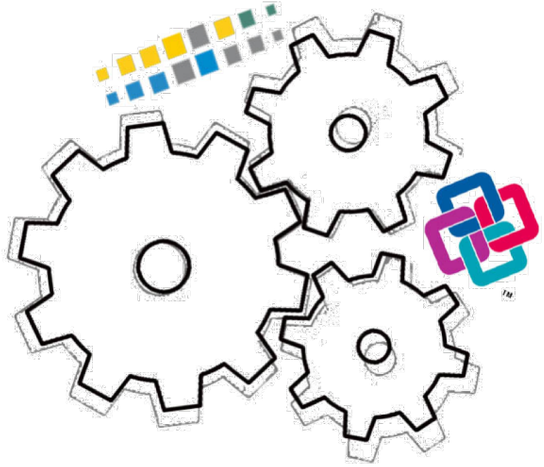
GeoBIM benchmark results: The support for CityGML within GIS (and other) tools

Stelios Vitalis, Francesca Noardo, Ken Arroyo Ohori

Amsterdam
2019-12-02



isprs
information from imagery

EuroSDR

NUS National University of Singapore

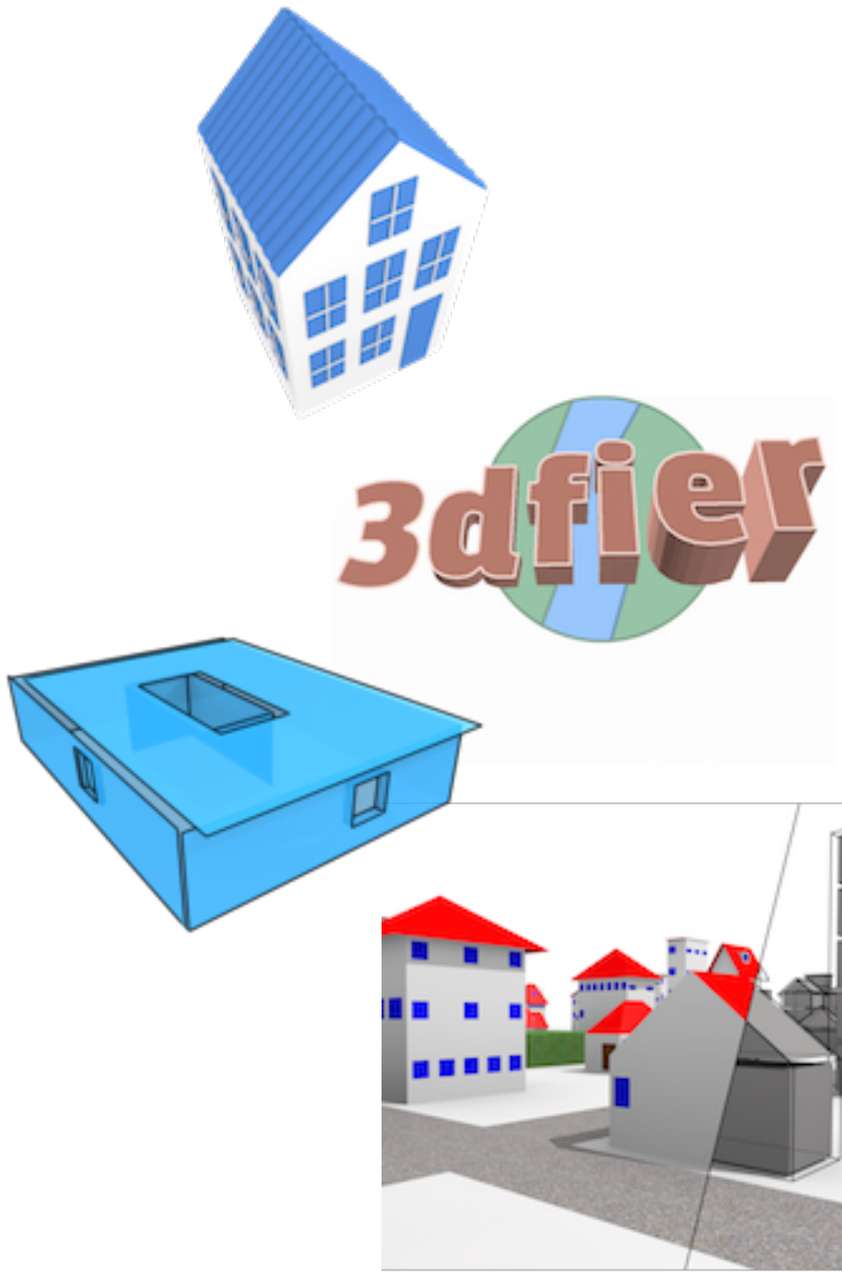
Lund University

TU Delft 3Dgeoinfo

UCL

National Technical University of Athens

We work a lot with CityGML



Journal articles

2019

Harmonising the OGC Standards for the Built Environment: A CityGML Extension for LandInfra. Kavisha Kumar, Anna Labetski, Ken Arroyo Ohori, Hugo Ledoux and Jantien Stoter. *ISPRS International Journal of Geo-Information* 8(6), 2019.

