IFC-CityGML LOD Mapping Automation based on Multi-Processing

T.W. Kang[a] and C.H. Hong[b]

[a]ICT Lab, Korea Institute of Civil engineering and building Technology, Republic of Korea
[b]ICT Lab, Korea Institute of Civil engineering and building Technology, Republic of Korea
E-mail: laputa99999@gmail.com, chhong@kict.re.kr

ABSTRACT

Over the last few years, to increase the competitiveness of domestic construction industries, the governments of US, UK, Japan, and other developed countries have introduced Building Information Modeling (BIM) as part of their support policy; as a result, BIM information is increasing rapidly. In this regard, the need for studies on integrating or mapping BIM object information to Geographic Information Systems (GIS) is increasing. These studies have been conducted with the purpose of utilizing the advantages of BIM and GIS; however, many difficulties are encountered in model integration and mapping because of the different schema structures of such studies. In this study, the productivity of the mapping work is improved by developing a method that automates the Level of Detail (LOD) mapping using BIM and GIS standard models, Screen Buffer Scanning-based multiprocessing, Industry Foundation Classes (IFC) model, and the CityGML model. The proposed method can dramatically improve the productivity of LOD shape modeling work because, unlike conventional methods, the proposed method uses multiprocessing. Finally, the developed results are quantitatively evaluated and future study directions are derived.

Keywords – IFC; CityGML; LOD; multiprocessing; mapping; conversion

1 Introduction

Building Information Modeling (BIM) technology was applied extensively to the architecture, engineering, and construction (AEC) industry sector in 2005. Since then, to increase the competitiveness of domestic construction industries, the governments of many countries introduced BIM as part of its support policy; as a result, BIM information is increasing rapidly. In this context, the need for studies on integrating or mapping BIM object information to Geographic Information Systems (GIS) is increasing. These studies have been conducted with the purpose of utilizing BIM advantages, which can manage the entire lifecycle of facility information, and those of GIS, which can provide decision-making information based on state and city [20]. For this purpose, standard information models such as Industry Foundation Classes (IFC), CityGML, and the Ontology model, in addition to technologies such as web services, are used.

IFC is a data exchange format developed with the purpose of describing information about buildings. This format is neutral and open and describes information using the EXPRESS language. Until recently, the IFC format had been developed and maintained by buildingSMART. CityGML is an industry standard model developed by Open Geospatial Consortium (OGC) for exchanging GIS data. IFC and CityGML possess rich semantic information from the perspective of buildings and cities. Presently, studies that analyze schema characteristics have been conducted to apply IFC and CityGML effectively to industries [16, 17].

In particular, CityGML is optimized to manage and express city facilities information by GIS. Recently, CityGML has been applied to various use cases, such as energy management [11], water and wastewater utility management [2], indoor and outdoor navigation [6, 7], and urban planning [4].

The two standard model schemata are objectively analyzed and designed, wherein schemata consist of elements such as object meaning and attributes, shape and object relationship, and status. The object analysis targets for IFC and CityGML differ in the building lifecycle information and GIS urban facility objects, and in the type and number of objects. In particular, when expressing shape in GIS, there is a large difference in the Level of Detail (LOD). The shape structure map of IFC has a parameter-based shape or Boundary Representation (B-Rep) structure. On the other hand, given that CityGML has a surface-based structure, the outer surface and the inner surface are divided [22, 23]. These features cause difficulties when converting BIM standard models such as IFC to GIS standard models such as CityGML. In particular, when considering the conversion processing time, no significant problems are encountered during the conversion of a single facility. However, when targeting a large number of facilities management, the required conversion processing performance adversely affects conversion work efficiency. This is because the complex shape structure of IFC requires a significant number of operations and computer processing time during conversion that corresponds to the LOD shape structure as defined by CityGML.

In the present study, the multiprocessing-based IFC-CityGML LOD generation automated method, which improves performance problems during conversion of IFC to the LOD model of CityGML, is proposed. For this purpose, LOD shape generation automation method using a Screen Buffer Scanning-based Multi-Processing (SBSMP) that includes semantic mapping is defined and developed. The proposed method can dramatically improve LOD generation automation performance because, unlike conventional methods, the proposed method uses screen buffer scanning-based multiprocessing. Finally, the developed results are evaluated and future study results are derived.

