EARTH OBSERVATION AND GEOMATICS ENGINEERING

website: https://eoge.ut.ac.ir

Introducing a Customized Framework for 3D Spatial Data Infrastructure of Iran Based on OGC Standards

Behnam Alizadehashrafi

Faculty of Multimedia, Tabriz Islamic Art University

Article history:
Received: 4 November 2018, Received in revised form: 11 April 2019, Accepted: 21 April 2019

ABSTRACT

This paper describes a framework for 3D geospatial data infrastructure based on Open Geospatial Consortium (OGC) Standards in Iran, especially Tabriz, as one of the oldest cities with more than 3000 years of history. The external code lists based on local culture, vegetation and heritage landmarks were proposed for indexing 3D city structures, buildings, statues and city furniture of Tabriz. These code lists can be used between different governmental agencies as a communication tool and utilized for indexing in the 3D spatial Database (DB). There are some predefined code lists from Germany as the founder of City Geography Markup Language (CityGML), which can be utilized for Iranian context, along with some defined codes based on local cultures and structures. These code lists can be defined for all street furniture, sculptures, and facade textures in some applications such as city planning, built environment, disaster management, and so forth. The code lists can also be used for the components of the buildings, for instance, the facade of the building in different layers as an entity or multi-patch feature class or implicit geometry such as windows, doors, and backgrounds. In addition, the code list can be used to index city elements and enhance the use of 3D SDI for a variety of privileges from end-users to professionals in the near future for different organizations and management levels. The framework for web-based CityServer3D application has been discussed in this research paper. CityGML, as a standard data exchange format, has been utilized for developing 3D SDI for Iran, and implicit geometry representation has been used to avoid lagging while rendering the 3D models during navigation in the 3D virtual environment. The Implicit Geometry Representation (IGR) has been defined in CityGML as prototypic geometry which can be parameterized for multiple usages (Löwner et al., 2016).

KEYWORDS

Code lists
CityServer3D
3D SDI
CityGML
Interoperability

1. Introduction

3D SDI is normally used to describe metadata relevant to the collection of technologies, technical methods, processes, policies, and institutional arrangements that facilitate the access to 3D spatial data and retrieving information from geospatial DB. A lack of metadata introduces uncertainty into models that are already full of assumptions and estimations (Labetski et al., 2018). In this framework, the available OGC Web service such as Web3D Service has been utilized. Features that have been implemented by (Basanow et al., 2008) seem very appropriate with some modification to suit local requirements in Tabriz. Semantic modeling and Surveying systems like total stations, terrestrial or airborne laser scanners or photo-matching techniques via AgiSoft Photoscan software or even manual methods can be utilized for modeling 3D low poly geometries. The generalization process can be conducted in order to minimize the number of polygons within a 3D geometry to a certain extent along with textures and higher Level of Realism (LoR). Semantic DB of the geometries can be retrieved by queries, using CityGML and CityServer3D. Google Earth 3D models generally consist of geometry, location, and appearance information rather than semantic DB, the meaning of the objects, topology, and environmental datasets. The Building Information Model (BIM) domain focuses on information about the design and construction of building sites and thus has highly detailed information such as blueprints, materials, costs, duration and so on. Meanwhile, GIS represents information about the environment 'as-built' at different points in time, thus

* Corresponding author
E-mail addresses: b.alizadehashrafi@tabriziau.ac.ir (B. Alizadehashrafi)
DOI: 10.22059/eoge.2019.285974.1056

yielding less detailed but regularly updated datasets covering vast regions (Arroyo Ohori et al., 2018). Decision-makers must deal with two very different information scopes and standards from BIM as the micro (building asset) from CityGML as the macro (neighborhood) levels (Zadeh et al., 2019). Several aspects of CityGML along with automatic reconstruction of building information models from uninterpreted 3D models have been studied by (Nagel et al., 2009). CityGML is an Extensible Markup Language (XML) based open data model for storing and exchanging virtual 3D city models. It is an application schema for the Geography Markup Language version 3.1.1 (GML3), the extendible international standard for spatial data exchange issued by the OGC and the International Standards Organization (ISO) Technical Committees (TC 211) (Technical Program Manager [TPM] & Editorial Program Manager [EPM]) (Podobnikar, 2012). The aim of the development of CityGML is to reach a common definition of basic entities, attributes, and relations of a 3D city model. This is especially important concerning the cost-effective sustainable maintenance of 3D city models, allowing the reuse of the same data in different application fields. The 1.0.0 version of CityGML was adopted in August 2008 and CityGML 2.0.0 in late 2011, and the last but not least, CityGML 3.0 in early 2019. CityGML is a standardized information model and data exchange format that puts focus not only on the objects' geometry but also on their semantics, topology, and appearance. Key features of CityGML are:

- Objects may comprise coexisting geometric representations for different levels of detail (Level of Details (LoD) concept).
- Topological relations between objects can be recognized via links between geometries (XLink concept).
- The complexity of the variable in the geometry, semantic and coherent structures can be entirely supported in CityGML (Stadler & Kolbe, 2007).
- Aggregation hierarchies on the part of both geometry and semantics support complex object structures (hierarchical structuring).

BIM can be used as detailed semantic modeling same as CityGMLs' external code lists. It might be employed for enabling interoperability between varieties of applications in the future. BIM-based Integrated Project Delivery might be a common process for managing a project via distributed systems. Urban management tasks such as disaster management, delivery of goods and services, and cityscape visualization are managed by using Geographic Information Systems (GIS) and modeling tools. Some of these tasks, such as response time for a fire management system, require detailed geometric and semantic information about buildings in the form of geospatial information, while tasks such as visualization of the 3D virtual city require less geometric and semantic information. Nowadays, service-oriented architectures are becoming more popular in terms of enabling integration and collaboration over distributed environments. The enhancement of BIM Web Service patterns, proposed by (Isikdag et al., 2010) is called Representational State Transfer ful (RESTful) BIM, which may support interoperability between BIM and CityGML. Unfortunately, until now, there is no complete interoperability and transfer of semantic data and spatial relationships from BIM to the geospatial environment except, to some extent, Feature Manipulation Engine (FME) software that is not highly reliable unless with supervision and manual data entry.

