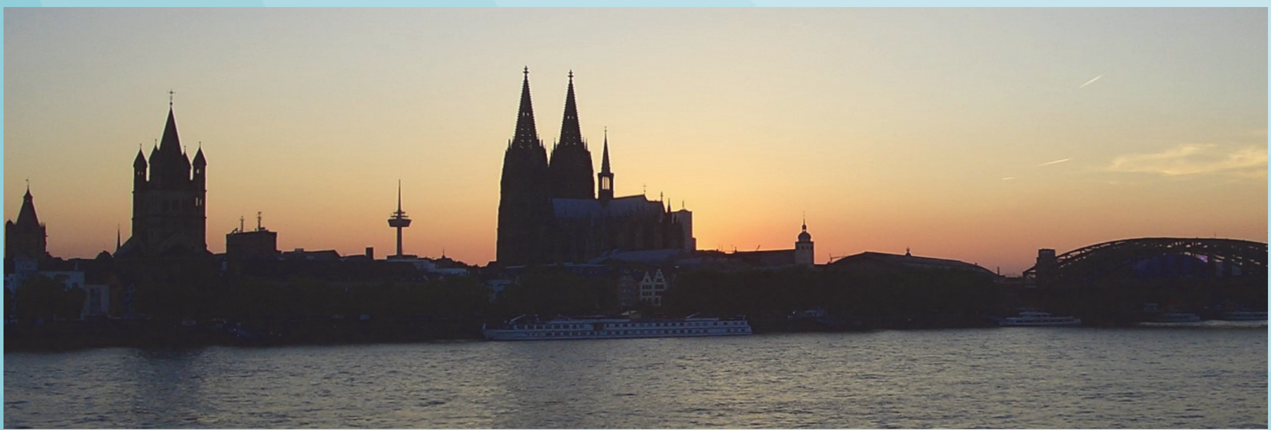
Publikationen der Deutschen Gesellschaft für Photogrammetrie, Fernerkundung und Geoinformation e.V.

DGPF

Band 24 **2015**



50° 55′ 39″ N
6° 56′ 10″ E
Geographisches Institut
Universität zu Köln

Universität zu Köln

Vorträge

35. Wissenschaftlich-Technische Jahrestagung der DGPF

16. – 18. März 2015
in Köln

Bridging Scales -
Skalenübergreifende Nah- und Fernerkundungsmethoden

ISSN 0942-2870

Thomas P. Kersten, Hrsg.

ISSN 0942-2870

Publikationen der
Deutschen Gesellschaft für Photogrammetrie, Fernerkundung
und Geoinformation (DGPF) e.V.
Band 24 421 S., Hamburg 2015
Hrsg.: Thomas P. Kersten

Zu beziehen durch:

Geschäftsstelle der DGPF
c/o Technische Universität München
Institut für Geodäsie, GIS und Landmanagement
Lehrstuhl für Geoinformatik
Arcisstraße 21
D-80333 München
Tel.: 089 289-22578, E-Mail: geschaeftsstelle@dgpf.de

Redaktion:

Thomas P. Kersten
HafenCity Universität Hamburg
Labor für Photogrammetrie & Laserscanning
Überseeallee 16, 20457 Hamburg
E-Mail: Thomas.Kersten@hcu-hamburg.de

Web-based Exploration of and Interaction with Large and Deeply Structured Semantic 3D City Models using HTML5 and WebGL

KANISHK CHATURVEDI[1], ZHIHANG YAO[2] & THOMAS H. KOLBE[1]

Abstract: The aim of the presented work is to present a framework and implementation of a web-based 3D client for processing, visualization and analysis of very large semantic 3D city models. In addition, the framework supports rich interaction with deeply nested structures of the 3D city objects. To improve the performance and user-experience on the web environment, the proposed framework utilizes HTML5 and WebGL allowing the application to be cross-browser, cross-platform and plugin-independent. For visualization purposes, WebGL based Cesium virtual globe has been used and extended in this work to support visualization, exploration and interaction with large and deeply structured semantic 3D city objects.

1 Introduction

Semantic 3D city models describe the urban topography by decomposing and classifying the occupied physical space according to a semantic data model. The relevant real world entities are represented by objects with thematic and spatial attributes and interrelationships to other objects. Today, more and more cities worldwide are representing their 3D city models according to the CityGML standard issued by the OGC. CityGML allows to further decompose complex objects like buildings into their parts like walls, stairs, etc. and these may again consist of parts like windows or doors (GRÖGER et al. 2012). Since CityGML objects can have attributes and relations on all levels of this aggregation hierarchy, the exploration of, querying on and interaction with such a 3D city model must take into account these deeply nested structures.

A plethora of software have been developed for processing and visualizing CityGML data, for example, Safe Software's FME, Bentley Map or Autodesk's InfraWorks etc. However, these software are required to be installed on the user machine in order to work with CityGML file. The visualization of and especially interaction with complex 3D city models represented in CityGML on the web is still a challenging area. In this paper, we present a framework and implementation of a web-based 3D client for processing, visualization and analysis of very large semantic 3D city models and at the same time, supporting complex interactions utilizing HTML5 and WebGL.

