

Lesson 3.1

Three-dimensional geometries in geoinformation

GE01004:
3D modelling of the built environment

<https://3d.bk.tudelft.nl/courses/geo1004>



3D geoinformation

Department of Urbanism
Faculty of Architecture and the Built Environment
Delft University of Technology

About GIS standards

Standardisation organisations

- **ISO:** International Standardisation Organisation
- **OGC:** Open Geospatial Consortium
- **INSPIRE:** Infrastructure for Spatial Information in Europe
- **NEN:** *normalisatie en normen* (“Dutch ISO”)

ISO 191xx series of geographic information standards


- A main goal of the ISO 191xx series is to facilitate the interoperability of geographic information systems, i.e. ability to discover, access, understand and use the information and tools, independently from the platform supporting them
- Structured set of standards specifying methods, tools and services for the management of geographic information, including:
 - Definition of data,
 - Access to data
 - Presentation of data
 - Transfer between users
- weirdly, and annoyingly, one has to pay to read these 😞

ISO 191xx series of geographic information standards

6709 - Standard representation of latitude, longitude and altitude for geographic point locations	19122 - Qualifications and Certification of personnel
19101 - Reference model	19123 - Schema for coverage geometry and functions
19101-2 - Reference model - Part 2: Imagery	19124 - Imagery and gridded data components
19103 - Conceptual schema language	19125-1 - Simple feature access - Part 1: Common architecture
19104 - Terminology Introduction	19125-2 - Simple feature access - Part 2: SQL option
19105 - Conformance and testing	19126 - Profile - FACC Data Dictionary
19106 - Profiles	19127 - Geodetic codes and parameters
19107 - Spatial schema	19128 - Web Map server interface
19108 - Temporal schema	19129 - Imagery, gridded and coverage data framework
19109 - Rules for applicaiton schema	19130 - Sensor and data models for imagery and gridded data
19110 - Methodology for feature cataloguing	19131 - Data product specifications
19111 - Spatial referencing by coordinates	19132 - Location based services - Reference model
19112 - Spatial referencing by geographic identifiers	19133 - Location based services - Tracking and navigation
19113 - Quality principles	19134 - Multimodal location based services for routing and navigation
19114 - Quality evaluation procedures	19135 - Procedures for registration of geographical information items
19115 - Metadata	19136 - Geography Markup Language

19115-2 - Metadata - Part 2: Extensions for imagery and gridded data	19137 - Generally used profiles of the spatial schema and of similar important other schemas
19116 - Positioning services	19138 - Data quality measures
19117 - Portrayal	19139 - Metadata - Implementation specification
19118 - Encoding	19140 - Technical amendment to the ISO 191** Geographic information series of standards for harmonization and enhancements
19119 - Services	19141 - Schema for moving features
19120 - Functional standards	
19121 - Imagery and gridded data	

OGC: Open Geospatial Consortium®



The screenshot shows the OGC website homepage. The browser address bar displays "https://www.ogc.org". The navigation menu includes "ABOUT", "MEMBERSHIP", "STANDARDS & RESOURCES", "INNOVATION", and "NEWS & EVENTS". The main banner features the text "122ND OGC MEMBER MEETING Connecting Location Globally" and "28 February - 4 March, 2022 | VIRTUAL". An AWS logo is present with the text "PROUDLY SPONSORED BY: aws". Below the banner, the text reads "The Home of Location Technology Innovation and Collaboration" and "Your Global Resource for Geospatial Information and Standards". A section titled "Upcoming Events" lists several events with dates.

**The Home of Location Technology
Innovation and Collaboration**

Your Global Resource for Geospatial Information and Standards

Welcome to OGC, a worldwide community committed to improving access to geospatial, or location information. We connect people, communities, and technology to solve global challenges and address everyday needs. The organization represents over 500 businesses, government agencies, research organizations, and universities united with a desire to make location information FAIR – Findable,

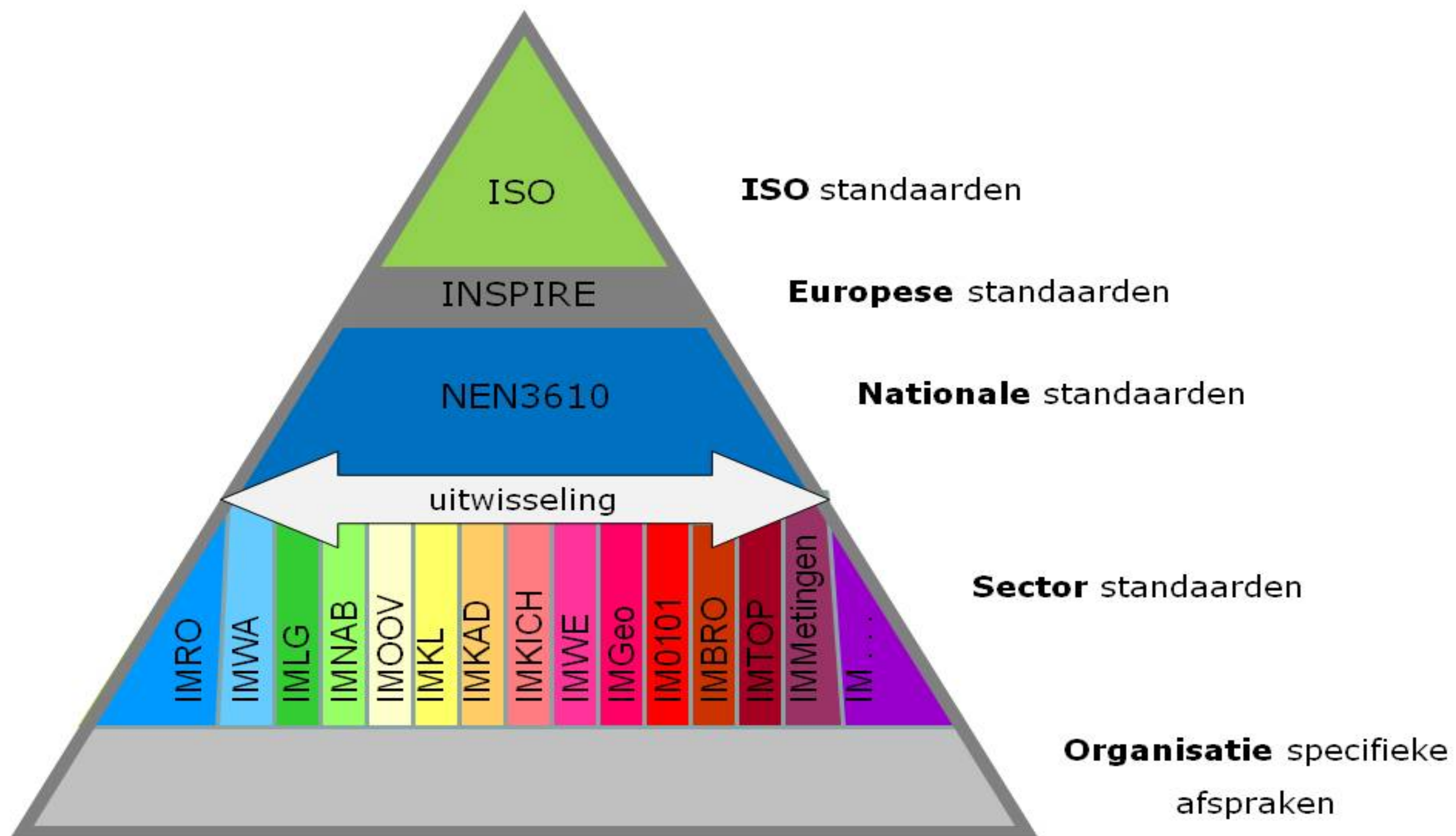
Upcoming Events

- [Pre-Sprint Webinar \[2022 Joint OGC - OSGeo - ASF Code Sprint\]](#)
23 Feb - 23 Feb
- [2nd Geospatial Knowledge Infrastructure Summit – Advancing Industry Adoption](#)
24 Feb - 25 Feb
- [Testbed-18 Bidders Q&A Webinar](#)
24 Feb - 24 Feb
- [AAG 2022 NYC - Annual Meeting](#)
25 Feb - 01 Mar

[View all events...](#)

- many standards are aligned with ISO's (ie, they are exactly the same)
- OGC usually standards closer to implementations than ISO
 - GML
 - KML
 - netCDF
- abstract specifications = ISO
- implementation specification = OGC
- unlike ISO's, OGC's documents are free 😊

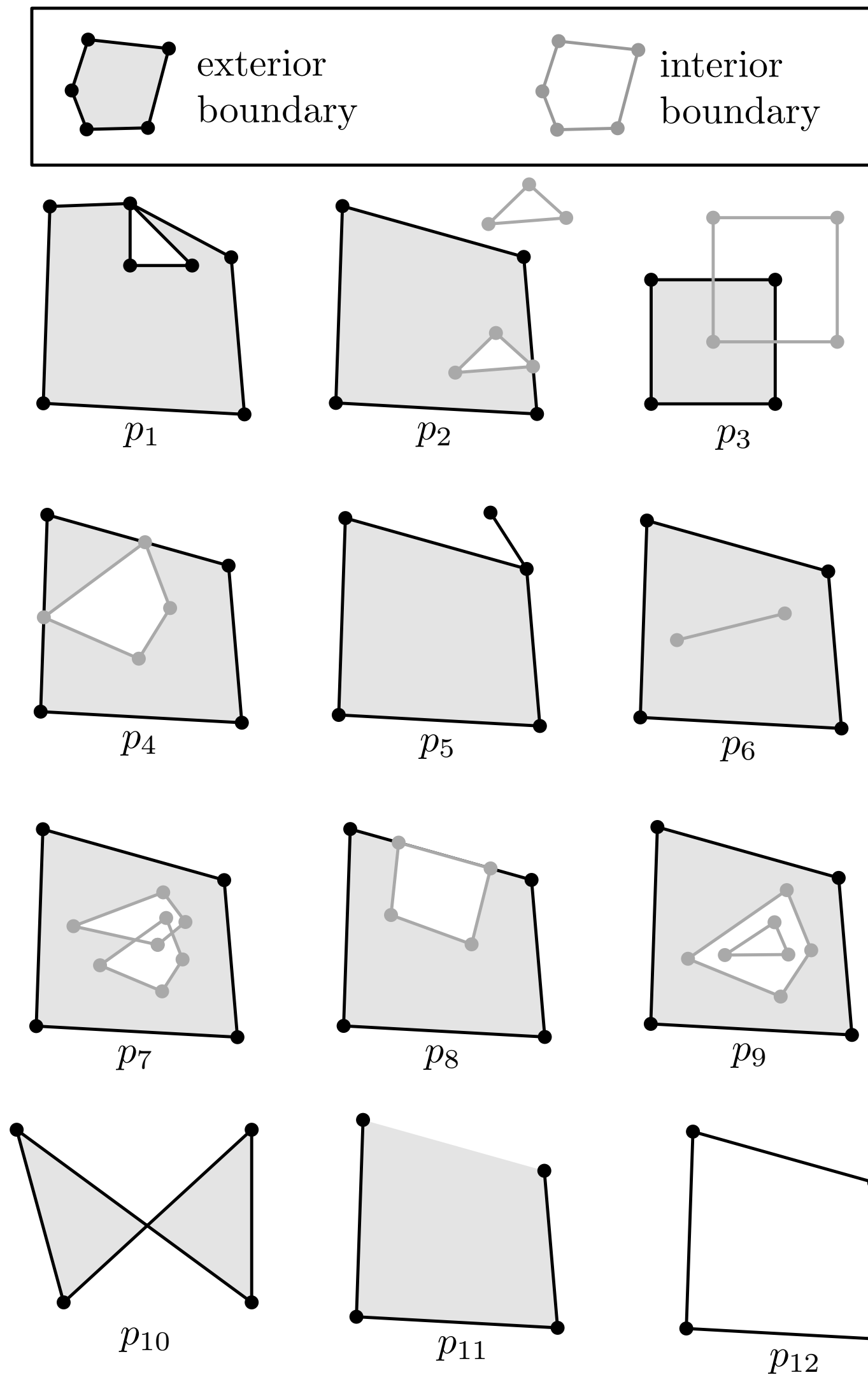
The standards from a Dutch point-of-view



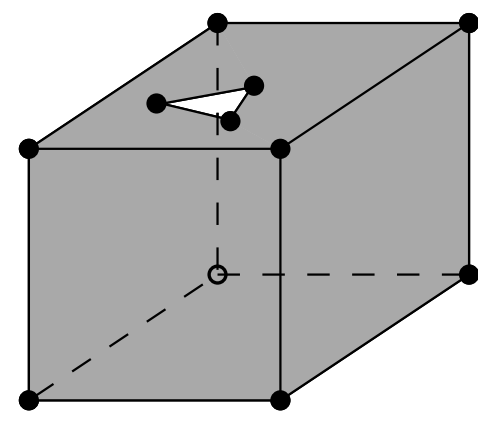
Rules for validation in 2D

OGC Simple Features and ISO19107 rules:

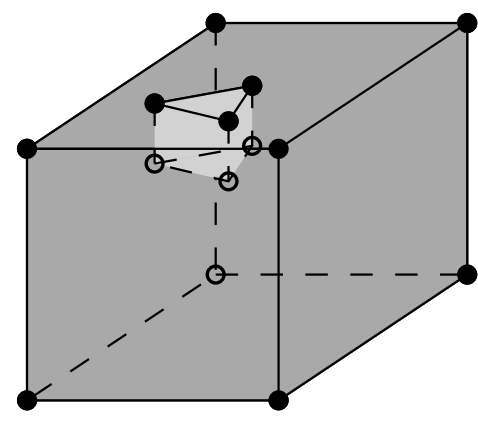
- 1 no self-intersection
- 2 closed boundaries
- 3 rings can touch but not overlap
- 4 no duplicate points
- 5 no dangling edges
- 6 connected interior
- 7 etc



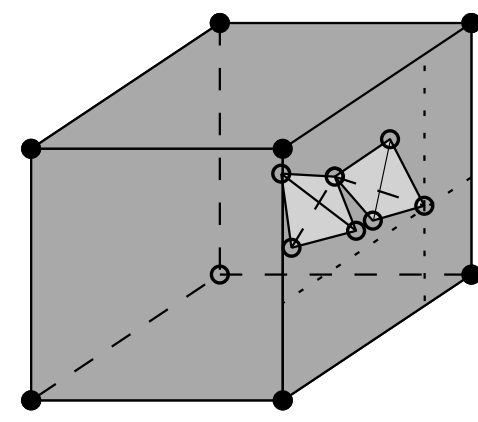
In 3D it's way more complex...



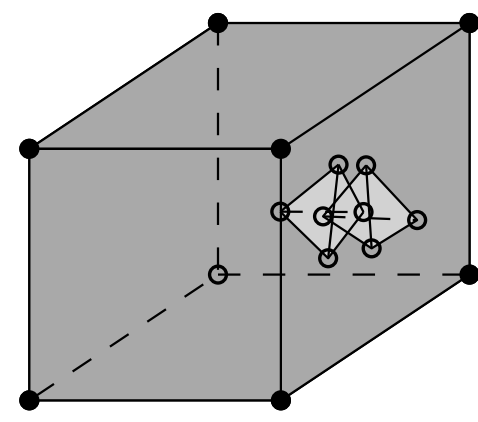
s_1
invalid (1)



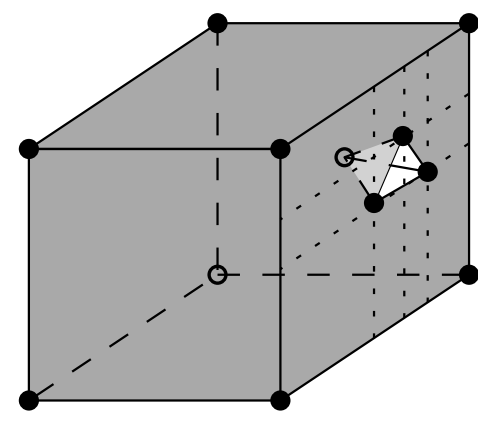
s_2
valid



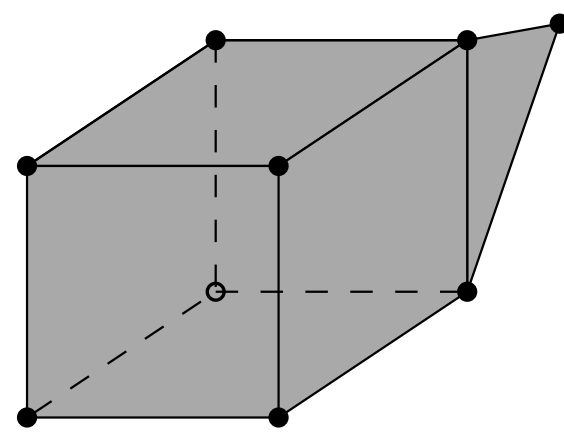
s_3
valid



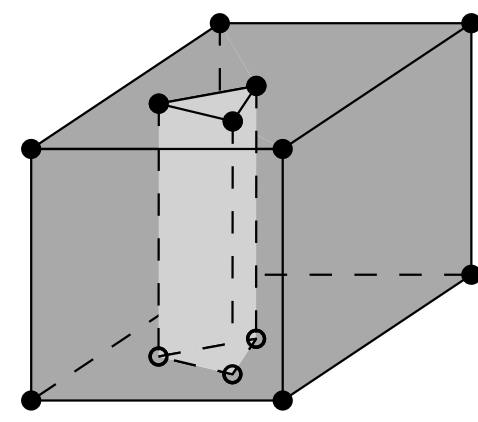
s_4
invalid (3, 6)



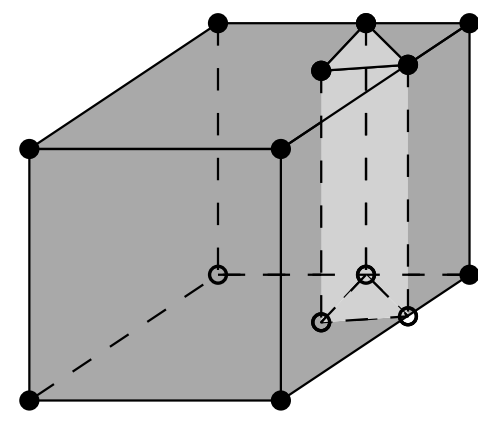
s_5
invalid (6)



