Land Use Policy 98 (2020) 104418

Contents lists available at ScienceDirect

Land Use Policy

journal homepage: www.elsevier.com/locate/landusepol

Design, development and usability testing of an LADM compliant 3D Cadastral prototype system

Barbara Cemellini, Peter van Oosterom*, Rod Thompson, Marian de Vries

Delft University of Technology, The Netherlands

ARTICLE INFO

Keywords:
3D Cadastre
3D Visualization
3D Spatial Data
Data Capture
LADM
Usability testing

ABSTRACT

This paper describes research into the design, development and visualization of mixed 2D and 3D Cadastre. A schema has been developed to accommodate this data, with provision for a time component. This paper describes the schema, the visualization requirements, and the provision of LADM-compatible views of the data for the purpose of developing the 3D Cadastral prototype. A significant volume of 2D + t Cadastral data, which also contained 2D + t footprint representations of 3D parcels, is currently incorporated in the Cadastral Database of Queensland. A moderate number of 3D building units, and a smaller number of volumetric parcels have been hand-encoded (from the survey plans), and added to this database. The mixture has been disseminated and displayed in KML through Cesium JS.

The visualization of cadastral parcels in 3D is a challenge, since legal boundaries are, in many cases, invisible in the real world; so how can we properly represent something that is not visible to our eyes? This paper uses the results from research looking into problems of occlusion and ambiguous perception (in terms of position, size and shape) of objects in the context of 3D cadastre visualization. The exploration of specific interaction techniques is essential to overcome these issues. After an initial internal usability test (with colleagues/ friends of the developers) our 3D Cadastres web-based dissemination prototype was improved. Next a public usability test is carried out to obtain feedback from different groups of professional users (legal, survey, ICT backgrounds). During the test, the users are asked to perform a series of tasks typical of cadastral systems. Each task is accompanied by a description to give the users some context. Then, each user is asked to reflect on his or her experience. In this paper we present the main results of the public usability test of the 3D Cadastres web-based dissemination prototype.

1. Introduction

In our 3D Cadastres (van Oosterom & Stoter, 2006) research we look for generic solutions, suitable to be deployed in many jurisdictions. To assess our design and development of an LADM compliant 3D Cadastral prototype system via usability testing we selected the case of Queensland, Australia as this state has among the largest number of real 3D Spatial Units (parcels). The state of Queensland, Australia has maintained a land administration system, based on Torrens System of registration (Queensland Government, 2018). As modern practices in land use have developed, the administrative system has developed with them, accommodating different definitions of property. Thus, the early Cadastre was a register of simple areas of land, this has been augmented with 3D unit rights, timeshare, and volumes of space defined by metes and bounds. The state maintains a database, known as the Digital Cadastral Database (DCDB), which contains a record of all spatial units, but this is reduced in scope to carrying only 2D representations.

In Queensland many thousands of 3D parcels, both building unit format (more than 200 000) and volumetric format (3 000), have been submitted over the past decades (Thompson et al., 2015). The footprints of these 3D Spatial Units have been extracted from the survey plans and included in the DCDB. Despite this rather long history and experience, there are a number of remaining challenges. In this paper we focus on the dissemination and visualization of the 3D Cadastral parcels. In other jurisdictions several other (prototype) systems have been developed and presented (Pouliot et al., 2018), however great challenges remain with respect to interaction and visualization of large numbers of 3D parcels, relating these to the existing 2D parcels, the earth surface, and 3D reference objects (physical building, bridges, tunnels, infrastructure, etc.). The existing data model of the Queensland DCDB is not described using the concepts and terminology of ISO's Land Administration Domain Model (ISO-TC211, 2012). Further the model does not support 3D

* Corresponding author.
E-mail address: p.j.m.vanoosterom@tudelft.nl (P. van Oosterom).

https://doi.org/10.1016/j.landusepol.2019.104418
Received 22 April 2019; Received in revised form 31 October 2019; Accepted 5 December 2019
Available online 03 January 2020

parcels and reference objects (earth surface, topographic objects). We are therefore developing an extension of the DCDB to support 3D parcels as used in the Queensland legislations, regulations and existing systems (DNRM, 2013,DNRM, 2016), incl. the DCDB and collection of submitted survey plans. Next, for a selected study area with all relevant types of 3D Cadastral parcels occurring, both building format and volumetric format parcel (above and below the earth surface, and covering both land and water areas), we are converting the information of the survey plan documents (pdf or tiff) into 3D geometry as collections of polyhedron faces. These activities consist of a mix of manual interpretation of the survey plans and using a dedicated (Java) program as support to generate the correct faces of the polyhedron. Currently only planar faces are supported, but curved surfaces do occur (e.g. cylinder patches) and they need to be supported also in the near future (most likely via approximations).

The case study area is in central Brisbane on Kangaroo Point and is adjacent to the Story Bridge over the Brisbane River; see Section 3. The polyhedron faces are loaded in a prototype version of the extended DCDB (stored in PostgreSQL/PostGIS) and combined with other 2D and 3D spatial data. This now makes it possible to ensure that a collection of 3D parcels is not overlapping. So far this is still the 'internal side' of the 3D Cadastre in Queensland. The next step is dissemination and visualizing the 3D parcels. When disseminating data it is important to use standard terminology and concepts. Therefore, the IS 19152 (LADM) standard was used. Persons not familiar with the Queensland terminology can thus understand the content

Note – this paper will preferentially use the language of the Land Administration Domain Model (LADM) (ISO-TC211, 2012), but the terminology used within the DNRM (Department of Natural Resources and Mines) will occasionally be needed. thus: 1. a spatial unit (LA_SpatialUnit) is internally referred to as a "parcel" and 2. a basic administrative unit (LA_BAUnit) as a "lot". In representing the Cadastre digitally, the current DCDB is as follows:

- 2D spatial units are recorded as polygons with linestyles. A subset of them (known as "base parcels") are constrained to form a complete, non-overlapping coverage.
- 2D secondary interests (e.g. easements) are recorded as polygons which overlap base parcels.
- Spatial units defined by the building walls are recorded as attributes (unit number, and floor area only) within the base parcel (i.e. no graphical representation is present). The building footprint is recorded as a polygon within the base parcel.
- Volumetric spatial units are "flattened" into a 2D polygon, which overlaps the base parcel(s).

But, note that the "2D" DCDB has a time component, using a variant of the "Versioned Object" pattern, and maintains a history (since about 1990). Thus it can be better described as 2D + t. Adopting a web-based solution is optimal for dissemination of the data as web browsers offer a relatively hardware/software independent platform, reaching many possible users without great efforts at the user side. A custom-made Java encoding software extracts the 2D and 3D cadastral parcel geometry for a certain selected area from a PostGIS database into KML encoded files that are placed on the web server. In order to retrieve the persons/parties and the RRRs related to the 2D and 3D parcels a Web Feature Service (WFS) is set up that gives access to a number of Land Administration Domain Model (LADM) views in the database.

At the client side, a geo-information aware WebGL based solution is applied to visualize the 3D parcels (and reference objects) and interact with the cadastral information (Cesium JS based client). The following tools related to 3D visualization are included in the prototype: Navigation tools and view controls, Tooltip (which shows information about the parcel and the administrative data related to it), Integration of topography (i.e. a DTM), Transparency, Object selection and highlight, Object search (to check which parcels are owned by a certain person), Dynamic elevation tool to solve the problem of subsurface visualization, and Camera start-up position (implemented to start up the viewer at the right location).

Following this introduction (Section 1), a summary of our wish list for a 3D Cadastral visualization system is given, together with the analysis of various 3D web-viewer platforms in Section 2 (and in the end selecting Cesium JS). The design and development of our 3D Cadastral prototype based on the LADM schema (in PostgreSQL/ PostGIS) and using the Cesium JS platform is presented in Section 3, which also elaborates on data conversion and data loading. The organization of the usability test and the results of the usability tests are given in Section 4. Conclusions and future work are given in Section 5.

2. WISH LIST AND 3D WEBVIEWER OPTIONS

Using a web-based solution makes a lot of sense for dissemination as web browsers offer a relatively hardware/ software independent platform, reaching many possible users without great efforts at the user side. Two types of wish lists are identified: 3D cadastral data visualization requirements (functional/platform independent in Subsection 2.1) and wish list for the 3D web viewer (as this is target dissemination platform; see Subsection 2.2). Subsection 2.3 presents the analysis of a range of 3D webviewers and based on the two wish lists, makes a platform selection for our prototype.

2.1. Wish list for 3D visualization of cadastral data

In order to define what is the best way to visualize 3D cadastral data it is important to analyze the existing literature and define a list of requirements. Both the issues related to 3D visualization and 3D cadastre will be taken into account. In this section, the requirements regarding 3D visualization of cadastre are listed, more details in (Cemellini, 2018, Thompson et al., 2018):

1 Navigation tools and view controls (Shojaei et al., 2013),
2 Integrating topography and reference objects (Vandysheva et al., 2012, Shojaei, 2014, Kalogianni, 2016, Pouliot et al., 2016)
3 Transparency (Ying et al., 2012, Wang et al., 2016, Pouliot, et al., 2016); see Fig. 1.
4 Object selection (Vandysheva et al., 2012, Shojaei, 2014, Pouliot et al., 2016)
5 Object search (van Oosterom et al., 2000, Shojaei, 2014)
6 Wireframe display (Shojaei, 2014)
7 Explode view (Shojaei, 2014, Ying et al., 2016); see Fig. 2.
8 Sliding (Vandysheva et al., 2012, Shojaei, 2014)
9 Cross-section view (Shojaei et al., 2013, Shojaei, 2014, Pouliot et al., 2016); see Fig. 2
10 Visualization cues (Pouliot et al., 2016)
11 3D measurement tools (Shojaei et al., 2013, Shojaei, 2014)

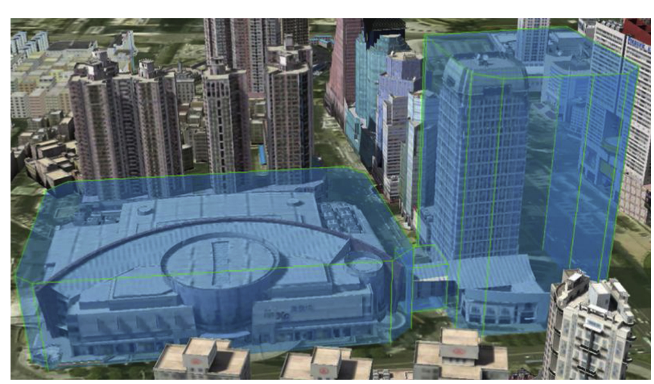


Fig. 1. Use of transparency to enhance the difference between physical and legal objects (Ying et al. 2012).

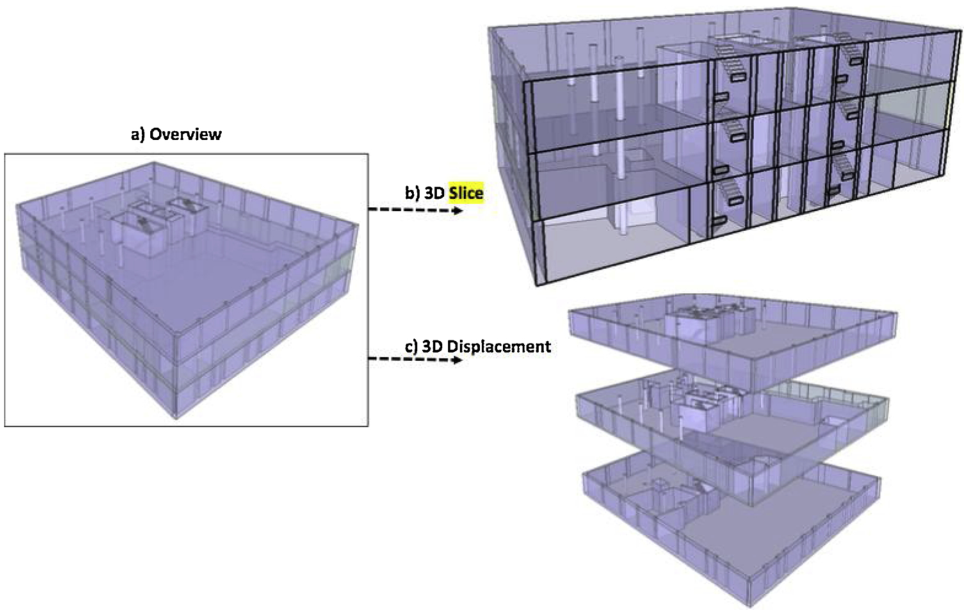


Fig. 2. Combination of 'explode view' and 'cross-section functionalities' (Pouliot, et al., 2017).

12 3D buffer (Shojaei, 2014)
13 Display partly unbounded objects and ‘complex’ geometries (Shojaei et al., 2013, Thompson et al., 2015)

2.2. Wish list for the 3D web viewer

In addition to the functionalities related to 3D visualization, the web viewer requirements must be considered too. One of the main aims of cadastre is to make information available to everybody and the web is a powerful tool which can make this happen. Consequently, the choice of the most appropriate web platform is crucial. Having a good 3D visualization would be useless if the platform in which the data is visualised is not suitable for that purpose. Choosing the proper web based platform for the dissemination of 3D cadastre is not an easy task. Most of the following functionalities should be present:

- Platform and browser independence/use GPU (Shojaei et al., 2013, Khronos Group., 2017)
- Handling massive data and caching/tiling between server and client (Shojaei, 2014, Rovers et al., 2017)
- Layers control (van den Berg et al., 1999, Shojaei, 2014).
- Database support (Shojaei, 2014).
- Support different models: vector/polyhedral, raster/voxel, point clouds (Pouliot et al., 2016)
- Support of basic 3D topographic visualization (Shojaei, 2014)
- Support for geo-referencing (Stoter et al., 2012, Pouliot et al., 2016).
- Ensure spatial validity, 3D vector topology (Shojaei, 2014, Thompson et al., 2017)
- Underground View (Shojaei, 2014); see Fig. 3.

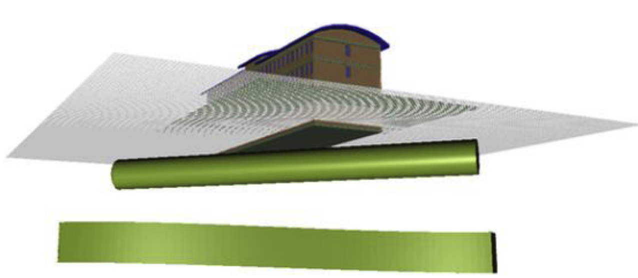


Fig. 3. Example of underground view (Shojaei, 2014).

- Open source platform (Shojaei, 2014)
- Possibility for the platform to be extended (Shojaei, 2014).
- 2D overview map for orientation (van den Berg et al., 1999, Shojaei, 2014)

2.3. Analysis of 3D webviewers

Choosing the appropriate visualization application for 3D cadastre is a big challenge because of the wide variety available due to the ever-growing new technologies. For this reason, a good knowledge of existing platforms and their capabilities can help building a successful cadastral prototype (Shojaei, 2014). The web viewers analysed will be based on the cutting-edge WebGL technology, mainly because of its plug-in free interface in all mainstream browsers.

