Visualization/dissemination of 3D Cadastral Information

9-5-2018

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

TS05C: Developing 3D Models for a 3D World, Tuesday, 8 May 2018
XXVI FIG Congress, 6-11 May 2018, İstanbul Congress Center, Turkey

Presentation outline

1. Introduction
2. Requirements overview
3. Webviewer options
4. Data preparation
5. Initial results
6. Future work

Introduction

3D CADASTRE

WEB

VISUALIZATION

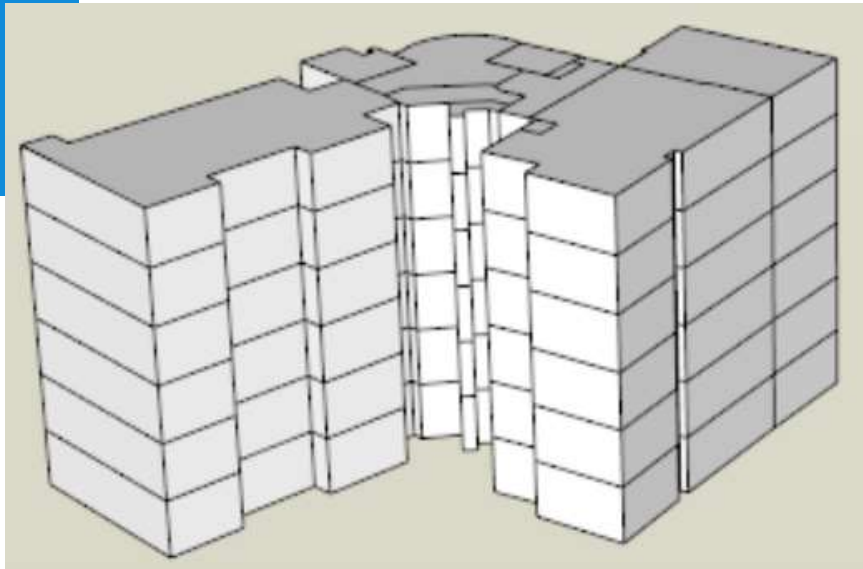
More and more countries are developing 3D cadastre

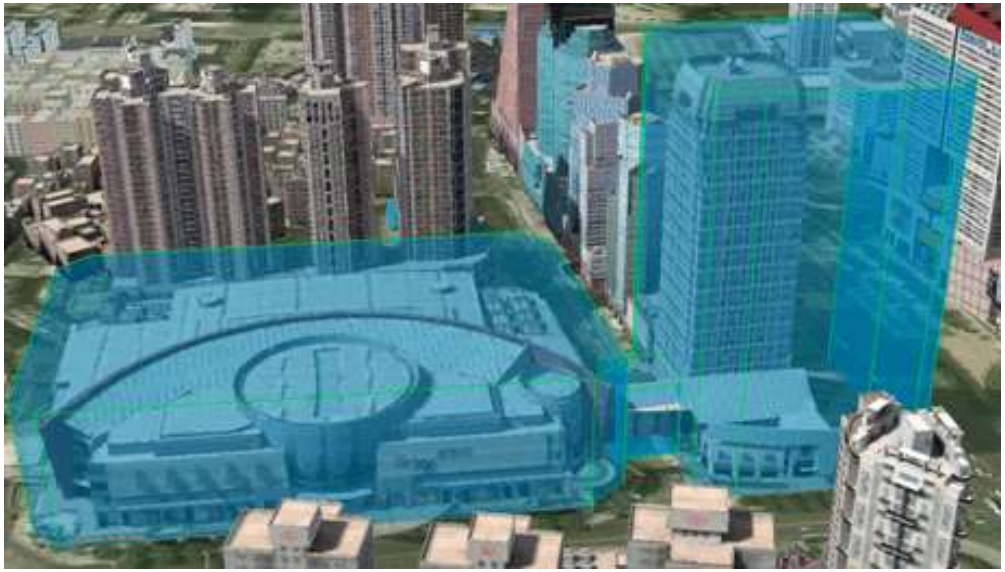
Cadastre is about making the information available to the public and therefore visualizing it on the web is very suitable

...but visualization is still a challenge!

The main issues are: occlusion, distortion, unbounded volumes, perception of position, size and shape of an object.

Introduction





Occlusion issues:

- Compromise the correct perception of parcels
- Impossibility to see all parcels

Combination of reference objects and legal boundaries:

- Good for orientation and reference purposes ☺
- A further challenge regarding occlusion ☹