2 Study objectives

In this study, an LOD generation automated method is proposed for mapping object shape from IFC to CityGML using screen buffer considering multiprocessing. This paper is organized as indicated in the following paragraph.

The points to be considered during the study and the improvement areas of the present study are clearly defined by reviewing attempts related to the present study through the study and literature survey related to the present study topics. A method for Geometry LOD mapping is defined in Section 4, 5. For this, the schema structures of the BIM and GIS standard models, IFC and CityGML, are first analyzed. In particular, the shape representation structure related to the present study is closely viewed. Furthermore, the SBSMP-based LOD generation algorithm and related structure proposed in this study are defined. The rules for semantic mapping from IFC to CityGML are defined in Section 5.4. The results of implementing the proposed method as a prototype are shown in Section 6. The effectiveness of the proposed method in generating LOD is proven quantitatively using the developed prototype. In the last section, conclusions are derived by summarizing previous study content, and suggestions for future studies are made. (Figure 1) is a representation of the study method.

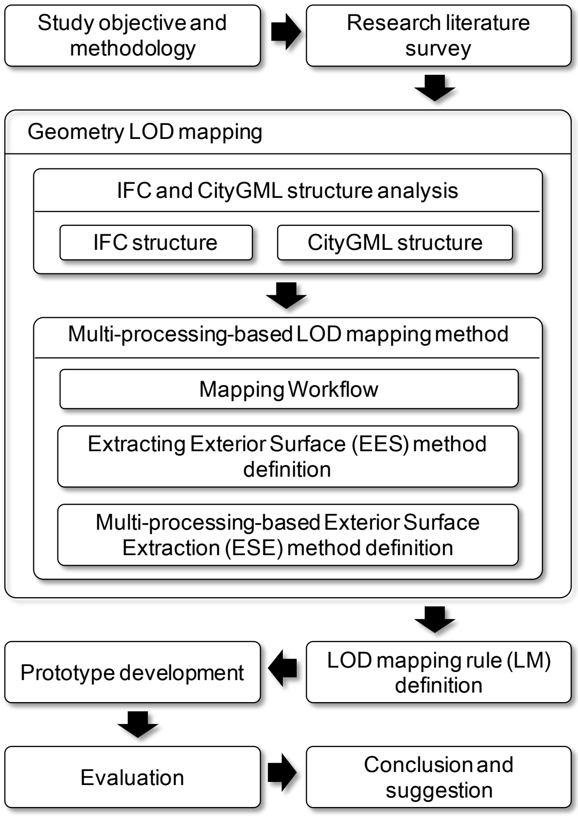


Figure 1. Research flow

3 Literature Survey

CityGML is a GIS-based spatial information model based on GML for shape modeling. The LOD concept is implemented to effectively represent a significant number of objects on GIS, wherein the objects include transportation, urban, water resource, land, and other objects [23]. Studies related to normal methods for developing 3D Spatial Data Infrastructure (3D SDI) have been conducted [14, 15]. For example, some of the commercial tools used for converting from IFC to CityGML include the Feature Manipulation Engine (FME) plug-in of the Bentley Map; a conversion case in which said plug-in was used is mentioned briefly in this study. Because the FME tool regularizes the conversion process between different data models, the LOD generation method can be determined; consequently, a method that maps from various data sources, such as 2D drawings, point cloud, or photo images, to a 3D model can be defined. Studies related to LOD generation from various data sources include an analysis case wherein the mapping process between photo image and 3D model was analyzed using a CityGML generation tool, Toposcopy [18]. In addition, there are studies on generating LOD of CityGML through a point cloud obtained from a laser scanner [1].

