# Introduction:

The increasing demand on facility management, inspection and maintenance operations has caused an urgent need to integrate building information models and geographic information systems models. Consider the following real world example: someone made a request for maintenance to look at the temperature in “core chem Physics office A249”, since it was too cold (14 degrees). The operators need to use a campus building operation system to locate the particular building and room in the work request. Then they check the heating and cooling systems that serve that particular area or room. In addition to the location of the equipment (such as the heating system) in the building, the maintenance personnel may need information about the manufacturer, serial number, maintenance history information, service manual, and spare part information about the specific equipment that needs to be maintained or repaired or replaced. In order to create such a comprehensive campus operation system to retrieve both building information and explore the relationships among buildings in a campus area, we need the system to contain both the geospatial campus data and maximum level of detail of real buildings. Currently, we do not have any single model or system which carries both of them; they exist only in separate models (e.g., GIS and BIM respectively).

We tried to build our operation system by integrating the detailed semantic building information into the geospatial information of a specified area, such as the UBC campus, so that the operation system can trace the 3D geospatial network of the campus to locate the specified room where the equipment is stored or installed in the building in order to assist the operation group to perform the maintenance task.

There are many different building information models and 3D geospatial information models representing real world surrounding objects from both geometry and semantics perspectives. “Industry Foundation Classes (IFC) and City Geography Markup Language (CityGML) today are considered as two of the most prominent semantic models for the representation of design and real world objects, respectively. [10]” IFC [5] has been introduced as a standard for describing building components and construction data. CityGML [7] is a common information model and XML-based encoding for the representation, storage, and exchange of virtual 3D city and landscape models. Integration of IFC and CityGML is seen as a necessary step for getting a complete 3D city modeling with detailed building information. The reason is that this integration would assist us in handling the operation and maintenance requests by exploring the 3D campus view to locate the desired buildings (using CityGML) and zoom into the building to display the layout of every floor so that we can easy identify the room that store the equipment that needs to be checked and repaired (using IFC).

Generally speaking, all the previous integration approaches are the implementation of semantic and geometrical integration of IFC and CityGML models. Mostly, they [12][14][15] are focused on unidirectional conversion from IFC to CityGML, because CityGML is more popular in the decision-making as it has broader view. They define some standard mapping schemas between IFC and CityGML based on the semantic agreement of their interpretation of the same objects. There are conversion tools [15] which extend CityGML with rich semantic information of IFC and ADE. In addition, commercial software products for conversion from IFC to CityGML such as IfcExplorer [18] and Safe software [19] make a great effort to the development of 3D city modeling integration.

Our goal is to build an operation and maintenance system to allow users explore and retrieve a wide range of campus data, starting from campus scale and zooming into a room in a building. To allow this integration, we need to convert data from IFC to CityGML. CityGML is a 3D City model which represents geospatial information of our real world, and it can also describe the building components. On the other hand, IFC only has the detailed building information, excluding the outer surrounding environment. After taking consideration of the previous conversion approaches [12][14][15][16] and commercial software products [18][19][21], we want to build our integration system on top of the existing work. We select two most suitable frameworks as a start point of our conversion from IFC to CityGML, BIMServer [21] and Feature Manipulation Engine (FME) [23].

After comparing and analyzing the converted results from BIMServer and FME, we decided to extend the CityGML schema to accommodate the rich semantic building information that the existing conversion tools cannot cover. After running several experiments, we concluded that it is not currently possible to bring all the details for buildings such as the mechanical components and facility information into CityGML. We encountered a number of problems in our attempt: (1) the standard CityGML schema cannot currently support such kinds of information , and (2) file size increases dramatically when small amounts of additional information is added in. The file size is far beyond the maximum capability that the current conversion applications can handle.

In order to solve the challenges, before we start the conversion process, we need to narrow the scale of our project and define our problem by giving some concrete scenarios to make the integration feasible. Meanwhile, we will efficiently utilize the information we need most from both models and integrate them together and keep looking for other possible platforms on which we will develop the 3D city modeling integration.

In the remainder of this paper, section 2 explains the terminologies and concepts used in our project, such as the IFC and CityGMLmodels, and section 3 introduces the previous and ongoing research work and commercial products of conversion frameworks and tools. In section 4, we describe our strategies of trying to integrate rich semantic building information into geospatial outer environment and list the challenges that we are facing and our alternative strategies.

# Background:

As we discussed above, to fully handle maintenance and operation requests, we need to access a complete picture of 3D city modeling at high levels of detail. In this section we review the source of much low-level details Building Information Model (BIM) [2] in Section 2.1 and then discuss the higher-level Geographic information system (GIS) [4] in Section 2.2.

## BIM and IFC

“Building Information Modeling (BIM) is a digital representation of physical and functional characteristics of buildings.”[3] A BIM describes buildings with respect to their geometric and semantic properties. Generally, it is generated at the early stage of the building’s lifecycle to facilitate the architects, civil engineers and stakeholder in planning, designing and constructing. It has the ability to organize huge volumes of data related to buildings, the semantic information of building parts and spatial relationships between them, and also supports sophisticated 3D visualization and manipulation. Unlike CAD models, which represent buildings as a collection of points and lines, the semantic information that BIMs carry makes great contributions to the data analysis and decision making regard to buildings. It is an object oriented building modeling, which defines building components as elements and their properties and the relationships between them.

