



LOW CARBON LIVING  
CRC

# Progressing precinct modelling on the UNSW campus and beyond: BIM/PIM and 3D GIS

## 3D PIM Modelling and Data Import



Authors	Wei Li, Mitko Aleksandrov, Jack Barton, Abdoulaye Diakit�, Jinjin Yan, Sisi Zlatanova
Title	Progressing precinct modelling on the UNSW campus and beyond: BIM/PIM and 3D GIS: 3D PIM Modelling and Data Import
ISBN	
Date	May 2019
Keywords	3D model, geospatial data, BIM, IFC, CityGML, database, PIM
Publisher	CRC for Low Carbon Living
Preferred citation	Li, W., M. Aleksandrov, J. Barton, A. Diakit�, J. Yan, S. Zlatanova, 2019, Progressing Precinct Modelling on the UNSW Campus and Beyond: BIM/PIM and 3D GIS- 3D PIM Modelling and Data Import, CRC Low Carbon Living report, 28p.



**Australian Government**  
**Department of Industry,  
Innovation and Science**

**Business**  
Cooperative Research  
Centres Programme

## Acknowledgements

This research is funded by the CRC for Low Carbon Living Ltd supported by the Cooperative Research Centres program, an Australian Government initiative

## Disclaimer

Any opinions expressed in this document are those of the authors. They do not purport to reflect the opinions or views of the CRCLCL or its partners, agents or employees.

The CRCLCL gives no warranty or assurance and makes no representation as to the accuracy or reliability of any information or advice contained in this document, or that it is suitable for any intended use. The CRCLCL, its partners, agents and employees, disclaim any and all liability for any errors or omissions or in respect of anything or the consequences of anything done or omitted to be done in reliance upon the whole or any part of this document.

## Peer Review Statement

The CRCLCL recognises the value of knowledge exchange and the importance of objective peer review. It is committed to encouraging and supporting its research teams in this regard.

The author(s) confirm(s) that this document has been reviewed and approved by the project's steering committee and by its program leader. These reviewers evaluated its:

- originality
- methodology
- rigour
- compliance with ethical guidelines
- conclusions against results
- conformity with the principles of the [Australian Code for the Responsible Conduct of Research](#) (NHMRC 2007),

and provided constructive feedback which was considered and addressed by the author(s).

© 2019 Cooperative Research for Low Carbon Living

## Contents

List of Tables .....	4
List of Figures .....	5
Acronyms .....	6
Executive Summary .....	7
Introduction .....	8
City Geography Mark-up Language (CityGML) .....	8
Modularisation .....	8
Multi-scale modelling .....	8
Semantic .....	9
Geometry and Topology .....	9
Industry Foundation Classes (IFC) .....	9
IFC .....	9
CityGML and IFC .....	10
Precinct Information Modelling (PIM) .....	11
BIM, PIM and GIS .....	11
Precinct Information Repository .....	11
Precinct Object Concepts .....	12
Data modelling .....	13
UML class diagram .....	13
Data type adaption .....	13
Project specific classes and class attributes .....	13
Simplified design of GML geometry classes .....	13
Relational database schema .....	16
Data import and export .....	16
Interaction and visualisation with PIM data .....	18
Discussion and Conclusion .....	25
References .....	26

## List of Tables

Table 1 Data type mapping .....	16
Table 2 Building table.....	17
Table 3 Terrain table .....	17
Table 4 Greensense table.....	17
Table 5 Archibus table.....	17
Table 6 Tree table .....	17
Table 7 Lawn table.....	17
Table 8 Bim table .....	17
Table 9 Road table.....	18
Table 10 Myair table.....	18

## List of Figures

Figure 1 Modularization of CityGML 2.0.....	8
Figure 2 The five level of detail (LOD) defined by CityGML ( <a href="https://www.opengeospatial.org/standards/citygml">https://www.opengeospatial.org/standards/citygml</a> ) .....	9
Figure 3 Coherence of semantics and geometry in CityGML (Yao, Z. et al., 2018) .....	9
Figure 4 UML diagram of CityGML’s geometry model (Open Geospatial Consortium, 2012).....	9
Figure 5 Spatial information platforms for built environment (Newton et al 2017) .....	11
Figure 6 Temporal information platform for the built environment: precinct life-cycle (Newton et al 2017) .....	11
Figure 7 Common precinct objects with their representation in CityGML and the PIM data model (Plume et al., 2017)...	12
Figure 8 UML diagram of core model in PIM.....	14
Figure 9 UML diagram of building model in PIM .....	14
Figure 10 UML diagram of city furniture model in PIM .....	14
Figure 11 UML diagram of terrain model in PIM.....	15
Figure 12 UML diagram of land use model in PIM .....	15
Figure 13 UML diagram of transportation model in PIM.....	15
Figure 14 UML diagram of vegetation model in PIM .....	15
Figure 15 UML diagram of energy and air model in PIM.....	16

## Acronyms

BIM	Building Information Modelling
CAD	Computer Aided Design
DBMS	Database Management System
GIS	Geographic Information System
GML	Graphics Mark-up Language
IFC	Industry Foundation Classes
ISO	International Standards Organisation
OGC	Open Geospatial Consortium
PIM	Precinct Information Modelling
SQL	Structured Query Language



## Executive Summary

Precinct Information Modelling (PIM) describes the process of creating a virtual 3D model at precinct scale, defined as a special region in built environment. PIM contains all the information pertinent the given precinct held in different data type and supports the process of management and analysis.

This technical investigation presents the PIM data schema and database design. The schema is a simplification of CityGML and IFC standards which are widely used for 3D city modelling and building works. PIM considers buildings and infrastructure equally as

“built facilities”, which contains buildings, roads, vegetations, terrain etc.

PIM is a whole process that can also be supported by spatial relational database, in order to make it as a central platform for a wide range of internal and external practitioners for planning, designing, delivering and operational management work in the built environment project. It goes further since PIM can become an important resource for the community/city/precinct who use and interact with the built environment, lending critical support for the smart cities and communities that are emerging in response to the challenges of rapid urban development in Australia.



## Introduction

The built environment is increasingly supported by an information infrastructure that is critical to the efficient and effective planning, design and management of its physical infrastructure. Precinct Information Modelling (PIM) describes the process of creating a 3D digital model at the scale of a precinct, defined as any area of the built environment that is of interest for some practical purpose. These precincts constitute the critical building blocks of good urban planning and design and their performance is central to the resilience and sustainability of cities.

The UNSW Kensington Campus lies at the heart of an extended precinct which is the subject of intensive strategic planning and assessment as part of both the Greater Sydney Commission Randwick Collaboration Area and an emerging living laboratory under Randwick City Council's Smart Cities Strategy. The management of the University is keen to develop an example of a 'smart campus' with intelligent structuring and management of geospatial data by employing open standards. Therefore, the campus was selected as a use case for developing a Precinct Information Model (PIM) and corresponding tools for management and visualisation. The data provided by Estate Management of UNSW

UNSW EM uses many spatial and non-spatial data sets (Zlatanova et al., 2019) in their daily work. However, the integration and maintenance of these data is very problematic. Data and models are scattered in different file formats and layers and maintained in a variety of software packages by different departments and institutions. This complicates the update of data and the use and re-use of information.

This report will outline progress from research at the CRC for Low Carbon Living directed towards the development of an open platform for effective sharing of precinct scale information across all urban planning and design activities to achieve expected performance outcomes for UNSW campus.

The introductory section will explain the two most prominent open standards – Industry Foundation Classes (IFC) for the BIM domain (ISO, 2013; Building SMART International, 2013), and OGC standard CityGML for the GIS domain (Open Geospatial Consortium, 2012). IFC models represent the physical elements of single constructions in great detail, and CityGML models represent entire cities in a simpler format that is usable for exchange, dissemination and spatial analyses, such as solar potential and energy consumption estimations.

## City Geography Mark-up Language (CityGML)

City Geography Mark-up Language (CityGML, cf. (Open Geospatial Consortium, 2012)) is an open data model that came into being, which utilize XML-based format for the storage and exchange of virtual 3D city models. Due to its different, well-defined Levels-of-Detail

(multiresolution model) definitions, CityGML is employed for virtual 3D precinct modelling in order to address various complex GIS simulation and analysis tasks, which go far beyond pure 3D visualization.

In this section, we give an overview of CityGML. The focus is set on the most important aspects: Modularisation, Multi-scale modelling, Semantic, Geometry and Topology.

### Modularisation

CityGML data model consists of class definitions for the most important types of objects that occur urban areas. These objects are organized into modules and can be arbitrarily combined according to their specific information needs. For this purpose, modularisation is applied to the CityGML data model.

The CityGML (2.0) data model is functionally decomposed into a *core module* and *thematic extension modules*. The core module defines the base classes for all features in CityGML. Base on the core module, each extension provides the definition of a specific thematic field, such as *Building*, *CityFurniture*, *LandUse*, *Transportation*, *Vegetation*, etc.

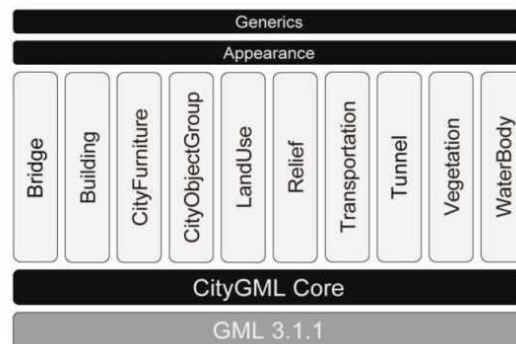


Figure 1 Modularization of CityGML 2.0

### Multi-scale modelling

CityGML differentiates five consecutive levels of detail, where objects become more detailed with increasing LOD regarding both their geometry and thematic differentiation. Further, LODs facilitate efficient visualisation and data analysis (see Fig. 2). A CityGML can contain multiple representations with different LOD for same object simultaneously.

