

Building Interior Model Interpolation between IFC and CityGML

Li Zhu and Huaqing Mao*

Oujiang College, Wenzhou University, Chashan University Town, Wenzhou, Zhejiang, China

Abstract: The use of building interior model in indoor emergency response and building interior navigation is becoming more and more common. Although the LoD4 of CityGML can express building interior spatial information quite well, it is hard to gather or obtain the spatial data in the building. In particular, building interior model needs rich semantics information (e.g. the width of window) to perform extra calculation such as the fire evacuation scenario. This paper addresses this issue by presenting a framework of interpolation between IFC and CityGML, which uses IFCXML as intermediate output file. There are two advantages relative to the direct conversion. One advantage is that IFCXML is defined in XML which has better readability, scalability and flexibility than IFC, and IFCXML can be integrated into XML database or web service. Since IFC contains the whole construction engineering information, the other advantage is that we don't need to convert whole the IFC file. The analysis results showed that this interpolation framework is practical and feasible, and it provides a partially converting solution between IFC and CityGML.

Keyword: BIM, CityGML, IFC, IFCXML.

1. INTRODUCTION

The traditional building design was largely reliant upon 2D drawings (plans, elevations, sections, etc.) and additional attributes. As a result, instead of using 3D model, the majority of interior navigation systems are visualized by 2D. The spatial objects of interior have complicated and various forms, and the spatial relationships between those objects are sophisticated. How to introduce the 3D model into interior application is the key problem to solve that 2D model can't represent 3D spatial interior objects [1, 2]. Xie and Lawver presented a 3D Micro-Spatial Modeling Tools (3D-MSMT) for Building Interiors emergence response [3]. Jiyeong Lee proposed a new topological data model: the node relation structure (NRS) which was achieved by network model of building to analyze the shortest path in 3D Scenario [4, 5]. M. Meijers *et al.* accomplished the logical model to represent the 3D interior structure and applied the model to the intelligence calculation and path evaluation [6]. Those 3D interior models have different purposes, so they choose variants of methods to resolve the problem of representing the spatial relationship. All those models using the structured presentation have some major defects: topological operation is difficult; the semantic descriptions of spatial objects are insufficient; and the scalability of model is limited.

Building information modeling (BIM) is a process involving the generation and management of digital representations of physical and functional characteristics of a facility [7]. Currently, BIM is used for a number of construction engineering applications, including cost-estimations, energy analysis and production planning. The BIM model has rich semantic information and focuses on building interior

information, but it can not process spatial analysis. Conversely, Geographic Information System (GIS) is often used in virtual reality such as digital city, digital agriculture which is mostly large-scale deployed. Other pros and cons between BIM and GIS are described in Table 1. Since the building environment is a primary factor in our daily lives and most of human life is spent inside the buildings, how to construct the model from the exterior into the interior becomes a hot research spot in the field of interpolation between BIM and GIS. An increasing number of GIS applications like interior emergence management, interior utility management, disaster simulations and personal navigation systems require additional information about building interior which is difficult to achieve from traditional methods such as CAD. However, due to the features of the BIM data, it should be probably used in GIS especially building interior.

Table 1. The difference between BIM and GIS.

	BIM	GIS
Emphasis	Indoor	Outdoor
Scale	Small-scale	All scale
Purpose	Construction project	Virtual reality
Spatial Analysis	N/A	Powerful
Semantic Information	Rich	Insufficient

This paper suggests a conversion using the IFCXML as an intermediate step, which focuses on the building interior transformation. Before we discuss the conversion, it is necessary to discuss each model separately which is introduced in section 2 and 3. In section 4, we propose and demonstrate a framework for transformation of building interior informa-

tion between Industry Foundation Classes (IFC) and CityGML. The conclusion and future work are discussed in section 5.

2. RELATED WORKS

As we know, that the BIM is used to integrate and manage the information of the building itself and the GIS is used to analyze and process the exterior environment of the building. It is necessary to create an interpolation framework and compromise the external macroscopic factors and the internal microscopic factors.

