A light weight algorithm for large-scale BIM data for visualization on a web-based GIS platform

J. E. Kim[1], C. H. Hong[1] & S. D. Son[2]
[1]ICT Convergence and Integration Research Institute,
Korea Institute of Civil Engineering and Building Technology, Korea
[2]Gaia3D, Korea

Abstract

BIM technology contains data from the life cycle of buildings through 3D modelling. One building produces a huge file because of the massive amount of data. Industry Foundation Classes (IFC), which is the representative standard, has some serious issues about data processing of large-scale data based on geometry and property data of objects. It increases the rendering speed and constitutes much of a graphics card, so large-scale BIM data is inefficient to the user for screen visualization. The light weighting has to resolve processing and quality of the program essentially. This study has been researched and confirmed with regard to light weight techniques from domestic and overseas research. Based on this research, to control and visualize large data effectively, we proposed and verified the algorithm which is able to optimize the BIM character. The general aim of this paper is to propose and verify the light weight algorithm making the best use of property of BIM to control and visualize the large-scale BIM data effectively.
Keywords: building information modelling, geographic information system, visualization, data light weighting, block-reference.

1 Introduction

These days according to an increase of needs of Building Information Modelling (BIM) technology, a lot of meaningful research has been studied for the construction of 3D spatial information using architectural BIM data in domestic and overseas projects. However different data schema between BIM and Geographic Information System (GIS) takes into effect the incompatibility and

WIT Transactions on The Built Environment, Vol 149, © 2015 WIT Press
www.witpress.com, ISSN 1743-3509 (on-line)
doi:10.2495/BIM150301

there are no standards for interoperability. BIM technology contains data from the whole life cycle of building (Plan-Design-Construction-Maintenance) through 3D modelling as well as 2D drawing data, so the building generally produces a huge data file. Industry Foundation Classes (IFC), which is the representative standard, has some serious issues about data processing of large-scale data based on geometry and property data of objects. They make the rendering time increase and require much volume within the graphics card, and finally the screen visualizing is inefficient.

The light weighting of large-scale data has to be solved for processing and quality of the user's program. The overseas research has been figuring out the diverse solutions for that. The general aim of this research is to propose and verify the light weight algorithm making the best use of property of BIM to control and visualize the large-scale BIM data effectively. The specific aims are threefold: first to propose the profitable data model for a web-based platform, second to apply the IFC data effectively applying the characteristic of BIM, and third to propose the model which is profitable to web-based data visualization among the various usages of facility.

2 Background literature

Most researchers have been studying data processing of large-scale data especially in the game field handling 3D animation and the laser scanning field such as making medical instruments and reconstruction of cultural heritage. Even they have a huge data capacity, the main memory of hardware is not controlled easily. The profitable way to control these diverse situations is required to operate 3D data effectually on each user's operating system (OS) and each field. To overcome these limits, there has been studies on mesh simplification (compression, division), external memory reference, etc.

Han [1] proposed a method that 3D point cloud stored on a hard disk drive directly instead of using DB or storing in the main memory for building OcTree. For this, she compared and analysed a memory-referring method and a file-referring method. Na and Hong [2] defined spatial data LOD from LOD1 to LOD4 for combining construction and spatial information, and proposed the weight lighting algorithm of a large 3D object based on LOD for effective visualization. Fan and Meng [3] proposed an approach of simplifying 3D model building by CityCML with 3-step for deriving LOD2 from LOD3 models which were building envelopes without openings. Glander and Dollner [4] generalized virtual 3D city models by units for good representation with 3D block cell technique which were formed by the landmark buildings with geometry data, infrastructure like roads, coast line with point, line and the area with polygon. Based on the clustering concept, Xu *et al.* [5] proposed an extraction algorithm which was specified to GPU-based rendering and profited other GPU platforms. They also proposed a parallel frame interpolation algorithm based on GPU for large-scale structural dynamic analyses. Cignoni *et al.* [6] proposed a method of huge mesh management on a limited core memory footprint adopting an external memory technique for simplification. Okamoto *et al.* [7] adopted a cloud computing technology as a solution for large data processing without a processing cost to the client user. For balance of effective load between server and clients using Grid-Lumigraph, they applied both model-based rendering and image-based rendering.