Nowadays, due to the availability of broadband and high-speed internet, service-based architectures are feasible for remote applications to interoperate using standard web interfaces. Services can be based on thin servers (Google Earth)/ thick servers (Google map) along with thicker client-side/ thinner client applications. Furthermore, because of unlimited number of clients, distributed systems, parallel processing and multithreading can be employed for higher efficiency and better response time. A service-orientated system enables loose coupling of applications over the web. Several applications can communicate and interact with each other without having a detailed knowledge of each other's working environment. Each of these data layers involved in a serving form as a data or component or application service is a part of a web service. Service Oriented Architectures (SOA) is a software architecture built upon web services. Although the trend in the software industry is towards enabling application interoperability over web services, the Architecture, Engineering, and Construction (AEC) industry is still not fully benefiting from the service-oriented approaches, as the focus of the industry is still very data integration oriented.

2. CITYGML

2.1. Servers and Clients of CityServer3D

The server stores data in the DB, accessible via a variety of interfaces. In addition to querying the stored data in the CityServer3D DB, it is also possible to query additional DB sources. The CityServer3D uses a meta-model to handle this concept via two components called AdminTool and Viewer. After processing the query, the result can be transferred in various ways to other server's functional unit or clients via different interfaces in 2D or 3D formats. The interoperability of the server must be guaranteed for data exchange securely and confidently. E-commerce can be used on the system based on the server components for authentication and transactions monitoring and recording. The system consists of different layers such as Interface Layer (IL), Converter Layer (CL), Functional Layer (FL), Data Schema Layer (DSL) and DB Management System layer (DBMSL). The DBMSL is based on relational DB concept and Oracle Spatial and DSL composed of GeoDB1, GeoBase2, which can be employed not only for geometry but also for storing thematic and spatial classification of datasets. Controller Unit (CU) controls the interaction between layers. Firstly, a web-based visualization system and secondly, management of the DB for importing and exporting data are integrated for the clients of CityServer3D to manage the contents and access the 3D city model (Figure 1) (Haist & Coors, 2005).

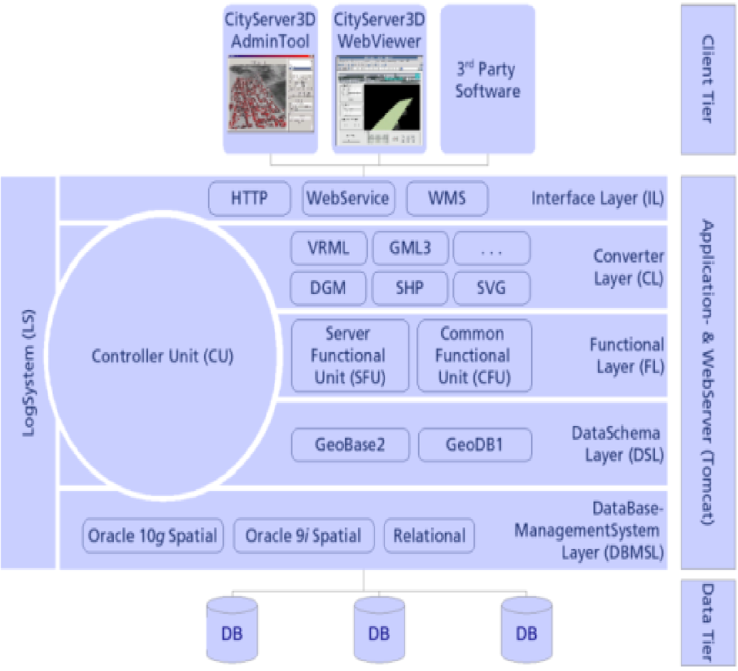


Figure 1. Architecture of CityServer3D (Haist & Coors, 2005)

2.2. Iranian 3D spatial data infrastructure

In order to achieve the integration of those spatial data, applications, and services, a 3D geospatial data infrastructure should be built to connect the existing information "islands." The core component of this data infrastructure is a domain-specific ontology that specifies the knowledge stored in the overall city information model. CityGML is a candidate for such an ontology (Gröger et al., 2012). It was accepted as an OGC standard for the representation, storage, and exchange of 3D city objects and landscape models. For specific domain areas, CityGML also provides an extension mechanism to enrich the data with identifiable features under preservation of semantic interoperability. This extension mechanism is essential to make use of CityGML in various other applications such as flood simulation and energy consumption map. The Industrial Foundation Classes (IFC) was developed from BIM by the International Alliance for Interoperability (IAI) as a well-defined data model for data interchange of BIMs'. The definition of mapping both CityGML and IFC is a challenge that would result in a city information model at all levels from city-wide models to highly detailed BIMs'. The second challenge towards a 3D geospatial data infrastructure is the definition of standard interfaces to access distributed data sources. These interfaces will enable individual access to the relevant information sources for a specific task. The OGC working group 3D Information Management (3DIM) is developing such interfaces to support a framework of data interoperability for the lifecycle of building and infrastructure investment: planning, design, construction, operation, and decommissioning. Web service specifications are needed for visualizing 3D city and landscape models (Schilling & Kolbe, 2010). The Iranian 3D SDI (Ir3DSDI) aims to provide 3D information to many users within the existing Iranian Geospatial Data Infrastructure (IRGDI) framework that finally will be proposed to the OGC and ISO. This IRGDI contains various 2D spatial data and layers for cadastral information in Tabriz, Tehran, and many megacities all over the country in the IT department of municipalities. In the current research project from the IT department of Municipality of Tabriz, this megacity could be part of a "bigger picture" of the proposed Ir3DSDI to serve the community with appropriate interface especially applications like navigation, urban planning, crime map, building management, homeland security, energy consumption map and 3d cadastral information, which is currently missing to some extent.

