Lesson 3.2
Three-dimensional geometries in geoinformation

GEO1004:
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**: **In**frastructure for **S**patial **I**nfo**r**mation in **E**urope
- **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

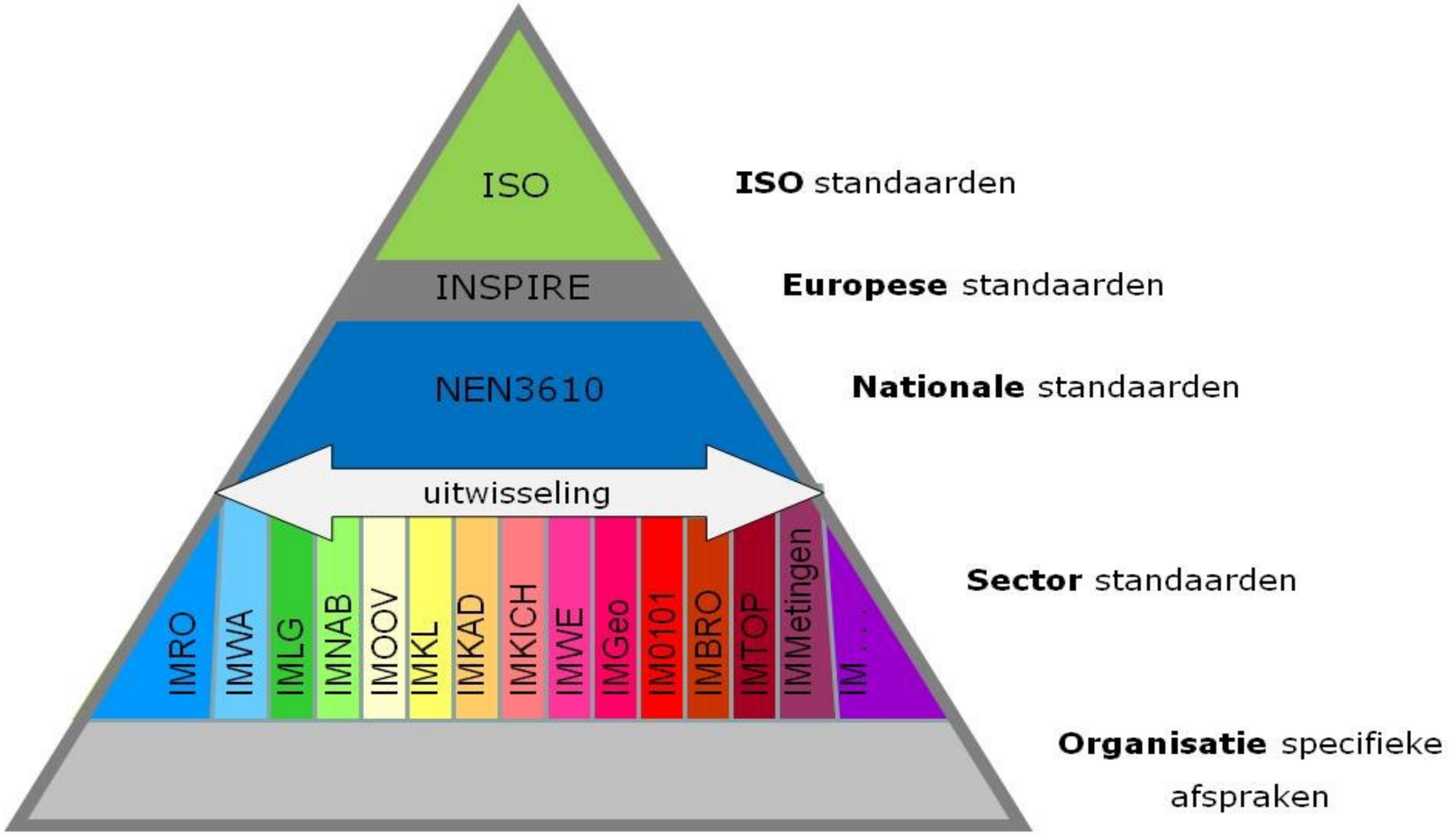
6709 - Standard representation of latitude, longitude and altitude for geographic point locations	19122 - Qualifications and Certification of personnel
19101 - Reference model	19123 - Schema for coverage geometry and functions
19101-2 - Reference model - Part 2: Imagery	19124 - Imagery and gridded data 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®

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

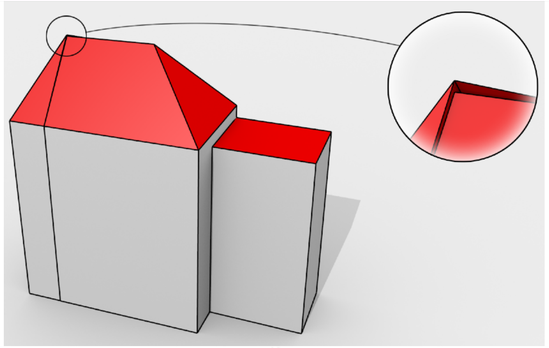
The standards from a Dutch point-of-view

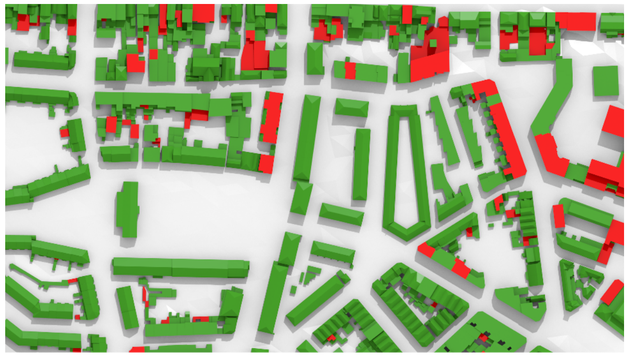


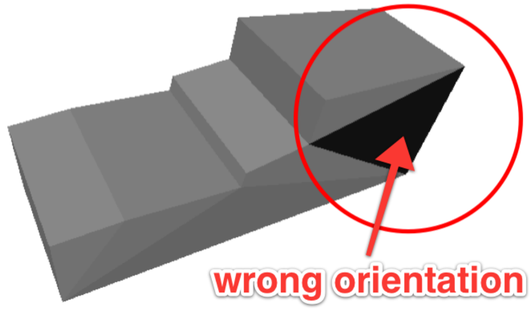
Do current 3D city models often contain these geometric errors?

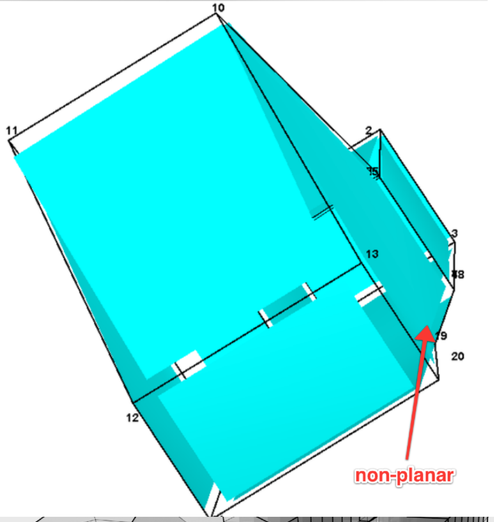
tl;dl: **YES.**

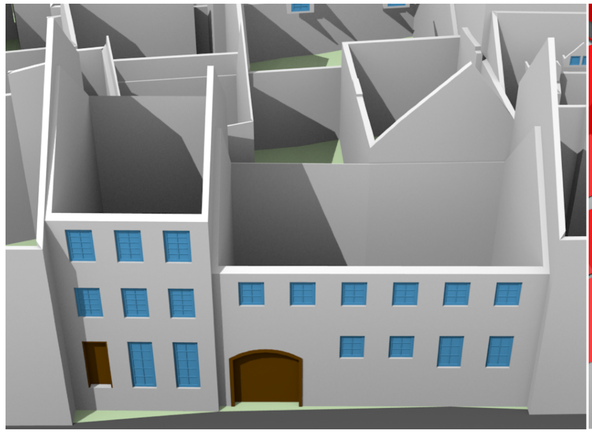
Errors are *very* common in 3D models: the wall of shame

