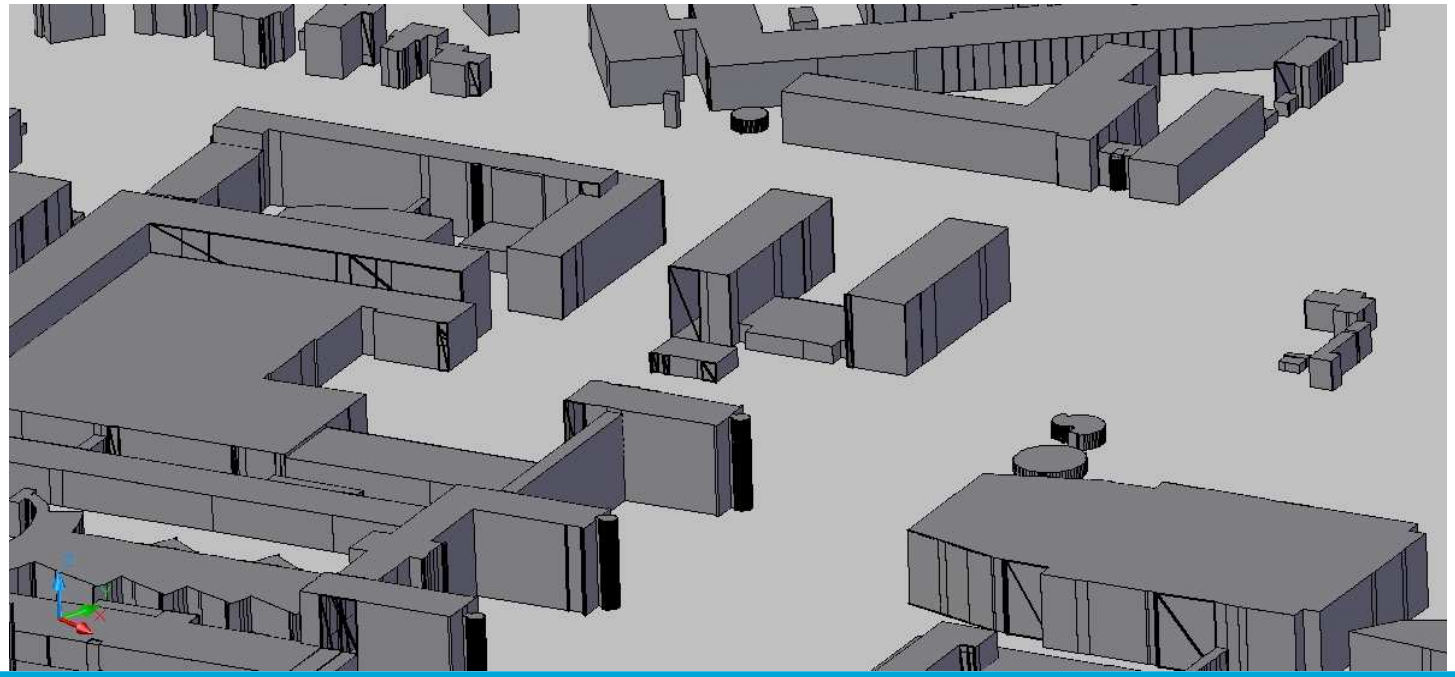
3D Extrusion Model of TU Delft

3D Extrusion using programming language Java

Dave Houben

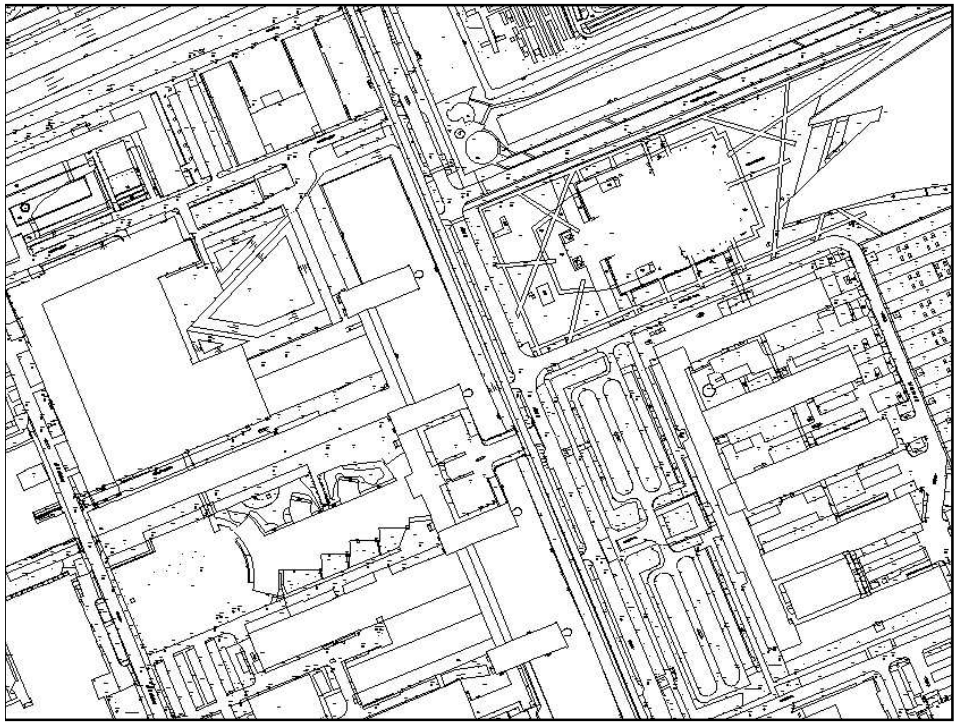


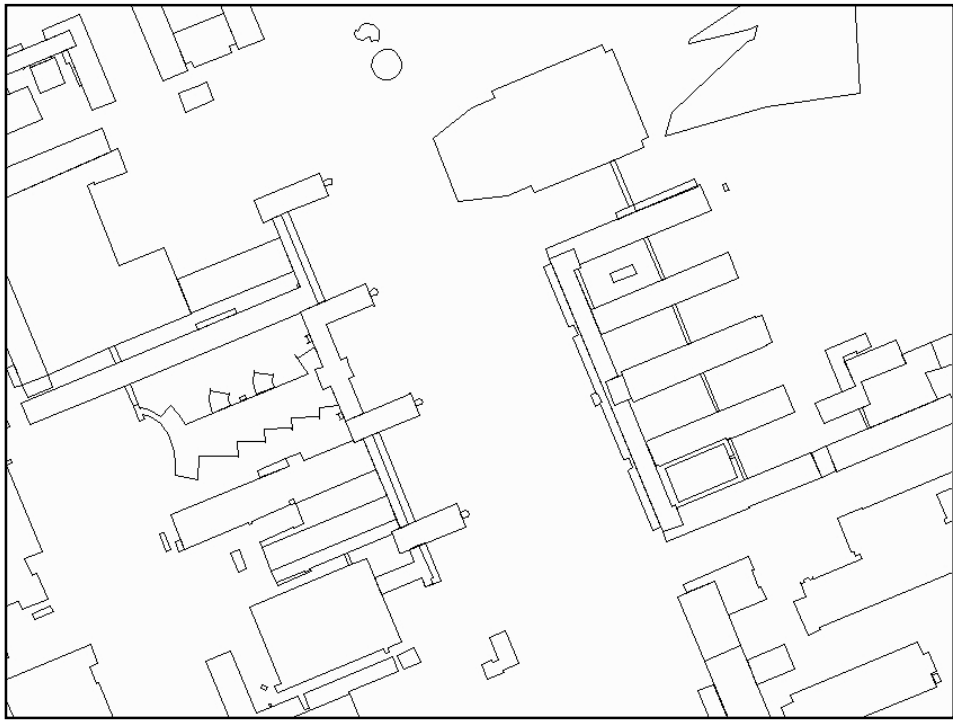
TU Delft
Delft University of Technology

Input data set