HTML5 (W3C RECOMMENDATION 2014) is an Open Standard format and provides common platform for applications to be developed and used on the web. HTML5 enables the new generation browsers to support multi-threading, which allows to perform parallel execution of different tasks within one web page. WebGL (MARRIN 2013) is an extension of HTML5 canvas element, which is now widely used for developing web applications requiring 3D visualization. Applying such an approach, 3D capabilities have been realized directly in all major web browsers running on all major operating systems without needing additional plug-ins or extensions. Another benefit of WebGL is that it utilizes hardware's graphics card memory for displaying and performing operations on 3D contents and hence, it provides hardware accelerated 3D functionality on the

[1] Technische Universität München, Lehrstuhl für Geoinformatik, Arcisstraße 21, 80333 München; E-Mail: [kanishk.chaturvedi, zhihang.yao, thomas.kolbe]@tum.de

web (TARALDSVIK 2011). In our work, Cesium virtual globe (ANALYTICS GRAPHICS INC. 2015) has been used as a visualization engine. Cesium is an Open Source JavaScript package supporting the presentation of 3D contents within the web browser where users can dynamically switch between 3D globe visualization and 2D map projection. It utilizes WebGL to provide hardware acceleration and plugin independence and provides cross-platform and cross-browser functionality.

In this paper, after a brief problem statement, we present our web-based 3D client and the overall architecture behind the 3D web client. We also explain the new functionalities developed using Cesium package to support visualization, exploration and interaction with large and deeply structured semantic 3D city objects. Along with, we demonstrate some example scenarios using the 3D web client.

2 Problem statement

2.1 Interaction with large and deeply structured semantic 3D city models

The management and interaction with deeply structured semantic 3D city models is an important problem. CityGML comprise the spatial and graphical aspects of the city objects along with the ontological structure including thematic classes, attributes, and their interrelationships. Objects are decomposed into parts on the basis of deeply nested structures that can be observed in the real world. For example, a building will be decomposed into different (main) building parts like walls, stairs, etc. and these may again consist of parts like windows or doors (KOLBE 2009). Since CityGML objects can have attributes and relations on all levels of this aggregation hierarchy, the exploration of, querying on and interaction with such deeply nested structures of the 3D city objects must be taken into account.

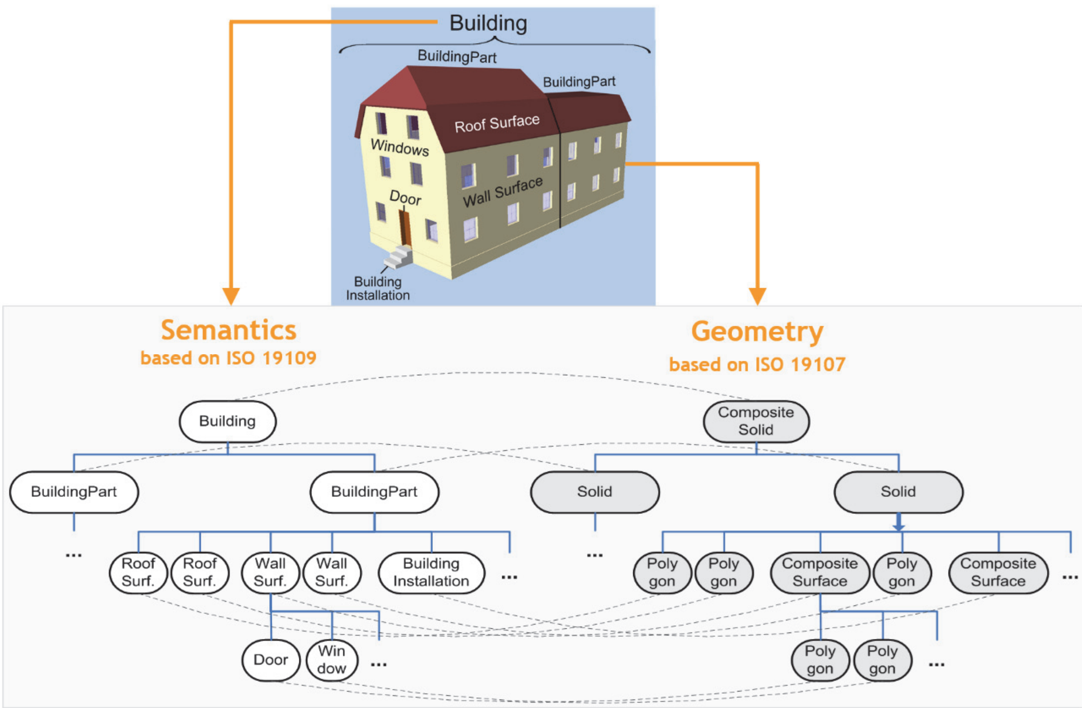


Figure 1: CityGML building object with aggregation hierarchy (Stadler & Kolbe 2007)

Another important problem is that the CityGML documents containing the entire semantic 3D city model are very large in size. This results a significant performance issues in their management and visualization in especially web based applications. In order to achieve efficient visualization, it is important to develop tiling strategy.

