    }
}
```