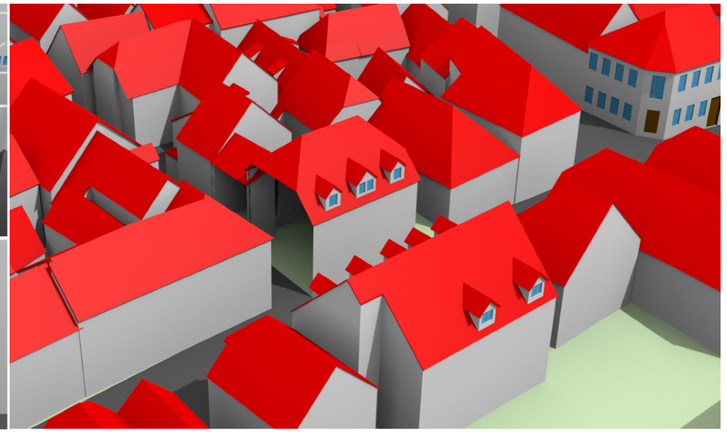


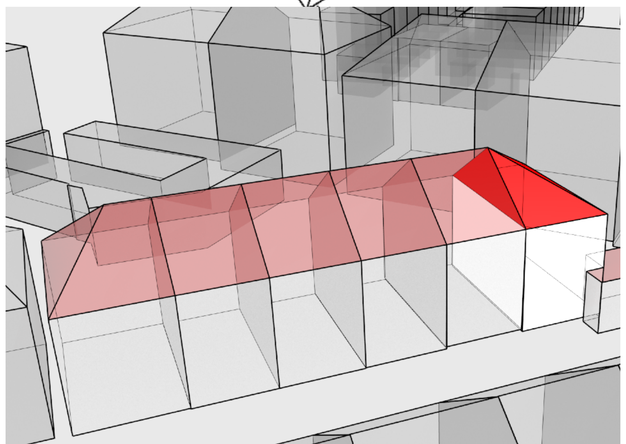


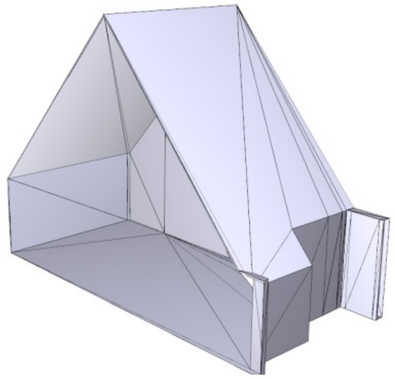


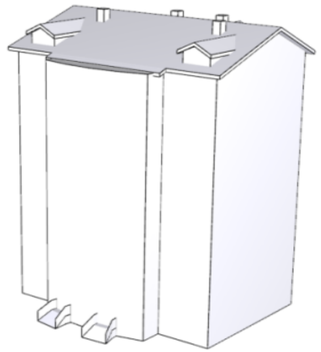


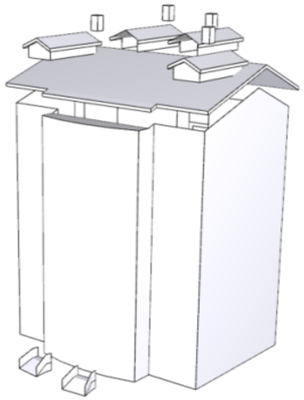










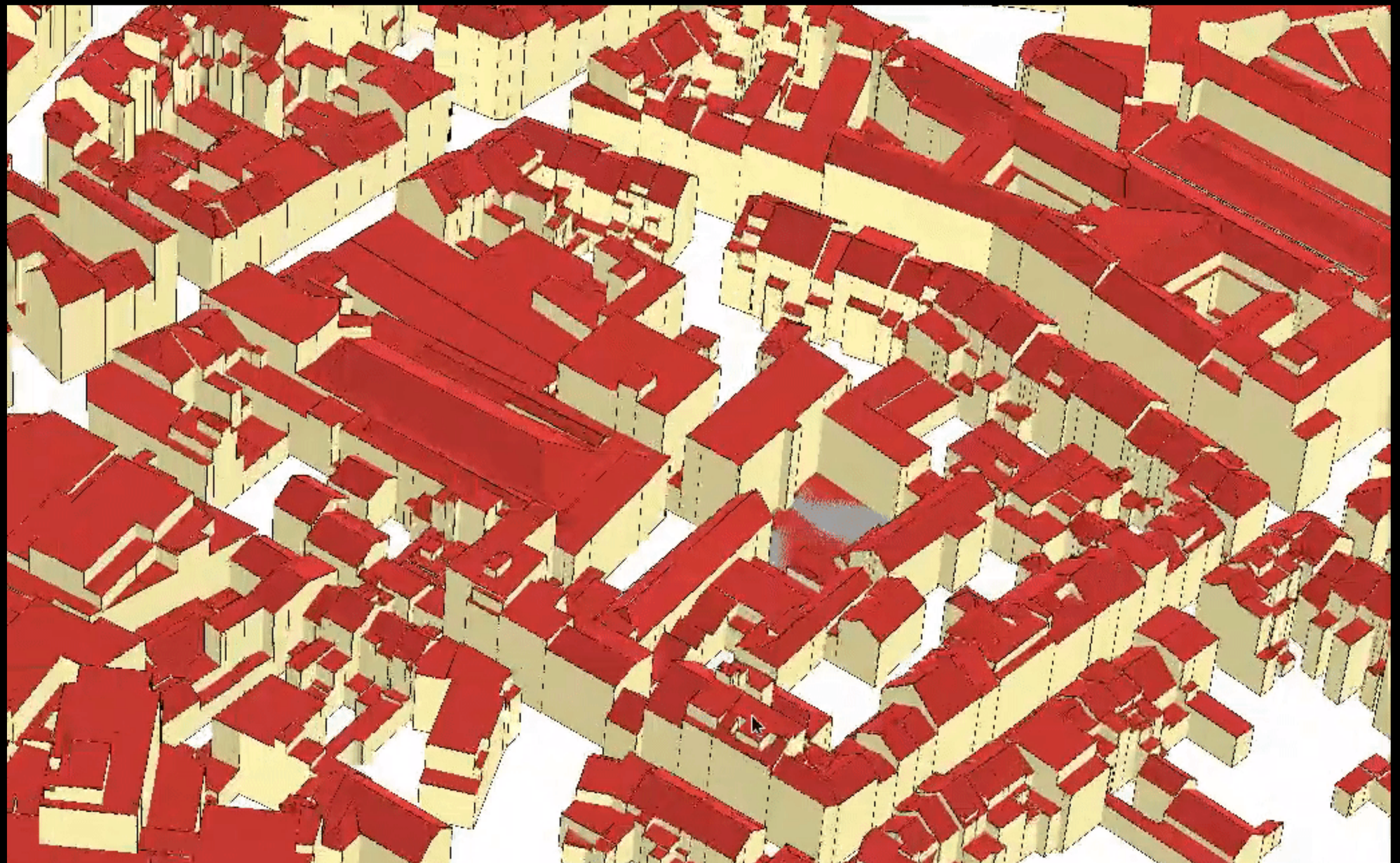


Should we care?

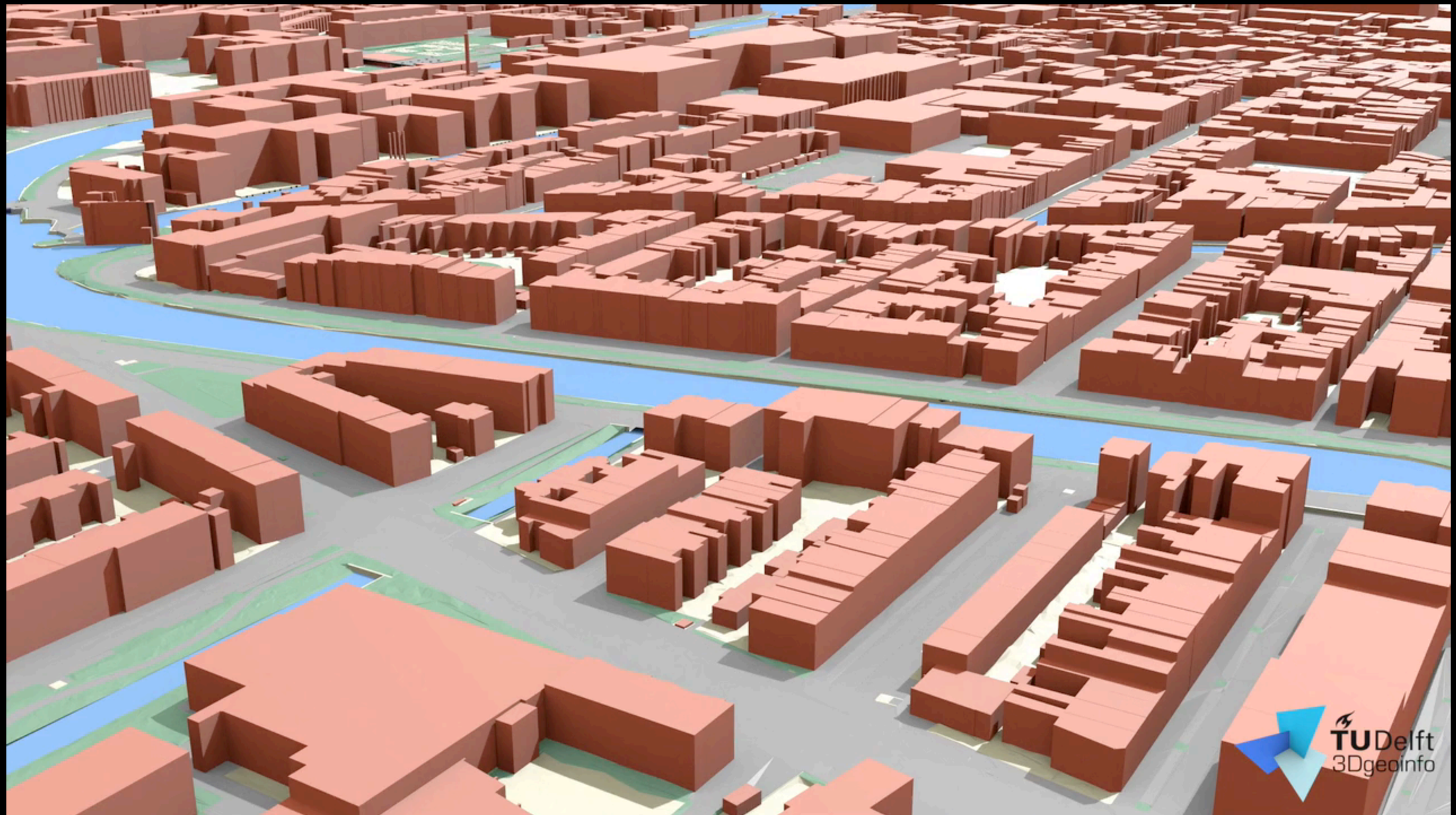
YES!

(especially if you want to pass this course 😅)