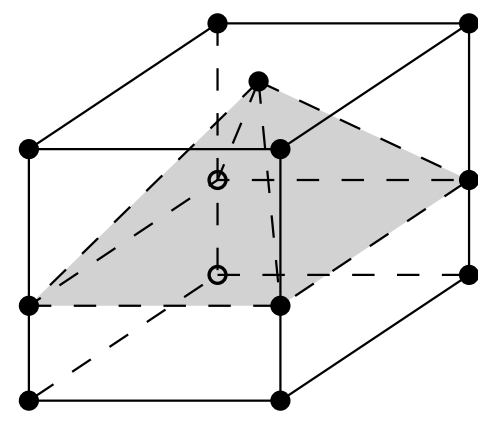
s_6
invalid (4)



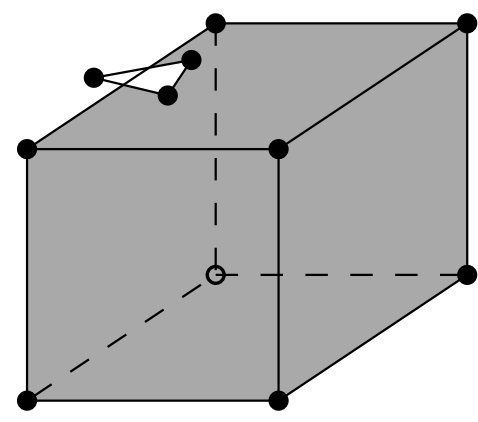
s_7
valid



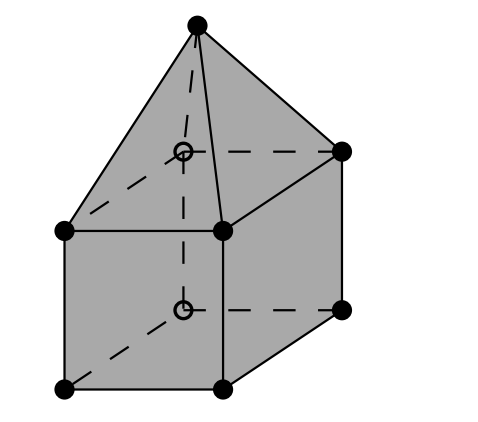
s_8
invalid (2, 5)



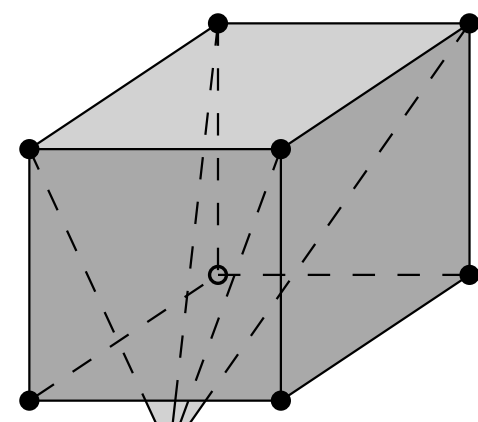
s_9
invalid (5)



s_{10}
invalid (3 in 2D)



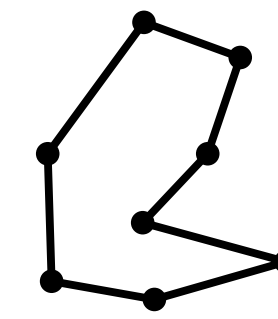
s_{11}
valid



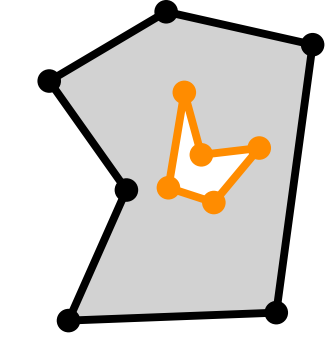
s_{12}
invalid (2)