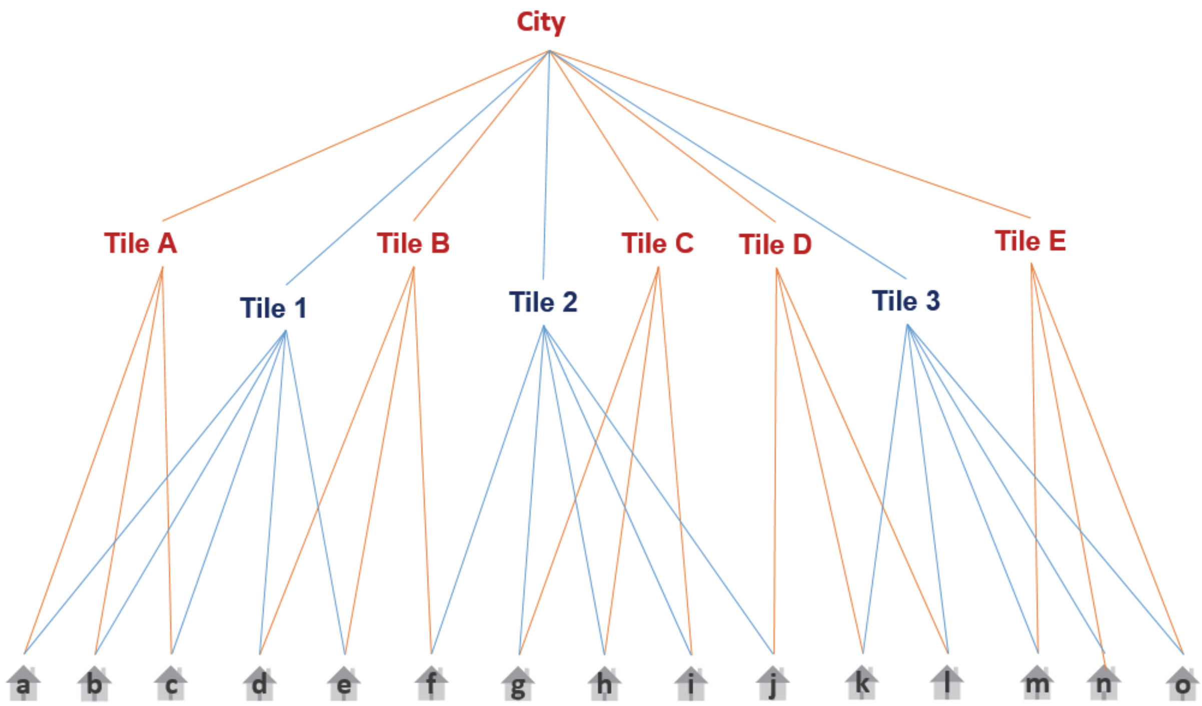


Figure 2: Tiling within city model.

As shown in the Fig. 2, the entire city model is composed of several tiles. These tiles can be based upon the requirements of the user and each tile is composed of specific number of buildings along with thematic and texture information. Due to the possibility of large number of tiles, the efficient management of such tiles is a major challenge in any web based application. The data tiles are required to be loaded or unloaded according to the vicinity of the camera, which improves the overall performance and better user experience. It is also important to provide rich interaction with city objects in the form of highlighting or picking the objects on mouse over and mouse click. Due to a large size of the entire city model and large number of tiles, the performance and rich interaction in combination is a significant challenge.

3 Architecture for the 3D city model web application

In order to solve the problems mentioned in the previous section, the multi-level system architecture has been established allowing different user groups with different backgrounds to easily interact with large and deeply structured semantic 3D city models on the same interface. The three levels of architecture are information backbone, application level and end users (Fig. 3).