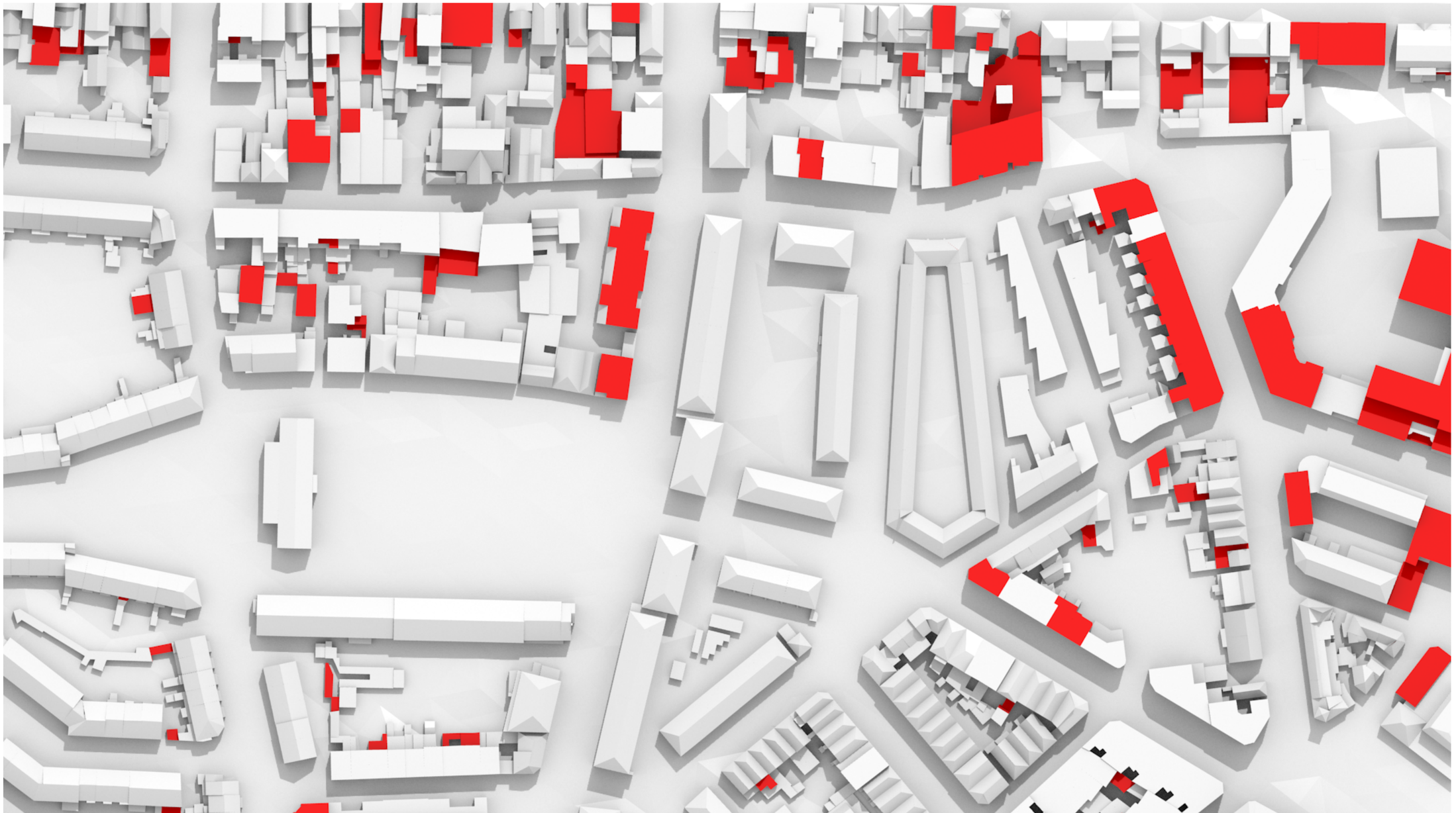
Visualisation — duplicated surfaces == annoying



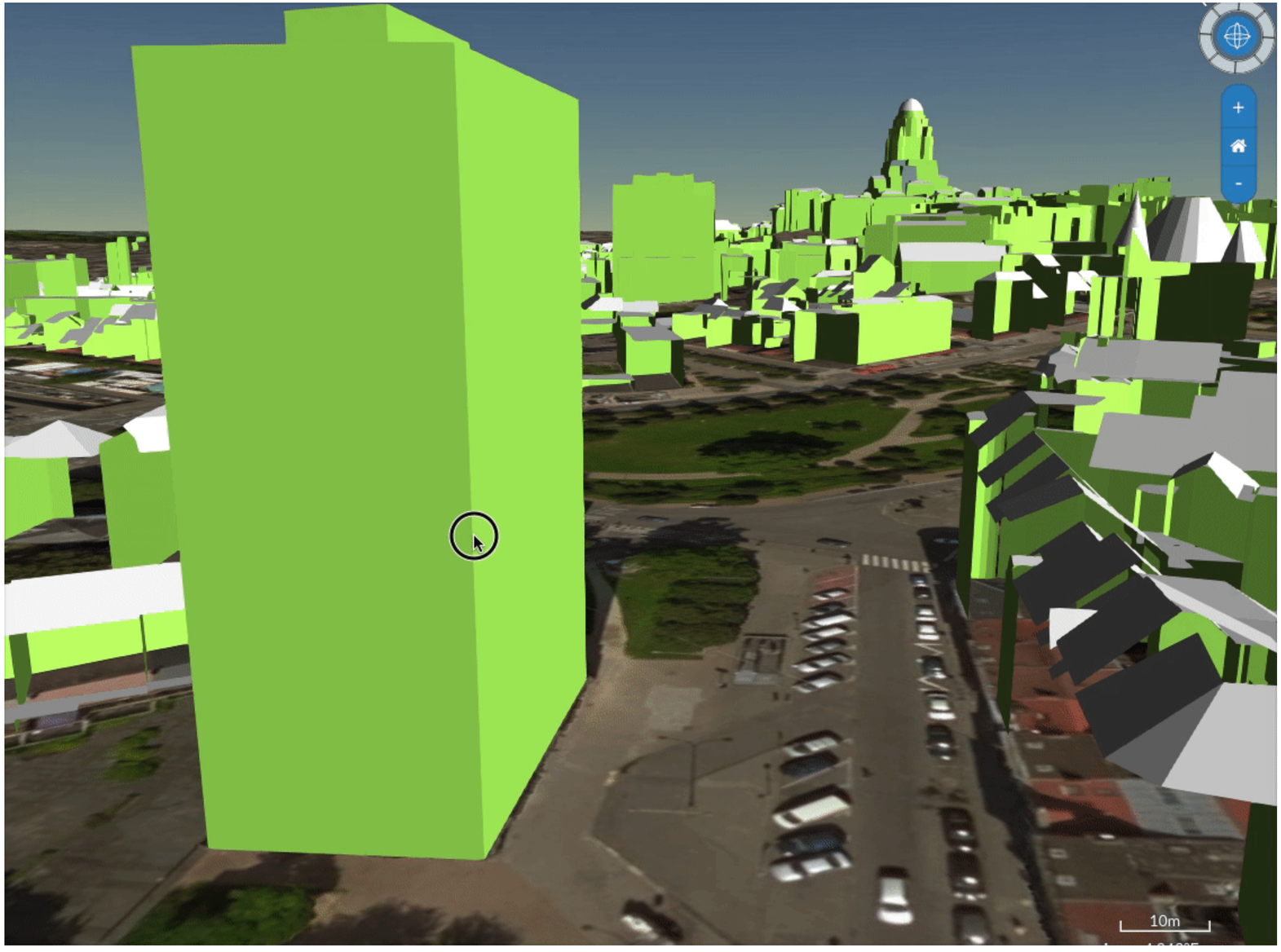
Visualisation — duplicated surfaces == distracting



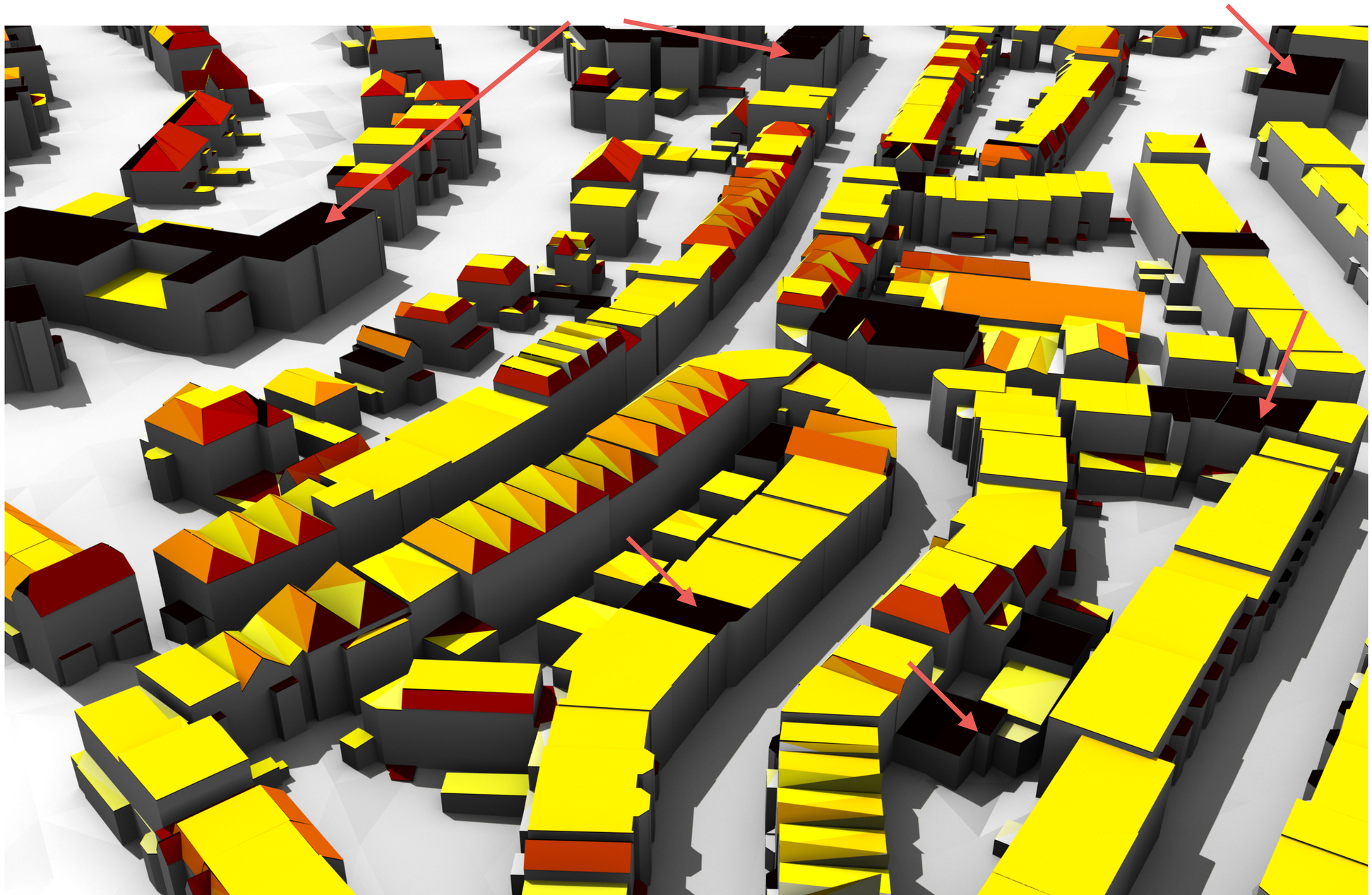
Visualisation – wrong orientation of surfaces (red ones)