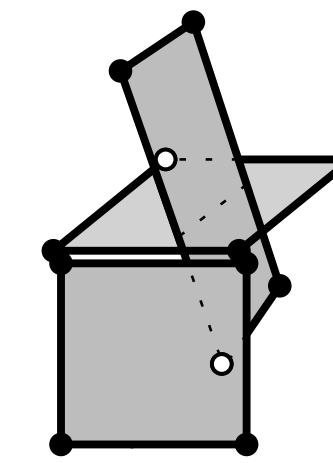
Point



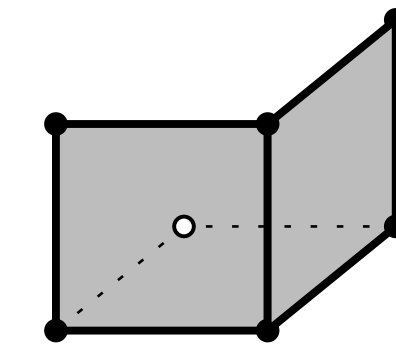
LinearRing



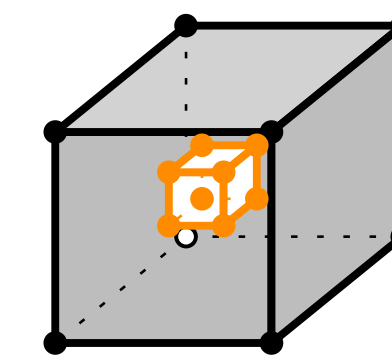
Polygon



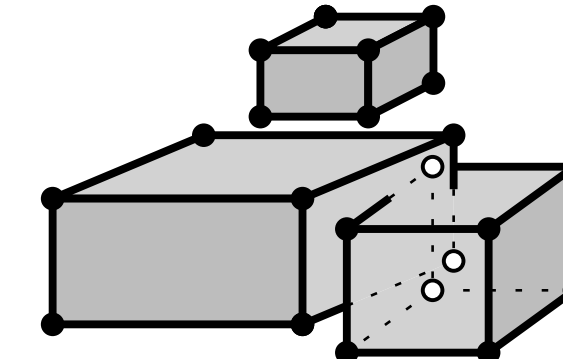
MultiSurface



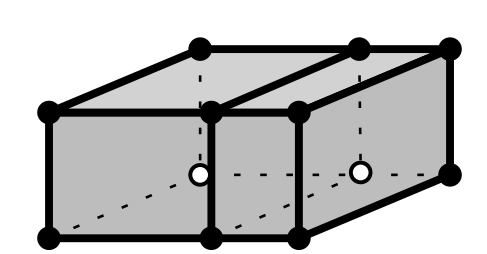
CompositeSurface



Solid



MultiSolid

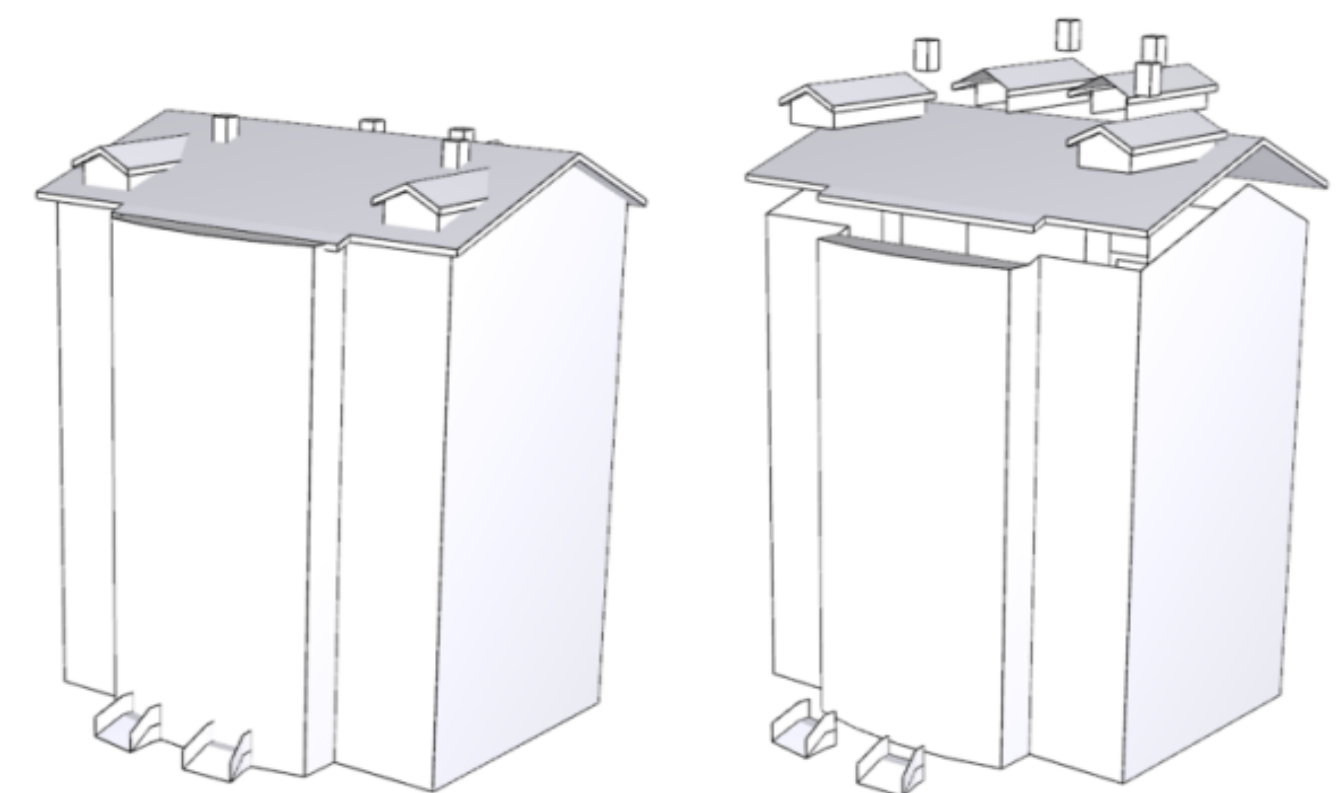
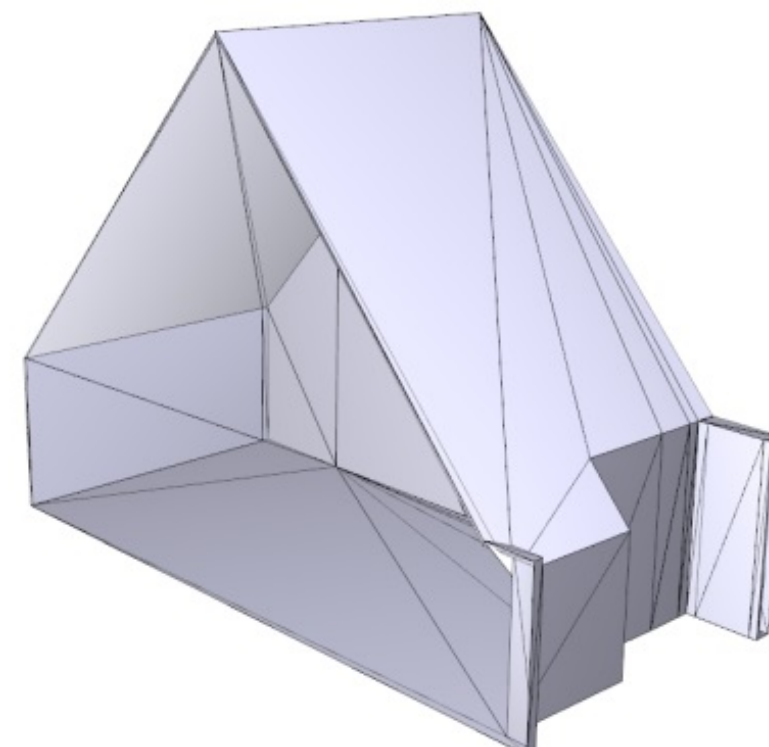
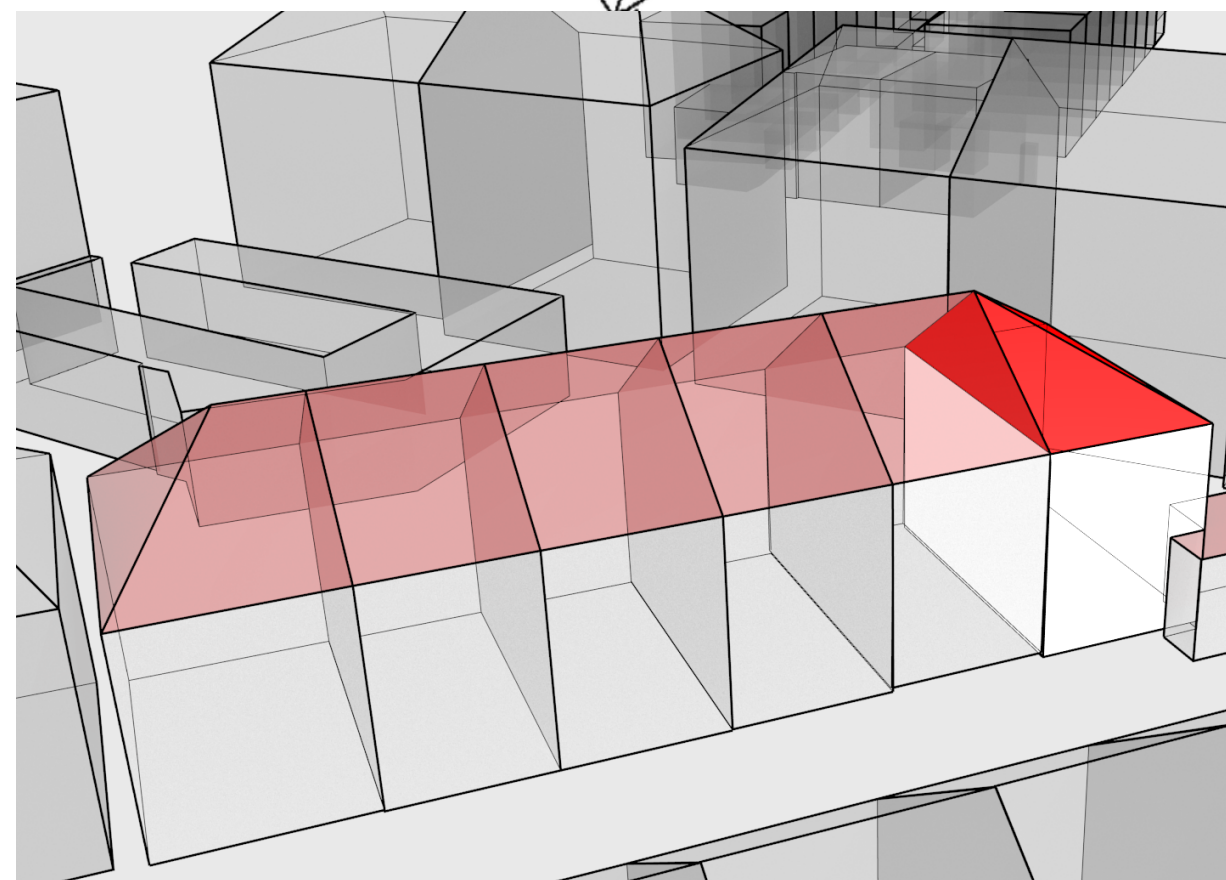
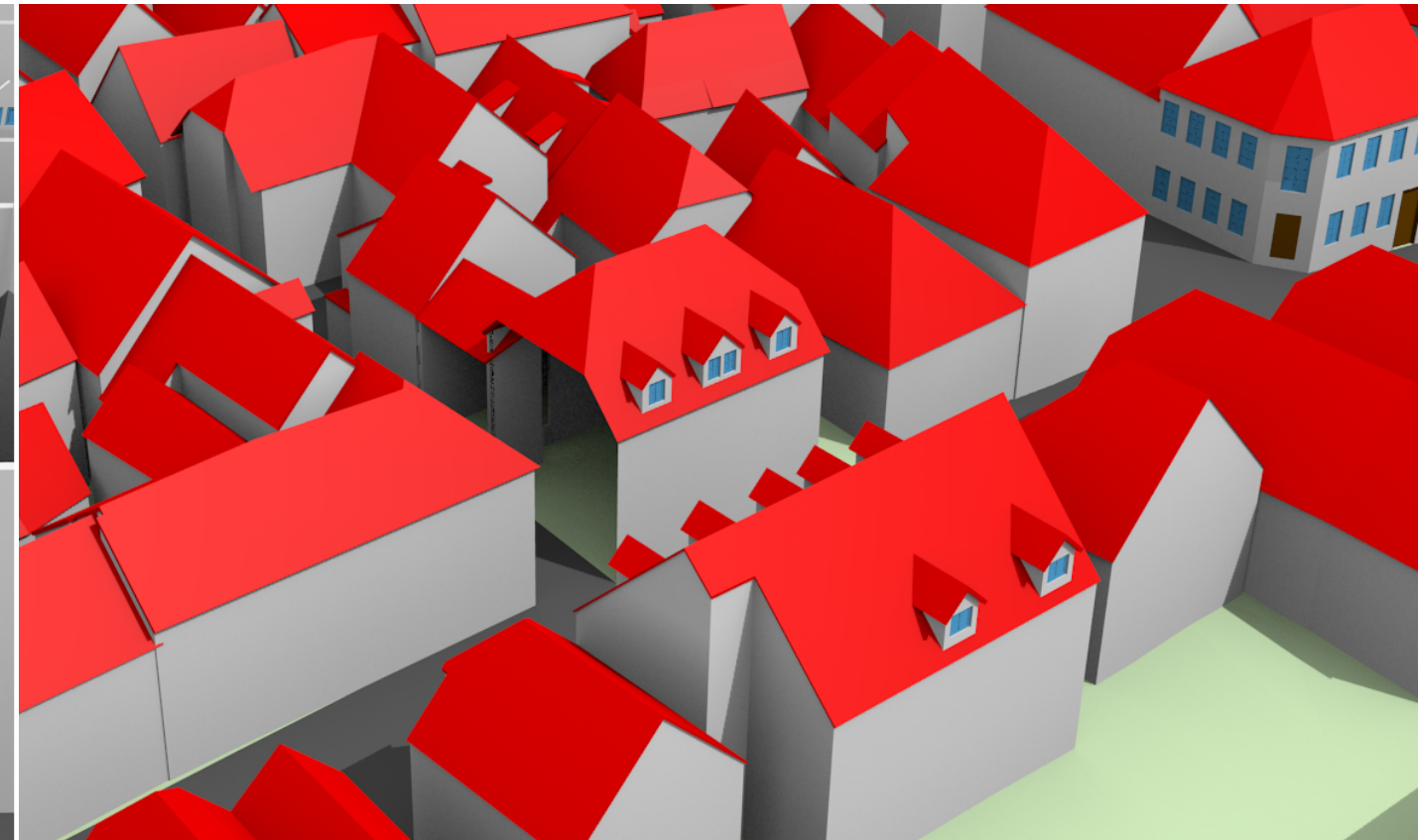
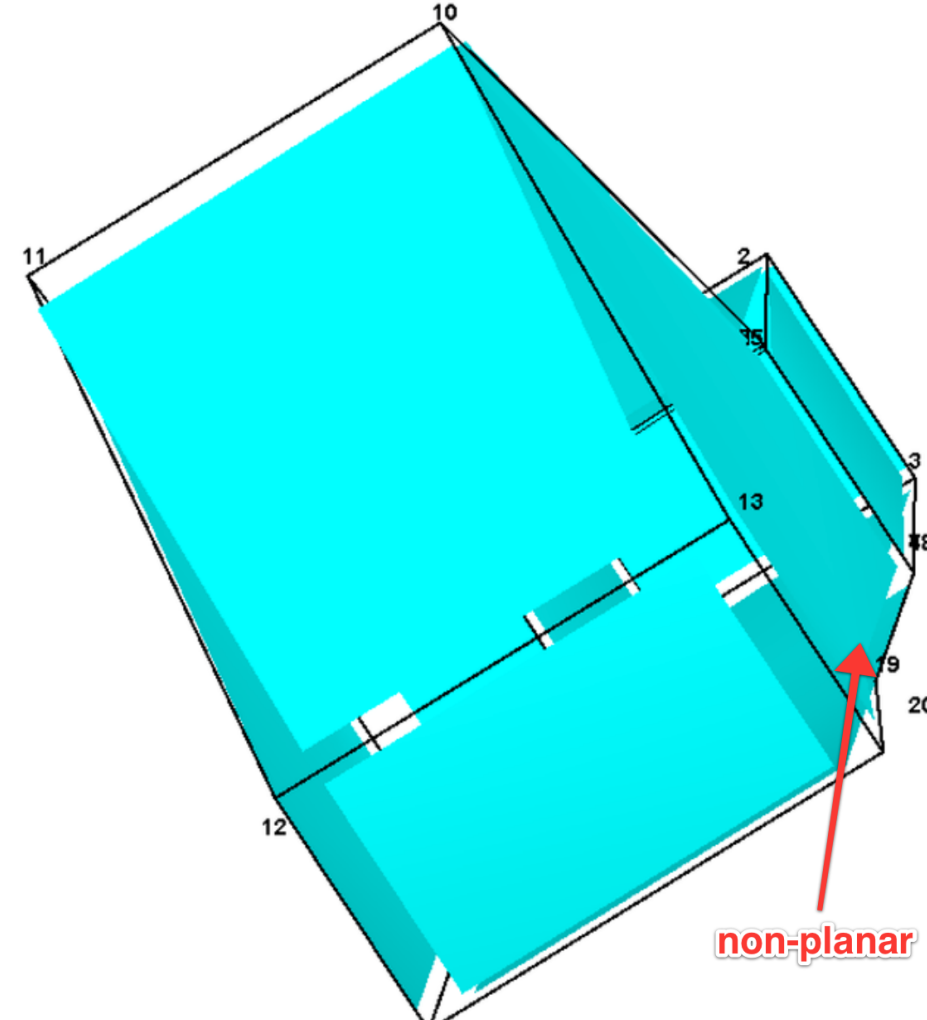
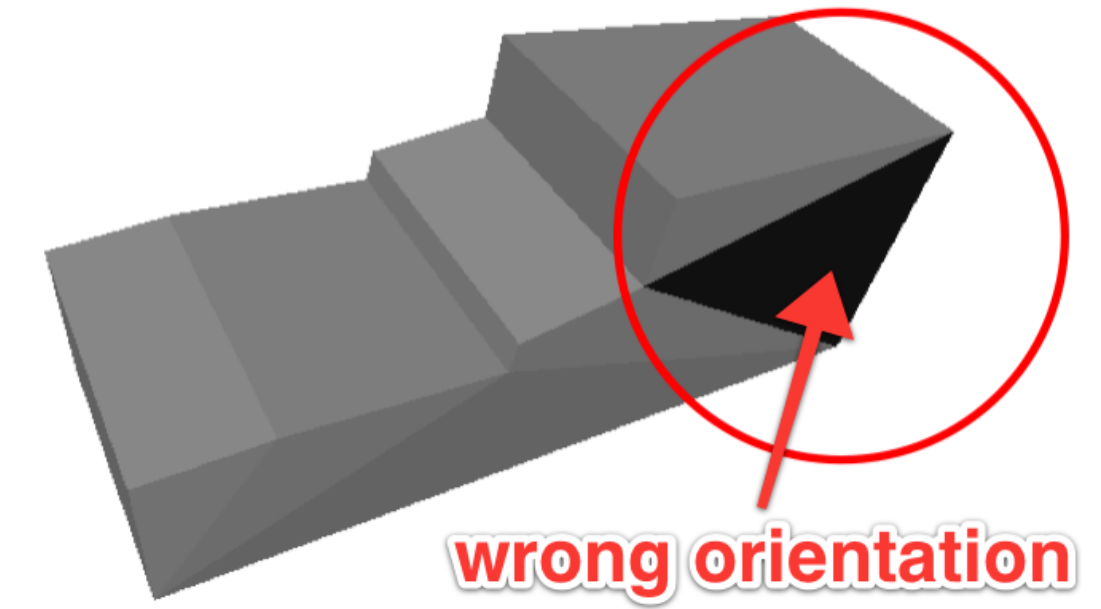
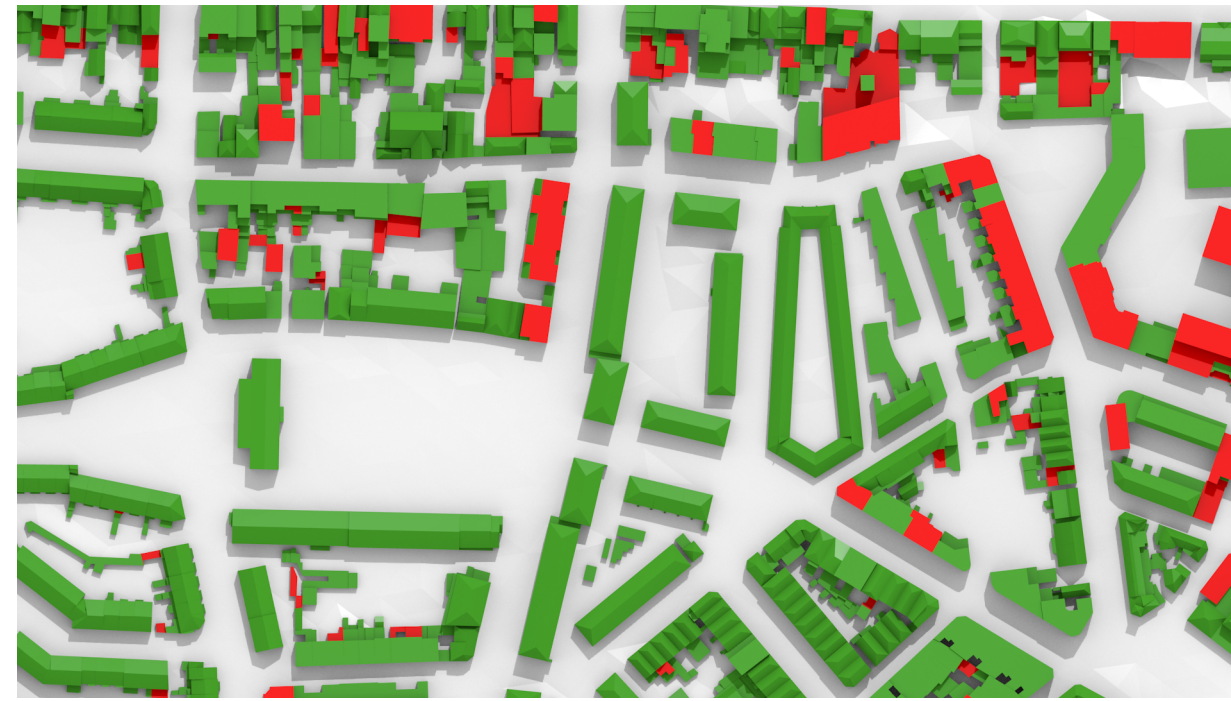
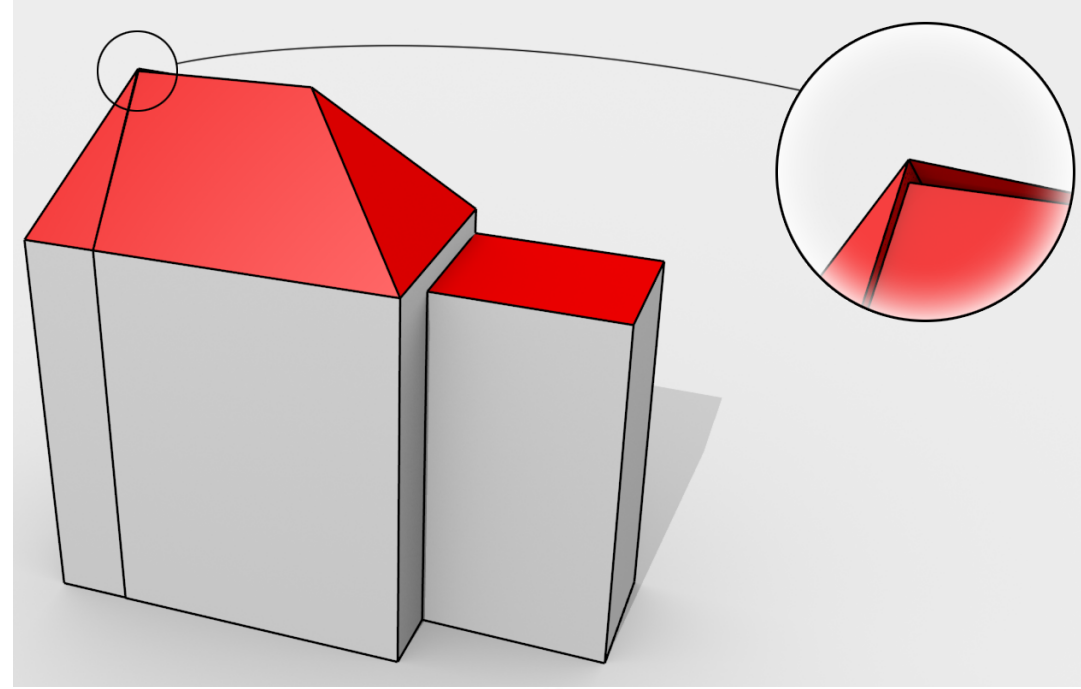


CompositeSolid

**Do current 3D city models
often contain these
geometric errors?**

tl;dl: **YES.**

Errors are *very* common in 3D models: the wall of shame

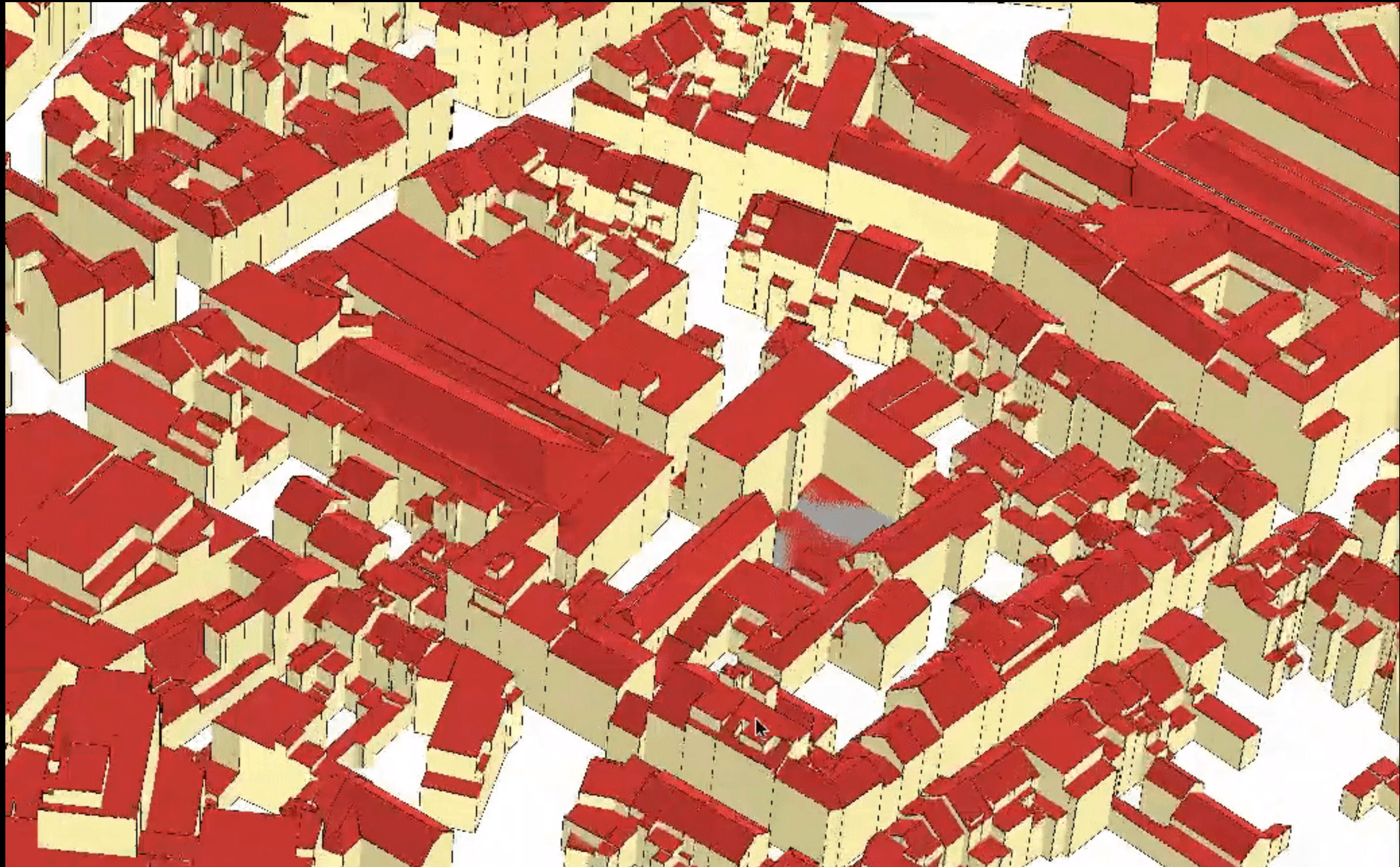


Should we care?

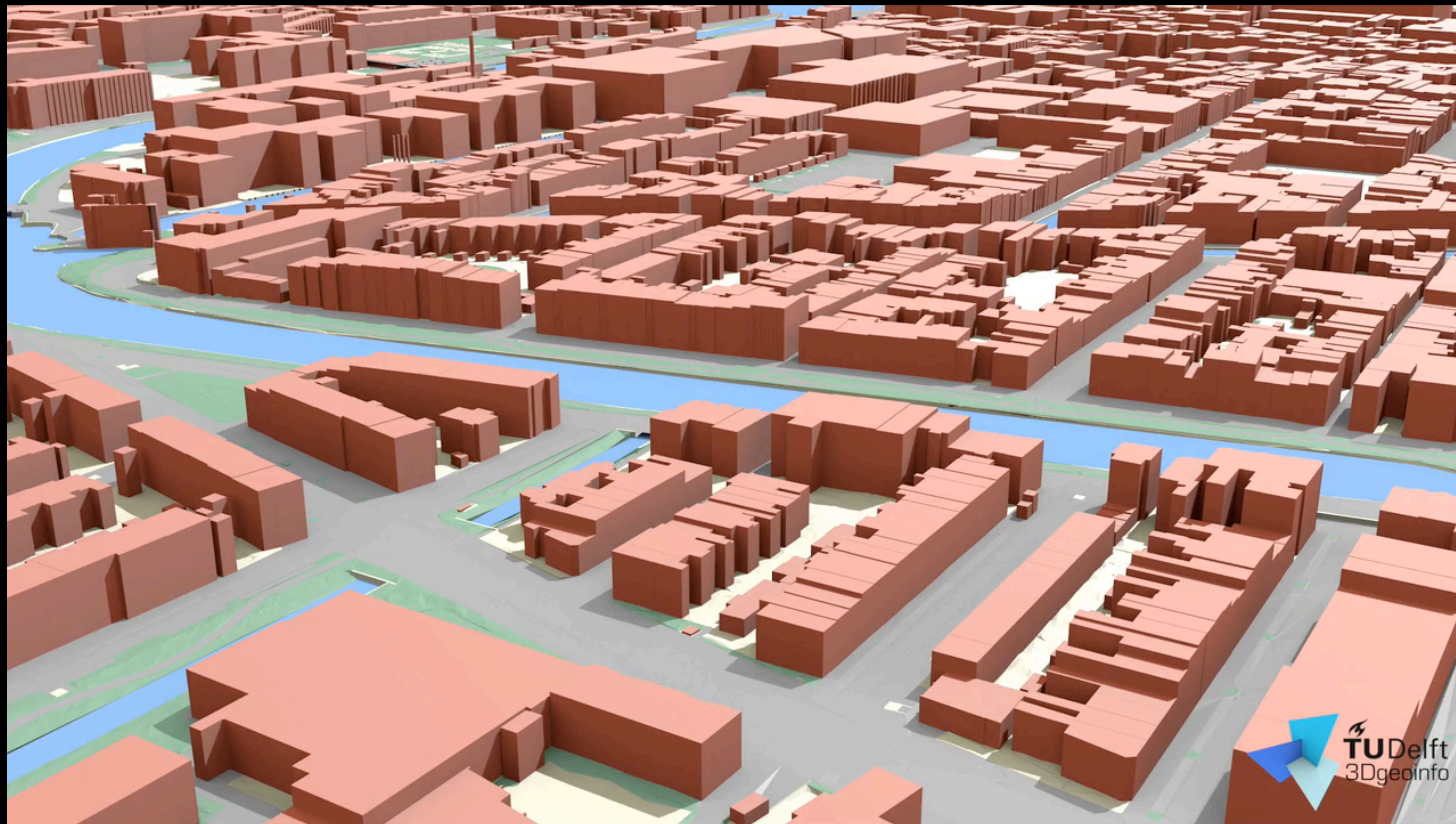
YES!