One of the most developed and established semantic models that implement BIM concepts is the Industry Foundation Classes (IFC) [5]. As an open standard schema, IFC is popularly used to exchange and share BIM data between different applications. Its standard schema comprises information contributing to a building’s whole lifecycle: from conception, through design, construction and operation to maintenance and destruction [6]. IFC schema is a conceptual model for buildings which represents building structural components and their relationships in a semantic way as shown in the below Unified Modeling Language diagram in Figure 1. It describes the components of spatial objects as classes and different arrows means different relationships and association between classes. IFC has an IfcBuilding class which consists of one or more IFCBuildingStorey. In each IFCBuildingStorey, there are several IfcSpaces instead of rooms. The building elements are walls, roofs, beams, columns, stairs, also contains openings, such as doors, windows.

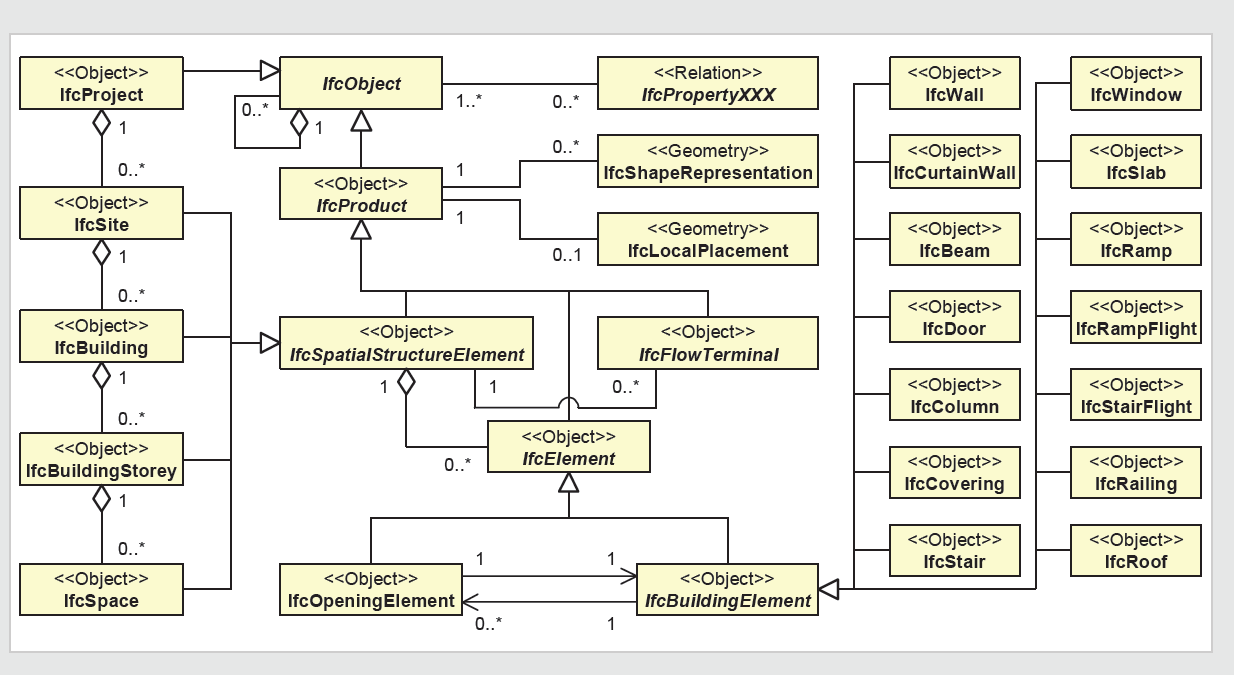
**

Figure 1: A UML notation of IFC building model.

## GIS and CityGML

Geographic information system (GIS) [4] is designed to represent, manipulate and analyze all kinds of geographical data, thus it plays an important role to represent the surrounding world and to model spatial objects in urban and regional areas. It is widely used to model different focused areas around and between buildings such as the lands, streets as well as urban furniture. GIS applications are well developed for analysis, user interactive queries and presenting operation results of spatial information, furthermore, they are ideal for projects in a campus or multi-site environment. Therefore, it makes a great contribution to the city planning, facility operation and maintenance. In order to efficiently represent geographic information of large number of spatial objects in vast regional and urban areas, it does not focus much on the details of buildings.

CityGML [7] is defined as a semantic information model, and it is used as a standard representation to store and exchange virtual 3D objects and city models among different applications. It defines in five different levels of detail (LoD), which are used to represent the same city model objects in different degrees of detail, where objects become more detailed with the increasing LoD regarding both geometry and thematic differentiation:

* LoD0 represents a two and half dimensional digital model of urban surface in which buildings are represented as footprints. As shown in the Figure 2, it is a topographic model of bare-earth terrain which displays the elevations of natural terrain features such as barren ridge tops and river valleys [28]. However, elevations of the buildings and roads are digitally removed, so buildings are represented as 2D rectangle shapes as you can see from top down.
* LoD1 is the first representation of buildings as 3D objects. All buildings look like blocks with the same flat roofs.
* In LoD2 and LoD3, a building is represented in an architectural model with details of roofs, walls, some exterior elements, such as balconies, and opening in the boundary surfaces such as doors or windows. Lod3 applies more details and facade textures to the roofs and walls than LoD2, and it covers detailed vegetation and transportation objects which cannot be found in LoD2. [9]
* The most detailed level, LoD4 [9] adds detailed interior elements to buildings, such as rooms, interior doors, interior wall surfaces, stairs, furniture, electricity units.