As described by the Khronos Group (2017) who developed it, WebGL is a cross-platform open source web standard for a low-level 3D graphics API based on OpenGL ES (Khronos Group, 2017). The API context is obtained from the HTML5 < *canvas* > element, which means that no plugin is required in the web browser to use the application (Pereira, 2013). Since WebGL is a low-level API, drawing a simple 3D model needs a lot of work. Consequently, several open source JavaScript libraries have been developed to simplify the programming process. One of the most popular ones is *three.js* which provides higher level access to the API to make programming simpler (Shojaei, 2014). A big advantage of WebGL is that it brings 3D into the web without the installation of plug-ins and it is implemented directly into the web browser. Most browser vendors like Apple, Google and Mozilla are members of the WebGL Working Group (Khronos Group, 2017). Hence, WebGL is supported by all major browsers and it works on many mobile platforms (Mackey, 2017). The payoff in using WebGL is that it is much faster than the 2D canvas context, so it performs rather well in case of complex visualizations. Also, it can produce a degree of realism and configurability that is hardly possible with other solutions (Danchilla, 2012). Although WebGL represents the state-of-the-art in 3D graphics for the web, some limitations must be pointed out. First of all, visualizing massive datasets in WebGL can be an issue. Often browsers and mobile devices have a limited amount of cache memory which cannot be exceeded. Hence a smooth visualization is not always possible. Secondly, WebGL has been designed for today’s graphic cards, therefore old generation computers may not support it (Shojaei, 2014). In addition, setting up WebGL without a framework is not convenient due to

Table 1
Requirements check in the different WebGL platforms.

	Platforms					
Requirements	**iTowns**	**Cesium JS**	**OSM Buildings**	**WebGL Earth**	**GeoBrowser 3D**	**ESRI CityEngine Web Viewer**
Navigation tools and view controls	✔	✔	✔	✔	✔	✔ but not tooltips
Integrating topography and reference objects	✔	✔	✔	✔	✔	✔
Transparency	✔	✔	✔	✔	✔	✔
Object selection	?	✔	✔	✔	✔	✔
Object search	?	✔	✔	✔	✔	✔
Wireframe display	✔	✔	?	?	?	✔
Explode view	?	?	?	?	?	?
Sliding	?	?	?	?	?	?
Cross-section view	?	?	?	?	?	✔
Visualization cues	?	?	?	?	?	?
3D measurement tools	✔	✔	?	?	✔	✔ only through programming
3D buffer	?	✔	?	?	?	?
Display partly unbounded objects and 'complex' geometries	?	?	?	?	?	?
Platform and browser independence	✔	✔	✔	✔	✔	✔
Handling massive data and caching/tiling between server and client	✔	✔	✔	✔	✔	? but foresees the possibility of handling massive cadastral data
Layers control	✔	✔	✔	✔	✔	✔
Database support	?	?	?	?	?	✔
Support different models (vector/polyhedral, raster/voxel, point clouds)	✔	✔	✗	✗	✔	✔
Support of basic 3D topographic visualization	✔	✔	✗	✔	✔	✔
Support for geo-referencing	✔	✔	✔	✔	✔	✔
Ensure spatial validity (3D vector topology)	?	?	?	?	?	?
Underground View	?	?	?	?	?	✔
Open source platform	✔	✔	✔	✔	✔	✗
Possibility for the platform to be extended	✔	✔	✔	✔	✔	✔ Python scripting interface
2D overview map (orientation)	✔	✔	?	?	?	?

the amount of work needed to build up the application from scratch. For this reason, a good option is use an existing WebGL based viewer in which many of the functionalities needed are already implemented and some others can be added (Danchilla, 2012).

Serveral different WebGL based viewers have been identified: iTowns (2017) (IGN, 2015), Cesium, 2017, OSM Buildings (2017), WebGL Earth (2017), (Chaturvedi, 2014), GeoBrowser 3D (2017), and ESRI CityEngine Web Viewer (2017) (Ribeiro et al., 2014). The goal is to look for platforms which already have a considerable number of requirements implemented to facilitate and fasten the development of the 3D Cadastral prototype. If many of the functionalities are already present in an existing demo, this means that there is the concrete possibility to implement them again and find related support to do so. It follows that the extensibility of the platform is a *'must'* to develop the remaining requirements. The testing phase consists of checking the presence of the requisites in demos or prototypes built on top of that specific application. Nevertheless, some requirements are hard to check only based on evidence, in this case the related literature has also been considered. Table 1 summarizes the requirements check according to the different platforms. The green checkmark (✔) means that the requirement has been verified to exist, the orange question mark (?) means that no evidence has been found or no one still implemented it, and the red cross mark (✗) means that it is certainly not possible to implement that requirement. In addition, most of the demos have different purposes than 3D cadastre and many of the functionalities are just not needed for those use cases.

The table is divided into two main sections corresponding to the two main groups of requirements described earlier in the paper. In general, all the basic functionalities for 3D visualization, such as navigation tools, transparency, topography support, and so on, are present in all web viewers. On the other hand, there is no evidence of the range of capabilities requested by 3D cadastre, like explode view and cross section view. A possible explanation could be the different purpose of the demos analyzed. Most of them are in the geospatial domain, but also meteorology, history, defense and smart cities seemed to be quite popular fields. Since none of these domains is directly related to 3D cadastre, there is no need for these applications to develop such functionalities. Nevertheless, thanks to the extensibility of the platforms, most of these requirements can probably be implemented. Considering all the tested web viewers, along with the specified requirements, the most likely applicable and capable of supporting 3D cadastre visualization seem to be iTowns, Cesium JS and ESRI CityEngine. The first two have many of the requirements already implemented and a vast community of users, and on top of that, dedicated forums and blogs are actively helping developers to implement their applications. Moreover, Cesium JS has a detailed API documentation, code examples and tutorials to practice and get to know the library. In the same way, iTowns provides great support for users and a wide variety of examples. ESRI CityEngine is the only commercial and non-open-source solution analysed. The number of requirements met is considerably high and could be sufficient to implement some basic functionalities. In addition, the integrated Python scripting interface enables the user to carry out attribute queries, control repetitive tasks and automate specific actions. As Cesium JS and iTowns, also OSM Buildings provides a good API documentation but it owns less functionalities, therefore this candidate is less likely to be used in the actual implementation of the 3D cadastre viewer. WebGL Earth and GeoBrowser 3D are still in their early development stages, their community is relatively small and therefore they have limited support compared to more mature web viewers. As already mentioned, they are powered by Cesium JS which has a wide community of users and good support for developers. Although, compared to Cesium JS, they have implemented a smaller number of requirements.

Cesium JS is an open-source JavaScript library which uses WebGL to create 3D geospatial applications. It is led by the Cesium Consortium and thanks to the Apache 2.0 license it is free for both commercial and non-commercial uses (Cesium, 2017). Cesium JS supports open formats wherever available and develops new open formats when they are not; in this way, it sets new standards for open 3D geo-spatial formats. The big advantage of Cesium JS is the existence of a big community which supports developers in their work; developers can post questions on the Cesium forum about specific problems and get answers from other members of the community. By doing so, the expertise can be easily shared (Cesium, 2017). Furthermore, in Cesium JS it is possible to switch among 3D globe, 2D map and 2.5D Columbus view within a single API. Time dynamic simulations are also supported, together with realistic environment features such as sunlight, atmosphere, fog, water and moon (AGI, 2017). Cesium also includes an interesting application called 'Sandcastle', which provides live coding on the web browser (Chaturvedi, 2014).

3. PROTOTYPE 3D CADASTRAL SYSTEM

The prototype focusing on the visualization and dissemination of 3D cadastre is currently under development. As explained in Section 2 a web based solution has been designed and developed, allowing maximal reach. An overall diagram showing the server-client system architecture is shown in Fig. 4. At the server side, a web data service (WFS) is established on top of the LADM views in the database, exposing the 2D and 3D cadastral information to the outside world. At the client side a 3D geo-information aware WebGL based solution is applied to visualize the 3D parcels (and reference objects) and interact with the information (Cesium JS based client). Important is that the client uses a selection of information (that is, not all available data in a state or nation-wide database) as all information in the cadastral database would be far too much; only looking at the 2D parcels this is already too much (millions of parcels, with many attributes). Server-client caching techniques are used to manage well the communication between server and client when panning over the area. For the back-end storage, the database option has been selected for two reasons. The advantage of having the data in one place and always up to date, and the fact that the 2D parcels of (nearly) all cadastral systems are already stored in a DBMS. A custom-made Java encoding software extracts the 2D and 3D cadastral parcel geometry for a certain selected area from a PostGIS database into KML encoded files that are placed on the web server. In order to retrieve the persons/parties and the RRRs related to the 2D and 3D parcels a Web Feature Service (WFS) is set up that gives access to a number of Land Administration Domain Model (LADM) views in the database. Apache Tomcat is used as a web server and GeoServer is used as WFS server.

As test data a large selection of 2D parcels have been loaded from the Queensland DCDB. This will ensure that there are enough data to make the system loads realistic. Capturing the 3D parcels has been more difficult, but progress has being made. In particular, it combines different types of data to give context to the 3D cadastral parcel. Among the data to be included in the prototype we can find: 3D survey plans (from Queensland Cadastre), either in' building format units' and in' volumetric parcels', which will be described in the next subsections (3.1 and 3.2); 2D cadastral parcels (from Queensland Cadastre); Rights, restrictions and responsibilities (RRRs are faked for privacy reasons); Elevation data in order to make the visualization more complete and meaningful; and Reference data such as topographic objects, either in 2D or 3D. Subsection 3.3 describe the client side priorities in the implementation. The dynamic elevation tool to avoid subsurface parcels being invisible in Cesium JS is described in Subsection 3.4. The LADM complaint database schema is introduced in Subsection 3.5, while Subsection 3.6 covers the data capture and conversion.

3.1. Building format units

These are defined by the building itself, typically by the walls (to the middle, or to the outside etc. of the wall). This means that the use of a Building Information Model (BIM), or an as-constructed building maintenance plan is a good ideal source for data (Aien et al., 2011, Atazadeh et al., 2017), but only where such a model is available. Ideally, the data would be submitted electronically as a component of the

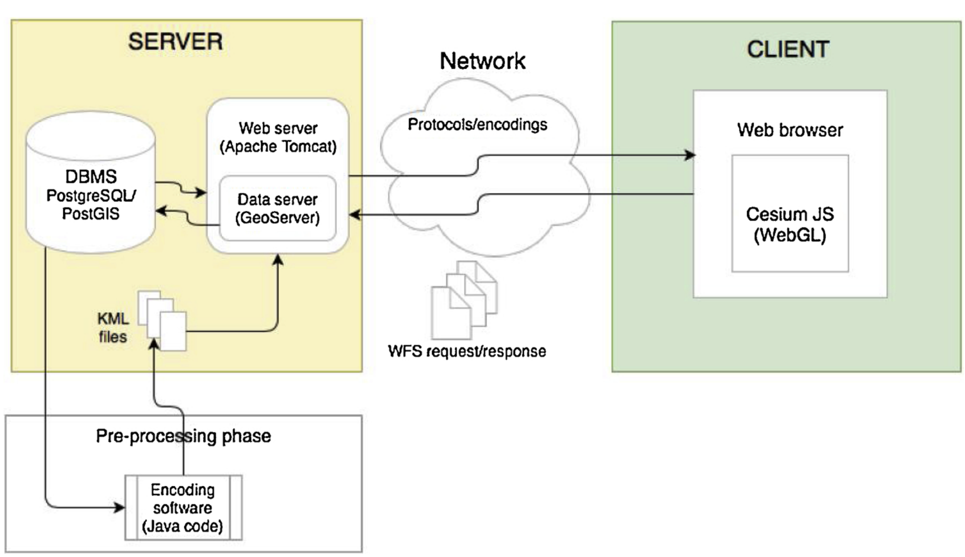


Fig. 4. Server-client system architecture diagram of the 3D cadastre prototype.

Fig. 5. Three plans of building format units.

plan of survey that is used in Queensland to register the properties (Khoo 2012, Karki et al., 2013, Gulliver et al., 2017, Thompson et al., 2017). Given that the 3D properties in the area in question were registered at different times in the past, when survey plans were submitted on paper, or later in pdf or tiff format, the only option available for this R & D project was manual capture. For the building format survey plans, two or more known existing cadastral points were identified with points in the plan, and a simple linear fitting of the incoming data applied. Where floor plans did not identify any true cadastral points, it was necessary to identify corners of units on other floors. This produced results consistent with the accuracy of the unit sketches. Further, for the building format survey plans there is usually no height information apart from an indication of the ground level (height above datum). The height of the top of the building was guestimated from Google earth, and the height of individual floors calculated from the roof height and ground level. To date, three non-trivial plans have been converted and entered, consisting of five buildings (two of the plans are for complexes of two buildings each), and 349 individual units (but note that in some cases, balconies and car parks are counted as separate units, so the true number of units would be closer to 200). These can be seen in Fig. 5, which also shows the 2D parcels from the DCDB. The building in the foreground of Fig. 5 has been modified in terms of the subdivision into units, and if viewed as at an earlier date looks a bit different. This is an issue to be investigated in the near future, to allow visualization of variations in the Cadastre – perhaps with a "time slider".

3.2. Volumetric Parcels

Volumetric survey plans (e.g. the tunnel) are required to carry a connection to existing cadastral points or survey marks, allowing more easy geographic referencing. Further, the Reduced Levels (RL) are given for every cadastral corner as an absolute height above/below datum. A section of the "Clem 7" road tunnel has been captured and is visible in Figs. 5 and 6. It should be noted that the tunnel volume is fragmented at the boundaries of the surface parcels, so that the section is composed of many individual parcels. The large grey "monolith" is a parcel which runs from the tunnel to a height of 200 m. There is clearly no construction within the space, which has probably been reserved to prevent building above a tunnel access point. In Fig. 6, although it is being viewed over Google Earth, all parcels (2D and 3D) have been pushed up by 50 m. This allows below ground parcels to be seen by making the 2D parcel representations partially transparent.

3.3. Used client side tool and priorities in the implementation

As has been motivated in Subsection 2.3, Cesium JS has been chosen as best platform to develop the client side of the 3D cadastre prototype. Some implementation priorities have been added the wish lists introduced in Subsections 2.1 and 2.2. Priority items on the wish list related to 3D visualization are:

- *Navigation tools and view controls.*
- *Tooltip*, which shows information about the parcel and the administrative data related to it.
- *Integration of topography* (i.e. a DTM).
- *Transparency.*
- *Object selection and highlight.*
- *Object search*, to check which parcels are owned by a certain person.
- *Dynamic elevation tool* to solve the problem of subsurface visualization.
- *Camera start-up position*, implemented to start up the viewer at the right location.

Note that the last two items in the list were not included among the initial requirements as defined in Subsection 2.1, but they have been added to overcome some technology issues and to increase the user-friendliness of the prototype. The following items on the wish list related to the web viewer are prioritized to be included in the 3D cadastre prototype:

- *Platform and browser independence* thanks to the WebGL technology.
- *Layers control*, although it is not possible for the user to add his own layers or modify the existing ones (i.e. by changing the level of transparency). On the other hand, the users can toggle on and off the 2D parcels according to the information that they want to visualize.