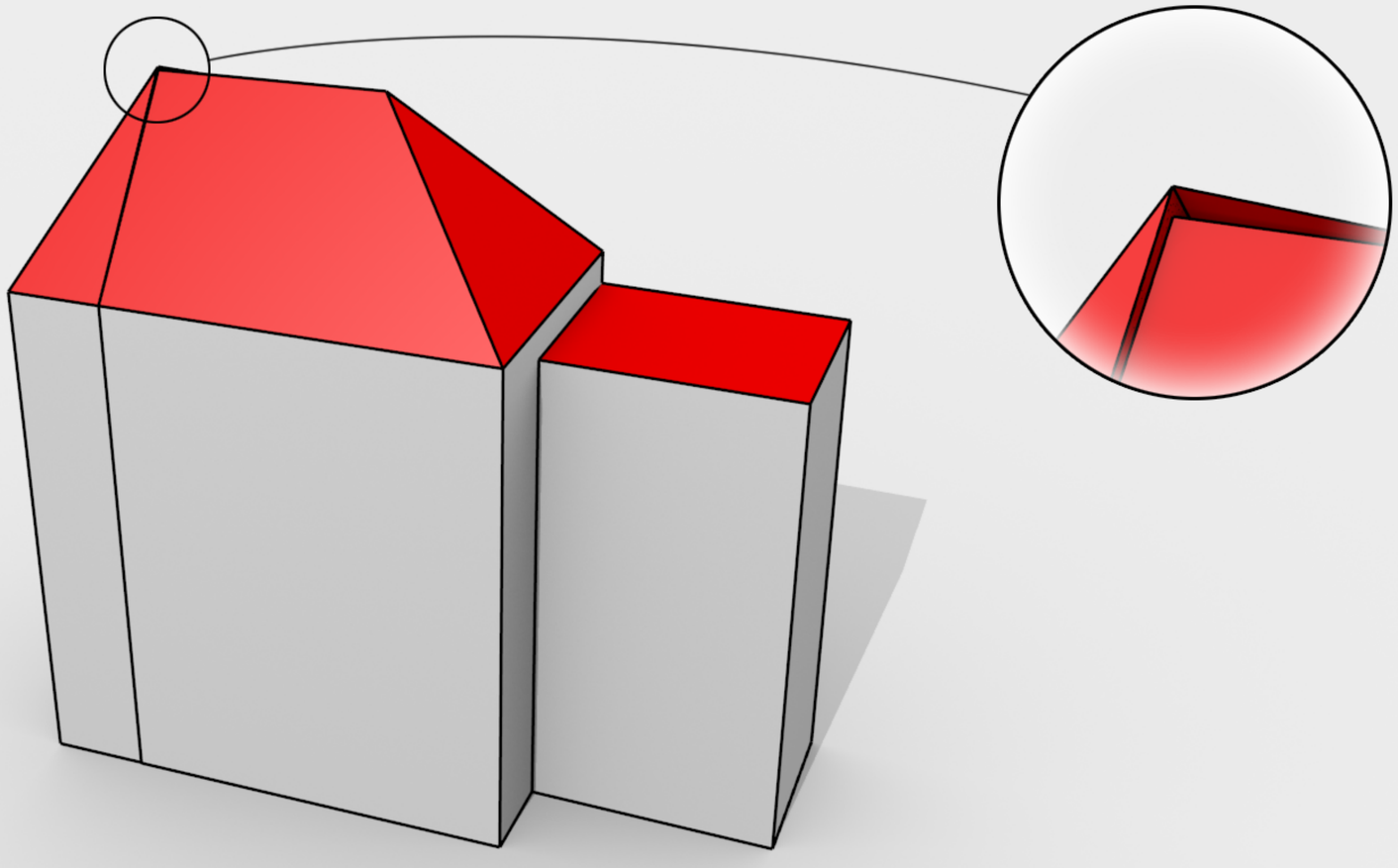
Visualisation — wrong orientation == missing faces



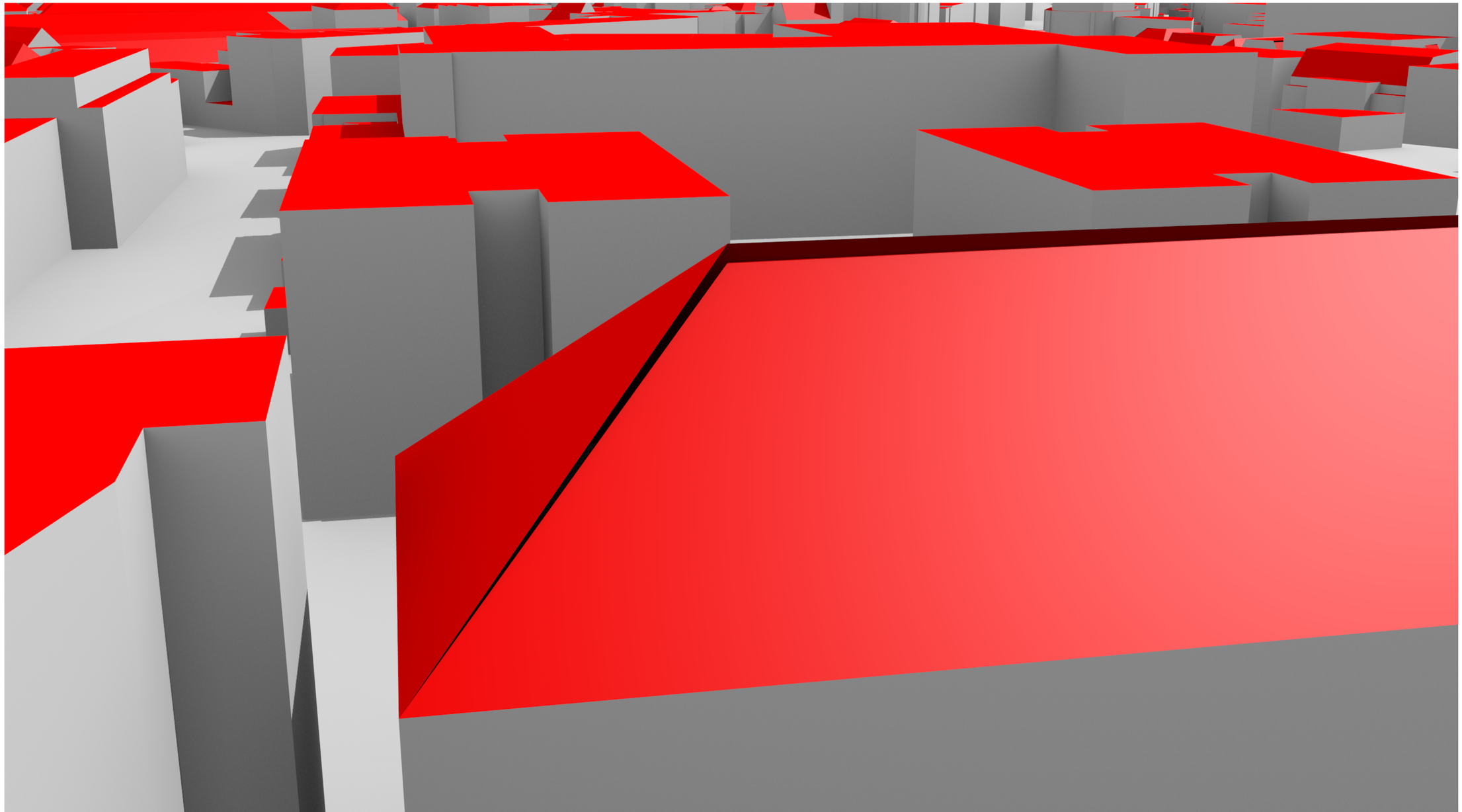
Solar potential — wrong orientation == no potential assigned



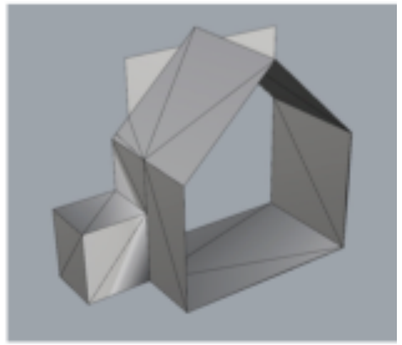
Volume calculation — tiny problems == impossible to calculate

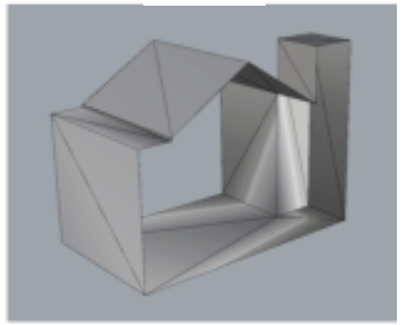


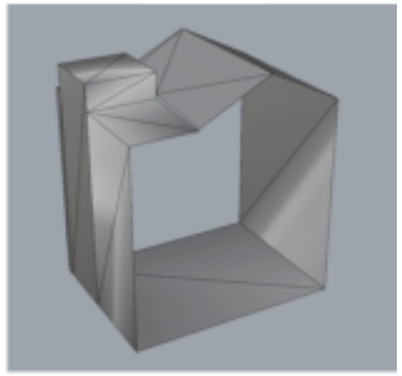
Volume calculation — tiny problems == impossible to calculate