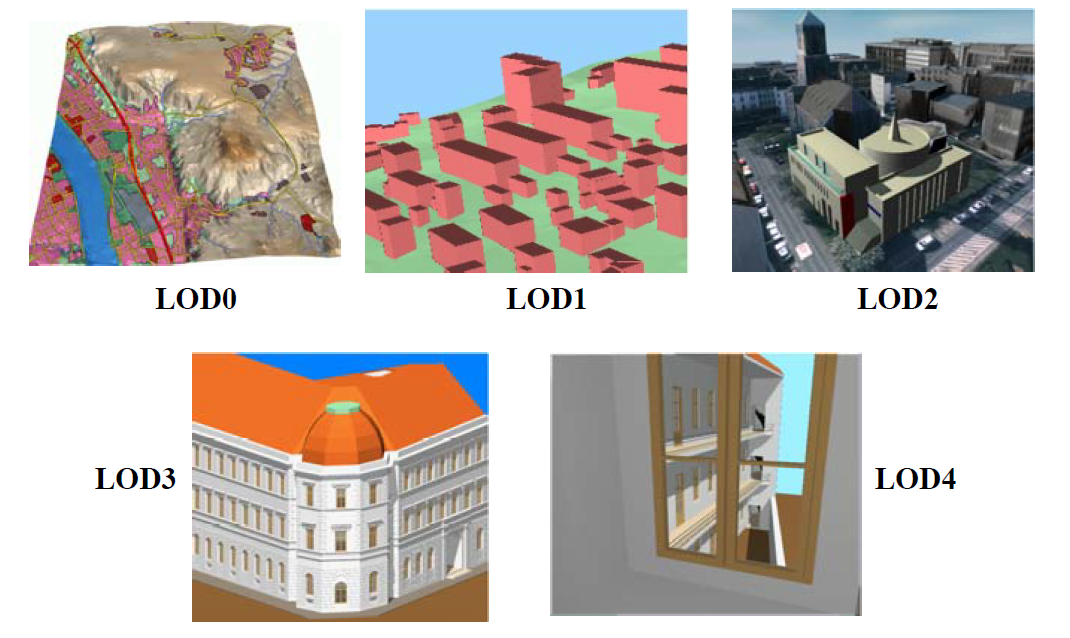


Figure 2: The five levels of detail (LOD) of CityGML (Source: IKG Uni Bonn [9]).

Thanks to these different levels of detail, CityGML is highly scalable, so the building details can be flexible adjusted to meet projects’ special needs in order to make efficient and sophisticated analysis. Comparing to IFC, in Figure 3 CityGML does not have storey information, and it considers rooms as building components rather than spaces in IFC. CityGML represents the geographic information of spatial objects, thus the measurement properties of each building component are represented as geographic coordinates instead of lengths and widths.

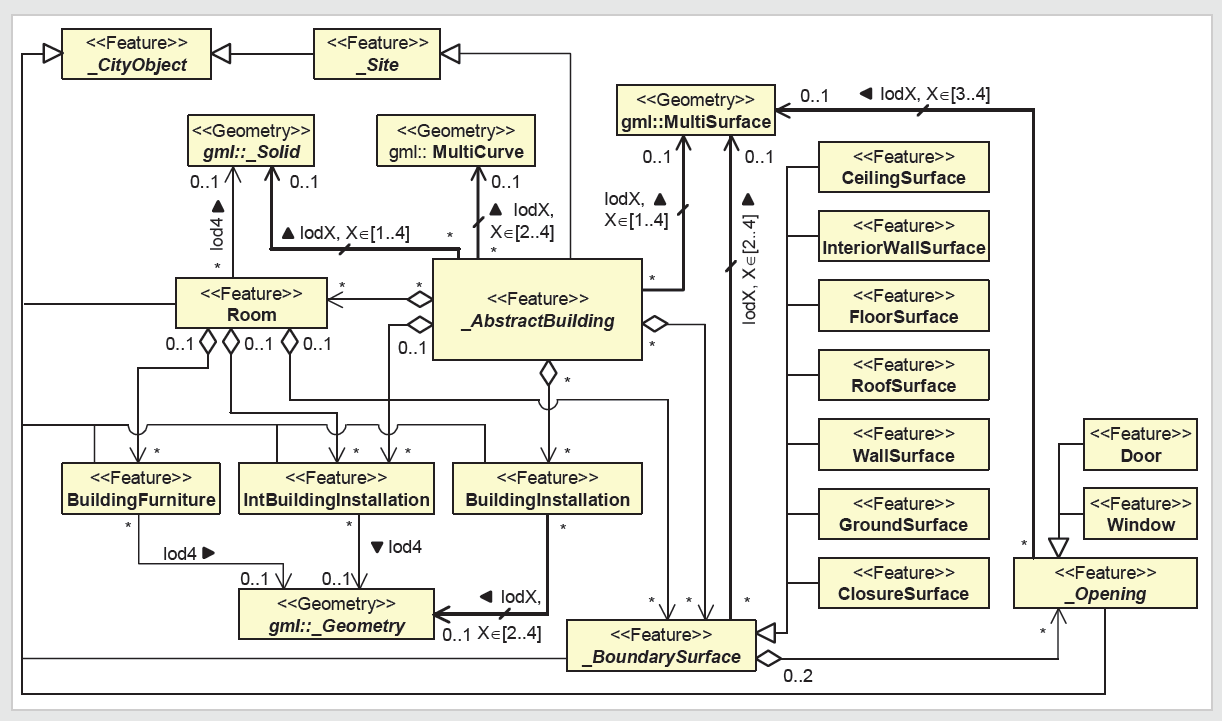


Figure 3: A UML notation of CityGML Model.

Generally, [8] IFC, an element-based volume model, uses constructive solid geometry with volumetric, parametric primitives representing the structural components of buildings. However, a 3D GIS, a surface model, uses boundary representation which is the accumulation of observable surfaces of topographic features.