For instance, in the Tabriz 3D project, the semantic information was defined along with code lists for Iranian 3D SDI within CityServer3D and CityGML. The DB consists of names, roof types, number of the floors under/above the ground level, statues and so on. CityServer3D has a very nice graphical user interface for adding data and attributes, similar to Data Definition Language (DDL) and Data Manipulation Language (DML) in the DB for end-users. The system automatically updates CityGML file based on defined code lists in LoD2 or LoD3. In this research, due to heritage landmarks in the testbed, instead of BIM, CityGML has been used. LoD4 was also tested for School of Multimedia in Tabriz Islamic Art University, along with many statues as a yard art (Figure 2).

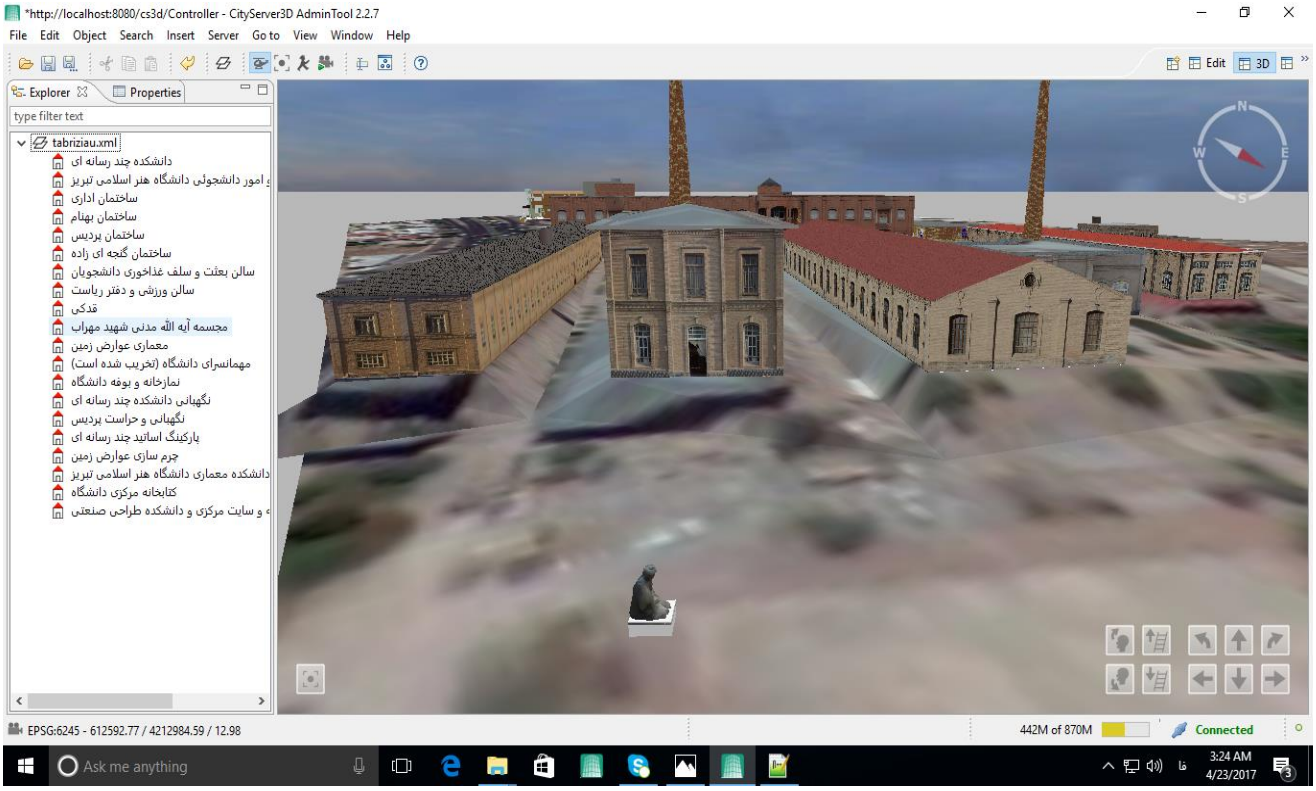

*http://localhost:8080/cs3d/Controller - CityServer3D AdminTool 2.2.7
File Edit Object Search Insert Server Go to View Window Help
Edit
3D
Explorer
Properties
type filter text
tabriziau.xml
دانشکده چند رسانه ای
ی امور دانشجوئی دانشگاه هنر اسلامی تبریز
ساختمان اداری
ساختمان بهنام
ساختمان پردیس
ساختمان گنجه ای زاده
سالن بعثت و سلف غذاخوری دانشجویان
سالن ورزشی و دفتر ریاست
قدکی
مجسمه آیه الله مدنی شهید مهراب
معماری عوارض زمین
مهمانسرای دانشگاه (تخریب شده است)
نمازخانه و بوفه دانشگاه
نگهبانی دانشکده چند رسانه ای
نگهبانی و حراست پردیس
پارکینگ اساتید چند رسانه ای
چرم سازی عوارض زمین
دانشکده معماری دانشگاه هنر اسلامی تبریز
کتابخانه مرکزی دانشگاه
ه و سایت مرکزی و دانشکده طراحی صنعتی
N
W
E
S
EPSG:6245 - 612592.77 / 4212984.59 / 12.98
442M of 870M
Connected
Ask me anything
3:24 AM
4/23/2017

Figure 1. Semantic 3D modeling of Tabriz Islamic Art University in CityServer3D and CityGML based on 3D SDI.

3. A SUSTAINABLE 3D SDI

3.1. Standardized Web Services

Developing 3D data infrastructures along with general SDIs, in particular, is a critical issue to achieve sustainability. Mostly in temporarily limited research projects, the collected data gets lost when the project is finished because of the absence of persons or researchers who know where the data is, how to make use of the data, which formats are available, how to update and maintain, and so forth. To avoid these problems, the Ir3DSDI should be stored in a standardized way and provided by different geographical web services along with documentation and instructions for usage, maintenance and avoiding data loss. With the use of the standards of the OGC, it is possible to gain interoperability, which means the data can not only be used by the web application but can also be accessed with any GIS system, supporting OGC standards. This method guarantees flexibility in further usage of the data beyond project duration. For instance, the web atlas is fed by four different OGC-Web-Services (OWS), listed as following (Auer et al., 2011).