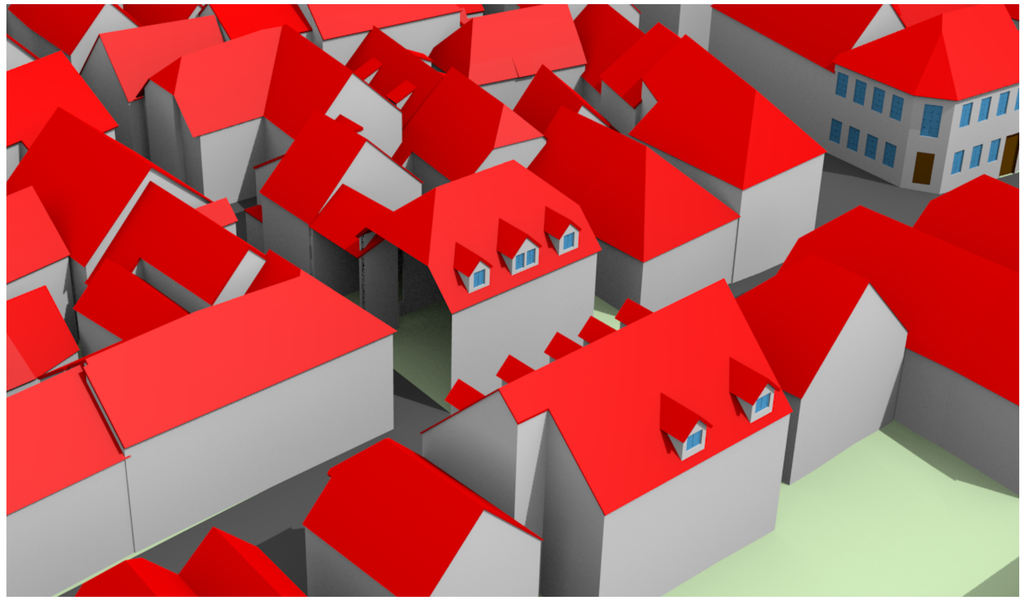


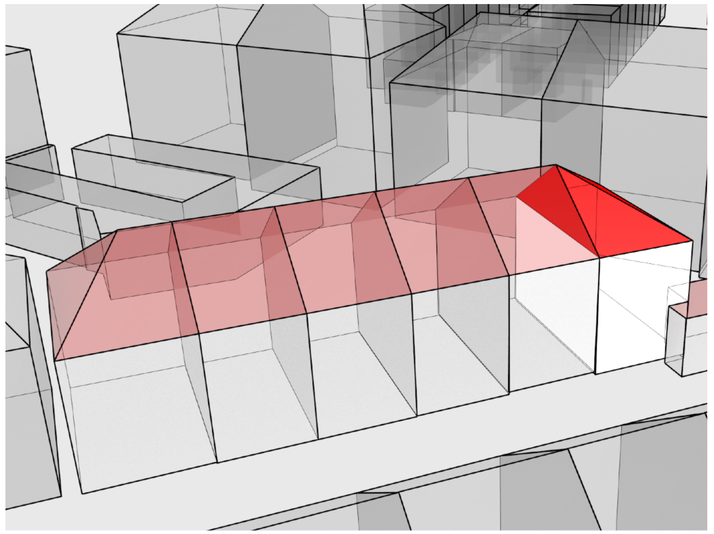
Volume calculation — big problems == also impossible to calculate...