CityJSON: a compact and easy-to-use encoding of the CityGML data model. Hugo Ledoux, Ken Arroyo Ohori, Kavisha Kumar, Balázs Dukai, Anna Labetski and Stelios Vitalis. *Open Geospatial Data, Software and Standards* 4(4), 2019.

2018

CityGML Application Domain Extension (ADE): overview of developments. Filip Biljecki, Kavisha Kumar and Claus Nagel. *Open Geospatial Data, Software and Standards* 3(13), August 2018.

Compactly representing massive terrain models as TINs in CityGML. Kavisha Kumar, Hugo Ledoux and Jantien Stoter. *Transaction in GIS* 22(5), Sep 2018, pp. 1152–1178.

A metadata ADE for CityGML. Anna Labetski, Kavisha Kumar, Hugo Ledoux and Jantien Stoter. *Open Geospatial Data, Software and Standards* 3(16), 2018.

HSW: Heuristic shrink-wrapping for automatically repairing solid-based CityGML LOD2 building models. Junqiao Zhao, Hugo Ledoux, Jantien Stoter and T. Feng. *ISPRS Journal of Photogrammetry and Remote Sensing* 146(289–304), 2018.

2016

Automatic conversion of IFC datasets to geometrically and semantically correct CityGML LOD3 buildings. Sjors Donkers, Hugo Ledoux, Junqiao Zhao and Jantien Stoter. *Transactions in GIS* 20(4), 2016, pp. 547–569.

2015

Automatically enhancing CityGML LOD2 models with a corresponding indoor geometry. Roeland Boeters, Ken Arroyo Ohori, Filip Biljecki and Sisi Zlatanova. *International Journal of Geographical Information Science* 29(12), December 2015, pp. 2248–2268. ISSN: 1365–8816 (Print), 1362–3087 (Online).

But there is something we noticed (as users)…

- Lack of software
- (City)GML files are inconsistent

But there is something we noticed (as devs)…

- (City)GML files are inconsistent (and unpredictable)
- GML is hard to handle with most languages

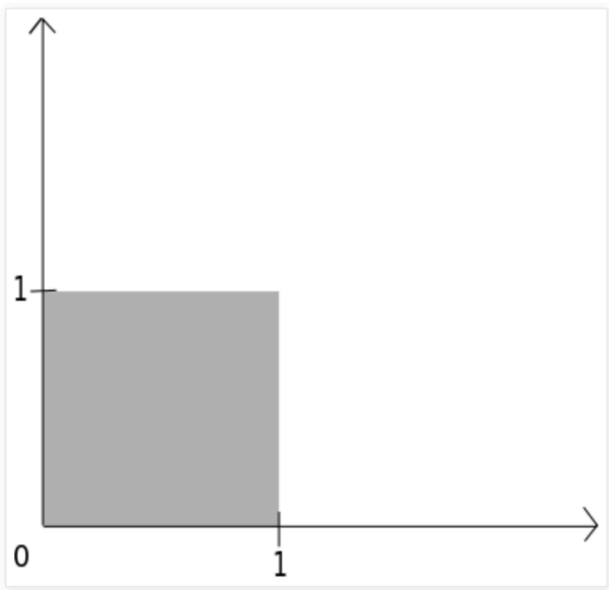
Community agrees

dimanche 6 avril 2014

GML madness

I am convinced that most people wonder "how many ways are there to encode a polygon in GML ?" If you have never considered that before, you might be interested in reading the following lines.

To start gently, let us consider the following grey shape :



James Fee
@jamesmfee

Follow

I just got sent a CityGML file.

11:41 PM - Jun 29, 2016

1 2 8

Let’s measure that!

