ONTOLOGY BASED PARTIAL BUILDING INFORMATION MODEL EXTRACTION

Le Zhang, Ph.D. Candidate, zhangle@ufl.edu
Raja R. A. Issa, Professor, raymond-issa@ufl.edu
University of Florida, Gainesville, USA

ABSTRACT

The research and application of building information modeling (BIM) has been focused on the entire project and the complete life cycle. However, the daily routine on a construction job site has specific requirements and bears certain limitations regarding the usage of information stored in a BIM model. The limitations include scarcity of computing power and trained personnel. One of the requirements is to view a partial model instead of the original, complete model. The partial model may be defined by certain location parameters such as storey numbers and/or building grid lines.

In this paper we discuss an ontology-based method to extract a partial model from a complete BIM model. The partial model, as well as the complete model, should be defined in industry foundation classes (IFC) format, which is the widely supported open standard data exchange format for BIM. Theextraction is based on an IFC-based ontology which defines the necessary building blocks of a valid IFC model and the rules of extraction. The whole process is to be implemented as a Web service allowing remote accessibility from various computing platforms. The Web service system could also be linked to other construction software applications for automating construction management functions.

Keywords: BIM, IFC, Ontology, Web services, Partial model.

1. INTRODUCTION

The construction industry is information-intensive (Brandon 1995, Wang 2010). With the development of Building Information Modeling (BIM), the media for the information storage and exchange in a construction projects is evolving from traditional paper-based documents into electronic models. The transition brings great benefit to the industry, including visual advantages and direct use of the model in a variety of computer-based analyses. However, associated with this trend there are also some problems.

The key concept of BIM is a complete object-oriented digital model of the building. Besides the geometry information, other building information that is not available in traditional CAD solutions is also stored in the model, including but not limited to building materials and costs, project specification and contract information, building components manufacturer, price and warranty, etc. The stored information could be retrieved and reused easily once they are entered, thus eliminating the need to re-input or even re-collect information that is valuable in later stages of the building life cycle. Moreover, the stored information can readily be used for various analysis of the building, such as structural stability, energy consumption or building code compliance (Eastman et al. 2008).

However, ambitious and promising as it is, the majority of current BIM applications are used during the building design and pre-construction stages. BIM application on construction jobsites, especially on small projects, is rare. One of the reasons is that BIM models are usually computing intensive, while the required computing power and trained personnel may not be available on construction jobsites. In addition, most of the daily work on a construction site is being accomplished by specialty subcontractors, who only deal with small portions of the project and do not need to access the whole model.

There are also other situations that more intelligent and information-rich BIM models are purposefully evaded. For example, in many construction education simulation projects, the authors resort to gaming engines to generate a 3D environment instead of using dedicated or even existing BIM models. One of the reasons is that the simulation may only need the exterior dimension and texture information of a building or a room, while the extra information stored in a BIM model makes it cumbersome and hard to manipulate.

Currently most of the research and application of BIM have been focused on adding information to the model so that it could be used on the entire project and through its complete life cycle. Actual application scenarios of BIM models, such as daily construction routine on a construction jobsite or construction education simulation, etc. may have specific requirements and bears certain limitations regarding the usage of information stored in a BIM model. Hence, how to filter redundant information for specific scenarios and extract certain information from a complete BIM model is another important issue to make full use of the BIM models. This study proposes an ontology built with Web Ontology Language (OWL) based on Industry Foundation Classes (IFC) specifications to help the information retrieval process from an IFC model. The development of the ontology and the partial model extraction based on the ontology is discussed.

2. LITERATURE REVIEW

2.1 The Industry Foundation Classes

As “the only comprehensive international standard” for BIM interoperability, IFC is a set of definitions describing the consistent data representation of building components (Liebich 2010). Developed and maintained by buildingSMART (formerly known as International Alliance for Interoperability (IAI)), it is designed to be able to store and exchange building information over the whole building lifecycle. The IFC object specification includes not only the geometric information but also physical properties and relationships, and endows the IFC objects with intelligence.

As an instance of the International Standard Organization (ISO) 10303 international standard, one of the advantages of IFC is that IFC specifications are open source and publicly accessible, so everyone has full access to the information it contains. Therefore it is ideal for transferring data between different software platforms. The native IFC format is based on plain text, and will become quite large if used to store all the building information in one file. IFC also supports XML format storage. It allows any IFC model to be described in XML format under the ifcXML schema. The native language of IFC is EXPRESS. EXPRESS-G is a more human readable graphical expression of the EXPRESS schema.