Great deals of works have been conducted to explore the transformation BIM data into GIS and vice versa. For instant, Hijazi *et al.* made a brief introduction on an IFC to CityGML Transformation Framework which provides the geometric and semantic transformation of utilities infrastructure inside buildings between the two data models, IFC and CityGML[8]. Isikdag and Zlatanova defined a framework for automatic generation of buildings in CityGML using BIM, which indicated that it is possible to define rules for facilitating semantic matching between BIM and CityGML models[9]. Ruben and Léon developed a CityGML extension called GeoBIM to get semantic IFC data into a GIS context and implemented the conversion of IFC to CityGML in the open source Building Information Modelserver[10]. Irizarry *et al.* presented an approach for integrating GIS and BIM that enables managers to visualize the 3D model of tower cranes in their optimal locations[11].

To sum up, the previous work places great emphasis on geometry conversion that is the most complex part in the transformation. BIM represents the structural components of buildings by using the constructive solid geometry and parametric primitives, while GIS uses the B-rep to represent the topological and geometric information. Previous studies on transformation from BIM to GIS can be summarized as follows:

- 1) The conversion between the IFC and CityGML is directly.
- 2) The conversion is covering all different lower Levels of detail (LoDs) in CityGML.
- 3) Two-way conversion is in development.
- 4) Consider BIM as the data source of GIS, it is only part of BIM file is available, but mostly researches are focus on the whole file transformation.

3. CITYGML AND BIM STANDARD FOR BUILDING INTERIOR MODEL

3.1. CityGML

In recent years, the majority of virtual 3D city models are made by graphics or geometry shapes which ignore the semantic information. Those models used for virtualizations can not give sufficient satisfaction to the demands like subject query, analysis and spatial data mining. To improve the extensibility, reconfiguration and portability of the model, a general modeling solution which can satisfy all kinds of requirements needs to be settled. CityGML is a general model that can share the information for various applications, and it

defines the relationship between the entities and properties. CityGML has been widely used in different fields of application: urban planning, architectural design, environmental simulation, emergency (disaster) management and navigation system. The most important rule of CityGML is the connection between semantics and geometry or topological property.

From the semantic point of view, the entity of real world can be represented through some features such as: walls, windows, doors and rooms. The information of entity includes the relationship of property and feature which can be described by semantics without geometry relationship. In the area of spatial aspect, the feature of spatial objects can be represented by spatial direction and domain. The character of CityGML that combines the semantics and geometry information can process the spatial data in two levels. For example: there are two windows and a door on the wall that can be represented both in semantics and geometry way. Although the CityGML has defined the semantic interface, the semantic information is hard to achieve from the traditional ways.

In CityGML standard, five consecutive levels of detail (LoD) were defined which are used to represent city objects. In terms of representing buildings, 4 out of 5 LoDs are used. As explained by [12] as follows:

1. LoD0 is region model, it use Digital Terrain Model(DTM) to represent the landscape, and it normally is aerial photographic image.
2. LoD1 is block model, it includes the buildings without roof and represent those building by cuboids or cylinders.
3. LoD2 is districts model, it extends and builds upon LoD1, and it adds the roofs and different surfaces to the building.
4. LoD3 is architectural models, it use the texture map to depict realistically the buildings. In this model, building always includes more details such as balconies, bays and projections.
5. LoD4 is interior model, in addition to the external model of building; LoD4 also has 3D interior structures such as hallway, stairs.

LoD4 is the most important model in converting IFC to CityGML. There are several primary classes to represent the room in LoD4, the Room class is used to represent the structure of the interior, and the definition of Room class includes the boundary surfaces of a room such as wall or floor surface. Building Furniture class is used to represent the movable objects such as lamps, table and chairs. The other static object such as pillars and stairs are represented with Building Installation class.

3.2. IFC (Industry Foundation Classes)

The key issue to exchange and share information is the standard which can communicate with different applications. With increasing technological sophistication, there are various data exchange standard. IFC is a common standard for exchanging data in architectural software [13-16] which is organized by IAI (International Alliance for Interoperability). IFC covers the life cycle of architecture engineering

including: planning, design, construction and property management.