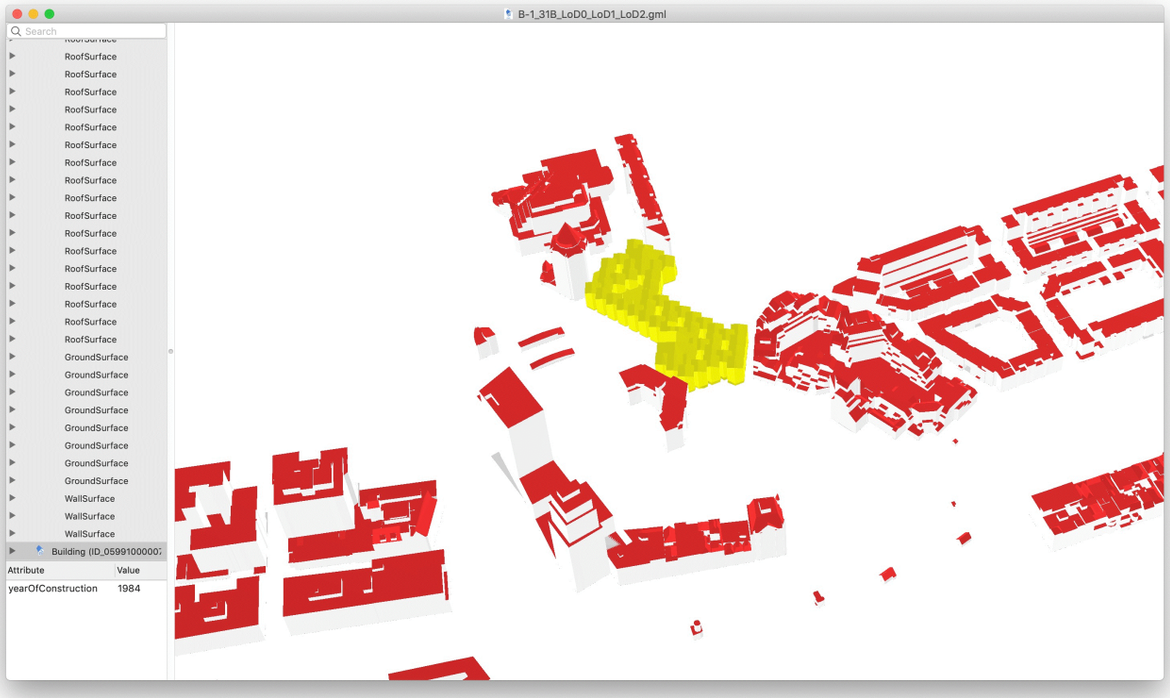
GeoBIM benchmark
Task 3

Measuring the support for CityGML in software

Datasets

- Rotterdam (LoD1 and LoD2)
- Buildings (LoD3)
- Amsterdam (LoD1)

Rotterdam (LoD1 and LoD2)