The coarsest level LOD0 is essentially a two and a half dimensional digital terrain model (DTM). LOD1 is the well-known blocks model, without any roof structures. A building in LOD2 has distinctive roof structures and larger building installations like balconies and stairs. LOD3 denotes architectural models with detailed wall and roof structures, doors, windows and bays. LOD4 completes a LOD3 model by adding interior structures like rooms, stairs, and furniture. Figure 2 illustrates the five different LODs in CityGML.

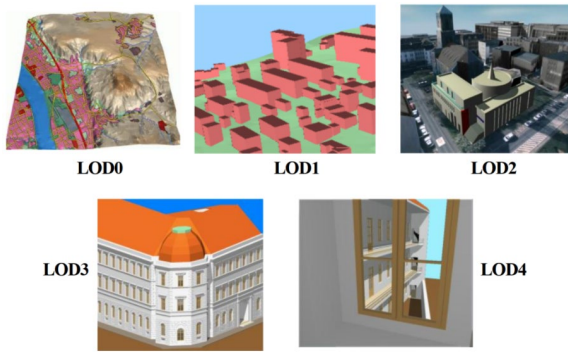


Figure 2 The five level of detail (LOD) defined by CityGML (<https://www.opengeospatial.org/standards/citygml>)

### Semantic

CityGML is very flexible regarding the expression of spatial properties of semantic objects. For example, a building can consist of building parts, which again can consist of roof, wall, and ground surfaces etc. At the semantic level, real-world entities are represented by features, such as buildings, walls, windows, or rooms. The description also includes attributes, relations and aggregation hierarchies (part-whole-relations) between features. Thus, the part-of-relationship between features can be derived at the semantic level only, without considering geometry. The semantic objects are usually used to query and analyse the building components and their thematic attributes. When objects are decomposed in the same way regarding their semantic as well as their spatial structure, they are considered to be spatial-semanticly coherent. This is illustrated for a building model in Fig. 3.

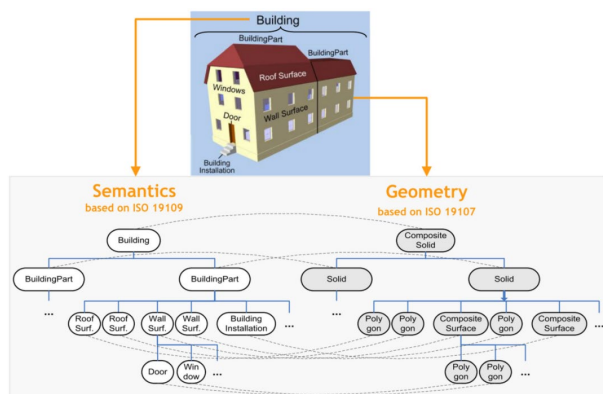


Figure 3 Coherence of semantics and geometry in CityGML (Yao, Z. et al., 2018)

### Geometry and Topology

The geometric and topological model of CityGML is realized based on the geometrical model provided by GML3 (<https://www.opengeospatial.org/standards/gml>), which is based on the ISO 19107 standard 'Spatial Schema' (Herring, J.R., 2001, Hatcher, A., 2001) for representing geometrical and topological features of real-world objects.

The geometric primitives supported by GML3 include *Point*, *Curve*, *Surface*, and *Solid*, which allow to represent spatial properties of city objects in 0 to 3

dimensions. At the same time, a solid is bounded by surfaces and a surface by curves. For each geometry type, more complex geometries with composite or aggregated hierarchies (see illustration in Fig. 4) can be constructed. The difference between aggregate and composite geometries lies in the topological relationships between the respective geometry components. Regarding aggregate geometries (*MultiPoint*, *MultiCurve*, *MultiSurface*, and *MultiSolid*), they may be disjoint, overlapping, touching, or disconnected. In contrast, a composite geometry like *CompositeCurve*, *CompositeSurface*, or *CompositeSolid* is topologically structured: its parts must be disjoint, must not overlap and are allowed to touch, at most, at their boundaries or share parts of their boundaries.

Aiming to represent topological relationships between geometries, CityGML utilizes the XML concept of links according to the GML specification. Each geometry object can have a unique identifier and can form a shared part of different aggregate or composite geometries.

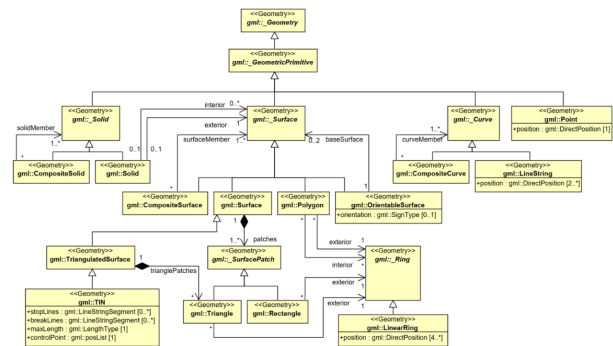


Figure 4 UML diagram of CityGML's geometry model (Open Geospatial Consortium, 2012)

### Industry Foundation Classes (IFC)

#### IFC

The Industry Foundation Classes (IFC) standard is an open data model used in the Building Information Modelling (BIM) domain for the exchange of construction models, mainly including 3D models of buildings. It has also been adapted as the ISO 16739 international standard (ISO, 2013). The geometry definitions are, however, mostly defined or derived from a different standard, ISO 10303 (ISO, 2014), which also specifies the STEP Physical File (SPF) encoding that is most commonly used in IFC files (.ifc).

Compared with CityGML, which is specified as Extensible Markup Language (XML), IFC is defined as the data modeling language EXPRESS (Borrmann, A. et al., 2015), and the entities in IFC are referred by line number. IFC also has an XML version, IfcXML, which is XML based and can be exchanged over the Internet using web-services but is not as widely used as EXPRESS-based IFC. IFC is an object-oriented data schema and largely based on STEP technology (exchange of product model data). STEP addresses the product data representation and exchange among different domains, such as manufacturing, oil and gas,

building construction, mechanical design, etc. In addition, some objects, for example road objects, are not provided by IFC; however, supports and tools are available to develop new standardized road objects. Recently, in the latest version, IFC 4 (Building SMART International, 2013), further steps have been made towards the interoperability with GIS by including new geographic elements, such as "IfcGeographicElement" and "IfcGeographicElementType".

IFC files can contain many types of classes. A considerable subset of these pertain to several different representation paradigms which can be combined freely. In practice, most IFC objects are built using sweep volumes, explicit faceted surface models and Constructive Solid Geometry (CSG) (El-Mekawy, M., & Östman, A., 2010). Elements are modelled in local coordinate systems defined by a hierarchical set of transformations that correspond to the levels in a decomposition structure (typically a site, project, building and individual floors). The representation paradigms include:

**Primitive instancing:** an object is represented based on a set number of predefined parameters. IFC uses this paradigm to define various forms of 2D profiles (Figure 2), as well as volumetric objects such as spheres, cones and pyramids.

**CSG and Boolean operations:** an object is represented as a tree of Boolean set operations (union, intersection and difference) of volumetric objects (see (Requicha, A. G., 1980) for more details). Half-spaces are often used to cut out the undesired parts of surfaces or volumes.

**Sweep volumes:** a solid can also be defined by a 2D profile (a circle, a rectangle or an arbitrary polygon with or without holes) and a curve (Wang, W. P., & Wang, K. K., 1986) along which the surface is extruded.

**B-rep:** an object is represented by its bounding surfaces, either triangulated meshes, polygonal meshes or topological arrangements of free-form surfaces.

#### CityGML and IFC

3D modelling is not only investigated in the context of geoinformation systems. The field of architecture, engineering, construction, and facility management (AEC/FM) as well as the field of computer graphics provide their own standards for the representation and exchange of 3D models. In this section, we discuss the relation of CityGML to IFC.

Building Information Modelling (BIM) means the semantic modelling of objects and processes in the field of AEC/FM and CAAD. Like in CityGML, thematic objects are represented with their 3D spatial properties and interrelationships. Data are typically exchanged using the Industry Foundation Classes (IFC), an ISO standard describing a product model and data exchange format for the built-up environment developed by the International Alliance for Interoperability (IAI).

IFC provides a very detailed semantic model for 3D building representations using constructive elements like beams, walls etc. Like in GML, IFC geometries are spatial properties of semantic objects. IFC has a very

flexible geometry model (CSG, BRep, and Sweep representations), but does not provide support for CRS. Thus, georeferencing is not possible (yet).

Since the scope of IFC is restricted to buildings and sites, no topographic feature classes like terrain, vegetation, water bodies etc. are included. These would have to be modelled and exchanged in IFC as generic objects (i.e. IFC\_Proxy).

IFC is a semantic model like CityGML, but with a different scope and at a different scale. IFC models can be converted to CityGML in different LODs preserving most of their semantic information (Benner, J., Geiger, A., & Leinemann, K., 2005, June). CSG and sweep geometries have to be converted to BRep though. This way, IFC objects can be brought into the context of a city model within a GIS or spatial database and could then become subject to spatial and thematic queries (see (Lapierre, A., & Cote, P., 2007)).

The derivation of IFC objects from CityGML data is a topic of future research, because from CityGML's surface model 3D volumetric components would have to be reconstructed. However, CityGML may be a good intermediate step in the (semi-)automatic acquisition of IFC models, because the CityGML object classes like *WallSurface*, *RoofSurface* etc. are much closer to photogrammetric observations or geodetic measurements (including laser scanning) than the component models of IFC (cf. (Kolbe, T. H., Nagel, C., & Stadler, A., 2008)).