In order to optimize and make it easy for the users to learn IFC, the data schema architecture of IFC defines four conceptual layers which is describe as following:

1. Resource layer: this layer is the lowest of the four layers, and it includes all individual schemas containing resource definitions, such as: personnel information, document information, geometry and topology information;
2. Core layer: this layer is consisted of the kernel schema and the core extension schemas. The most general entities are defined in those schemas, and the Identification information such as globally unique id, optional owner and historical information are included in entity definitions;
3. Interoperability layer: the third layer includes schemas containing entity definitions that are specific to a general product, process or resource specialization used across several disciplines. This layer is usually used in exchange and sharing of construction information;
4. Domain layer: the highest layer includes thematic information of each domain. This layer is typically utilized for intra-domain exchange and sharing of information.

In IFC standard, relations among building objects are complicated which include directly and indirectly references. Fig. (1) shows the logical relationship of building interior.

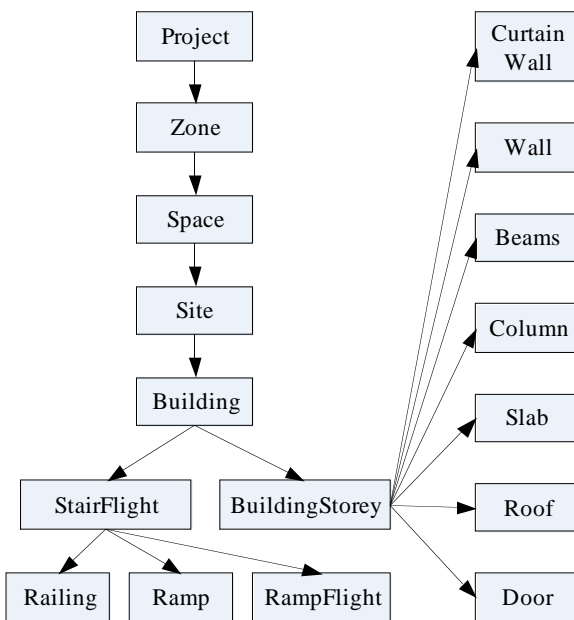


Fig. (1). Logical relationship of building interior.

3.3. IFCXML

In the latest version of IFC 4R4, there is another data exchange file named IFCXML. IFCXML data files are normally given the extension .xml or occasionally .ifcxml or .ifx. The IFCXML representation is an alternative to the IFC representation which is not envisioned to replace the SPF data format of IFC. In offering an XML representation of IFC data, it is anticipated that a broader community of appli-

cations will be able to access a unified schema, representing the building interior environment and the related resources. XML is a textual format for data, and it has a extensive range of supporting utilities and database implementations. It is the basis for most applications and web services. Most of web browsers support XML format, which make the information immediately accessible on computers or the other mobile devices.

Table 2. Difference between IFC and IFCXML.

	IFC	IFCXML
Readability	Poor	Good
File size	Small	Big
File type	spf	ifcxml
Application range	Whole project	Part Project
Compatibility	Poor	Good
GIS support	Poor	Good
Definition language	EXPRESS	XSD
Database support	Poor	Good

4. FRAMEWORK FOR CONVERSION BETWEEN IFC AND CITYGML

In the transaction from IFC to CityGML, it only needs IFC provider building interior model information without the other information like: cost management, collision detection, progress management. However, IFC file does not support partial file transaction. We propose to use IFCXML file as intermediate work file. Although the file size of IFCXML is bigger than IFC's, but it supports partial file extraction and has been widely supported by database. The whole transaction framework is shown in Fig. (2).

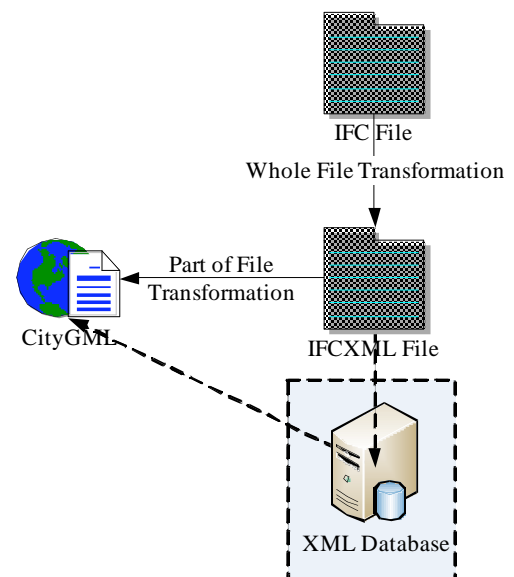


Fig. (2). Conversion framework.