- Web Map Service (WMS) for 2D maps presentation
- Web Feature Service (WFS) for vector data
- Web Processing Service (WPS) for analytical processing
- Web 3D Service (W3DS) for 3D scenes

This can be implemented for Ir3DSDI as well. Before 2010, W3DS was in the state of an OGC discussion paper to become a new standard (Schilling and Kolbe, 2010), and now it is. Ir3DSDI started from Tabriz 3D project, based on these initiatives, in School of Multimedia at Tabriz Islamic Art University.

3.2. Integration of Data Infrastructure Design

Integration of an existing XML document DB and service used and maintained by the historians into the 3D SDI design was a challenging work on the service infrastructure level. This XML DB stores the text-scientific results, transcriptions, catalog of meta-data for the inscriptions, context data about inscription sites and caves and other information. The XML documents about the inscription sites, caves, and the metadata documents have been supplemented with a spatial component in the form of geographic coordinates. This introduces the potential of spatial analysis and visualization of the stored information. As the XML documents do not make use of geographic standard formats, it cannot directly be distributed by OGC Services. To enable a standardized access to the data as well as to support the given structure of the XML DB, an automated conversion step has been integrated which pulls new documents, whenever they are available, from the XML DB and writes them as Simple features into a spatially enabled PostgreSQL12 DB with PostGIS extension (Figure 3).

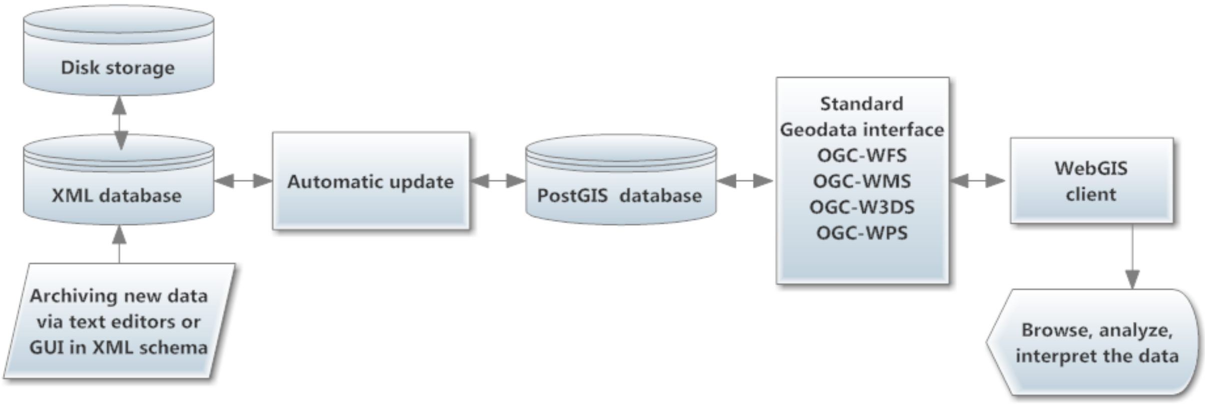

Disk storage
XML database
Archiving new data via text editors or GUI in XML schema
Automatic update
PostGIS database
Standard Geodata interface
OGC-WFS
OGC-WMS
OGC-W3DS
OGC-WPS
WebGIS client
Browse, analyze, interpret the data

Figure 3. Integration of an XML-Document Service into the spatial data infrastructure(Auer et al., 2011).

From this source, the information can be distributed easily by different OGC services to web clients or any other GIS clients. Archiving new XML data is feasible via XML editors along with DB connection and ensures automated propagation update of the PostGIS DB that new features will be automatically shown on the map client and are available for GIS analysis through standardized geo-data interfaces. In this case, the OGC-WMS and OGC-WFS interface is provided by the open-source software *GeoServer*. The OGC-WPS has been realized with PyWPS (implementation of the WPS standard from the OGC, written in Python) using Geographic Resources Analysis Support System GIS (GRASS-GIS) processes behind it while the W3DS is an implementation of the University of Heidelberg (Basanow et al., 2008). The provided data can be received via user-friend Graphic User Interface (GUI) which can convert the data to XML schema before feeding in the form of XML DB.

4. Code lists

In order to represent the object attributes of the city, having an enumerative range of values is necessary, and the concept of dictionaries as provided by CityGML should be used. The values are defined in a file called CityGML_ExternalCodeLists.xml, which comes with the CityGML schema document, but is not a normative part of this schema, since it might be modified, augmented, replaced by users or other communities all over the world based on different local requirements. The actual values in the file CityGML_ExternalCodeLists.xml are a suggestion of the Special Interest Group 3D (SIG3D). In the process of designing semantic and thematic information for 3D city objects, the predefined attributes are necessary to assign for these objects which can be standardized for each area and country such as Iran as a branding and indexing tool. Attributes are employed to classify the objects and recognize them via queries or by clicking and selecting the objects within 3D virtual environment of CityServer3D. It is similar to the indexing system, in which a unique value relates to each attribute within a DB. These values are defined according to the name of the attribute. The files are placed in a directory, which is called *code lists*. The code lists for the predefined objects in Germany can be extracted from the CityGML schema, and the data can be changed, modified, enlarged, and replaced according to the requirements in Iran. These code lists are initially equipped by SIG 3D and defined by OGC in CityGML.

The external code list file defines the value of attributes and assigns a unique identifier to each value. In a CityGML instance document, an attribute value is denoted by an identifier of the value, not by the value itself. These identifiers are known terms for the operators and users. Thus, typos are avoided, and it is ensured that the same concept is denoted in the same way by the same identifier and not by two different terms with identical meanings. This is why the use of code lists facilitates semantic and syntactic interoperability since they define standard terms within an information community.