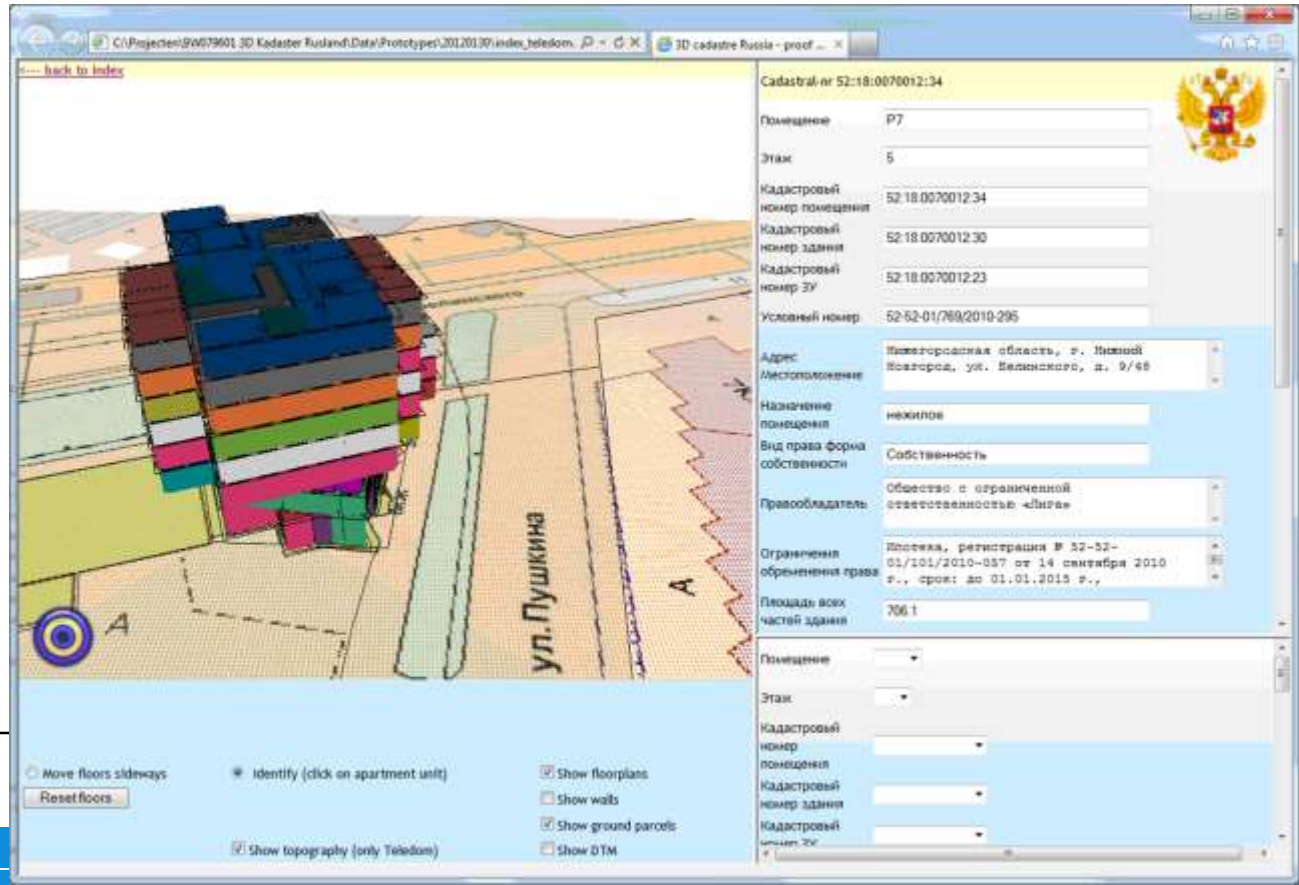
Our related work: The Russian Prototype

Aim: display 3D objects and their legal boundaries

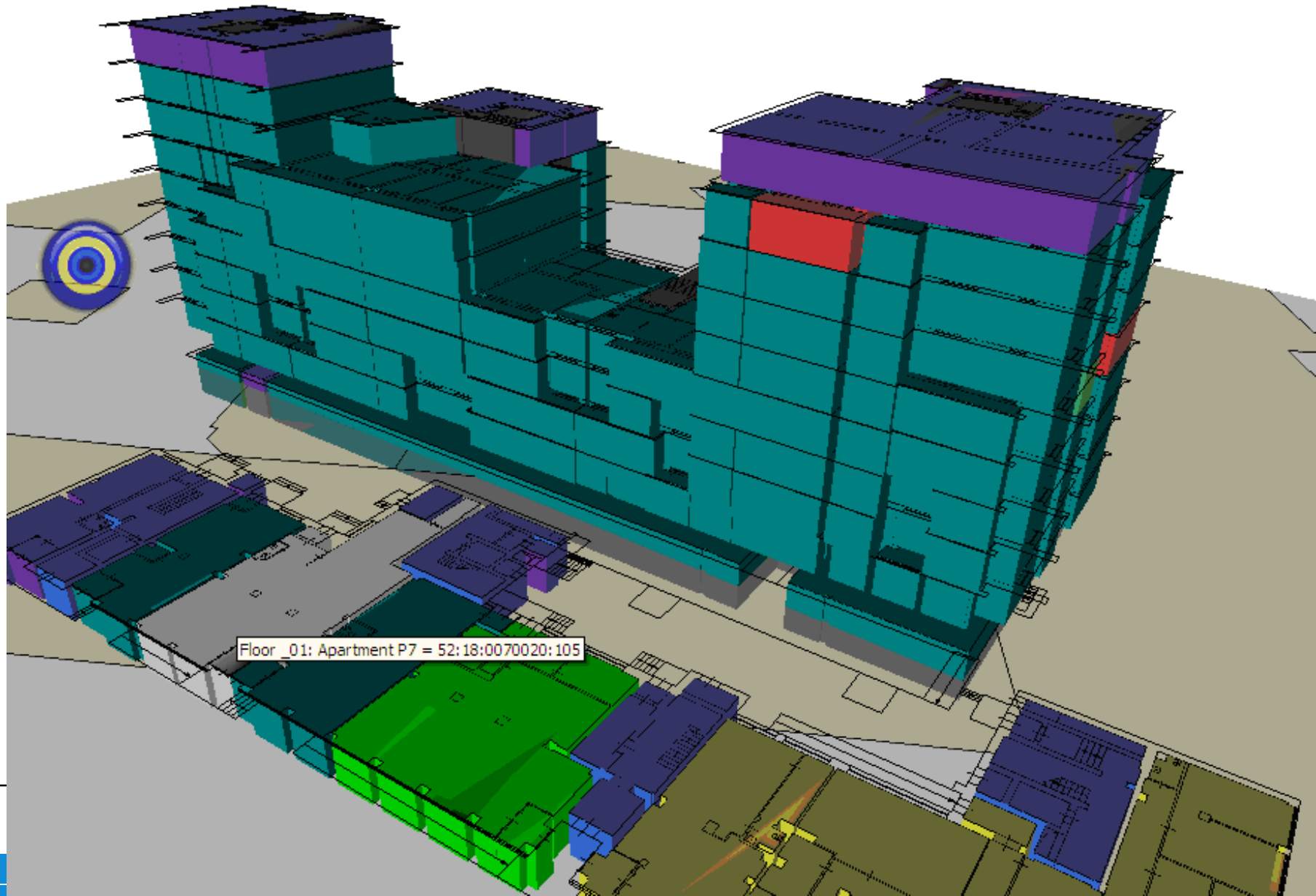
- Geometry stored in X3D files
- Corresponding administrative information stored in XML files
- Requires the installation of a plug-in in the web browser
- Slide layers

Issues:

- Plug-in installation
- Information not in a unique place (i.e. DBMS)



Slide-out interface (look inside)



Presentation outline

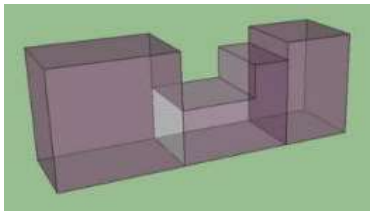
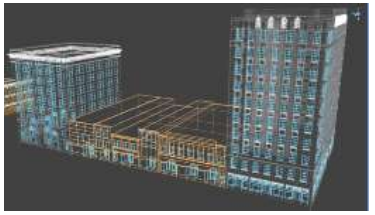
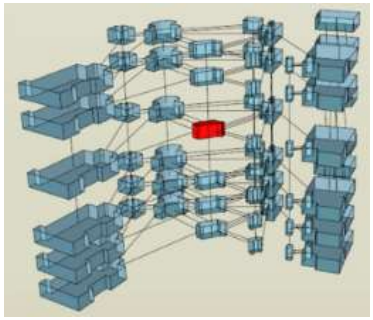
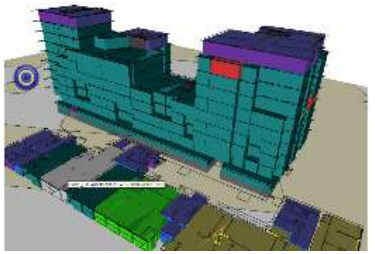
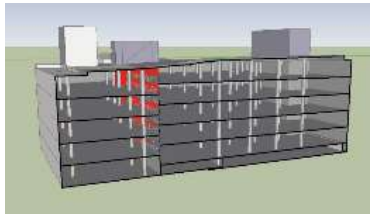
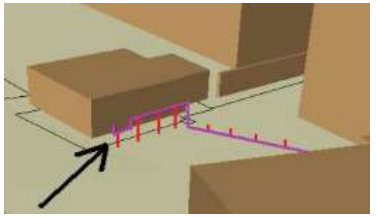
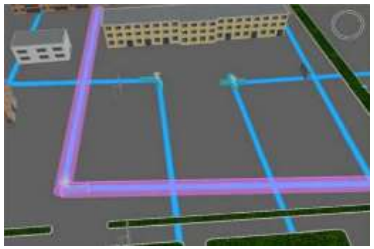
1. Introduction
2. Requirements overview
3. Webviewer options
4. Data preparation
5. Initial results
6. Future work

Requirements inventory

- Based on earlier project team experience and literature study
- Two categories:
  1. Requirements for 3D visualization
  2. Requirements for the web viewer
- Three levels (in our project):
  1. Yes → green
  2. Maybe → black
  3. No → red

Requirements for 3D visualization

- Navigation tools and view controls
- Integrating topography and reference objects
- Transparency
- Object selection
- Object search
- Wireframe display
- Explode view
- Sliding
- Cross-section view
- Visualization cues
- 3D measurement tools
- 3D buffer
- Display partly unbounded objects and ‘complex’ geometries

Requirements for the web viewer

- Platform and browser independence
- Handling massive data and caching/tiling between server and client
- 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
- 2D overview map (orientation)

Presentation outline

1. Introduction
2. Requirements overview
3. Webviewer options
4. Data preparation
5. Initial results
6. Future work

Comparison WebGL platforms

- **Cross-platform open source** web standard for a low-level **3D graphics API**
- Brings 3D into the web browser **without** the installation of **a plug-in**
- **Supported by all** major **browsers**
- Performs rather well in case of **complex visualizations**

Study of available WebGL based platforms:

CESIUM

iTowns

CityEngine Web Viewer

OSM Buildings

GeoBrowser 3D

WebGL Earth

Comparison: 3D visualization requirements

	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	?	?	?	?	?	?