In building interior GIS application, it is not necessary for multiple IFC file, so the XML database is optional in the framework.

4.1. IFC to IFCXML

Since IFC file is designed for whole file data exchange, there are several rules to improve performance:

1. Skip the name of property and data type;
2. Use the abbreviation to compress the file size: T. means true, 1. means 1.0.

While IFCXML is focus on part of file data exchange and readability, the property and data type are fully presented. The example of entity definition in IFC file and corresponding IFCXML file is as following:

IFC file:

```
#102 = IFCWINDOW('0tA4DSHd50le6Ov9Yu0I9X',
#2, 'Window for Test Example', 'Description of Window', $,
#103, #106, $, 1000., 1000., .skylight., .single_panel.);
```

IFCXML file:

```
<Window id="_102">
<GlobalId>0tA4DSHd50le6Ov9Yu0I9X</GlobalId>
<ownerHistory>
<OwnerHistory xsi:nil="true" href="_2">
</ownerHistory>
<Name>Window for Test Example</Name>
<Description>Description of Window</Description>
<ObjectType></ObjectType>
<ObjectPlacement>
<LocalPlacement xsi:nil="true" href="_103">
</ObjectPlacement>
<Representation>
<ProductDefinitionShape xsi:nil="true" href="_106">
</Representation>
<Tag></Tag>
<OverallHeight>1000.0</OverallHeight>
<OverallWidth>1000.0</OverallWidth>
<PredefinedType>skylight</PredefinedType>
<PartitioningType>single_panel</PartitioningType>
```

As previously mentioned, the content of IFCXML file is more fertile than IFC. To convert the IFC to the IFCXML, it is necessary to complement the IFC information which can achieve by scanning the definition EXPRESS file of IFC. To avoid the scanning the EXPRESS file from hard drive frequently, we designed three hash tables to store the metadata in the memory. Those hash tables are Entity-Property, Type-Value, Id-Entity, and their structures are shown in Fig. (3).

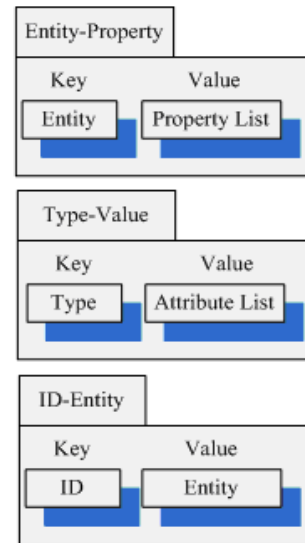


Fig. (3). Hash table for IFC additional information.

For the Entity-Property hash table, we use the IFC entity name as key and use the property set as the value. To optimize the conversion speed, we put the IFCXML entity name into the first place of property list. The example of the IFCWALL Entity-Property hash table is shown in Fig. (4).

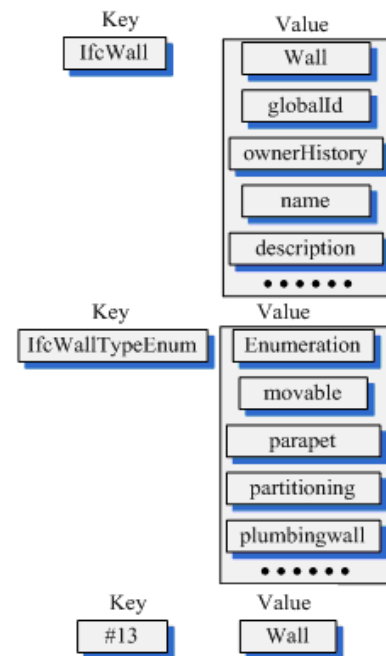


Fig. (4). Examples of HASH Tables.

There are four kinds of property type in IFC and IFCXML including defined types, enumeration types, select types and reference types. We use Type-Value hash table to store those property type. We use IFC type name as key and the range of value list as value. We put the property type into the first place of list. The example of IfcWallTypeEnum Type-Value hash table is shown in Fig. (4).

Since the IFC use the entity id to express the reference relationship, but in the IFCXML, it needs additional entity

name information. We use the Id-Entity hash table to retrieve the entity name by id.

The transaction between IFC and IFCXML falls roughly into three steps:

1) Header transaction:

In IFC file, header section is from “ISO-10303-21” to “DATA”. The head information mapping is described in IFCXML4 specification methodology [19].