As the above research shows, the ways to decrease data capacity have a varied approach. These were focused on visualization of general 3D node data or mesh data to reflect the system environment like smart device, desktop. A few researches about light weighting of BIM data likewise proposed related technology based on the system characteristic.

Since BIM data contains a huge amount of data from a whole lifecycle of building, the light weighting of data capacity for visualization must be preceded before system operation. This research has focused on IFC data which is the standard data format of BIM, more than the system environment of platform. So we analyse the IFC modelling technique based on BIM and propose the light weight algorithm expanding it.

3 Algorithm of data light weighting

This research is from the project which has been developing 'BIM on GIS platform for interoperability', one of the main projects of Korea Institute of Civil Engineering and Building Technology (KICT). It is mainly converting from BIM geometry data of facility to Interoperable Spatial Model (ISM) that is the inner format for visualizing on a screen of platform, but also building a database for presenting BIM property data. This chapter would introduce the 'BIM on GIS platform' and primary issues from large-scale BIM data, finally proposing the adoption of the 'Block-Reference' technique for large-scale data processing.

3.1 Overview of BIM on GIS platform with ISM format

A rise in the necessity of high-technology such as an integrated management system of BIM based construction information, means that an integrated operation of procurement of construction spatial information is required. It is based on the interoperability between BIM and GIS in the early stage for building future software for construction technology and responding to the world market. This research [8] is part of the project "Development of Convergence Technology on Construction Information & Spatial Information based on BIM/GIS Interoperability". The open platform has been developing that is able to make a connection between BIM, dealing with the indoor space of a single building, and GIS dealing with city-urban space. It is a complicated and intelligent smart city converging architectural data in a construction process to GIS from a building to earth. WorldWindJAVA (WWJ) – an open source from NASA – is a base engine and the project is trying to develop the open platform and related technologies based on the requirement qualification. BIM data is usually designed using Revit architecture and ArchiCAD. The main platform server is benchmarking the several functions of the BIM server. For the part of GIS, GIS server and PostGIS server are working with this platform.

This year the 4th step of the whole 5 phases is to upload the BIM modelling data on GIS. Property data and geometry data are separated from each other for procuring interoperability between IFC, which is standard of BIM, and CityGML, which is standard of GIS. The facility is then modelled in BIM data and the database of property data is constructed separately for uploading onto the GIS platform. Finally the geometry data is extracted from the IFC file, it is then converted to Interoperable Spatial Model (ISM). ISM is a data format of the spatial data model connecting BIM-GIS that is possible to present: 1) the outdoor object of CityGML, 2) concept of Level of Detail (LOD), and 3) the indoor building object of IFC. Figure 1 is the image for checking an indoor data uploaded on BIM on the GIS platform after converting the IFC sample data to ISM.

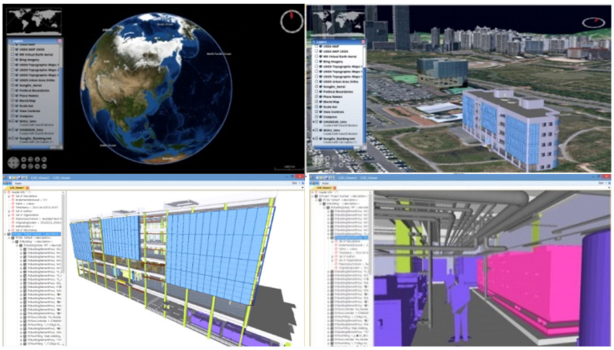

Figure 1: BIM on GIS platform with IFC sample data.

3.2 The main issue about large-scale BIM data

The BIM on GIS open platform is supposed to provide and manage a service of large-scale data for the city area and offer the integrated environment which can handle effective services for each different type of data like BIM, GIS. This research has been developing ISM for interoperability of BIM-GIS as a solution which integrates BIM and GIS data efficaciously, even each disparate character, and performs the query effectively, searching and showing on a screen. Figure 2 is the concept image for uploading the process IFC file to the platform. First of all, property data and geometry data of IFC, generating all data through a life-cycle of facility, were operated dividedly. For this, the platform inner data format itself (ISM) was developed and made geometry data of IFC transfer in DB. In the process of converting IFC to ISM, an increase in data capacity was confirmed.