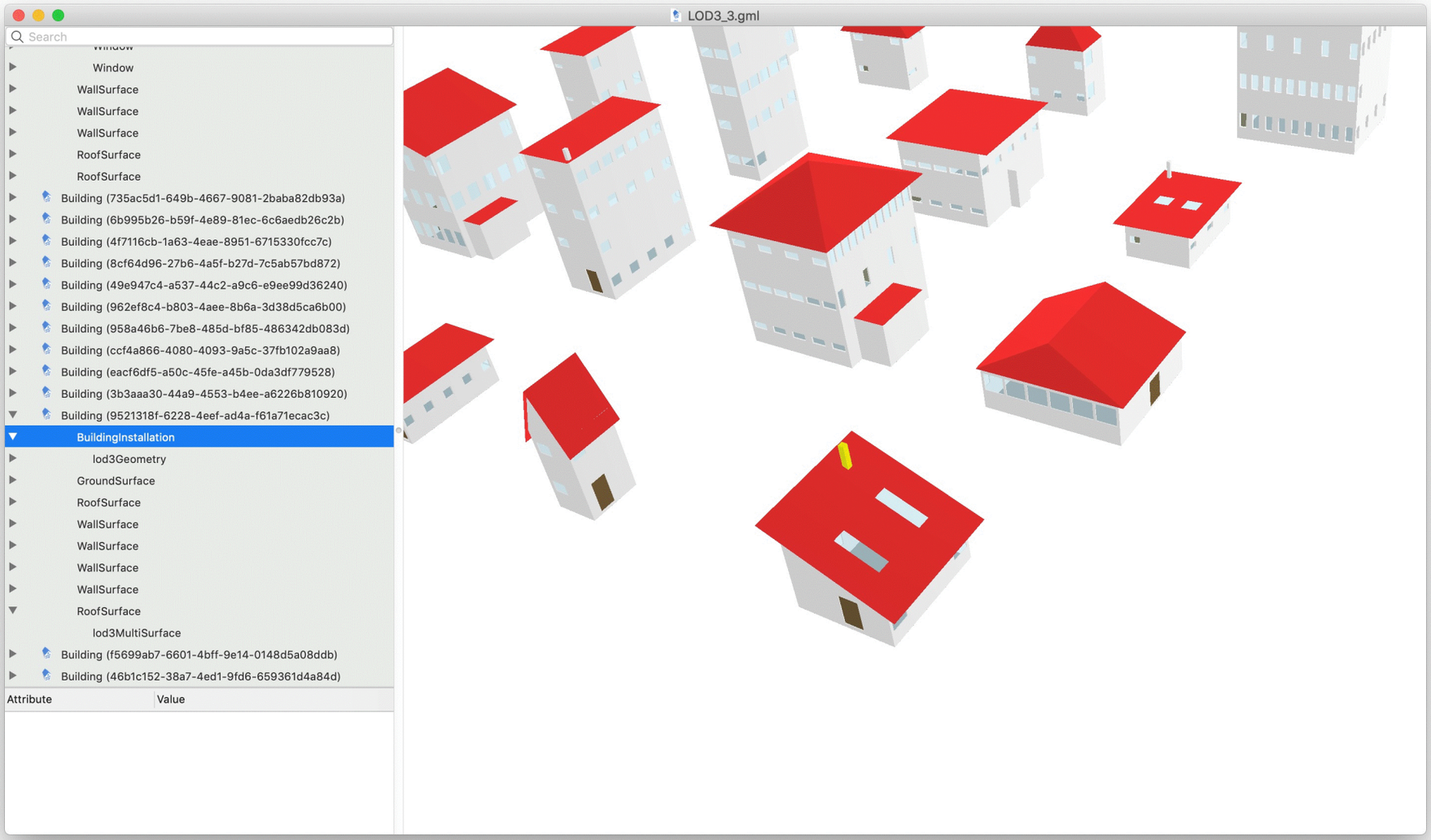


~ 35 MB size

Interesting points:

- Multi-LoD
- Grouping relations

Buildings (LoD3)



1,4 MB

Interesting points:

- High details
- Extensive semantic surfaces

Amsterdam (LoD1)

~5GB size

Interesting points:

- Many object classes (roads, water, vegetation, etc.)
- Challenging size

What we wanted to test

- Import functionality
- Georeferencing information
- Semantics
- Geometry
- Model management
- Export functionality

Submissions

Submissions

- 23 answers received
- 11 different applications tested (14 variations overall)

Commonalities about software

- No software was able to reproduce the same city model.
- Time performance seems insignificant.
- Different datasets change software’s behaviour

Commonalities about output

- Schema is usually valid, besides occasional losses
- Geometries always have issues

Remarks (Best and worse)

- FME and 3DCityDB show best support
- GIS software is lacking

Remarks (Import)

- GIS have trouble handling semantics (at best parent-child relationships through IDs)
- Low support for Multi-LoD

Remarks (Import -> Amsterdam)

- Few submissions
- ArcGIS, ArcGIS Pro, tridicon, eveBIM and 1Spatial Elyx crashed

Remarks (Analysis)

- Only FME offers wide range of 3D analysis
- 1Spatial Elyx 3D: visibility analysis and buffers
- novaFACTORY: visibility analysis, shadows analysis, sun analysis and extract height profiles.

Remarks (Editing)

- Only QGIS and ArcGIS offer extensive editing (semantics and geometry)
- Some 3D viewers (e.g. eveBIM, novaFACTORY) support semantics editing
- FME supports geometry editing, but in batch form (through transformations)

Remarks (3DCityDB)

- Alters the input the least
- Only one file was broken (bad xlink)
- Slight reduction in objects in Amsterdam dataset

Remarks (FME)

- FME makes more changes but results are generally okay
- Some methodologies can result in splitting of objects using the same IDs or in the loss of some object classes (i.e. bridges)

Remarks (QGIS)

- Only one file submitted with QGIS
- Total loss of data
- Output doesn’t even have valid schema

Remarks (ArcGIS)

- Surfaces are converted into independent CityObjects
- Big increase in objects and the loss of some semantics.
- Minor schema issues (e.g. empty dates).