There have been studies to further automatize the mapping from IFC to CityGML using methods other than those used in previous studies. In addition, there have been studies related to the extraction of the LOD3 shape of CityGML from IFC [8]. In this study, the structure of IFC and CityGML is analyzed, and to maintain possible geometric properties during the conversion from IFC to CityGML, an LOD3 shape generation method is studied through a geometric calculation that requires significant calculation time, such as cross calculation. Furthermore, there have been studies on converting automatically from IFC to CityGML through semantic mapping by focusing on object and attribute information conversion [21]. Moreover, there have been study cases on an algorithm that includes semantic mapping, and that generates LOD of CityGML by dividing the inner and outer surface of a given shape [5]. In this study, to distinguish the indoor and outdoor three-dimensional shapes that constitute the IFC model, the even-odd rule (EOR) [12] is applied through the polygon that composes a shape boundary, and through infinite linear cross calculation.

Because CityGML is a complex semantic model, in order to visualize large city facility objects on the actual GIS, significant geometric calculation time is required. There have been studies to improve calculation performance using GPU [19].

Heretofore, we examine the study trends related to the present study. There have been continuous attempts to utilize the advantages of BIM and GIS and apply them to use cases such as more effective urban facility management; in this regard, there have been continuous attempts to integrate or map the information models of each area. This study is related to the mapping of BIM with a focus on GIS; in particular, the calculation performance problem encountered during the mapping of the LOD of GIS using a standard model is solved using multiprocessing. Most existing studies focused on semantic mapping. However, studies on improving processing performance during LOD mapping from large numbers of high capacity IFC models to CityGML models are relatively fewer.

4 Geometry LOD mapping

4.1 Overview

IFC and CityGML schemata are object models built through object-oriented principles; various class types and properties are defined for the exchange of semantic information from buildings and cities. In general, the object structure is constituted from object, property, method, and relationship components. The objects are instantiated by the class types that define the object frame. Properties are defined by basic type, such as whole numbers and real numbers, and by scalable type, such as list and structure. From the object-oriented perspective, shape can be viewed as a property in the case of IFC or CityGML. However, because shape is treated as an important element in the present study, it should be classified and managed separately. Method is defined in the form of function in classes with the purpose of verifying the integrity of the object information. Relationship defines relationships such as ‘inclusion’, ‘derivation’, and ‘dependency’ between classes. In terms of semantics, relationships have a meaning of ‘near’, ‘meet’, and ‘connection’. With regard to all these parts, it is difficult to convert one schema to another schema model [9, 10]. The present study is limited to mapping with regard to the shape and properties of objects.

For the two schema models to be able to mutually map completely without losses, all the object-oriented elements constituted by the schemata must be mapped in a ratio of 1:1. In the case of N:1 mapping, the way in which N elements are converted and mapped to one should be considered. In this case, depending on the type of mapping adopted (bidirectional or unidirectional mapping), the element type information is added during conversion, and can be added to the property information. In the case of 1:N mapping, as is the topic of the present study (for example, the object shape of IFC should be mapped to the LODs of CityGML), to map the original source elements to the converted target elements, there is need for a process that conceptually separates the original source elements according to the type of target elements, and then performs the conversion. Furthermore, the conversion process differs according to bidirectional mapping and unidirectional mapping. It is impossible to map IFC and CityGML schemata in a ratio of 1:1 for the aforementioned reason.

4.2 IFC and CityGML structure analysis

In the present section, the structure of CityGML and IFC is analyzed for object shape mapping. In particular, the structure is analyzed with a focus on the geometrical structure required for LOD generation.

4.2.1 IFC structure

IFC consists of set relationships between entire buildings, building space, and building member. This implies a whole/part relationship that is supported by IfcRelAggregates. Because a relationship always occurs between two objects, the relationship is defined by an aggregates/part relationship. Aggregates are defined by a RelatingObject of IfcObjectDefinition type, and parts are defined by RelatedObjects. For example, a building is an aggregate and each floor of the building is parts. IFC spatial structure is divided into four types of concepts: IfcSite, IfcBuilding, IfcBuildingStorey, and IfcSpace.

IfcBuildingStorey can include building members such as IfcWall, IfcSlab, and IfcColumn; these building members are represented by classes generalized by IfcProduct. Space and building member objects have a relationship with each other.

