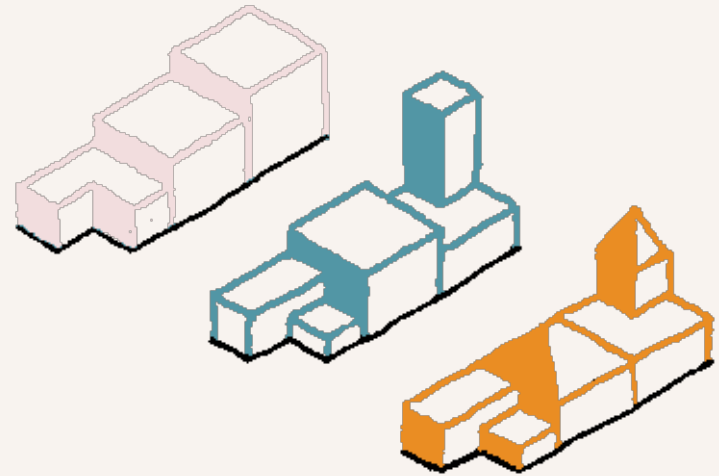


Open 3D workflows

Standards, software, and collaboration for
maintainable 3D GIS





01 / Framing

Intro

Open, modular ecosystem

3DGI

- Small company, 2 engineers
- Open-source software for 3D city model workflows
- CityJSON
- 3DBAG
- Roofer, Tyler, Roofer Online

Open, modular tools enable transparency and control.

Why openness matters

Published 3D data is often treated as trusted public information.

Control over the workflow:

- Can we review it?
- What assumptions are inside?
- Can the workflow be rerun, checked, or adapted?
- Can tools or suppliers be replaced?
- What are limitation in the produced data?

Openness supports ownership in data, knowledge and process.

02 / CityJSON

Representation

CityJSON: open and pragmatic

- Representation affects the whole workflow
- CityJSON as technical anchor
- Open, practical, software/developer-friendly
- Lower barrier for tool development
- Pragmatic format for 3D city models



```
{  
  "type": "CityJSON",  
  "version": "2.0",  
  "extensions": {},  
  "transform": {  
    "scale": [1.0, 1.0, 1.0],  
    "translate": [0.0, 0.0, 0.0]  
  },  
  "metadata": {},  
  "CityObjects": {},  
  "vertices": [],  
  "appearance": {},  
  "geometry-templates": {}  
}
```

Efficiency

Three types of efficiency:

- **Computational:** read, write, transform, analyse
- **Development:** build reliable tools faster
- **Cognitive:** understand and debug easily

CityJSON is a useful middle ground - not optimal for every task, but effective as a workflow foundation.

Foundational libraries:  [3DGI/cityjson-rs](https://github.com/3DGI/cityjson-rs)

CityJSON variants

Same data model, different processing targets.

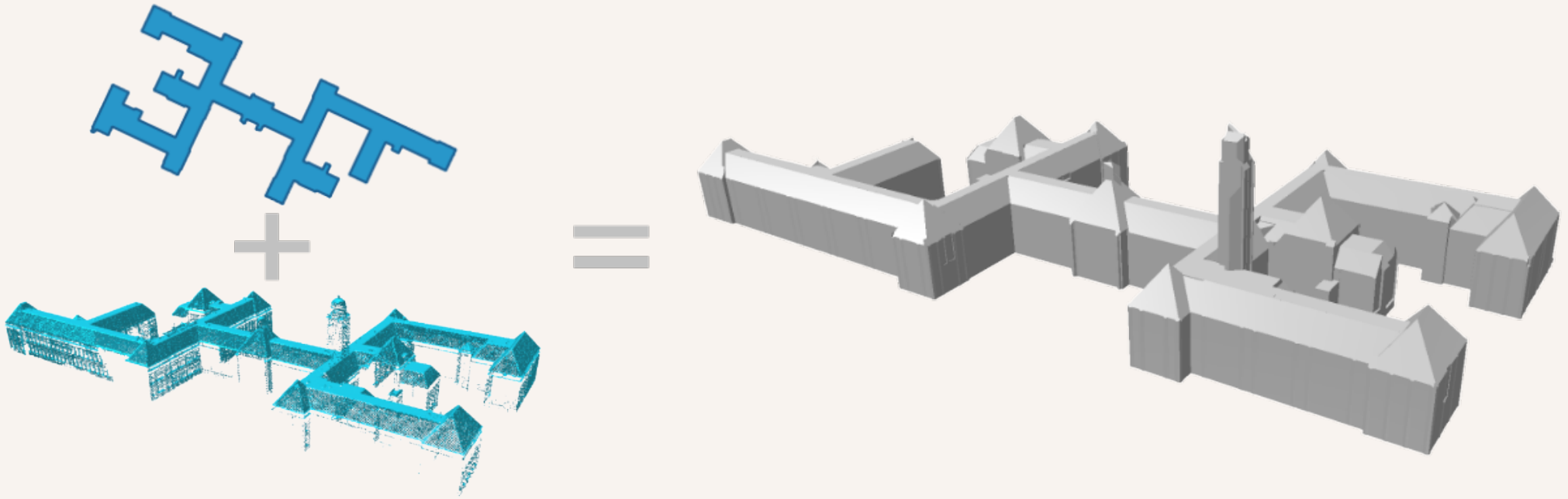
- *CityJSON*: regular JSON
- *CityJSONSeq*: newline-delimited, streaming
- *FlatCityBuf*: FlatBuffers-based binary alternative, cloud-native
 -  <https://www.cityjson.org/flatcitybuf>
- *CityArrow* / *CityParquet*: experimental columnar approaches, cloud-native
 -  <https://github.com/3DGI/cityjson-rs/tree/main/crates/cityjson-arrow>
 -  <https://github.com/3DGI/cityjson-rs/tree/main/crates/cityjson-parquet>