Input data set from 2D dwg drawing

- Extracting buildings from the original drawing





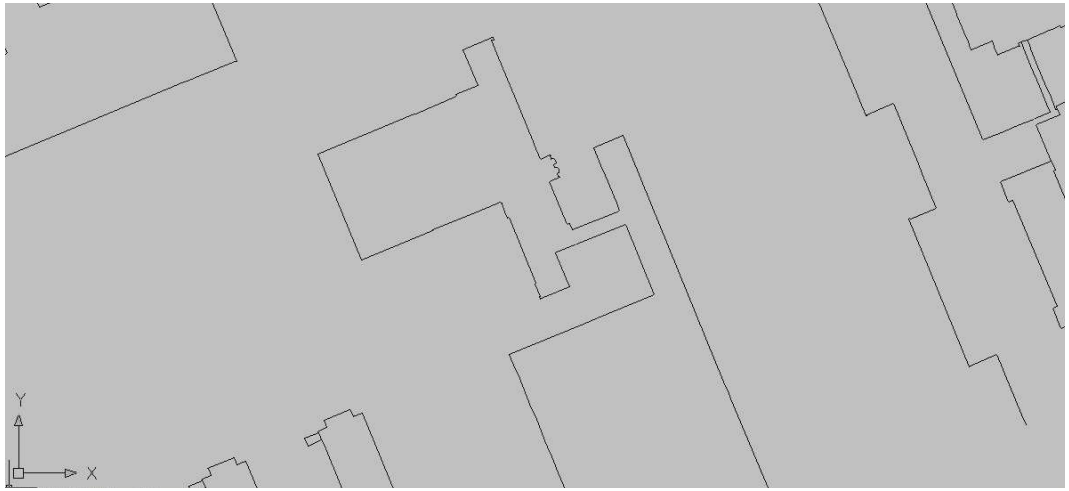
TU Delft

Input data set

Input data set from 2D dwg drawing

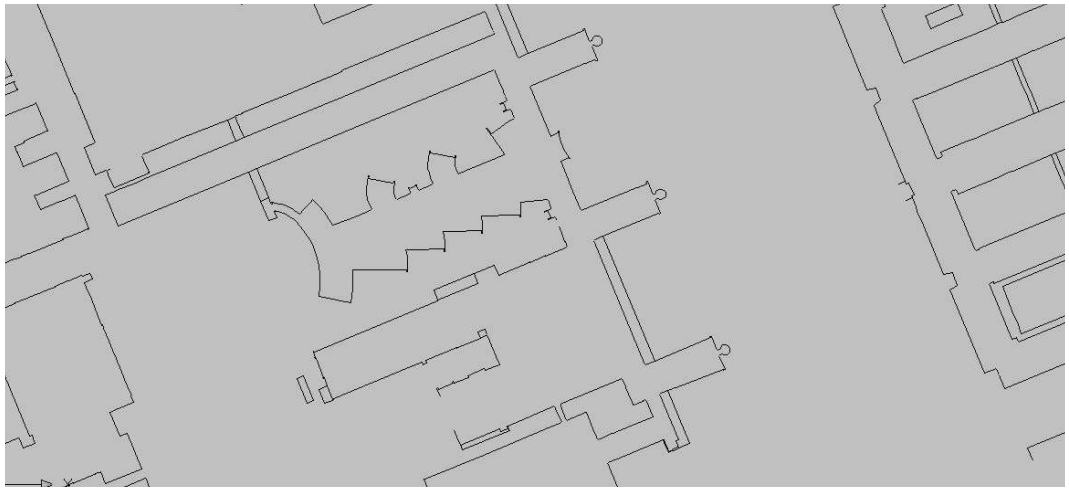
- Extracting buildings from the original drawing
- Manipulate data manually and with FME before extrusion
  - Add "missing" geometries to buildings
  - Snap lines to close areas
  - Create polygons
  - Simplify geometry
  - Delete polygon inside polygon
  - Delete unintentional polygons

