Interoperability Definition Framework for Creating Consistent Constraints in BIM Data Exchange

Liu H.[a,b,*], Gao G.[b,c,*], Gu M.[b,c]
[a] Digital Horizon Technology Co., Ltd., China. [b] Tsinghua University, China. [c] Beijing National Research Center for Information Science and Technology (BNRist), China.
liuhan@digital-h.cn; gaoge@tsinghua.edu.cn

Abstract. To improve the interoperability in the data exchange of Building Information Models (BIMs), various standards and methods have been proposed to add constraints to the data exchange process, including da

2.2 Methods for Integrating the Knowledge about BIM Data Exchange

The creation of BIM data exchange rules requires the integration of knowledge about both the engineering domain and information technology. The communication between the domain experts and data engineers may take lots of effort, and there are also tedious tasks for viewing the documents and converting the constraints into computer language. Related studies tried to set up the knowledge model for BIM data exchange specifications, and to support the computer-aided process of creating the runnable rulesets.

The semantic web technology is a promising approach to formally represent the classification of domain concepts and the definition of data schema in order that the knowledge can be shared and reused (Venugopal, et al., 2012; 2015; Lee, et al., 2016). The classification of domain entities and definition of properties are represented as an ontology. The IFC schema and the mappings from domain terminologies to data objects are also represented as ontologies. By reusing the knowledge about domain concepts and mappings, MVD rulesets and documents can be automatically generated according to some formalized representations of exchange requirements. The consistency inside of the ruleset can also be automatically checked to avoid ambiguity and unsatisfied necessary components. In another related study (Son, et al., 2022), the buildingSMART Data Dictionary (bSDD) is used as the definition of domain concepts, and the idmXSD is used as the formal representation of exchange requirements. The information units can be mapped to the IFC data entities with the help of bSDD. When the MVD ruleset is generated, the validity of the ruleset is then automatically checked to ensure that the information units conform to the IFC schema and that the exchange requirements are fully covered.

The above approaches focus on the binding of domain concepts on an open-source data schema, typically the IFC. However, the community starts to realize that the constraints need to be applied to the whole process of data creation, conversion and reception to ensure the interoperability of data exchange, and more software-specific instructions and configurations are needed to support the easy creation of data in a non-IFC modeling tool. Several commercial tools have taken steps in this direction. BIMQ (AEC3, 2016) is an online collaborative tool for specifying the exchange requirements and editing the MVD rules. BIMQ provides a graphical user interface for defining domain concepts, exchange requirements, IFC mappings and constraint rules. Based on the defined MVD rules, BIMQ can also generate configuration files for specific tools like Autodesk Revit and GraphiSoft ArchiCAD, so that the property fields and IFC mappings can be loaded by the tools. The IDS toolkit (ACCA, 2021) initiates the IDS format, and provides user interfaces to define the exchange requirements and constraints. The IDS ruleset is designed to be loaded on an upstream BIM authoring tool for providing human-readable instructions and general constraints on property values, so that the exported IFC data tends to satisfy the exchange requirements.

In comparison, BIMQ tends to make use of the existing configuration files of each software tool, but IDS tends to collect all needed settings and instructions into one standard IDS ruleset, and asks the software vendors to support loading and utilizing the rules. Anyway, compared with the reviewed methods based on the open-source IFC schema, the validation of data compliance and the configuration of data mappings on software-specific formats are still challenging. There is still a need for a methodology to implement software capability confirmation and data validation on various software-specific formats.

3. The Interoperability Definition Framework Knowledge Graph

In this paper, the Interoperability Definition Framework (IDF) method is proposed for integrating the knowledge about data exchange from multiple stakeholders. During the creation

of the IDF knowledge graph, the input knowledge is automatically checked to find the conflicts. The MVD and IDS rulesets can then be generated to provide consistent constraints for the creation, conversion and validation process of BIM data. The knowledge graph can also support the flexible confirmation of software capability for user-defined exchange requirements.

The idea of the IDF method comes from the observation that although the data formats are close-sourced, the software development kits (SDKs) are provided to access the data content. The SDKs are usually in object-

(5) Converter model. The converter model is established between the source schema and target schema, defining the capability of the converter. In the converter model, the ability of the converter to transform a source data item (a class or a property) to a target data item is defined as an ñidf:Conversionò node. One source data item may be allowed to convert into several optional targets, and each conversion is ñenabledò by one or more ñidf:ConverterOptionò nodes. Some of the options are exclusive (such as a group of radio buttons in the converter UI), and a valid configuration of the converter can be defined as a set of non-exclusive option nodes. An example converter model is shown in Figure 5.

(6) Transmitted mapping. The transmitted mapping is an automatically generated mapping model between the source schema and the target schema. Each entry in the transmitted mapping

in the process to bu

requirements and cons

converters may also support instance-level overrides of data type mappings. More detailed instructions are needed in the IDS rulesets for such instance-level configurations. Third, the