As for the time of writing, the newest is version 2x4 RC 2, available since September 2010. The new version features a new documentation format under ISO documentation requirements, besides other improvements. This new version will be the basis for a new full ISO standard numbered ISO 16739 (Liebich 2010, buildingSmart 2010). This new version of IFC, however, does not yet have an XML schema available.

The open standard nature of IFC makes the information available to everybody, but the IFC specification itself is too complicated for direct use for even experienced developers (Liebich 2010). Currently the most widely used method to extract information from an IFC model is to use third party software such as IFCEngine.dll, which works on the native .ifc file (TNO 2010, Beetz et al. 2010).

2.2 Current Research on Partial IFC Model Extraction

The requirement and advantages for partial model extraction has been identified by several researchers. The primary purpose of generating a partial model is to reduce the size and complexity of an IFC model, either to fit into the domain application requirement or the data transmission requirement (Beetz et al. 2009, Weise et al. 2003). Two kinds of partial model could be identified based on different scenarios. The first one is to extract information on a certain view or aspect of the building out of the complete model, e.g. a model with only the geometries,or a model with only the information related to green building rating. The Model View Definition (MVD) initiative of buildingSMART is an exemple of work on such extraction (buildingSMART 2011). Beetz et al. (2009) proposed a graph query method on the IFC ontology that compartmentalizes the ontology, which could filter out the geometry and topology information, leaving the rest for logic-based reasoning.

The second kind of partial model is a subset of all the entities in the original model. The Generalized Model Subset Definition (GMSD) schema proposed by Weise et al. (2003) is a good example of such extraction. The work is based on EXPRESS, the native modeling language of IFC. Although the compatibility with IFC is ensured, EXPRESS enjoys a far less research population and readily available tools. Along with the partial model extraction, merging a potentially updated partial model back into the original model is also addressed.

Different as they are, one of the major similarities of the above mentioned methodologies are that all the instances of the original model are kept unchanged, which may be a result of the avoidance of imperative programming involvement. However, in real application scenarios, an entity might need to be modified to suit the partial model extraction process. One such example is described in the extraction algorithms section of this paper.

2.3 Ontology and Web Ontology Language

The term ontology originates in Philosophy and is used to designate "the branch of metaphysics that deals with the nature of being". There are several widely cited definitions when it is adopted in computer science and artificial intelligence. The one preferred is defined by Studer (1998) as "formal, explicit specifications of shared conceptualizations," which is also adopted by Guarino (2009). Ontology modeling is the process to model concepts and relationships in a specific field into formal ontologies, i.e. in an ontology language with formal syntax.

In 1999 the World Wide Web Consortium (W3C) published the Resource Description Framework (RDF), a recommended specification for a system on how to locate and describe information. Using RDF, each individual tag is linked to a Universal Resource Identifier (URI), which contains the definition of a concept mentioned in the tag. Moreover, RDF uses triplets (subject + verb + object) to link the individual XML metadata tags to form rules that could be used in the reasoning of computer programs. RDF is the basis of the W3C ontology recommendation for Semantic web, Web Ontology Language (OWL), the ontology modeling language we choose in this research.

The current version of OWL is OWL 2, which was completed in 2009. An OWL 2 ontology could be subdivided into two parts: syntax and semantics. The semantics is the meaning of the ontologies. There are two alternative ways to assign meanings to ontologies: OWL 2 DL and OWL 2 Full. The same semantics could be expressed in different syntaxes. The RDF/XML style is the primary and mandatory one specified by W3C (W3C 2009).2.4 Current Ontology Research in Construction

Domain ontology defines concepts, activities, objects and the relationships among elements within a certain domain. As an information-intensive industry, the construction industry is seeing more domain ontology research. Several sources have been explored; pilot projects have been implemented; and different ontology building process models have been proposed.

The ontology research in the construction industry could be roughly divided into three stages. The first one is before the year 2000. During this time use of the term "ontology" was still very rare. Most related research in the construction industry focused on artificial intelligence (AI). For example, Chinowsky (1995) tried to judge whether a design satisfied certain non-numerical specifications based on design CAD files.

After the year 2000 comes the second stage, in which ontology became a popular research topic in construction industry. Construction industry knowledge management is among the first disciplines focusing on the building and application of industry-wide ontologies. Several projects in Europe have addressed this problem. The construction knowledge management platform e-COGNOS project emphasized ontology as "a basis for knowledge indexing and retrieval (Wetherill et al. 2002)". This field is still under research currently. For example, Wang (2010) made use of ontology to represent and reason on context-sensitive construction information as an alternative way of construction information management. Ontology is also used in product modeling. Beetz (2006) used ontology in a topological reasoning service to deduce the area of a zone from CAD files (note that in a BIM model this information may already be available).