Right now, FME Data Inspector supports the visualization of the IFC and CityGML files. Solibri Model Checker provides sophisticated functions to analyze BIM in IFC format. Also, Galdos GML Inspector and LandXplore can view the CityGML data.

# Existing Conversion Approaches

In order to process the maintenance requests such as the previous example’s uncomfortable room temperature in “core chem Physics office A249”, we need to show not only the map of the entire campus, which indicates the paths to locate that building, but also the details of each floor in the specified building to access the room where the heating system is located. Building the GIS model of the whole campus is considered as a good starting point and then increasing includes the detailed building information from BIMs. In sum, for all the operation requests, the most important thing we should do is to define the scale and region that we will operate at. We believe the best solution will be the integration of BIM and GIS to present the behavior of the entire campus and also the details of each individual building.

“Researchers all agree that the best approach for the integration of BIM and GIS is harmonized semantics” [22], which creates formal mappings between the detailed inner building models and outdoor real world GIS. Both IFC and CityGML use different terminologies to describe the same domain. Semantic interoperability ensures that both IFC and CityGML share the same meaning for a defined spatial object. All the previous approaches are based on the idea of harmonized semantics; some of them focus on a unidirectional method (mostly from IFC to CityGML) for the conversion process. Some of this attempt is to develop an automatic framework to transform IFC building models to CityGMLwith regard to each level of details, and other is to convert building information to GIS from CAD instead of BIM. Bidirectional conversion recently draws our attention to fully integrate IFC and CityGML in terms of a conceptual model.

Some of the most significant existing approach of integration of IFC and CityGML are listed below:

**IFG Project**

The framework of the IFC for GIS (IFG) project (in 2003) [11] is designed to exchange building information between CAD systems and GIS using IFC, so that it can support the analysis of the relationships with a building areas or other buildings by accessing to both the building details and their surrounding environment. The project succeeded in creating a mapping specification from IFC geometry to GIS and vice versa by identifying the entities in IFC schema that can support GIS application and creating mappings between them and the real GIS entities.

**Nagel’s framework of conversion of IFC to CityGML**

A framework was proposed by Nagel (2007) [12] which aimed at developing an algorithms that automatically transforms IFC building models into CityGML models through a series of steps that create the separate footprints of each storey within their own boundary surfaces and finally merge them together. This research focused on only level of detail LOD1 and LOD2 of CityGML and the purpose of the algorithms is to create a geometrically and semantically valid representation of LOD1 which can also be applied to LOD2.

**Isikdag and Zlatanova’s advanced framework**

Isikdag and Zlatanova (2009b) [14] have extended Nagel’s framework by proposing a framework for automatic generation of building semantics and components in CityGML from their BIM representation. Because CityGML and IFC are designed for two different domains, they have very diverse objects and classes and cannot be directly and easily mapped to each other. Therefore, they generate semantic and geometric mappings for each LoD of CityGML separately since the same object in one schema can be mapped to different objects in the other for different levels of details. In order to simplify and facilitate the conversion process, for each of level of detail, all the objects and attributes in both schemas which need to be transformed are first defined and same as the mapping rules.

**A 3D Conversion Framework by Thomas Kolbe**

With the aim of creating a holistic view of 3D city, a team led by Thomas Kolbe at the Technical University of Berlin [13] proposed a framework that incorporates the semantic spatial context data into 3D graphics/data of buildings and urban areas stored in formats such as X3D, DXF, KML, and COLLADA. CityGML is selected as an intermediate layer for the conversion process to IFC is because that it is a very rich semantic model and it offers similar notion of building semantics as IFC. Therefore, it is considered as a bridge to link 3D visualised models to the sophisticated and detailed building models. The simple summary of this conversion framework is shown below:

Geometric/graphics Models 🡪 Semantic City Model 🡪 Building Information Model.

Their corresponding formats are:

X3D, DXF, KML, COLLADA 🡪 CityGML 🡪 IFC

**The development of the GeoBIM -- CityGML extension**

Léon (2009) [15] demonstrated the latest application domain extension (ADE) which converts building information model (BIM) in the format of the open standard IFC into CityGML (van Berlo, 2009). CityGML represents building information in high level of details, hence, in order to represent rich semantic information of IFC into CityGML, they extend CityGML schemas with extra objects and properties. However, there are only few of IFC classes that can be transformed to CityGML extension and have real meanings in it, because IFC and CityGML describe buildings in different representations and aspects.

**Unified Building Model**

The Unified Building Model (UBM) [16][17] is the first full integration framework of IFC and CityGML that IFC can be traced to CityGML and vice versa. The reference ontology in this study is defined as an expressive ontology for IFC and CityGML semantic models, which is a superset model that is extended to contain all the features and objects from both IFC and CityGML building models respect to all levels of details including inner and outer spatial structures. The integration approach is performed in two steps: first a building model can be converted from the source model to UBM, and second from UBM to the target building model. UBM is considered as a standard schema, and mappings from both data sources to the standard schema are generated. In terms of this standard schema, it is easy to be extended if there is a demand of transformation from a new schema to IFC or CityGML.

**Commercial software products**

Safe Software Company [19] developed several applications which support the management, exchange and visualization of spatial and non-spatial data in more than 255 different formats. The feature manipulation engine (FME) is their core data transformation product which supports the translation, transformation and integration of spatial data. Right now, it supports the direct conversion of IFC to CityGML and vice versa.

Another commercial software product for conversion from IFC to CityGML is IfcExplorer [18][20] which is an implementation of Nagel’s conversion algorithm by the Research Centre Karlsruhe, Institute for Applied Computer Science, Germany. IfcExplorer is designed to make an automatic conversion of an IFC model into CityGML regard to the selection of the specified LODs and building elements.