In the case of IFC, when working 3D facility modelling, geometry data is basically saved as parametric data of b-rep. However in the case of ISM, geometry data of IFC is converted and saved with surface model based on mesh. It made the compatibility between different formats high, but because of the capacity change, the loading time for rendering process of facility was longer and the speed was increasing. These results are just about uploading a single facility, so the measure has to be considered about what can be handled for the various capacities of facilities on web-based BIM on GIS platform.

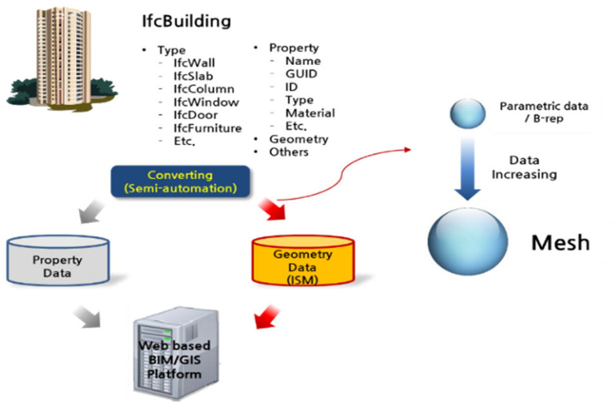

Figure 2: Process of IFC file uploading.

3.3 Introducing the background of block-reference concept

As mentioned above, there are diverse techniques about light weighting of large-scale 3D data. A huge 3D mesh data has been controlled easier than before, but still general computers have some problems in loading and processing data owing to the limitation of the main memory. Massive data processing should be related to light weighting and effective visualization through it for users. A simplifying algorithm based on LOD, an external reference technique, etc. are suitable for the part requiring visualization of 3D data concretely.

‘Block-Reference’ concept is one main function which is used a lot in AutoCAD, and it makes objects or object groups defined as ‘Block’ for applying the design plan efficiently. BIM data modelling usually produces and saves a building or included objects as a ‘library’ before, then when the user needs to request them, they are recalled easily from the saved library. This research makes the best use of object-based BIM modelling characteristics. Due to the building data designed artificially, it is devised as most of the objects are, designed by type, not each unique geometry data except for freeform architecture. Also this works for the light weighting process which maintains the original data with the advantage of foundation technique rather than simple light weighting techniques. Thus the Block-Reference concept was chosen as the main key for light weighting, and the algorithm was developed along with the background for research.

4 Adoption of block-reference and design of the algorithm

4.1 Algorithm design: light weighting for large-scale BIM data

In the case of BIM data, all objects generally do not have unique geometry data because of ‘copy’ and ‘paste’ like AutoCAD. As the objects with the same shapes except for just the one geometry data and they are visualized on actual location using reference data, the size of geometry data can be much reduced within the representative data of 3D modelling object. In that context, the first concept of light weight algorithm is that one geometry data representing the same shape of several objects is defined ‘Block’ and the relating data is defined ‘Reference’.

When the large-scale geometry data is produced, Block-Reference technique makes the Block which is the first type of same shape geometry data at first. Then based on this, it produces the Reference data – how each object uses what kind of geometry data. The main mechanism of this technique is that all BIM data is called for converting to ISM format and the data about copy objects not Block is visualized from Reference DB. Figure 3 shows the concept applying the Block-Reference technique to ISM between before (ISM v1.0) and after (ISM v1.1).

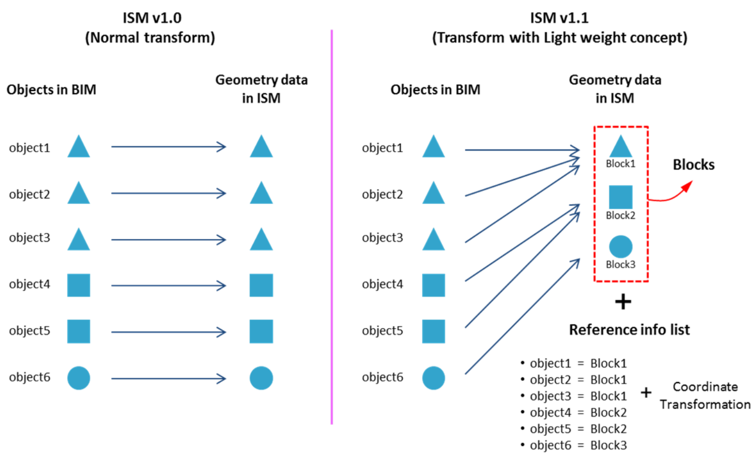

Figure 3: Comparison of ISM v1.1 with v1.0 which contains the light weight concept based on Block-Reference.