Comparison: web viewers' requirements

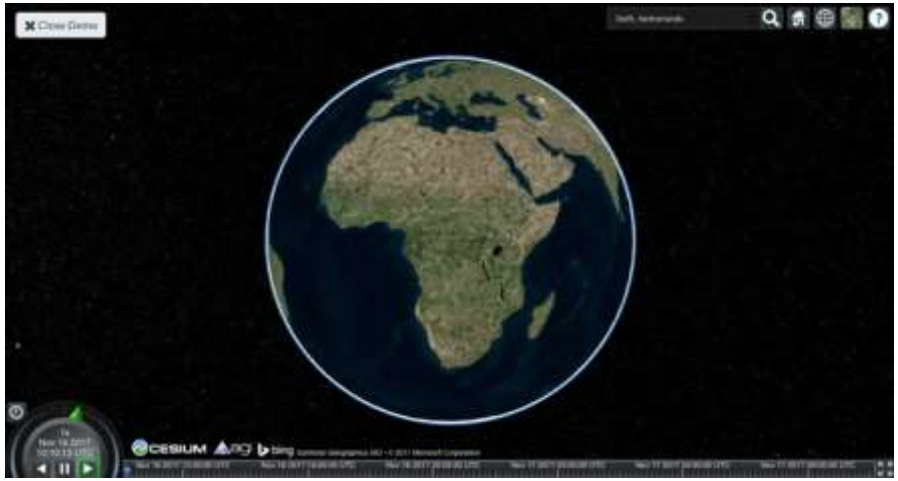
	Platforms					
Requirements	iTowns	Cesium JS	OSM buildings	WebGL Earth	Geobrowser 3D	ESRI Cityengine Web Viewer
Platform and browser independence	✔	✔	✔	✔	✔	✔
Handling massive data and caching/tiling between server and client	✔	✔	✔	✔	✔	? but foresees the possibility of handling massive cadastral data
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)	✔	✔	?	?	?	?

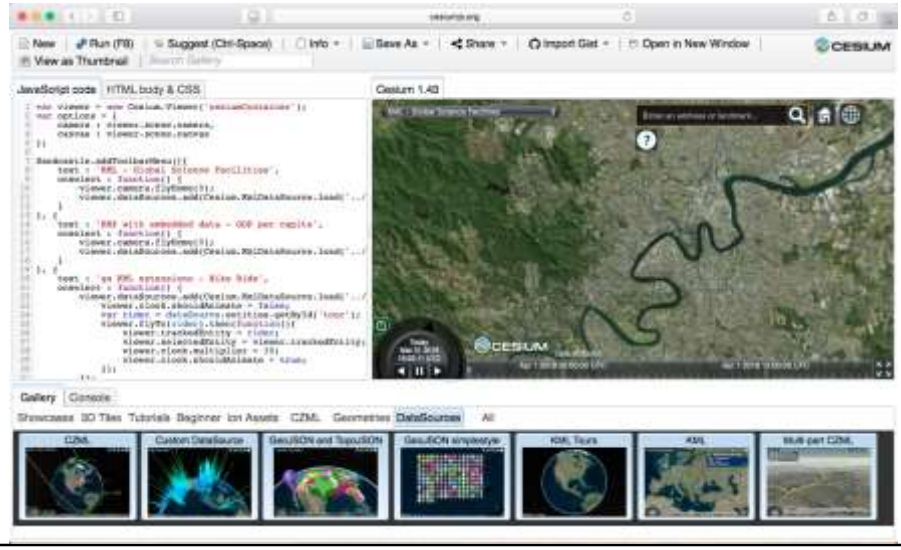
Selection of the most suitable platform

Cesium JS is a WebGL based open-source JavaScript library to create 3D geo applications

Cesium has active forum to help developers

Sandcastle: live code editor and example gallery





Presentation outline

1. Introduction
2. Requirements overview
3. Webviewer options
4. Data preparation
5. Initial results
6. Future work

The Queensland Cadastre

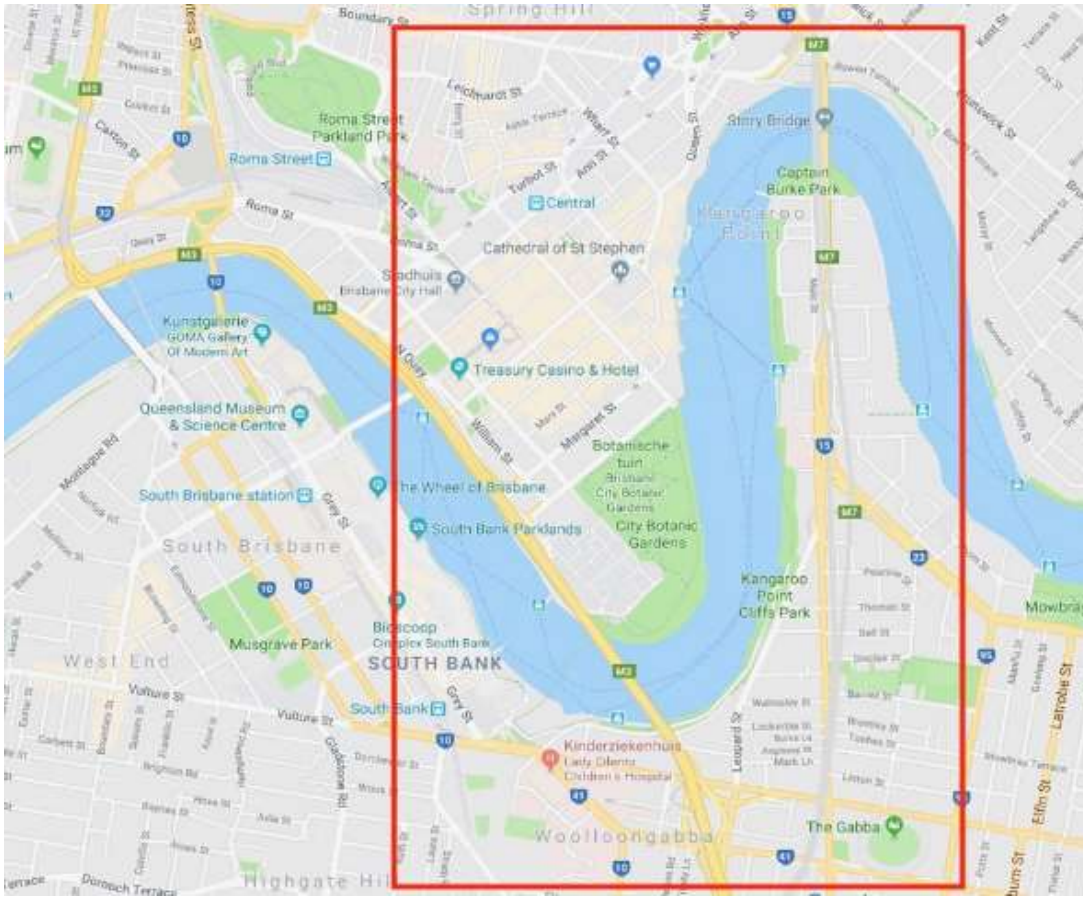
- Since 1997 possible to create parcels with 3D geometry
- Cadastral map only contains footprints of 3D parcels
- 3D survey plans are stored as paper drawings/PDF files

Issues:

- The 3D parcels cannot be interactively visualized
- No spatial validity checks possible
- 3D information is not stored together with 2D cadastral map

Area of interest

Brisbane City centre (Story Bridge and Kangaroo Point area)



The **Queensland Digital Cadastral Database (DCDB)** has a long tradition and the biggest amount of data available so far.