**Building Information Model server**

The Building Information Model server (BIMserver) software [21] is free and open source and is intended to centralize information from any building related project. It uses IFC as its core standard building model and stores building information in the format of IFC in an underlying database, so it is possible to query, merge and filter all the BIM models and generate IFC files on the fly. It also supports the exporting functionality in various formats including CityGML, therefore we consider it as a conversion tool from IFC to CityGML.

The increasing research work which contributes to the integration of IFC and CityGML is driven from the demand of exchanging and sharing spatial information between different applications, some of which are focused on buildings, others are about wide range of city areas. To summarize, the contributions of all the existing research on the integration of IFC and CityGML are: they all define semantic mappings between IFC and CityGML; [14][15][16] develop frameworks and algorithms to make an automatic bidirectional or unidirectional conversion in terms of geometric and semantic information; the integration is done with regard to each level of details of CityGML[12][14]. [15] also discusses on rich semantics of IFC and how it can be applied to more detailed CityGML models.

BIM and GIS are intended to serve for two different domains and scales, one is for inner building structural components, and the other is for outer surrounding real world, and they use different representation to describe the same spatial objects, thus they have very diverse objects, classes and properties. In order to make an accurate and efficient integration, most of the previous works only focus on the main structural components, such as walls, roofs, doors, windows and so on, of which IFC and CityGML have the same sematic and geometric representations. Even though, some entities can be semantically mapped from IFC to CityGML, it is still difficult to create the geometric matching relations between them. For example, for a beam which is across two rooms, in CityGML, it is represented as two thematic objects because it is observable from both rooms, but they are aggregated into one object in IFC. Because of the different geometric representations that IFC and CityGML use, sometimes researchers [13] apply an evaluation function on all the possible ambiguous conversions to come up an optimal one. Since the CityGML is capable to represent the detailed building information in at most LoD4, and it is extended to model noise, tunnel, bridge, hydro, utility network [27], it still cannot represent the mechanical part. In addition, during the conversion process, the properties and parameters attaching to the components need to be kept in the target schema, but CityGML is not capable to keep them. Under the above restriction, the complex schemas and components are beyond the scope of the current research, as well as the components with complicated geometric shape.

# Experiment and Result

After serious study of the existing approaches and potential strategies, we defined our approach as two steps: the first step is to evaluate and compare the currently available conversion tools and select the ones that are most suitable for our project, the operation and maintenance system. After we analyzed all the converted results from the first step, we tried to complement the results by adding the missing parts to ensure that our system has sufficient information to answer all sorts of queries on facility management, maintenance operation and inspection operation.

By looking through the previous and ongoing conversion approaches, we decide to use FME and BIMServer, because FME is a sophistical software product and BIMServer is an open source and it can achieve our purpose. Other frameworks or applications that I mentioned above are neither unavailable nor not appropriate for our project.

[23] “FME provides unrivalled format support for data translation and integration, and unlimited flexibility in data model transformation and distribution.” FME Workbench provides various transformers to carry out all kinds of features restructuring and data manipulating through format translation. It supports schemas and mappings edition. Transformation occurs as the data is passed from reader to writer through a series of these transformers. We applied the geometric transformers on the building elements that we want to convert to CityGML, such as walls, roofs, windows, doors, to set the levels of details in CityGML, their corresponding LoD names and define feature types (eg. walls, roofs…) corresponding to different LoDs, such as for LoD2, just walls and roofs, LoD3 adds doors and windows, LoD4 adds rooms. The conversion process is not easy and accurate, because it is not a simple 1-1 correspondence between each building element, for example, in order to convert IFC space information into CityGML room information, we need to identify its associated walls, ceiling and roof. It successfully creates the CityGML model in LoD3, but partly in LoD4, as viewed by some inspectors, we can see that the buildings have facade textures of the exterior walls and roofs, stairs and interior walls with doors and windows attached on them, but they do not have room information, furniture, electricity units and some complex geometric shaped building components

(eg. curtain walls).

[21] “BIMServer is an open source model server, which is the ideal tool to support dynamic design, engineering, construction and building maintenance process.” Its core model server interprets all the building models into their IFC format and stores them in a common database and exports them in various format, CityGML is one option. We upload our IFC files to the BIMServer, and it exports the files in CityGML format in LoD4 to us. By viewing them in the 3D model visualization tools, we find out the results do not conform to the standard CityGML schema, such as the floor surface is supposed to be part of the boundary surface in standard CityGML schema, but in the result, the floor surface is attached to the room feature. Also, the number of floor surfaces and rooms cannot correctly match to the original IFC schemas. Besides, it does not contain the stairs, furniture, electricity units and some complex geometric shaped building components as well.

Landxplorer [24] seems a very promising application which can support the visualization, manipulation and analysis of the geospatial environment and objects. It allows aggregating and visualizing heterogeneous data and models from GIS and CAD to build 3D information products. It also enables to create, maintain and distribute large and realistic 3D city models so that we can run spatial queries and analysis on it and it provides the visual answers. Therefore, if we can incorporate our building model into CityGML model, and load them to the existing campus geometric information platform, we can accomplish our goal of facility operation and maintenance.