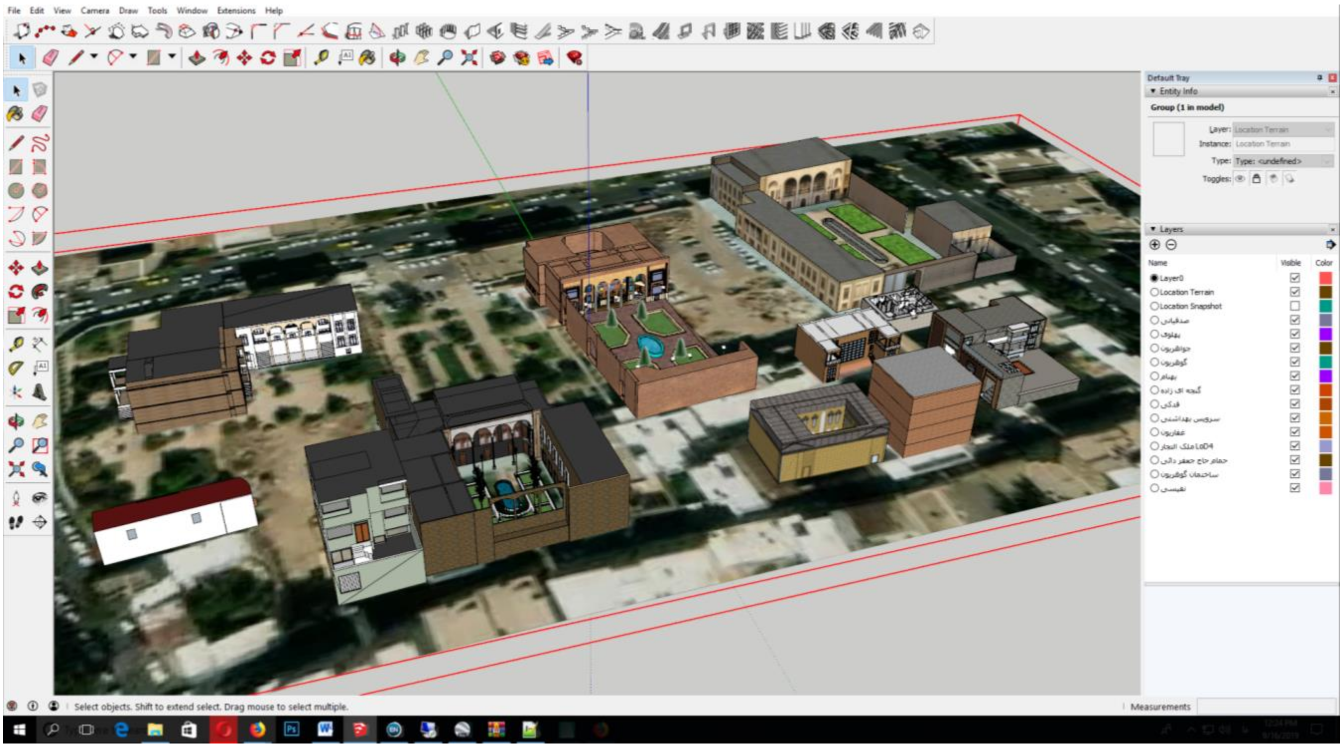
Furthermore, the dictionary concept enables more than one term to be assigned to the same dictionary entry. Thus, the same concept may be explained in different languages. Code spaces are used to differentiate between the languages. An example of an enumerative attribute is *RoofType*, which is defined by the external code list file as “1000” (See the appendix).

‘DefinitionCollection’ element is used in the dictionary concept for representing the values of an attribute, where a Definition entry introduces each value. In CityGML, a definition entry is identified by the name element, which is qualified by the SIG 3D code space. The unqualified name element represents the value of the attribute and an optional description defines the value. CityGML does not use GML identifiers (gml:id) to link to attribute values, since IDs are restricted syntactically, and must be globally unique, which is not feasible for code lists (Gröger et al., 2012). For instance, code lists of the building type can be summarized in the following types, whose valid values are explicitly enumerated for indexing and retrieving the process as fast as possible.

- BuildingClassType
- BuildingFunctionType
- BuildingUsageType
- RoofTypeType
- BuildingInstallationClassType
- BuildingInstallationFunctionType
- BuildingInstallationUsageType
- IntBuildingInstallationClassType
- IntBuildingInstallationFunctionType
- IntBuildingInstallationUsageType
- BuildingFurnitureClassType
- BuildingFurnitureFunctionType
- BuildingFurnitureUsageType
- RoomClassType
- RoomFunctionType
- RoomUsageType

5. Methodology and research scenario

In this research, the testbed is Tabriz Islamic Art University with ten historical landmarks in the School of Architecture and Urbanism (Ganjeizadeh, Ghadaki, Behnam, Nafisi, Ghoharion, Malekottojar, Sadaghiani, Javaherion, Pahlavi, Haj Jafar Daei Bath) (Figure 4) and six old buildings of a historical leather factory of Khosravy that used to be the power supplier of half of Tabriz almost one hundred years ago and since 2005 have been restored and turned into the educational buildings as the Schools of Islamic design, Carpet, Applied Arts and the last but not least Islamic arts (see Figure 2).



Group (1 in model)

Layer: Location Terrain
Instance: Location Terrain
Type: Type: <undefined>
Toggles:

Name	Visible	Color
● Layer0	☑	🟥
○ Location Terrain	☑	🟫
○ Location Snapshot	□	🟩
○ صدقیانی	☑	🟦
○ پهلوی	☑	🟪
○ جواهریون	☑	🟫
○ گوهریون	☑	🟩
○ بهنام	☑	🟪
○ گنجه ای زاده	☑	🟧
○ قدکی	☑	🟧
○ سرویس بهداشتی	☑	🟧
○ غفاریون	☑	🟧
○ ملک التجار LoD4	☑	🟦
○ حمام حاج جعفر دائی	☑	🟫
○ ساختمان گوهریون	☑	🟦
○ نفیسی	☑	🟥

Select objects. Shift to extend select. Drag mouse to select multiple.

Measurements

Figure 4. School of Architecture and Urbanism, Tabriz Islamic Art University.