Input data set – Missing geometry



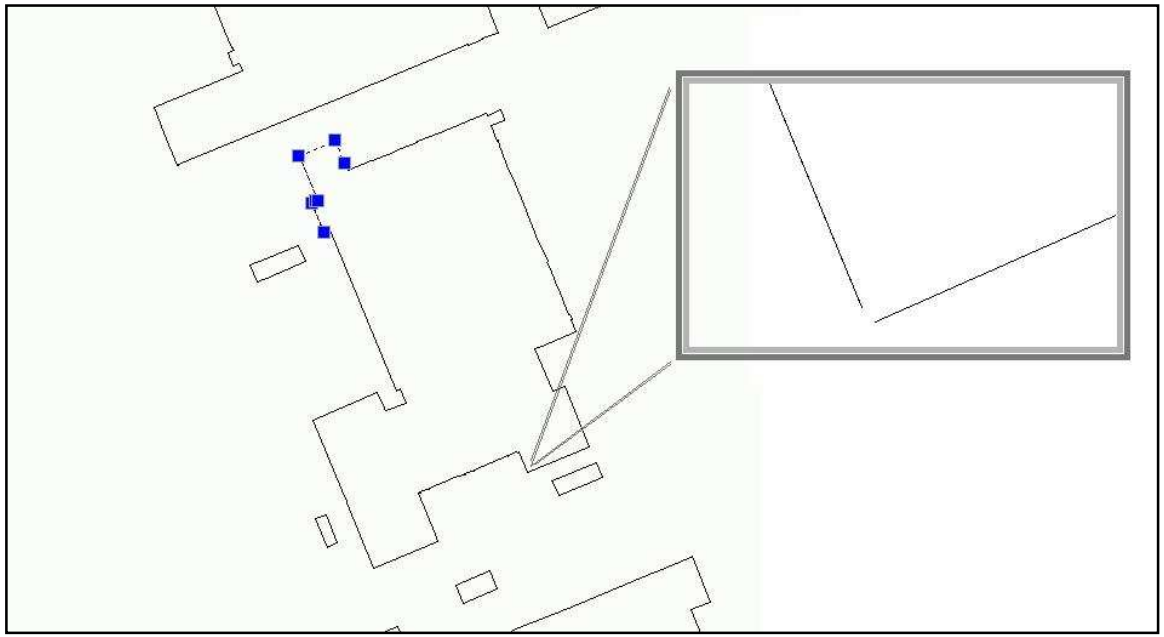
TU Delft

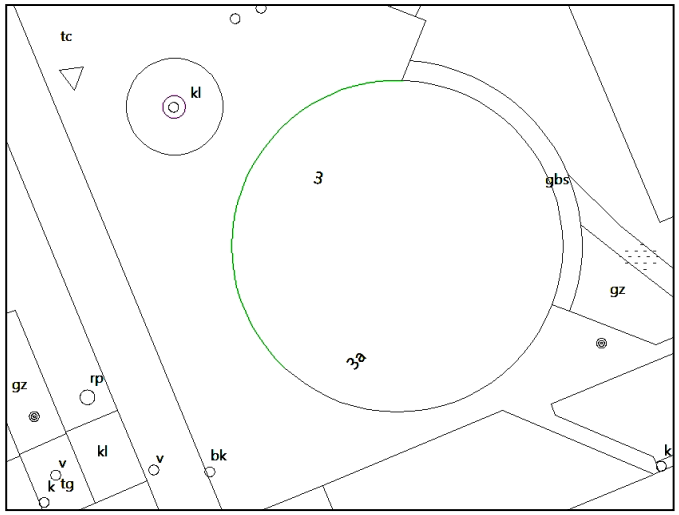
Input data set – Missing geometry



TU Delft