Remarks (NovaFactory)

- novaFACTORY is hit or miss
- Perfect output with Rotterdam dataset (100% valid)
- Amsterdam can’t even be read

Discussion

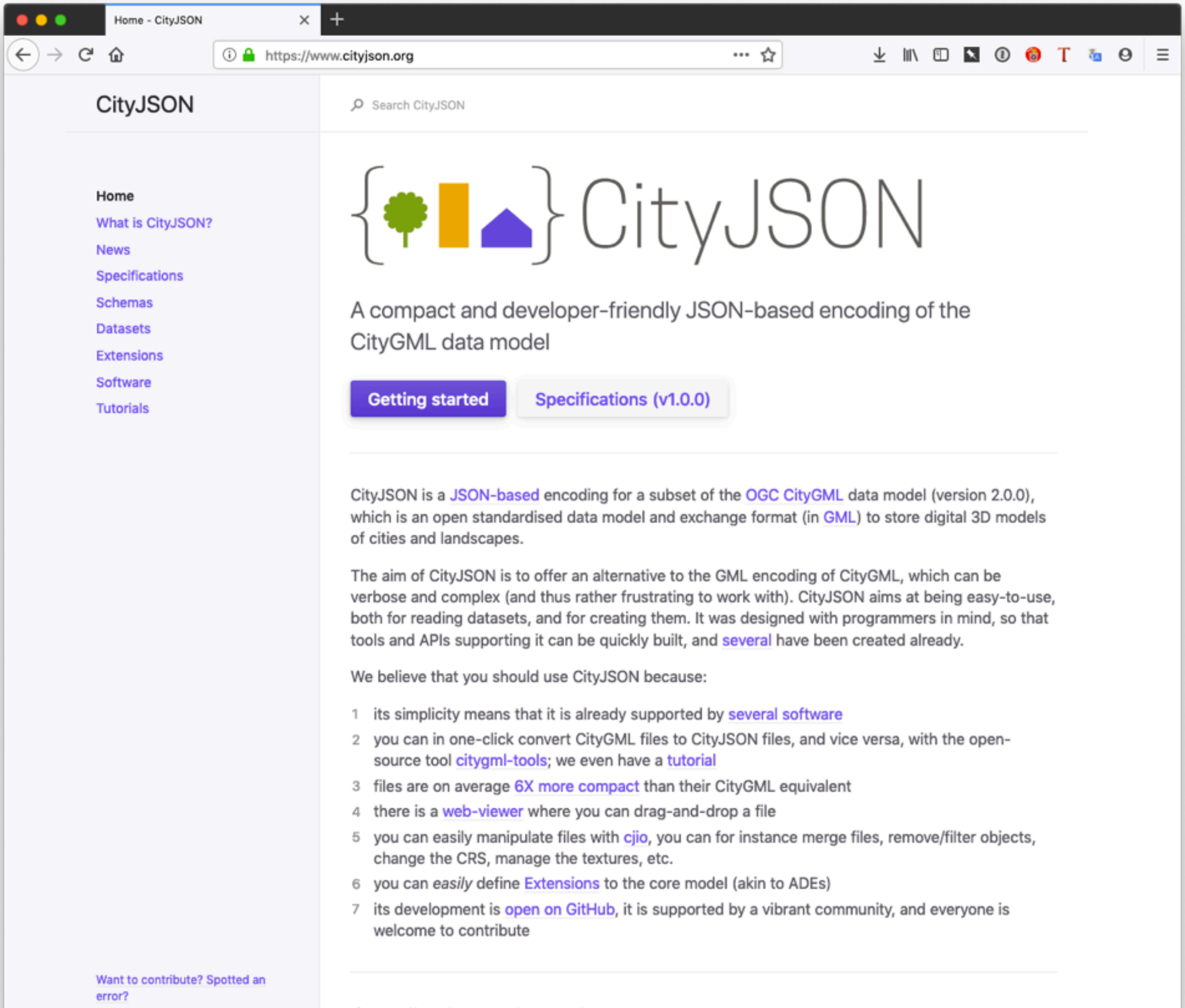
Conclusions

- GIS software does not fit with CityGML
- ETL and specially tailored software is required when working with CityGML
- Schema not as much as a problem as Geometry

What can we learn?

- A standard can be a ”standard” without really standardising much.
- Stay away from XML/GML.
- Complicated hierarchies come with a price.
- Favour constraints over ambiguity.