Attempts have been made to interoperate CityGML and IFC for seeking useful common applications. These bidirectional methods can lead to development of unified applications in the areas of urban planning, building construction analysis, homeland security, etc. The benefits of these applications clearly appear at the operational level (e.g., cost reduction, unified data-view), and at the strategic level (e.g., crisis management and increasing the analyses capabilities).

In CRC-LCL project, we build a unified 3D Precinct Information Model defined as a superset model concept that is extended to contain all the features and objects from both IFC and CityGML models. It is an intermediate model relating objects from both models. Due to only building objects in our 3D campus model have both BIM and GIS data, this PIM should be capable of capturing information about spatial structures and building objects from both IFC and CityGML building models. This 3D model can be used as a starting point to support applications where information from both view (CityGML and IFC) is required for analyses. It can also facilitate modelling a database schema that is capable of capturing information that is required for all level of details.

To build the PIM, all classes with their concepts were initially collected from both models while omitting their relationships. Overlapping concepts were merged and new objects were created in such a way that both indoor and outdoor objects are captured. Finally, relationships between the objects were redefined to produce our PIM. UML notations are used for representing its objects and relationships between them.



## Precinct Information Modelling (PIM)

The built environment is increasingly supported by an information infrastructure that is critical to the efficient and effective planning, design and management of its physical infrastructure. Precinct Information Modelling (PIM) describes the process of creating a 3D digital model at the scale of a precinct, defined as any area of the built environment that is of interest for some practical purpose. These precincts constitute the critical building blocks of good urban planning and design and their performance is central to the resilience and sustainability of cities.

In this section, we give an overview of PIM. In 2013, study for the Australia CRC for Low Carbon Living (Newton et al., 2013) firstly proposed the concept – “Precinct Information Modelling (PIM)”, which was conceived as a digitally enabled information platform comprising a set of standards and protocols for urban modelling at precinct scale.

### BIM, PIM and GIS

The spatial scales for built environment modelling range from product scale objects to building objects to neighbourhood representations and upwards to cities and regions (see Figure 5). As scales change so do opportunities for different types of information modelling application: BIM -> PIM -> GIS, with the need for data interoperability across these key analytical lenses (Figure 5). Urban design spans planning and architecture as well as the GIS-BIM information and modelling environments, operating primarily at neighbourhood/precinct scale. The concept of PIM is an extension of the now familiar concept of BIM (Building Information Modelling), a 3D digital modelling process that is used widely within the building design, construction and facility management professions. PIM as proposed here is central to this scale of urban innovation.

At precinct scale, the volume of data and information required to effectively model the built environment expands significantly beyond that required for an individual building. PIM is emerging as a critical platform for more effective planning, design and management of relevant spatial data at that scale. Since it is held in an accessible open standard format, any precinct can be

### Precinct Information Repository

A precinct model can be regarded as a repository of information about a geographic area that is based on a 3D geometric object representation along with the properties and relationships of those objects: the geometry allows the precinct to be spatially visualised and navigated (but is not mandatory). Therefore, precinct models can be very large and complex, the repository usually be held in an object-oriented database. There are both proprietary and open-source products that provide database technologies capable of

modelled to accommodate the disparate needs of the range of analysis and operational activities that support more sustainable performance of precincts throughout their life-cycle (Figure 6). This is a fundamental, but still almost universally absent, component of urban development at either building or precinct scale.

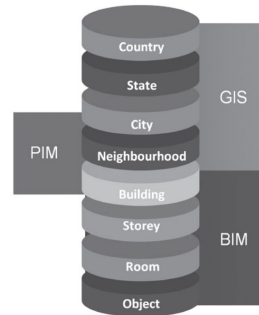


Figure 5 Spatial information platforms for built environment (Newton et al 2017)

In a precinct model, objects are typically represented as geometric forms that are spatially located, so it is often thought of as a 3D model (with the associated ability to visualise it). The relationships between objects make contribution to understand the whole built environment. Those relationships are explicitly represented within the models and may define concepts such as containment, adjacency, interdependence, connection and assemblage.

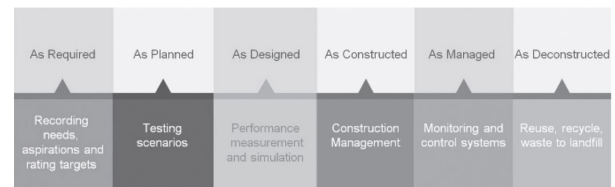


Figure 6 Temporal information platform for the built environment: precinct life-cycle (Newton et al 2017)

Notably, while the 3D model itself is important, the processes that surround this new way of working provide an opportunity to use the model in a collaborative fashion among the many professional participants involved, based on shared information within the building model.

handling PIM open data, providing different levels of functionality.

In actual, part of the data used for precinct analysis may be from multiple sources, so supporting access to multiple sources of information should be an essential aspect of the PIM property. That is important, because the PIM data standard should not replicate open data standards that already exist (or are under development) in other domains, but rather should provide support to link to such data sources at the object level (Plum et al., 2017). For example, Figure 7 - Common precinct objects with their representation in CityGML and the PIM data model.

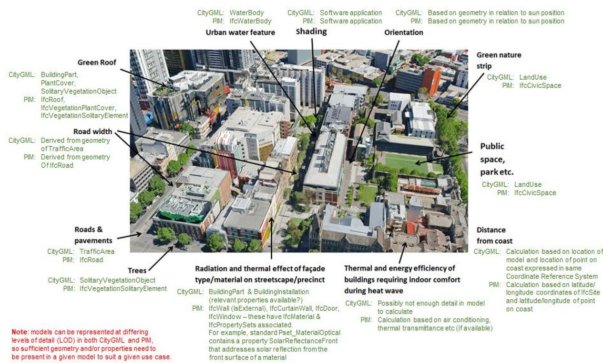


Figure 7 Common precinct objects with their representation in CityGML and the PIM data model (Plume et al., 2017)

### Precinct Object Concepts

Apart from spatial entities, there are many other precinct objects that make up the physical fabric of the built environment. Many of those are already defined in the IFC data model for buildings, but others need to be defined to support specific uses cases in PIM. Precincts can be represented at four levels of granularity (Functional, Built facility, Elemental, Occupancy), generally corresponding to the issues that are considered as precincts go through various planning and design processes.

**Elemental entities:** They are the finest level of granularity, which include elements in IFC and are assembled to make or construct the fabric of built facilities.

**Built facilities:** They represent the constructed entities in the built environment, being anything that needs to be referenced as a single unit. For example, buildings, roads, which are all the high-level spatial concepts. At that same time, some entities with functional purpose like bus stops, street furniture, are also included.

**Functional entities:** They provide the coarsest level of granularity, representing areas or regions that are designated for a specific function or identified by some common set of properties.

**Occupancy (actor):** It can be used to identify any kind of individual or group that is associated with a precinct. These may be end users, occupants, operators or managers. They may be part of a defined organisational structure, with reporting obligations and roles to play.

## Data modelling

### UML class diagram

In this section the slightly simplified data model with respect to CityGML and IFC is described at the conceptual level using UML class diagrams. These diagrams form the basis for the implementation-dependent realization of the model with a relational database system which is presented in data input report. However, UML diagrams may also form the basis for other implementations e.g., for the definition of an exchange format based on XML or GML.

#### Data type adaption

Data types specified in CityGML were substituted by data types which allow an effective representation in the database. Strings for example are used to represent code types and number vectors; GML geometry types were changed to the database geometry data type.

#### Project specific classes and class attributes

The 3D PIM database may contain some classes for representation of project specific metadata, version control and attributes for representation of additional project specific information. Since this information is represented in the CityGML and IFC specification differently or even not at all, appropriate classes and class attributes are added or respectively adopted.

#### Simplified design of GML geometry classes

Spatial properties of features are represented by objects of GML3's geometry model based on the ISO 19107 standard 'Spatial Schema' (Herring, J.R., 2001, Hatcher, A., 2001), representing 3D geometry according to the well-known Boundary Representation (B-Rep (Foley, J. et al., 1996))). Actually, only a subset of the GML3 geometry package is used.

#### UML class diagram

Design decisions in the model are explicitly visualised within the UML diagrams. Following categories are presented in an integrated model:

- Building
- Energy and Air
- City furniture
- Terrain
- Transportation
- Vegetation
- Land use

Among them, the thematic model consists of the class definitions for the most important types of objects within virtual 3D city models. Most thematic classes are derived from the basic classes *Feature* and *FeatureCollection*, the basic notions defined in ISO 19109 and GML3 for

the representation of features and their aggregations. Features contain spatial as well as non-spatial attributes, which are mapped to GML3 feature properties with corresponding data types. The thematic model also comprises different types of interrelationships between Feature classes like aggregations, generalizations, and associations.

The aim of the explicit modelling is to reach a high degree of semantic interoperability between different applications. By specifying the thematic concepts and their semantics along with their mapping to UML and GML3, different applications can rely on a well-defined set of *Feature* types, attributes, and exchange of objects and/or attributes that are not explicitly modelled in CityGML. The concepts of *GenericCityObjects* and *GenericAttributes* have been introduced.

The base class of all thematic classes within CityGML's data model is the abstract class *\_CityObject*. As we have narrow down the scope of modelling to the UNSW campus, *\_CityObject* will be replaced by *\_CampusObject*, which provides a creation and a termination date for the management of histories of features as well as generic attributes and external references to corresponding objects in other data sets. Such a reference denotes the external information system and the unique identifier of the object in this system. *\_CampusObject* is a subclass of the GML class *Feature*, thus it may inherit multiple names from *Feature*, which may be optionally qualified by a *CodeSpace*. The generalisation property *generalizesTo* of *\_CampusObject* may be used to relate features, which represent the same real-world object in different LoD.