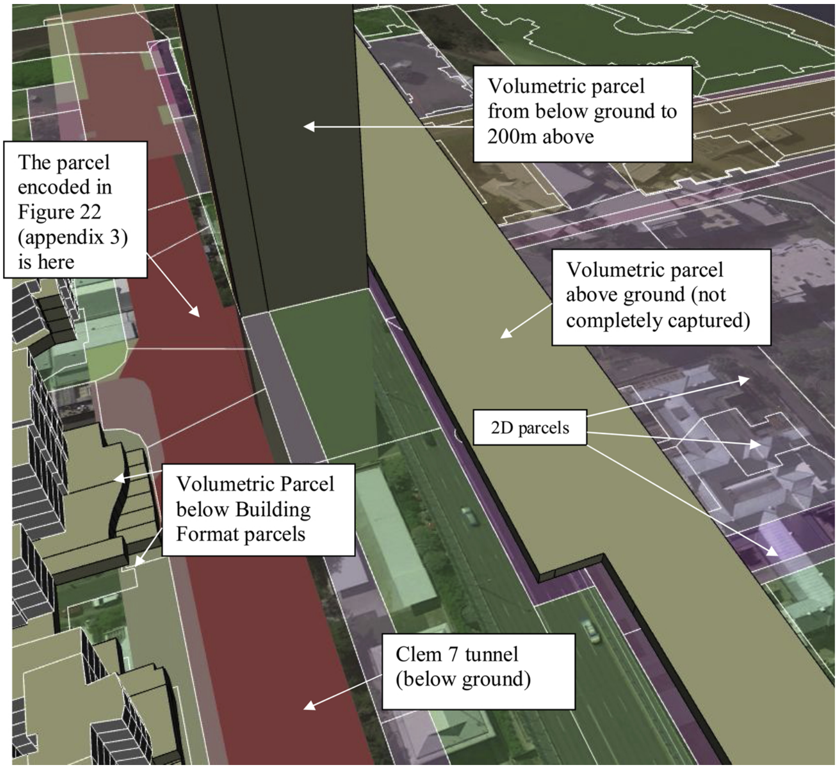


Fig. 6. A mixture of 2D parcels, Building format and Volumetric parcels.

- *Support for topographic visualization* from a technological point of view.
- *Support for geo-referencing* thanks to the virtual globe technology.
- *Underground visualization* is now possible as a result of the vertical shifting of the parcels.
- *Open source platform.*
- *Possibility for the platform to be extended.*

The last requirement on the list is probably the most important one, since it made possible the implementation of all the others by changing the JavaScript source code of Cesium JS and add new functionalities to the already existing ones (i.e. feature highlight and layer control among others). Fig. 7 shows the interface of the 3D cadastre prototype containing the GUI elements just described.

3.4. The dynamic elevation tool

The dynamic elevation tool mentioned above needs special attention since it was not included in the initial list of requirements contained in Cemellini et al. (2018a). The dynamic elevation tool is needed to overcome the problem of underground visualization. This tool would provide the possibility to move the 3D objects up or down of a user

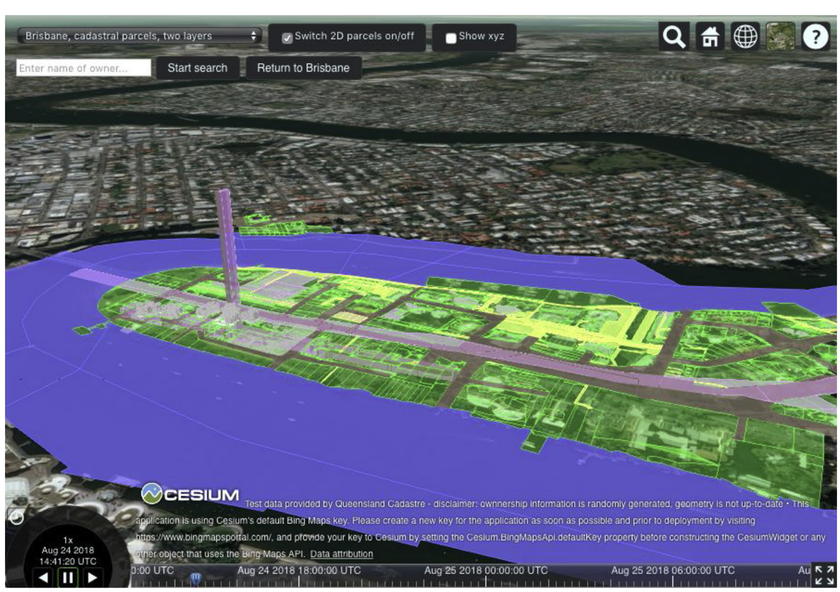


Fig. 7. Interface of the 3D cadastre prototype.

defined amount to be able to visualize in detail the underground parcels that are hidden by the earth surface. The advantage of using this tool is that the user can have a reality-like visualization at the viewer start up (i.e. with the parcels in the correct location and height) and, if needed, alter the visualization in order to better visualize the hidden parcels. This results in a better understanding of the situation, the user still has a feeling of the actual context of the parcels before the visualization gets distorted to visualize the details. Different implementations of the dynamic elevation tool are possible according to the set of active objects to be shifted, the direction of the shift and the graphical control element used to power the tool.

Set of active objects: The set of active objects to be considered is central in the implementation of the dynamic elevation tool since it determines which and how many objects are altered from their initial position.

- A single parcel could be shifted in case the user needs to analyse in detail the boundaries of that specific parcel, without the context provided by the surrounding objects.
- A (sub-)set of parcels could be used when the user needs to inspect an underground object composed of multiple parcels (i.e. the 'Clem' tunnel in the Brisbane city centre)
- All the parcels could be shifted if there are no occlusion issues involved in the visualization and the user wants to have a comprehensive view of both above ground and underground parcels.

In addition, the set of active object could grow (i.e. adding new objects to the set) or shrink (i.e. removing objects from the set) according to the administrative/spatial queries that the user wishes to perform.

Direction: The term refers to the orientation of the shifting of objects with respect to their initial position.

- Vertical. Shifting the parcels in vertical direction, from the bottom towards the top, is probably the most straightforward way to implement this tool.
- Horizontal. After the parcel has been raised above ground, a horizontal shift could be applied in case other parcels occlude the view of a parcel. This functionality is similar to the sliding tool as described in Cemellini, 2018. This is a good way to look inside a building by making all parcels of a specific floor part of an active set of active objects when sliding out.
- In every direction. Nevertheless, in case the user requests a higher degree of freedom, the tool could be implemented by allowing the shift in every direction, maybe also supporting rotations of objects.

Graphical control element: The graphical control element in a Graphical User Interface (GUI) refers to an element of interaction.

- Drop-down menu. The use of a combo box/drop-down menu allows the user to choose between a limited set of predefined Z values to shift the (selected) parcels.
- Text entry box. Using a text entry box allows the user to type any Z value without minimum and maximum value restrictions. Although, the risk is that the users types an invalid or non-meaningful value.
- Interactive slider. The implementation of a slider/track bar is, likely, one of the most advanced options for the implementation of the dynamic elevation tool, since it allows performing an interactive elevation on the fly and it does not have the problem of inserting invalid values.

For the purpose of this research, the dynamic elevation tool has been implemented by means of a drop-down menu on the whole set of parcels (included the DTM) supporting a shift in the vertical direction only.

3.5. Storage schema

The data for research described here does not at present cover the entire requirements of a cadastral database, notably excluding the party and RRR (Rights, Restrictions and Responsibilities) data, however the schema is designed flexibly, to accommodate: Parcel data in 2D (from the Queensland Government), Elevation data (courtesy of Fugro), Information form plans of survey (from the Queensland Government), Building /topographic (references objects) 2D/3D, Rights (or RRRs more general), and Parties.

The Land Administration Domain Model (LADM) provides a formal framework for describing land administration data, such as cadastral and deeds data. It is implemented by developing an application schema of the LADM model. Also, it enables involved parties, both within one country and in different countries, to communicate based on a shared vocabulary, suggested by the model. An early decision was made to keep the database in a form equivalent to that used in the current Queensland DCDB, but to expose views of that data in a form which is compatible with the LADM. This achieves four purposes: 1. It allows simpler loading of future data from the Queensland DCDB. 2. It allows modification of the prototype database structure without invalidating work being done on the visualization. 3. It provides the possibilitiy of defining a LADM derived protocol for delivery of mixed 2D/3D + t Cadastral data. 4. It indicates that a database which is not defined with LADM in mind can still support such a protocol. The database used in this research is also planned to be used for research on appropriate storage for mixed 2D + t and 3D + t Cadastre, and as such will be quite volatile in structure. To avoid the different strands of research interfering, it was decided that the data would always be made available for viewing/manipulation in a standard, but quite basic form. Thus 3D spatial units will be made visible (in the format of LA_SpatialUnit objects), with the extents defined by plane faces (LA_BoundaryFace). There is no attempt made to share faces between adjoining spatial units, and all faces observe anti-clockwise sense (viewed from the outside) (That is the convention of the normal vector pointing outwards is used). 2D spatial units will be made visible as LA_SpatialUnit objects, with the choice of a single LA_BoundaryFaceString making a closed polygon (anti-clockwise when viewed from above), or a number of face strings (each attributed with a "line style") which may be joined in order to make a closed polygon. The polygon may be complex (with holes), but not multiple. The lot details will be made visible in the format of LA_BAUnit objects. The term "made visible" in the preceding is intended to indicate that there may be views of database tables created that have the appropriate appearance and behavior, or redundant tables (materialized views) may be created if the views cannot be implemented efficiently. As research on the database structure, the decision view/redundant table may change without having a major effect on front-end development.

The objects described in Fig. 8 represent the data as seen by a front-end process. Note the close relationship between these objects and the LADM defined objects, the prefix QC (for Queensland Cadastre) has been used to distinguish them from the LADM equivalents. The actual database structure is (at least initially) as shown in Fig. 9. The mapping between the object attributes in the original DCDB to LADM and the SQL necessary to define the views can be found in Appendix A.

In the case of a LA_BoundaryFaceString ⇔ Parcel, the Parcel.Linestyles text is unpacked, and the values are associated to the line segments of the P_Shape polygon – that is, the polygon is divided into individual linestrings at each change of linestyle. The view described in Appendix A, Table A6 requires this unpacking of the polygon object and text string. To date, it has been implemented in procedural language, but may not be possible in a simple SQL view. A simplified version of the view, which omits the linestyle is sufficient for the present.

When selecting data via the views, the users will see LADM terminology only (for tables, attributes, relations, code lists, etc.). Using a

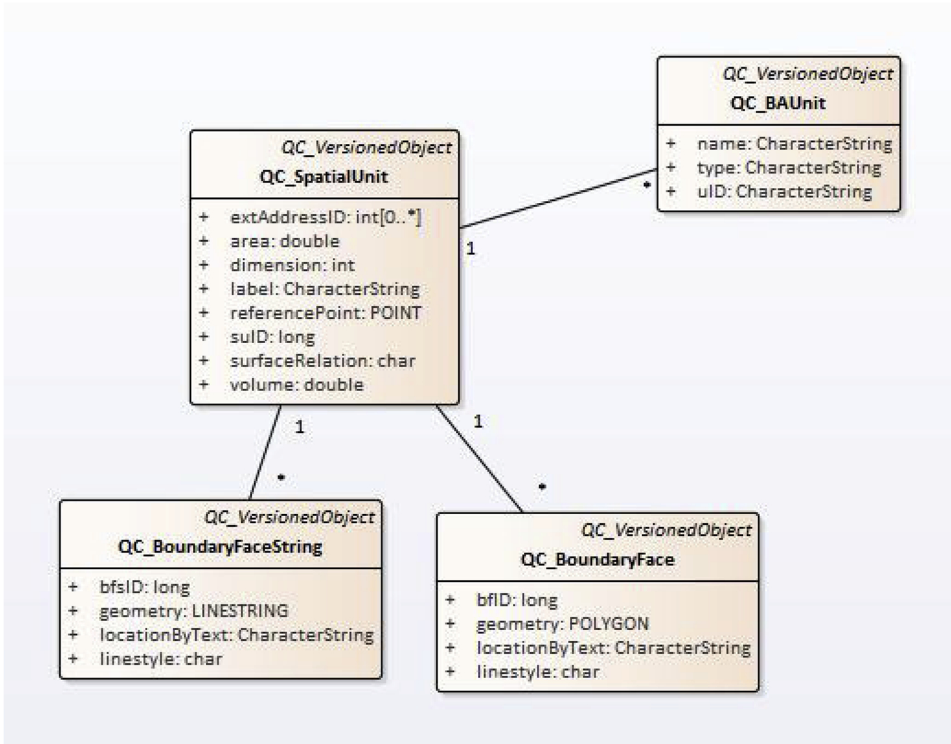


Fig. 8. Spatial Unit structure based on LADM.

web-based solution makes a lot of sense for dissemination as web browsers offer a relatively hardware/ software independent platform, reaching many possible users without great efforts at the user side. Therefore at the server side, a web data service (WFS) is established on top of the LADM views in the database, exposing the 2D and 3D cadastral information to the outside world. At the client side a 3D geo-information aware WebGL based solution is applied to visualize the 3D parcels (and reference objects) and interact with the information. Important is that the client uses a limited amount of information as all information in the DCBD would be far too much; only looking at the 2D parcels this is already too much (more than 3 million parcels, with many attributes). Server-client caching techniques are used to manage the communication between server and client.

3.6. Data capture and conversion

In order to further investigate the dissemination and visualization of 3D Cadastral parcels, it was decided that this needs to be in the context of a mixed 2D/3D DCDB. It is not considered acceptable that a different (and separately maintained) DCDB should exist for the 3D spatial units. For research purposes, a significant subset of the current DCDB (which is stored in an Ingres database), has been loaded into PostgreSQL/ PostGIS (more than 3 000 000 parcels). For the prototype database, the basic tables of the current DCDB (in Ingres), have been converted to PostGIS, with no change in logical structure. The spatial primitives of PostGIS have been used in preference to the locally developed methods used in DNRM (developed in the mid 1990's), but the semantics are unchanged. The main addition for the prototype is a table of faces, used to represent the boundaries of the 3D spatial units (parcels). In order to give a good selection of 3D spatial units, a number of parcels in suburb the Kangaroo Point have been manually encoded. This area as shown in Fig. 10 was chosen, because, while not in the most dense part of the city, it has a good representation of different classes of 3D object:

- Building format units (See Subsection 3.1),
- A major underground road tunnel, registered via volumetric format survey plans (in part running under building format units) (See Subsection 3.2),
- Other volumetric format parcels (not defined by buildings) (Subsection 3.2),
- Watercourse and land spatial units, and
- 3D parcels which have changed within the DCDB history period.

Details with respect to the actual conversion and encoding building format spatial units can be found in Appendix B, while details with respect to the actual conversion and encoding volumetric format spatial units can be found in Appendix C. In the analysis of 3D parcels of both kinds, it became clear that the DCDB has not been static since these parcels were first recorded. It proved necessary to allow to back-capture the history, and for historic details to be accommodated in the prototype database.

The DCDB records a single stream history of the database representation of the 2D spatial units, reaching back to the data load. Thus there is nearly 19 years of history available. This is not a “Bi-temporal” history (Jensen et al., 1994, Snodgrass et al., 1998, Thompson & van Oosterom 2019), but keeps a record of the contents of the database prior to and following each update, using a variant of the versioned object pattern (van Oosterom, 1997). If errors are found in the database, there is no provision to update the historical record. i.e. it will show what we thought in 2014 the real-world situation was in 2014 – not what we now know it to have been.

The Queensland DCDB contains 2D representations of 3D spatial units, so when the spatial units were were encoded in 3D, they were given a creation time stamp based on that of the 2D representation. Thus historically, they appear to have been originally captured in 3D. This is considered to be a form of "back capture". A further form of back capture was facilitated by the software, which allowed the encoding of a subsequent version of a spatial unit, forcing a retirement of the earlier version.