3DBAG production uses CityJSONSeq between processes.

03 / Roofer

Production

Roofer: automated building reconstruction

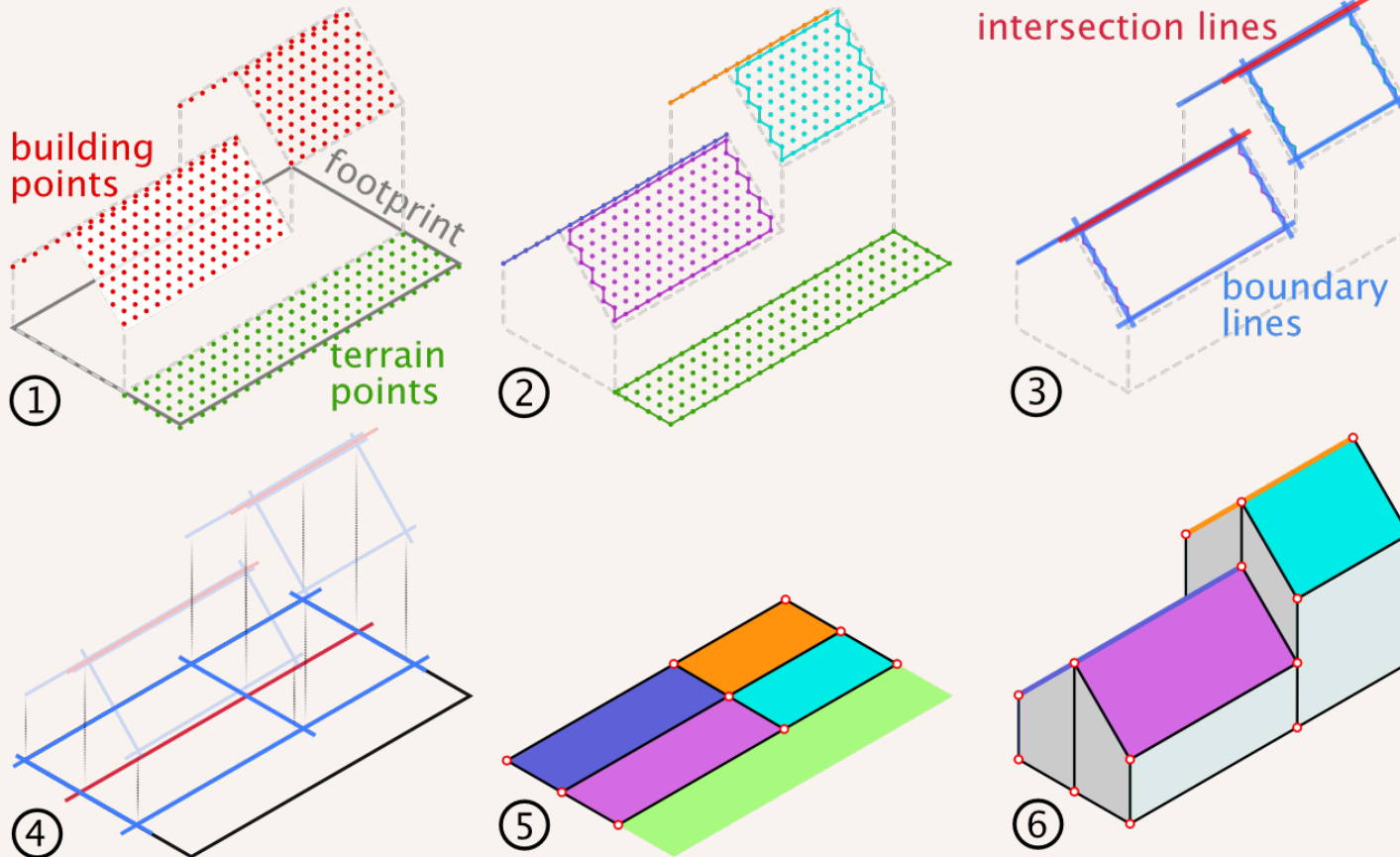


Roofer: automated building reconstruction

- **Input:**
 - point cloud: classified (building + terrain) LiDAR
 - building polygons
- **Output:**
 - semantic 3D building models
 - LoD 1.2 / 1.3 / 2.2