We decide to use the IFC model of the CIRS building at UBC [25] as our experiment, which is recognized as a leader of sustainable buildings. It has completed documentations of CAD and BIM files. After exported to the IFC 2x2 format by AutoDesk Revit, the file size is too large to be uploaded to BIMServer, so we cut off the architecture of CIRS building into two floors with the mechanical part. After converted it to CityGML format by FME Workbench and BIMServer, overall, both CityGML files have part of the building components of Lod4, but not completed, and none of them can convert the mechanical part. I list the observation from the visualization of the resulted CityGML files:

1. Mechanical part is totally missing in the CityGML model.
2. Furniture part (chairs, desks) is also missing in the CityGML model.
3. Stairs and railings are missing in the CityGML Model.
4. The curtain exterior surface walls cannot be converted to CityGML components.
5. The number of doors does not match.
6. The property of rooms (only in CityGML) does not match to the space (only in IFC).
7. For CityGML model, the walls, doors, windows do have area measurement, just geometric points coordinate, but in IFC model, they are measured by width, length and area.
8. No storey information is in the CityGML Model.
9. For each wall, window, door or room, there is not any associated information about storey level.

The challenges that we are facing are:

BIM and GIS models are originally designed for different domains and purposes and to serve for different areas of interest. Therefore, there is a significant technology barrier which prevents the automatic transformation between them. BIM is made for detailed building information, whereas GIS is to represent large scale of real world in an efficient and simple way. As a result, CityGML cannot accommodate much detailed information that IFC contains.

Most of the current research works only consider the transformation of the building’s architectural elements, such as walls, spaces, doors, and it concentrates on the geometry transformation issues. To our knowledge, there is no systematized study on interoperability between IFC and GIS for utility networks and mechanical elements.

The semantic mappings between IFC and CityGML components are too complicate to satisfy with both geometry and semantics agreement. Regard to geometric representation, every single object in BIMs can be represented in two or even more different objects in CityGML at different LoDs. Also, complicated shapes cannot easily be modeled as or converted to GIS because of its limited capability of geometrical representation.

Old buildings do not use BIMs as building models or not completed BIMs, since their digital building models are mainly created from CAD drawings. CAD models are not object oriented, and they do not carry rich semantic information and spatial relationship between building elements, which makes the integration of building models into CityGML even harder.

In addition, right now there is not enough experiment and test which are run on the current conversion frameworks and algorithms, and no verification method is provided for the mappings and conversion results.

Our strategies to complement the results in order to answer the desired queries on facility management and maintenance operation queries are the development of a CityGML extension, GeoBIM [22] and a data warehouse.

GeoBIM is the Application Domain Extension (ADE) for CityGML by adding features and properties from IFC and implemented by the BIMServer. We studied the GeoBIM project [15], and downloaded their code from Google Code. Their experiment results show that the extended objects (stairs) in the conversion results cannot be displayed in some inspectors such as LandExplorer, but other views such as FZK viewer work well. After conversion of IFC to CityGML with the extension data, the size of the CityGML files is significantly larger than the original IFC files; they are increased by a tenfold or more. It is not possible to semantically generate a network structure in CityGML in the current research work because the pipes cannot connect as a circle. Thus, CityGML cannot model heating and sewerage systems. Even if we can extend the CityGML schema to satisfy our needs and convert them into CityGML, the LandXplore cannot display them, because they are not included in the standard CityGML schemas, other applications may not recognize them neither. Therefore, we make the conclusion that forcing CityGML to accommodate whatever we want from IFC is not a good idea, since they are two different standard schemas which serve two different domains.

Even if we can add the BIM into GIS and there are some applications which can view them, the system will be very slow, because it carries too much information. The sum of the geospatial information of the whole campus and the detailed information about each building in the campus range is significant large, and may exceed the maximum capacity of the current applications. The efficiency of running the queries on them is very low, and it will affect the user experience. Hence, we think it is not worthy and necessary that we take all of this information into the system at the same time. At first, we need to access the geospatial information of the campus, if we can locate the specified building, and then we can load the detailed building information of that building and find the room that stores the problematic equipment. By thinking of this way, we do not need to transfer BIM into GIS, we can build a data warehouse.

Our alternative approach is to build a data warehouse, which transfers and loads the databases that store all the related information of BIMs and GIS into a central data warehouse. First, we need to inspect and collect all the desired databases from a wild range of fields, such as building information, facility information, maintenance requests, geospatial 3D city model. As we know, the 3D City Database Import/Export Tool [26] is an application which is designed for the 3D City Database importing and exporting. “The 3D City Database is a free 3D geo database to store, represent, and manage virtual 3D city models on top of a standard spatial relational database” [29]. The schema of the 3D City Database is based on the City Geography Markup Language. Next, designing mediated schemas to efficiently retrieve and analyze data is our important step. We will examine and extract the most important and common spatial elements and their properties, relations to build the mediated schemas. The integration of BIM and GIS is defined at different levels and different scopes, so our mediated schemas should be defined flexibly.

The mechanical information will be added to the data warehouse attached to the room where they are stored. When people ask for any equipment, we will first find the equipment, and then its located room, which has its building and floor information. Then with the assist of geospatial database, we can quick locate the building, and find the shortest path from any given location. The challenges of this approach are both IFC and CityGML schemas are too complicate to understand, and we only need a few of them. We need to extract the necessary and related features and properties from both of them and load them into the data warehouse. There are some spatial relation database products in use right now which may facility our work that we can use them as part of our data warehouse. Creating efficient intermediate schemas to answer the queries is also difficult, since it has to cover all the possible queries’ keywords and it also need to be considered its flexibility when the schemas in the data warehouse are modified. Furthermore, since we create our own data warehouse, the schemas are not standard so it is not easy to share and exchange information. Also, it is difficult to find appropriate applications to visualize them; thus, we need to create our own visualization tool to make the query interface reliable and user-friendly to wide range of users.

(I will add more when my second part of the project has some results, the data warehouse one.)