2) Entity transaction:

- (a) Read the entity information into string “str_Entity” in unit of row.
- (b) Truncate the substring of “str_Entity” before the “=” as ID information, truncate the substring of “str_Entity” from “=” to “(” as entity name.
- (c) Retrieve the corresponding property list in Entity-Property hash table by using entity name as key value.
- (d) Truncate the substring of “str_Entity” from “(” to “)” as property value list. There is an one-to-one correspondence between the property list and property value list. When the property value in IFC is “\$”, then it is NULL in IFCXML. If the property has aggregated attributes, those attributes can be retrieved from Type-Attribute hash table and put into IFCXML as child element of the property. If the attribute still has aggregated attributes, then we retrieve IFCType hash table until the attributes become simple attributes.

3) Entity reference transaction:

In the property value list, it is maybe reference of the entity like “#ID”, and it is need to retrieve ID-Entity hash table and get entity name to IFCXML.

4.2. IFCXML to CityGML

Since IFC and IFCXML file are the different expressions for same data model, there are one to one relationships between the instance and property. CityGML is designed for representing virtual reality scenarios rather than construction project, so it has mismatch in storage of building information. In order to accomplish the conversion from IFCXML to CityGML, we create four dictionary hash tables, a detailed description of each of these table follows.

- 1) Building element table: it is used for saving the instance of IFCXML corresponding to CityGML. Those instances are wall, beam and column besides door and window. We put instance ID as key and instance name as value.
- 2) IFC instance table: it uses instance ID as key, and sets all the structures of other properties into value.

Those instances are coherent elements of building element such as position, material.

- 3) Relation table: this table contains the relationship between building element and its coherent element. It use building element ID as key and coherent element ID as value.
- 4) Door and window table: in this table, we attach the doors and windows to the wall, so we set wall ID as key and set

the ID list of doors and windows belong to the wall as value.

After the hash tables are created, we are ready to convert the IFCXML file. The conversion process can fall into three levels:

- 1) File level: First step, generate the CityGML header file, then read the IFCXML root element and extract the configure data. Second step, read the child element of root element, and put them as well as their properties into associated dictionary hash table. For the details of this step, see instance level. Third step, traverse the building element table and extract the required data from the other three tables. In the end, construct the building element in CityGML and release the tables.
- 2) Instance level: First step, acquire the instance name and id of element, if the element is building element, put the information into building element table, otherwise skip to next element. For the details of processing building element, see building element level. Second step, analyze the properties by reference the root element configuration and put them into IFC instance table with the instance id. Third step, put the relating instance id into relation table.
- 3) Building element level: First step, extract the related properties of building element from IFC instance table. For the reference property, it can retrieve the data from IFC instance table, and construct the location, shape, orientation of building in CityGML. Second step, retrieve the material id from relation table and achieve the material information by querying the IFC instance table. Third step, retrieve the window or door id from the door and window table and use the id to achieve the information of window or door by querying the IFC instance table. In the end, synthesize all the information to construct CityGML model.

4.3. Analysis Results

A series of conversion times were calculated for the different IFC model. To demonstrate the efficiency of our method, we contrasted conversion time and file size with traditional way which is convert the IFC to CityGML directly. In our experiment, we use four models to demonstrate the relationship between conversion speeds and file size. The four models are described as follows: model1 is a simple building with different furniture; model2 is a family building with basement and furniture; model3 is a complex office building with sixth floors; model 4 is a suburban, dense Mixed-Use (Residential, Retail/Office and Structured Parking) building.

Fig. (6) exhibits that IFC, IFCXML, CityGML have different sets of features. IFC is compacted in design and it has smallest size among three types of file, it is used to data exchange between CAD software with high performance. IFCXML has same content with IFC, it is more readable than IFC, but it is about three times bigger than IFC. Additionally, IFCXML can be stored in database which can query the data effectively and conveniently. CityGML is used for visualization, it is based on XML format, but it has medium size in three types of file because it only needs part content of IFCXML.