(especially if you want to pass this course 😄💧)

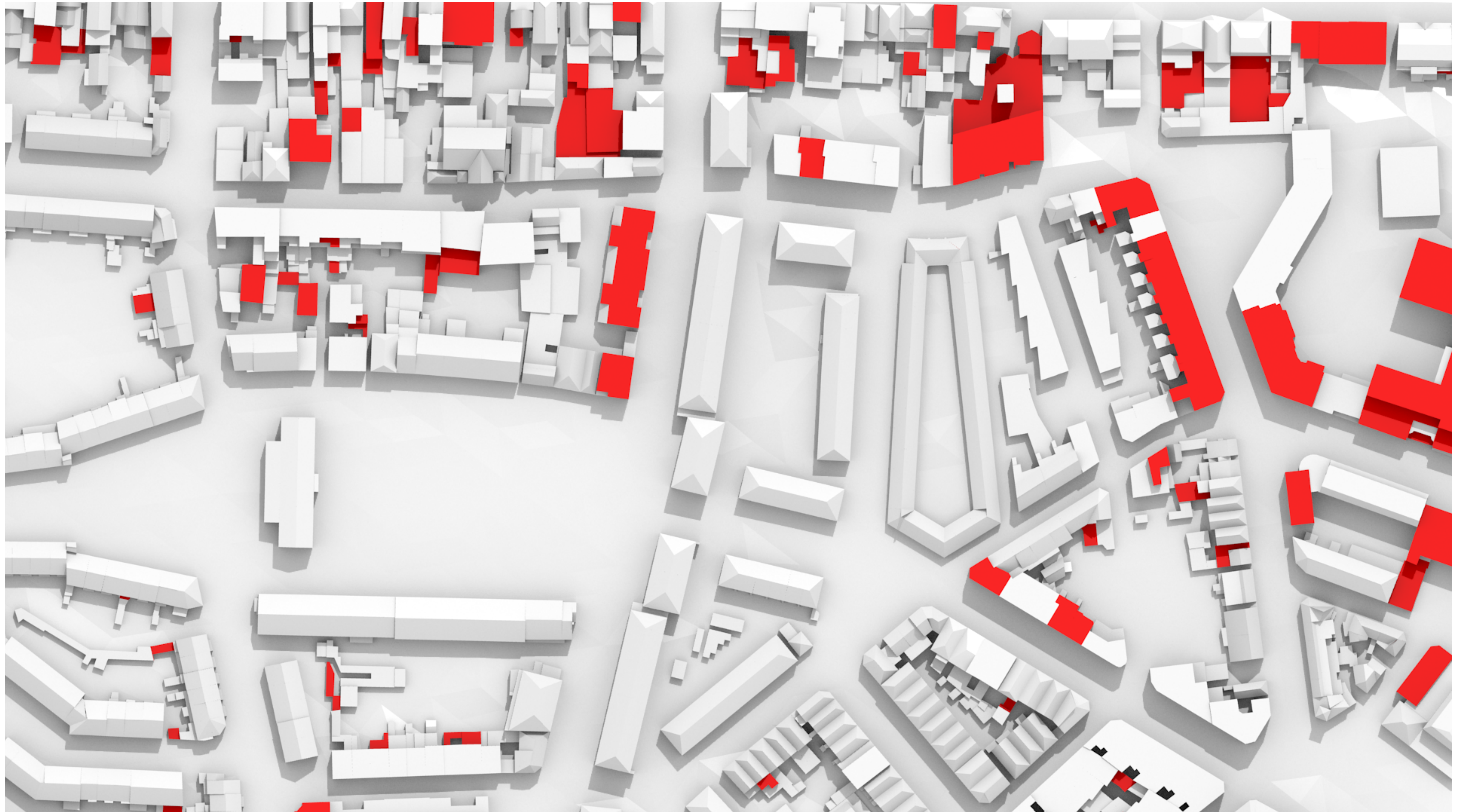
Visualisation – duplicated surfaces == annoying



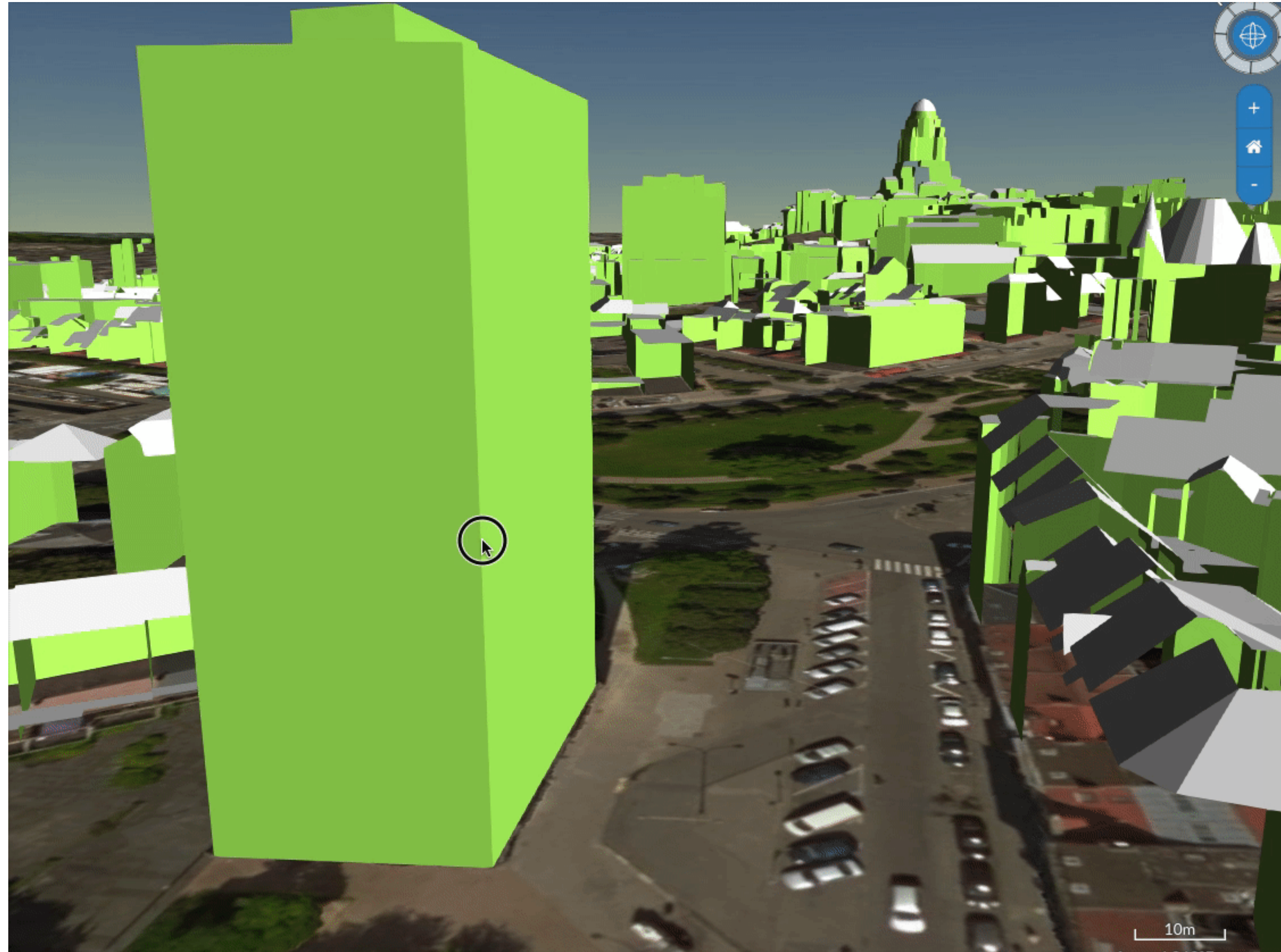
Visualisation – duplicated surfaces == distracting



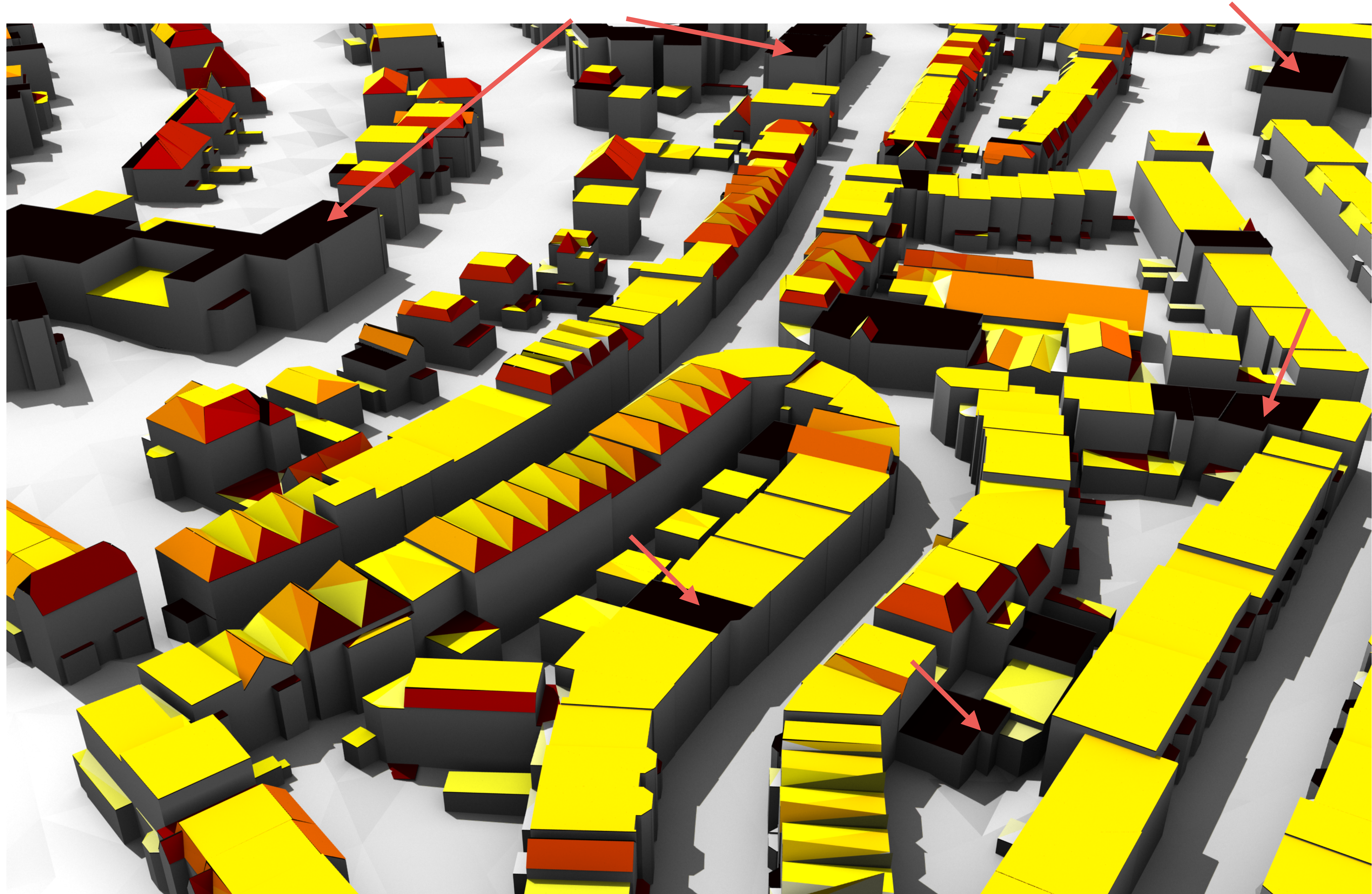
Visualisation – wrong orientation of surfaces (red ones)



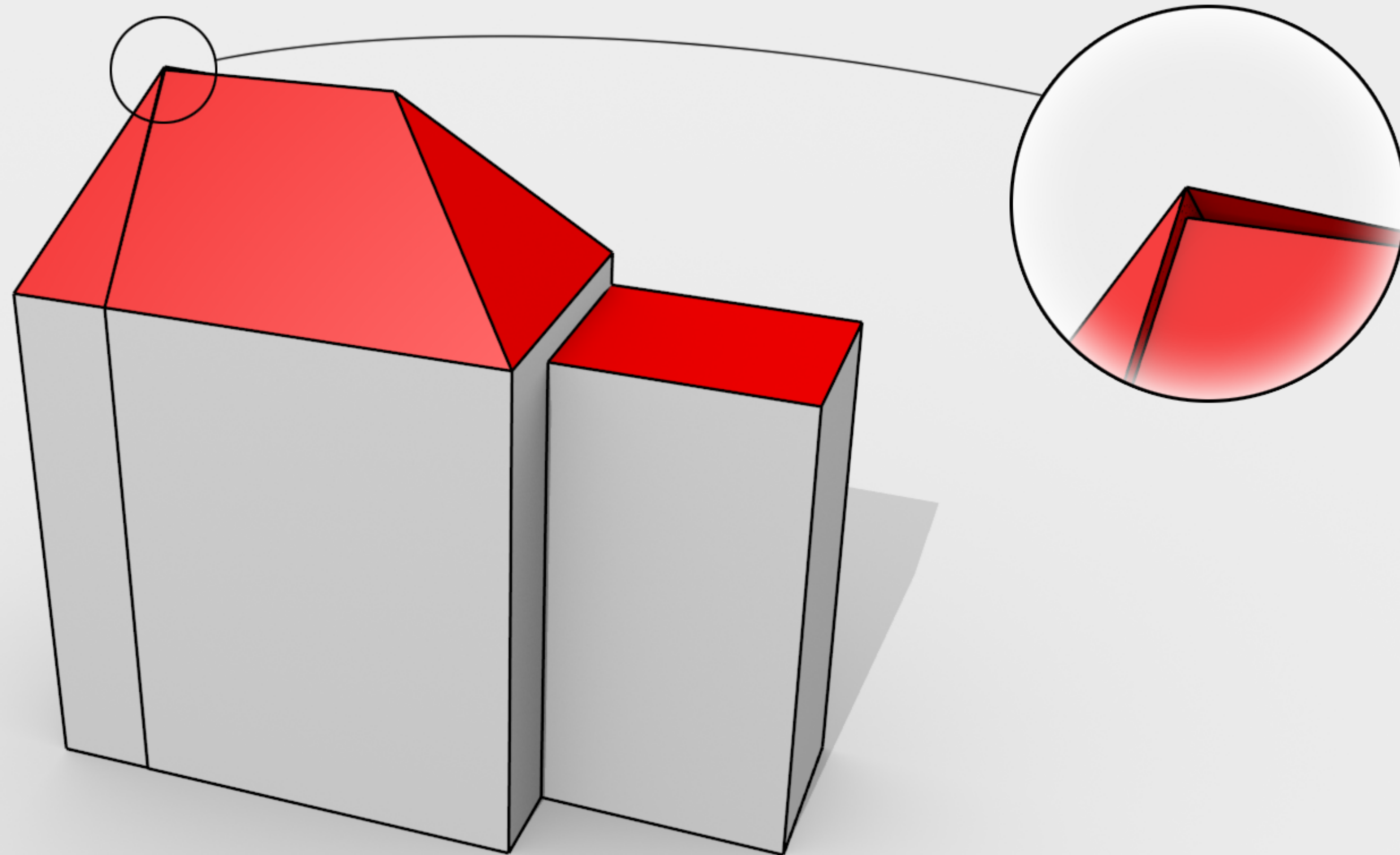
Visualisation – wrong orientation == missing faces