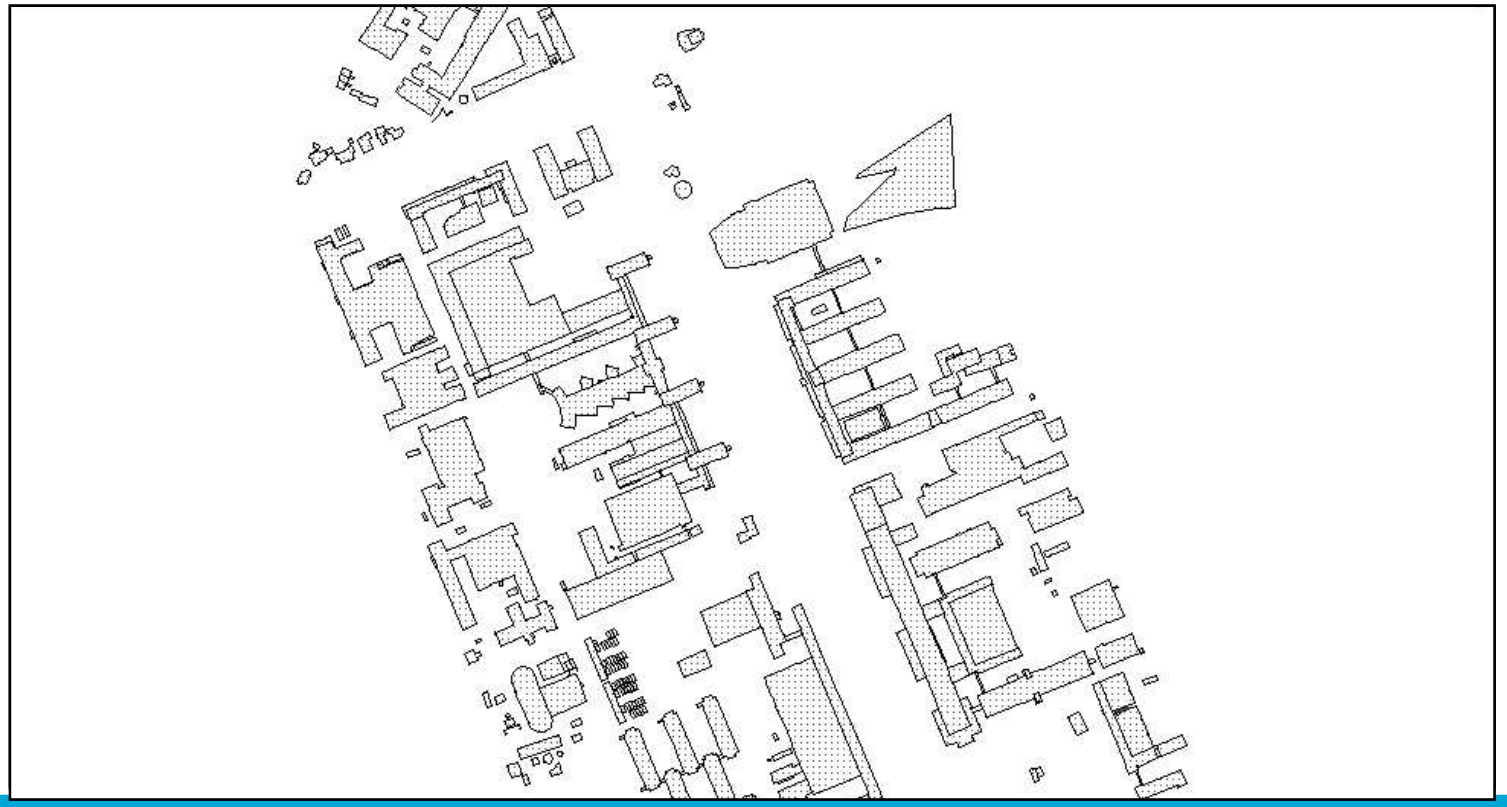
Input data set – Cleaning up





Point in Polygon

3D extrusion using laser elevation data and building polygons from 2D dxf drawing