Another research field during this period that is closely related to ontology is the multi-agent systems. Ren and Anumba (2004) reviewed the basic agent and multi-agent system (MAS) concepts and application in construction research. Smart agents are defined as an autonomy program that is capable of co-operate with and learn from other agents and the environment. There is debate on the difference between an agent system and a Web service. Some scholars even claim they are essentially the same (Pathak 2006). The concept adopted in this study is that Web services is a special kind of software agents, which conforms to W3C Web services standards (W3C 2004).

The third stage is about ontology research on BIM. The ISTforCE project explored the development of an ontology to decode IFC models (Katranuschkov 2003). In the model checking framework proposed by Hjelseth and Nisbet (2010), ontology is treated as the building block of knowledge and the implementation basis of the meta-model. In 2006, the Open Geospatial Consortium (OGC) examined the feasibility of representing Geography Markup Language (GML) in Web Ontology Language (OWL) as part of the preliminary effort to extend existing services, encodings and architectures with Semantic Web technologies (Akinci et al. 2008). As a relatively new field, BIM ontology research needs more exploration. Jung (2010) noted that ontology research for BIM, especially on the hierarchy structure of BIM objects, was still rare. IfcOWL (Beetz et al, 2009) is by far the most complete effort to lift the IFC specification onto ontology level. However, since most of the ontology elements generated are strictly rooted in IFC specification, its flexibility in different application scenarios might be restricted.

Different possible sources have been explored as potential ontology sources. Besides more structured documents like specifications and OmniClass, less structured construction documents like OSHA safety recommendations have also been explored as a source of ontology (Wang 2010). Building codes also seem to be a promising alternative (Cheng et al. 2008). When several ontologies are available, it is necessary to choose one of them to use for a specific scenario. Ontology matching and mapping has become an interesting topic since it plays an important role in joining heterogeneous ontologies to work together (Paolucci et al. 2002; Cheng et al. 2008).

Ontology alone may not prove very useful in the real world. Many researchers have resorted to Semantic Web and Web services to exert the power of ontology. Issa and Mutis (2006) proposed a Semantic Web framework to address the reconciliation on different construction ontologies. Wetherill et al. (2002) suggested a knowledge management platform based on the Web services model, using construction domain ontology as “a basis for knowledge indexing and retrieval.” Shen, Hao and Xue (2010) proposed approach for facility lifecycle information integration and decision making is also an agent-based service-oriented system. build on The effort of Vacharasintopchai et al. (2007) to build a working semantic Web Services framework for computational mechanics is a good example of combining the Semantic Web and Web services together to work in the real world outside academic laboratories. Their framework was built on smart phone rather than normal desktop operating systems, which is very promising for mobile computing requirements such as those found on a construction job site.

3. IFC-BASED CONSTRUCTION INDUSTRY ONTOLOGY DEVELOPMENT

According to Corcho (2002), an ontology should include the following minimal set of components: classes or concepts (with attributes describing the class), and relations or associations between concepts. Attributes and binary relations should be distinguished. Attributes are represented by basic data types, such as a number or a string, while a relation is between two classes. For example, “the height of a window is 2 meters” describes the attribute of a window, while “the window is in a brick wall” describes the relation between the window and the wall.

Although the newest version of IFC specification is IFC2x4, the ontology source chosen for this research is the newest stable version of IFC specifications IFC2x3. This version is stable and widely used, and there is a corresponding XML schema publicly available. In addition, the core section of the IFC specification - the 3D building element breakdown, the spatial structure and the shape representation- is well established and minimally changed (Liebich 2010). The OWL 2 DL is the target ontology format, because it balances expressiveness and reasoning power.

3.1 Basic Ontology Development

The basic ontology is the part of ontology components that can be derived from the IFC specifications directly. The contents of the IFC specifications could fulfill most of the ontology components requirements, and forms the basics of the whole ontology.

The IfcWindow specification is used here as an example. IfcWindow is a typical entity in the IFC specifications. The specification page of IfcWindow includes the following sections: summary, material use definition, property set use definition, quantity use definition, containment use definition, geometry use definition, EXPRESS specification, EXPRESS-G specification, attribute definitions, formal propositions and inheritance graph (buildingSmart 2010). The sections contained in other IFC elements are different due to the nature of each element, but are all structurally similar. The information contained in these sections fits into the different components required for developing an ontology.