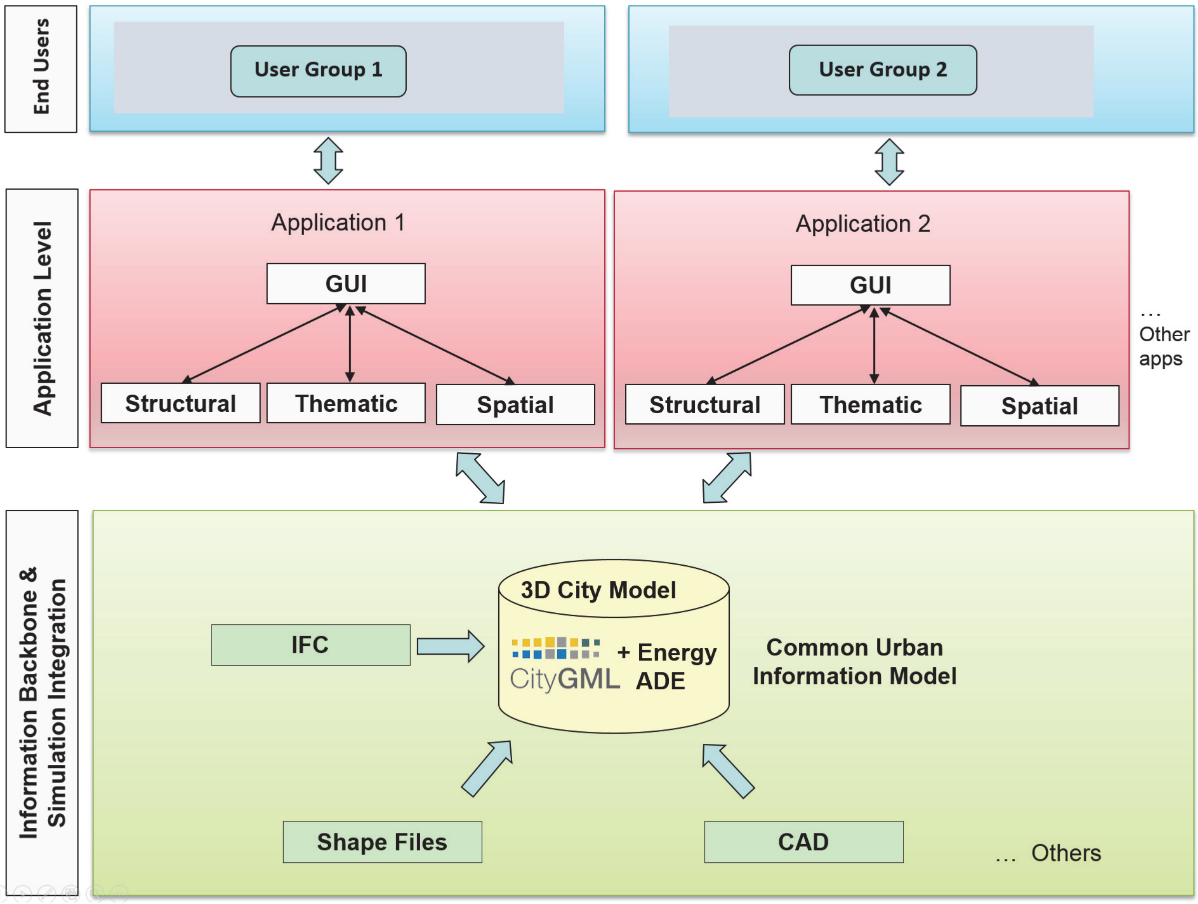


Figure 3: Multi-level architecture for the 3D city model web application

On the lowermost level (Information backbone), the semantic 3D city models are stored which may contain the city level information from different sources such as Industry Foundation Classes (IFC), ESRI shapefiles, CAD files and others. The city information from different sources can be integrated into the common data model and exchange format according to the CityGML standard. The common city information model is integrated into a shared spatial database, which allows efficient management of the large 3D city models which can be accessed by different application domains. The application level acts as a "bridge" between the end users and the information backbone. Typically, different applications require different information from the 3D city model. Also, the analysis functions vary depending on the application. In order to provide tailored and useful application to the specific end-users, it is useful to make clear at the application level about what relevant information is extracted from the city model. The relevant information can be mapped into spatial, thematic and structural information. The application level contains the user friendly graphical interface that helps users to access to those spatial, thematic and structural information of the semantic 3D city models. CityGML standard is a data exchange format and is not suitable for visualization purposes due to its complex structure which contains not only the spatial information but also the semantic and topological information. Due to this reason, the spatial information including large number of tiles is exported in this framework for efficient visualization. However, during the process, the structure information such as level of the aggregation hierarchy and attribute information are lost. Thus, it is required to export the semantic and structural information along with the spatial information. The semantic information can be a simplified data structure having the attribute level information. The structural information contain not only the different aggregation hierarchy but also the meta-information of the exported city objects. All of these information are logically connected by a common identifier. At the top level, there may be different groups of end users, working on the specific problems on the 3D city models. These groups may, for example, be housing companies, working on specific decision support for investment in construction or may also be the energy suppliers reporting of renewable energy production possibilities.

4 Implementation

4.1 3D city database

The 3D city database (3DCityDB) is an open source geodatabase schema containing a set of tools for management and visualization of large 3D city models. It includes a comprehensive set of documents and SQL scripts which allow to import the CityGML datasets in the database. Furthermore, it includes a separate Java frontend application named 3DCityDB Importer/Exporter which allow for high performance importing and exporting CityGML dataset. It also allows to export the contents in the form of KML/COLLADA format allowing it to be viewed and interactively explored in the 3D viewers such as Cesium or Google Earth. It also allows to extend the functionalities in a modular way with installation of plugins, which adds specific abilities to interact with the 3D city database. For instance, by using the Spreadsheet Generator Plugin, the arbitrary subsets of the city model data can be exported in tabular form having the selected attributes from 3D city database instance whether as a CSV file or directly be uploaded as a Google Spreadsheet Document using the corresponding Google Cloud Service (Google Docs) (Fig. 4).