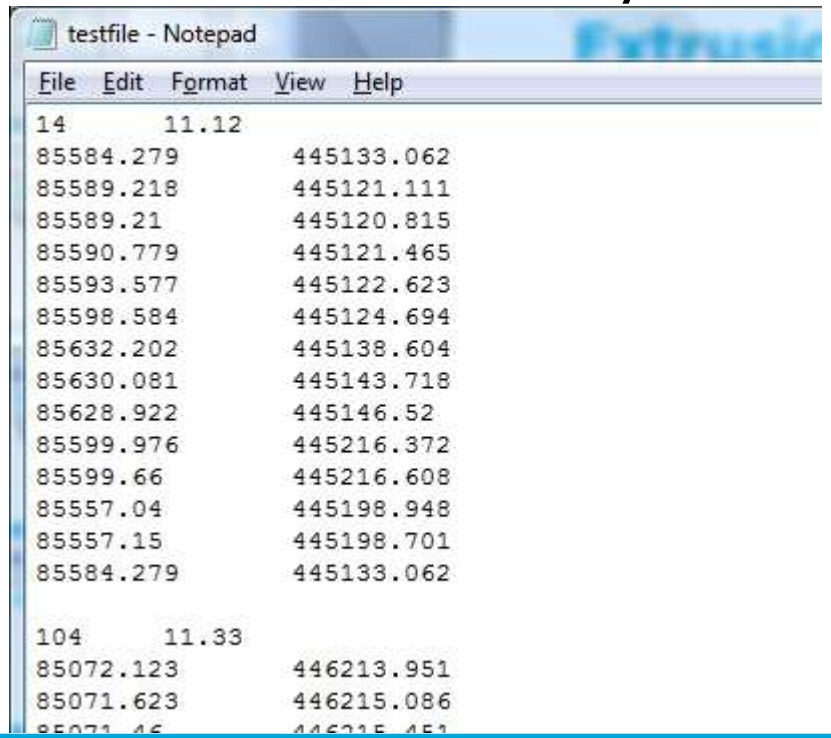
TU Delft

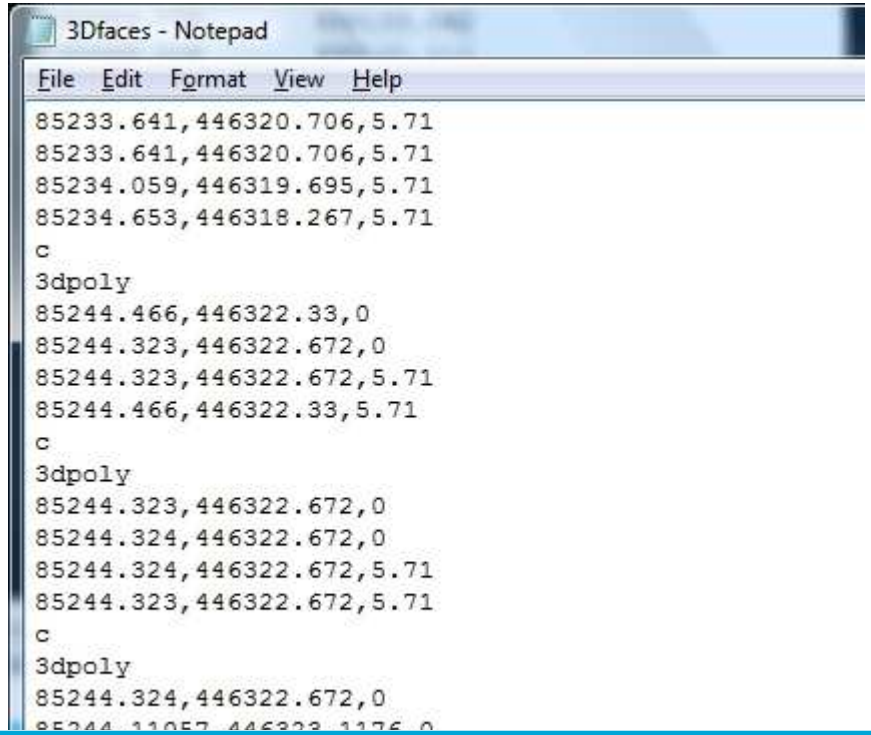
Point in Polygon

TU Delft

Extrusion to Polylines

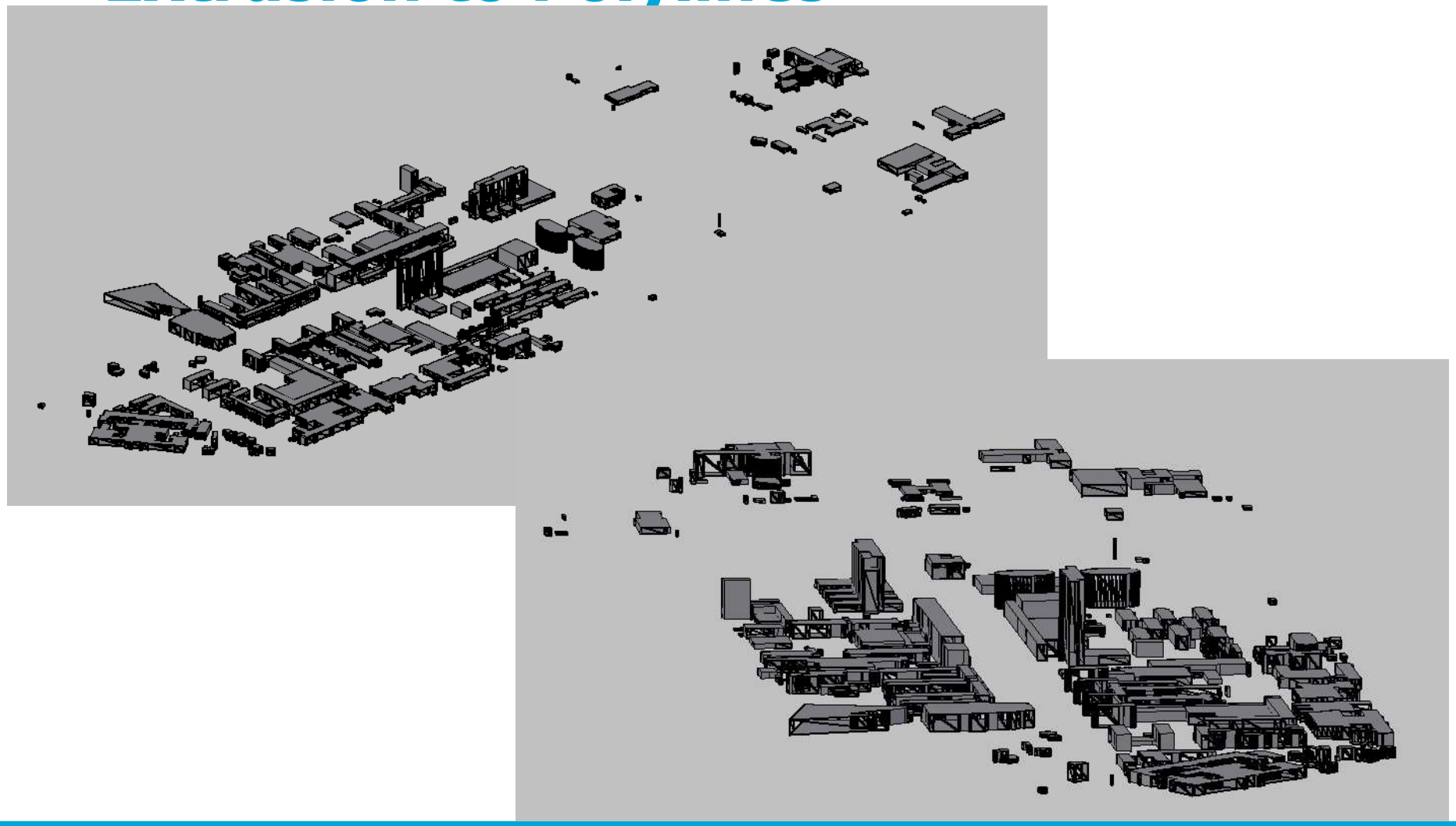
Java programming language is used to read the ascii input and create autodesk script output. The script can be read with any autodesk CAD software.