# Summary

UBC’s Operation Service Center receives all sorts of requests on facilities, including maintenance requests, management requests and inspection requests. These facilities are widely located around the whole campus, thus any application that we build to help them will need to locate the requests at the room level and provide related information to the concerned parties. This work is currently usually done manually by few steps: (1) searching the building in the campus map and finding the shortest route to reach there, (2) locating the rooms where the facilities are stored or installed and providing details of floor information and the map of that floor so people can find the rooms in the building, and (3) In order to assist fixing the problem, information about the manufacturer, serial number, maintenance history, service manual or spare part of this specified equipment that needs to be repaired or replaced should also be provided. Our goal is to design a system which is able to trace from the geospatial network to locate the desired equipment to the room level on the campus area.

The increasing complex operation and maintenance requests draw our attention to the demand of sophisticated and efficient integration framework and technology for sharing and exchanging information in the building and geospatial outdoor models. There are two major information models to describe the spatial objects around us, such as buildings, streets, lands, city furniture, and transportation: BIM and GIS. A BIM is all about building details; it is a semantic representation of buildings and their structural components. However, GIS is a 3D city model, which is a digital representation of earth’s surface and its spatial objects. Integration of building details within their broader context seems a very promising solution to fulfill the demand of applications of management and maintenance of the facilities and utilities.

Taking advantage of the previous integration approaches and the current 3D city visualization and management applications, we decided to build our system on the BIMServer or FME WorkBench, which seem to be the only two applications available. After we get the converted CityGML files from them, we look through their content and hierarchy. In sum, their building models can achieve the details of LoD3, but only part of the LoD4, as they do not have the inner furniture, stairs and electricity units. Neither of them can convert the mechanical components in the buildings. In order to answer the queries of finding a mechanical system, we need to add this part into the current CityGML files. First we think of GeoBIM, which is an extension of CityGML. Because the standard CityGML schemas do not cover the mechanical and utility parts, we need to design the extended schemas first and implement them, which take a lot of effort. In addition, the extended parts cannot be successfully displayed in the current CityGML visualization applications, so we cannot utilize them to analyze the operation and maintenance requests.

We start our alternative approach to build a data warehouse. We can extract BIM and GIS to their respective databases separately, and build intermediate schemas, on which the operation queries can be run. We will design a data warehouse to store the building information extracted from IfcXML files, which contain the information of all the rooms and the concerned facilities and equipment in each building. Combining with the geospatial data stored in 3DCityDB in CityGML format, we are able to answer the above maintenance queries. There are some further details needed to be concerned in the future.

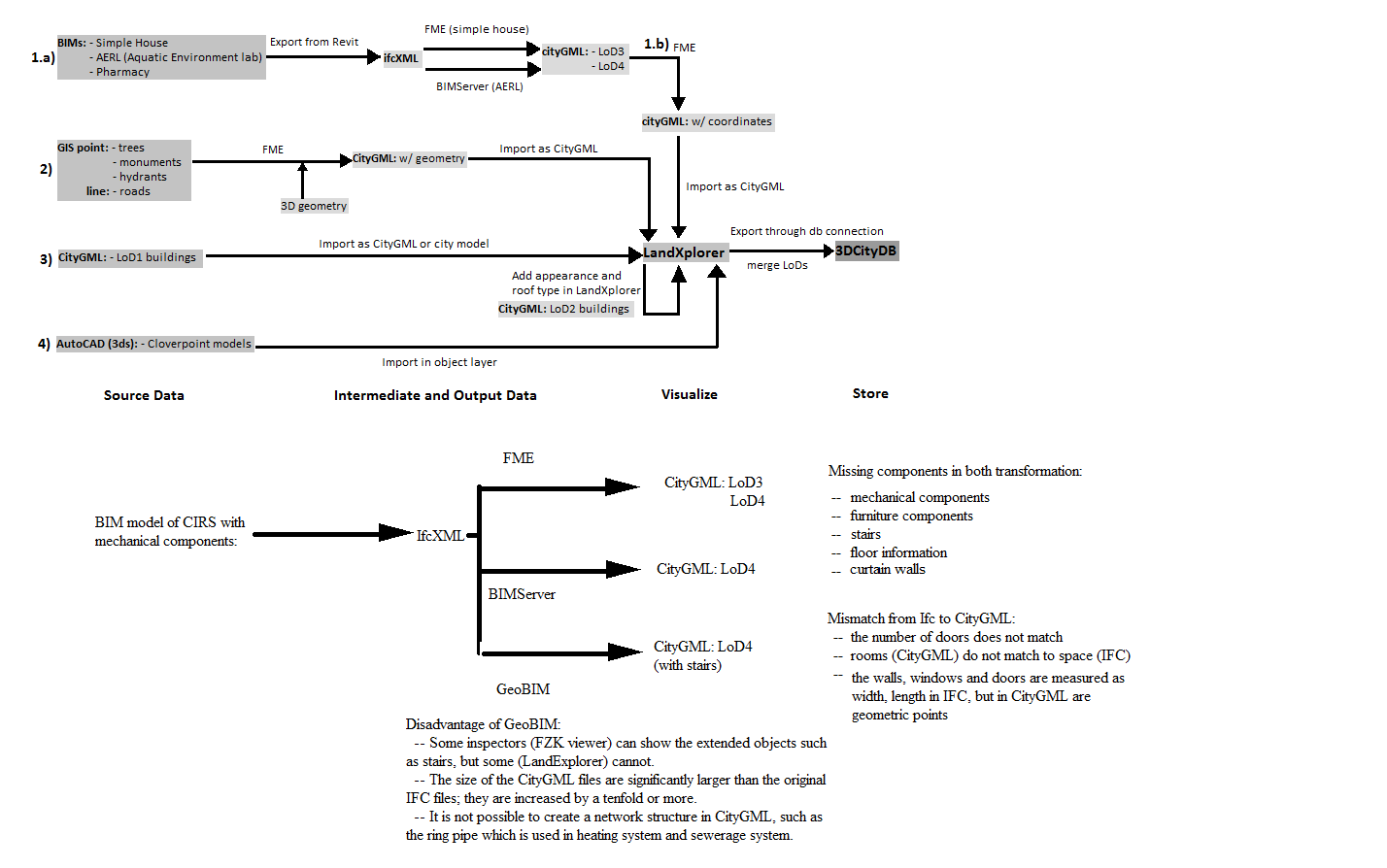


Figure 4: A summary of our current work.