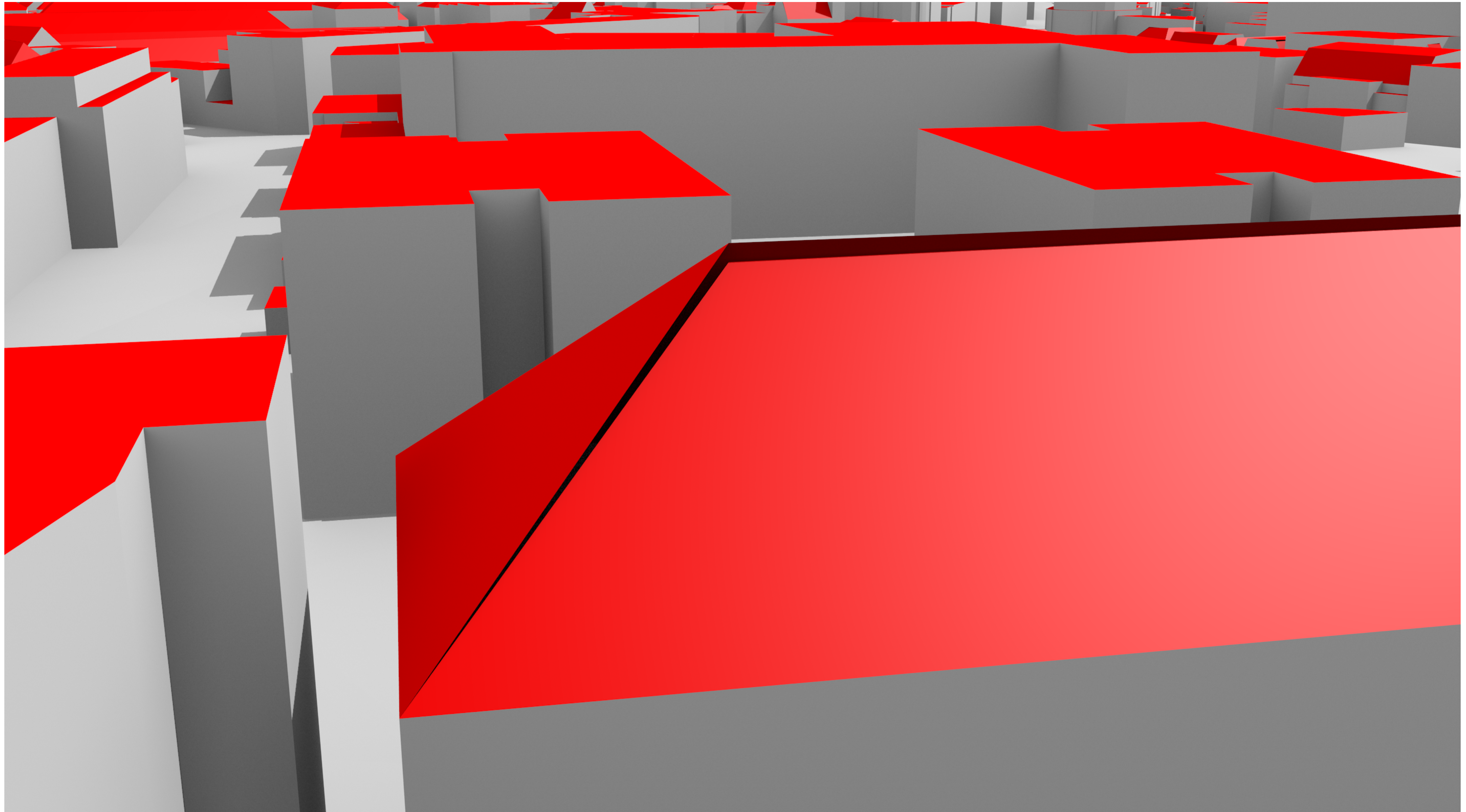
Solar potential – wrong orientation == no potential assigned



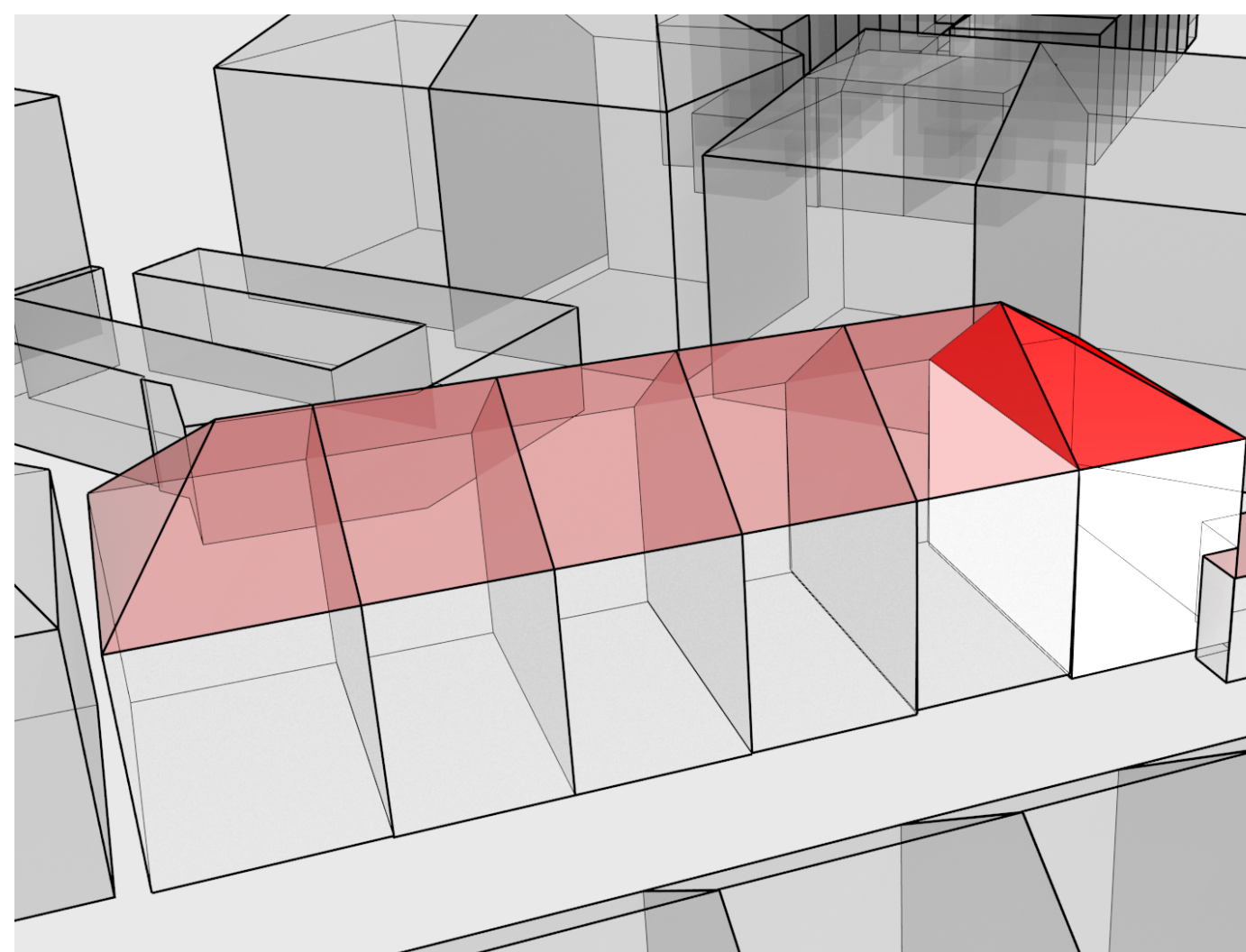
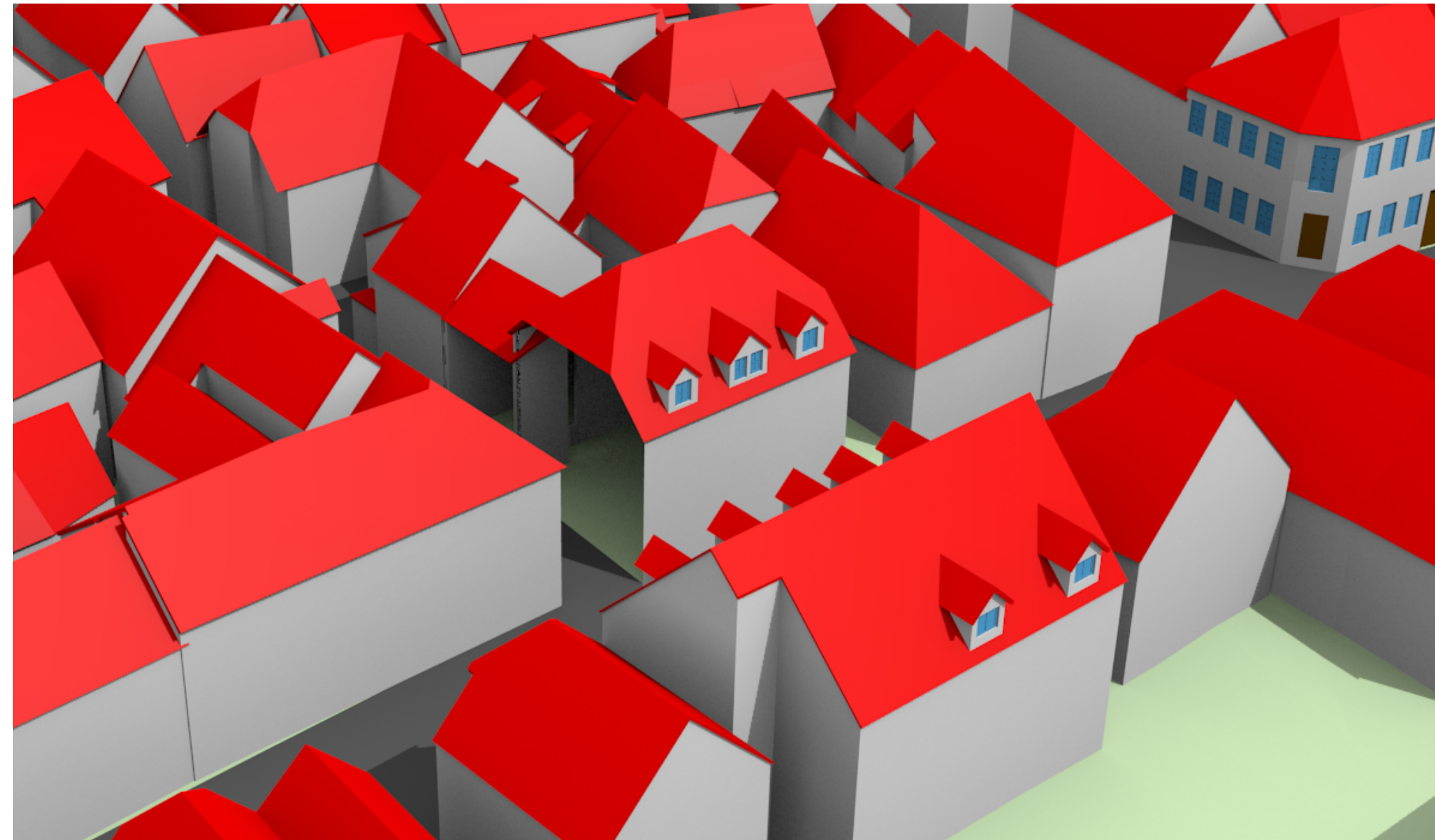
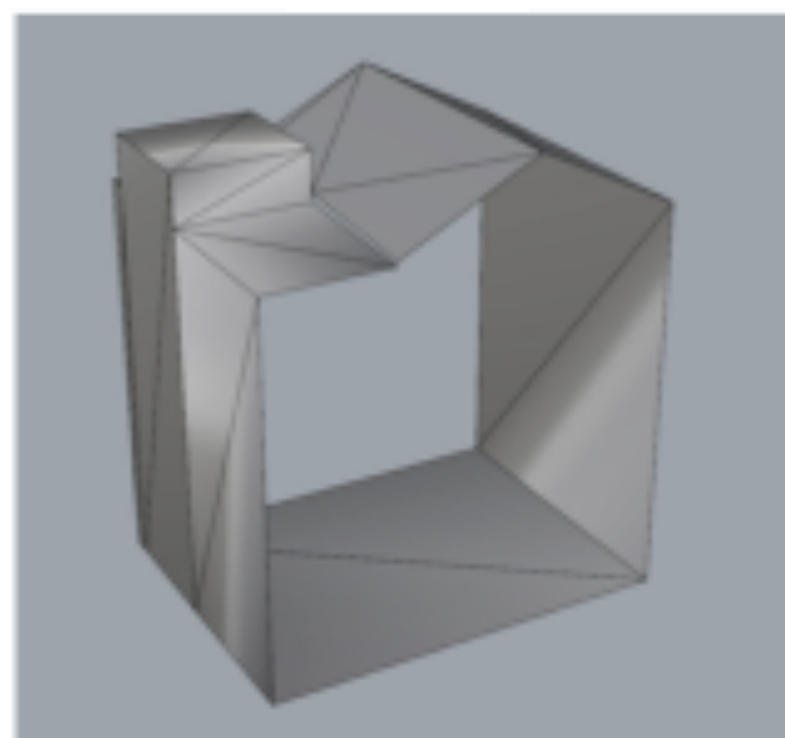
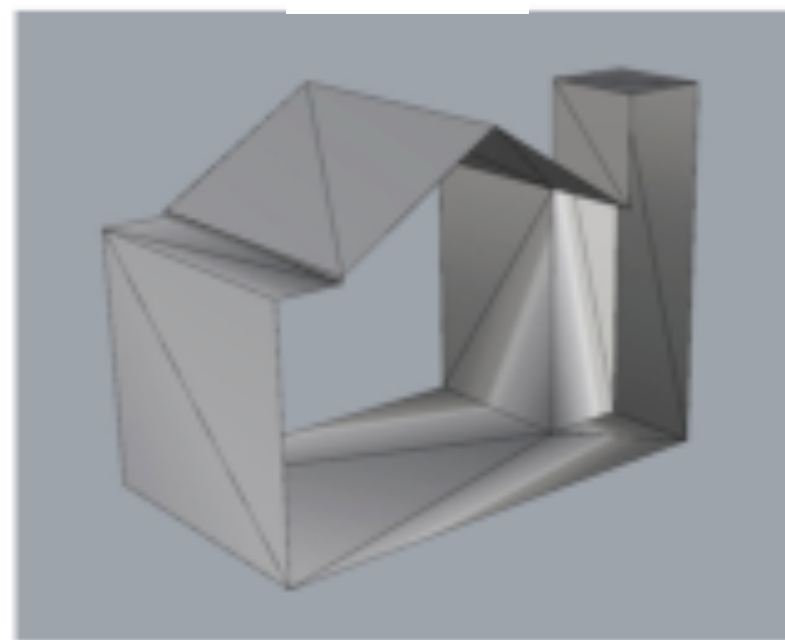
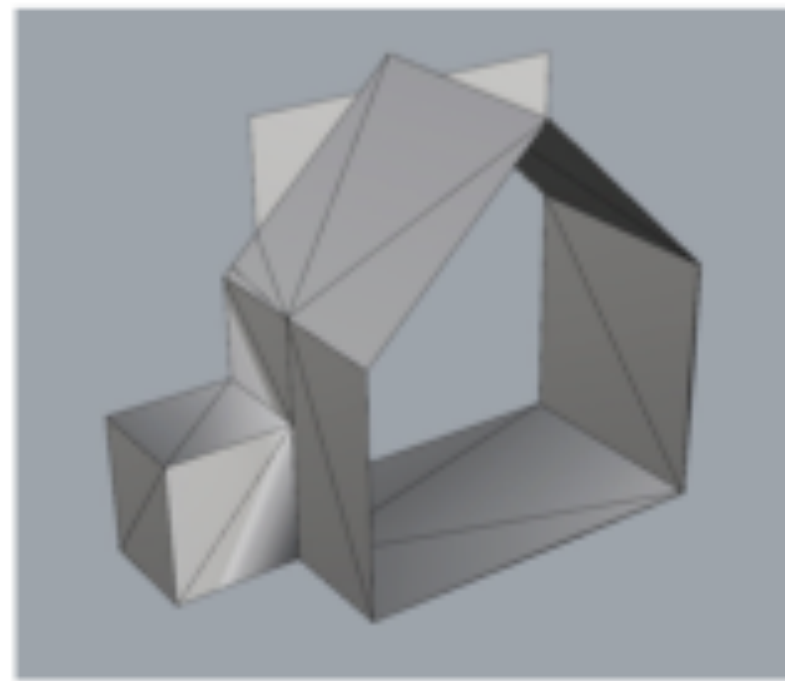
Volume calculation – tiny problems == some methods do no work



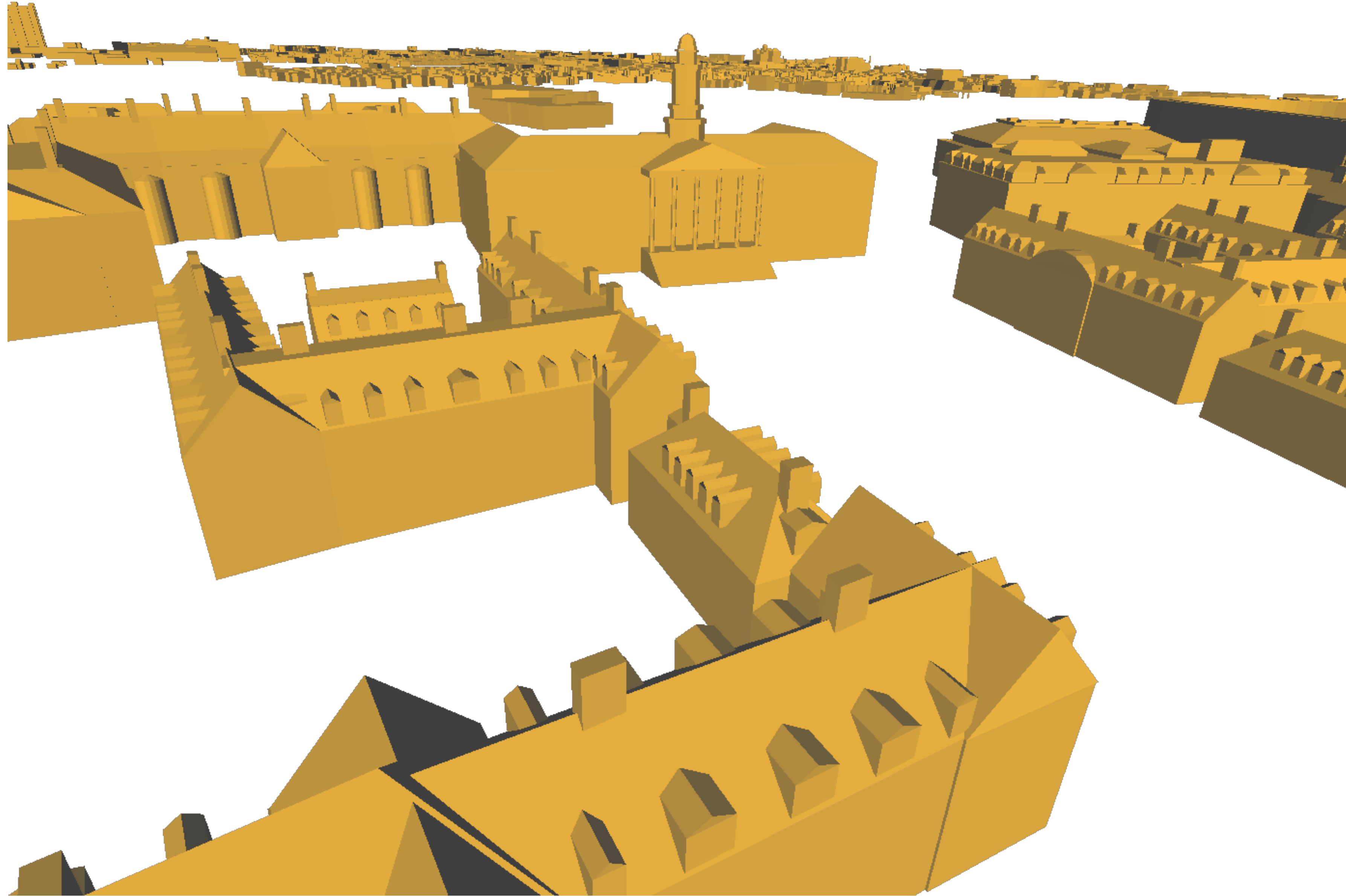
Volume calculation – tiny problems == some methods do no work