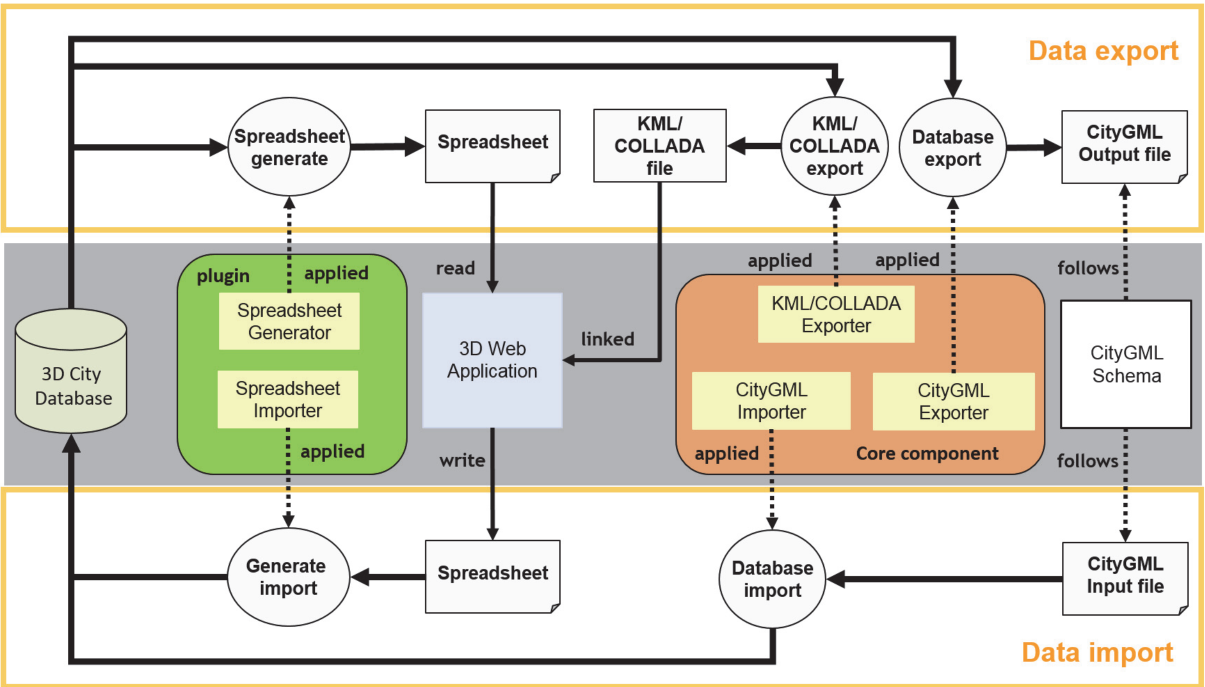


Figure 4: 3D City Database software suit (Yao et al. 2014)

4.2 Cesium

Cesium provides a higher level of abstraction, making it a highly easy-to-use tool. Its architecture includes four main layers as shown in Fig. 5:

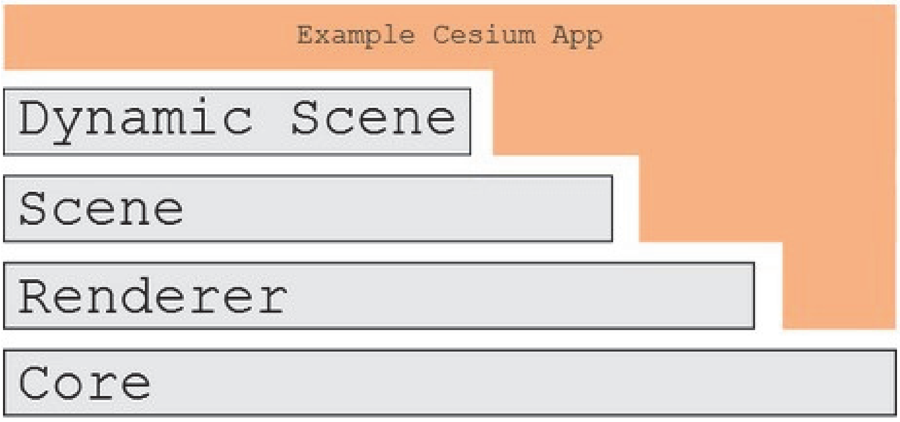


Figure 5: Cesium architecture (Analytics Graphics Inc. 2015)

The different layers of the architecture are responsible to add specific functionality and to raise the level of abstraction. The layer is usually dependent on the layers underneath it. The details are:

- Core – This is the lowest layer in Cesium and includes mainly low-level functions. These functions majorly include computations and calculations such as mathematical conversions, transformations and projections.
- Renderer – This layer is a thin abstraction over WebGL. It includes already available GLSL functions to provide shader programs, textures and buffers.
- Scene – This layer is mainly responsible to provide overall functionality of the globe. It includes high-level globe and map constructs such as 3D globe or map, handling layer imageries from multiple sources, creation of geometries and materials, camera control and animation.
- Dynamic Scene – This is the top-most layer of Cesium, which provides dynamic visualization of the data with the help of its in-built language CZML. It allows to store the data in dynamic objects, loads and renders the dynamic objects altogether instead of rendering every frame.

The most distinguishing feature of Cesium are:

- It is most suitable for dynamic geospatial data visualization with the help of Cesium Language (CZML). CZML is a JSON based schema, which describes geospatial data along with their properties that vary over time.
- It can integrate layer imageries from different sources, such as OpenStreetMap, Bing Maps, ArcGIS MapServer and standard image files. Even, the external WMS and TMS can be integrated. Each layer, then, can be visualized according to specific brightness, contrast or saturation.
- It includes extensive libraries which support 2D as well as 3D geometries. The user can draw polyline, polygon, ellipsoid, sphere, labels, billboards and sensors.

- It supports data imports from KML, ESRI Shapefiles and JSON.
- It includes handlers to control mouse/keyboard events, camera movements and zoom and pan the virtual globe.
- It supports extensive materials to describe the surface appearance of the objects. It also supports custom materials for the objects.
- It supports math libraries to support major reference frames such as World Geodetic System (WGS84) and International Celestial Reference Frame (ICRF). The libraries also support conversions of coordinates and Cartesians.

However, Cesium does not provide direct support of CityGML data. Hence, as part of pre-processing, CityGML is converted into KML format using 3D city database, which is used for visualization on the Cesium. Along with, Cesium does not support dynamic loading of portions from large 3D city models yet. In our work, existing functionalities of Cesium virtual globe have been extended to support the tiling of large 3D city models and dynamic loading of the tiles.