Data

- **2D cadastral parcels** (from Queensland Cadastre)
- **3D survey plans** (from Queensland Cadastre), either in *'building format'* and in *'volumetric format'*. The volumes will represent three main categories in the real world: buildings, tunnels and air space.
- Registration of **rights**, restrictions and responsibilities
- **Persons/parties**
- **Elevation data** (DTM or contour lines)
- **Reference data** (topographic objects in 2D or 3D)

- RRRs/parties if available, otherwise 'make-up'

3D parcels form building format or volumetric survey plans

2D Parcels

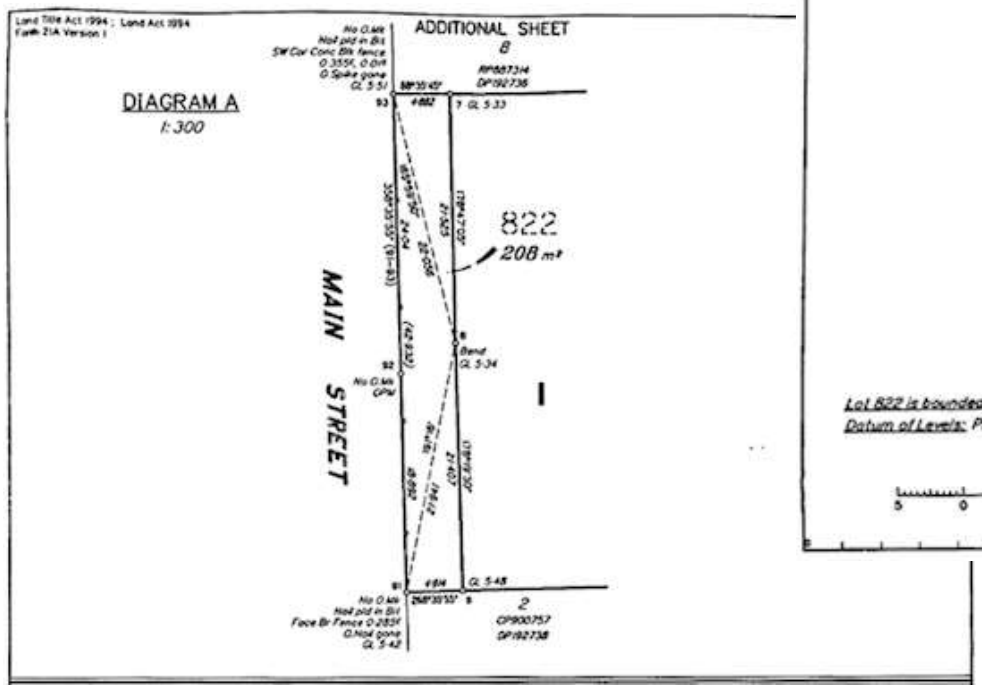
Volumetric parcel from below ground to 200m above

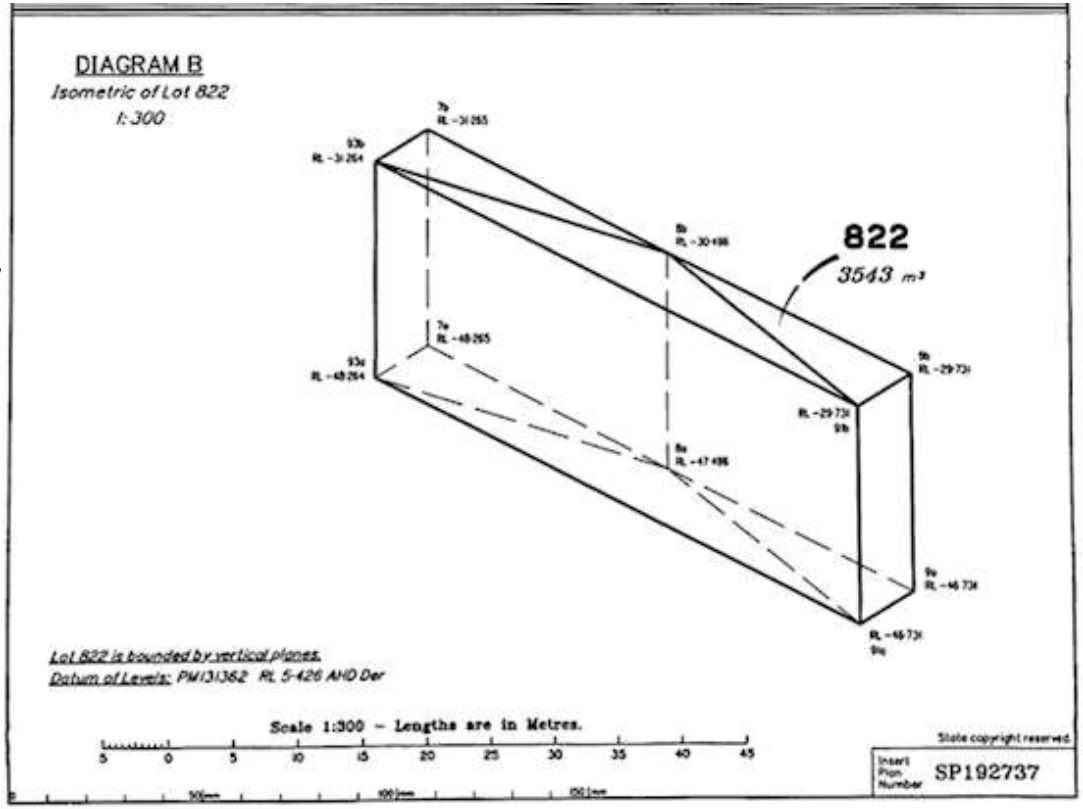
Volumetric parcel below Building Format parcels

Building format Parcels

Volumetric parcel above ground

Data preparation
(encoding of the 3D survey plans)