Features of *\_CampusObject* and its specialized subclass may be aggregated to a *CampusModel*, which is a feature collection with optional metadata. The subclasses of *\_CampusObject* comprise the different thematic fields of a city model, in the following covered by separate thematic models: building model (*\_AbstractBuilding*), city furniture model (*\_CityFurniture*), digital terrain model (*Terrain*), land use model (*LandUse*), transportation model (*TransportationObject*), vegetation model (*\_VegetationObject*). In addition, we merge energy and air data into PIM by directly adding Energy and Air class in UML diagram.



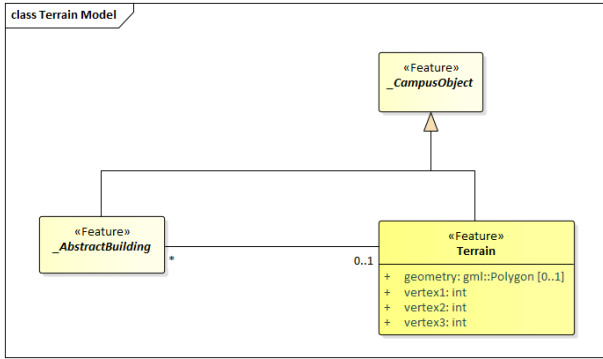


Figure 11 UML diagram of terrain model in PIM

*LandUse* objects describe areas of the earth's surface dedicated to a specific land use. They can be employed to represent parcels in 3D. Every *LandUse* object may have the attributes *class* (e.g., settlement area, industrial area, farmland), and *geometry* which can be used to express geometry feature by *Polygon*.

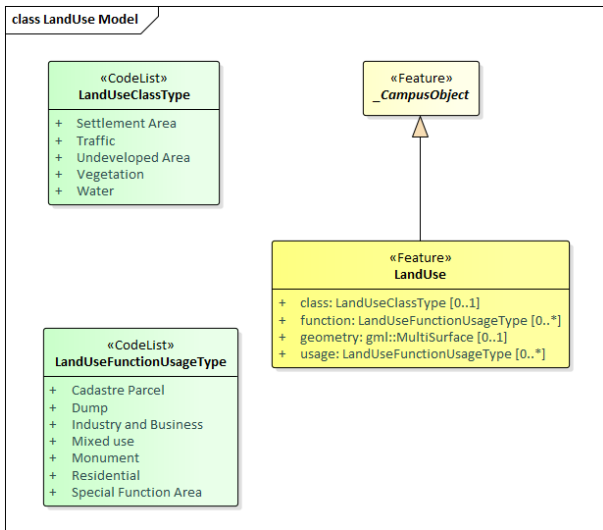


Figure 12 UML diagram of land use model in PIM

The transportation model of CityGML is a multi-functional, multi-scale model focusing on thematic and functional as well as geometrical/topological aspects. The main class is *TransportationComplex* which represents, for example, a road, a track, a railway, or a square. It is composed of the parts *TrafficArea* and *AuxiliaryTrafficArea*. In our campus model, we don't consider the *AuxiliaryTrafficArea*. The road itself is represented as a *TransportationComplex*. *TransportationComplex* Objects can be thematically differentiated using the subclasses *Road*.

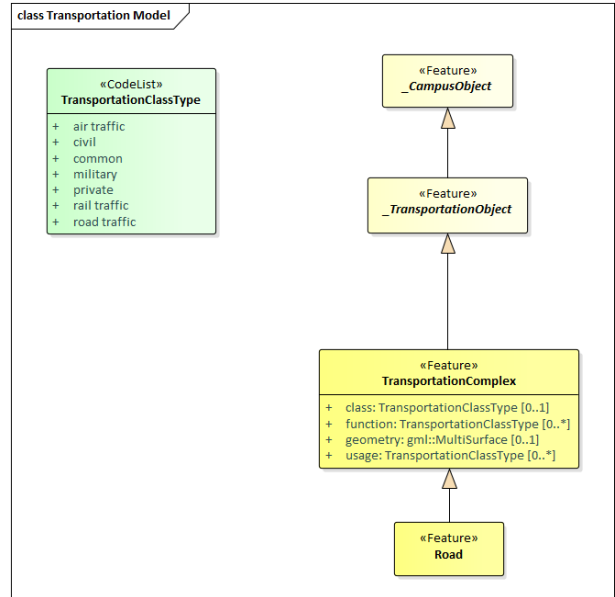


Figure 13 UML diagram of transportation model in PIM

Vegetation features are important components of a 3D campus model, because they support the recognition of the surrounding environment. By the analysis and visualisation of vegetation objects, statements on their distribution, structure and diversification can be made. The vegetation model of CityGML distinguishes between solitary vegetation objects like trees and vegetation areas, which represent biotopes like forest or other plant communities. Single vegetation objects are modelled by the class *SolitaryVegetationObject*, while for areas filled with specific vegetation the class *PlantCover* is used.

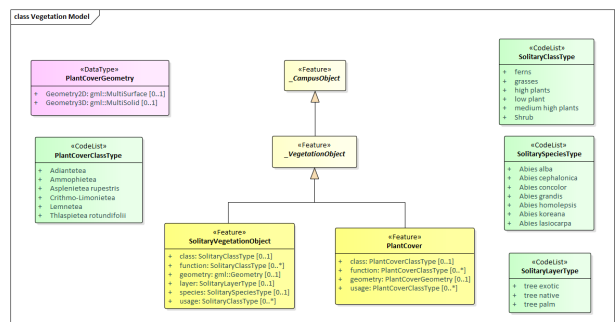


Figure 14 UML diagram of vegetation model in PIM

### Energy and Air

Electricity use and gas consumption data are monitored through Greensense system. The data is attributed to the buildings and therefore there is no possibility to specify the level of energy consumption per room or as part of the building. On the contrary, Air quality data is obtained from installed sensors within several rooms at Red Centre Building. Indoor CO2 data is measured and maintained by the system Myair. This data is quite interesting for the purpose of the project as they are recorded and maintained per room.



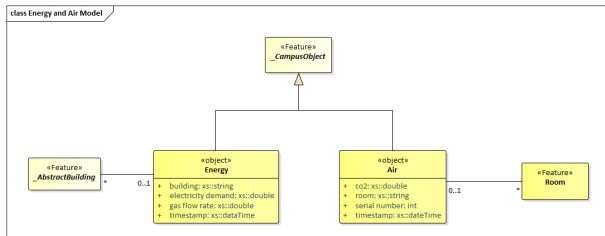


Figure 15 UML diagram of energy and air model in PIM

## The Unified Building Model

As BIM data is limited to building structure, the objects beyond building (such as terrain, road) are not involved integration. Here, we only consider building object.

A building in CityGML consists of rooms, where a room is a space surrounded by different boundary surfaces. Storeys are not explicitly defined. However, in IFC, the structure is smoothly organized by breaking down a building into storeys and then into spaces that form a specific storey. In the PIM, we consider concepts from IFC as well as CityGML. A building in the PIM consists of explicit definition of storeys, at least one storey in a building. Each building storey may have zero or more spaces related to it. A space can be either opened or closed, and the closed space represents a room which corresponds to the room definition in CityGML.

## Relational database schema

In this section, we give an overview of database solutions support the management of CityGML data. Since CityGML is a GML application schema, these software systems are able to automatically create database schemas for storing CityGML data for various database management systems like the *ORACLE Spatial* or *PostgreSQL/PostGIS*, using the CityGML XML Schema definition files.

There are strong reasons to employ spatially-extended relational database management systems (SRDBMS) to store and manage complex 3D city models. First, SRDBMS support all required geometry types and provide means for proper spatial indexing as well as for geometric and topological analyses. Second, SRDBMS can directly be used by most geoinformation systems (GIS) or spatially enabled ELT (Extract, Transform, Load) tools. There exists a variety of non-relational databases like object-oriented databases, which are increasingly investigated and employed in many application fields (cf. (Ordonez, C. et al., 2010)). However, they are currently still more or less limited in their capabilities and performance regarding spatial operations and coordinate transformations, which are of great importance for the enterprise use in GIS applications (cf. (Agoub, A. et al., 2016)). Therefore, SRDBMS such as the commercial software *ORACLE Spatial/Locator* and the Open Source software *PostgreSQL with PostGIS extension* play a major role for GIS due to their extensive capabilities in handling 3D spatial data.

**Mapping rules, schema conventions:** Generally, one or more classes of the UML diagram are mapped onto

one table; the name of the table is identical to the class name (a leading underscore indicating an abstract class is left out). The scalar attributes of the classes become columns of the corresponding table with identical name.

**Multiplicities of attributes:** Attributes with a variable amount of occurrences (\*) are substituted by a data type enabling the storage of arbitrary values – String. This means that object attributes can be stored in a single column.

**Cardinalities and types of relationships:** n:m relations require an additional table in the database. This table consists of the primary keys of both elements' tables which form a composite primary key. If the relation can be restricted to 1:n or n:1 relationship the additional table can be avoided. Therefore, all n:m relations in CityGML were checked for a more restrictive definition. This results in simplified cardinalities and relations.

**Simplified treatment of recursions:** Some recursive relations are used in the CityGML data model. Recursive database queries may cause high cost, especially if the amount of recursive steps is unknown. In order to guarantee good performance, implementation of recursive associations receive two additional columns which contain the ID of the parent and of the root element.

The types of the attributes are customized to corresponding database (PostgreSQL/PostGIS) data type (see Table 1). Some attributes of the data type date were mapped to *TIMESTAMP WITH TIME ZONE* to allow a more accurate storage of time values.