4.3 3D web client

The web based 3D client has been developed which act as a user interface to the end user to perform the functions such as interactive 3D visualization and exploration. The client is web based which can be accessed using the web browser without having to install any software locally. It is a static application based on ExtJS JavaScript-based web framework and can be operated with any web server like Apache without the need of an application server. The Cesium API has been used as a visualization engine which enables to visualize the graphical representation of the 3D building models. Furthermore, user can control the dynamic elements of 3D building models using JavaScript commands embedded within Cesium JavaScript API (see Fig. 6).
The highlighted features of the 3D web client are:

- **Data exploration**: Cesium virtual globe does not provide support of dynamic loading of portions from large 3D city models yet. In our work, Cesium has been extended to support the tiling of large 3D city models and dynamic loading of the tiles. The web client has been implemented on top of the extended Cesium virtual globe to process and visualize the large 3D city models with different level of details. Utilizing multi-threading capabilities of HTML5, the time-costly operations such as parsing of multiple 3D objects are delegated to a background thread running in parallel. At the same time, another thread monitors the interactions with the virtual camera and takes care of loading and unloading the data tiles according to their visibility. That improves the user interaction with the 3D city models on the virtual globe.
- **Managing interaction**: The developed functionality allows to select one or more objects and display their attribute information in the tabular form. To provide rich user-interaction, the highlighting can also be enabled while selecting the specific objects.
- **Interaction with different aggregation levels**: Another outstanding feature in this work is to manage interaction with deeply structured 3D city models. It is a major problem in conjunction with tiling of visualization model as its structure information such as level of the aggregation hierarchy and attribute information are lost. Under this study, besides visualization model, the web client also allows to display extra structure information and thematic information from the CityGML data repository. To achieve the same, not only a visualization model (in KML/COLLADA) is exported, but also additional JSON encoded data about the decomposition of all complex city objects like buildings and thematic

information in the tabular form are also exported. This is then exploited by the web client to expose the different aggregation levels to the user, resulting in a better user interaction with complex structured 3D city models.

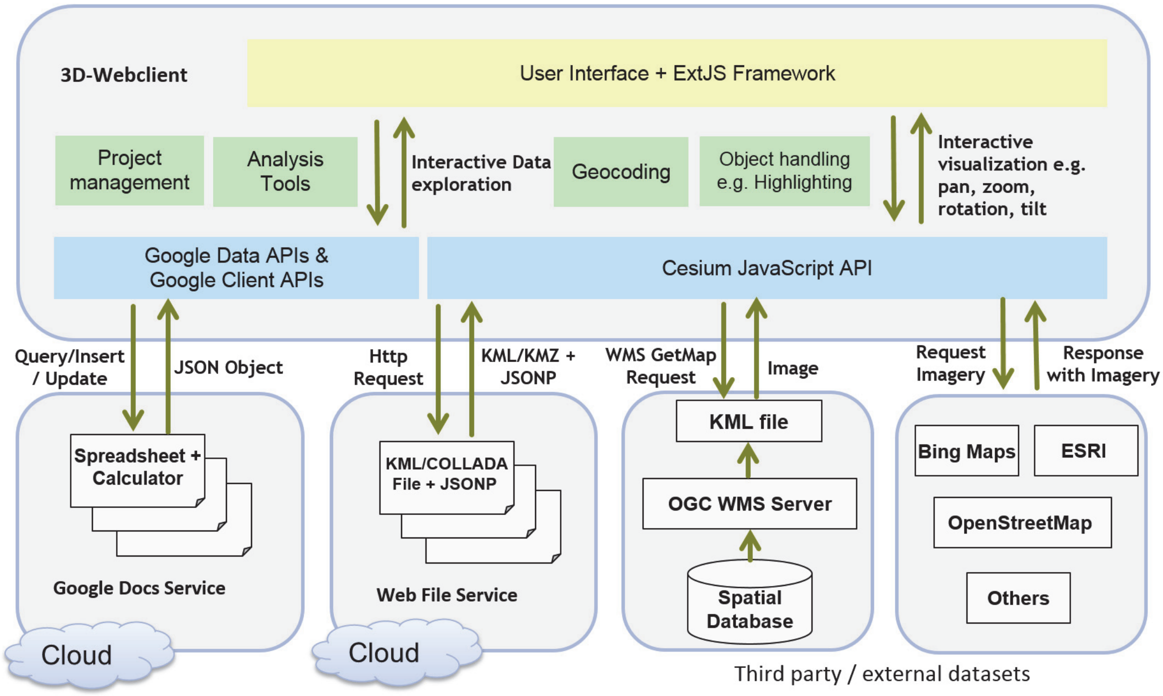


Figure 6: Structure of the 3D web client

- **Query and analysis**: The city objects can be queried on the basis of their attributes. Furthermore, for a group of selected city objects, the aggregation functions such as sums, averages, minimum, or maximum values of numeric attributes can be calculated directly in the 3D Web client, and results are displayed in the Attribute List and the resulting objects are highlighted. The property information of the selected city objects may include thematic information which can further be exported in an HTML table as a report. With the presented functionality, the application range of the 3D Web client varies from the pure visualization of 3D geometries to the exploration of attribute data of the 3D City models in the form of tables to the calculation of results of complex analysis and simulation of the semantic city models.