The data package of a single building is composed of Header including three factors: 1) general building information, 2) Block which is the representative type based on same geometry data, and 3) Reference which is the data list for every object referring Block. In the case of ISM v1.0 not applying the light weight technique, since all geometry data from IFC was implemented at first, the data loading speed and the data capacity were very inefficient. On the other hand, ISM v1.1 applied the above technique and considered the characteristics of facility modeling, then most part of the data capacity could be reduced.

4.2 Algorithm design: OcTree methodology for spatial indexing and LOD adoption

The second concept is to apply LOD-based OcTree, one of the spatial indexing techniques, to the light weight algorithm for data visualizing of Block-Reference. Spatial indexing is one of the important factors to show 3D data effectively and naturally, so OcTree technique is profit to present 3D modelling data. Kim [9] proposed the spatial indexing technique based on OcTree for efficient data processing of large-scale BIM data. The research verified the frame speed through a view point path of user after designing the IFC schema-based algorithm and applying it to IFC sample data on the BIM on GIS platform. In the case of applying the algorithm, the number of frames were more than the case of not applying it from 3 frames per second to 14 frames per second, it also proposed the most effective screen to the user with visualizing more data in the same time.

When working with spatial indexing, according to a size of building, the volume of first space and the depth of OcTree level can be decided variously. This research defines that the first space surrounding a building is set to a regular hexahedron and the depth is set to 3rd level consistently. The reason for using this simple structure is what can be very easy to load and implement indoor objects for visualizing indoor data of building. Furthermore the processing speed is not very reduced when the test with the real spatial data model are conducted based on client of BIM on GIS platform. Figure 4 shows a result by location of view point for spatial indexing of the sample building data. The size of the first regular hexahedron chosen for spatial indexing was presented larger than the size of the building. So there were many empty spaces that did not exist in the real data after spatial indexing. The main reason for this is for the efficiency of the query. It would not only search which nodes were in the Viewing Frustum area, but also the visualizing speed of indoor data for those nodes were not reduced too much.

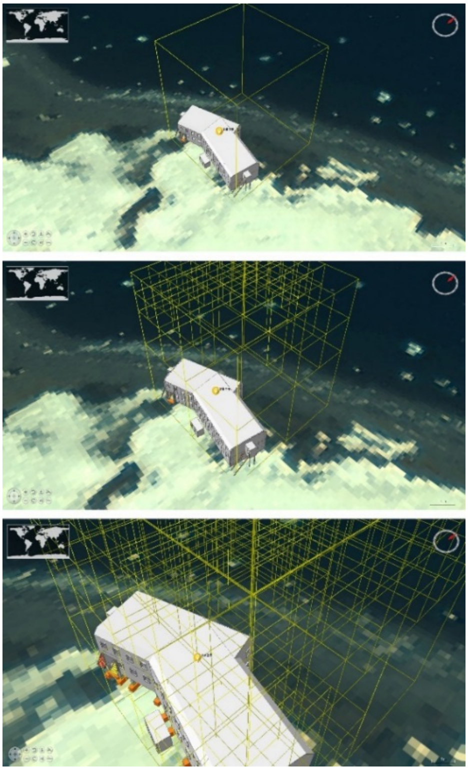

Figure 4: Example of BIM data spatial indexing on BIM on GIS platform.

As an extension to previous research, this study applied LOD including external and internal of building to ISM v1.1. For this, Skin reference list, saving objects based on LOD for outdoor, and Node reference, list saving objects based on LOD for indoor, were generated and objects by LOD were grouped. As a result, objects including the lowest child nodes of OcTree processed spatial indexing, but the direction of parent nodes was not decided. In order to use the parent nodes, the merging of objects and policy for Point Reduction and relating rule should be determined.