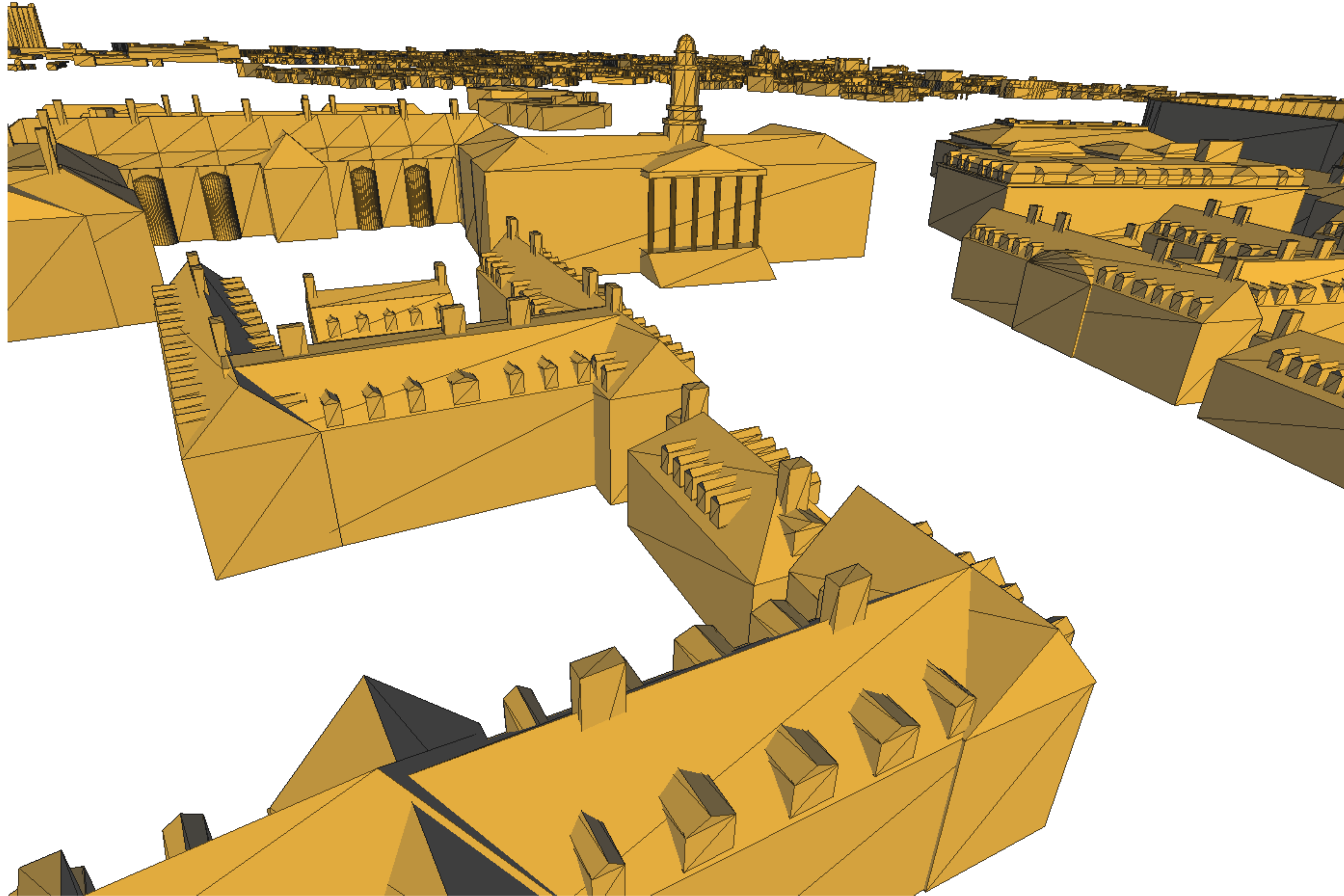
Volume calculation – big problems == some methods do no work



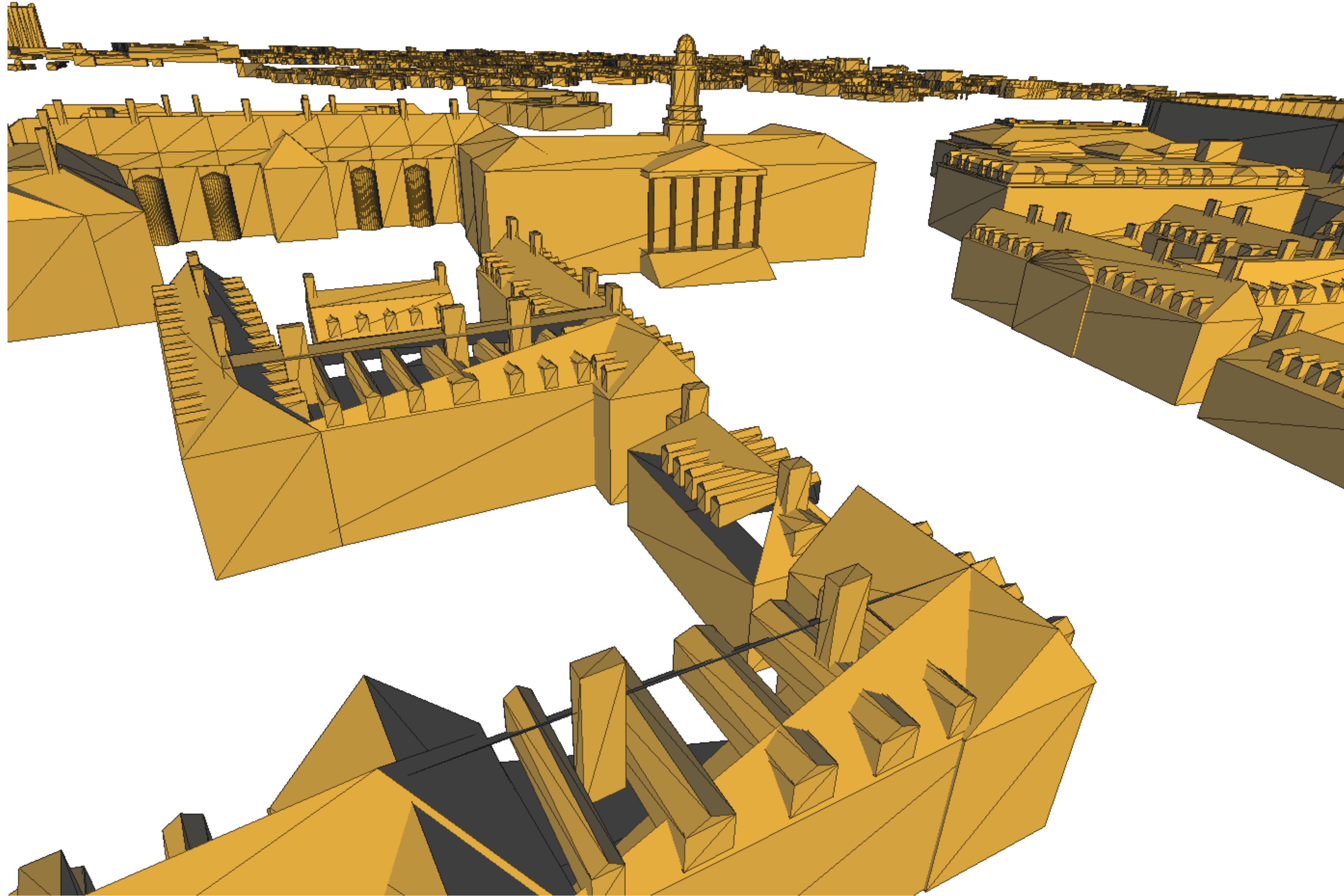
Volume calculation – superstructures not topologically connected



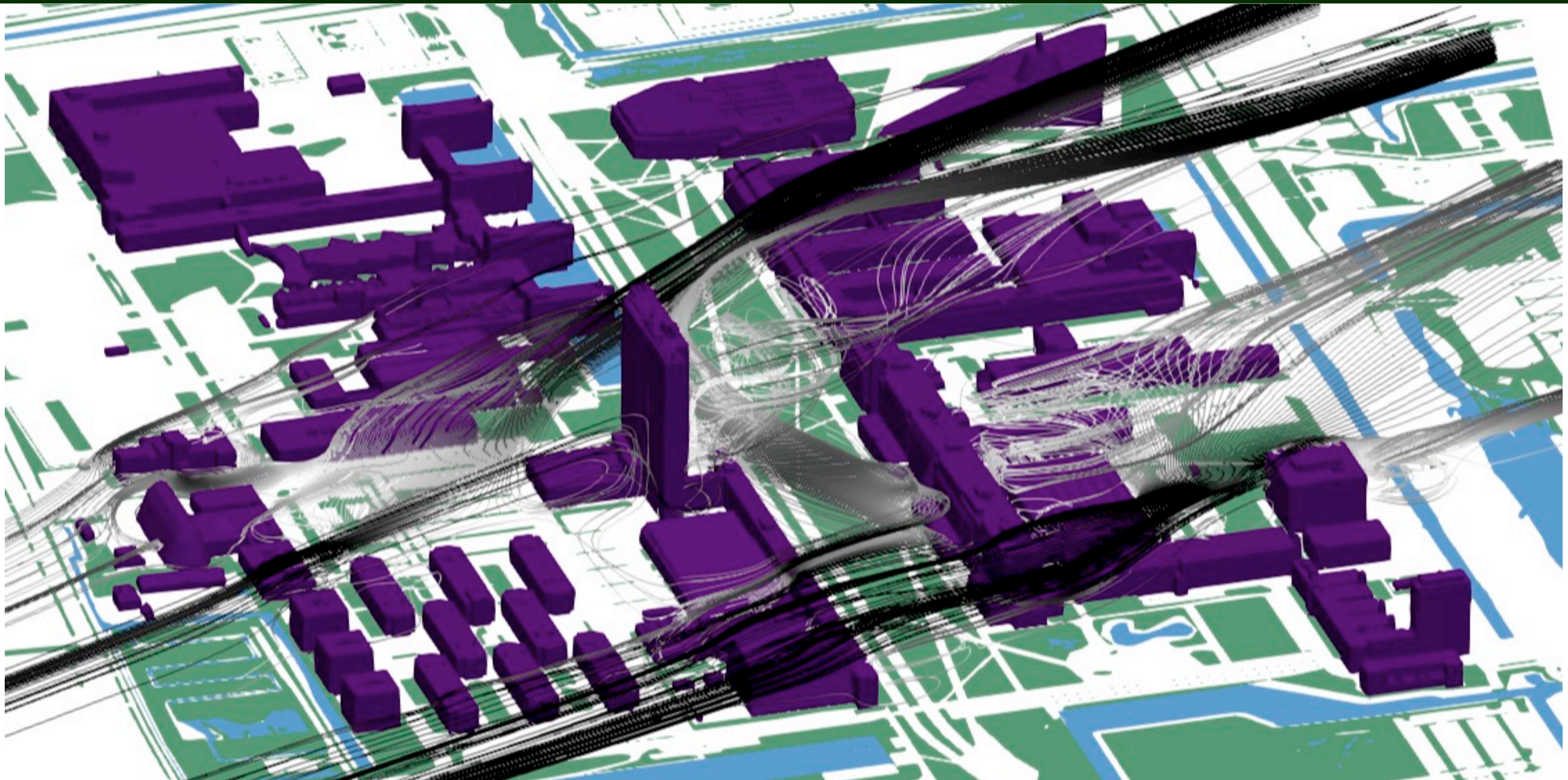
Volume calculation – superstructures not topologically connected