- Used for 3DBAG



Roofer: algorithm

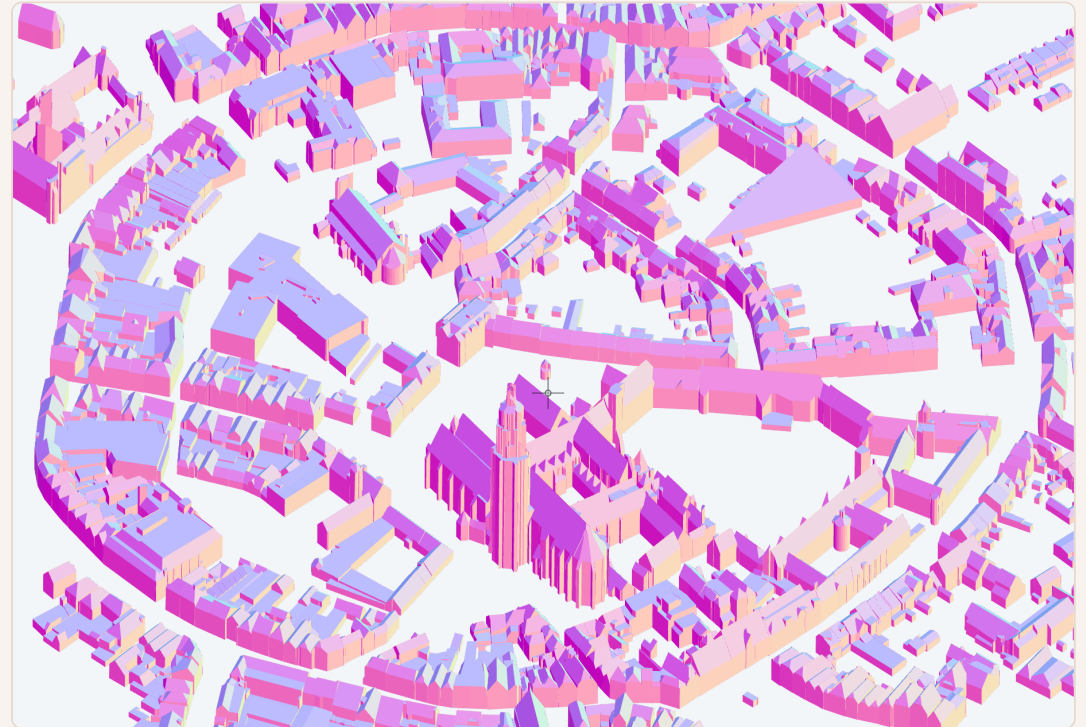


Roofer: from research to production

Open source helped Roofer mature.

- Started at TU Delft
- Continued and productionised by 3DGI
- Used internationally
- External review, issues, tests, and contributions

The production step behind published 3D data does not have to be a black box.