The classes or concepts requirement of an ontology is about the nature or definition of certain terminology. They are also known as entities (this is the name used in IFC) or sets. The "summary" section of IfcWindow gives formal definitions of a window from ISO as well as an explanation of other IFC entities used or related to IfcWindow. The Uniform Resource Locator (URL) of the webpage could be used as the URI to identify the terms in the ontology. Classes are usually organized in taxonomies with inheritance information. This information is available in the IFC "Inheritance Graph," which traces the inheritance relationship back to the abstract entity IfcRoot, which is the ancestor of all the independent IFC entities. The inheritance could be expressed by a subclass in the ontology. Table 1 indicates the inheritance relationship from IfcRoot to IfcWindow. The (abs) after the entity name indicates the entity is an abstract entity.

The attributes of the class could be found in the following sections of IFC specifications: property set use definition, quantity use definition, geometry use definition and attribute definitions. Property sets are most typical attributes information. A property set is a group of properties that applies to each entity. The IfcWindow entity has three property sets: WindowCommon, DoorWindowGlazingType and DoorWindowShadingType. The WindowCommon property set includes common attributes found in all windows, including acoustic rating, fire rating, etc. Each property is described in a word (string) or a number, which are referred to as IfcPropertySingleValue in IFC. The other two property sets are properties about the glazing and shading of the window, which also apply to doors. The reason all these properties are divided into three groups is to promote the re-use of each property set through different entities. Other sections are also sources of entity attributes, for example, the geometry use definition section includes the height and width of the window with each value represented as a number.

Table 1: Inheritance relation for IfcWindow.

IFC Entity	**IFC Schema**

IfcRoot (abs)
  IfcObjectDefinition (abs)
    IfcObject (abs)
      IfcProduct (abs)
	Core – Kernel

IfcElement (abs) IfcBuildingElement (abs)	Core – Product Extension
IfcWindow IfcWindowStandardCase	Shared – Shared Building Elements

The relations in an ontology are also called roles. They denote how the classes or entities are associated with others. Most of the relations are binary, with two classes involved. In IFC, most of the relations are defined as the subclasses of IfcRelationship, with prefix IfcRel. IfcRelationship is an abstract entity inherited from IfcRoot, on the same level of IfcObjectDefinition shown in the table above. The relation of the IfcWindow class with other classes is described in the "summary" and "containment use definition" sections. The relations a window may be involved in include "fill" and "void". A window "fills" an opening, which "voids" a wall. Corresponding IFC entities are IfcRelFillsElement and IfcRelVoidsElement. Together, the window, the opening and the wall are “contained” (IfcRelContainedInSpatialStructure) in a building story. The building itself is an “aggregation” (IfcRelAggregates) of several stories.

Figure 1 shows the UML format of the partial sample ontology about IfcWindow. It is a subclass of IfcBuildingElement and superclass of IfcWindowStandardCase. It has four properties or attributes, including simple ones like width and height expressed in simple numeric measurements and complex properties defined by other IFC elements. The same ontology is also expressed in RDF/XML format, as shown in Figure 2.

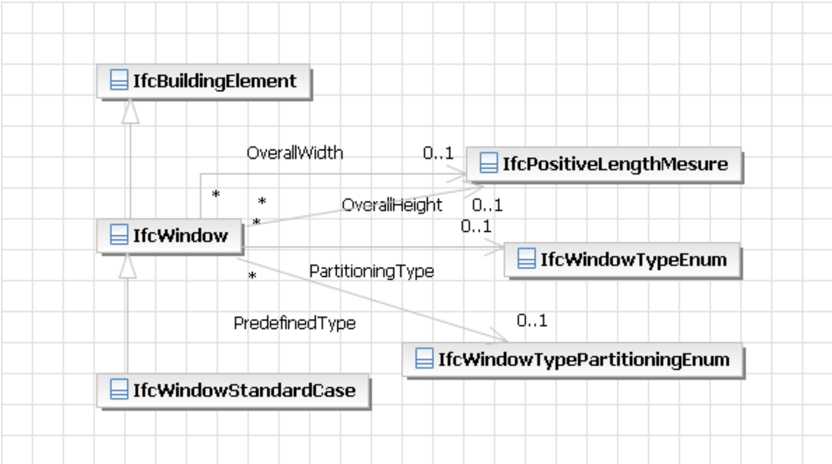

Figure 1: A sample IfcWindow ontology in UML.

```
<rdf:RDF
   xmlns="http://lezhang.net/IfcOntology">
   <owl:Ontology rdf:about="http://lezhang.net/IfcOntology"/>
   <owl:Class rdf:about="IfcWindow">
     <rdfs:subClassOf rdf:resource="IfcBuilidngElement"/>
   </owl:Class>
   <owl:ObjectProperty rdf:about="OverallHeight">
     <rdfs:domain rdf:resource="IfcWindow"/>
     <rdfs:range rdf:resource="IfcPositiveLengthMeasure"/>
   </owl:ObjectProperty>
...
```

Figure 2: A sample IfcWindow ontology in RDF/XML.