4.3 Algorithm structure

This section is about the design of ISM v1.1 to improve the performance of ISM v1.0 considering an application of BIM on GIS platform. On the basis of the main concept of algorithm mentioned in section 4.1, the data package structure is reflected in the advancement of representation technology based Adaptive LOD and the light weight/structuration technology. Block-Reference is classified according to Skin, Bone, and spatial indexing objects. These objects are classified in each file after being gathered by the lowest child nodes of OcTree. Figure 5 shows the data package structure which has the whole LOD from 0 to M (total M+1), each indoor building LOD from 0 to N (total N+1) and outdoor building LOD from N+1 to M (total M–N). Table 1 shows the detailed description of the ISM data package.

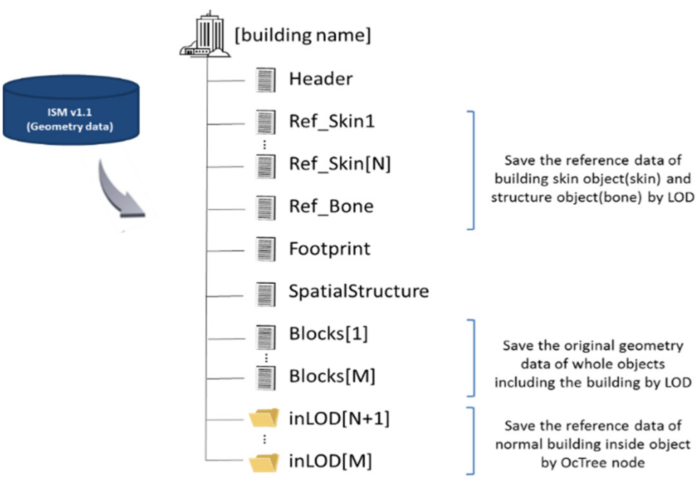

Figure 5: Data package structure of ISM v1.1 format.

For adapting the Adaptive LOD, ISM v1.1 generates Block list to match up with all LOD levels one to one and Reference list for Block-Reference data of objects included in each level. Also when the data for indoor LOD level is generated, this light weight algorithm makes the Block list on the condition that the Reference list assorts by each OcTree node of spatial indexing.

Table 1: Description of ISM data structure.

No.	Component	Description
1	Header	General data includes as follow: - Project, version and file name of ISM based on 3D BIM model - The number of 3D objects, LOD levels, OcTree levels, etc. including in ISM - A factor of location, rotation and scale for mapping relative coordinates to 3D absolute coordinates of ISM
2	Ref_Skin1 ~ Ref_Skin[N]	Block-Reference data file of object which is located in building skin by LOD level (LOD0~[N])
3	Ref_Bone	Block-Reference data file of object which should be shown, regardless of spatial indexing when rendering the indoor building
4	FootPrint	Expanded foot print data file of building/Used to find whether camera is inside or outside
5	SpatialStructure	Saved spatial structure data in IFC file
6	Blocks[1] ~ Blocks[M]	Original geometry data file of object in building by LOD level
7	inLOD[N+1] ~ inLOD[M]	Reference data of basic object included building inside by OcTree nodes/Each node is composed with groups by LOD

Modelling data of LOD0 is shown as a simple rectangular which can express with a bounding box. The header contains its related basic information, so LOD0 is not produced individually. A folder that includes LOD data for building inside the data package structure is made up files. Each file saves a present condition of the Block-Reference of the object including the lowest child nodes. In this file, there are a unique index of object, Block index referring to the object and Transform matric of the object. The next number of Ref_NodeData refers to a particular child node. The cipher is same with the number of OcTree levels except at Root level, and cipher can hold from 0 to 7. For instance, Ref_NodeData342 is situated at the objects of 3rd child node or 5th child of 4th child or Root node as Figure 6.

4.4 Method of finding Block-Reference object

How to find Block data and Reference data from BIM with IFC format using the 3 step process sequentially. The first step is to analyse and confirm the object's IFC type. This might be the fastest way to filter the same objects. Only when modelling BIM data, the correct type input is a precondition. The second step is to analyse the number of points organizing the objects. Thereby comparing the