Volume calculation – superstructures not topologically connected



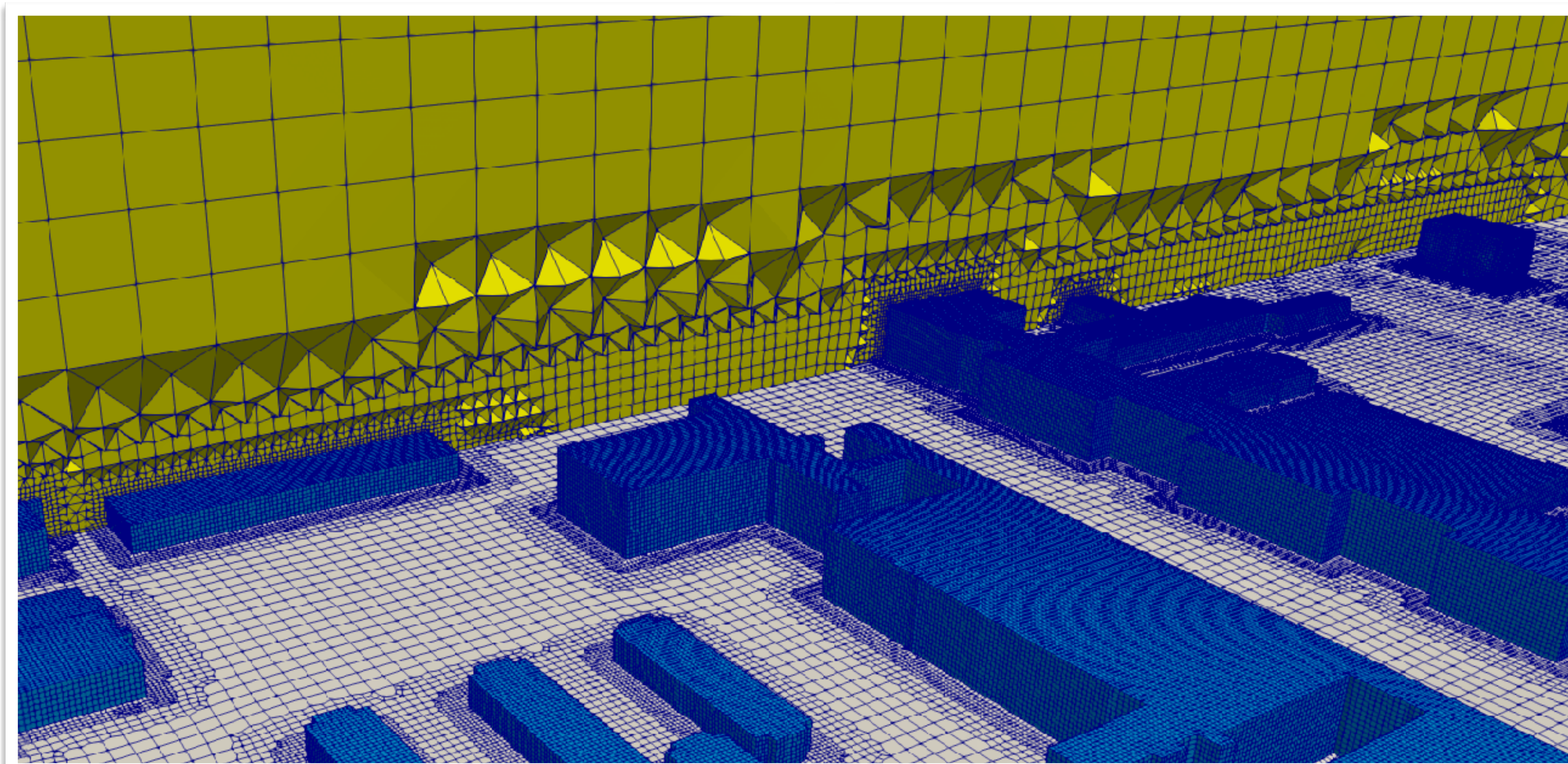
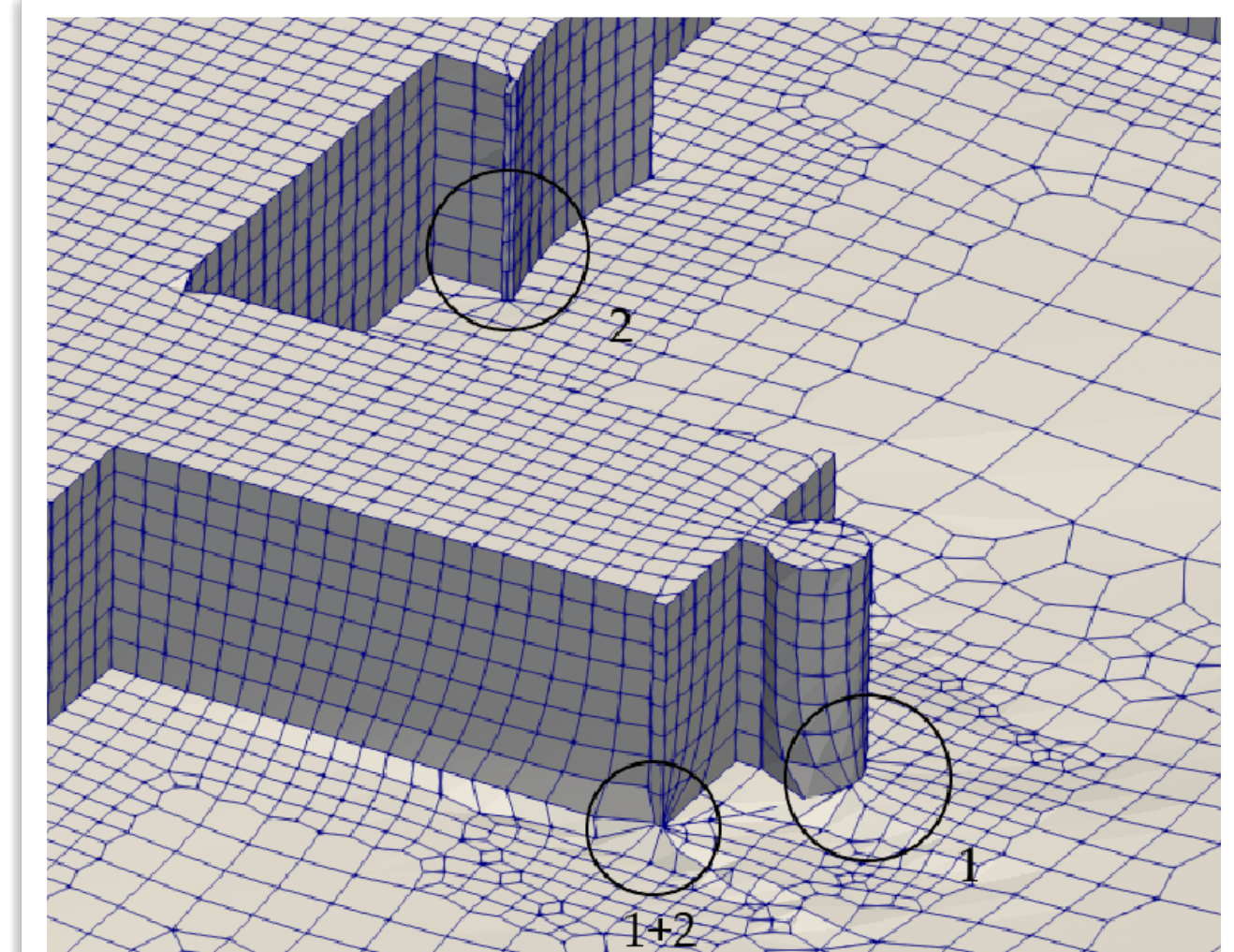
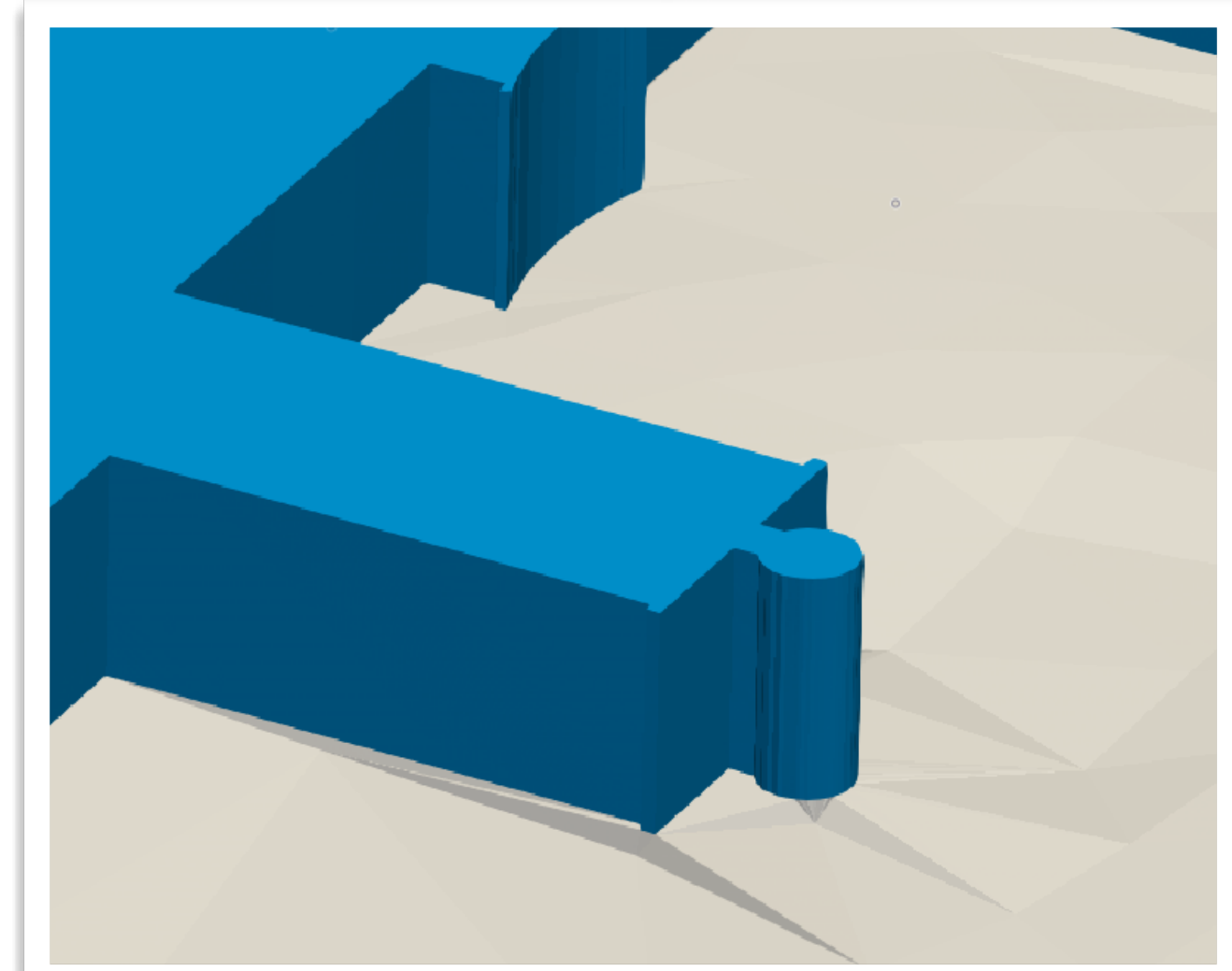
Advanced simulations – wind comfort for pedestrians



Advanced simulations – CFD methods: subdivision of volume (mesh)

computational fluid dynamics

GE05015: Modelling wind and dispersion in urban environments (Q4!)

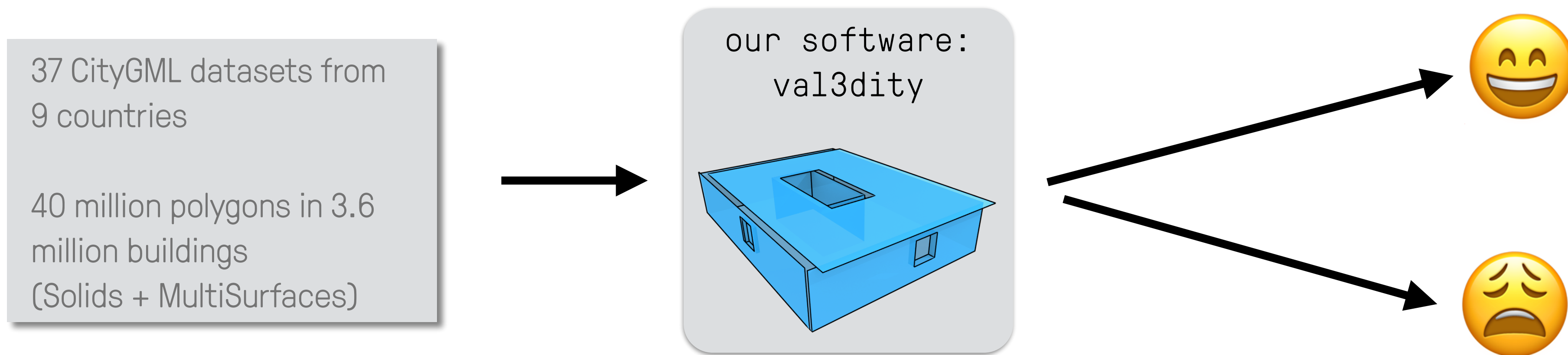


Advanced simulations – very strict input requirements

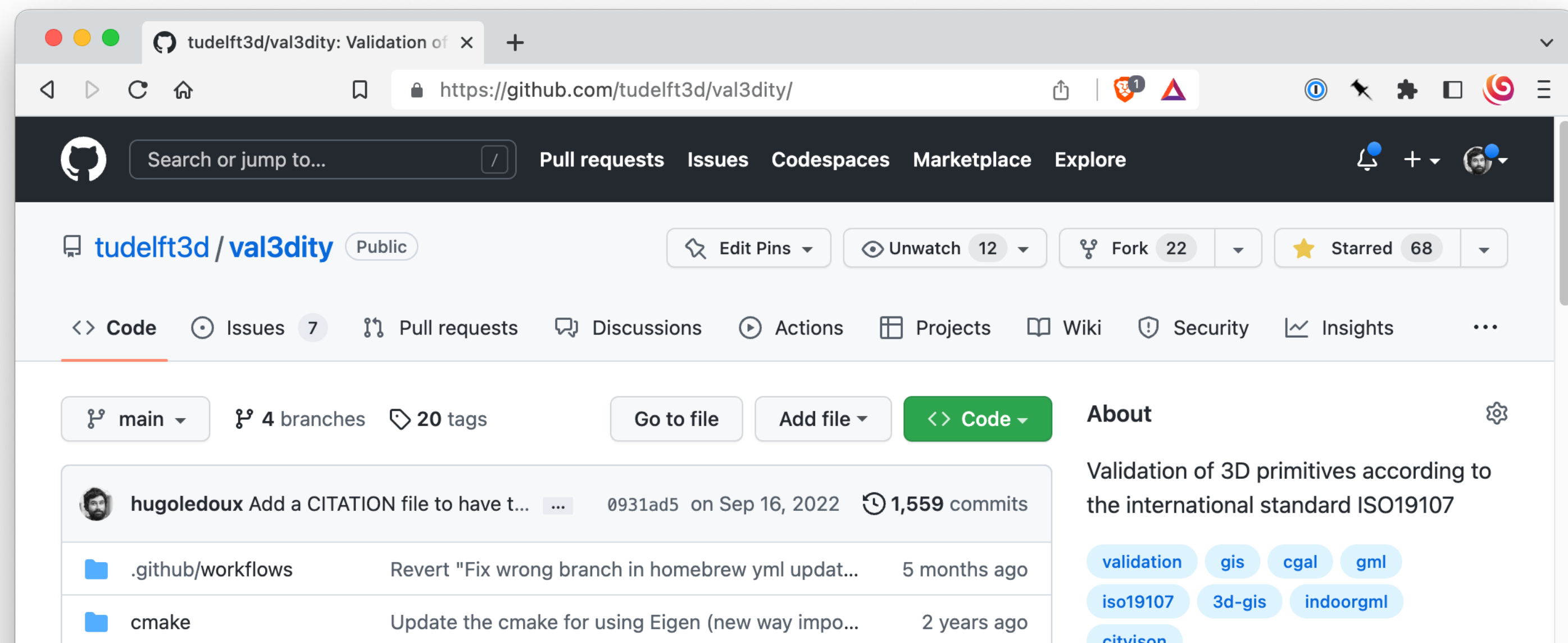
- no holes
- no intersections
- triangles must have certain shape (no *slivers*)
- a “perfect” model is thus required
- in practice, that still means several hours of (semi-)manual repair



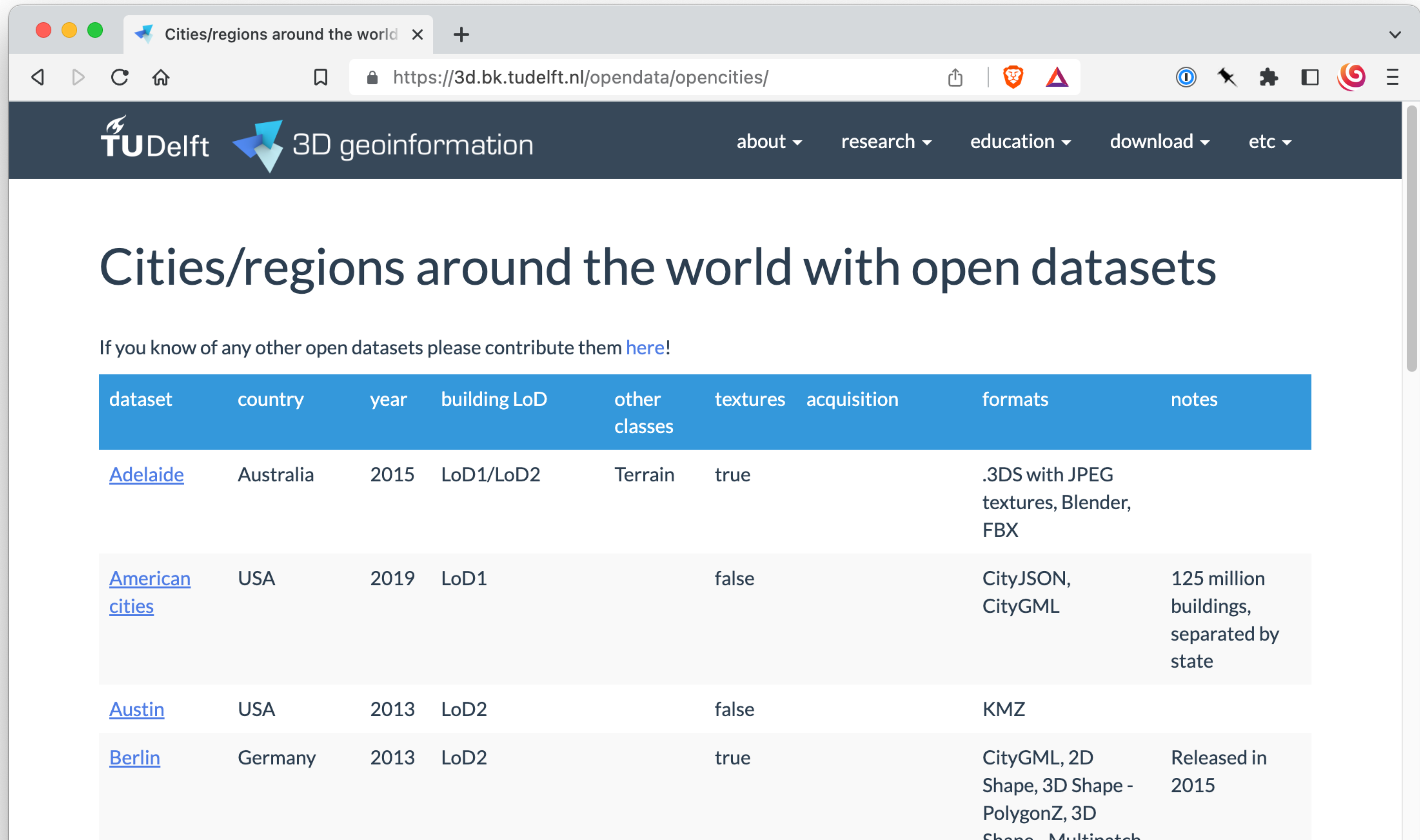
Our methodology



- 37 datasets (3D cities) from 9 countries
- 40 million polygons in 3.6 million buildings



Many cities around the world have 3D models openly available

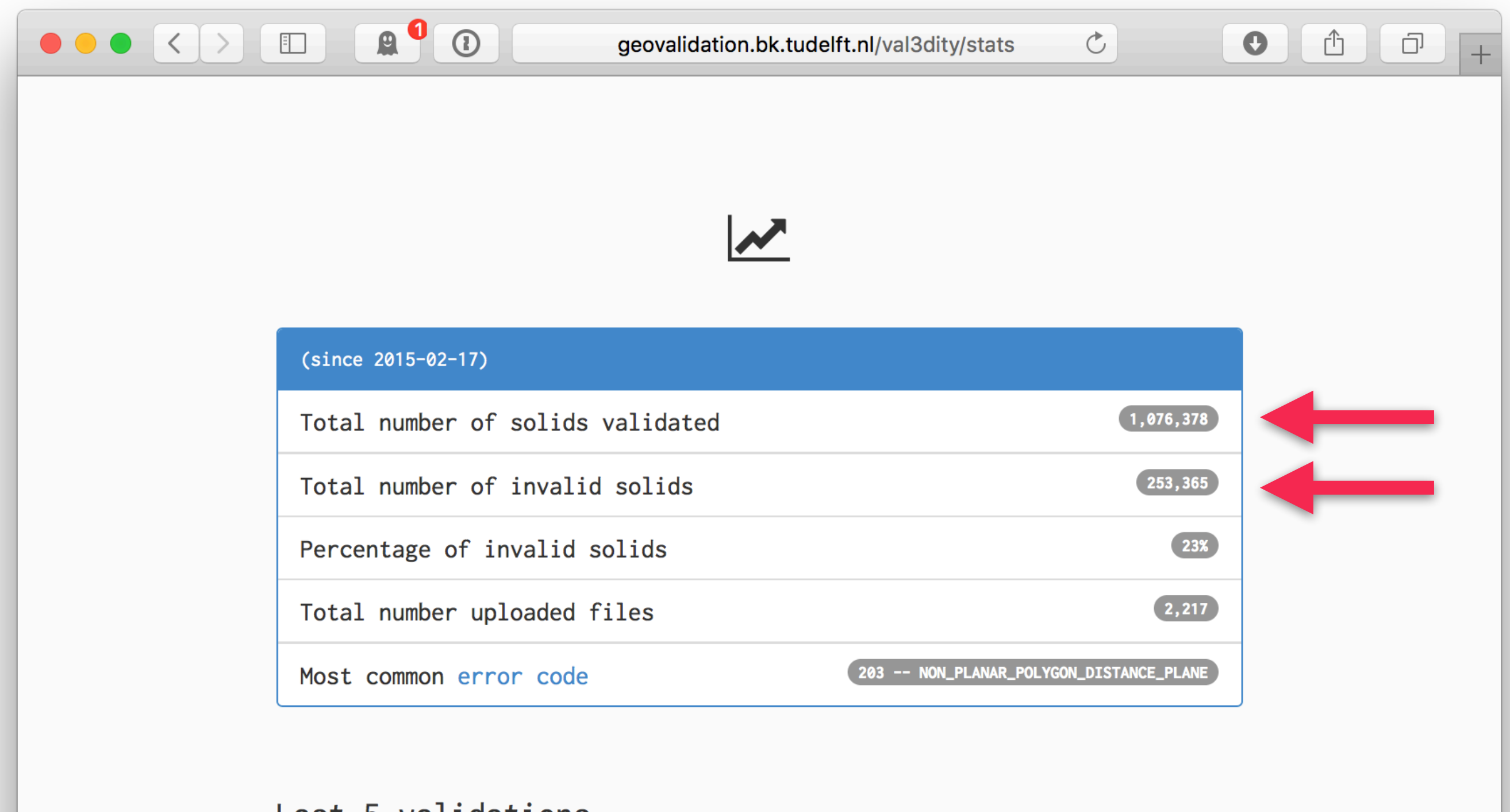


The screenshot shows a web browser window with the URL <https://3d.bk.tudelft.nl/opendata/opencities/>. The page title is "Cities/regions around the world with open datasets". Below the title, there is a text prompt: "If you know of any other open datasets please contribute them [here!](#)". The main content is a table with the following columns: dataset, country, year, building LoD, other classes, textures, acquisition, formats, and notes. The table lists four datasets: Adelaide (Australia, 2015, LoD1/LoD2, Terrain, true textures), American cities (USA, 2019, LoD1, false textures, 125 million buildings), Austin (USA, 2013, LoD2, false textures), and Berlin (Germany, 2013, LoD2, true textures).