Enters…

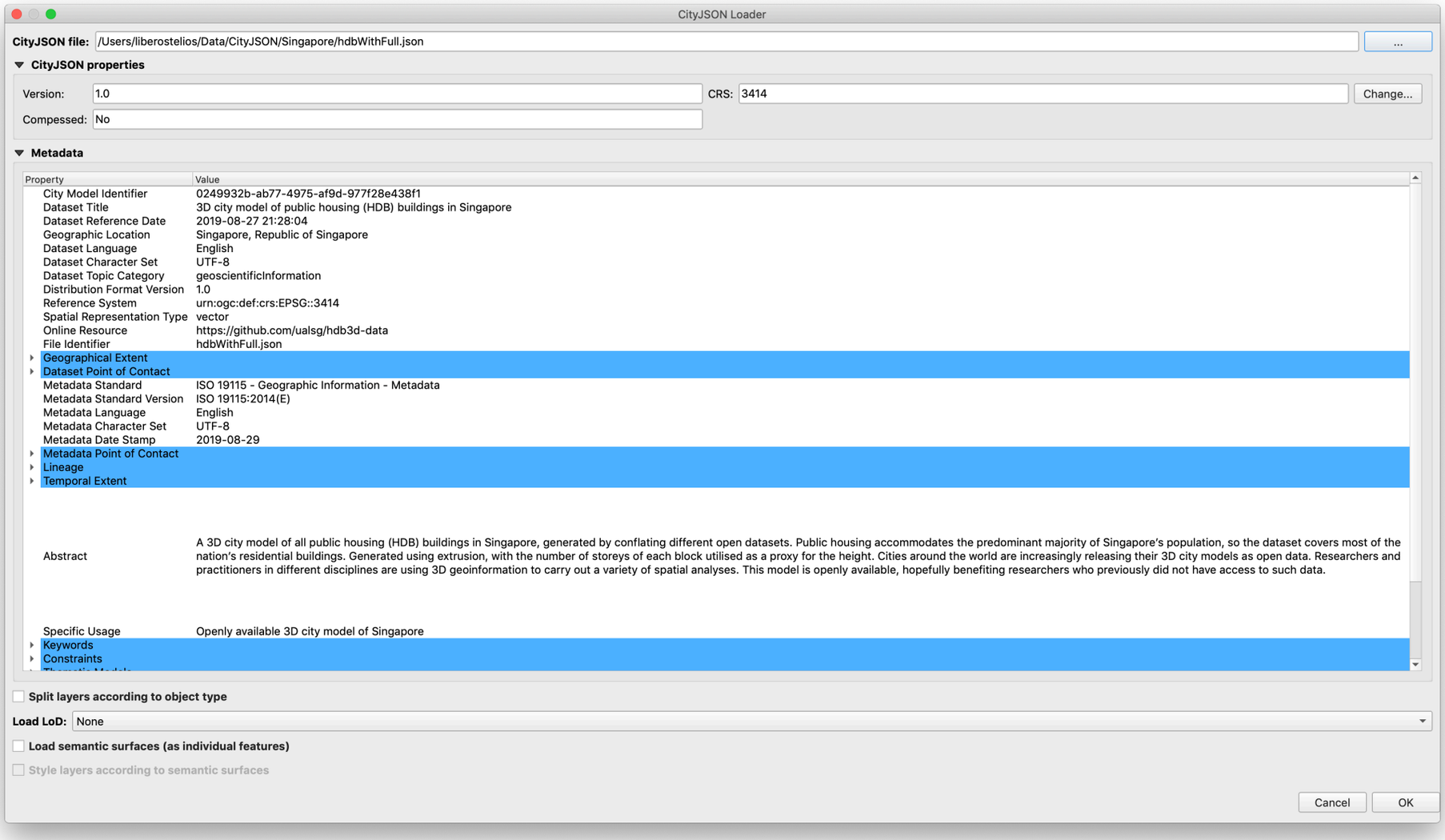

Home - CityJSON
https://www.cityjson.org
CityJSON
Search CityJSON
Home
What is CityJSON?
News
Specifications
Schemas
Datasets
Extensions
Software
Tutorials
Want to contribute? Spotted an error?
CityJSON
A compact and developer-friendly JSON-based encoding of the CityGML data model
Getting started
Specifications (v1.0.0)
CityJSON is a JSON-based encoding for a subset of the OGC CityGML data model (version 2.0.0), which is an open standardised data model and exchange format (in GML) to store digital 3D models of cities and landscapes.
The aim of CityJSON is to offer an alternative to the GML encoding of CityGML, which can be verbose and complex (and thus rather frustrating to work with). CityJSON aims at being easy-to-use, both for reading datasets, and for creating them. It was designed with programmers in mind, so that tools and APIs supporting it can be quickly built, and several have been created already.
We believe that you should use CityJSON because:
1 its simplicity means that it is already supported by several software
2 you can in one-click convert CityGML files to CityJSON files, and vice versa, with the open-source tool citygml-tools; we even have a tutorial
3 files are on average 6X more compact than their CityGML equivalent
4 there is a web-viewer where you can drag-and-drop a file
5 you can easily manipulate files with cjio, you can for instance merge files, remove/filter objects, change the CRS, manage the textures, etc.
6 you can easily define Extensions to the core model (akin to ADEs)
7 its development is open on GitHub, it is supported by a vibrant community, and everyone is welcome to contribute