Table 1 Data type mapping

Data Type Mapping	
UML	PostgreSQL/PostGIS
String, anyURI	VARCHAR, TEXT
Double	NUMERIC
Enumeration	VARCHAR
GML Geometry	GEOMETRY
CodeType	VARCHAR
Date	DATE, TIMESTAMP WITH TIME ZONE

## Data import and export

Tables in PostGIS contain IFC objects and CityGML objects. Following tables are presented in our database server:

- Building
- Terrain
- Greensense
- Archibus
- Tree
- Lawn
- Bim

- Road
- Myair

Table 2 Building table

Name	Data type	Length	Description
id	INT		primary key (PK)
name	VARCHAR	200	building name (FK)
building_id	INT		building ID
max_height	REAL		building max height
geom	geometry		geometry

Table 3 Terrain table

Name	Data type	Length	Description
id	INT		primary key (PK)
vertex1_id	INT		vertex ID
vertex2_id	INT		vertex ID
vertex3_id	INT		vertex ID
terrain_id	INT		terrain ID
geom	geometry		geometry

Table 4 Greensense table

Name	Data type	Length	Description
id	INT		primary key (PK)
building	VARCHAR	200	building name (FK)
electricity_demand	real		Electricity Demand (kW)
gas_flow_rate	real		Gas Flow Rate (m <sup>3</sup> /s)
timestamp	TIMESTAMP		Timestamp

Table 5 Archibus table

Name	Data type	Length	Description
id	INT		primary key (PK)
building	VARCHAR	200	building name (FK)
floor	VARCHAR	50	floor name

room	VARCHAR	50	room name (FK)
room_type_description	VARCHAR	200	room type
room_function_description	VARCHAR	200	room function
division_faculty_name	VARCHAR	200	division faculty name
unit_name	VARCHAR	200	unit name
current_room_capacity	INT		room capacity
room_area_m_2	NUMERIC	(5,2)	room area (m <sup>2</sup> )
room_comments	VARCHAR	200	comments

Table 6 Tree table

Name	Data type	Length	Description
id	INT		primary key (PK)
layer	VARCHAR	200	
tree_id	VARCHAR	200	tree ID
code	VARCHAR	200	tree code
species	VARCHAR	50	species
geom	geometry		geometry

Table 7 Lawn table

Name	Data type	Length	Description
id	INT		primary key (PK)
lawn_id	INT		lawn ID
name	VARCHAR	200	lawn name
geom	geometry		geometry

Table 8 Bim table

Name	Data type	Length	Description
id	INT		primary key (PK)
building	VARCHAR	200	building name (FK)
ifc_class	VARCHAR	50	class type
ifc_guid	VARCHAR	200	object guid

ifc_name	VARCHA R	200	object name
ifc_description	VARCHA R	2000	description
ifc_containing_store y	VARCHA R	200	storey
geom	geometry		geometry

Table 9 Road table

Name	Data type	Length	Description
id	INT		primary key (PK)
road_id	INT		road ID
geom	geometry		geometry

Table 10 Myair table

Name	Data type	Length	Description
id	INT		primary key (PK)
room	VARCHAR	50	room name (FK)
myair_num	INT		myair number
co2	INT		co2 emission
timestamp	TIMESTAMP		timestamp

## Interaction and visualisation with PIM data

The precinct information model (PIM) is stored in an object-oriented database -- PostGIS in our project, which can be remotely accessed via host address, port, and password. Data in our database can also be accessed via web requests, which allows external application developers to create either web-based or standalone applications that interact with relevant data entities in PIM for particular purposes. Precinct stakeholders are allowed to query and update their objectives against the existing precinct entities in PIM.

Further, QGIS software can be used for visualising PIM entities. It can open a model across internet connection to the database server, edit that model and save the amended model either to a model file on the local system or merge it back into the source model on the server.

The following cases briefly outlined below involve PIM-based data query in UNSW campus.

Q1: Select all buildings within 300 m from Red centre (the center point of Red Centre and radius 300m).

```

1. SELECT DISTINCT
2. b1.id, b1.name, ST_AsText(b1.geom)
   AS geom
3. FROM
4. building_mesh_3D b1,
   building_mesh_3D b2
5. WHERE b2.name LIKE 'Red %'
6. AND ST_DWithin(b2.geom, b1.geom,
300)
7. AND b1.id!=b2.id
8. AND b1.name IS NOT NULL;

```

```

id      name      geom
1      Science Theatre - F13
      MULTIPOLYGON Z (((336340.893175
6245605.12916 29.0131375552,336374.977724
6245591.42136 29.8644541456,336375.426945
6245599.85659 30.143911876,336340.893175
6245605.12916 29.0131375552)),((336340.893175
6245605.12916 29.0131375552,336379.005882
6245624.38317 30.0502286574,336375.426945
6245599.85659 30.143911876,336340.893175
6245605.12916 29.0131375552)), ...)

2      Barker Street Apartments - N13
      MULTIPOLYGON Z (((336285.513289
6245358.06803 27.4794220633,336283.254867
6245350.85761 27.2200197434,336289.006128
6245350.3571 27.6106464631,336285.513289
6245358.06803 27.4794220633)),((336283.254867
6245350.85761 27.2200197434,336287.251232
6245343.1044 27.3621510132,336289.006128
6245350.3571 27.6106464631,336283.254867
6245350.85761 27.2200197434)), ...)

3      New College - L6      MULTIPOLYGON Z
(((336057.054083 6245471.21322
26.7521974499,336057.989662 6245465.10489
26.6992086528,336084.187718 6245448.18611
26.6877966811,336057.054083 6245471.21322
26.7521974499)),((336057.054083 6245471.21322
26.7521974499,336080.882549 6245506.99413
26.8991932228,336086.495548 6245448.56379
26.6973721672,336057.054083 6245471.21322
26.7521974499)),...)

```

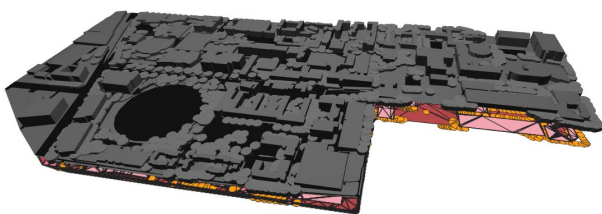
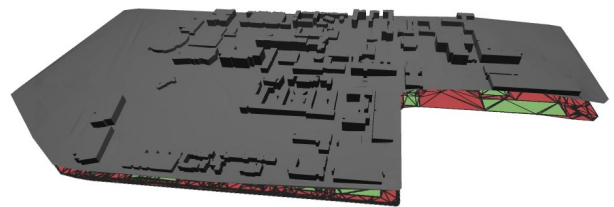


Figure 16. 3D campus map with building, lawn, tree and road



**Q2: Query rooms of red centre building whose area is larger than 10 m<sup>2</sup> and total co2 emission is larger than 5000 in Dec. 2018.**

```

1. SELECT DISTINCT ifc_guid, ifc_name,
2. ST_AsText(R.geom) AS geom
3. FROM
4. PUBLIC."Red_Center_H13_IFC_color" R,
5. (SELECT room FROM myair
6. GROUP BY room HAVING sum(co2) >
7. 5000) AS tmp1,
8. (SELECT
9. ST_Z((ST_DumpPoints(geom)).geom)
10. AS z FROM
11. PUBLIC."Red_Center_H13_IFC_color"
12. WHERE ifc_class='IfcSpace') AS tmp2
13. WHERE ifc_class='IfcSpace'
14. AND z > 10
15. AND room=ifc_name;

```

```

ifc_guid  ifc_name  geom
23gGJrgOb10Pi5JS92M88k      2035
POLYHEDRALSURFACE Z (((151.229605597 -
33.917961174 40.600780359,151.229605194 -
33.917963364 40.600780359,151.229605194 -
33.917963364 44.641246,151.229605597 -
33.917961174 40.600780359)),((151.229605194 -
33.917963364 40.600780359,151.229605597 -
33.917961174 40.600780359,151.229639866 -
33.917967764 40.600780359,151.229605194 -
33.917963364 40.600780359)),((151.229605194 -
33.917963364 44.641246,151.229605194 -
33.917963364 40.600780359,151.229639866 -
33.917967764 44.641246,151.229605194 -
33.917963364 44.641246)),((151.229605597 -
33.917961174 40.600780359,151.229605194 -
33.917963364 44.641246,151.229605597 -
33.917961174 44.641246,151.229605597 -
33.917961174 40.600780359)),...)

23gGJrgOb10Pi5JS92M8rP      2001
POLYHEDRALSURFACE Z (((151.229297938 -
33.917839951 40.600780359,151.229196671 -
33.917827099 40.600780359,151.229196671 -
33.917827099 44.641246,151.229297938 -
33.917839951 40.600780359)),((151.229196671 -
33.917827099 40.600780359,151.229297938 -
33.917839951 40.600780359,151.229237409 -
33.917887772 40.600780359,151.229196671 -
33.917827099 40.600780359)),((151.229196671 -
33.917827099 44.641246,151.229196671 -
33.917827099 40.600780359,151.229186388 -
33.917882897 44.641246,151.229196671 -
33.917827099 44.641246)),((151.229297938 -
33.917839951 40.600780359,151.229196671 -
33.917827099 44.641246,151.229297938 -
33.917839951 44.641246,151.229297938 -
33.917839951 40.600780359)),((151.229237409 -
33.917887772 40.600780359,151.229297938 -
33.917839951 40.600780359,151.22929536 -
33.917853942 40.600780359,151.229237409 -
33.917887772 40.600780359)),...)

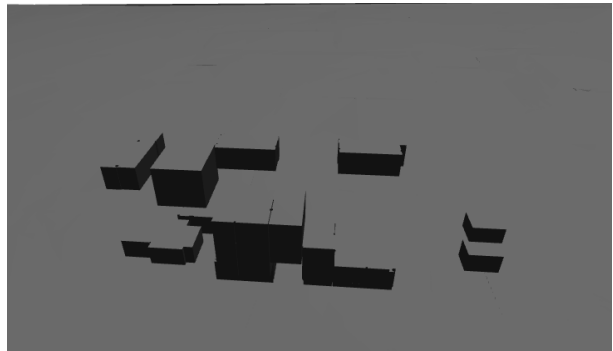
1vTwSEsEXDnPuWEiU6ic1G      6005
POLYHEDRALSURFACE Z (((151.229673197 -

```

```

33.917870225 56.762642923,151.229672702 -
33.917869905 56.762642923,151.229672702 -
33.917869905 61.397294688,151.229673197 -
33.917870225 56.762642923)),((151.229672702 -
33.917869905 56.762642923,151.229673197 -
33.917870225 56.762642923,151.229673191 -
33.917870257 56.762642923,151.229672702 -
33.917869905 56.762642923)),((151.229672702 -
33.917869905 61.397294688,151.229672702 -
33.917869905 56.762642923,151.229672311 -
33.917869497 61.397294688,151.229672702 -
33.917869905 61.397294688)),((151.229673197 -
33.917870225 56.762642923,151.229672702 -
33.917869905 61.397294688,151.229673197 -
33.917870225 61.397294688,151.229673197 -
33.917870225 56.762642923)),...)
.....