Since, for privacy reasons, we used fake names for the parties (based usually on the plan identifiers - e.g. "RP12345, Mr and Mrs"), we also generated changes of “ownership” of parties, with timing based on

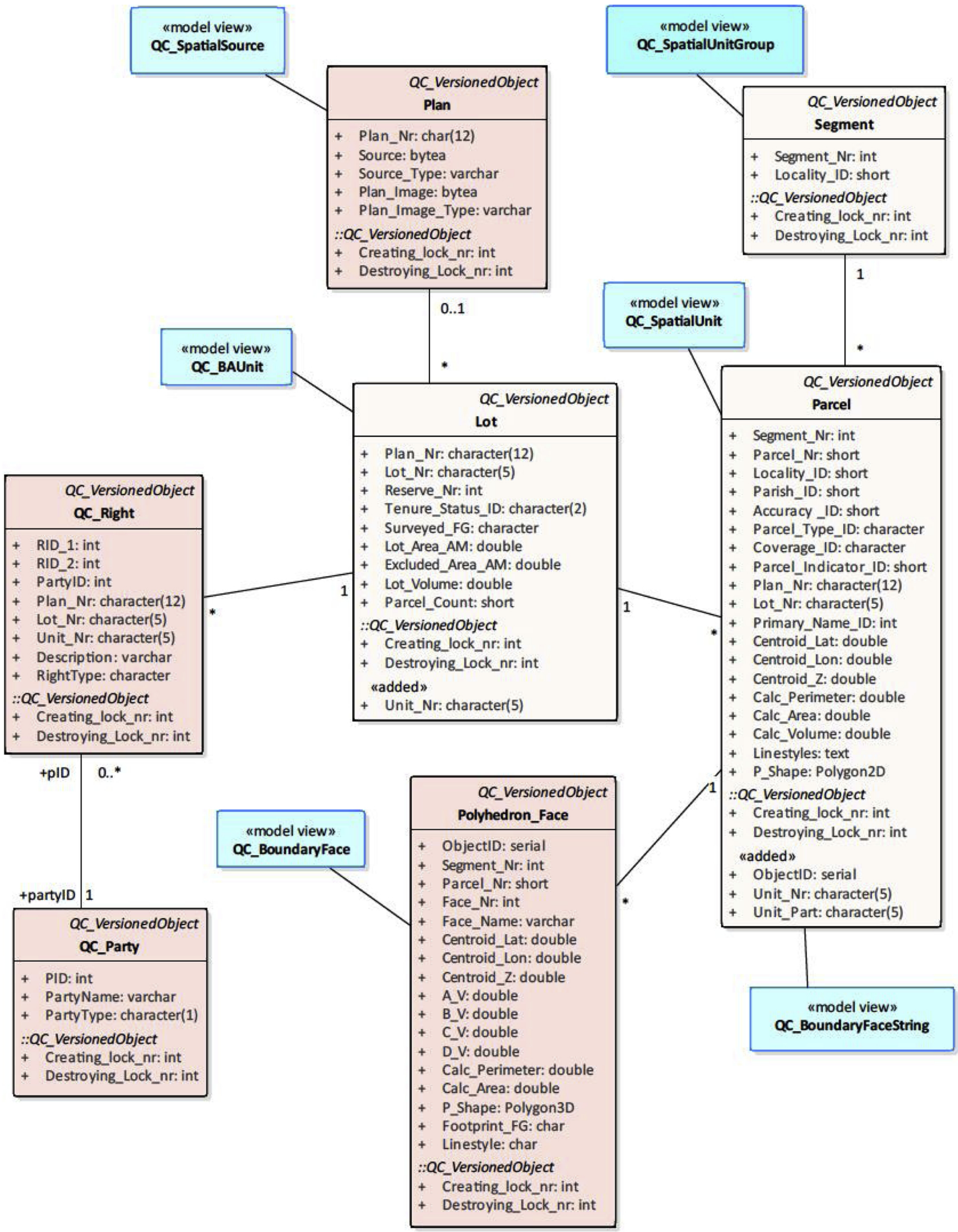


Fig. 9. Database schema of the 3D cadastre prototype. Original tables from the Queensland DCDB are shown in light orange. New tables are shown in red, while the 'views' are in blue.

actual (but probably unrelated) database events. Thus a fictitious ownership history was created – fictitious, but plausible in terms of quantities of data. In addition, the "back capture" functionality mentioned above was used on a number of 3D spatial units to create further fictitious geometric and RRR history.

In summary, there is a significant true history of the geometry of 2D spatial units; a significant but fictitious history of the RRR details of the 2D spatial units; and a smaller but still useful history of the 3D spatial units (real and fictitious). Full back capture of history has not been attempted yet, but provision has been made in the schema and capture software.

4. USABILITY TESTING

This section describes the usability test that was carried out by potential users to assess the developed 3D cadastre prototype. In order to carry out usability tests it is important to give a definition to the term” usability” in this particular context.

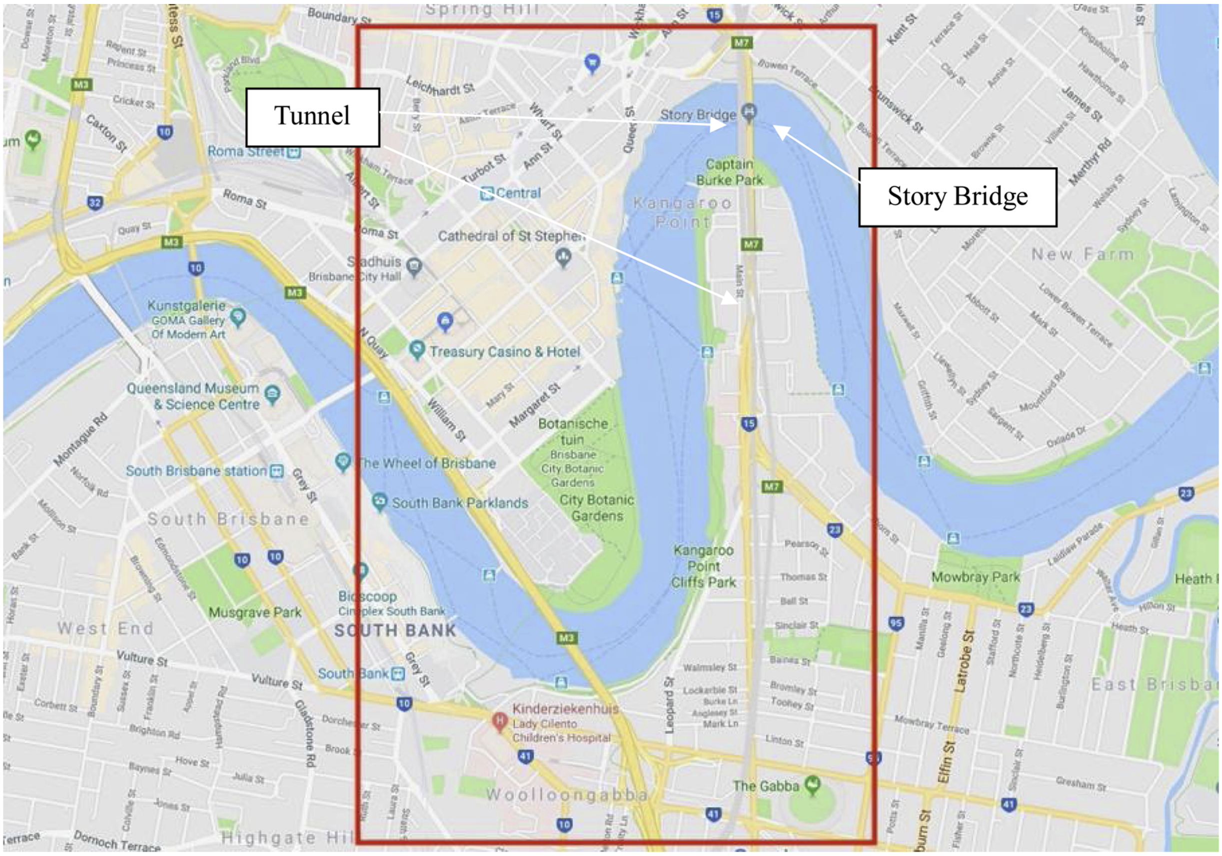


Fig. 10. Bounding box of the Kangaroo Point, Brisbane area chosen.

4.1. Generic remarks usability testing

According to the ISO 9241-11 usability is: *'the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use'*. Thus, usability is about effectiveness, in other words, whether users can complete tasks and achieve their goals. Usability is also about efficiency, which is often measured in time and effort for the user to complete his tasks. Finally, usability is connected with user's satisfaction and his opinion about the product/prototype (ISO, 2018). The absence of frustration while using something is what makes it usable. For this reason, usability is considered an issue only when it is lacking or absent. Usability testing is considered a research tool, a fundamental evaluation method which enables to assess a product or, in this case, a prototype. This process uses people as testing participants who are sampling the user groups to which the prototype is addressed (Rubin & Chisnell, 2008). Although usability testing is a crucial phase for the assessment and further development of the 3D cadastre prototype, some limitations of this approach must be pointed out:

- *Testing is an artificial situation.* In other words, it depicts an actual situation of usage, but it is not the situation itself. Therefore, in some cases this can affect the results.
- *The results of the testing do not prove that a product works.* The results depend on the way in which the test is conducted and it is not guaranteed that they are reliable.
- *The participants are not always fully representative of the actual user groups.* The actual end users of the prototype can be hard to fully identify, thus the completeness of the representatives depends on your ability to identify the target audience.
- *Testing through a questionnaire with sample users is not always the best method to assess usability.* In some situations, other techniques are preferred in terms of costs, time and accuracy. An example is a method called 'expert review', a heuristic evaluation made by an expert in the field, called to assess the prototype.

In spite of these limitations, usability testing is still considered a valid method to evaluate the usability of a product and to spot its weaknesses (Rubin & Chisnell, 2008). Through usability testing, developers can have direct information about the usability and user friendliness of their applications and an insight into the possible issues encountered by the testers (Ivory, 2003). In the usability tests we are looking for advice from the participants what are the advantages/shortcomings of the approaches. Nielsen (2000) demonstrated that five users are enough to spot about 80% of the potential issues.

4.2. Identifying the user groups

The first phase of usability testing involves the definition of the potential user groups of the 3D cadastre prototype. In the context of the 3D cadastre prototype, the potential users can be condensed into six different groups, in particular: 1. Researchers in the field of 2D and 3D cadastre (including staff of the MSc Geomatics for the Built Environment of TU Delft), 2. Professionals using cadastral applications (lawyers, notaries, engineers, architects, land surveyors, building managers, etc.), 3. Managers in the government and municipal authorities in charge of the maintenance of the cadastral system, 4. Public and private entities, companies, 5. Students of Geomatics for the Built Environment at TU Delft, and 6. General public, citizens.

Note that, this is a non-exhaustive list and new groups of end users can become apparent, being this a rather important innovation in the field of cadastre. Additionally, some studies recommend testing the prototype on 'extreme users', which means both regular and non-regular users. Regular users are professionals and people involved in the field of cadastre, while non-regular users are people that are not involved in the field and do not have any (or only little) knowledge of cadastral applications (Dam & Siang, 2018). In the list above, non-regular users are represented by the general public/citizens. Once the users have been identified and selected, the next step is to define a clear goal that the users have to keep in mind while performing their tests. The overall goal is to ensure the usability of the 3D cadastre prototype. More in detail, this means that the prototype must: a. be useful to the target users, b. help users to be effective and efficient in performing their tasks, c. be easy (or even satisfying) to learn/use, and d. eliminate issues and frustration for the users (Rubin & Chisnell, 2008).

4.3. Design of tasks and formulation of a questionnaire

The description of the task to be performed gives context to the test user and explains why that functionality is crucial in a 3D cadastre application. In order to obtain feedback, there is the need to specify clearly-defined tasks to be carried out by the users (on the basis of the main goal). Here is a list of defined tasks for the 3D cadastre prototype:

1 Navigate through the viewer, pan, zoom and rotate view to get familiar with the controls;
2 Toggle on and off the visibility of a layer;
3 Visualize the underground parcels, i.e. zoom close enough to see the details and navigate around it to see the boundaries from every angle;
4 Visualize information about a single parcel, i.e. ownership information, and unit/lot/plan number, etc.;
5 Search for a single owner and visualize all the parcels owned by that person.

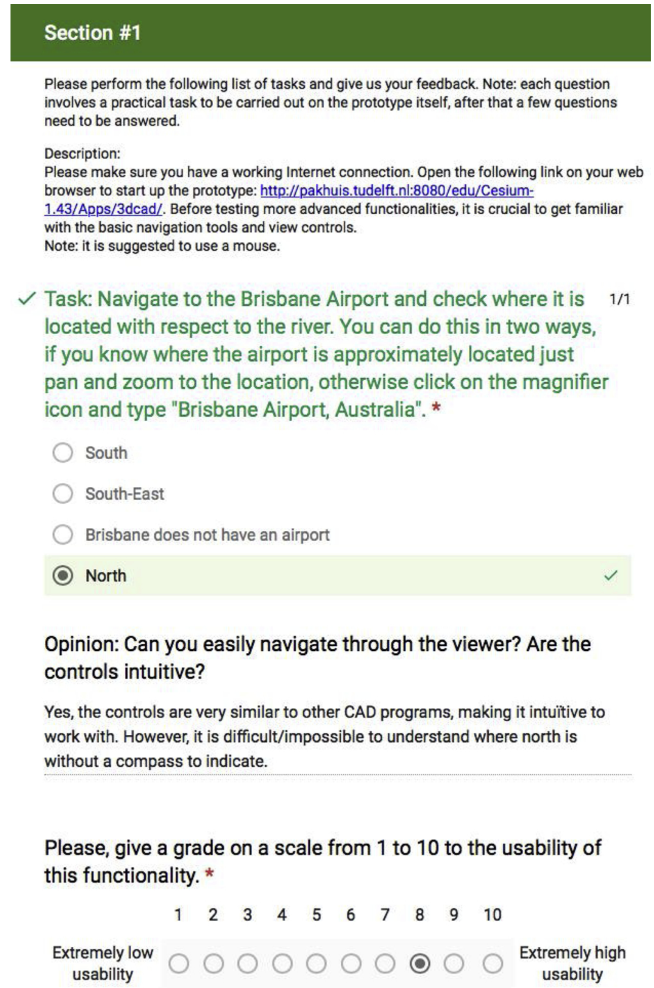
These tasks must be formulated in a clear and not misleading way, possibly accompanied by a short explanation, i.e. an example scenario, to give the user some context about the action that he/she is asked to perform (Nielsen, 2014). The next phase is the formulation of a questionnaire to be handed over to the test users. The questionnaire has been created with Google Forms, an online tool to create surveys, questionnaires, etc. The purposes of the questionnaire are (1) to guide the users through the usability test with detailed explanations, (2) to present the tasks for the test persons, and (3) to collect the answers from the test. Before starting, the users need to fill in their personal details and answer some multiple-choice questions about the user group they belong to and their knowledge of Cesium JS. The core of the questionnaire is composed of five main sections. Each section includes a description, a task, an opinion and a grade as shown in Fig. 11. For a complete description of the questionnaire, see Appendix D (Cemellini, 2018).