5 Use cases

The datasets for New York (LOD1) and London (LOD2) have been used for demonstration purposes of our application:

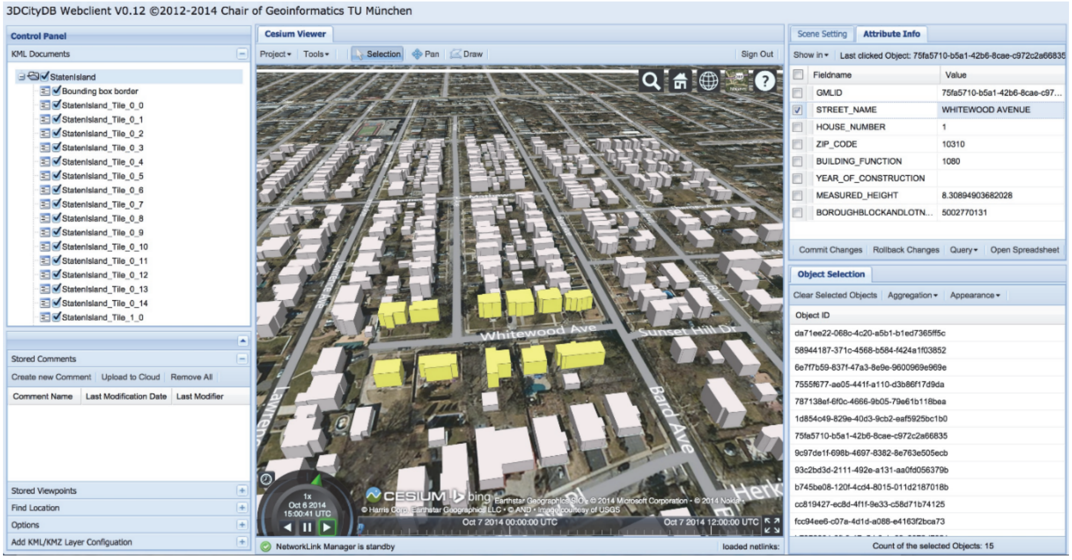


Figure 7: 3D web client displaying different tiles with spatial and semantic information of New York (LOD1) dataset

As shown in the Fig. 7, the interactive exploration of LOD1 building objects from New York data set are displayed. The entire dataset is composed of several tiles which are dynamically loaded according the vicinity of the virtual camera. The highlighting of the building objects provide better user interaction with the objects and by clicking on it, the semantic details of the city object are retrieved from the Google spreadsheet and displayed in the simple tabular format in the top-right panel. Similarly, in the Fig. 8, the LOD2 building objects of London data are displayed in the similar fashion.

Figure 8: 3D web client displaying different tiles with spatial information of London (LOD2) dateset

The web client also allows to query the building objects based on attribute information or aggregation functions such as sum, average, minimum or maximum. As shown in the Fig. 7, the objects are queried on the basis of the buildings located on a particular street. Similarly, in the Fig. 9, all the buildings are queried on the basis of a specific building function.

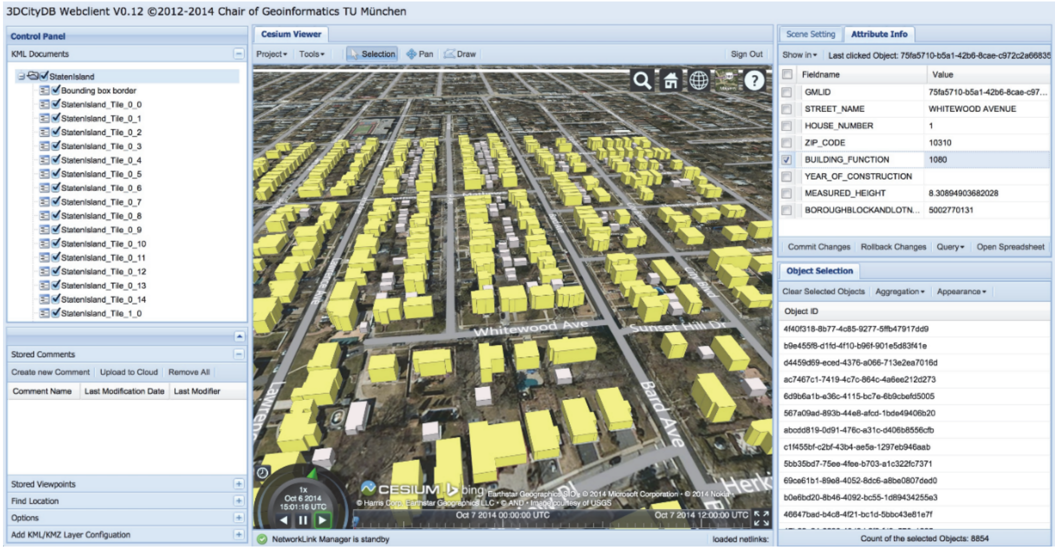


Figure 9: Example of querying on the basis of Building function

As shown in the Fig. 10, the user can interact with different aggregation levels of the specific building object. The information about the decomposition of the building object such as wall and roof surfaces are stored in the JSON encoded format. Utilizing that, the user is able to interact with the specific aggregation level of the city object.