```



**Q3: Select all the rooms of Red Centre that are higher than the building of Mining.**

```

1. SELECT R.ifc_guid, R.ifc_name,
2. ST_AsText(R.geom) AS geom
3. FROM
4. PUBLIC."Red_Center_H13_IFC_color" R,
5. (SELECT ifc_guid, MIN(z) AS height
6. FROM
7. (SELECT
8. ST_Z((ST_DumpPoints(geom)).geom)
9. AS z, ifc_guid, ifc_name, geom
10. FROM
11. PUBLIC."Red_Center_H13_IFC_color"
12. WHERE ifc_class='IfcSpace')
13. AS tmp1
14. GROUP BY ifc_guid) AS tmp2,
15. (SELECT MAX(CAST(max_height
16. AS double precision))
17. AS height
18. FROM building_mesh_3d
19. WHERE name='Old Main - K15')
20. AS tmp3
21. WHERE tmp2.ifc_guid=R.ifc_guid
22. AND ifc_class='IfcSpace'
23. AND tmp2.height > tmp3.height;

```

```

ifc_guid  ifc_name  geom
1HsQ1r1QP7KBWmsuuZc92q      5009
POLYHEDRALSURFACE Z (((151.22967532 -
33.917884831 53.53027041,151.22968102 -

```

```

33.917885555 53.53027041,151.22968102 -
33.917885555 56.762642923,151.22967532 -
33.917884831 53.53027041)),((151.22968102 -
33.917885555 53.53027041,151.22967532 -
33.917884831 53.53027041,151.229681424 -
33.917883364 53.53027041,151.22968102 -
33.917885555 53.53027041)),.....)

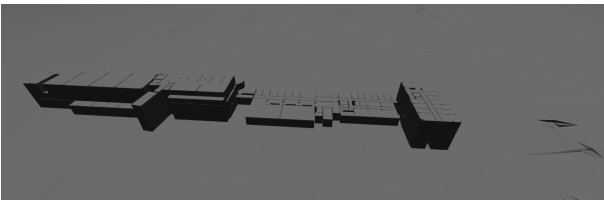
1HsQ1rlQP7KBWmsuuZc998          5005
POLYHEDRALSURFACE Z (((151.22930677 -
33.917831692 53.53027041,151.229331865 -
33.917834877 53.53027041,151.229331865 -
33.917834877 56.762642923,151.22930677 -
33.917831692 53.53027041)),((151.229331865 -
33.917834877 53.53027041,151.22930677 -
33.917831692 53.53027041,151.229311658 -
33.91780517 53.53027041,151.229331865 -
33.917834877 53.53027041)),.....)

.....

1HsQ1rlQP7KBWmsuuZc99B          5006
POLYHEDRALSURFACE Z (((151.22946408 -
33.917843573 53.53027041,151.229463648 -
33.917843287 53.53027041,151.229463648 -
33.917843287 61.397294688,151.22946408 -
33.917843573 53.53027041)),((151.229463648 -
33.917843287 53.53027041,151.22946408 -
33.917843573 53.53027041,151.229464066 -
33.917843646 53.53027041,151.229463648 -
33.917843287 53.53027041)), .....)

.....

```



Q4: Which building has the highest roof and how high is it?

1. **SELECT** id, name, max\_height,
2. **ST\_AsText**(geom) **AS** geom
3. **FROM** building\_mesh\_3d
4. **WHERE** max\_height =
5. (**SELECT MAX**(max\_height)
6. **FROM** building\_mesh\_3d);

id	name	max_height	geom
27	Lowy Cancer Research Centre - C25	86.2825	MULTIPOLYGON Z (((336907.382023 6245615.87726 56.6924806458,336907.036978 6245615.93595 56.6872231836,336907.31358 6245615.56431 56.6861475709,336907.382023 6245615.87726 56.6924806458)),((336908.74003 6245623.86191 56.6867575495,336908.394985 6245623.92059 56.6929547008,336908.686376 6245623.54644 56.6932187815,336908.74003 6245623.86191 56.6867575495)),((336910.102404 6245631.87223 56.6058756423,336909.757359 6245631.93092

```

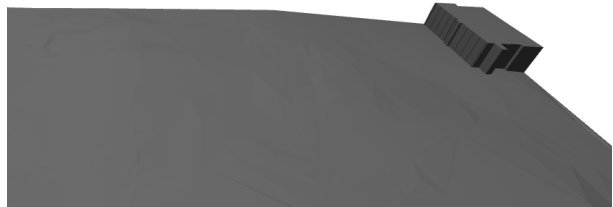
56.6131606489,336910.04875 6245631.55676
56.6064888952,336910.102404 6245631.87223
56.6058756423)),...)

38      Wallace Wurth Building - C27      86.2825
MULTIPOLYGON Z (((336907.382023
6245615.87726 56.6924806458,336907.036978
6245615.93595 56.6872231836,336907.31358
6245615.56431 56.6861475709,336907.382023
6245615.87726 56.6924806458)),((336908.74003
6245623.86191 56.6867575495,336908.394985
6245623.92059 56.6929547008,336908.686376
6245623.54644 56.6932187815,336908.74003
6245623.86191 56.6867575495)),((336910.102404
6245631.87223 56.6058756423,336909.757359
6245631.93092 56.6131606489,336910.04875
6245631.55676 56.6064888952,336910.102404
6245631.87223 56.6058756423)),...)

109     Biological Sciences      86.2825
MULTIPOLYGON Z (((336907.382023
6245615.87726 56.6924806458,336907.036978
6245615.93595 56.6872231836,336907.31358
6245615.56431 56.6861475709,336907.382023
6245615.87726 56.6924806458)),((336908.74003
6245623.86191 56.6867575495,336908.394985
6245623.92059 56.6929547008,336908.686376
6245623.54644 56.6932187815,336908.74003
6245623.86191 56.6867575495)),((336910.102404
6245631.87223 56.6058756423,336909.757359
6245631.93092 56.6131606489,336910.04875
6245631.55676 56.6064888952,336910.102404
6245631.87223 56.6058756423)),...)

.....

```



Q5: Which is the highest point of the terrain? Consider a vertex of the TIN?

1. **SELECT DISTINCT** tmp1.id,
2. **\_vertex1\_id,**
3. **\_vertex2\_id, \_vertex3\_id,**
4. **ST\_AsText**(tmp1.geom) **AS** geom
5. **FROM**
6. (**SELECT**
7. **ST\_Z**((**ST\_DumpPoints**(geom)).geom)
8. **AS** z,
9. **id, \_vertex1\_id,**
10. **\_vertex2\_id, \_vertex3\_id, geom**
11. **FROM PUBLIC."terrain\_unsw")**
12. **AS** tmp1,
13. (**SELECT MAX**(z)

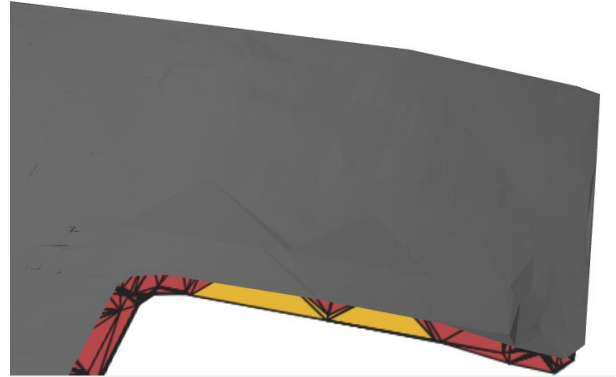


```

13. AS z_max
14. FROM (SELECT
15. ST_Z((ST_DumpPoints(geom)).geom)
16. AS z, id, geom
17. FROM PUBLIC."terrain_unsw")
18. AS tmp2) AS tmp3
19. WHERE tmp1.z=tmp3.z_max;

```

id	_vertex1_id _vertex3_id	_vertex2_id geom		
86596	1048579	1359873	1208321	POLYGON Z ((336606.660046 6245487.39815 51.228937325,336722.675919 6245470.72536 59.650013086,336655.472253 6245524.98877 52.8464366994,336606.660046 6245487.39815 51.228937325))
87608	1208321	1359873	1327106	POLYGON Z ((336655.472253 6245524.98877 52.8464366994,336722.675919 6245470.72536 59.650013086,336723.260182 6245474.72288 58.4439901563,336655.472253 6245524.98877 52.8464366994))
88915	1327105	1327106	1359873	POLYGON Z ((336723.545254 6245474.68122 58.4892291965,336723.260182 6245474.72288 58.4439901563,336722.675919 6245470.72536 59.650013086,336723.545254 6245474.68122 58.4892291965))
89520	1318916	1359873	1347589	POLYGON Z ((336770.47859 6245498.35416 55.6888736921,336722.675919 6245470.72536 59.650013086,336810.673657325 6245457.98234863 58.7283314138114,336770.47859 6245498.35416 55.6888736921))
89522	1318916	1318919	1359873	POLYGON Z ((336770.47859 6245498.35416 55.6888736921,336767.12676 6245498.91544 55.5680336294,336722.675919 6245470.72536 59.650013086,336770.47859 6245498.35416 55.6888736921))
89530	1318919	1327105	1359873	POLYGON Z ((336767.12676 6245498.91544 55.5680336294,336723.545254 6245474.68122 58.4892291965,336722.675919 6245470.72536 59.650013086,336767.12676 6245498.91544 55.5680336294))
89843	1347586	1347589	1359873	POLYGON Z ((336811.141735 6245457.13749 58.8493496588,336810.673657325 6245457.98234863 58.7283314138114,336722.675919 6245470.72536 59.650013086,336811.141735 6245457.13749 58.8493496588))



Q6: Which building is most far away from Red centre?