4.4. First usability test and improving the prototype

In the first usability test, 20 users participated. The majority of these users were students of the MSc Geomatics at TU Delft and researchers in the fields of 2D or 3D Cadastre (including the staff of the MSc Geomatics). Two test persons were very familiar with Cesium JS, three persons used it a few times before, and the others had never used it before. The average grade as given by the users for each of the 5 different functionalities are (Cemellini et al., 2018b):

- Section #1, Navigate (pan, zoom, rotate): 7.7.
- Section #2, Switch layers on/off: 7.2.
- Section #3, Visualize underground parcels: 6.2.
- Section #4, Get parcel information: 7.5.
- Section #5, Find parcels owned by person: 6.2.

Visualizing underground parcels was not so straightforward and some bugs were found in the 'Show XYZ' functionality. Also, some


Section #1

Please perform the following list of tasks and give us your feedback. Note: each question involves a practical task to be carried out on the prototype itself, after that a few questions need to be answered.

Description:
Please make sure you have a working Internet connection. Open the following link on your web browser to start up the prototype: http://pakhuis.tudelft.nl:8080/edu/Cesium-1.43/Apps/3dcad/. Before testing more advanced functionalities, it is crucial to get familiar with the basic navigation tools and view controls.
Note: it is suggested to use a mouse.

✓ Task: Navigate to the Brisbane Airport and check where it is located with respect to the river. You can do this in two ways, if you know where the airport is approximately located just pan and zoom to the location, otherwise click on the magnifier icon and type "Brisbane Airport, Australia". * 1/1

South
South-East
Brisbane does not have an airport
North ✓

Opinion: Can you easily navigate through the viewer? Are the controls intuitive?

Yes, the controls are very similar to other CAD programs, making it intuitive to work with. However, it is difficult/impossible to understand where north is without a compass to indicate.

Please, give a grade on a scale from 1 to 10 to the usability of this functionality. *

1 2 3 4 5 6 7 8 9 10
Extremely low usability — Extremely high usability

Fig. 11. Usability test questionnaire, Section #1. Each section contains description, task, opinion and grade.

remarks about speed were made, since the navigation/interaction was sometimes an obstacle due to the slow performance (i.e. during the load of the application and while clicking/highlighting some features). The less intuitive functionality (and most difficult to achieve) according to a large part of the test users was the parcel search functionality, which in most cases did not lead to any result. Therefore, for the future development of the prototype, this tool should be improved since it has also been indicated as the most crucial in the context of cadastre. The most appreciated feature was, instead, the possibility to retrieve administrative data from the database by means of a WFS request. Accordingly, the vast majority of the users managed to perform this task correctly, giving positive feedback about the functionality. For every task the majority of the users answered the question correctly. In general, the users achieved correctly 4 out of 5 tasks with an average of 3.7 points. Based on the experiences and feedback of the initial usability test, the following improvements have been made to the 3D Cadastres prototype (as used in the second and public usability test):

1 more consistent data (ground + elevated): On top of the Cesium globe with areal imagery, both the 2D and the 3D parcel layers can be displayed. To enable visualizing subsurface parcels, it is possible to elevate the 3D parcels. In the improved proved prototype the 2D and 3D parcels move together, which is easier for the user.
2 improved parcel search by owner: Initial prototype could not handle well searches where specified name is somewhere in the middle of the owner name as stored in the cadastral database. This has now

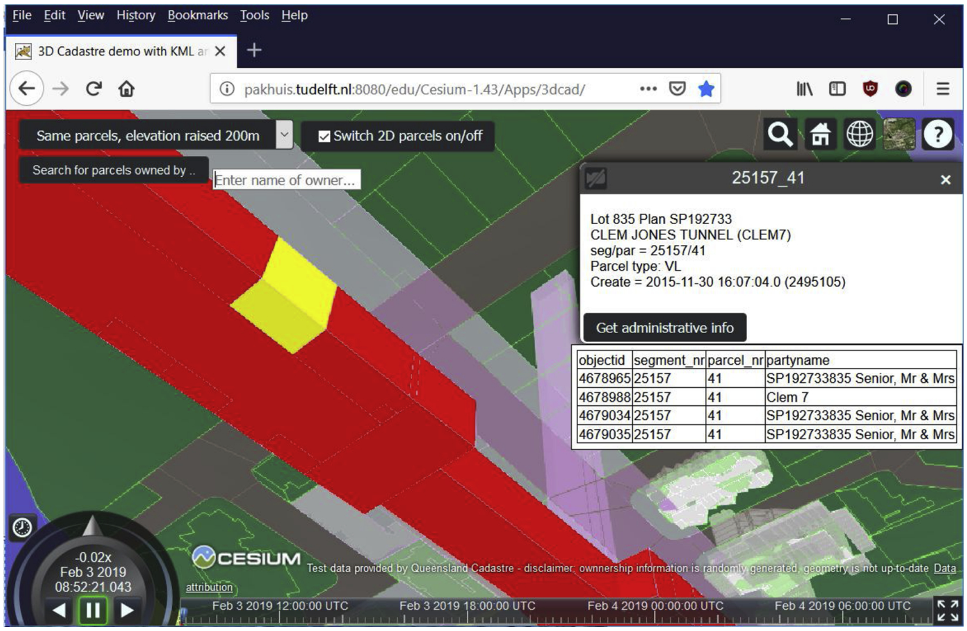


Fig. 12. The improved version of the "Get administrative info" now showing all associated records.

been corrected and making owner much easier.

3 multiple rights/owners per parcel: When a parcel has been selected, it is possible to get the legal/ administrative information. The initial prototype just fetched the first related record, the improved prototype gets all the related administrative records; see Fig. 12.

4 more direct feedback: Some actions may take some time (e.g. initial loading of the data), resulting in user being unsure what is happening. This has been improved by providing some feedback on top of the interface: "Loading …"; see Fig. 13.

5 back to initial position (Home-button): When zooming and panning with Cesium (or doing a topographic search), it is possible to get lost and the it may be hard to find back the 3D Parcels in Brisbane. The Cesium Home-buttom has now be programmed to always go back to initial view with the 3D parcels.

6 improve slow responses: when interacting the response was slow is some cases (and fast in others). Though in a web-based session there may be network / server delays it has been successfully attempted to speed-up some of the slower actions by improved implementation.

4.5. Results of second and public usability test

The final part of the usability test foresees the presentation and analysis of the responses obtained from the test users over a period from 8 November to 22 December 2018. The feedback and suggestions of the 17 members of the FIG Working Group who took part in the questionnaire will be used to improve the prototype in the future. In order to read the results in a meaningful way, it is important to know two factors that will influence the overall feedback of the questionnaire: the

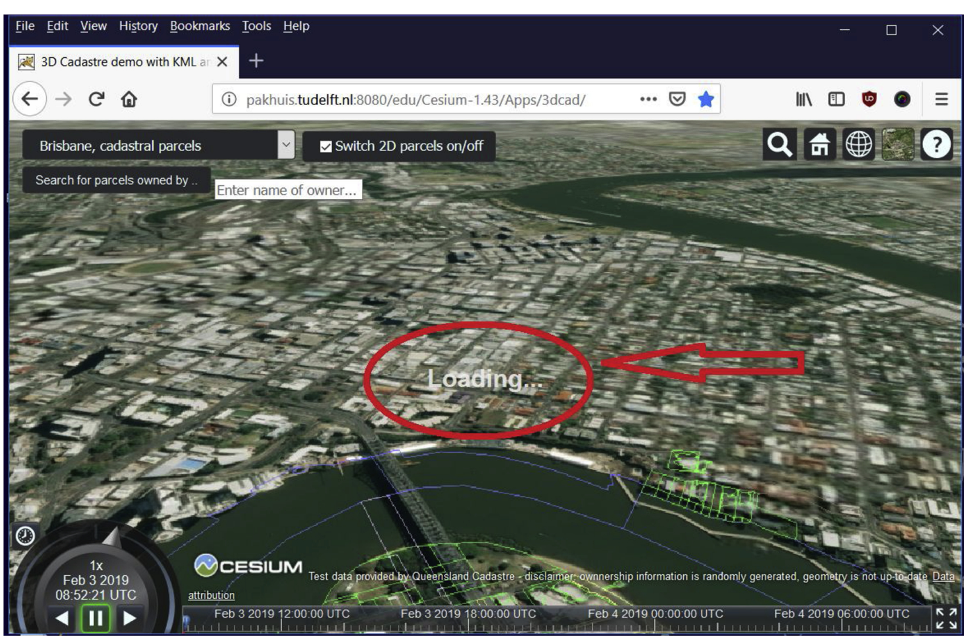


Fig. 13. Showing a feedback message during data load. Note that the message "Loading…".

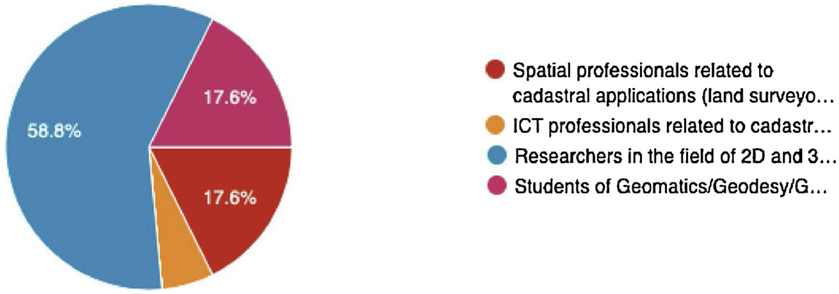


Fig. 14. Pie chart showing the representatives of the user groups who carried out the questionnaire.

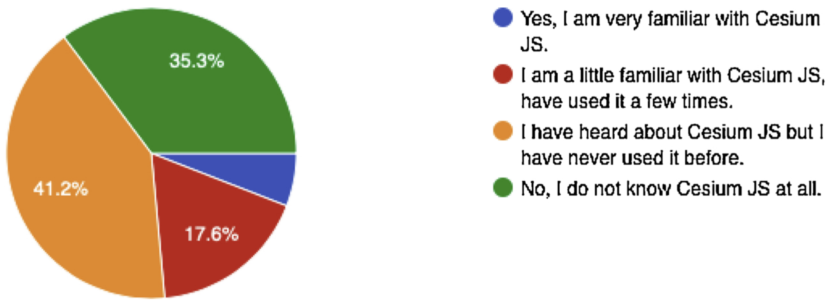


Fig. 15. Pie chart showing the knowledge of Cesium JS among the test users.

composition of the group representatives and their knowledge and familiarity with the online Cesium platform. As can be seen in the pie chart in Fig. 14 more than a half of the users are researchers in the field of 2D and 3D cadastre. In addition, a significant portion of users is composed of professionals and students in the geo-information field. Overall, these specialized representatives are supposed to have a solid background about cadastral/spatial data and geo-web applications.

Only one of the seventeen users that took the test was very familiar with Cesium JS, the rest was a little familiar or completely new to the application, as shown in the pie chart in Fig. 15. In the second usability test tasks 3 and 4 were swapped for a better flow of the usability test. The average grade as given by the users for each of the 5 different functionalities are:

- Section #1, Navigate (pan, zoom, rotate): 7.4.
- Section #2, Switch layers on/off: 7.9.
- Section #3, Get parcel information: 6.9.
- Section #4, Visualize underground parcels: 5.6.
- Section #5, Find parcels owned by person: 8.2.

More detailed analysis of the users comments per task can be found in Appendix E. The last aspect of the evaluation of the usability test, is checking the quality of the tasks as performed by the test persons. Out of the 5 tasks, the test persons typically performed 3 to 5 tasks well and provided the right answers (and wrong answers for the other tasks). The correctness score of the test persons was on the average 3.6 points (see Fig. 16).

From the responses of the test users it emerged that the main limitations of the prototype are the following: a. the slow performance of the viewer, b. the fact the icons and search options can be confusing and sometimes “buggy” (i.e. when a wrong name is inserted, no error message is shown), c. it is sometimes difficult to orientate in the viewer due to a lack of reference points (i.e. use of the North arrow), and d. the different styling of the 3D parcels according to their type (volumetric and building parcels) is not clear to some users. Moreover, it could be convenient to use different colours to distinguish between parcels below and above ground level.

On the other hand, the users also appreciated a number of functionalities of the 3D cadastre prototype, such as: a. the search functionality based on the owner’s name. Although a suggestion was to integrate the search tool with other attributes to make it more complete, b. the possibility to select a parcel and visualize cadastral information about it, c. the vertical shift of the parcels in order to visualize the details of the underground geometries, and d. the possibility to switch on/off layers to better visualize the different datasets. Additionally, some suggested to switch the visibility of the above ground and under the ground parcels separately, in order to make the distinction clearer.

Finally, the three crucial functionalities for the betterment of the prototype according to the test users are: 1. cross-section view, a tool that cuts a slice out of a volume in order to better visualize its internal subdivision, 2. object search which, in the description of the requirements, is intended as a search tool on either spatial and non-spatial data and it can be based on address, geocode, owner’s name (as the one implemented in the prototype), coordinates, etc., and 3. 3D measurement tool, which allows to estimate the dimensions of the parcels by performing different measurements, such as, area, volume, distance between points, and so on.

Comparing the results of the first and second usability test, on the average there has been made some modest progress (with improved prototype). It should be noted that one has to be careful when comparing the first and second usability test. The average appreciation of all test persons and all tasks increased from 7.0 to 7.2 (on scale 0 to 10). When looking at the individual tasks, then for 4 tasks the average difference between first and second usability test was rather small (less than 0.7 points). Only for the last task from section #5 (Find parcels owned by person), there was a much higher appreciation: raised from a 6.2 by 2.0 points to a 8.2 score. This can be explained by the improved search by owner functionality in the second version of the prototype. Looking at the correctness of the performed tasks, this has slighted dropped from 3.7 to 3.6 (on scale from 0 to 5).

5. CONCLUSIONS AND FUTURE RESEARCH

In this paper a contribution was made to solve 3D cadastral data usability challenges. A database schema has been developed to

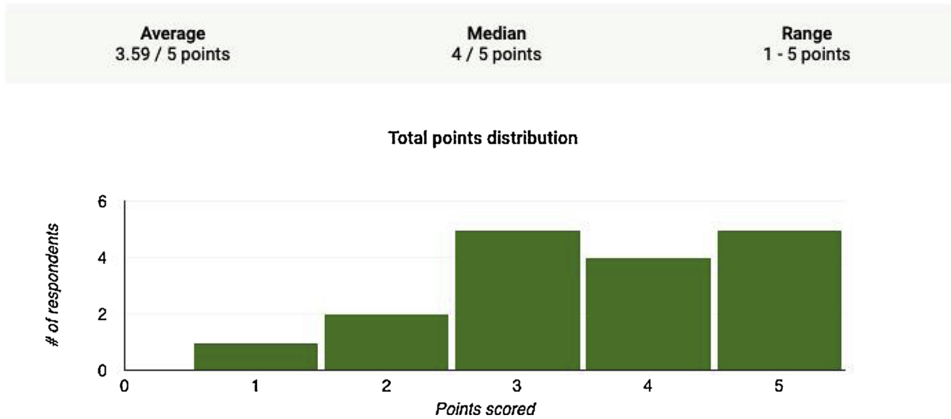


Fig. 16. Bar chart showing the total points (= number of correctly performed tasks) distribution.

accommodate a mixed 2D and 3D Cadastre, with a time component. This schema provides sufficient functionality to provide LADM-compatible views of the data for the purpose of developing visualization tools, and for the further development of the schema itself. A significant amount of 2D + t Cadastral data has been loaded, which also contained 2D + t representations of 3D parcels. A moderate number of 3D building units, and a smaller number of volumetric parcels have been semi-automated/ hand-encoded using bespoke software, and added to this database. The mixture has been displayed in KML through Cesium JS.