IfcProduct object includes IfcProductRepresentation to represent a given shape. IfcProductDefinitionshape and IfcLocalPlacement, which are included by this object, allow multi-shape representation. Shapes such as 'BoundingBox', 'SurfaceModel', 'Brep', and 'MappedRepresentation' are possible [3]. 'BoundingBox' is used to represent the boundary box of an IfcProduct object, and 'SurfaceModel' is used during the representation of a multi-surface model. Shape is defined by IfcCurve, IfcPoint, IfcSurface, and IfcSolidModel, which are derived from IfcGeometricPresentationItem. Each shape object defines coordinate points and curve/surface parameters that represent the shape. By adopting a boundary representation method, the 'Brep' approach represents topology using IfcFace, IfcLoop, IfcEdge, and IfcVertex, which are derived from IfcTopologicalRepresentationItem objects, and then represents the shape that is topologically linked to the represented topology.

4.2.2 CityGML structure

CityGML is an industry standard model for managing and visualizing urban facility information based on the GML of OGC. Based on the top class object, GML defines Geometry that represents the geometric model, Feature that represents a geographic feature, and Definition class that defines the data dictionary. CityGML extends the geographic feature model by facilities, transportation, water resources, and topography through the CityObject class derived from Feature; to represent this shape, Geometry is extended. To address many city objects, the city object class is derived from the FeatureCollection class, which is a feature basic class [24]. FeatureCollection can manage many city objects because it uses a composite design pattern [13] as object containers. This acts similar to the IfcRelContainedInSpatialStructure relationship of IFC. CityGML divides shape representation levels from LOD0 to five in order to effectively visualize many city facilities. This point is significantly different from the shape representation method of IFC.

Each LOD is defined by _GeometryPrimitives such as _Solid, _Surface, and _Curve, which are derived from _Geometry. Because shape is set from such Primitives, complex shapes are implemented through composite design pattern structures such as MultiSolid, MultiCurve, and MultiSurface, which are derived from _AbstractGenericAggregate [24]. Shape structure is relatively simple compared to IFC, and cannot support topology such as IfcLoop. In this study, MultiSurface is generated based on the application rule for each LOD by extracting complex shapes from each IfcProduct object of IFC, and by automatically extracting the outer surface.

5 Multi-processing-based LOD mapping

5.1 Mapping workflow

LOD should be mapped by considering the shape model structure of IFC and CityGML, which are analyzed in the previous section. First, for LOD mapping, the geometry representation part of IfcProduct needs to be converted to a simplified B-Rep structure of CityGML form. B-Rep, which is converted by each IfcProduct, should extract Exterior Surfaces (ES) for LOD mapping. LOD mapping of CityGML and IFC should occur according to the semantic similarity between objects, and according to the LOD concept of CityGML. Therefore, the ES derived for each B-Rep of each IfcProduct is mapped to the LOD of CityGML using the mapping rule that defines the semantic and LOD mapping method.

The LOD mapping flow of the process discussed presently is as shown in (Figure 2).

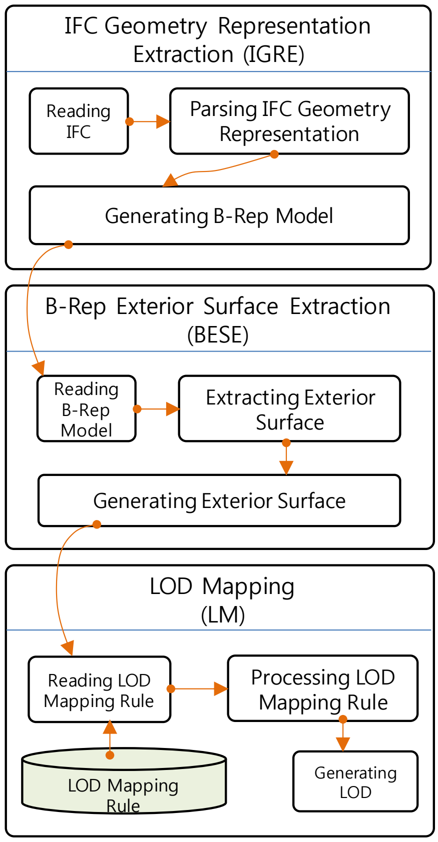


Figure 2. CityGML LOD mapping flow chart in IFC

The processing operations for each major step are listed in (Table 1). Among these steps, EES and PLM, which play a critical role in LOD mapping, are examined in detail in this study.