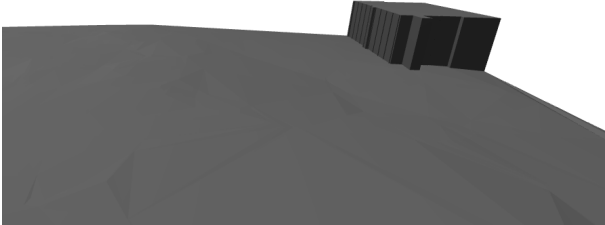
```

1. SELECT b2.id, b2.name,
   ST_AsText(b2.geom) AS geom
2. FROM building_mesh_3d b1,
   building_mesh_3d b2
3. WHERE b1.name LIKE 'Red%' AND
   ST_Distance(b1.geom, b2.geom) =
4. (SELECT MAX(ST_Distance(b1.geom,
   b2.geom))
5. FROM building_mesh_3d b1,
   building_mesh_3d b2
6. WHERE b1.name LIKE 'Red%');

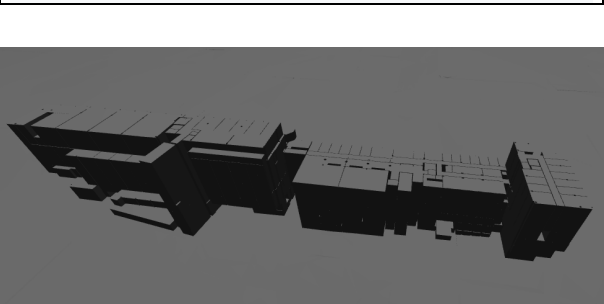
```

id	name	geom
27	Low Cancer Research Centre - C25	MULTIPOLYGON Z (((336907.382023 6245615.87726 56.6924806458,336907.036978 6245615.93595 56.6872231836,336907.31358 6245615.56431 56.6861475709,336907.382023 6245615.87726 56.6924806458)),((336908.74003 6245623.86191 56.6867575495,336908.394985 6245623.92059 56.6929547008,336908.686376 6245623.54644 56.6932187815,336908.74003 6245623.86191 56.6867575495)),...)
38	Wallace Wurth Building - C27	MULTIPOLYGON Z (((336907.382023 6245615.87726 56.6924806458,336907.036978 6245615.93595 56.6872231836,336907.31358 6245615.56431 56.6861475709,336907.382023 6245615.87726 56.6924806458)),((336908.74003 6245623.86191 56.6867575495,336908.394985 6245623.92059 56.6929547008,336908.686376 6245623.54644 56.6932187815,336908.74003 6245623.86191 56.6867575495)),...)
109	Biological Sciences	MULTIPOLYGON Z (((336907.382023 6245615.87726 56.6924806458,336907.036978 6245615.93595 56.6872231836,336907.31358 6245615.56431 56.6861475709,336907.382023 6245615.87726 56.6924806458)),((336908.74003 6245623.86191 56.6867575495,336908.394985 6245623.92059 56.6929547008,336908.686376 6245623.54644

```
56.6932187815,336908.74003 6245623.86191
56.6867575495)),...)
```



```
52.484502833,151.229991113 -33.917908309
48.063758073)),((151.229990558 -33.917909359
48.063758073,151.229991113 -33.917908309
48.063758073,151.229994052 -33.917906733
48.063758073,151.229990558 -33.917909359
48.063758073)),((151.229990558 -33.917909359
52.484502833,151.229990558 -33.917909359
48.063758073,151.229989444 -33.917910038
52.484502833,151.229990558 -33.917909359
52.484502833)),((151.229991113 -33.917908309
48.063758073,151.229990558 -33.917909359
52.484502833,151.229991113 -33.917908309
52.484502833,151.229991113 -33.917908309
48.063758073)),...)
```



Q7: What is the area of all rooms in the Red Centre?

1. **SELECT** ifc\_name, ST\_Area(geom),
2. ST\_AsText(geom) **AS** geom
3. **FROM**
4. public."Red\_Center\_H13\_IFC\_color"
5. **WHERE** ifc\_class='IfcSpace';

```
ifc_name  st_area  geom
4041      2.92E-08  POLYHEDRALSURFACE Z
(((151.229772927 -33.917930247
48.063758073,151.229764984 -33.91797335
48.063758073,151.229764984 -33.91797335
52.484502833,151.229772927 -33.917930247
48.063758073)),((151.229764984 -33.91797335
48.063758073,151.229772927 -33.917930247
48.063758073,151.229813097 -33.917979358
48.063758073,151.229764984 -33.91797335
48.063758073)),((151.229764984 -33.91797335
52.484502833,151.229764984 -33.91797335
48.063758073,151.22976392 -33.917973215
52.484502833,151.229764984 -33.91797335
52.484502833)),...)
```

```
4040      3.32E-09  POLYHEDRALSURFACE Z
(((151.229886354 -33.917895148
48.063758073,151.229886137 -33.917895745
48.063758073,151.229886137 -33.917895745
52.484502833,151.229886354 -33.917895148
48.063758073)),((151.229886137 -33.917895745
48.063758073,151.229886354 -33.917895148
48.063758073,151.22988372 -33.917906733
48.063758073,151.229886137 -33.917895745
52.484502833,151.229886137 -33.917895745
52.484502833,151.229886137 -33.917895745
52.484502833)),((151.229886354 -33.917895148
48.063758073,151.229886137 -33.917895745
52.484502833,151.229886354 -33.917895148
52.484502833,151.229886354 -33.917895148
48.063758073)),...)
```

```
4042      5.15E-09  POLYHEDRALSURFACE Z
(((151.229991113 -33.917908309
48.063758073,151.229990558 -33.917909359
48.063758073,151.229990558 -33.917909359
```

Q8: Which ration room/CO2 emission is the worst?

1. **SELECT** MAX(tmp.co2/room\_area\_m\_2)
2. **FROM** archibus,
3. (**SELECT** room, SUM(co2) **AS** co2
4. **FROM** myair
5. **GROUP BY** room having SUM(co2)>0)
6. **AS** tmp
7. **WHERE** archibus.room=tmp.room;

31706.80123

Q9: Are here buildings that intersect (the extruded buildings check with ANYINTERSECT)

1. **SELECT DISTINCT** b1.id, b1.name,
2. ST\_AsText(b1.geom) **AS** geom
3. **FROM** building\_mesh\_3d b1,
4. building\_mesh\_3d b2
5. **WHERE** b1.id!=b2.id
6. **AND**
7. ST\_3DIntersects(b1.geom,b2.geom);

```
id      name      geom
6      Red Center (East) - H13
MULTIPOLYGON Z (((336399.763146
6245519.7228 31.2876782523,336399.734945
6245519.52985 31.2842089058,336399.987264
6245519.49297 31.288701432,336399.763146
6245519.7228 31.2876782523)),((336393.757059
6245520.60133 31.0180259232,336393.728858
6245520.40838 31.0253569546,336394.014325
6245520.56373 31.0427176759,336393.757059
6245520.60133 31.0180259232)),((336390.73918
```

```

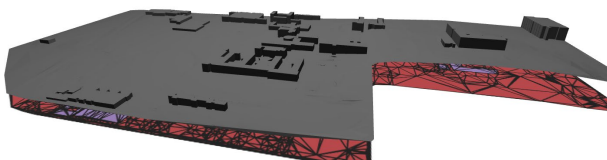
6245521.04282 30.7283576353,336390.710979
6245520.84987 30.7356886667,336390.996433
6245521.00512 30.7530531471,336390.73918
6245521.04282 30.7283576353),...)

7      UNSW Village      MULTIPOLYGON Z
(((336421.095163 6245739.03126
33.5081358908,336410.362141 6245741.81211
33.4014360849,336410.63874 6245735.48555
33.3388278269,336421.095163 6245739.03126
33.5081358908)),((336421.095163 6245739.03126
33.5081358908,336432.482008 6245732.34738
33.643546748,336434.518318 6245737.07459
33.6392693782,336421.095163 6245739.03126
33.5081358908)),((336435.079412 6245731.96704
33.6739526781,336434.518318 6245737.07459
33.6392693782,336435.782188 6245736.88233
33.6489296532,336435.079412 6245731.96704
33.6739526781)),...)

8      UNSW Hall Courtyard MULTIPOLYGON Z
(((336459.250175 6245697.85406
34.5841691928,336459.781875 6245701.58429
34.5578243334,336436.244393 6245699.46481
33.7292844961,336459.250175 6245697.85406
34.5841691928)),((336458.733753 6245694.11522
34.6301063329,336459.250175 6245697.85406
34.5841691928,336436.244393 6245699.46481
33.7292844961,336458.733753 6245694.11522
34.6301063329)),((336458.172141 6245690.3503
34.7278512743,336436.075456 6245698.30894
33.7332944128,336457.602129 6245686.62673
34.785972674,336458.172141 6245690.3503
34.7278512743)),...)

.....