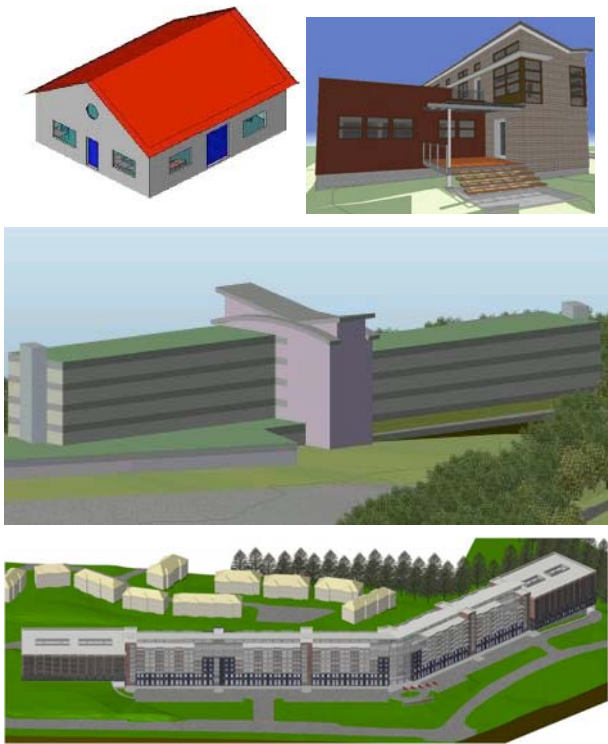


Fig. (5). 3D View of 4 Models [17, 18].

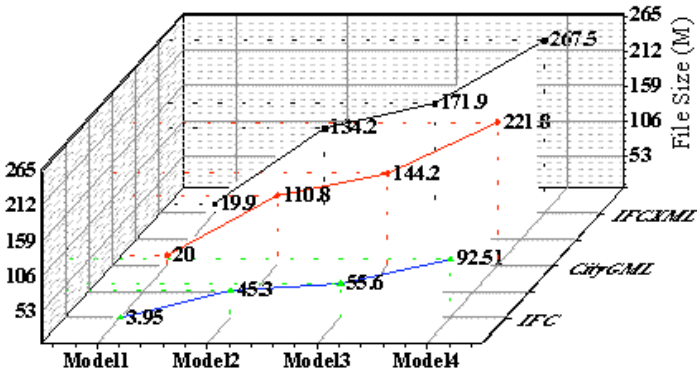


Fig. (6). Different file size of 4 models.

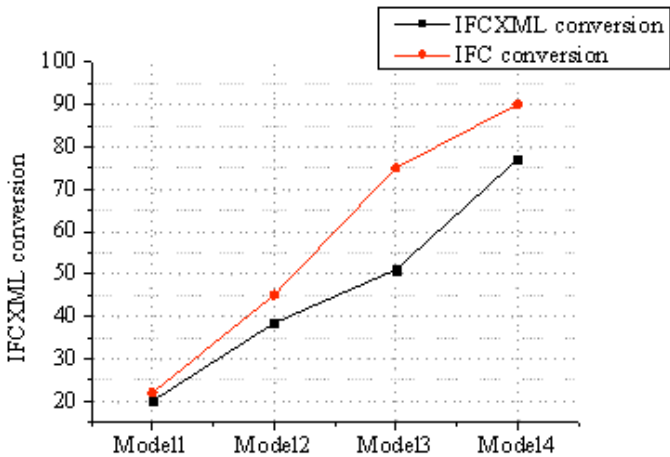


Fig. (7). A comparison of conversion speed between IFC and IFCXML.

Fig. (7) shows that using IFCXML as intermediate file is faster than IFC because IFCXML can retrieve the partial file and XML file format has better query efficiency. Since IFC and IFCXML of model1 have similar file size, the conversion speeds are more or less the same.

CONCLUSION AND FUTURE WORK

Both IFC and CityGML can represent building interior spatial objects. In order to fulfill the demands in indoor GIS application, it is very important to convert the information in IFC to CityGML which includes Semantic and topological information. The background literature review is as follows, IFC model contains all the necessary information for representation of building interior in CityGML model, but the transaction of existing approaches is inefficient cause which makes the IFC does not support the partial file transaction. In general, IFC standards the whole life-cycle of construction project which includes data management, project-planning, design, and property management, most of those information are useless in building interior GIS application. The paper presents a general overview of information transformation from IFC into CityGML models by using IFCXML as intermediate output files which can contribute to the interoperability between BIM and GIS. Limitations of our method are, (a) IFCXML to CityGML conversion is not fully demonstrated. (b) The XML database is not in use. (c) The conversion between IFCXML and CityGML has mismatch issues.