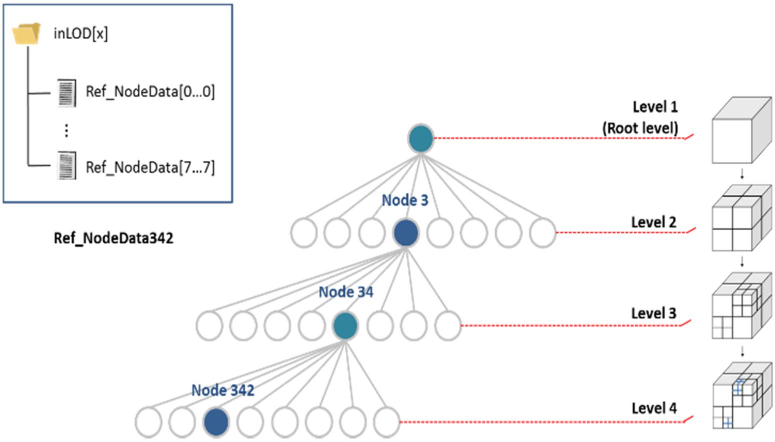

Figure 6: Data structure of inLOD and NodeData 342 example.

number of points of filtered objects with the same type, the same objects can be searched more accurately. Last and third step is to calculate the distance from first point to each point and compare the total distance after figuring up. Through this step, the same object group is gathered and one object is selected for Block. In the last step, the coordinate transformation between comparison objects is calculated and contained within the Reference data, and finally the system saves the Block-Reference data.

BIM data of specific building applying the Block-Reference concept is shown in Figure 7. The green objects are researched with Reference, and the others are Blocks. As most objects are classified as Reference, the more one building has objects represented as Reference, the more a reduction effect of capacity increases.

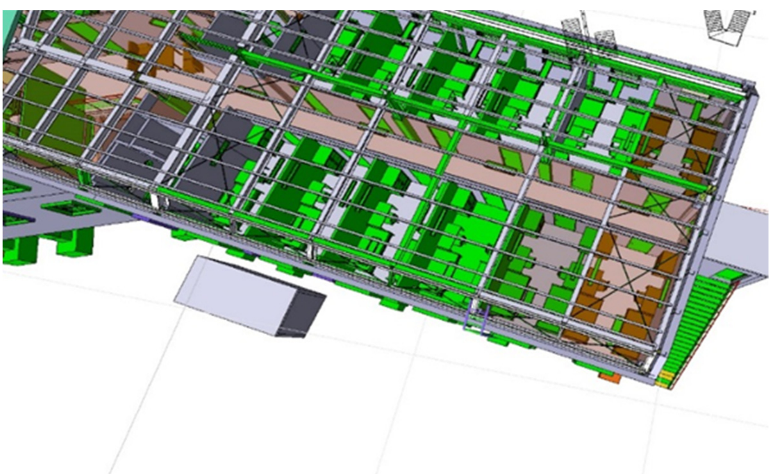

Figure 7: Representation of Block and Reference objects of BIM data.

5 Adoption of block-reference and design algorithm

To verify the Block-Reference algorithm, Chapter 5 had a test with three IFC sample data which composed of different size and building usages. The IFC model with ISM v.1.1 was compared with the model with ISM v1.0, not applying the Block-Reference concept. After each sample file was converted to ISM, the test deduced results about the capacity of data and the loading speed of files.

There are two main concepts of ISM: one is that these sample data contain both geometry/property data, and the ISM contains geometry data for visualizing IFC and property data for partial spaces. While IFC data converts to ISM, the geometry factors for data visualization save as files. A part of space data and property data offer service with database. The other main concept of ISM is the integrated visualization of large-scale GIS data and BIM data. Thus visualization of geometry data is one of the main issues in tests.

Table 2 shows the results of data capacity among original IFC, ISM v1.0 and ISM v1.1. Totally the average reduction ratio was 89.3% and as building usages, the ratio was high in the order of A-C-B. This was affected by the number of Block and Reference according to the building design based on building usages. The formalized buildings like office (A), residence (C) have a few Blocks and lots of Reference, so the data capacity, which is composed of geometry data, could be reduced dramatically. On the other hand, others like unusual office (B), cultural asset, museum, freeform building have a quantity of Blocks as design of buildings was getting complicated, and the reduction ratio could be increased relatively.

Table 2: The test results of data capacity.