Reference

[2]Building Information Modeling

<http://en.wikipedia.org/wiki/Building_information_modeling>

[3]Building Information Modeling (BIM)

<http://www.buildingsmartalliance.org/index.php/nbims/faq/>

[4]Geographic information system (GIS)

<http://en.wikipedia.org/wiki/Geographic_information_system>

[5]Industry Foundation Classes

<http://en.wikipedia.org/wiki/Industry_Foundation_Classes>

[6]Model - Industry Foundation Classes (IFC)

<http://www.buildingsmart.com/standards/ifc>

[7]Information of CityGML

<http://www.citygmlwiki.org/index.php/Basic_Information>

[8]Slide from Léon van Berlo: CityGML Extension for BIM / IFC information

<http://tools.bimtoolset.org/BIMserver/foss4g/20091022_FOSS4G-CityGML-IFC-ADE.pdf>

[9]Candidate OpenGIS® CityGML Implementation Specification (City Geography Markup Language)

<https://portal.opengeospatial.org/modules/admin/license_agreement.php?suppressHeaders=0&access_license_id=3&target=http://portal.opengeospatial.org/files/?artifact_id=16675>

[10][M.; Östman, A. (2010). "Semantic Mapping - An Ontology Engineering Method for Integrating Building Models in IFC and CityGML" [Conference paper] 3rd ISDE DIGITAL EARTH SUMMIT, 12-14 June](http://www.digitalearth-isde.org/bulgaria/pdf/32_el-mekawy_sweden_paper.pdf%5d%5bEl-Mekawy," \t "_top)

[11]IFC for GIS from CityGML Wiki:

<http://www.citygmlwiki.org/index.php/IFC_for_GIS>

[12]Nagel C. (2007), Conversion of IFC to CityGML; Meeting of the OGC 3DIM Working Group at OGC TC/PC Meeting, Paris (Frankreich), July 2007.

[13][C. Nagel, A. Stadler, T. H. Kolbe "Conceptual Requirements for the Automatic Reconstruction of Building Information Models from Uninterpreted 3D Models" Academic Track of Geoweb 2009 Conference, Vancouver.](http://www.isprs.org/proceedings/xxxviii/3_4-c3/paper_geow09/paper26_nagel_stadler_kolbe.pdf)

[14][Isikdag, Umit and S. Zlatanova (2009). "Towards defining a framework for automatic generation of buildings in CityGML using building Information Models" in: 3D Geoinformation and Sciences J. Lee and S. Zlatanova (Eds), Springer Berlin Heidelberg, pp. 79-96.](http://www.springerlink.com/content/q646537175631114/" \t "_top)

[15]Berlo L., Laat R., (2011): Integration of BIM and GIS: The development of the

CityGML GeoBIM extension, 5th International 3D GeoInfo Conference,

November 3-4, 2010, Berlin, Germany.

[16]El-mekawy M., (2010): Integrating BIM and GIS for 3D City Modelling : The

Case of IFC and CityGML, Licentiate Thesis, Royal Institute of

Technology (KTH).

[17]Mohamed El-Mekawy and Anders Östman (2010), Semantic Mapping: An

Ontology Engineering Method for Integrating Building Models in IFC and CityGML, In Proceedings of the 3rd ISDE DIGITAL EARTH SUMMIT, 12-14 June, 2010, Nessebar, Bulgaria.

[18]IFCExplorer CityGML Export from IFC Wiki: <http://www.ifcwiki.org/index.php/IfcExplorer_CityGML_Export>

[19]Safe Software from Wikipedia:

<http://en.wikipedia.org/wiki/Safe_Software>

[20]IFCExplorer from Karlsruhe Institute of Technology:

<http://www.iai.fzk.de/www-extern/index.php?id=1566>

[21]Building Information Model Server

<http://bimserver.org/>

[22][Ihab Hijazi, Manfred Ehlers, Sisi Zlatanova, Thomas Becker, and Le´on van Berlo. "Initial Investigations for Modeling Interior Utilities Within 3D Geo Context: Transforming IFC-Interior Utility to CityGML/UtilityNetworkADE" in: 3D Geoinformation and Sciences](http://www.gdmc.nl/publications/2011/Interior_Utilities_3D_Geo_Context.pdf" \t "_top)

[23] FME Desktop documentation

<http://docs.safe.com/fme/pdf/FMEGettingStarted.pdf>

[24] The Autodesk LandXplorer Studio

http://www.3dgeo.de/landx.aspx

[25] The Centre for Interactive Research on Sustainability (CIRS)

<http://cirs.ubc.ca/>

[26] 3D City Database Importer/Exporter

<http://www.3dcitydb.net/index.php?id=1880>

[27]CityGML Application Domain Extension from CityGML Wiki:

<http://www.citygmlwiki.org/index.php/CityGML-ADEs>

[28]NEXTMap Database

<http://www.intermap.com/en-us/databases/nextmap.aspx>

[29] 3D City Database

http://www.3dcitydb.net/index.php?id=1880