```



Q10: The same query for the rooms of the Red Centre.

```

1. SELECT DISTINCT b1.ifc_guid,
   b1.ifc_name,
2. ST_AsText(b1.geom) AS geom
3. FROM
4. PUBLIC."Red_Center_H13_IFC_color"
   b1,
5. PUBLIC."Red_Center_H13_IFC_color" b2
6. WHERE b1.ifc_class='IfcSpace'
7. AND b2.ifc_class='IfcSpace'
8. AND b1.ifc_name!=b2.ifc_name
9. AND ST_3DIntersects(b1.geom,
   b2.geom);

ifc_guid ifc_name geom

```

```

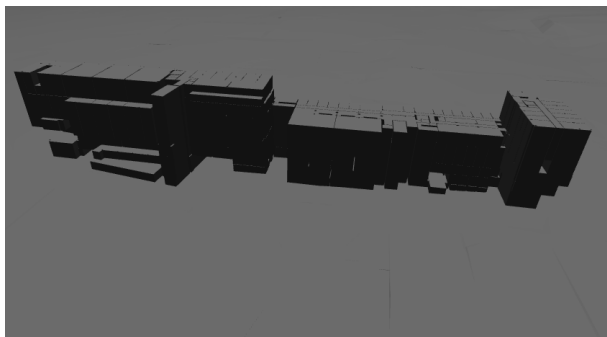
0iQbw7RMD2D0sbJT54o_E$      1036
POLYHEDRALSURFACE Z (((151.230266422 -
33.917881639 37.784338133,151.230265637 -
33.917881683 37.784338133,151.230265637 -
33.917881683 41.194966483,151.230266422 -
33.917881639 37.784338133)),((151.230265637 -
33.917881683 37.784338133,151.230266422 -
33.917881639 37.784338133,151.230295653 -
33.917885348 37.784338133,151.230265637 -
33.917881683 37.784338133)),((151.230265637 -
33.917881683 41.194966483,151.230265637 -
33.917881683 37.784338133,151.230259511 -
33.91791492 41.194966483,151.230265637 -
33.917881683 41.194966483)),...)

0iQbw7RMD2D0sbJT54o_Ea      1030
POLYHEDRALSURFACE Z (((151.230075372 -
33.917856916 37.784338133,151.230067612 -
33.917899024 37.784338133,151.230067612 -
33.917899024 41.194966483,151.230075372 -
33.917856916 37.784338133)),((151.230067612 -
33.917899024 37.784338133,151.230075372 -
33.917856916 37.784338133,151.230104439 -
33.917861081 37.784338133,151.230067612 -
33.917899024 37.784338133)),((151.230067612 -
33.917899024 41.194966483,151.230067612 -
33.917899024 37.784338133,151.23009754 -
33.917902823 41.194966483,151.230067612 -
33.917899024 41.194966483)),...)

0iQbw7RMD2D0sbJT54o_Ei      1031
POLYHEDRALSURFACE Z (((151.230107078 -
33.917861416 37.784338133,151.23010627 -
33.917861313 37.784338133,151.23010627 -
33.917861313 41.194966483,151.230107078 -
33.917861416 37.784338133)),((151.23010627 -
33.917861416 37.784338133,151.230098596 -
33.917902957 37.784338133,151.23010627 -
33.917861313 37.784338133)),((151.23010627 -
33.917861313 41.194966483,151.23010627 -
33.917861313 37.784338133,151.230098596 -
33.917902957 41.194966483,151.23010627 -
33.917861313 41.194966483)),((151.230107078 -
33.917861416 37.784338133,151.23010627 -
33.917861313 41.194966483,151.230107078 -
33.917861416 41.194966483,151.230107078 -
33.917861416 37.784338133)),...)

.....

```







## Discussion and Conclusion

In this report a precinct information model with GIS and BIM data for 3D campus modelling was presented. IFC and CityGML represent BIM and GIS objects of a city. In order to fulfil the demands in urban planning applications and construction analysis, it is important to integrate IFC and CityGML. As it can be seen from the UML diagrams, there is a significant overlapping for information content for mapping IFC and CityGML to the PIM. As intended in this research no new concepts have been created but only enriched and adapted to the specific uses. There are concepts in the current PIM model that are adopted as in IFC, there are others that are similar or adopted from CityGML definitions. The work described here towards defining a PIM schema shows that relatively few extensions need to be added to the existing CityGML and IFC schema in order to address precinct-level

modelling. In addition, optimizations on the object model and its mapping to a relational schema were discussed.

Data visualization results have been shown in the system visualisation report (Aleksandrov et al., 2019).

In the future, the semantics and geometry issues of integration of BIM and GIS will be continued to be further studied. There are many semantics developments within BIM and GIS, but they are within the domains and not across domains. Moreover, there is room for performance optimisations.

## References

- Building SMART International, 2013. Industry Foundation Classes (IFC), IFC4. Available from: <http://www.buildingsmarttech.org/specifications/ifc-releases/summary>.
- ISO. Industry Foundation Classes (IFC) for Data Sharing in the Construction and Facility Management Industries; International Organization for Standardization: Geneva, Switzerland, 2013.
- ISO. Industrial Automation Systems and Integration—Product Data Representation and Exchange; International Organization for Standardization: Geneva, Switzerland, 2014
- Open Geospatial Consortium, 2012. OGC City Geography Markup Language (CityGML) Encoding Standard 2.0.0. Technical report. <https://www.opengeospatial.org/standards/gml>
- Agoub, A., Kunde, F. and Kada, M.A.R.T.I.N., 2016. Potential of graph databases in representing and enriching standardized Geodata. Tagungsband der, 36 ([pdf](#))
- Aleksandrov, M., A. Diakit , J. Yan, W. Li., J. Doig, S. Zlatanova, 2019 Progressing precinct modelling on the UNSW campus and beyond: BIM/PIM and 3D GIS System architecture and Visualisation, CRC Low Carbon Living report, 27p
- Benner, J., Geiger, A. and Leinemann, K., 2005, June. Flexible generation of semantic 3D building models. In Proceedings of the 1st international workshop on next generation 3D city models, Bonn, Germany (pp. 21-22) ([pdf](#))
- Borrmann, A., Kolbe, T.H., Donaubaue, A., Steuer, H., Jubierre, J.R. and Flurl, M., 2015. Multi - scale geometric - semantic modeling of shield tunnels for GIS and BIM applications. Computer - Aided Civil and Infrastructure Engineering, 30(4), pp.263-281 ([pdf](#))
- El-Mekawy, M. and  stman, A., 2010. Semantic mapping: an ontology engineering method for integrating building models in IFC and CityGML. In 3rd ISDE DIGITAL EARTH SUMMIT, 12-14 June ([link](#))
- Foley, J.D., Van, F.D., Van Dam, A., Feiner, S.K., Hughes, J.F., HUGHES, J. and ANGEL, E., 1996. Computer graphics: principles and practice (Vol. 12110). Addison-Wesley Professional ([pdf](#))
- Hatcher, A., 2001. Algebraic topology, 10th ed. Cambridge University Press, Cambridge.
- Herring, J., 2001. The OpenGIS abstract specification, Topic 1: Feature geometry (ISO 19107 Spatial schema), version 5. OGC document, (01-101)
- Kolbe, T.H., Nagel, C. and Stadler, A., 2008. CityGML—A framework for the representation of 3D city models from geometry acquisition to full semantic qualification. In Proc. of ISPRS Congress
- Lapierre, A. and Cote, P., 2007, October. Using Open Web Services for urban data management: A testbed resulting from an OGC initiative for offering standard CAD/GIS/BIM services. In Urban and Regional Data Management. Annual Symposium of the Urban Data Management Society (pp. 381-393) ([pdf](#))
- Newton, P., Marchant, D., Mitchell, J., Plume, J., Seo, S. and Roggema, R., 2013. Performance assessment of urban precinct design. Low Carbon Living CRC, Sydney, NSW ([pdf](#))
- Newton, P., Plume, J., Marchant, D., Mitchell, J. and Ngo, T., 2017. Precinct information modelling: a new digital platform for integrated design, assessment and management of the built environment. In Integrating Information in Built Environments (pp. 111-132). Routledge ([link](#))
- Ordonez, C., Song, I.Y. and Garcia-Alvarado, C., 2010, October. Relational versus non-relational database systems for data warehousing. In Proceedings of the ACM 13th international workshop on Data warehousing and OLAP (pp. 67-68). ACM ([pdf](#))
- Plume, J., Mitchell, J. and Marchant, D., 2017. Peter Newton Precinct Information Modelling. Position Paper. CRC for Low Carbon Living Ltd supported by the Cooperative Research Centres program, an Australian Government initiative.
- Requicha, A.G., 1980. Representations for rigid solids: Theory, methods, and systems. ACM Computing Surveys (CSUR), 12(4), pp.437-464 ([pdf](#))
- Wang, W.P. and Wang, K.K., 1986. Geometric modeling for swept volume of moving solids. IEEE Computer graphics and Applications, 6(12), pp.8-17 ([pdf](#))
- Yao, Z., Nagel, C., Kunde, F., Hudra, G., Willkomm, P., Donaubaue, A., Adolphi, T. and Kolbe, T.H., 2018. 3DCityDB-a 3D geodatabase solution for the management, analysis, and visualization of semantic 3D city models based on CityGML. Open Geospatial Data, Software and Standards, 3(1), p.5 ([pdf](#))

Zlatanova, S., S. Shirowzhan, J. Yan, M. Aleksandrov, A. Diakit , J. Barton, 2019, Progressing Precinct Modelling on the UNSW Campus and Beyond: BIM/PIM and 3D GIS- Data Inventory, CRC Low Carbon Living report, 28p

# Reference