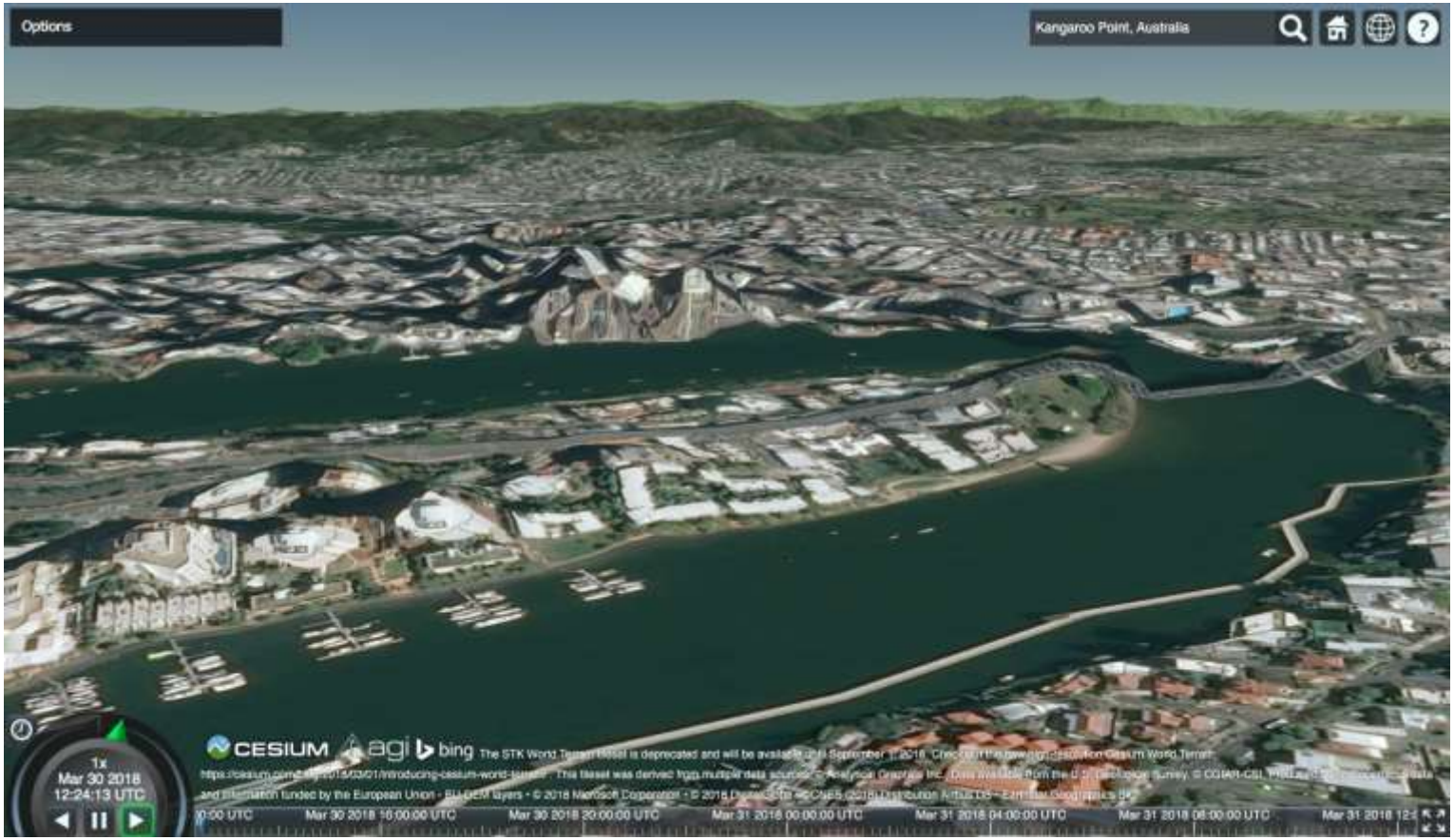


Data preparation
(encoding of the 3D survey plans)

1. Key-in essentials from survey plan (in structured xls)
2. Own software to convert to 3D faces in KML

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							

DTM selection



STK world terrain is a worldwide terrain elevation tileset provided by Cesium JS, but it is not accurate enough to place parcels on top of it

Better DTM

DTM provided by Fugro as a TIFF file,
it has a resolution of 50 centimeters

TIFF to KML:

1. From (geo)tiff to ESRI ASCII grid
2. From *.asc to space/comma delimited *.xyz
3. From ASCII XYZ to KML

Presentation outline

1. Introduction
2. Requirements overview
3. Webviewer options
4. Data preparation
5. Initial results
6. Future work

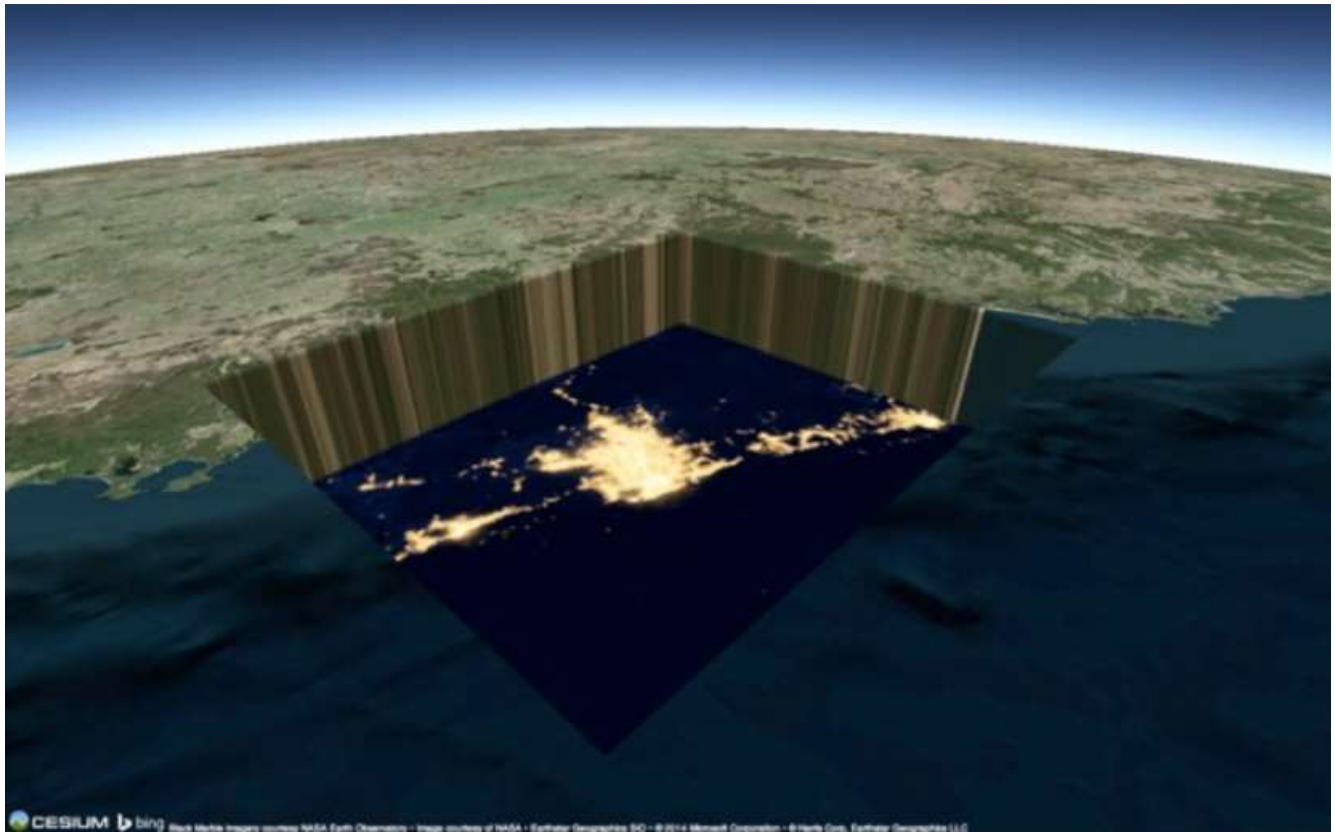
What's a good way to visualize underground parcels?

Cesium JS does not support underground view since the camera cannot go under the earth surface.

Three possible options to overcome this issue:

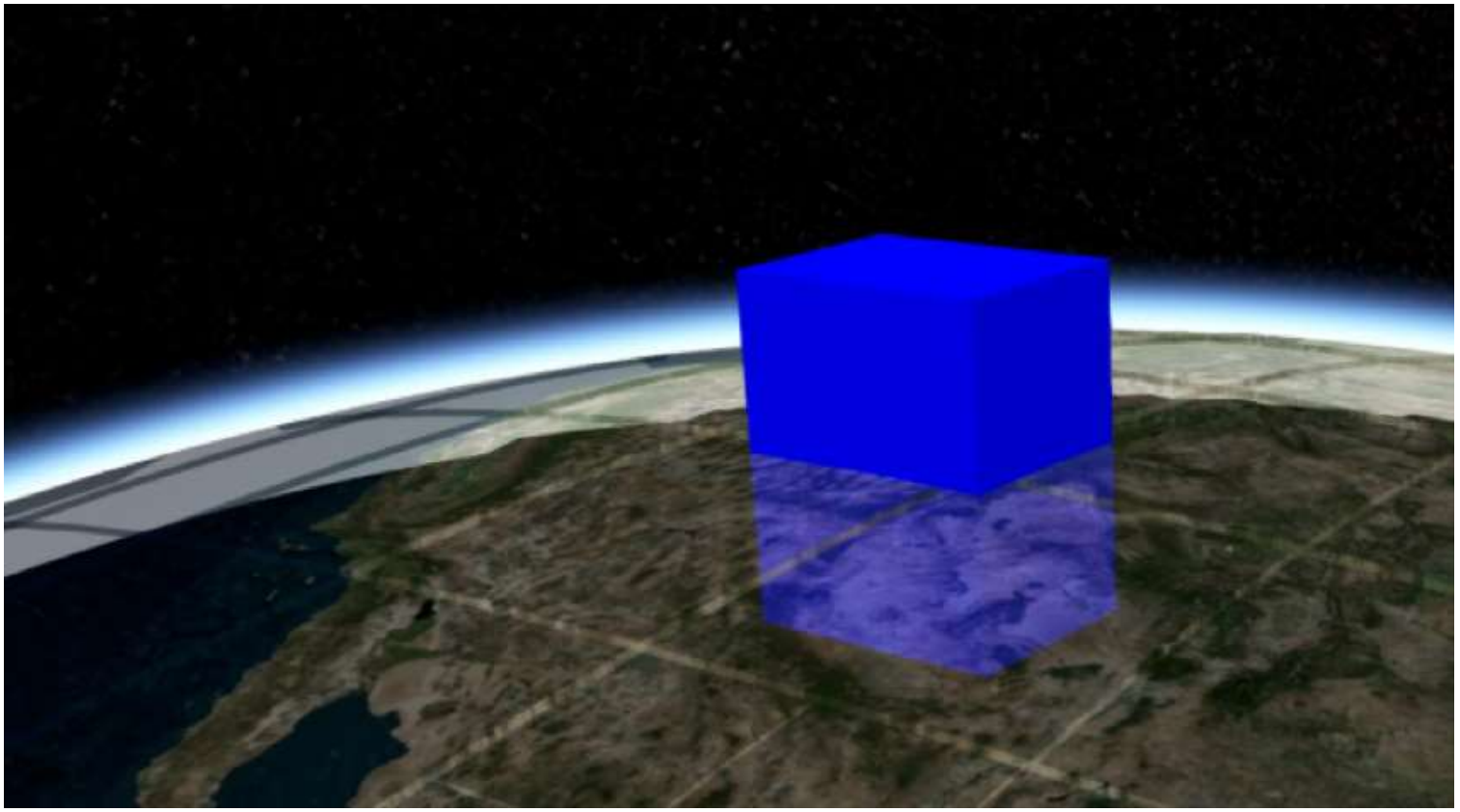
1. **Ground push plug-in**
2. **Translucent terrain**
3. **Move-up parcels (to fake elevated surface)**

Ground push plug-in



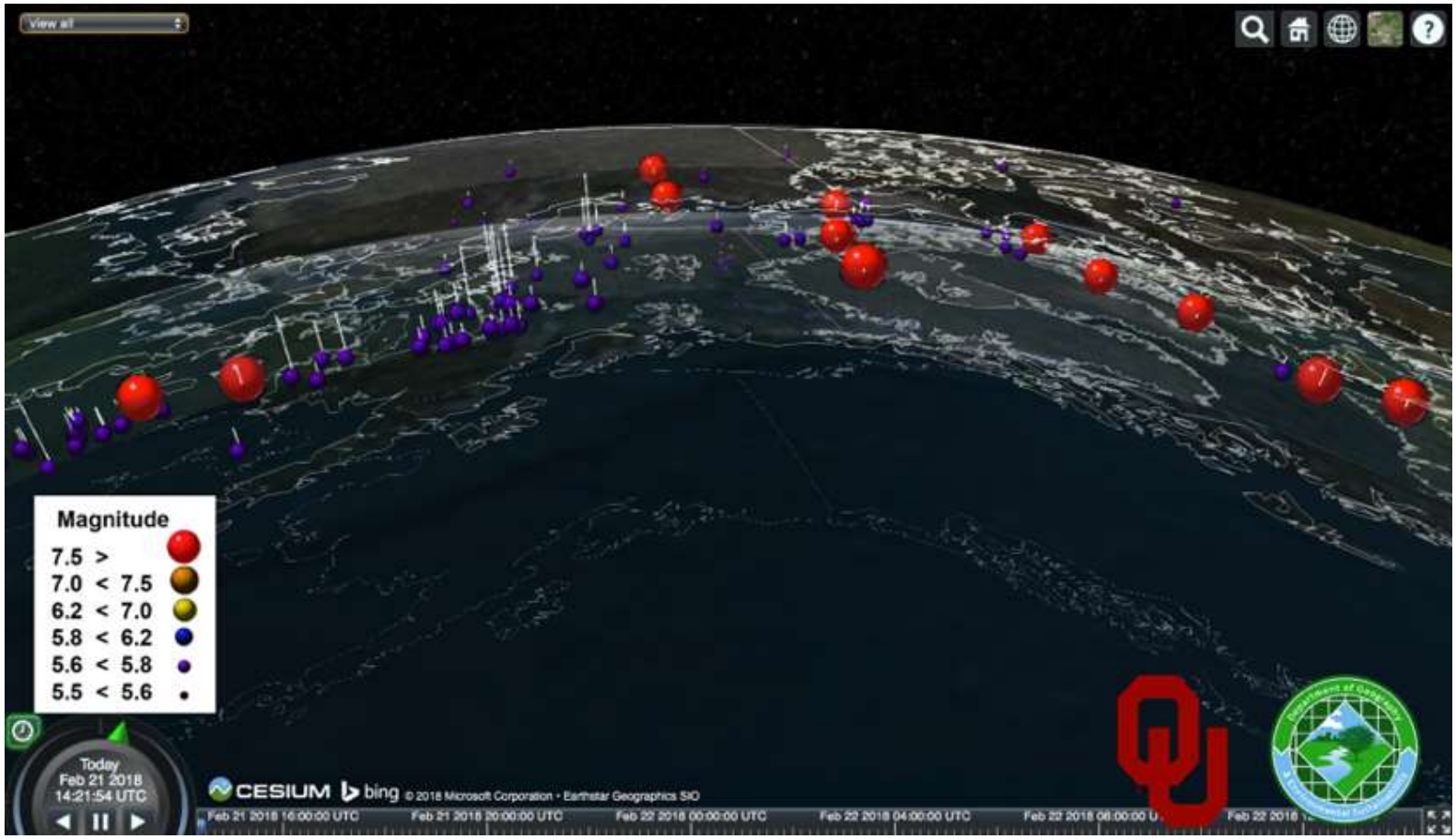
- Locally push down the surface
- shape of rectangle cannot be changed after initializing
- zooming too close makes the rectangle disappear.

Translucent terrain



- surface does not need to be moved/modified
- camera cannot go under the surface
  → the objects can only be seen 'from above'

Move-up parcels (to fake elevated surface)



- camera can go under the fake surface
- parcels need to be shifted up of a certain amount