Currently the ontology is derived from the IFC specifications manually. From the above description it is not difficult to see that this process is lengthy and error-prone. Because IFC has an XML format schema available, and OWL is also written in XML format, an automatic transmission from ifcXML schema to OWL via XSLT is currently being explored.

3.2 Extended Ontology Development

Extended ontology is the ontology components that are not originally included in the IFC specifications but are added according to the requirement of the specific system or requirement.

Since the purpose of the specific ontology we are using is information retrieval from an IFC model, some of the required ontology components may be out of the scope of the original IFC specifications. One of the extensions are to facilitate the information retrieval when the inputted query includes terminologies that are not readily available in the current IFC specifications. For example, if the inputted query includes a term “girder”, which is not a standard IFC term, an extension is needed to translate this term into the corresponding standard IFC term “beam” or “IfcBeam”. This kind of extensions can be implemented by using an “equivalent” relationship between the new class and existing IFC classes, and will not affect the hierarchical structure of the ontology.

Another important relation needed in partial model extraction is the “containment” relation, which can be mapped to several IFC relationships, including the above mentioned IfcRelAggregates, IfcRelContainedInSpatialStructure, and others. For example, an IfcBuilding is an “aggregate” of several IfcBuildingStory, while an IfcBuildingStory “contains” several IfcWalls. Since the difference between these two kinds of relationship is trivial for the purpose of partial model retrieval, those two kinds of relationship could be expressed in one single “containment” relation in the ontology.

While the basic ontology remains stable with each release of the IFC specifications, the extended ontology could be more versatile and updated more frequently according to the specific requirements of the different systems that the ontology is being used for.

4. PARTIAL MODEL EXTRACTION FRAMEWORK USING THE ONTOLOGY

4.1 The Semantic Web Services Framework

The ontology developed for the purposes of this study is further implemented in a Semantic Web Services system to finish the partial model extraction. The system structure is shown in Figure 3. The IFC model may be stored remotely on the company’s datacenter while the extracted partial model will be transmitted to the construction site.

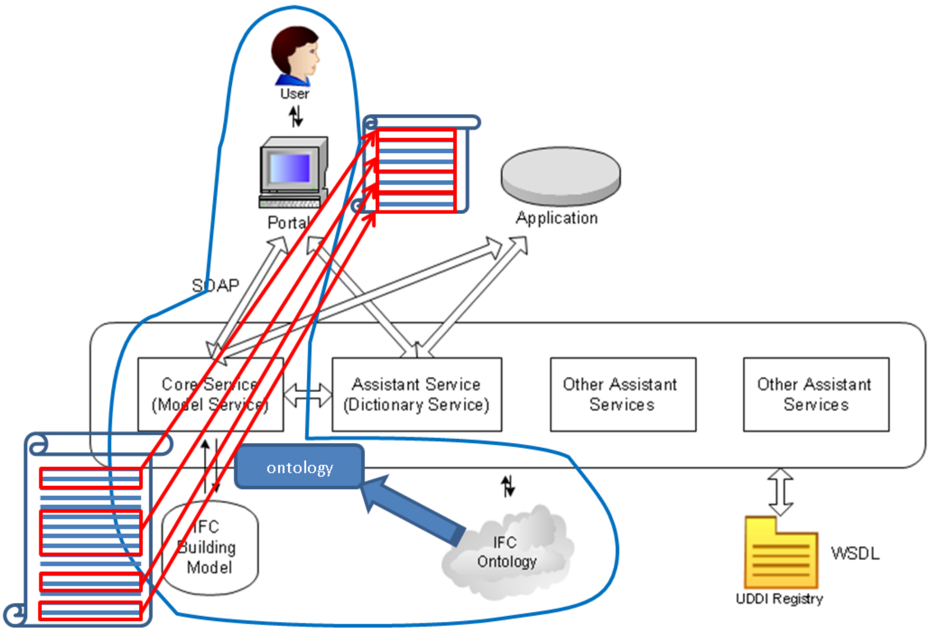

Figure 3: Partial model extraction framework using the ontology

Although an IFC/ifcXML file is text-based and could be easily opened in any text editor, simply copying and pasting a chunk of the file to create a new one is not enough to make it useful, because the elements in an IFC file may refer to many other elements and/or be referred to by other elements. For example, the following line is the representation of an IfcWindow element in an IFC file:

```
#281=IFCWINDOW('0_p6ZzFwjAovJ0NxnsEEW_',#42,'Fixed:36" x 48":36" x
48":157225',$,'36" x 48"',#280,#274,'157225',4.,2.999999999999999);
```