In addition to these 16 heritage landmarks, more than 25 non-historical buildings where modeled manually via SketchUp and automatic photo-matching applications based on epippolar geometry such as AgiSoft PhotScan version 1.5.0, has been used for modeling 35 sculptures. CityGML plugins for SketchUp 2013 was implemented with some modifications, to convert all geo-referenced 3D models based on World Geodetic System 1984 (WGS84) after grouping each model within the SketchUp (Figure 5).

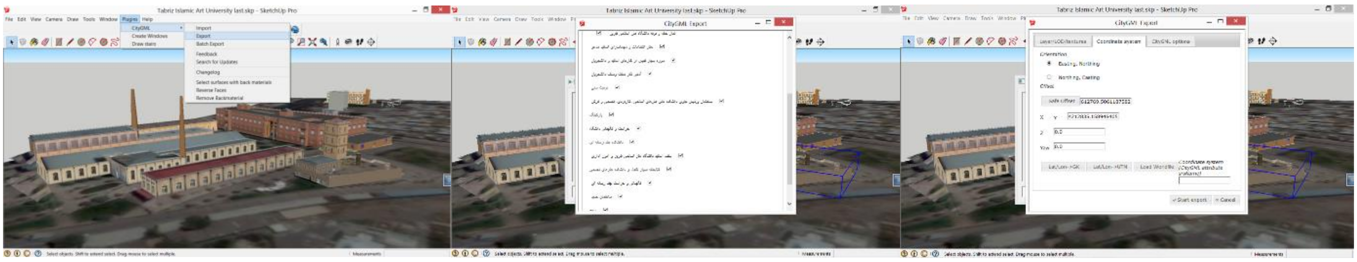


Figure 5. Left to right shows the SKP to CityGML conversion process

The evaluation process of counterclockwise polygons on the model has been conducted manually and oriented within the 3D model to avoid missing faces after the conversion process to CityGML (Figure 6) and finally the 3D models imported to the CityServer3D. CityServer3D has been installed on phpMyAdmin along with MySQL DB. The code lists were defined and assigned to the 3D models within the DB and have been tested via quires and inserting process to check the propagation update procedure. The different steps of the methodology can be seen in the flowchart (Figure 7).

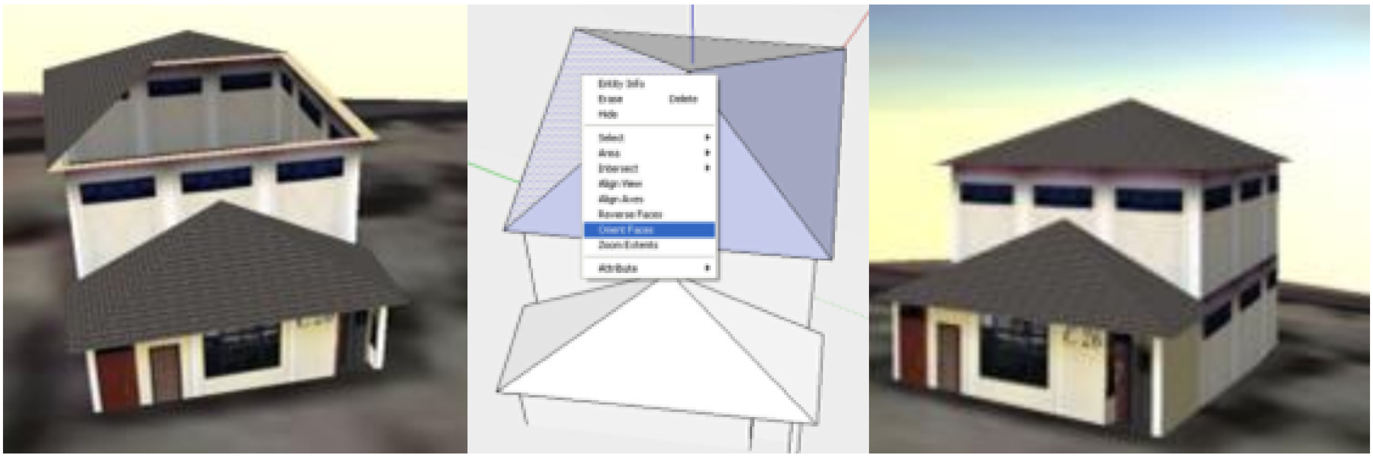
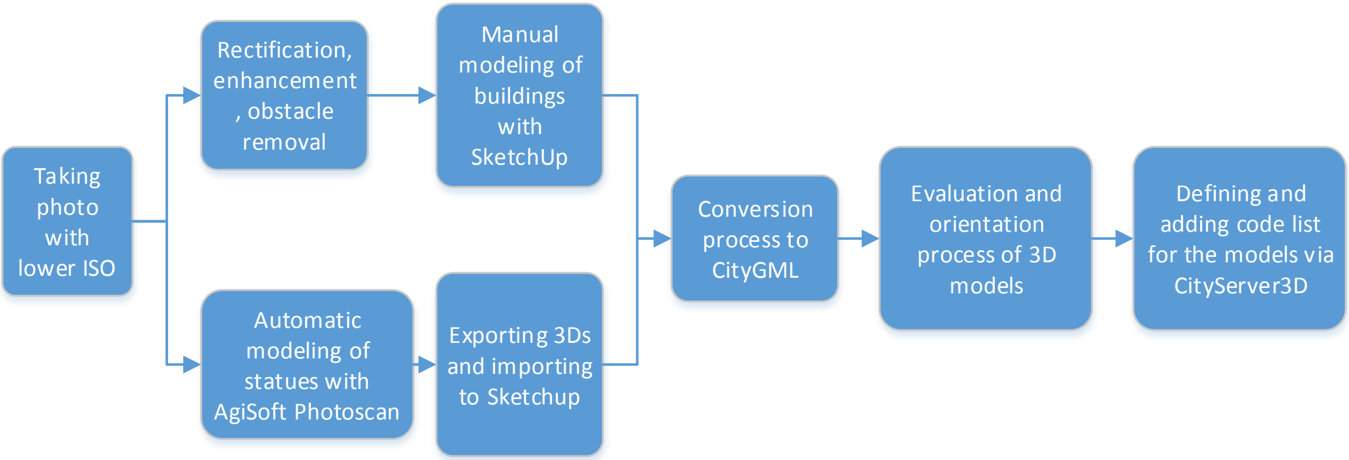


Figure 6. From the left (with missing faces) to the right (oriented), the evaluation and orientation process to avoid missing faces.