No.	Data Capacity (Mb)			Reduction Ratio (1 – ISM v1.1/ISM v1.0) * 100
	Original IFC	ISM v1.0	ISM v1.1	
A	66.29	176.05	15.3	91.3%
B	28.18	34.66	4.15	88.0%
C	65.33	74.89	8.41	88.8%

Table 3 shows the test results of the data loading speed between ISM v1.0 and ISM v1.1. The average reduction ratio was 81.1%, but when applying the light weight algorithm, the process for finding Reference objects, searching related Block and loading again consumed a little time. The data loading of ISM v1.0 was a series of process that IFC data was extracted and converted to ISM, then it was visualized to screen through a graphic card. Therefore ISM v1.0 made the rendering speed very slow and occupied a lot of space on the graphic card. In contrast, the loading speed of ISM v1.1 was similar with ISM v1.0 because of translation mechanism. But the amount of graphic card usage was decreased remarkably, so it could be a profitable environment to thin clients using web-based BIM on GIS platform.

Table 3: The test results of data loading speed.

No.	Data Loading Speed (sec)		Reduction Ratio (1 – ISM v1.1/ISM v1.0) * 100
	ISM v1.0	ISM v1.1	
A	4.889	0.64	86.9%
B	1.341	0.405	69.8%
C	3.651	0.483	86.8%

Ultimately, ISM v1.1 has a better condition for data capacity than ISM v1.0. In the future, the light weight algorithm should be extended to a detailed algorithm considering the characteristics of building usages and procure more sample data by building usages for verification. Also for effective system operation, specific LOD adoption has to be researched.

6 Conclusion

This research proposed the light weight algorithm based on IFC to visualize geometry data of large-scale BIM on BIM on GIS platform. While inputting IFC data, BIM on GIS platform which has been developing at KICT stores DB of geometry data and property data separately. The light weighting was the essential requirement to increasing the data capacity when geometry data converted to ISM. In order to resolve it, this paper analysed the diverse preceding studies related to light weight, and as the best way to reflect the research environment, Block-Reference technique was adopted.

First the building objects were grouped by types and the algorithm was designed that the type with the same geometry was defined to Block, the other copied objects were referencing the Block. For verifying it, based on three IFC sample data, there were a comparative analysis of ISM v1.1 which applied the Block-Reference algorithm and ISM v1.0 not applied. The results were the reduction of data capacity by 89.3%, the reduction of loading speed by 81.1%.

In the future, the research plans to develop the algorithm continuously with verification of added concepts by building usages and design types. Through the light weight algorithm, as well as looking forward to show large-scale BIM data on BIM on GIS platform, we exploit diverse service parts by user's aspect.

Acknowledgement

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 Civil engineering and Building Technology.

References

[1] Han, S. H., Implementation of file-referring Octree for huge 3D point cloud, Korean Society of Surveying, Geodesy, Photogrammetry and Cartography, 32(2), pp. 109-115, 2014.
[2] Na, J. Y. & Hong, C. H., A study on the weight lightening algorithm of 3-Dimensional large object based on spatial data LOD, Korea Spatial Information Society, 21(6), pp. 1-9, 2013.
[3] Fan, H. & Meng, L., A three-step approach of simplifying 3D buildings modeled by CityGML, International Journal of Geographical Information Science, 26(6), pp. 1091-1107, 2012.
[4] Glander, T. & Dollner, J., Abstract representations for interactive visualization of virtual 3D city models, Computers, Environment and Urban Systems, 33(5), pp. 375-387, 2009.
[5] Xu, Z., Lu, X., Guan, H. & Ren, A., High-speed visualization of time-varying data in large-scale structural dynamic analyses with a GPU, Automation in construction, 42, pp. 90-99, 2014.
[6] Cignoni. P., Montani. C., Rocchini, C. & Scopigno, R., External memory management and simplification of huge meshes, IEEE Transactions on Visualization and Computer Graphics, 9(4), pp. 525-537, 2003.
[7] Okamoto, Y., Oishi. T. & Ikeuchi. K., Image-based network rendering of large meshes for cloud computing, International Journal of Computer Vision, 94(1), pp. 12-22, 2011.
[8] Korea Institute of Civil engineering and Building Technology, Development of BIM/GIS Interoperability Open-Platform (3rd year), project report, 2014.
[9] Kim, J. E., Development of the spatial indexing method for the effective visualization of BIM data based on GIS, Korea Academia-Industrial cooperation Society, 15(8), pp. 5333-5341, 2014.