There are some explicitly described properties, including a string for its ID, another string for its type and two numbers for its width and height. This object is also referring to at least three other elements, namely #42, #280, #274, which might again refer to other elements. These referred elements may or may not include import information about this element. For example, #42 is IFCOWNERHISTORY, which is not really important, but #280 IFCLOCALPLACEMENT includes critical information on the exact location of the window. Besides, there are other syntax requirements (e.g. header information) for an IFC file to be valid.

Our proposed approach is to extract a part from an IFC file while keeping its integrity using the IFC-based ontology, which stores the information about possible relationships between each elements as well as necessary components for a valid IFC file as a whole. The detail is explained in the extraction algorithm part below.

In the test case shown in Figure 3, an extraction request is initiated by the user, with some kind of location information as input parameter. The location could be specified by an element, like a specific window. It could also be in the form of the grid lines if the grid system information is available in the model, in which case a comparison between IfcGrid and relevant building element is made to pin the desired location. The system analyzes the query input, finds the elements that satisfy the query using the extraction algorithm, and finally reassembles them into a new partial IFC model, which is transmitted back to the user.

4.2 The Extraction Algorithm

When the IFC file is read into the system, the IFC model is interpreted as a tree structure in the internal storage format, with the IfcProject element as the root, or level 1. All the elements with a containment relation specified in the ontology with level n elements are structured as level n+1 nodes, or the children of level n nodes in the element tree. For example, after the interpretation, the IfcBuilding "contains" (instead of "aggregates") several IfcBuildingStorey, which further "contains" other IFC building elements. The "basic" IFC elements that cannot hold or contain other elements are always placed as the leaf of the tree. For the elements on the same level, the "containment" relationship does not apply, but other relationship may still link them together. The interpreted tree structure is shown in Figure 4, with IfcBuilding being a "container" of all the elements that go into the building.

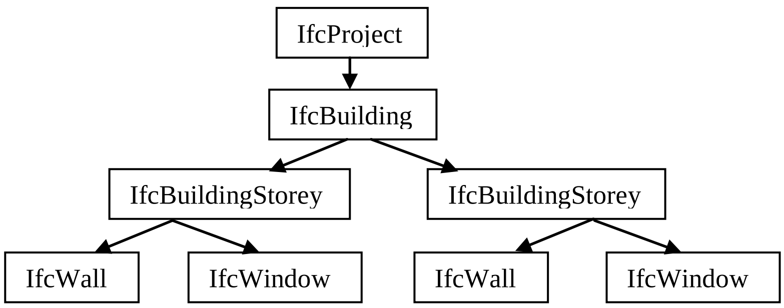

Figure 4: Tree structure of IFC elements

Based on the tree structure, a two-pass partial model extraction algorithm is developed. The algorithm is described with the example shown in Figure 5. As mentioned earlier, the partial model extraction starts with a specific element, or leaf node in the element tree. In the example in Figure 5, the window on the second floor has an unique object identification (OID) number that is passed as the input parameter. Another input parameter would be used to specify how large a partial model is needed, or the range of extraction, i.e. a whole floor or a room or simply a wall. The default value of the parameter would include all the objects that are immediately next to the specified element.

The first pass is going up the decision tree in Figure 4 starting from the element to locate a proper container that could hold all the elements required for the partial model. This pass usually ends up at one of the four subclasses of IfcSpatialStructureElement, namely: IfcSite, IfcBuilding, IfcBuildingStorey, or IfcSpace. According to the way the model is built, the end of the first pass for the window in the example could be different. For an architecture model a single wall is usually built from level 1 up to the roof, while in a construction model the same wall may be divided into several horizontal sections with one section for each floor. Without loss of generality, we assume that our first pass for the window actually ends up with the wall located on the first floor which is found in the first floor IfcBuildingStorey container.

After the first pass, the second pass is going down the tree from this container element to traverse all the potential elements. The elements that are connected with the starting element and other elements under the same container are checked. The location of each element is compared with the starting element. If the distance between the two elements is in a certain range, it will be selected for inclusion into the partial mode.

Next, all the selected elements are reassembled. At this point, some of the elements like the wall are customized according to the extraction range input, and their position is adjusted to a new reference level. This adjustment will indicate that the tailored wall is actually placed on the second floor. This will trigger another two-pass examination as described above and more elements may be selected. Finally, all the selected elements are reassembled into a new partial model.