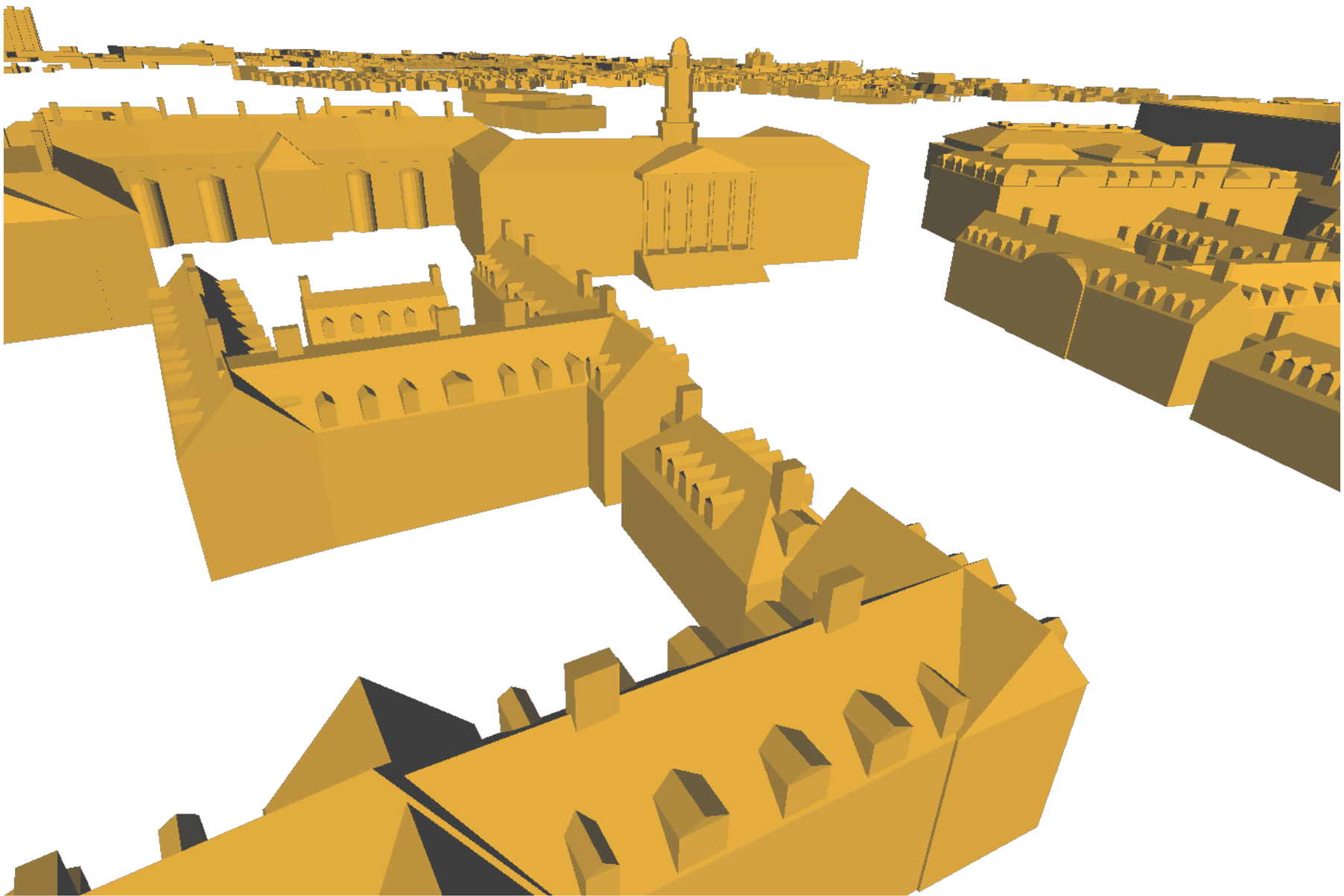




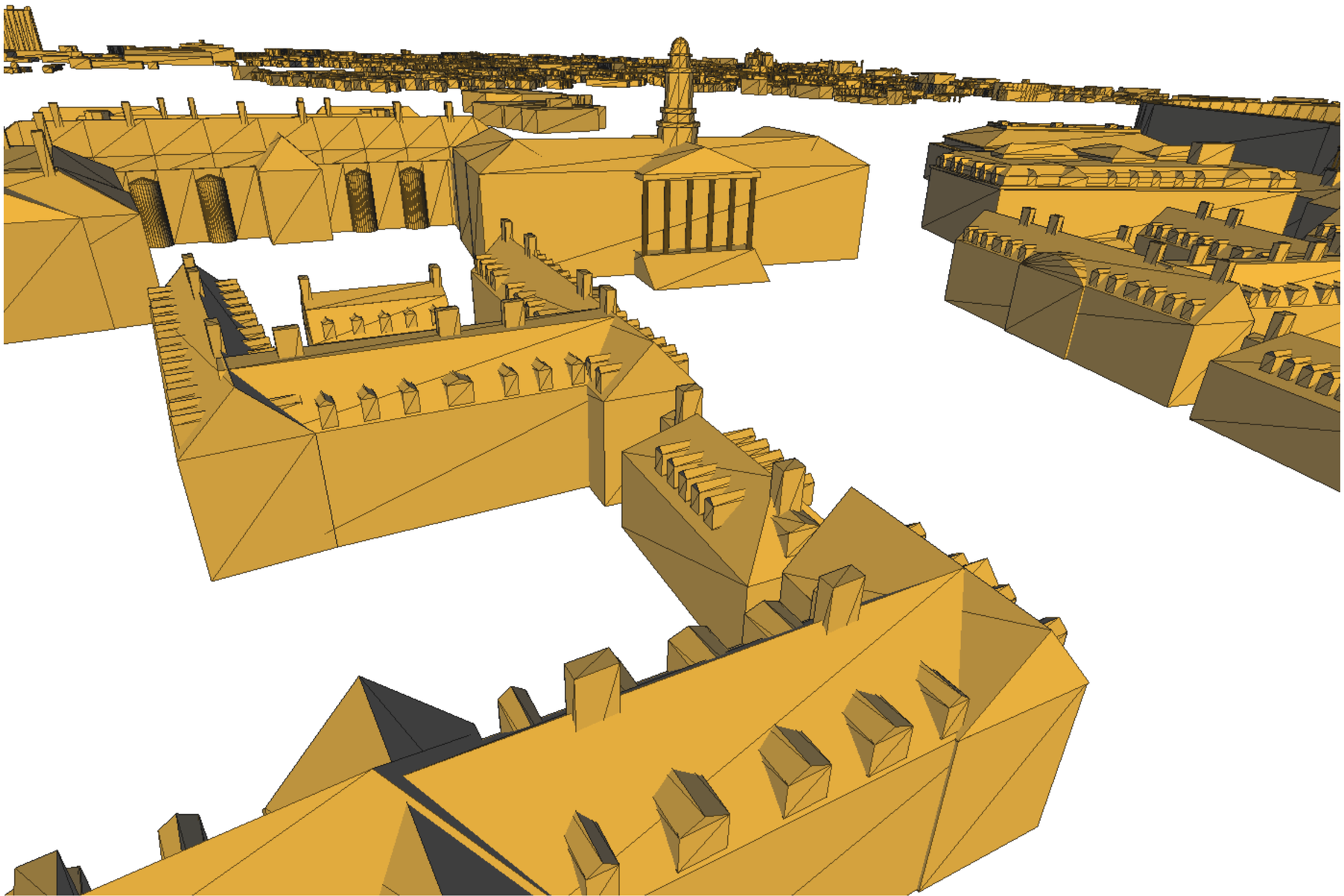




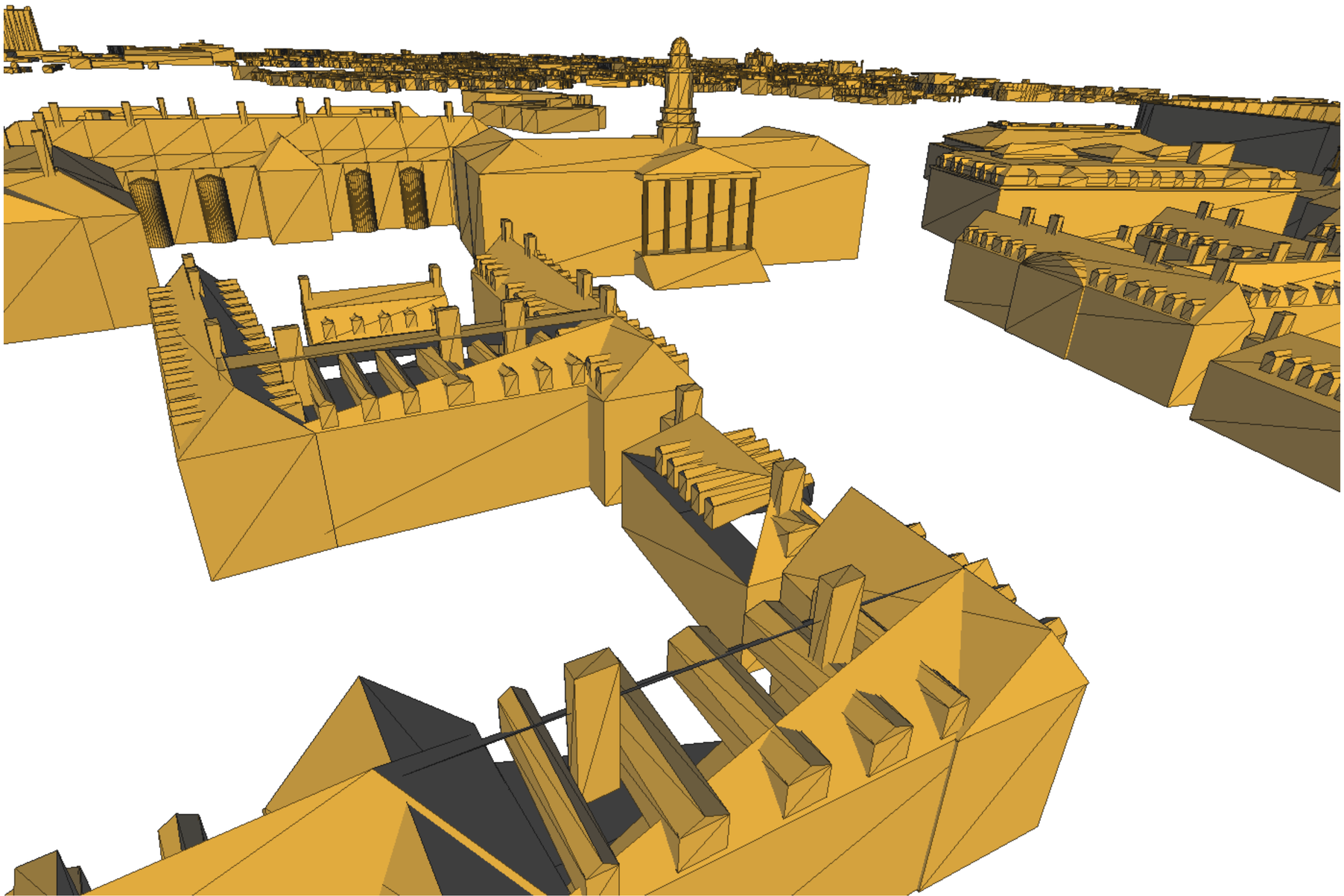
Volume calculation — superstructures not topologically connected



Volume calculation — superstructures not topologically connected



Volume calculation — superstructures not topologically connected



Advanced simulations — wind comfort for pedestrians

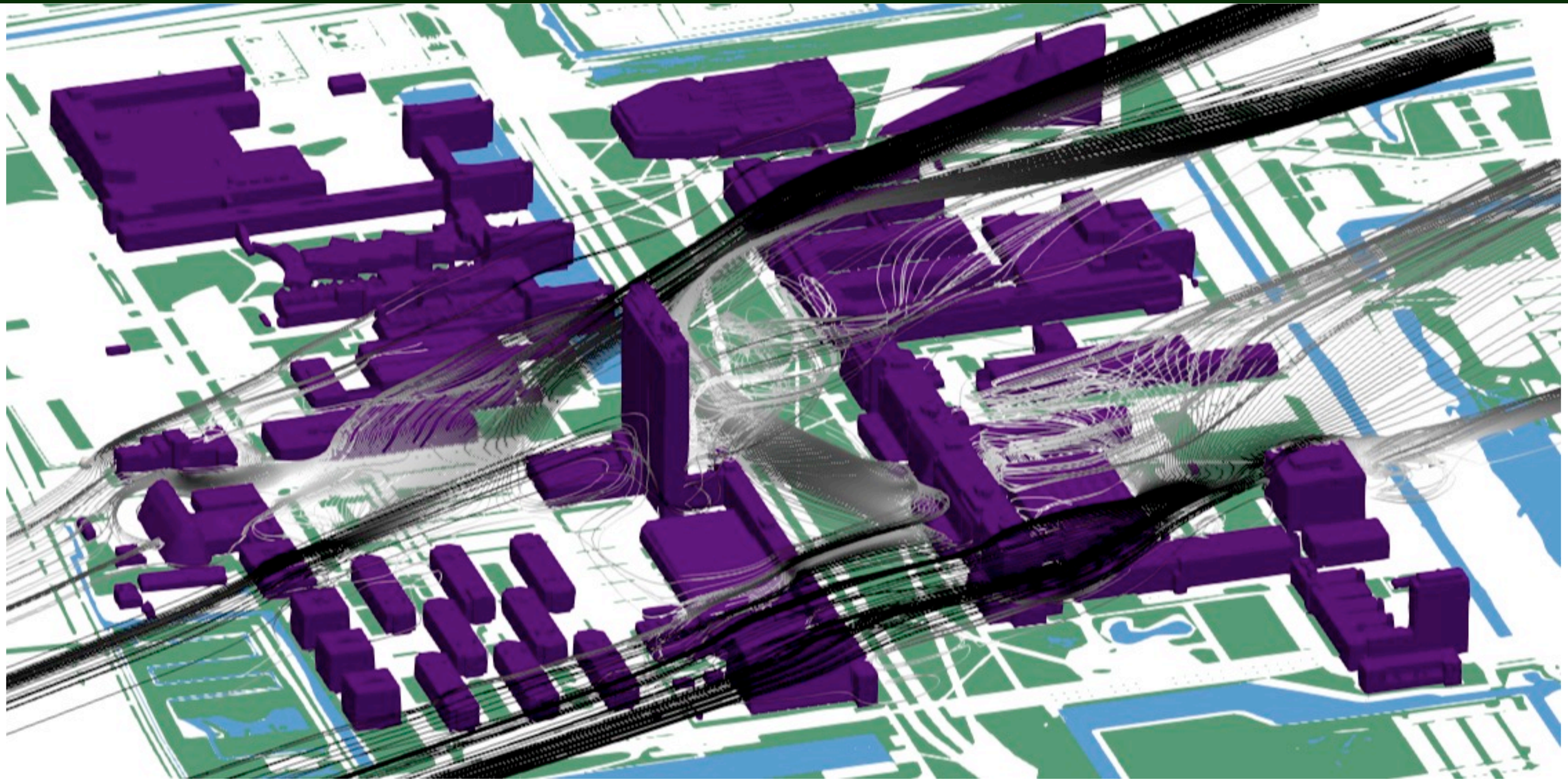
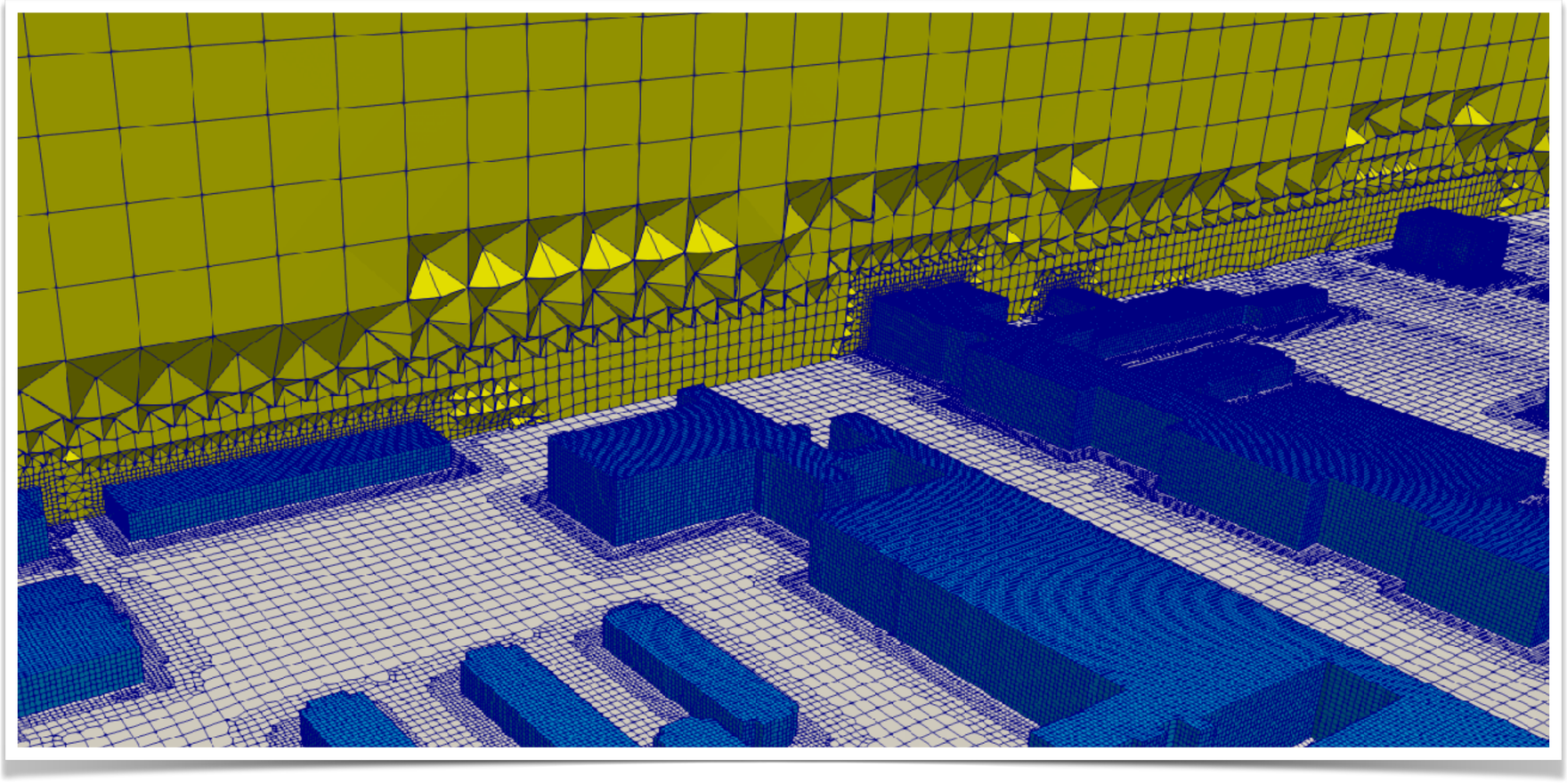