Two steps to solve your problems

1. Convert to CityJSON
2. Enjoy!

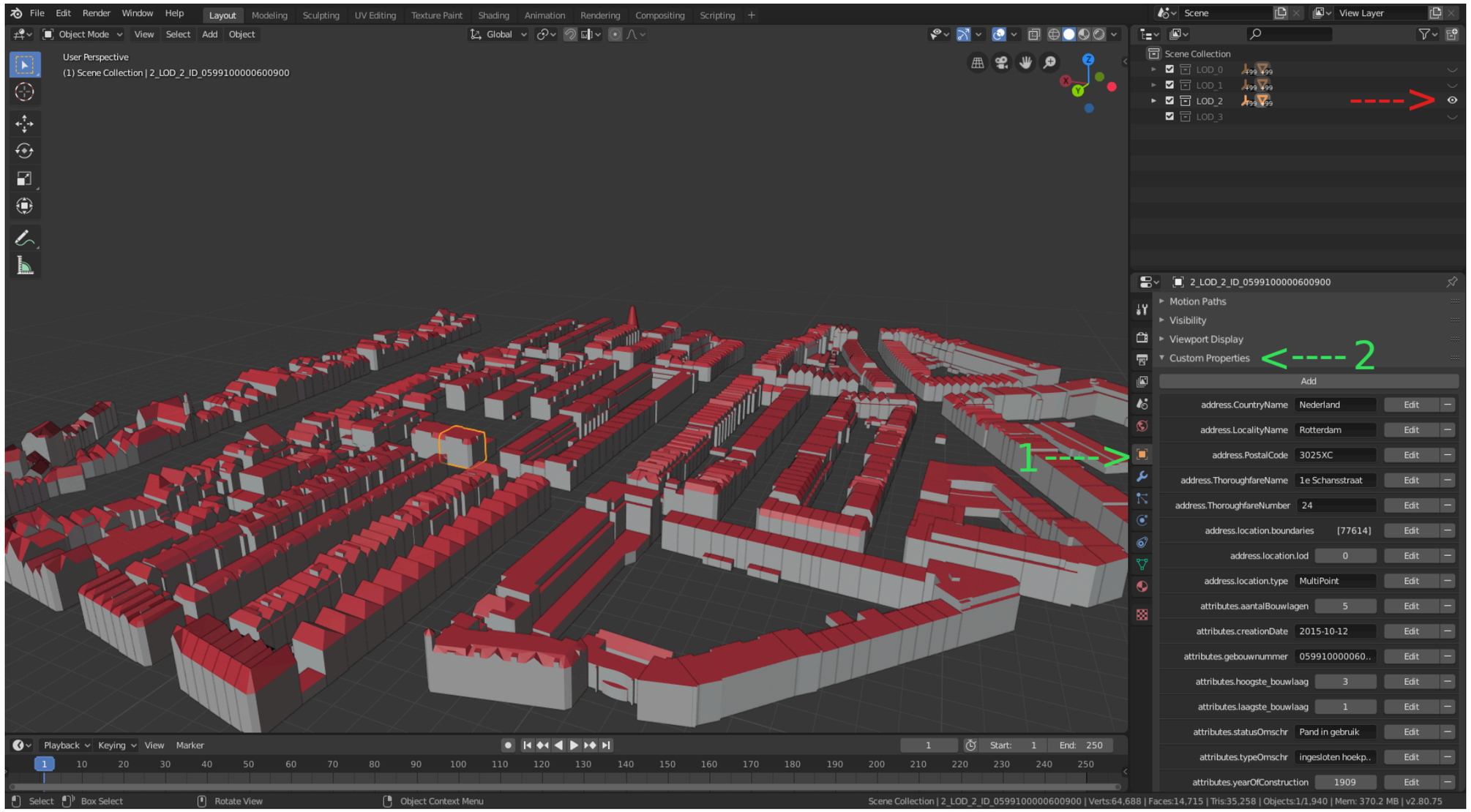
One last thing

Building software for GeoBIM users



- Robust import support for 3D city models
- Will incorporate 3D city models' toolbox (WIP)
- Install from within QGIS