The ongoing research will focus on the identified limitations which are developing a common database that data from IFC and CityGML can be automatically integrated and processed in different analyses.

CONFLICT OF INTEREST

The authors confirm that this article content has no conflict of interest.

ACKNOWLEDGEMENTS

This material is based upon work funded by Zhejiang Provincial Natural Science Foundation of China under Grant No. LQ13D010001. The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers.

REFERENCES

- [1] F. Bian, "Principle and Methodology of Digital Engineering," Surveying and Mapping Press, vol. 12, 2006.
- [2] F. Bian, "Introduction to Geospatial Information," Surveying and Mapping Press, vol. 8, 2006.
- [3] Y. Xie, and G. Lawver, "3D Micro-Spatial Modeling Tools (3D-MSMT) for Building Interiors," *NIJ Grant*, 2007.
- [4] J. Lee, "3D Data Model for Representing Topological Relations of Urban Features," In: *Proceedings of the 21st Annual ESRI International User Conference*, San Diego, CA, USA, 2001.
- [5] J. Lee, "A Three-Dimensional Navigable Data Model to Support Emergency Response in Microspatial Built-Environments," *Annals of the Association of American Geographers*, vol. no. 3, pp. 512-529, 2007.
- [6] M. Meijers, S. Zlatanova and N. Pfeifer, "3D geoinformation indoors: structuring for evacuation," In: *Proceedings of Next generation 3D city models*, Bonn, Germany, 2005.
- [7] C. Eastman, P. Teicholz, et al. "BIM Handbook: A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers and Contractors," Hoboken, New Jersey, Wiley, 2011.
- [8] I. Hijazi, M. Ehlers, S. Zlatanova, and U. Isikdag, "IFC to City GML transformation framework for geo-analysis: a water utility network case," In: Maeyer P de, Neutens T, Rijck M de (eds) 3D GeoInfo, Proceedings of the 4th International Workshop on 3D Geo-Information, 2009.
- [9] U. Isikdag, and S. Zlatanova, "Towards defining a framework for automatic generation of buildings in CityGML using building Information Models," In: Lee J, Zlatanova S (eds) 3D Geoinformation and Sciences, Springer, Berlin, Heidelberg, 2009.
- [10] R. de Laat, and Léon. van Berlo, "Integration of BIM and GIS: The Development of the CityGML GeoBIM Extension. Advances in 3D Geo-Information Sciences, Lecture Notes in Geoinformation and Cartography, 2011.
- [11] J. Irizarry, and P. K. Ebrahim, "Optimizing location of tower cranes on construction sites through GIS and BIM integration," *Journal of Information Technology in Construction (itcon)* vol. 17, pp. 351-366, 2012.
- [12] Gerhard Gr ger, Thomas H Kolbe, Angela Czerwinski, Claus Nagel. City Geography Markup Language (CityGML) Implementation Specification. Open Geospatial Consortium Inc. 2008-01-20 CityGML2007.
- [13] Autodesk Revit. <http://www.autodesk.com/products/autodesk-revit-family/overview>
- [14] Bentley CAD. <http://www.bentley.com/en-US/>
- [15] MagicCAD. <http://www.magicad.com/en>
- [16] ACE Project. <http://www.gehrytechnologies.com/>
- [17] Karlsruhe Institute for Technology. <http://www.iai.fzk.de/www-extern-kit/index.php?id=1135&L=1>
- [18] Nemetschek Vectorworks, Inc. <http://www.vectorworks.net/bim/projects.php>
- [19] Official homepage of CityGML. <http://www.citygml.org>.
- [20] Liebich, Thomas; Weise, Matthias. ifcXML4 Specification Methodology, 2012.

Received: April 10, 2015

Revised: May 20, 2015

Accepted: June 15, 2015

© Zhu and Mao; Licensee Bentham Open.

This is an open access article licensed under the terms of the (<https://creativecommons.org/licenses/by/4.0/legalcode>), which permits unrestricted, non-commercial use, distribution and reproduction in any medium, provided the work is properly cited.