Taking photo with lower ISO

Rectification, enhancement, obstacle removal

Manual modeling of buildings with SketchUp

Automatic modeling of statues with AgiSoft Photoscan

Exporting 3Ds and importing to Sketchup

Conversion process to CityGML

Evaluation and orientation process of 3D models

Defining and adding code list for the models via CityServer3D

Figure 7. The flowchart illustrates the steps of the methodology.

6. Implicit geometry in CityGML

CityGML supports implicit geometries, prototypic objects, scene graph concepts such as vegetation model, city furniture, and generic objects. Implicit geometry is a prototypical geometry which can be generated once and reused repeatedly based on nine well-known parameters (3 for rotation, 3 for translation and 3 for scale). The implicit geometry can be an external file in different formats such as VRML, DXF or 3Ds Studio MAX on a local or remote machine and loaded to CityGML scene via URL. Alternatively, it can be defined by a GML3 geometry object (Gröger et al., 2012). This concept can be utilized to represent uniform shapes such as coconut or palm trees, street furniture and so on. It can solve lagging and rendering problems in the visualization and navigation process and also, as a component, can be used to reduce the data size tremendously for web-based applications.

7. Proposed external code lists for Tabriz area

The external code list specifies the attribute value of city objects and gives unique identification. In CityGML, code lists can define most of the city objects separately, which are recommended by SIG3D. The code list is mainly used for the semantic and syntactic interoperability to define the objects in the information field. All of the required code lists are not available in CityGML external code lists, though it covers most of the city objects. Some code lists are proposed according to the suitability for Iran especially in building façade types, based on personal interests on constructions according beauty and price. Some examples of external code lists have been proposed for Iran for different city objects such as rubbish bins, statues, and so forth.

8. Interoperability and future work

Automatic conversion of IFC datasets to geometrically and semantically correct CityGML has been discussed by (Donkers et al., 2016). Actually, different formats can be exported from SketchUp Trimble such as WRL, DAE, 3DS, XML, and KMZ. The 3D model in Collada file format can be utilized via MultiPatch feature class within ArcScene and ArcGlobe. It is also possible to create the MultiPatch feature class automatically via "Arc toolbox -> 3D analyst tools -> conversion -> from file -> import 3D files" (Figure 8). Unlike 3DS, importing the WRL file format into SketchUp, unfortunately, has not been possible until now because of different standards that they are using. This is why the process of converting WRL file formats into SKP file formats can be carried out through 3Ds MAX software. In this case, the WRL file format can be imported to the 3Ds MAX software, and 3DS file format can be exported and then imported into SketchUp. Using FME (Not used in this project) or CityGML plugins for SketchUp 2013, it is possible to convert any 3D models in SketchUp to CityGML. It is better to make a group for each 3D model in SketchUp before exporting the CityGML file. Grouping each model in different layer should be applied to avoid breaking down the 3D building to its components tags within CityGML. The semantic information and code lists have been added via CityServer3D interface manually and updated automatically on the MySQL DB. It is possible to create W3DS Interface via JavaScirpt to visualize CityGML files and use Google spreadsheets for defining, manipulating, updating, and analyzing DB for future works. In this case selecting/hovering the mouse on the 3D models, aggregating and intersecting, creating buffer zone, and many other spatial analyses might be carried out via the web.

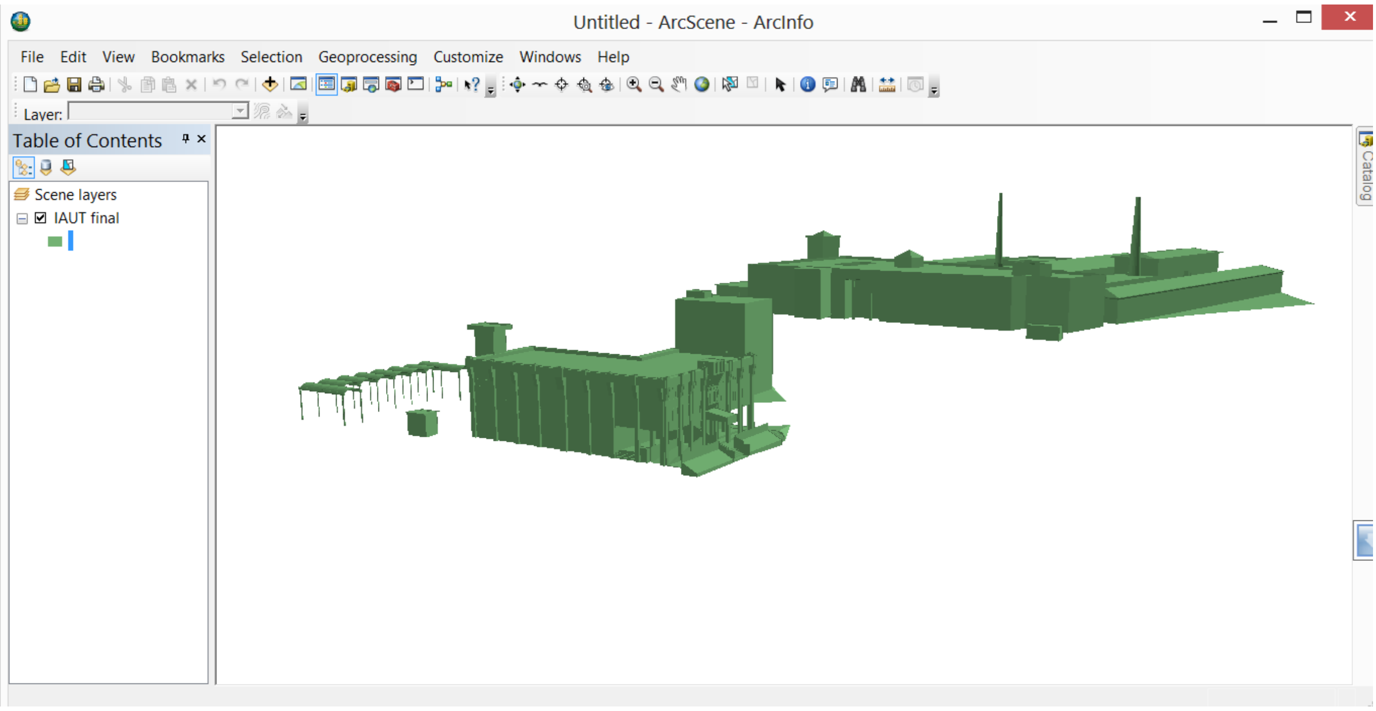

Untitled - ArcScene - ArcInfo
File Edit View Bookmarks Selection Geoprocessing Customize Windows Help
Layer:
Table of Contents
Scene layers
☑ IAUT final
Catalog