Table 1. Definition by each CityGML LOD mapping flow steps in IFC

Phase	Description
Parsing IFC Geometry Representation (PIGR)	Geometry and Topology information, required for generation of surface that composes shape, is derived by parsing geometry and topology structure of IfcProductRepresentation with regard to IfcProduct.
Generating B-Rep Model (GBRM)	To convert to LOD with respect to CityGML objects, conversion is performed by B-Rep model of mesh structure form that can map to MultiSurface.
Extracting Exterior Surface (EES)	In B-Rep model, ES is extracted using external surface extraction algorithm.
Processing LOD Mapping Rule (PLM)	Based on predefined LOD mapping rule, LOD mapping is performed after semantic mapping from IFC object to CityGML.
Generating LOD (GLM)	Mapped LOD is generated in CityGML format.

5.2 Extracting Exterior Surface (EES)

Because IFC is a complex shape model, it is converted from mesh structure to a simplified B-Rep structure in order to effectively extract an external surface. The external surface is extracted using this B-Rep structure. The known methods that are used for external surface extraction in existing studies include EOR [12] based on the LOD generation method [5].

For the EOR-based method, a camera is placed around a complex shape model used for extracting an external surface, and a virtual eye line $S(P_1, P_2)$ is created with the camera, which is placed externally with respect to the center of each constituent element of the complex shape. After calculating the intersection point i of the surface that constitutes the eye line S and the complex shape, if i is an odd number, the relevant constituent element is enclosed inside the other surface, and therefore, it is removed. Accuracy of the outer surface to be extracted is related to the shape of the complex model. In the case of a mild shape, only the EOR test can be performed by installing the camera in the representative direction. However, in the case of a complex shape (particularly, a concave shape), the outer surface cannot be extracted properly. Usually, when assuming a cube, in order to surround each surface by many camera viewpoints, 24 cameras are installed for three grades of points outside the relevant surface.

As shown in (Figure 3), the SBS-based method dramatically increases processing speed compared to the EOR method because it uses the Z-buffer test during rendering. The calculation speed for the EOR method is extremely slow because such calculation is performed proportionally to the number of all surfaces that compose the complex shape model. This is because the 3D surface-straight-line required during the EOR test demands a high real calculation cost. However, the SBS method scans the buffer using a simple Z-buffer test with regard to a screen buffer that renders the surfaces of relevant scenes per camera during the process of testing the external surface; for this reason, only the pixel values that confirm the external surface are checked. Moreover, because the SBS method uses the Z-buffer test or high-speed multiprocessing, external surfaces can be extracted at a high speed.

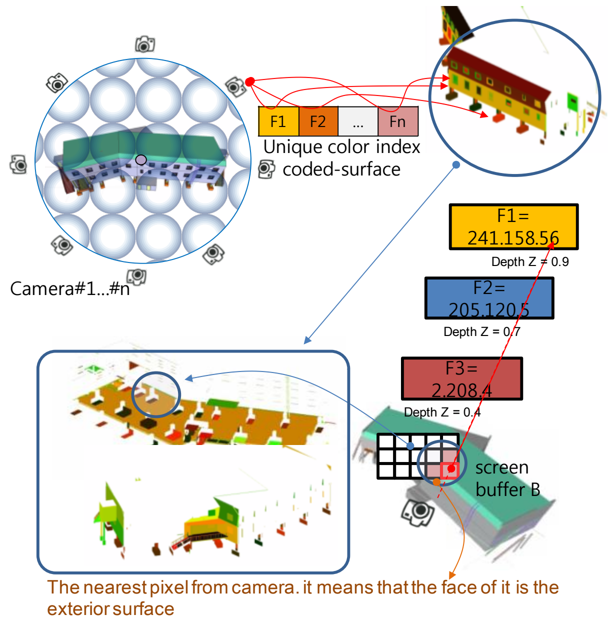


Figure 3. SBS-ESE

To define the pixel values that can confirm the external surface, the screen buffer (SB) should be scanned after setting a unique surface identifier (USI) for each surface that constitutes B-Rep using a color code (Unique Color Index—UCI), and after rendering the relevant surfaces. Using the scanned UCI, the external surface can be found directly by the B-Rep model. Because the screen buffer is generated by capturing one surface through various cameras, the scanned UCI might be duplicated. This duplication problem can be solved using techniques such as Hash-map. To define the screen buffers (n) with a screen image width (w) and height (h) about the B-rep Model, M was defined like below equations (1), (2).