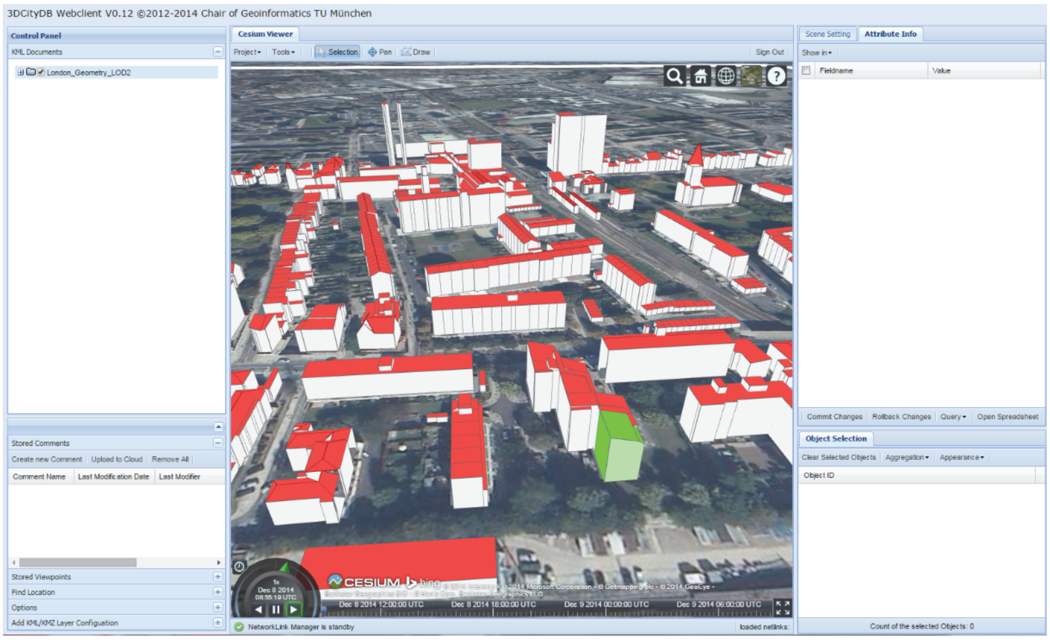


Figure 10: 3D web client displaying extra structure information of LOD2 building of London dataset

6 Conclusion

Semantic 3D city models are very large in size and complex in their structure. The OGC standard CityGML is widely used for representation and exchange of semantic 3D city models. Although a good amount of software have been developed to work with CityGML datasets, the web based solutions to visualize the CityGML documents require more attention. Our work has focused on web based framework to enable exploration and interaction with 3D city objects within CityGML documents. The web based 3D client has been developed on top of WebGL based Cesium virtual globe, which provides plug-in independence and hardware acceleration, resulting in improved and seamless performance. The tiling strategy has been implemented to explore large size semantic 3D city model with a better performance. In addition, the 3D client also allows to interact with different aggregation levels of the deeply structured city model. It enables the user to retrieve and interact with various sub-parts of the building. In this work, the datasets of New York (LOD1) and London (LOD2) have been used. However, in future, it is also intended to work with datasets with higher level of details. CityGML allows to extend the city models by addition of new properties specific to an application in the form of Application Domain Extensions (ADE). It is also recommended that such extendible city models should also be supported by the web client. In the end, it is also proposed to support the dynamic or time varying properties within city models by the web client. The web client is based on Cesium which allows the support of such time varying elements with the help of Cesium language CZML.

7 References

ANALYTICS GRAPHICS INC., 2015: Cesium - WebGL Virtual Globe and Map Engine. Retrieved January 17, 2015, from http://cesiumjs.org/

GRÖGER, G., KOLBE, T. H., NAGEL, C. & HÄFELE, K.-H., 2012: OGC City Geography Markup Language (CityGML) En-coding Standard. Open Geospatial Consortium Inc, OGC. Retrieved from http://www.opengeospatial.org/standards/citygml

KOLBE, T. H., 2009: Representing and exchanging 3D city models with CityGML. 3D geo-information sciences, Springer, S. 15-31. Retrieved from http://link.springer.com/chapter/10.1007/978-3-540-87395-2_2

MARRIN, C., 2013: WebGL Specification. Khronos WebGL Working Group. Retrieved January 17, 2015, from https://www.khronos.org/registry/webgl/specs/1.0/

STADLER, A. & KOLBE, T. H., 2007: Spatio-semantic coherence in the integration of 3D city models. Proceedings of the 5th International Symposium on Spatial Data Quality, Enschede. Retrieved from http://misc.gis.tu-berlin.de/igg/htdocs-kw/fileadmin/citygml/docs/SDQ2007_Stadler_Kolbe.pdf

TARALDSVIK, M., 2011: Exploring the Future: is HTML5 the solution for GIS Applications on the World Wide Web? Technical Report, NTNU. Retrieved from http://mats.taraldsvik.net/gd12/publications/html5andgis.pdf

W3C RECOMMENDATION, 2014: HTML5. Retrieved Jan. 17, 2015, from http://www.w3.org/TR/html5/

YAO, Z., SINDRAM, M., KADEN, R. & KOLBE, T. H., 2014: Cloud-basierter 3D-Webclient zur kollaborativen Planung energetischer Maßnahmen am Beispiel von Berlin und London. Geoinformationssysteme, Proceedings of the Münchner GI-Runde 2014, Kolbe, T., Bill, R. & Donaubauer, A. (eds.) 24-25 Feb., Munich, S. 40-52.