Figure 8. Importing 3D models as a MultiPatch feature class to the ArcScene

9. Conclusions

In this paper, the definition of SDI and 3D SDI are clarified, and some reasons are given for using these metadata as standard methods between governmental sections for better interoperability. CityGML and IR3DSDI along with semantic and thematic definition of the objects within a 3D virtual city were discussed in detail via CityGML or geospatial data. Sustainable 3D SDI was addressed, and the methods preventing data losses in the process of interoperability were highlighted as well. The process of conversion between different 3D models and also implicit geometry representation within CityGML was declared. The use of code list was addressed for indexing semantic information. Some code lists were proposed and developed for street furniture and vegetation, based on Iranian culture, environment, and weather in Tabriz Islamic Art University as a testbed infrastructure. The idea of 3D SDI can be further developed and extended for wholly urbanized areas in Iran along with 3D virtual visualization, analysis and 3D cadastral data for municipality.

Appendix

```
<gml:DefinitionCollection gml:id="id356">
 <gml:description>Codelist derived from German
authoritative standards ALKIS/ATKIS (www.adv-
online.de) by S. Schlueter and U.
Gruber</gml:description>
 <gml:name>RoofTypeType</gml:name>
   <gml:definitionMember>
     <gml:Definition gml:id="id357">
       <gml:description></gml:description>
       <gml:name
codeSpace="urn:d_nrw_sig3d">1000</gml:name>
       <gml:name>flat roof</gml:name>
     </gml:Definition>
   </gml:definitionMember>
   <gml:definitionMember>
     <gml:Definition gml:id="id358">
       <gml:description></gml:description>
       <gml:name
codeSpace="urn:d_nrw_sig3d">1010</gml:name>
       <gml:name>monopitch roof</gml:name>
     </gml:Definition>
   </gml:definitionMember>
 ….
</gml:DefinitionCollection>
```

References

ARROYO OHORI, K., DIAKITÉ, A., KRIJNEN, T., LEDOUX, H. & STOTER, J. 2018. Processing BIM and GIS models in practice: experiences and recommendations from a GeoBIM project in the Netherlands. ISPRS International Journal of Geo-Information, 7, 311.

AUER, M., HÖFLE, B., LANIG, S., SCHILLING, A. & ZIPF, A. 3D-Sutras: A web based atlas of laser scanned Buddhist stone inscriptions in China. International Conference on Geographic Information Science, Utrecht, The Netherlands, 2011.

BASANOW, J., NEIS, P., NEUBAUER, S., SCHILLING, A. & ZIPF, A. 2008. Towards 3D spatial data infrastructures (3D-SDI) based on open standards—experiences, results and future issues. Advances in 3D geoinformation systems. Springer.

DONKERS, S., LEDOUX, H., ZHAO, J. & STOTER, J. 2016. Automatic conversion of IFC datasets to geometrically and semantically correct CityGML LOD3 buildings. Transactions in GIS, 20, 547-569.

GRÖGER, G., KOLBE, T. H., NAGEL, C. & HÄFELE, K.-H. 2012. OGC city geography markup language (CityGML) encoding standard.

HAIST, J. & COORS, V. The W3DS-Interface of Cityserver3D. European Spatial Data Research (EuroSDR) ua: Next Generation 3D City Models. Workshop Papers: Participant's Edition, 2005. 63-67.

ISIKDAG, U., UNDERWOOD, J., KURUOGLU, M. & ABDUL-RAHMAN, A. 2010. Geospatial views for RESTful BIM. Handbook of Research on Building Information Modeling and Construction Informatics: Concepts and Technologies. IGI Global.

LABETSKI, A., KUMAR, K., LEDOUX, H. & STOTER, J. 2018. A metadata ADE for CityGML. Open Geospatial Data, Software and Standards, 3, 16.

LÖWNER, M.-O., GRÖGER, G., BENNER, J., BILJECKI, F. & NAGEL, C. 2016. PROPOSAL FOR A NEW LOD AND MULTI-REPRESENTATION CONCEPT FOR CITYGML. ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences, 4.

NAGEL, C., STADLER, A. & KOLBE, T. H. Conceptual requirements for the automatic reconstruction of building information models from uninterpreted 3D models. Proceedings of the Academic Track of the Geoweb 2009-3D Cityscapes Conference in Vancouver, Canada, 27-31 July 2009, 2009.

PODOBNIKAR, T. 2012. Universal Ontology of Geographic Space: Semantic Enrichment for Spatial Data: Semantic Enrichment for Spatial Data, IGI Global.

SCHILLING, A. & KOLBE, T. H. 2010. Draft for Candidate OpenGIS® Web 3D Service Interface Standard. OpenGeospatial Consortium.

STADLER, A. & KOLBE, T. H. Spatio-semantic coherence in the integration of 3D city models. Proceedings of the 5th International ISPRS Symposium on Spatial Data Quality ISSDQ 2007 in Enschede, The Netherlands, 13-15 June 2007, 2007.

ZADEH, P. A., WEI, L., DEE, A., POTTINGER, R. & STAUB-FRENCH, S. 2019. BIM-CITYGML data integration for modern urban challenges. Journal of Information Technology in Construction (ITcon), 24, 318-340.