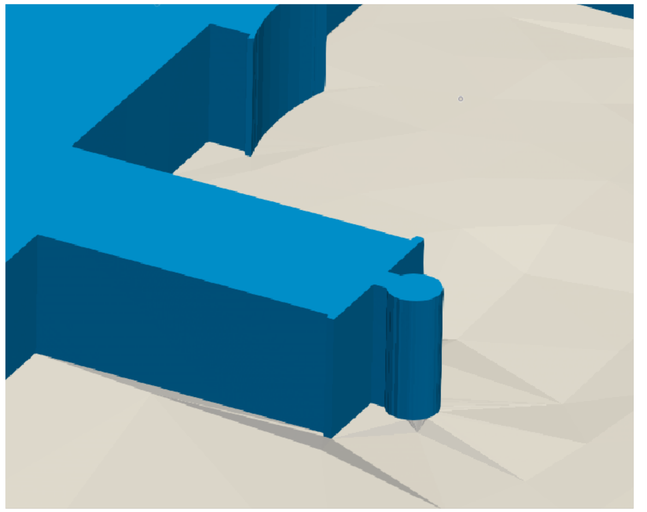
Figure from https://3d.bk.tudelft.nl/gsclara

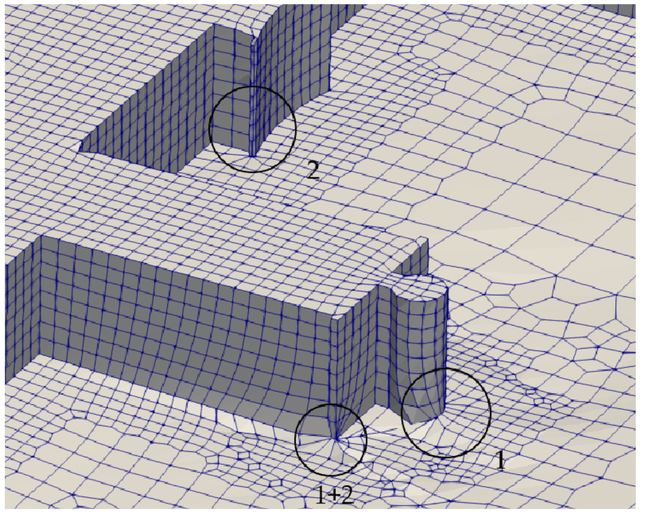
Advanced simulations — CFD methods: subdivision of volume (mesh)

computational fluid dynamics

GEO5015: Modelling wind and dispersion in urban environments (Q4!)







Advanced simulations — very strict input requirements

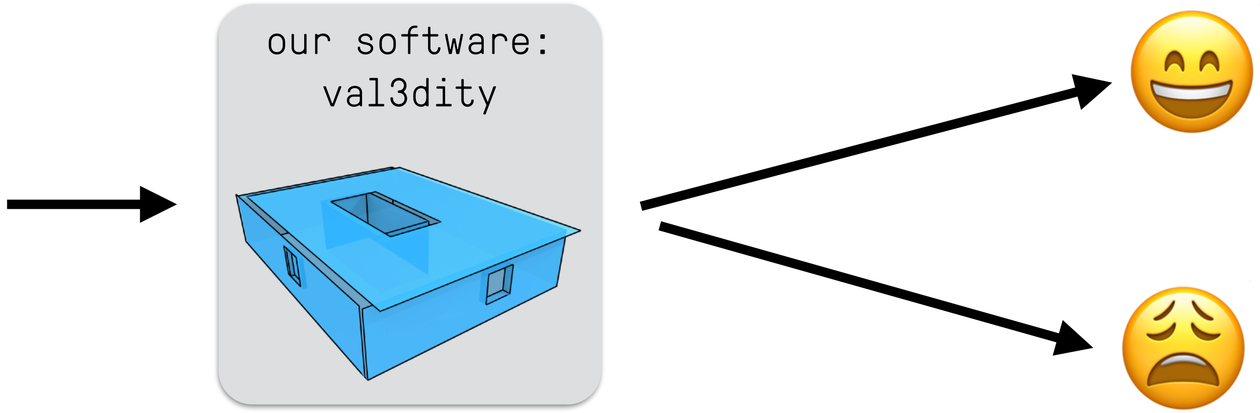
- no holes
- no intersections
- triangles must have certain shape (no *slivers*)
- a “perfect” model is thus required
- in practice, that still means several hours of (semi-)manual repair

Figure from CADFEM.de

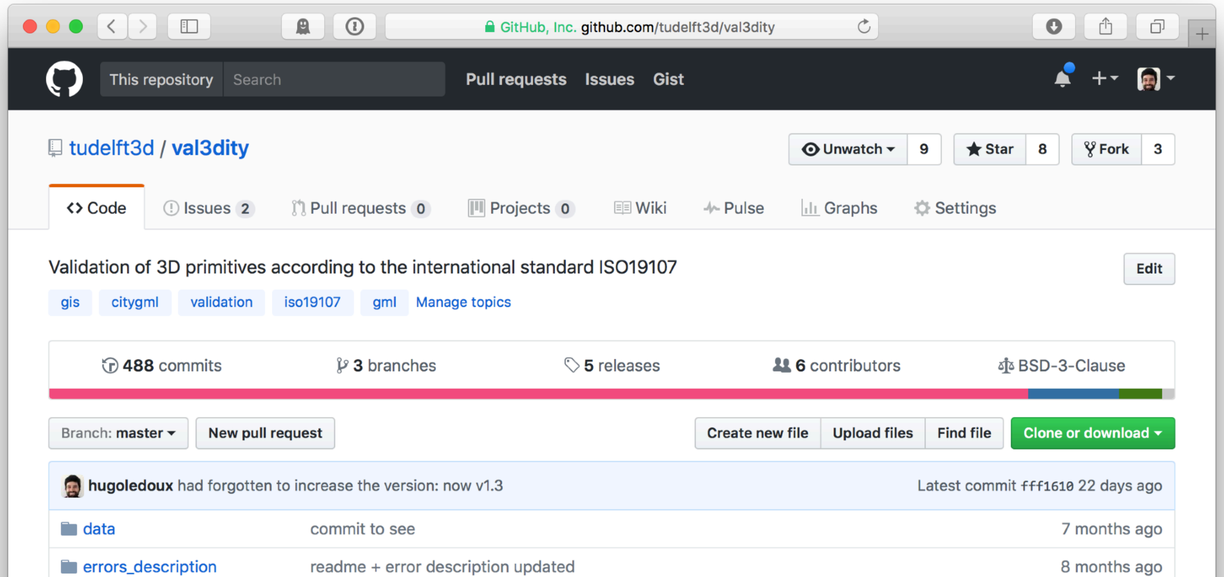
Our methodology

37 CityGML datasets from 9 countries

40 million polygons in 3.6 million buildings (Solids + MultiSurfaces)

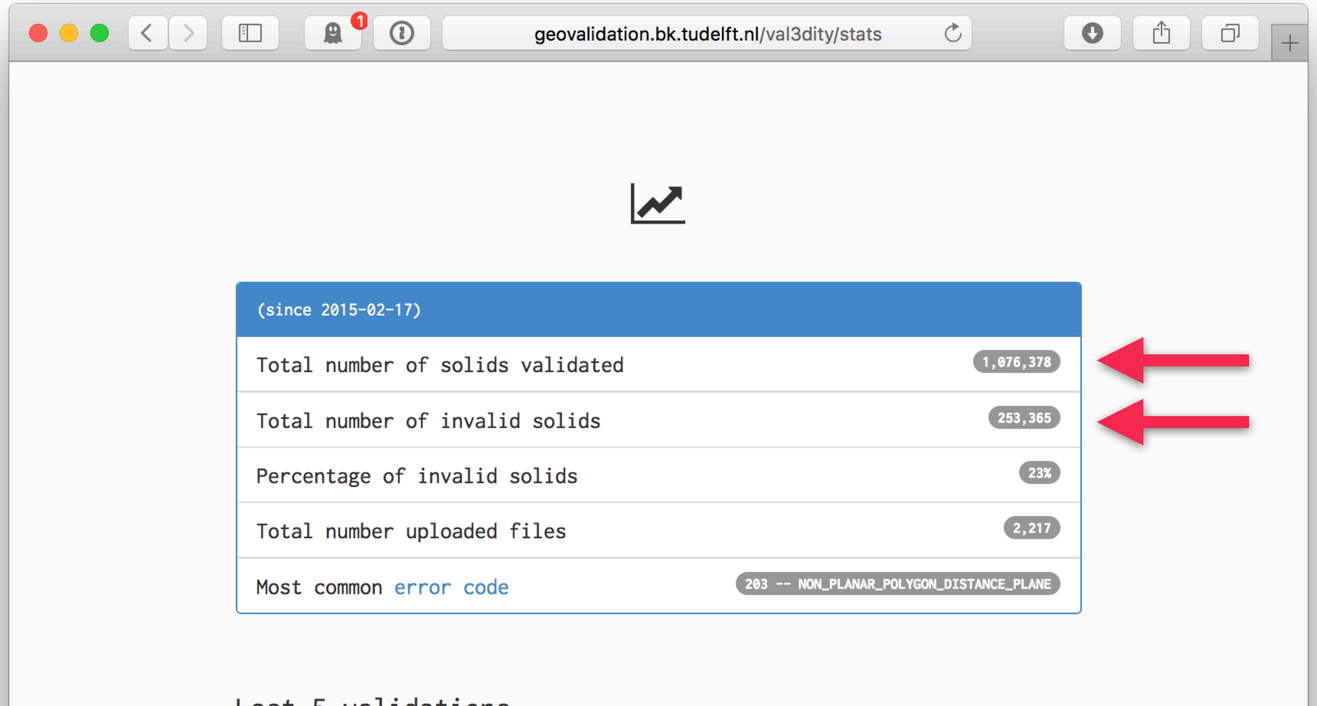


- 37 datasets (3D cities) from 9 countries
- 40 million polygons in 3.6 million buildings