However: visualization is confusing

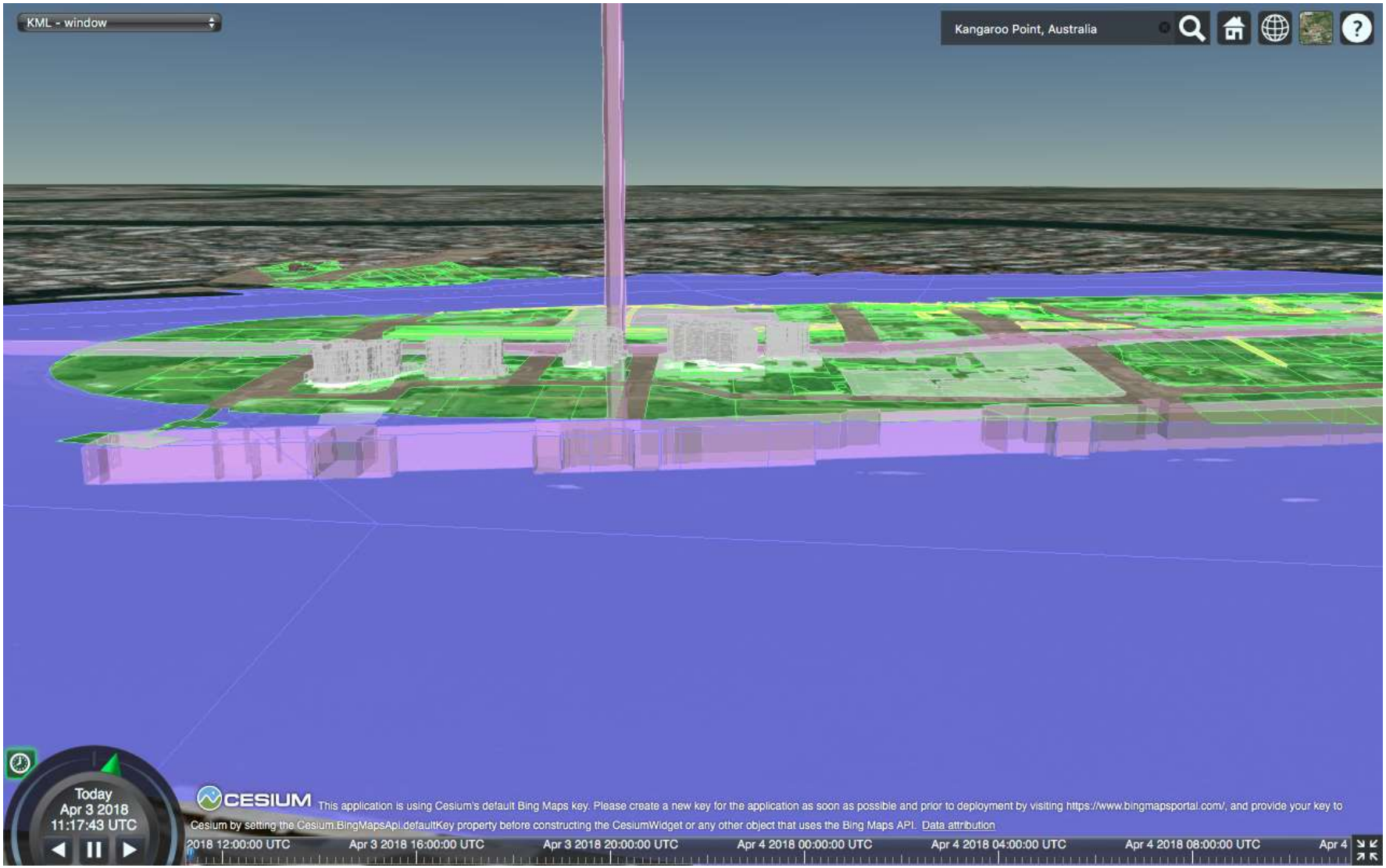
Improvement: Slide-up parcels

Definition of a new requirement for 3D cadastre visualization: the **interactive elevation tool → slide-up (vertical)**

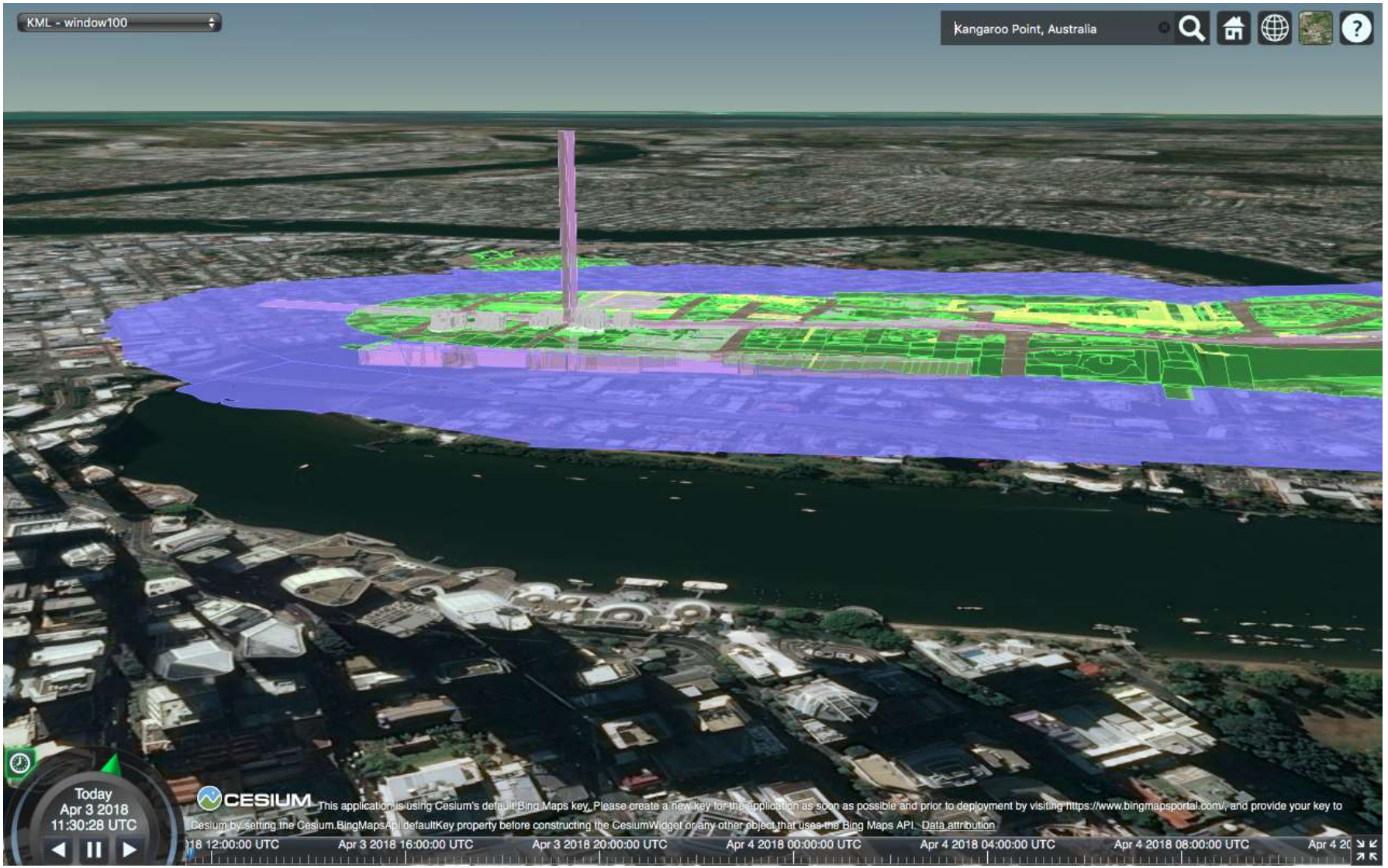
Possibility to move the 3D objects up or down of a user defined amount to be able to visualize in detail the underground parcel that are hidden by the earth surface

The user can navigate to the exact location and have a reality-like visualization. If needed, transformations can be applied to better visualize the hidden parcels