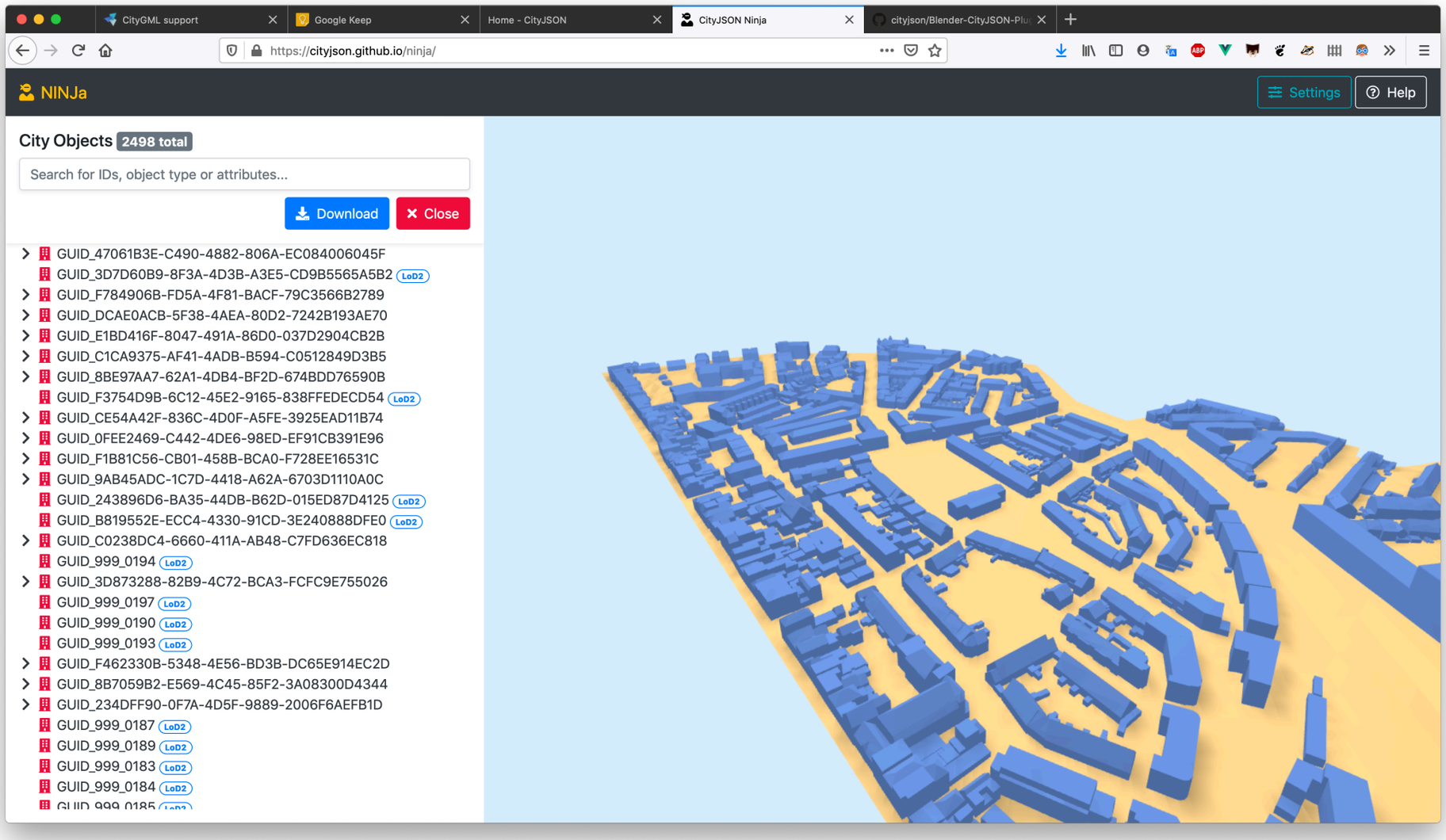
Building software for GeoBIM users



- First true 3D modelling software to support 3D city models
- Can act as ETL for BIM users (IFC import/export)

https://github.com/cityjson/Blender-CityJSON-Plugin

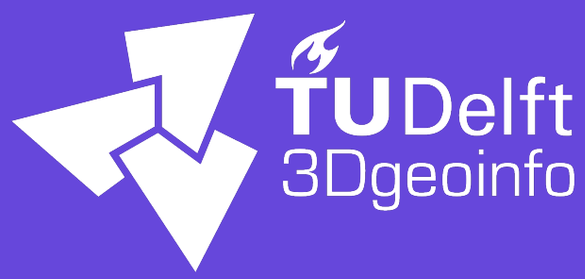
Building software for GeoBIM users



- A software to better explore the structure of CityGML data
- Basic editing capabilities

https://cityjson.github.io/ninja/

Thank you!



https://3d.bk.tudelft.nl
@tudelft3d
tudelft3d

https://3d.bk.tudelft.nl/svitalis
@liberostelios
liberostelios