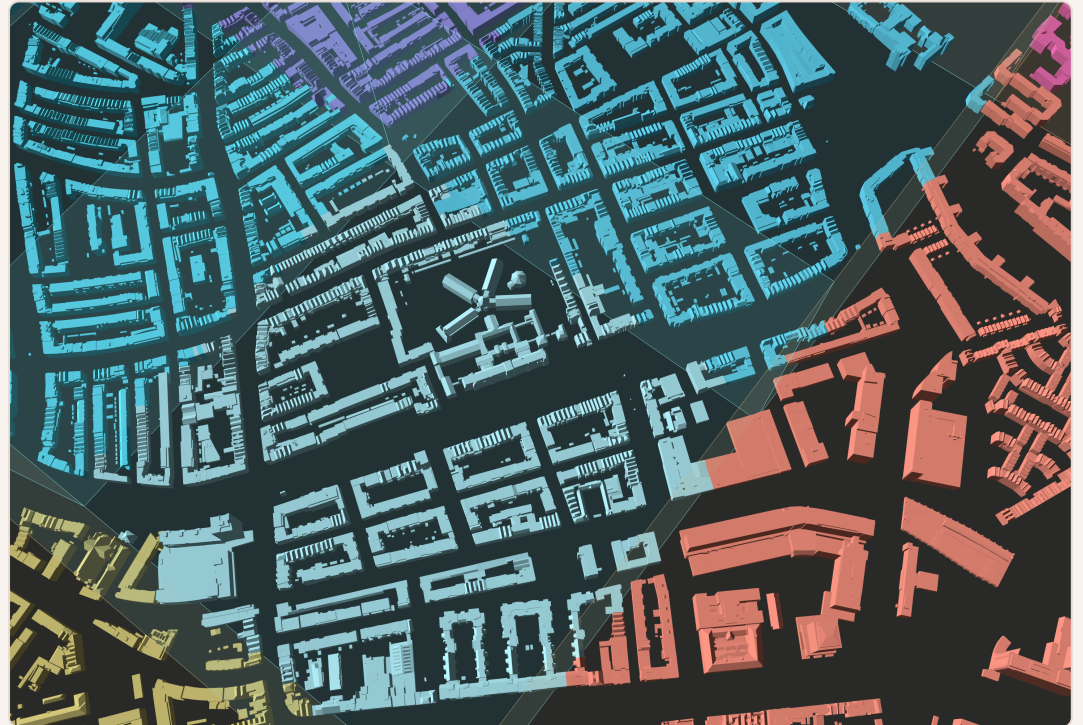


04 / Tyler

Publication

Tyler: from dataset to usable data

- Tiles large CityJSON datasets
- Converts to publication formats
- 3D Tiles v1.1: metadata + compression
- v1.0:
 - OBJ
 - CityJSON, CityJSONSeq
 - GPKG (planned)
 - CSV (planned)
- Built on top of our CityJSON foundational libraries in Rust



Tyler: tiling workflow

Bottom-up tiling:

1. Index CityObjects on a regular grid
2. Count vertices per grid cell
3. Merge cells into a quadtree
4. Read full CityObjects only for active tile
5. Convert tiles in parallel

Avoids loading the full input dataset into memory.



Tyler: open publication

Publication is part of the data product.

- Tiling strategy is explicit
- Attributes / LoD / colours / materials are configurable
- Output formats are reproducible
- Workflow can be rerun after updates