The second usability test also made clear: this research is very much a "work in progress", and therefore more activities are planned for the near future to resolve issues that now limit the user-friendliness and to add more functionality (see top-3 of suggestions as provided by the test persons). One of the main problems with the initial prototype was the performance and direct feedback to user actions. In the second prototype this has been improved (see Subsection 4.4). However, the feeling remained that performance was not optimal. After the second usability test the cause of this issue has been discovered: the use of KML instead of Cesium JS native 3D formats. After switching to glTF (the runtime asset format for WebGL) and b3dm (an extension to glTF specific to Cesium) not only the performance was much better, but also some mystery bugs disappeared. In addition, the prototype system should be further matured in a number of different ways: support for the temporal dimension (in storing, analyzing and visualizing 2D and 3D parcels), use even more standardized Client-Server Protocol (making it possible to mix-and-match different client tools with different server side solutions), implement some caching techniques (as for sure needed when using nation-wide data sets), and consider Topological Database Schema and topological queries (find neighbors of 2D or 3D parcels).

Our current method of data capture and conversion for 3D parcels is very demanding, time consuming and prone to error. For designed (new) objects, a more efficient and less error prone solution, especially for buildings and tunnels, is having 3D parcels defined by the (re-)use of data in the BIM environment for 3D Cadastre (Aien et al., 2011, Karki et al., 2011, Atazadeh et al., 2017, Oldfield et al., 2017, Meulmeester, 2019). In our future work we will design a web-based data submission system for uploading new 3D parcels, followed by automated validation, before final acceptance by the authorities. It has to be stressed again, that our current prototype is part of on-going design and development of a more complete web-based 3D Cadastral system. Complete functionality of this system is summarized in the list below:

- LADM compliant server (web/database),
- support for different types of 3D Cadastral objects (according to different spatial profiles),
- web-based submission (incl. conformance testing and accepting) of survey plans (GNSS, IFC,..),
- web-based query and visualization (in addition to the currently implemented top priorities),
- web-based 3D cadastral editing; e.g. a preliminary 3D parcel split (apartment),
- web-based 3D dissemination based on WFS (fitting in the SDI thinking), and
- having 3D physical reference objects (from other registrations) switched on/off.

As the prototype is work in progress, the best way to get updated is go life and experience the latest version (http://pakhuis.tudelft.nl:8080/edu/cesium/Apps/3dcad/).

Acknowledgements

The authors wish to thank Sudarshan Karki (Queensland Government, Department of Natural Resources, Mines and Water) for assistance and advice he has provided; Martin Kodde (was at Fugro, now at Geodelta) and Glen Ross-Sampson (Roames) for the provision of a surface model (DEM) of the region in question.

Appendix A

Mapping between the original DCDB attributes to LADM and the SQL views

The mappings between the DCDB object attributes and LADM are given in Tables A1,Tables A3A3,Tables A5A5,Tables A6A6. An indication of the SQL necessary to define these views can be found in Tables A2 and Tables A4A4. These will probably not be the final versions of the views, but the data content will remain constant (or equivalent) during the research.

Table A1
Mapping between QC_SpatialUnit (based on LA_SpatialUnit) and Parcel.

QC_SpatialUnit	Parcel
suID	Segment_Nr × 1000 + Parcel_Nr
area	Calc_Area
volume	Calc_Volume
dimension	3 if any Polyhedron_Face objects present, otherwise 2.
label	< empty string >
referencePoint	Point (Centroid_Lon, Centroid_Lat, Centroid_Z)
extAddressId	Primary_Name_Id
surfaceRelation	Coverage_Id

Table A2
SQL to create QC_SpatialUnit view.

```
create or replace view QC_SpatialUnit as
select
  segment_Nr * 1000 + parcel_nr as suID,
  case when plan_nr = '' then ''
      when unit_nr <> '' then plan_nr||'/'||lot_nr||'/'||unit_nr
      else plan_nr||'/'||lot_Nr end as uID,
  calc_area as area, calc_Volume as volume,
  case when centroid_z = 'NaN' then 2 else 3 end as dimension,
  '' as label,
  case when centroid_z = 'NaN'
      then st_setSRID(st_makepoint(centroid_lon, centroid_lat),4283)
      else st_setSRID(st_makepoint(centroid_lon, centroid_lat,
          centroid_Z),4283) end as referencePoint,
  primary_name_id as extAddress,
  coverage_id as surfaceRelation,
  creating_lock_nr, destroying_lock_nr
from parcel;
```

Table A3
Mapping between QC_BAUnit (based on LA_BAUnit) and Lot or Unit.

QC_BAUnit	Lot or Unit
uID	Plan_nr / Lot_nr or Plan_nr / Lot_nr / Unit_nr
name	< empty string >
type	Tenure_Status_Id (e.g. freehold, leasehold …)

Table A4
SQL to create QC_BAUnit view.

```
create or replace view QC_BAUnit as
select
   case when plan_nr = '' then ''
      when unit_nr <> '' then plan_nr || '/' || lot_nr || '/' || unit_nr
      else plan_nr || '/' || lot_Nr end as uID,
   '' as name,
   tenure_status_id as type,
   creating_lock_nr, destroying_lock_nr
from lot;
```

Table A5
Mapping between QC_BoundaryFace (based on LA_BoundaryFace) and Polyhedron_Face.

QC_BoundaryFace	Polyhedron_Face
bfID	(Segment_Nr × 1000 + Parcel_Nr) * 10000 + Face_Nr
geometry	P_Shape
locationByText	“B” for building format units, null otherwise
linestyle	Linestyle (indicates road boundaries etc)

Table A6
Mapping between QC_BoundaryFaceString (based on LA_BoundaryFaceString) and Parcel.

QC_BoundaryFaceString	Parcel
bfID	(Segment_Nr × 1000 + Parcel_Nr) * 10000 + corner number
geometry	P_Shape split into individual linestrings
locationByText	“N” for natural boundaries, null otherwise
linestyle	Linestyle (indicates road boundaries etc)

Appendix B

Encoding Building Format Spatial Units

As can be seen in Fig. B1, many building format plans show a diagram of the units (to scale), but no dimensions or connection to reference points. This is suitable for registration of the plan and the units in question because the legal definition of the property is given by the physical walls. It does not make the capture easy, and it has been necessary to develop some bespoke software for the purpose. Capture in this form (digitising from images) sets a limit on the accuracy of the data, but this is acceptable from a land administration perspective because it is not the legal boundary that is being captured, merely an indicator of the approximate position of the walls that do define the boundary. Using the bespoke software, it takes ½ to an hour per floor plan to encode a high-rise building of moderate complexity.

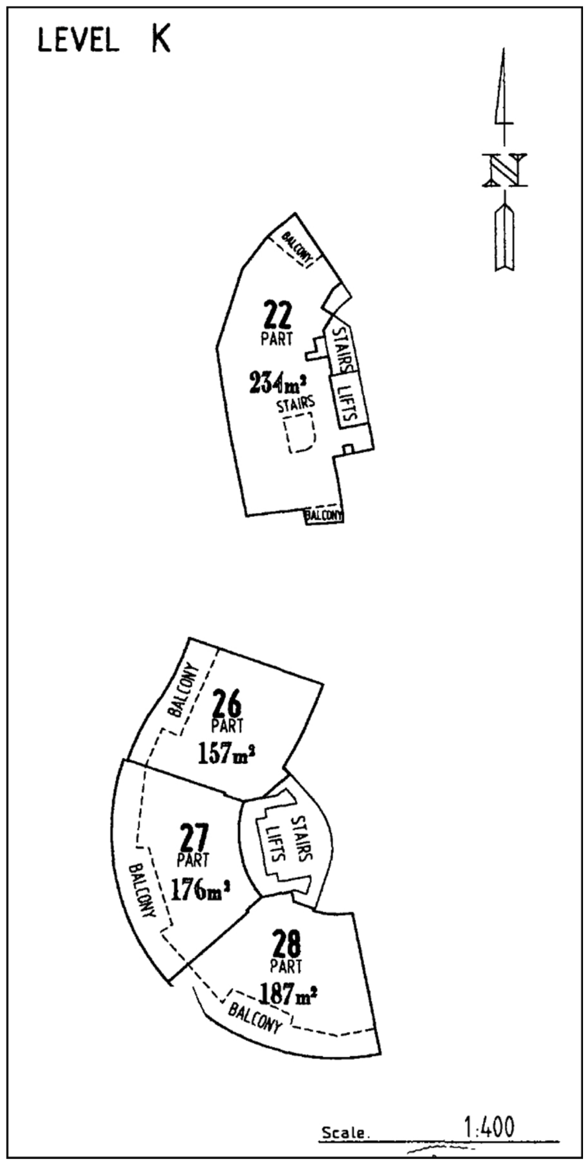


Fig. B1. Extract from a Building Format Plan.

Appendix C

Encoding Volumetric Spatial Units

In contrast to the above, the volumetric plans record the metes and bounds of the spatial units in the same form as the 2D survey plans. That is to say, bearing and distance measurements are supplied, sufficient to determine the X/Y location of all points that define the volumes. The Z values of points are supplied in later tables. Referring to Fig. C1 Diagram A, the position and location of the points numbered 22 to 29) are defined relative to one another in 2D. 3D points along the "poles" defines in diagram A are defined by assigning them Z values (e.g. 22a is assigned Z = 8.344, 22c is assigned Z = 9.737) Finally, the boundary faces are defined in an isometric view (Diagram B). The approach taken here is to encode these measurements, and connect at least two of the points to existing DCDB vertices.

Again, bespoke software has been written to accept this encoding, which also takes advantage of the fact that a large majority of 3D spatial units have a simple" polygonal slice" or" single-valued stepped slice" form (Thompson, van Oosterom et al. 2015). That is to say, in most cases of volumetric parcels in Queensland, all faces are planar, and either vertical or horizontal. In the case of Fig. C1 Diagram B, it is only necessary to encode the 2D footprint, the Z value of the bottom face, and the two top faces. The workload to encode a volumetric spatial unit is highly variable, as the complexity varies.

As previously mentioned, the 3D survey plans of the Brisbane DCDB are stored in plain PDF or TIFF files according to the Queensland regulations (DNRM, 2013) (DNRM, 2016). For this reason, an encoding process is needed to convert the information from paper to digital format. The paper format of the 3D survey plan is shown in Figs. C2–C4. Next, the data needs to be input by hand in a custom-made Excel file extracted from the 2D representation of the parcel in the DCDB, as shown in Fig. C5. Note that the corner numbers in the Excel sheet need to be encoded to correspond to the corners of the 3D parcel (i.e. 91, 9, 8, 7, 93, 92) as marked on the plan. The letter appended to a corner number indicates whether the corner is on the top ('b') or bottom ('a') of the parcel. The upper and lower footprints of the parcels are respectively counter clockwise and clockwise. This parcel has vertical walls, but the top and bottom are not horizontal.

Finally, a Java program reads the Excel file and loads the 3D parcel into the database. It is important to keep in mind that, even though the encoding of the parcels is performed automatically by a computer program, the passage from the PDF/TIFF survey plan to the Excel format has to be done manually. Therefore, it is a non-trivial and time-consuming operation.

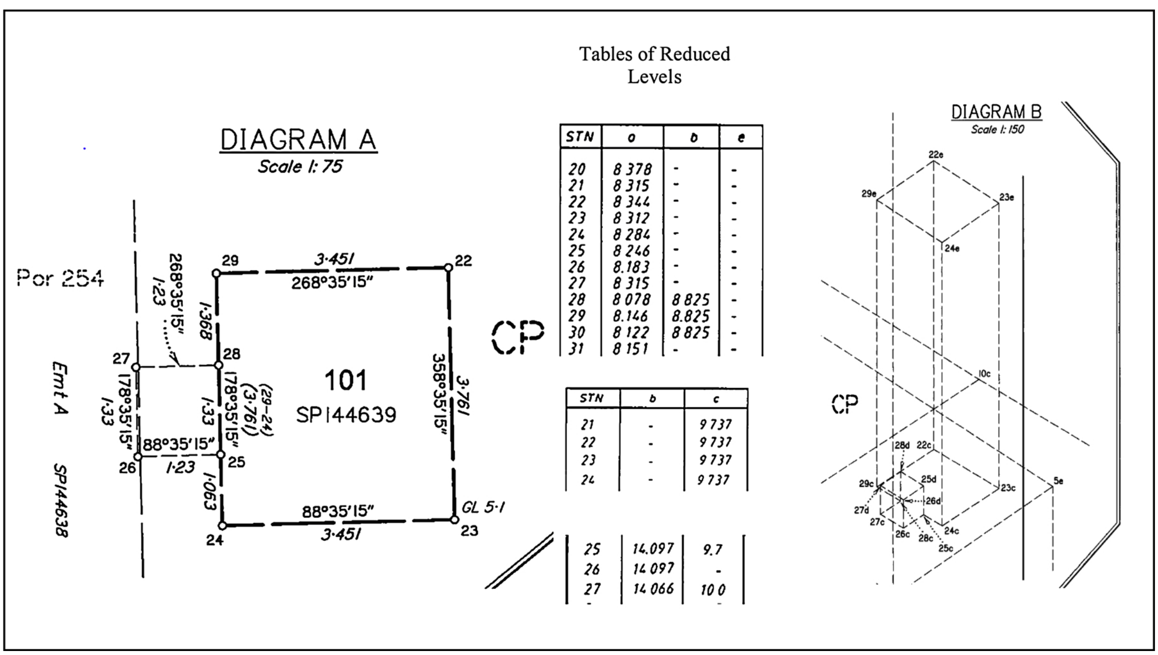
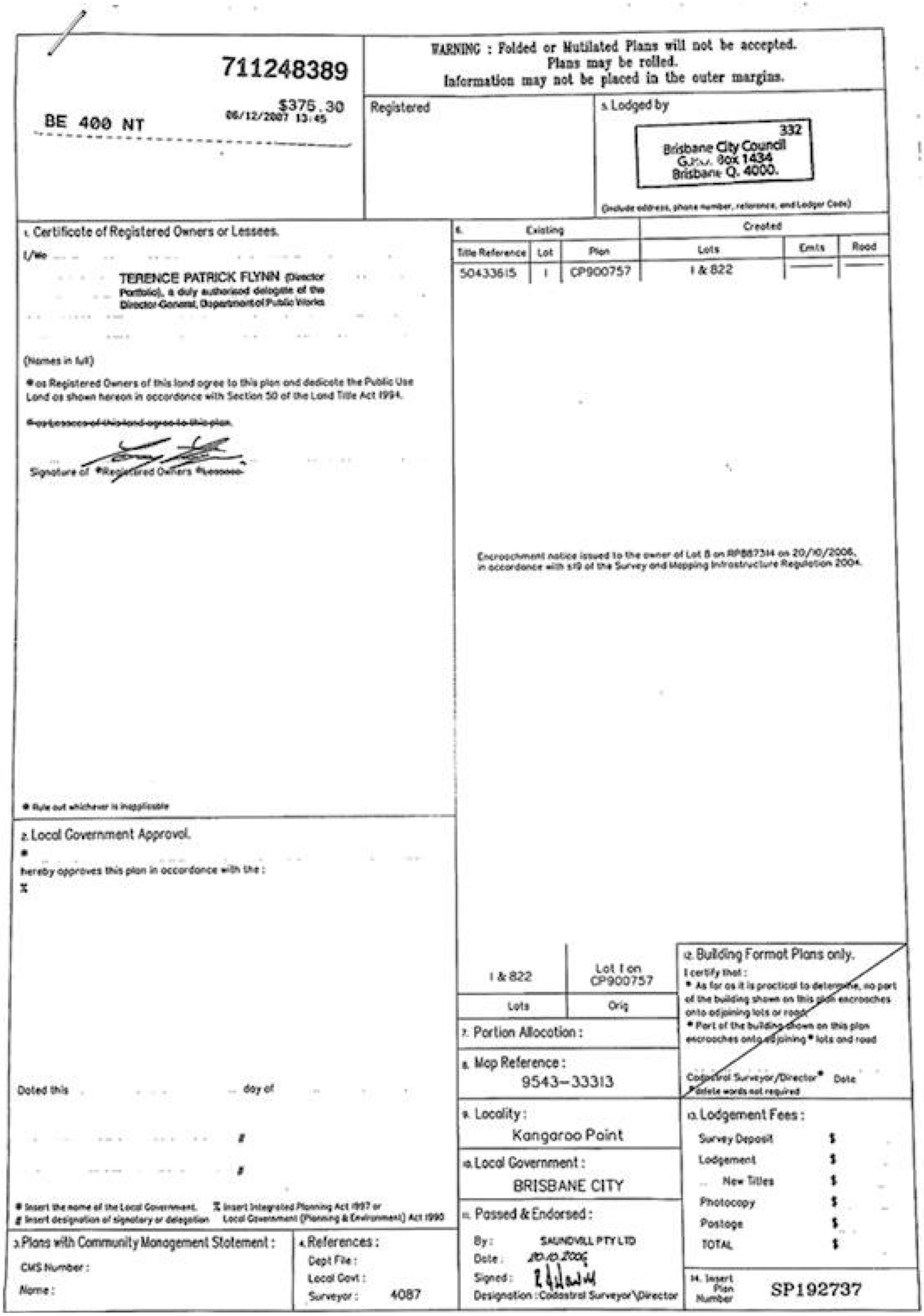


Fig. C1. Excerpts from a Volumetric Plan.