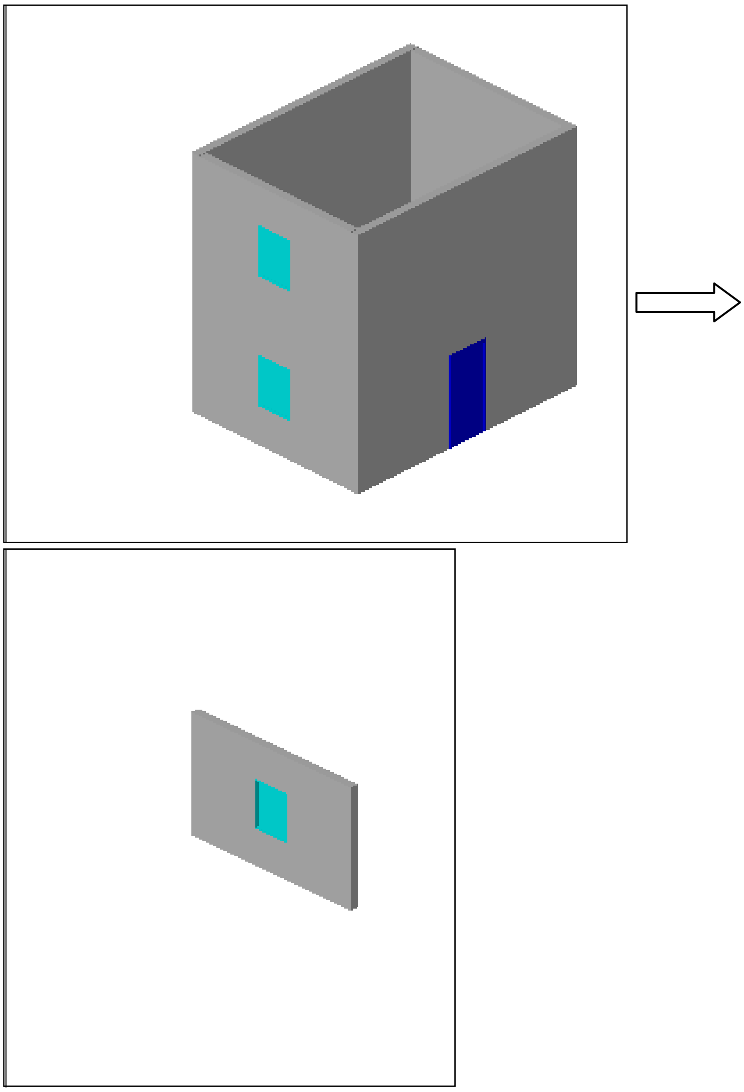

Figure 5: Extracted partial model

5. CONCLUSION AND FUTURE WORK

By combining the strength of ontology and IFC technologies, this study explores the possibility of extracting a partial model from an IFC model. A sample ontology is developed and used in a Web service for the extraction.

As an alternative way to utilize the information in a BIM model, the ontology-based partial model extraction approach described in this study is one of the many possibilities for utilizing a construction industry domain ontology. As mentioned in Katranuschkov (2003), a properly developed domain ontology could hide the model complexity, hence separating the more complicated process of knowledge building and the process of using the knowledge stored in an ontology.

Further research and exploration on both the ontology and the Web service framework is expected. The current sample ontology only covers a small fraction of IFC elements. A complete ontology would be far more complicated. The effectiveness of this approach is waiting further validation. A standard Web services interface will be implemented so that the service could be connected to by other applications, and possibly be used in a cloud computing framework. For example, every morning the

4D scheduling software could be configured to extract the partial model that the team is going to work on for that day from the complete model located at an off-site location, and it will have the partial model ready for the kickoff meeting before the crew arrives at work.

REFERENCES

Akinci, B., Karimi, H., Pradhan, A., Wu, C.-C., and Fichtl, G. (2008). "CAD and GIS interoperability through semantic web services." *Journal of Information Technology in Construction*, 13, 39.

Beetz, J., van Leeuwen, J., and de Vries, B. (2006). "Towards a Topological Reasoning Service for IFC-based Building Information Models in a Semantic Web Context." *Joint International Conference on Computing and Decision Making in Civil and Building Engineering*. Montreal, Canada.

Beetz, J., van Leeuwen, J., and de Vries, B. (2009) “IfcOWL: A case of transforming EXPRESS schemas into ontologies.” *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*. 23, 89 – 101.

Beetz, J., de Laat, R., van Berlo, L., and van den Helm, P. (2010). “Towards an Open Building Information Model Server.” DDSS 2010.