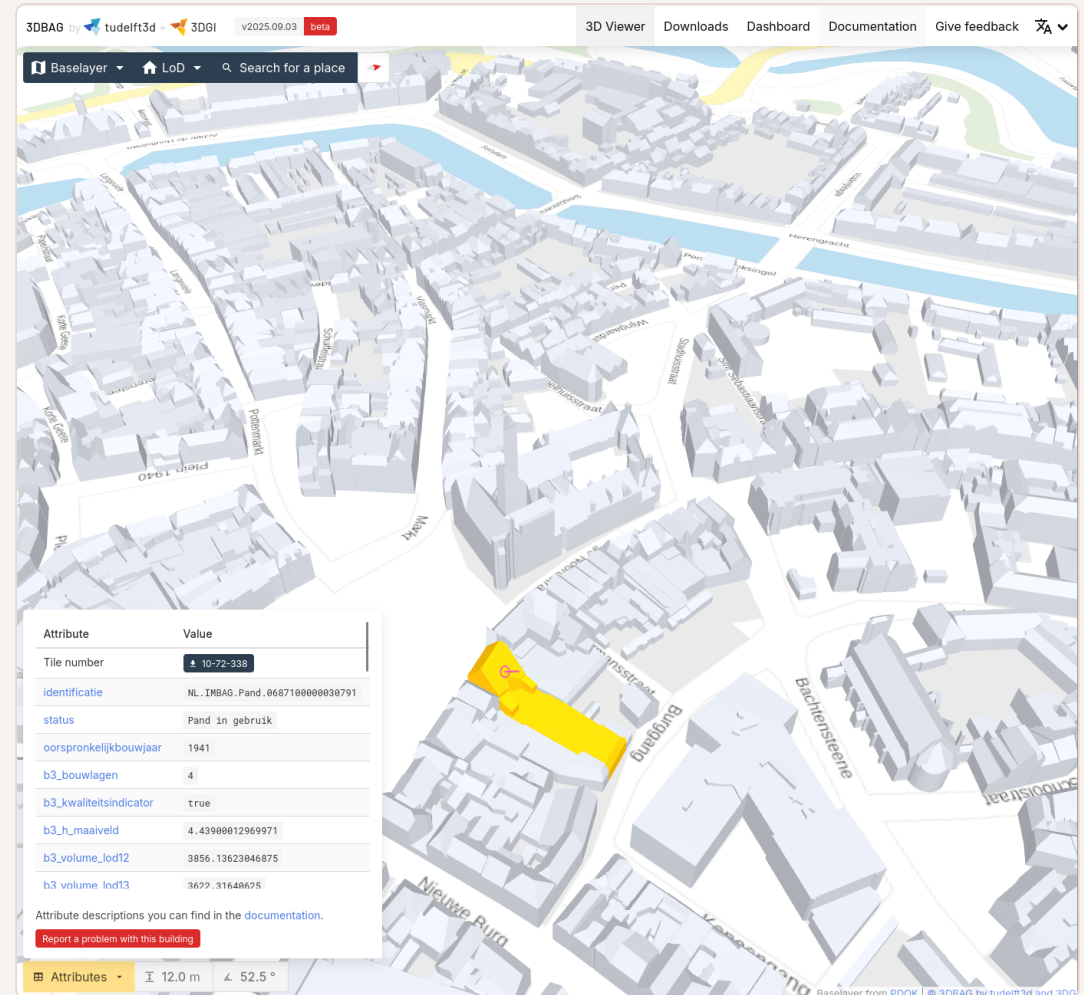
Roofer opens production. Tyler opens publication.

05 / 3DBAG + Roofer Online

Operational context

3DBAG

- Real-world context for the full workflow
- CityJSON for representation
- Roofer for production
- Tyler for publication
- Users need access for:
 - visualisation
 - analysis
 - research
 - planning



Roofer Online

- Web service around Roofer
- Built for Dutch municipalities
- User uploads point cloud
- Service uses Dutch BAG building polygons
- Roofer runs reconstruction

Lower entry barrier; open core remains inspectable.

 roofer-online.nl

Roofer Online

Puntenwolken

3D Reconstructies

3D Kaart

Beheer

Gebruikers

Domeinen Whitelist

Statistieken

Maandrapporten

Processen

Colofon

Help

Support

balazs.dukai@3dgi.nl

3D gebouw reconstructies

Dit is een overzicht van alle 3D gebouw reconstructies.

+ Nieuwe reconstructie

Naam	Peildatum BAG	Puntenwolken	Status	Acties
Nieuwe reconstructie 	2026-04-08	186500_433000.laz	 gereed	    
New 3D Model 	2026-01-08	nijmegen_2.laz	 gereed	    
Nieuwe reconstructie 	2026-01-08	nijmegen_2.laz	 gereed	    
New 3D Model 	2026-01-08	nijmegen_2.laz	 gereed	    
Nieuwe reconstructie 	2026-01-08	nijmegen_2.laz	 gereed	    
Nieuwe reconstructie 	2025-12-08	nijmegen_1.laz	 gereed	    
Nieuwe reconstructie 	2025-12-08	nijmegen_1.laz	 gereed	    
New 3D Model 	Onbekend	nijmegen_1.laz	 gereed	    

06 / Closing

Takeaway

Openness across the workflow

- Openness across the full 3D workflow
- Important for public organisations and NMCA's
- Transparency behind authoritative 3D data
- Understand dependencies
- Maintain the process over time

Open source is not the destination.

It is an enabler for 3D workflows that can be fully owned, maintained and adapted.