$$M = \sum_{i=1}^{n} \sum_{j=1}^{w} \sum_{k=1}^{h} SB_{i,j,k} \quad (1)$$

$$SB = \{UCI, \ldots\} \quad (2)$$

5.3 Multi-processing-based Exterior Surface Extraction (ESE)

In the case where the number of facilities with BIM number more than a million in city facility management use cases, and if extraction of the outer surface of each facility shape requires a considerable amount of time, an adverse effect occurs on computing time and cost during extraction. To improve such problems, the multiprocessing method is considered based on the SBS-ESE introduced in the previous section.

The SBS method independently scans the screen buffer generated by each camera, and requires synchronization processing between each buffer. In the case where the functions processed through multiprocessing can access and modify the same data sets, a synchronization process is required; in this case, it might be difficult to improve processing performance.

The multiprocessing methods presently known include the OpenMP method that utilizes the threading method of the multi-core CPU, and the Compute Unified Device Architecture (CUDA) method developed by NVIDIA, a development company that produces high performance graphic accelerators. Both methods use the same algorithm for external surface extraction and parallel processing.

In the present study, two types of approaches can be used for parallel processing. One is a method that divides the screen buffer for parallel processing unit (u), and the other is a method that processes in parallel the screen buffers generated by each camera. To allocate the buffers to parallel processing units, the divided processed data M_u is used in each screen buffer scan parallel processing function $SBS_{MP}()$. The $SBS_{MP}()$ extracts UCI by scanning the screen buffer assigned to the function, and processes the duplicate UCI. Equations (3), (4) show the algorithm flowchart that represents this process.

$$M_u = \sum_{e=1}^{u} M_e \quad (3)$$

$$ESM = SBS_{MP}(M, e) \quad (4)$$

5.4 LOD mapping rule (LM)

As mentioned before, for LOD mapping from IFC to CityGML, IfcProduct objects should first be mapped to the CityObject of CityGML through a semantic-based rule, and then LOD mapping should be performed. However, in the case of LOD1, after simply obtaining the Footprint of a building or facility, the results extruded in the height direction and the results extruded in the elevation direction are processed through cross Boolean operation; for this reason, a complex rule is required. Therefore, only the relevant operators are defined. For LOD mapping considering this part, the semantics mapping rule shown in (Figure 4) is defined.

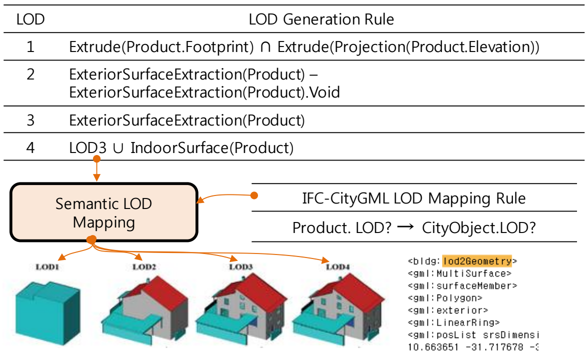


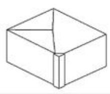
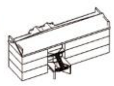
Figure 4. IFC-CityGML Semantic LM (IC-LM) definition.

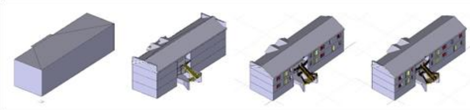
Depending on whether IfcProduct is within a building or outside a building, IfcProduct differs according to semantic concepts during mapping to the CityObject of CityGML. For example, in the case of IfcBeam, it should be mapped to IntBuildingInstallation for an indoor space, and to OuterBuildingInstallation for an outdoor space. Therefore, special operators such as '[Int | Outer]' can be attached in front of the mapped CityObject element. In this case, the mapping targets should be changed as inner building element or outer building element according to semantics.