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

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	–	✗
		2	0	0	0	0	0	–	✓
		3	0	0	0	0	0	–	✓
		4	0	0	0	0	0	–	✓
		5	0	0	0	0	0	–	✗
		6	0	0	0	0	0	–	✗
		7	0	0	0	0	0	–	✓
		8	0	0	0	0	0	–	✓
		9	0	0	0	0	0	–	✗
LOD2	`MultiSurface`	10	1	4	–	–	5	[e]	✓
		11	0	0	–	–	0	0	✓
		12	2	21	–	–	23	45	✓
		13	10	2	–	–	12	4	✗
		14	0	1	–	–	1	12	✓
		15	0	9	–	–	9	2	✓
		16	4	8	–	–	12	1	✗
		17	5	0	–	–	5	5	✗
		18	0	0	–	–	0	4	✗
		19	0	0	–	–	0	1	✗
		20	0	4	–	–	4	6	✗
		21	0	1	–	–	1	3	✗
LOD2	`Solid`	22	0	42	58	0	100	–	✓
		23[c]	–	–	–	–	–	–	✗
		24	0	31	1	3	35	–	✗
		25	4	0	16	2	22	–	✓
		26[c]	–	–	–	–	–	–	✗
		27	22	17	50	0	89	–	✓
LOD2	`MultiSurface`	28	0	42	1	1	44	0	✓
	and `Solid`[d]	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	✗
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	–	✗
		2	0	0	0	0	0	–	✓
		3	0	0	0	0	0	–	✓
		4	0	0	0	0	0	–	✓
		5	0	0	0	0	0	–	✗
		6	0	0	0	0	0	–	✗
		7	0	0	0	0	0	–	✓
		8	0	0	0	0	0	–	✓
		9	0	0	0	0	0	–	✗
LOD2	`MultiSurface`	10	1	4	–	–	5	[e]	✓
		11	0	0	–	–	0	0	✓
		12	2	21	–	–	23	45	✓
		13	10	2	–	–	12	4	✗
		14	0	1	–	–	1	12	✓
		15	0	9	–	–	9	2	✓
		16	4	8	–	–	12	1	✗
		17	5	0	–	–	5	5	✗
		18	0	0	–	–	0	4	✗
		19	0	0	–	–	0	1	✗
		20	0	4	–	–	4	6	✗
		21	0	1	–	–	1	3	✗
LOD2	`Solid`	22	0	42	58	0	100	–	✓
		23[c]	–	–	–	–	–	–	✗
		24	0	31	1	3	35	–	✗
		25	4	0	16	2	22	–	✓
		26[c]	–	–	–	–	–	–	✗
		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	✗
LOD4	`Solid`	37	0	0	3	0	3	68	✓

% of invalid

rather good, but it's "easy"

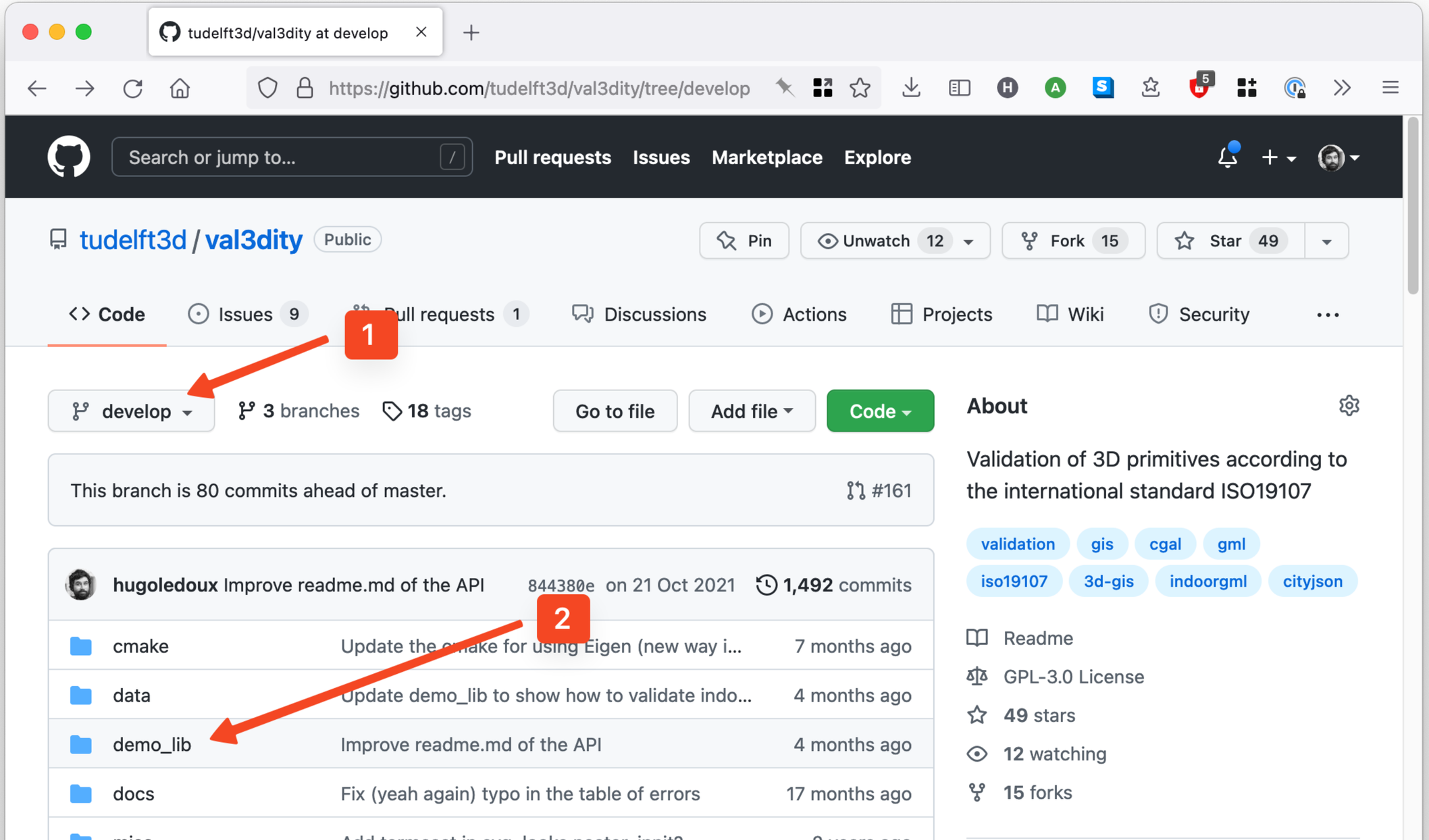
Level of detail	Primitive	ID[a]	Geometric validation					Semantics[f]	Schema
			1xx	2xx	3xx	4xx	Total[b]		
LOD1	`Solid`	1	0	0	0	0	0	–	✗
		2	0	0	0	0	0	–	✓
		3	0	0	0	0	0	–	✓
		4	0	0	0	0	0	–	✓
		5	0	0	0	0	0	–	✗
		6	0	0	0	0	0	–	✗
		7	0	0	0	0	0	–	✓
		8	0	0	0	0	0	–	✓
		9	0	0	0	0	0	–	✗
LOD2	`MultiSurface`	10	1	4	–	–	5	[e]	✓
		11	0	0	–	–	0	0	✓
		12	2	21	–	–	23	45	✓
		13	10	2	–	–	12	4	✗
		14	0	1	–	–	1	12	✓
		15	0	9	–	–	9	2	✓
		16	4	8	–	–	12	1	✗
		17	5	0	–	–	5	5	✗
		18	0	0	–	–	0	4	✗
		19	0	0	–	–	0	1	✗
		20	0	4	–	–	4	6	✗
		21	0	1	–	–	1	3	✗
LOD2	`Solid`	22	0	42	58	0	100	–	✓
		23[c]	–	–	–	–	–	–	✗
		24	0	31	1	3	35	–	✗
		25	4	0	16	2	22	–	✓
		26[c]	–	–	–	–	–	–	✗
		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	✗
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++



Potential exam question

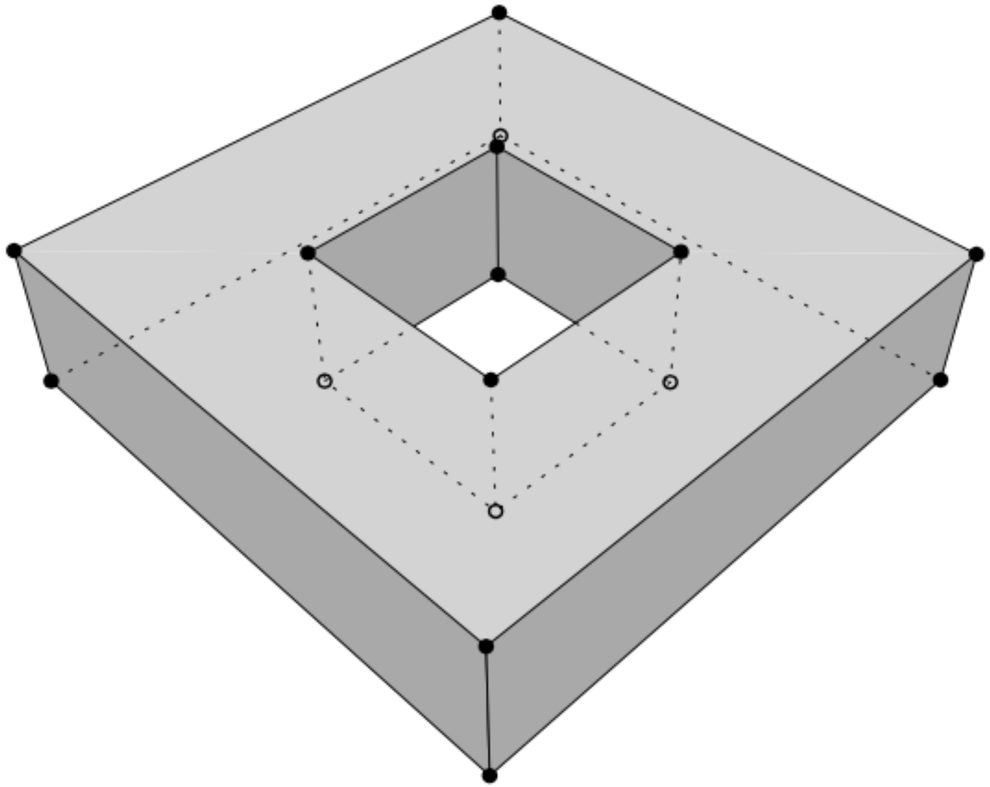


Figure 9.4: A 'squared torus' is modelled with one exterior boundary formed of ten surfaces.

1. How many interior boundaries (shells) does this torus have?
2. List all surfaces for it: how many are there?
3. If I triangulate its surfaces, how many surfaces?