Brandon, P. S., and Betts, M. (1995). *Integrated construction information*, Taylor & Francis.

buildingSmart (2010). "IfcWindow." <http://www.iai-tech.org/ifc/IFC2x4/beta3/html/ifcsharedbldgelements/lexical/ifcwindow.htm>. (5/20, 2010).

buildingSmart (2010). "Industry Foundation Classes Release 2x4 (IFC2x4) Release Candidate 2 ", <http://www.iai-tech.org/ifc/IFC2x4/rc2/html/index.htm>. (3/15, 2010).

buildingSmart (2011). "Model View Definition Summary", < http://buildingsmart-tech.org/specifications/ifc-view-definition/summary>. (7/15, 2011).

Cheng, C. P., Lau, G. T., Pan, J., Law, K. H., and Jones, A. (2008). "Domain-specific ontology mapping by corpus-based semantic similarity." *2008 NSF CMMI Engineering Research and Innovation Conference*. Knoxville, Tennessee.

Chinowsky, P. S., and Reinschmidt, K. F. (1995). "Qualitative Geometric Reasoner for Integrated Design." *Journal of Computing in Civil Engineering*, 9(4), 250-258.

Corcho, O., and Fernandez-Lopez, M. (2002). "Ontological engineering: what are ontologies and how can we build them?" *Semantic Web Services: theory, tools and applications*, J. Cardoso, ed., IGI Global, Portugal.

Eastman, C., Teicholz, P., Sacks, R., and Liston, K. (2008). *BIM Handbook: A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers, and Contractors*, Wiley, Hoboken, New Jersey.

Guarino, N., Oberle, D., and Staab, S. (2009). “What is an ontology.” *Handbook on Ontologies*. Springer, Berlin, Germany, 1 – 17.

Hjelseth, E. (2010). "Overview of concepts for model checking." *CIB W78 2010: 27th International Conference*. Cairo, Egypt.

Issa, R. R. A., and Mutis, I. (2006). "Ontology Based Framework Using a Semantic Web for Addressing Semantic Reconciliation in Construction." *EG-ICE 2006*.

Jung, Y., and Joo, M. (2011). "Building information modeling (BIM) framework for practical implementation." *Automation in Construction*, 20, 126 - 133.

Katranuschkov, P., Gehre, A., and Scherer, R. J. (2003). "An ontology framework to access IFC model data." *Journal of Information Technology in Construction*, 8, 413.

Liebich, T. (2010). "Unveiling IFC2x4 - the next generation of openBIM." *CIB W78 2010: 27th International Conference*. Cairo, Egypt.

OMG (2009). "Ontology definition metamodel." <http://www.omg.org/spec/ODM/>. (5/10, 2010).

Paolucci, M., Kawamura, T., Payne, T., and Sycara, K. (2002). "Semantic matching of web services capabilities." *The Semantic Web - ISWC 2002*, Springer, 333.

Pathak, J. (2006). “Difference between Agents and (Semantic) Web Services.” <http://lists.w3.org/Archives/Public/public-sws-ig/2006Oct/0074.html>. (3/15, 2011).

Shen, W., Hao, Q., and Xue, Y. "An agent-based service-oriented approach for facility lifecycle information integration and decision supports." *Proc., 2010 IEEE International Conference on Systems Management and Cybernetics (SMC)*, 1945-1952.

Studer, R., Benjamins, V. R., and Fensel, D. (1998). "Knowledge engineering: Principles and methods." *Data & Knowledge Engineering*, 25(1-2), 161-197.

TNO (2010). "IFC Engine DLL." <http://www.ifcbrowser.com/ifcenginedll.html>. (3/15, 2010).

Vacharasintopchai, T., Barry, W., Wuwongse, V., and Kanok-Nukulchai, W. (2007). "Semantic Web Services framework for computational mechanics." *Journal of Computing in Civil Engineering*, 21(2), 65.

W3C (2004). "Web Services Architecture." <http://www.w3.org/TR/ws-arch/>. (3/15, 2010).

W3C (2009). "OWL 2 Web Ontology Language Primer." <http://www.w3.org/TR/2009/REC-owl2-primer-20091027/>. (3/15, 2011).

Wang, H.-H., Boukamp, F., and Elghamrawy, T. (2010). "An ontology-based approach to context representation and reasoning for managing context-sensitive construction information." *Journal of Computing in Civil Engineering*.

Wetherill, M., Rezgui, Y., Lima, C., and Zarli, A. (2002). "Knowledge management for the construction industry: the e-COGNOS project." *Journal of Information Technology in Construction*, 7, 183.