6 Prototype development and evaluation

For this study, a prototype was developed using the MP-ESE algorithm and the LOD mapping method proposed in Section 4, 5. In the prototype, to confirm the effect of multi-processing-based outer surface extraction performance, EOR-ESE and MP-ESE were viewed by implementing a CPU-based single thread, an OpenMP-based multi-thread method. Because the sample used is an IFC model of small and medium scale, the inside indoor space was modeled.

Table 2. IFC-CityGML LOD mapping results

IFC Data	LOD1	LOD2	LOD3	LOD4
EOR Test Sample#1 (0.559 Mb)				
EOR Test Sample#2 (2.257 Mb)				

EOR Test Sample#3 (28.188 Mb)	
SBS Test Sample#1 (0.559 Mb)	
SBS Test Sample#2 (2.257 Mb)	
SBS Test Sample#3 (28.188 Mb)	

The prototype and sample results listed in (Table 2) indicate that mapping is processed for each LOD of CityGML through the proposed method. For comparison with other methods, a test was performed with the EOR-based method; this method should perform cross calculations evenly on all surfaces of the sample shape with respect to the eye line from the camera to the center of the object; for this reason, the possibility of calculation error occurrence is greater compared to the SBS-based method, and the phenomenon of not being able to extract the outer surfaces properly can be seen.

Table 3. Performance test results (second. Intel(R) Core(TM) i7 CPU Octa-core 860 2.80GHz. 12 GB RAM. 64-bit Operating System).

Algorithm	Sample#1	Sample#2	Sample#3
#1. EOR-based (single-thread)	16.75	211.79	8700.30
#2. EOR -based (multi-thread)	16.68	85.68	3141.55
#3. SBS-based (single-thread)	0.68	0.69	2.31
#4. SBS -based (Multi-thread)	0.67	0.68	2.31
#5. SBS -based (OpenMP)	0.60	0.62	2.29
Improved ratio (#1 / #5)	27.81	342.45	3805.76

As indicated in (Table 3), the performance of external surface extraction processes (in second units) were compared using the samples and prototype developed in Section 5. The results listed in (Table 3) indicate that the MP-SBS-based ESE method proposed in this study has maximum performance improvement that is greater than 27.8-3805.8 times that of other methods.

7 Conclusion and suggestions

In this study, related attempts were reviewed and a method was defined for Geometry LOD mapping from IFC to CityGML; furthermore, a SBSMP-based LOD generation algorithm and related structure were proposed. The proposed method was implemented as a prototype that was tested with a given sample. The test results showed that the proposed method had a minimum 27.8-3805.8 times performance improvement compared to existing methods. It can be confirmed that the difference in processing performance is less according to the size of IFC data. Moreover, it was quantitatively proven that the algorithm has higher efficiency compared to other methods.

In the present study, to interoperate BIM and GIS information models on standardized IFC and CityGML, the shape structure, which is one of the most heterogeneous structures of the two standard models, was proposed by semantically generalizing the method that performs LOD mapping through a multiprocessing-based SBS method; this type of proposal has great significance. Therefore, the proposed method in this study can be applied to other similar 3D BIM and GIS shape mappings.

However, the present study has a limitation in that it considers unidirectional mapping only, and does not consider mapping related to properties and topology. Future studies will be aimed at methods that allow bidirectional mapping and property and topology mapping.

Acknowledgements

This research was supported by a grant from a Strategic Research Project (Development of BIM/GIS Interoperability Open-Platform 2015) funded by the Korea Institute of Construction Technology.

References

[1] Akmalia, R., et al. TLS for generating multi-LOD of 3D building model. *In: IOP conference series: Earth and environmental science*, 18 (1), 012064. 2014.

[2] Becker, T., et al. Initial investigations for modeling interior utilities within 3D geo context: Transforming IFC-interior utility to CityGML/UtilityNetworkADE. *In: Advances in 3D geo-information sciences*, 95–113, 2011.