711248389

BE 400 NT

$375.30
06/12/2007 13:45

WARNING : Folded or Mutilated Plans will not be accepted.
Plans may be rolled.
Information may not be placed in the outer margins.

Registered

5. Lodged by

332
Brisbane City Council
G.P.O. Box 1434
Brisbane Q. 4000.

(Include address, phone number, reference, and Lodger Code)

1. Certificate of Registered Owners or Lessees.

I/We

TERENCE PATRICK FLYNN (Director Portfolio), a duly authorised delegate of the Director-General, Department of Public Works

(Names in full)

* as Registered Owners of this land agree to this plan and dedicate the Public Use Land as shown hereon in accordance with Section 50 of the Land Title Act 1994.

~~* as Lessees of this land agree to this plan.~~

Signature of *Registered Owners ~~*Lessees~~

* Rule out whichever is inapplicable

6. Existing			Created		
Title Reference	Lot	Plan	Lots	Emts	Road
50433615	1	CP900757	1 & 822	—	—

Encroachment notice issued to the owner of Lot 8 on RP887314 on 20/10/2006, in accordance with s19 of the Survey and Mapping Infrastructure Regulation 2004.

2. Local Government Approval.

*

hereby approves this plan in accordance with the :

%

Dated this ... day of ...

\#

\#

* Insert the name of the Local Government. % Insert Integrated Planning Act 1997 or
\# Insert designation of signatory or delegation Local Government (Planning & Environment) Act 1990

Lots	Orig
1 & 822	Lot 1 on CP900757

7. Portion Allocation :

8. Map Reference :
9543-33313

9. Locality :
Kangaroo Point

10. Local Government :
BRISBANE CITY

11. Passed & Endorsed :
By : SAUNDVILL PTY LTD
Date : 20.10.2006
Signed :
Designation : Cadastral Surveyor\Director

12. Building Format Plans only.
I certify that :
* As far as it is practical to determine, no part of the building shown on this plan encroaches onto adjoining lots or road;
* Part of the building shown on this plan encroaches onto adjoining * lots and road

Cadastral Surveyor/Director* Date
*delete words not required

13. Lodgement Fees :
Survey Deposit $
Lodgement $
... New Titles $
Photocopy $
Postage $
TOTAL $

3. Plans with Community Management Statement :
CMS Number :
Name :

4. References :
Dept File :
Local Govt :
Surveyor : 4087

14. Insert Plan Number SP192737

Fig. C2. Example of 3D survey plan - Lot 822 (page 1 of 3).

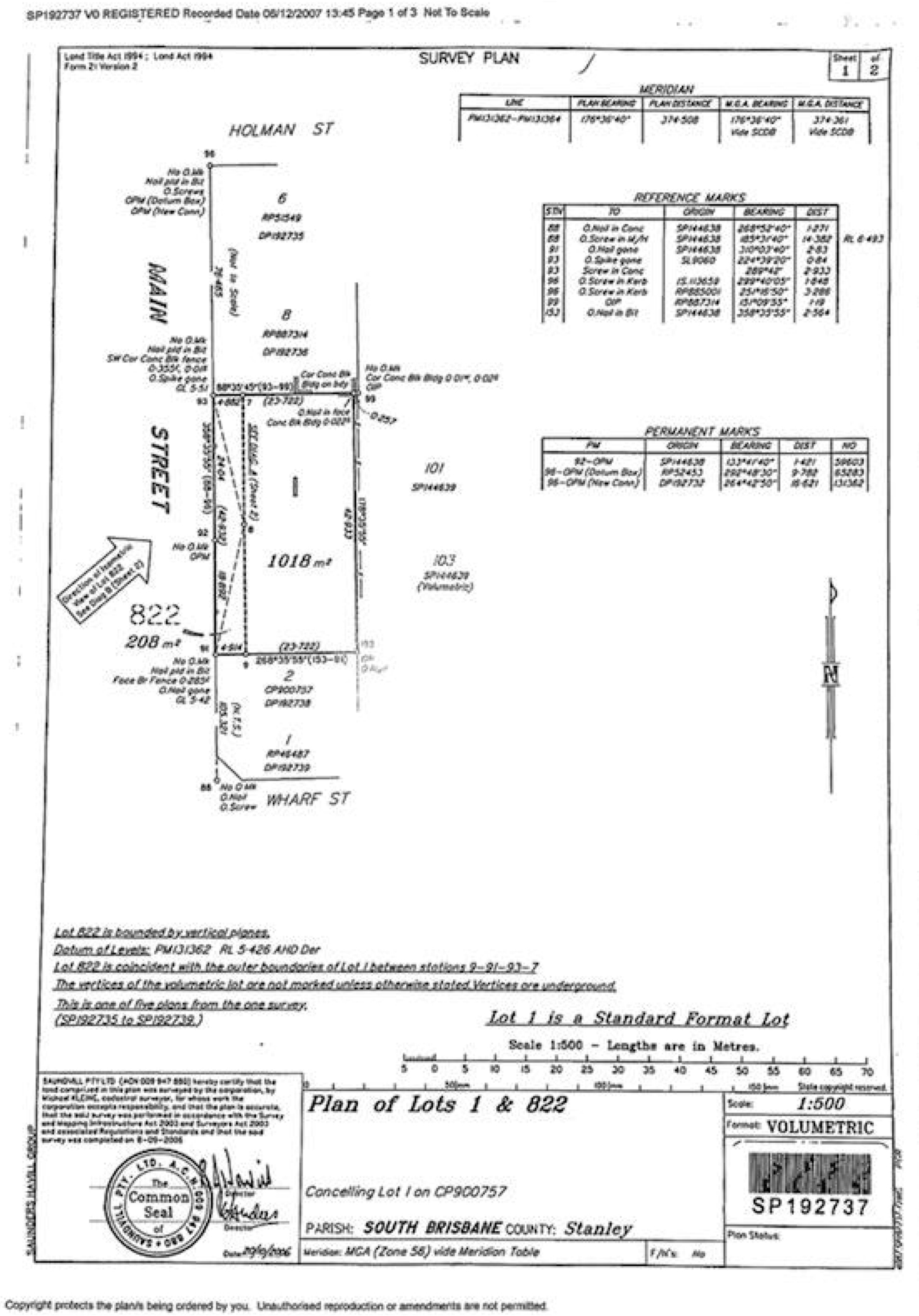


Fig. C3. Example of 3D survey plan - Lot 822 (page 2 of 3).

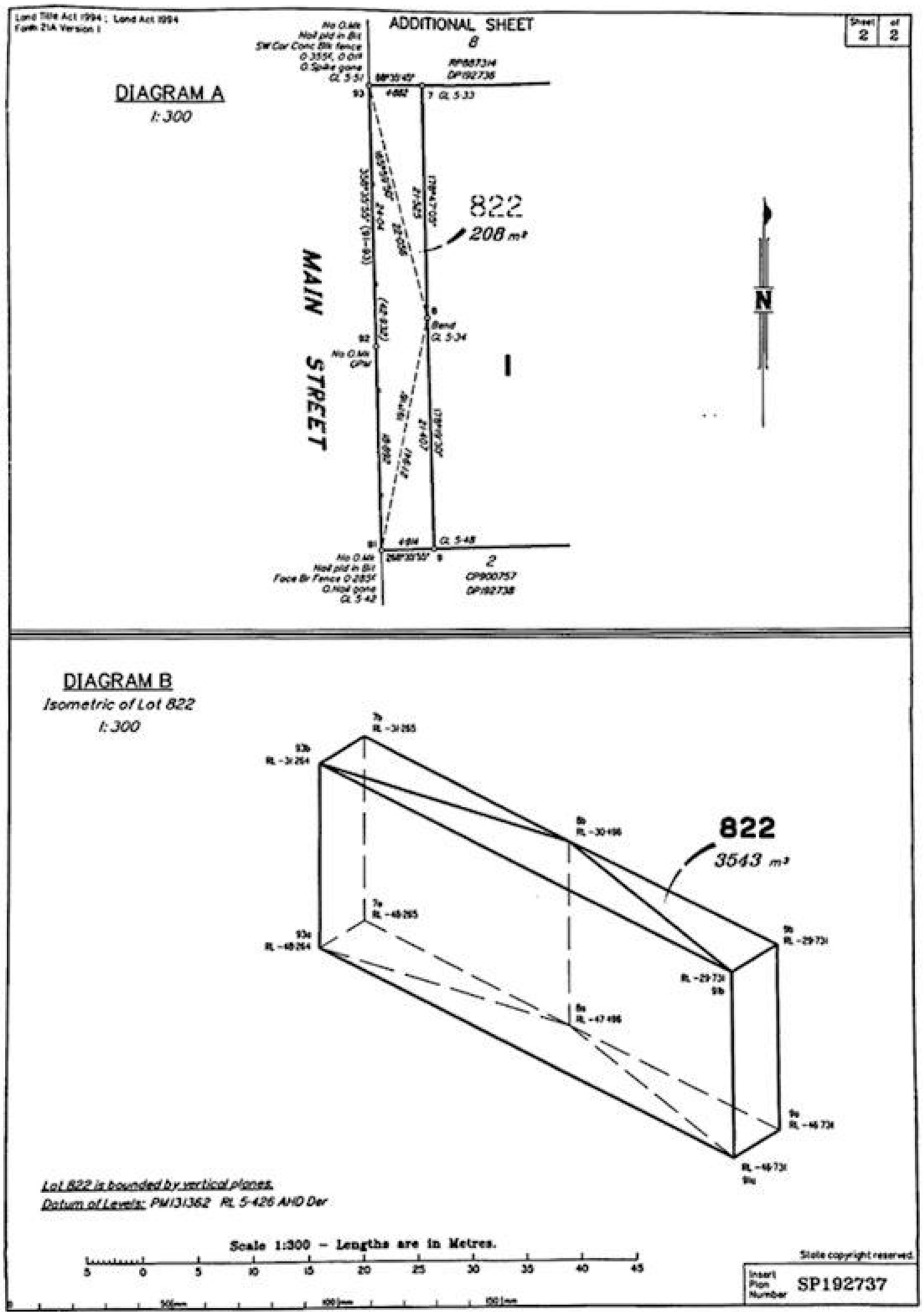


Fig. C4. Example of 3D survey plan - Lot 822 (page 3 of 3).

DCDB Parcel 822/SP192737 25159/233	Lon	Lat	Pid	Cnr Nr	X	Y	B and D	Elevations
S SW	153.0354519	-27.466975	-1	91	103	0		
SE E	153.0355025	-27.466974	-1	9	601	12		
	153.0354952	-27.466781	-1	8	530	2156		
N NE	153.0354911	-27.466598	-1	7	489	4181		
NW W	153.0354415	-27.466599	-1	93	0	4170		
	153.0354445	-27.466708	-1	92	30	2961		
****	additional edges if needed							
****	additional elevations if needed							
93 a -48.264 b -31.264								
91 a -46.731 b -29.731								
9 a -46.731 b -29.731								
8 a -47.496 b -30.496								
7 a -48.265 b -31.265								
****	parcels							
Parcels Lot 822 25159/233								
Footprint								
A 93a 7a 8a 9a 91a								
B 93b 91b 9b 8b 7b								
****	Textural data							
Date	06/12/2007							

Fig. C5. Example of Excel file - Lot 822 (manual data entry highlighted).

Appendix D

Design of the questionnaire for the usability test

This appendix gives an overview of the sections included in the questionnaire (Cemellini, 2018):

Section #1

Description: Please make sure you have a working Internet connection. Open the following link on your web browser to start up the prototype: http://pakhuis.tudelft.nl:8080/edu/Cesium-1.43/Apps/3dcad/. Before testing more advanced functionalities, it is crucial to get familiar with the basic navigation tools and view controls. Note: it is suggested to use a mouse.

Task: Navigate to the Brisbane Airport and check where it is located with respect to the river. You can do this in two ways, if you know where the airport is approximately located just pan and zoom to the location, otherwise click on the magnifier icon and type ‘Brisbane Airport, Australia’.

Opinion: Can you easily navigate through the viewer? Are the controls intuitive?

Section #2

Description: Once you are familiar with the web viewer you can start playing with the data available by changing the visibility of the layers. The prototype contains layers with cadastral parcels both in 2D and 3D. Depending on the application, you may want to visualize certain layers rather than others.

Task: Which layers can be toggled on/off?

Opinion: Was it easy to understand which layers could be toggled on/off? If applicable, please explain the difficulties you encountered.

Section #3

Description: One of the advantages of a 3D cadastral system is the possibility to store and visualize 3D underground parcels. A limitation that many globe based web-viewers have is that the camera cannot navigate under the ground surface, making impossible the visualization of subsurface parcels. For this reason, the ‘dynamic elevation tool’ has been implemented, so that the parcels can be shifted up of a defined amount to be able to see the ones under the earth surface.

Task: Suppose you want to take a look at the boundaries of the underground tunnel in our 3D Cadastre test area near the Brisbane city centre. In order to do that, you have to shift the ground surface and navigate around the parcels to visualise them in detail. What is the lowest z-value? (i.e. deepest point of the lowest 3D parcel below the surface)? Note: if you shifted the parcels you have to take in to account the amount of the shift.

Opinion: Were you able to navigate and see the details of the parcels from every angle? Could you see the coordinates on the screen?