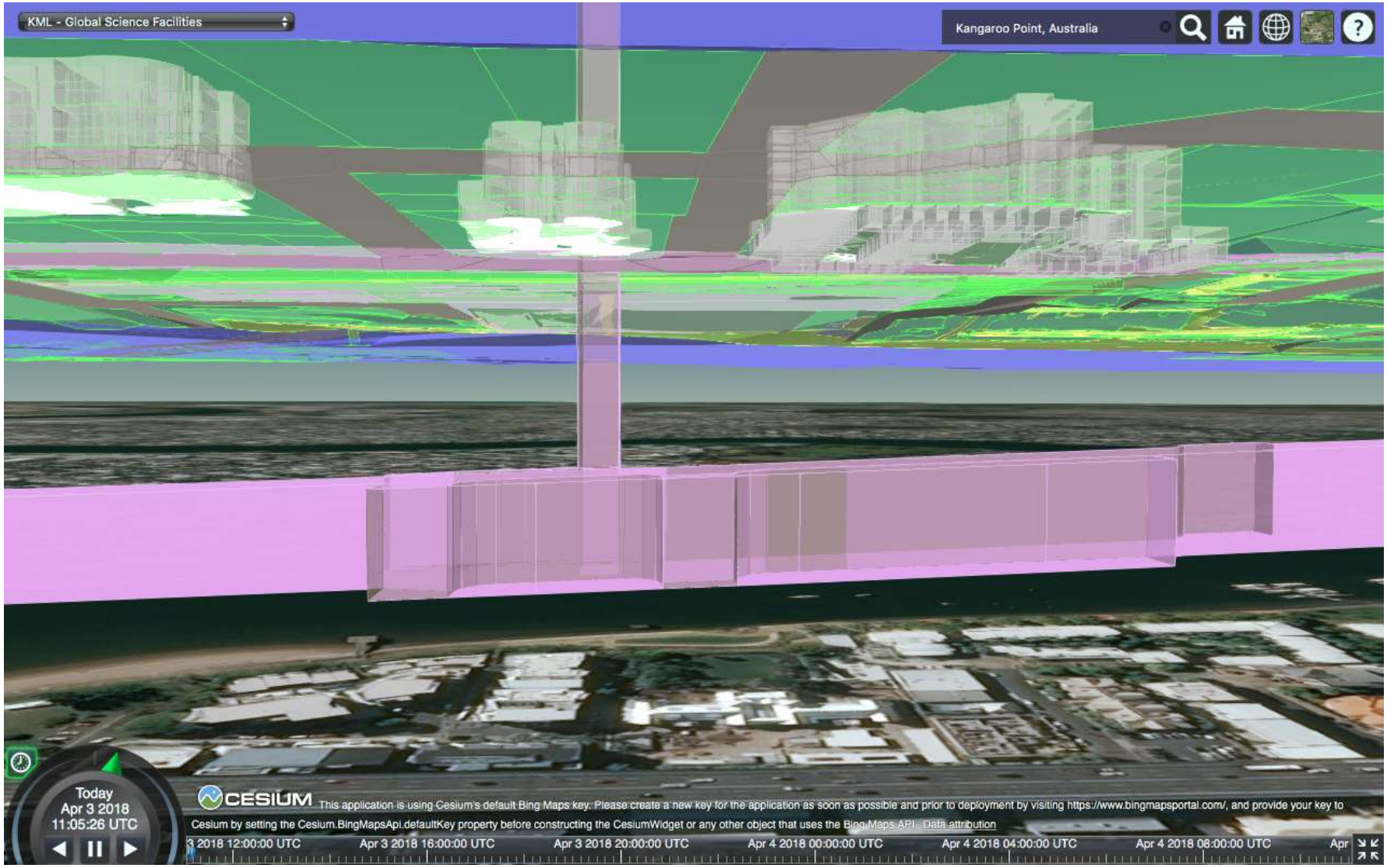
Visualization of the parcels in Cesium



Visualization with interactive elevation



View below surface



Dynamic elevation tool

Current implementation:
A drop down menu which allows the user to choose between different elevations: 50m, 100m, 150m and 200m.

Future implementation:
A slider which allows the user to set a value in the range of the slider and adjust the height accordingly
→ similar to vertical parcel slider (floor in building Russia)

Feature picking (highlight)

- User can click on feature and get information about it (currently feature does not get highlighted)
- KML support in Cesium JS is quite limited (JSON, glTF better?)

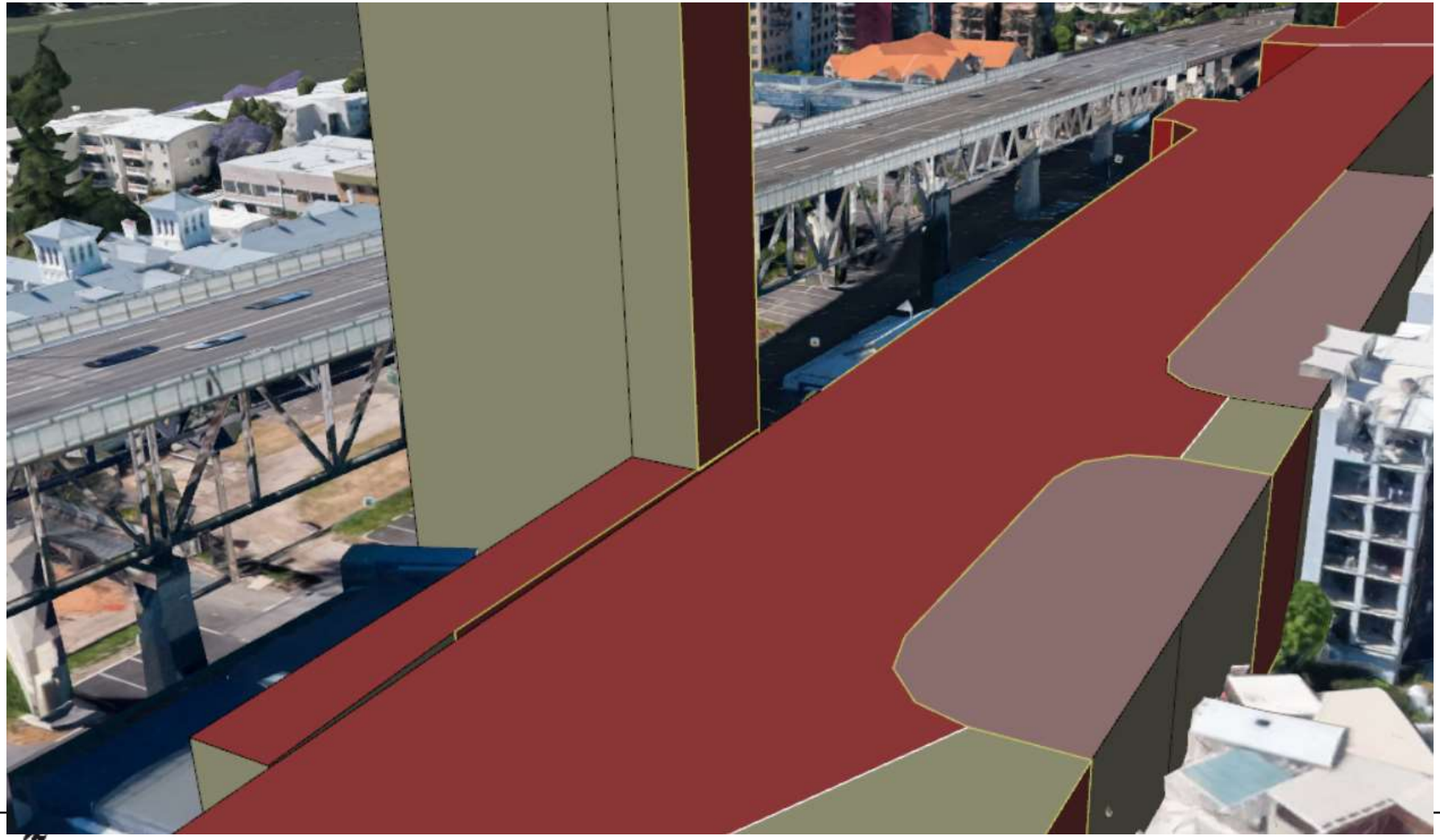
Tooltip functionality

The tooltip functionality is connected to feature picking

Unit/lot/plan numbers are shown

Alternative: Google Earth, parcels moved up (note below surface tunnel → red)

Details of the tunnel parts



Building format 3D parcel data

Presentation outline

1. Introduction
2. Requirements overview
3. Webviewer options
4. Data preparation
5. Initial results
6. Future work

Future work

- Towards **XML structured survey plans** (instead of pdf)
- Refine the **interactive elevation functionality** in order to solve the underground parcels visualization
- Have **highlight functionality** so that it is possible to visualize the whole parcel and not just a face
- Add **administrative** data (RRR, Parties)
- Implement the **object selection tool**
- Implement true **server-client communication**
- **Usability test** (formal or informal)

Smart Surveyors for Land and Water Management

kadaster

GIN Geo-Informatie Nederland

ITC UNIVERSITY OF TWENTE.