dataset	country	year	building LoD	other classes	textures	acquisition	formats	notes
Adelaide	Australia	2015	LoD1/LoD2	Terrain	true		.3DS with JPEG textures, Blender, FBX	
American cities	USA	2019	LoD1		false		CityJSON, CityGML	125 million buildings, separated by state
Austin	USA	2013	LoD2		false		KMZ	
Berlin	Germany	2013	LoD2		true		CityGML, 2D Shape, 3D Shape - PolygonZ, 3D Shape, Multinatch	Released in 2015

Summary of results

- No 3D city dataset is 100% valid, not even LoD1 😞
- Many simple errors, eg:
 - repeated vertices
 - non-planar polygons (most common error!)
- Notice that these are often not visible



Summary of results

ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume IV-2/W1, 2016
11th 3D Geoinfo Conference, 20–21 October 2016, Athens, Greece

THE MOST COMMON GEOMETRIC AND SEMANTIC ERRORS IN CITYGML DATASETS

F. Biljecki ^{a,*}, H. Ledoux ^{a,*}, X. Du ^a, J. Stoter ^a, K. H. Soon ^b, V. H. S. Khoo ^b

^a 3D Geoinformation, Delft University of Technology, Delft, The Netherlands — (f.biljecki, h.ledoux, x.du-2, j.e.stoter)@tudelft.nl

^b Singapore Land Authority, Singapore — (soon_KEAN_huat, victor_khoo)@sla.gov.sg

KEY WORDS: CityGML, 3D city models, Quality control, Validation, Error

ABSTRACT:

To be used as input in most simulation and modelling software, 3D city models should be geometrically and topologically valid, and semantically rich. We investigate in this paper what is the quality of currently available CityGML datasets, i.e. we validate the geometry/topology of the 3D primitives (*Solid* and *MultiSurface*), and we validate whether the semantics of the boundary surfaces of buildings is correct or not. We have analysed all the CityGML datasets we could find, both from portals of cities and on different websites, plus a few that were made available to us. We have thus validated 40M surfaces in 16M 3D primitives and 3.6M buildings found in 37 CityGML datasets originating from 9 countries, and produced by several companies with diverse software and acquisition techniques. The results indicate that CityGML datasets without errors are rare, and those that are nearly valid are mostly simple LOD1 models. We report on the most common errors we have found, and analyse them. One main observation is that many of these errors could be automatically fixed or prevented with simple modifications to the modelling software. Our principal aim is to highlight the most common errors so that these are not repeated in the future. We hope that our paper and the open-source software we have developed will help raise awareness for data quality among data providers and 3D GIS software producers.

1. INTRODUCTION

In recent years, several cities around the world have released their 3D city models (3DGM).

Related work considers additional aspects, e.g. resolvability of *XLinks*; application-specific rules, such as that a building is required to have a ground floor to form a volume (which is, surprisingly, not mandatory in CityGML); and the redundancy of the

Level of detail	Primitive	ID ^(a)	Geometric validation				Semantics ^(f)	Schema	
			1xx	2xx	3xx	4xx			Total ^(b)
LOD1	Solid	1	0	0	0	0	0	-	X
		2	0	0	0	0	0	-	✓
		3	0	0	0	0	0	-	✓
		4	0	0	0	0	0	-	✓
		5	0	0	0	0	0	-	X
		6	0	0	0	0	0	-	X
		7	0	0	0	0	0	-	✓
		8	0	0	0	0	0	-	✓
		9	0	0	0	0	0	-	X
LOD2	MultiSurface	10	1	4	-	-	5	(e)	✓
		11	0	0	-	-	0	0	✓
		12	2	21	-	-	23	45	✓
		13	10	2	-	-	12	4	X
		14	0	1	-	-	1	12	✓
		15	0	9	-	-	9	2	✓
		16	4	8	-	-	12	1	X
		17	5	0	-	-	5	5	X
		18	0	0	-	-	0	4	X
		19	0	0	-	-	0	1	X
		20	0	4	-	-	4	6	X
21	0	1	-	-	1	3	X		
LOD2	Solid	22	0	42	58	0	100	-	✓
		23 ^(c)	-	-	-	-	-	-	X
		24	0	31	1	3	35	-	X
		25	4	0	16	2	22	-	✓
		26 ^(c)	-	-	-	-	-	-	X
		27	22	17	50	0	89	-	✓
LOD2	MultiSurface and Solid ^(d)	28	0	42	1	1	44	0	✓
		29	2	35	54	0	92	4	✓
		30	0	10	0	1	11	2	✓
		31	0	0	0	0	0	0	✓
LOD3	MultiSurface	32	2	13	-	-	15	54	✓
		33	6	5	-	-	11	23	✓
		34	8	10	-	-	19	45	✓
		35	5	0	-	-	5	34	✓
		36	0	0	-	-	0	1	X
LOD4	Solid	37	0	0	3	0	3	68	✓

% of invalid

**actually all between
99.5% and 99.9%**

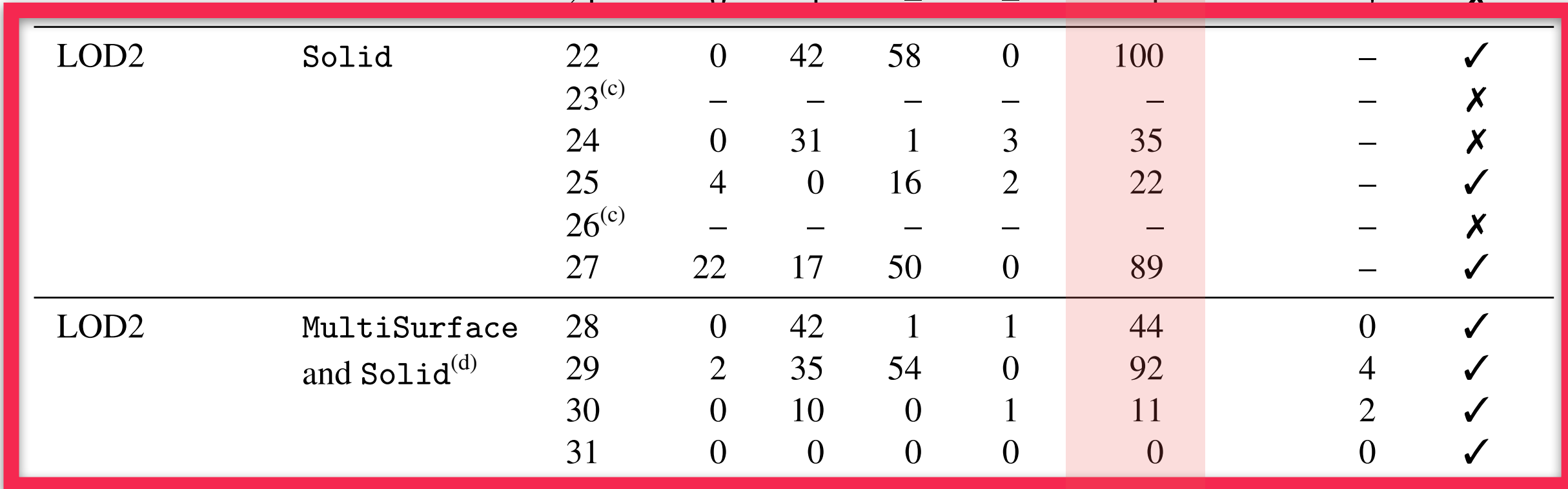
Level of detail	Primitive	ID ^(a)	Geometric validation				Semantics ^(f)	Schema	
			1xx	2xx	3xx	4xx			Total ^(b)
LOD1	Solid	1	0	0	0	0	0	-	X
		2	0	0	0	0	0	-	✓
		3	0	0	0	0	0	-	✓
		4	0	0	0	0	0	-	✓
		5	0	0	0	0	0	-	X
		6	0	0	0	0	0	-	X
		7	0	0	0	0	0	-	✓
		8	0	0	0	0	0	-	✓
		9	0	0	0	0	0	-	X
LOD2	MultiSurface	10	1	4	-	-	5	(e)	✓
		11	0	0	-	-	0	0	✓
		12	2	21	-	-	23	45	✓
		13	10	2	-	-	12	4	X
		14	0	1	-	-	1	12	✓
		15	0	9	-	-	9	2	✓
		16	4	8	-	-	12	1	X
		17	5	0	-	-	5	5	X
		18	0	0	-	-	0	4	X
		19	0	0	-	-	0	1	X
		20	0	4	-	-	4	6	X
21	0	1	-	-	1	3	X		
LOD2	Solid	22	0	42	58	0	100	-	✓
		23 ^(c)	-	-	-	-	-	-	X
		24	0	31	1	3	35	-	X
		25	4	0	16	2	22	-	✓
		26 ^(c)	-	-	-	-	-	-	X
		27	22	17	50	0	89	-	✓
LOD2	MultiSurface and Solid ^(d)	28	0	42	1	1	44	0	✓
		29	2	35	54	0	92	4	✓
		30	0	10	0	1	11	2	✓
		31	0	0	0	0	0	0	✓
LOD3	MultiSurface	32	2	13	-	-	15	54	✓
		33	6	5	-	-	11	23	✓
		34	8	10	-	-	19	45	✓
		35	5	0	-	-	5	34	✓
		36	0	0	-	-	0	1	X
LOD4	Solid	37	0	0	3	0	3	68	✓

% of invalid

rather good, but it's
"easy"

Level of detail	Primitive	ID ^(a)	Geometric validation				Semantics ^(f)	Schema	
			1xx	2xx	3xx	4xx			Total ^(b)
LOD1	Solid	1	0	0	0	0	0	-	X
		2	0	0	0	0	0	-	✓
		3	0	0	0	0	0	-	✓
		4	0	0	0	0	0	-	✓
		5	0	0	0	0	0	-	X
		6	0	0	0	0	0	-	X
		7	0	0	0	0	0	-	✓
		8	0	0	0	0	0	-	✓
		9	0	0	0	0	0	-	X
LOD2	MultiSurface	10	1	4	-	-	5	(e)	✓
		11	0	0	-	-	0	0	✓
		12	2	21	-	-	23	45	✓
		13	10	2	-	-	12	4	X
		14	0	1	-	-	1	12	✓
		15	0	9	-	-	9	2	✓
		16	4	8	-	-	12	1	X
		17	5	0	-	-	5	5	X
		18	0	0	-	-	0	4	X
		19	0	0	-	-	0	1	X
		20	0	4	-	-	4	6	X
21	0	1	-	-	1	3	X		
LOD2	Solid	22	0	42	58	0	100	-	✓
		23 ^(c)	-	-	-	-	-	-	X
		24	0	31	1	3	35	-	X
		25	4	0	16	2	22	-	✓
		26 ^(c)	-	-	-	-	-	-	X
		27	22	17	50	0	89	-	✓
LOD2	MultiSurface and Solid ^(d)	28	0	42	1	1	44	0	✓
		29	2	35	54	0	92	4	✓
		30	0	10	0	1	11	2	✓
		31	0	0	0	0	0	0	✓
LOD3	MultiSurface	32	2	13	-	-	15	54	✓
		33	6	5	-	-	11	23	✓
		34	8	10	-	-	19	45	✓
		35	5	0	-	-	5	34	✓
		36	0	0	-	-	0	1	X
LOD4	Solid	37	0	0	3	0	3	68	✓

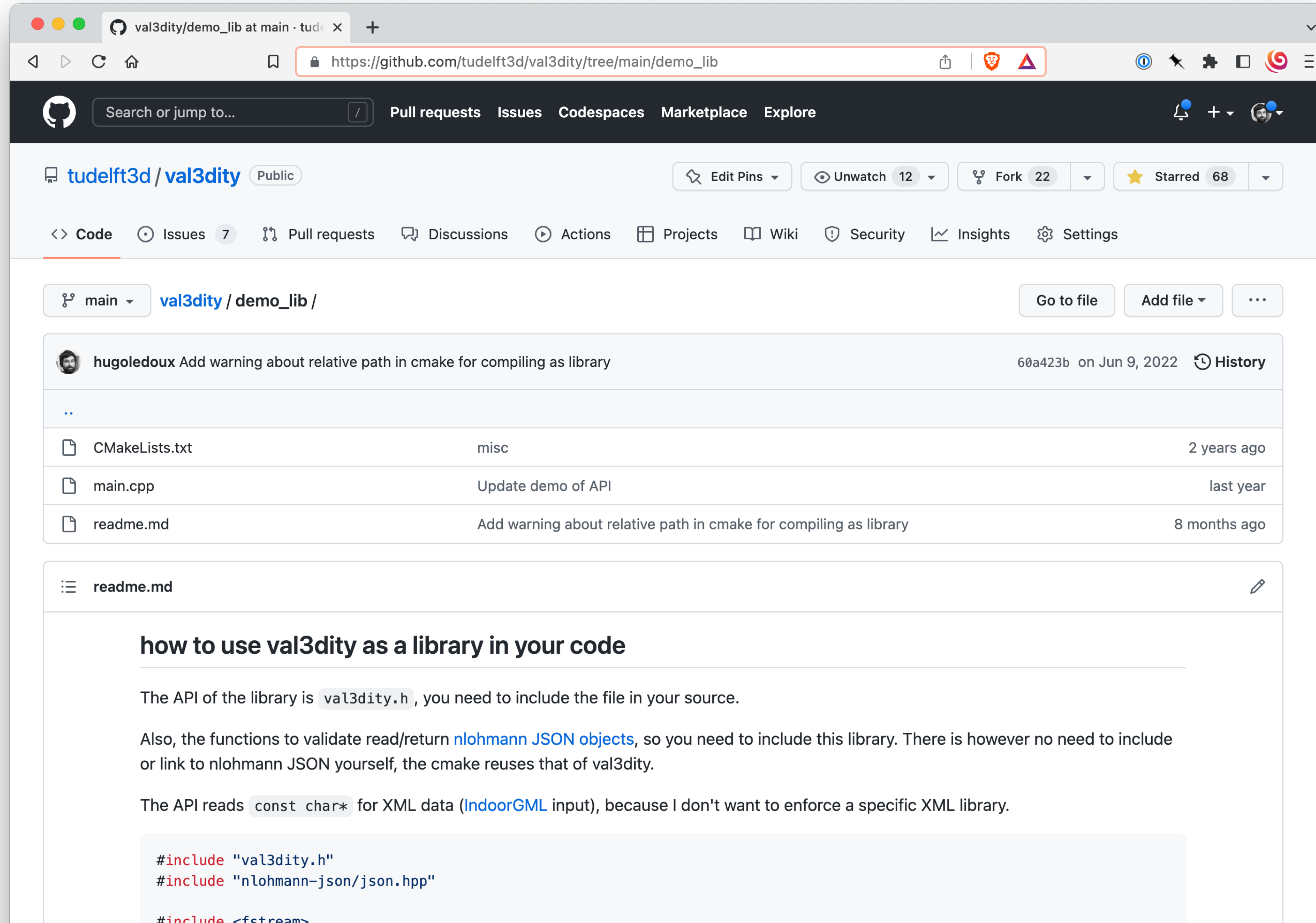
% of invalid



some datasets couldn't be read

high % of invalid

Using val3dity as a library in C++



val3dity/demo_lib at main · tudelft3d/val3dity

Search or jump to... Pull requests Issues Codespaces Marketplace Explore

tudelft3d / val3dity Public

Edit Pins Unwatch 12 Fork 22 Starred 68

Code Issues 7 Pull requests Discussions Actions Projects Wiki Security Insights Settings

main val3dity / demo_lib / Go to file Add file ...

hugoledoux Add warning about relative path in cmake for compiling as library 60a423b on Jun 9, 2022 History

..		
CMakeLists.txt	misc	2 years ago
main.cpp	Update demo of API	last year
readme.md	Add warning about relative path in cmake for compiling as library	8 months ago

readme.md

how to use val3dity as a library in your code

The API of the library is `val3dity.h`, you need to include the file in your source.

Also, the functions to validate read/return [nlohmann JSON objects](#), so you need to include this library. There is however no need to include or link to nlohmann JSON yourself, the cmake reuses that of val3dity.

The API reads `const char*` for XML data ([IndoorGML](#) input), because I don't want to enforce a specific XML library.

```
#include "val3dity.h"
#include "nlohmann-json/json.hpp"
#include <fstream>
```