Section #4

Description: Cadastre is not only about volumes and parcel boundaries, but it is also about legal information. Every parcel has attached legal information about owner(s), ownership rights, and identification number.

Task: Who is/are the owner(s) of the underground 3D tunnel parcels?

Opinion: Could you visualize the information easily? Was it easy to understand what to look for?

Section #5

Description: Cadastre is also about people and, in particular, owners. Even though the primary focus of the prototype is on visualization, it is important for the user to have access to the 2D and 3D parcels based on ownership of persons.

Task: Search for owner ‘Annie’ (note fake names in database for privacy). Which is the highlight colour used to show the parcels owned by Annie?

Opinion: Was the search functionality intuitive?

Finally, an additional section allows the users to suggest ways of improving the design and point out what they did like/dislike about the existing functionalities of the prototype.

Additional remarks

- What is the issue that mostly limits the user-friendliness and usability of the 3D cadastre prototype?
- What functionality did you like best? What do you think is the most useful functionality?
- Please choose 3 functionalities that, according to you, are crucial for the improvement of the 3D cadastre prototype. Here the user can choose among the items of the two lists of requirements or add his own suggestions.

Appendix E

Detailed analysis of the tasks in the the second usability test

An overview of the tasks is now given, highlighting the main feedback and the average score for each functionality. The weighted average score refers to the grade that the users gave to the usability of a functionality, based on their own experience. It should be noted that sections #3 and #4 were swapped compared to the initial usability test, because it was more logical to first retrieve the information about the owner of a parcel (easier task) and then perform more advanced operations like shifting the parcels and visualise the coordinates.

Section #1: Can you easily navigate through the viewer? Are the controls intuitive?

In general, most of the users (14 out of 17) carried out this task correctly. This implies that, even though most users are new to Cesium JS applications, the controls are intuitive and user friendly. The main suggestions for improvement concern mainly the loading time of the application and the difficulty of keeping orientation within the viewer after moving the cursor from the initial position.

The weighted average score for this functionality is: 7.4.

Section #2: Was it easy to understand which layers could be toggled on/off? If applicable, please explain the difficulties you encountered

Once again 14 out of 17 responses were correct, therefore we can say that almost the totality of the users completed the task easily. Nevertheless, some users pointed out some improvements that could be implemented to optimize the tool. First, assign a more appropriate (and explicative) name to the functionality. Second, speed up the loading time of the datasets. Third, even though the current technology does not allow looking under the earth surface in Cesium JS, the purpose of the 200 m elevation of the dataset results still confusing.

The weighted average score for this functionality is: 7.9.

Section #3: Could you visualize the information easily? Was it easy to understand what to look for?

The correct responses to this question were 9 out of 17. The users were split into two groups: the ones that could carry out the task easily and the ones who could not carry out the task at all. The latter group suggested a list of criticalities that could be improved in the prototype to make the functionality run smoothly. First of all, the slow performance of the prototype can confuse the user; therefore it would be good to shorten the loading speed. The use of cadastre specific terminology can be difficult to understand, especially for non-specialized users. Some terms, such as “partyname”, require some background knowledge to be understood. Finally, some users could not distinguish the underground parcels from the above ground ones because of the transparent finish of the earth surface.

The weighted average score for this functionality is: 6.9.

Section #4: Were you able to navigate and see the details of the parcels from every angle? Could you see the coordinates on the screen?

With only 8 correct responses out of 17, this is the only question that did not score a sufficient result. According to the test users, the cause is to be attributed to two main factors: the fact that the coordinates are obscured by the parcels or they disappear at times, and the fact that some users criticized the way in which the underground parcels are shown, defining inconvenient the computation to deduce the real height values of the parcels.

The weighted average score for this functionality is: 5.6.

Section #5: Was the search functionality intuitive?

Almost the totality of the test users responded correctly to this question. The search functionality scored 16 out of 17 correct responses and positive feedback about the usability of the tool. In addition, some users suggested a few betterments to improve the tool, such as zooming to the highlighted area and showing an error message when entering a wrong name.

It is no surprise that the weighted average score for this functionality is 8.2, the highest in the whole questionnaire.

References

AGI, 2017. (Analytical Graphics, Inc.) Retrieved November 11, 2017, from AGI. https://www.agi.com/products/open-source.

Aien, A., Rajabifard, A., Kalantari, M., Williamson, I.P., Shojaei, D., 2011. 3D Cadastres in Victoria: converting building plans of subdivision to LandXML.

Atazadeh, B., Kalantari, M., Rajabifard, A., Ho, S., Ngo, T., 2017. Building Information Modelling for High-rise Land Administration. Transactions in GIS 21 (1).

Cemellini, B., 2018. Web-based Visualization of 3D Cadastre. Delft University of Technology, Department of Architecture and the Built Environment, Delft, The Netherlands.

Cemellini, B., Thompson, R., de Vries, M., van Oosterom, P., 2018a. Visualization/dissemination of 3D Cadastre. XXVI FIG International Congress General Assembly, Istanbul, Turkey.

Cemellini, B., Thompson, R., van Oosterom, P., de Vries, M., 2018b. Usability testing of a web-based 3D Cadastral visualization system. Proceedings of the 6th International Workshop on 3D Cadastres 529–548 2018.

Cesium, J.S., 2017. Retrieved November 12, 2017. from Cesium JS. https://cesiumjs.org.

Chaturvedi, K., 2014. Web based 3D analysis and visualization using HTML5 and WebGL. University of Twente, Faculty of Geo-information Science and Earth Observation - Department of Geoinformatics, Enschede, The Netherlands.

Dam, R., Siang, T., 2018. Test Your Prototypes: How to Gather Feedback and Maximise Learning.

Danchilla, B., 2012. Beginning WebGL for HTML5. Apress.

DNRM, 2013. Queensland Government, Department of Natural Resources and Mines, Registrar of Titles Directions for the Preparation of Plans.

DNRM, 2016. Queensland Government, Department of Natural Resources and Mines, Cadastral Survey Requirements, Version 7.1.

ESRI, 2017. About CityEngine Web Viewer. Retrieved November 14, 2017, from ArcGIS. http://doc.arcgis.com/en/arcgis-online/reference/about-cityengine-web-viewer.htm.

Geobrowser 3D, 2017. Retrieved November 16, 2017. from. http://geobrowser3d.com.

Gulliver, T., Haanen, A., Goodin, M., 2017. A 3D Digital Cadastre for New Zealand and the International Opportunity. International Journal of Geo-Informaton 6 (375).

IGN, 2015. Retrieved November 5, 2017. from Institut national de l'information géographique et forestière:. http://www.ign.fr/institut/innovation/itowns.

ISO, 2018. Ergonomics of human-system interaction - Part 11: Usability: Definitions and concepts. Tratto da https://www.iso.org/obp/ui/#iso:std: iso:9241:-11:ed-2:v1:en. .

ISO-TC211, 2012. IS 19152, Geographic information — Land Administration Domain Model (LADM).

iTowns, 2017. Retrieved November 4, 2017. from. http://www.itowns-project.org.

Ivory, M., 2003. Usability Testing Methods - Researchers' and Pratictioners' Perspectives. In: In: Ivory, M.J. (Ed.), Automated Web Site Evaluation Vol. 3. Springer, Dordrecht, The Netherlands, pp. 23–37.

Jensen, C.S., Clifford, J., Elmasri, R., Gadia, S.K., Hayes, P., Jajodia, S., Dyreson, C., Grandi, F., Kafer, W., Kline, N., Lorentzos, N., Mitsopoulos, Y., Montanari, A., Nonen, D., Peressi, E., Pernici, B., Roddick, J.F., Sarda, N.L., Scalas, M.R., Segev, A., Snodgrass, R.T., Soo, M.D., Tansel, A., Tiberio, P., Wiederhold, G., 1994. A Consensus Glossary of Temporal Database Concepts.". ACM SIGMOD 23 (1), 52–64.

Kalogianni, E., 2016. Linking the legal with the physical reality of 3D objects in the context of land administration domain model. Master's thesis. Delft University of Technology - Department of OTB, Delft, the Netherlands.

Karki, S., Thompson, R., McDougall, K., Cumerford, N., van Oosterom, P., 2011. ISO Land Administration Domain Model and LandXML in the Development of Digital Survey Plan Lodgement for 3D Cadastre in Australia. In: van Oosterom, P., Fendel, E., Stoter, J., Streilein, A. (Eds.), Proceedings 2nd International Workshop on 3D Cadastres, pp. 65–84 Delft,.

Karki, S., Thompson, R., McDougall, K., 2013. Development of validation rules to support digital lodgement of 3D cadastral plans. In Computers, Environment and Urban Systems 40, 34–45.

Khoo, V.H.S., 2012. Towards "Smart Cadastre" that Supports 3D Parcels. 3rd International Workshop on 3D Cadastres: Development and Practices 243–252.

Khronos Group, 2017. WebGL overview. Retrieved November 6, 2017, from Khronos Group:. https://www.khronos.org/webgl/.

Mackey, A., 2017. Why you should learn WebGL. Retrieved November 6, 2017, from Pluralsight:. https://www.pluralsight.com/blog/software-development/webgl-basics.

Meulmeester, R.W.E., 2019. BIM Legal: Proposal for defining legal spaces for apartment rights in the Dutch cadastre using the IFC data model, Master's thesis. University of Technology, Geomatics, Delft, pp. pp. 127.

Nielsen, J., 2014. Turn User Goals into Task Scenarios for Usability Testing. Tratto da Nielsen Norman Group - Evidence-Based User Experience Research, Training, and Consulting. https://www.nngroup.com/articles/task-scenarios-usability-testing/.

Nielsen, J., 2000. Why You Only Need to Test with 5 Users. Tratto da https://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/. .

OSM buildings, 2017. Retrieved November 14, 2017. from. https://osmbuildings.org.

Oldfield, J., van Oosterom, P., Beetz, J., Krijnen, Th.F., 2017. Working with Open BIM Standards to Source Legal Spaces for a 3D Cadastre, In: ISPRS International Journal of Geo-Information. MDPI AG 6 (11), pp. 19.

Pereira, K., 2013.). Water Simulation on WebGL and Three.js. Bachelor thesis. The University of Southern Mississippi, Department of Computer Science.

Pouliot, J., Hubert, F., Wang, C., Ellul, C., Rajabifard, A., 2016. 3D Cadastre Visualization: Recent Progress and Future Directions. 5th International FIG 3D Cadastre Workshop - Athens, Greece. pp. 337–360.

Pouliot, J., Ellul, C., Hubert, F., Wang, C., Rajabifard, A., Kalantari, M., Shojaei, D., Atazadeh, B., van Oosterom, P., de Vries, M., Ying, S., 2018. Best Practices 3D Cadastres - Extended version. International Federation of Surveyors (FIG).

Queensland Government, 2018. Digital cadastral data. Retrieved from https://www.business.qld.gov.au/running-business/support-assistance/ mapping-data-imagery/data/digital-cadastral.. .

Ribeiro, A., Duarte de Almeida, J.-P., Ellul, C., 2014. Exploring CityEngine as a Visualisation Tool for 3D Cadastre. 4th International Workshop on FIG 3D Cadastres 197–218.

Rovers, A., Meijers, M., van Oosterom, P., 2017. Using a generic spatial access method for caching and efficient retrieval of vario-scale data in a server-client architecture. Bregt, Arnold, Sarjakoski, Tapani, van Lammeren, Ron, Rip, Frans (Eds.), Proceedings of the 20th AGILE Conference on Geographic Information Science pp. 6.

Rubin, J., Chisnell, D., 2008. Handbook of usability testing - How to Plan, Design, and Conduct Effective Tests.

Shojaei, D., Kalantari, M., Bishop, I.D., Rajabifard, A., Aien, A., 2013. Visualization requirements for 3D cadastral systems. Computers, Environment and Urban Systems 41, 39–54.

Shojaei, D., 2014. 3D Cadastral Visualisation: Understanding Users' Requirements. Phd thesis. University of Melbourne, Centre for Spatial Data Infrastructures (SDIs) and Land Administration,Department of Infrastructure Engineering - School of Engineering, Melbourne.

Snodgrass, R.T., Böhlen, M.H., Jensen, C.S., Steiner, A., 1998. Transitioning temporal support in TSQL2 to SQL3. Lecture Notes in Computer Science 1399, 150–194.

Stoter, J., Ploeger, H., van Oosterom, P., 2012. 3D cadastre in the Netherlands: Developments and international applicability. Computers, Environment and Urban Systems 40, 56–67.

Thompson, R., van Oosterom, P., Karki, S., Cowie, B., 2015. A Taxonomy of Spatial Units in a Mixed 2D and 3D Cadastral Database. FIG Working Week 2015. pp. 20 Sofia, Bulgaria.

Thompson, R., van Oosterom, P., Soon, K.H., 2017. LandXML Encoding of Mixed 2D and 3D Survey Plans with Multi-Level Topology. International Journal of Geo-Information 6 (6).

Thompson, R., van Oosterom, P., Cemellini, B., de Vries, M., 2018. Developing an LADM compliant dissemination and visualization system for 3D Spatial Units. 7th International FIG workshop on tha Land Administration Domain Model. Zagreb.

Thompson, R., Van Oosterom, P., 2019. A Suggested Terminology for Poiint-like Entities in a Bi-Temporal Representation of 2D and 3D Land Administration Data. van Oosterom, P., Lemmen, C., Rahman, A.A., Lumpur, Kuala (Eds.), 8th Land Administration Domain Model Workshop.

van den Berg, C., Tuijnman, F., Vijlbrief, T., Meijer, C., Uitermark, H., van Oosterom, P., 1999. Multi-server Internet GIS: Standardization and Practical Experiences (1999) In: Goodchild, M.F., Egenhofer, M.J., Fegeas, R., Kottman, C.A. (Eds.), Interoperating Geographic Information Systems. Kluwer Academic Publishers, Boston, pp. 365–378 International Conference and Workshop on Interoperating Geographic Information Systems, Santa Barbara, California, USA, December 3-4 and 5-6, 1997).

van Oosterom, P., 1997. Maintaining Consistent Topology including Historical Data in a Large Spatial Database. Auto Carto 13, Seattle, WA. .

van Oosterom, P., Maessen, B., Quak, W., 2000. Spatial, Thematic and Temporal Views. Proceedings 9th International Symposium on Spatial Data Handling - Beijing, China. pp. 8b.37–8b.52.

van Oosterom, P., Stoter, J., 2006. In: T. a. Group (Ed.), 3D Cadastre in an International Context - Legal, organizational and technological aspects. CRC Press.

Vandysheva, N., Sapelnikov, S., van Oosterom, P., de Vries, M., Spiering, B., Wouters, R., Hoogeveen, A., Penkov, V., 2012. The 3D Cadastre Prototype and Pilot in the Russian Federation. FIG Working Week.

Wang, C., Pouliot, J., Hubert, F., 2016. How users perceive transparency in the 3D visualization of cadastre: testing its usability in an online questionnaire. GeoInformatica 21, 599–618.

WebGL Earth. (2017). Retrieved from WebGL Earth: https://www.webglearth.com/.

Ying, S., Guo, R., Li, L., He, B., 2012. Application of 3D GIS to 3D Cadastre in Urban Environment. 3rd International Workshop on 3D Cadastres: Developments and Practices 253–272.

Ying, S., Guo, R., Li, W., Yang, J., Zhao, Z., Li, L., 2016. Visualization for the Coherent Set of 3D Property Units. 5th International FIG 3D Cadastre Workshop - Athens, Greece. pp. 361–372.