[3] buildingSMART. IFC2x Edition 4 alpha version. Available from: http://www.buildingsmart-tech.org/ifc/IFC2x4/alpha/html/index.htm, 2008.

[4] Cerny, M., et al. Towards an information rich 3D

city model: Virtual NewcastleGateshead *GIS integration. In: Proceedings of 12th international conference on computers in urban planning and urban management. Alberta, Canada*, 2011.

[5] Cheng, J., Deng, Y., and Du, Q.. Mapping between BIM models and 3D GIS city models of different levels of detail. *International conference on construction applications of virtual reality*, 502–514, 2013.

[6] Döllner, J., et al. Towards an indoor level-of-detail model for route visualization. *In: Mobile data management: Systems, services and middleware. MDM'09. Tenth international conference on. IEEE*, 692–697, 2009.

[7] Domínguez, B., Feito, F.R., and García, Á.L. Semantic and topological representation of building indoors: *An overview. CNKI proc., the International Society for Photogrammetry and Remote Sensing (ISPRS)*, 2011.

[8] Donkers, S. Automatic generation of CityGML LoD3 building models from IFC models. Dissertation (PhD). Delft University of Technology, 2013.

[9] El-Mekawy, M. Integrating BIM and GIS for 3D city modelling, 2010.

[10] El-Mekawy, M., Östman, A., and Shahzad, K. Towards interoperating CityGML and IFC building models: A unified model based approach. *In: Advances in 3D geo-information sciences. Springer Berlin Heidelberg*, 73–93, 2011.

[11] Fan, Q., Wu, W., and Yang, X. GIS-BIM based virtual facility energy assessment (VFEA)—Framework development and use case of California State University, Fresno. *In: Computing in civil and building engineering, ASCE*, 339–346, 2014.

[12] Foley, J.D. *Computer graphics: Principles and practice*. Addition-Wesley, 1990.

[13] Gamma, E., et al. *Design patterns: Elements of reusable object-oriented software*. Pearson Education, 1994.

[14] Goos, J., et al. A generic approach for 3D SDI in the Netherlands. *In: Proceedings of the joint ISPRS workshop on 3D city modelling & applications and the 6th 3D GeoInfo conference*, 26–28, Wuhan, China. 2011.

[15] Goos, J., et al. Towards a national 3D spatial data infrastructure: Case of the Netherlands. *Photogrammetrie-Fernerkundung-Geoinformation* 2011, 6, 405–420, 2011.

[16] Gröger, G., Kolbe, T.H., and Plümer, L. CityGML: Interoperable access to 3D city models. *In: Geo-information for disaster management. Springer Berlin Heidelberg*, 883–899, 2005.

[17] Gröger, G. and Plümer. L. CityGML–interoperable semantic 3D city models. *ISPRS Journal of Photogrammetry and Remote Sensing*, 71, 12–33, 2012.

[18] Groneman, A. and Zlatanova, S., Toposcopy: A modeling tool for CityGML. *Proceedings of the GSDI 11*, 2004.

[19] Guang, Z., et al. LOD methods of large-scale urban building models by GPU accelerating. *In: Computer Science and Network Technology (ICCSNT), 2012 2nd international conference on. IEEE*, 853–858, 2012.

[20] Hong, C.H. and Kang, T.W. The architecture development for the interoperability between BIM and GIS. *International conference on construction applications of virtual reality*, 480–491, 2013.

[21] Isikdag, U. and Zlatanova, S., 2009. Towards defining a framework for automatic generation of buildings in CityGML using building Information Models. In: 3D geo-information sciences. Springer Berlin Heidelberg, 79–96.

[22] Kolbe, T.H. CityGML–Background and design. *IKG–Institut für Kartographie und Geoinformation Universität Bonn*, 2006.

[23] Kolbe, T.H. CityGML tutorial. *In: Presentation in the 1st joint workshop on the Sino-Germany bundle project: Interoperation of 3D urban geoinformation.* China, 2007.

[24] Open Geospatial Consortium. Candidate OpenGIS® CityGML Implementation Specification. City Geography Markup Language, 2007.