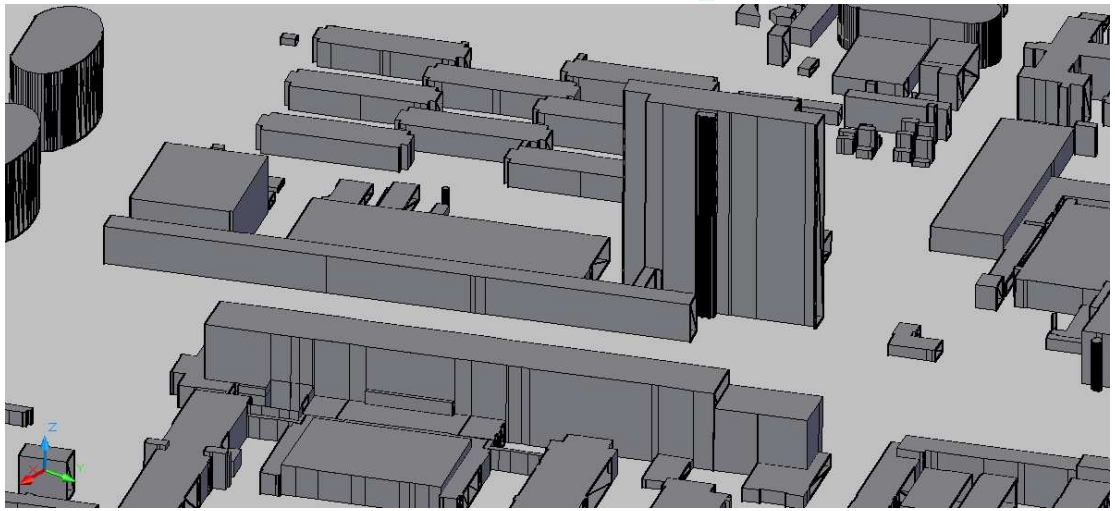
TU Delft

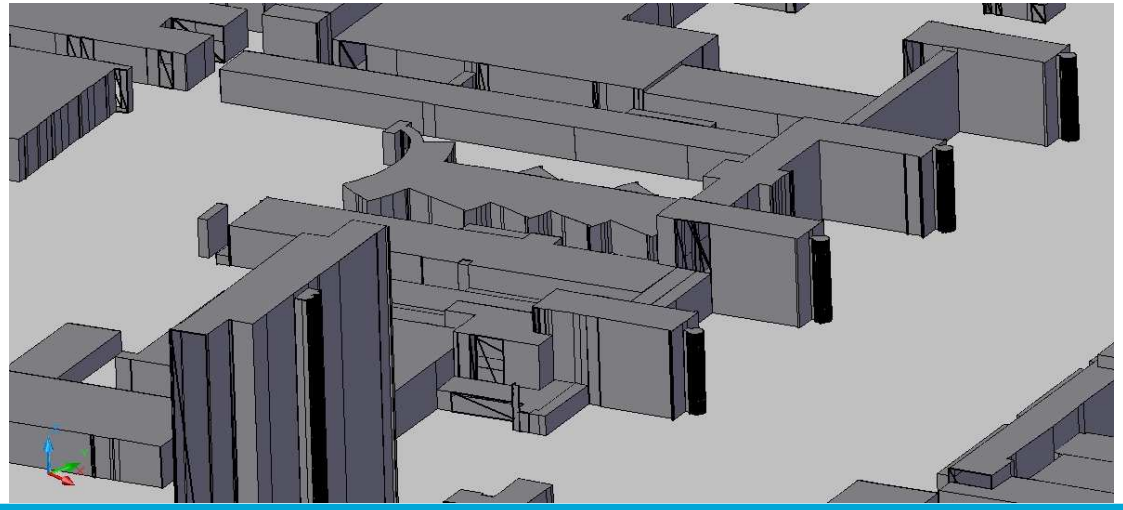
Extrusion to Polylines



TU Delft

Extrusion to Polylines





TU Delft

Different data output

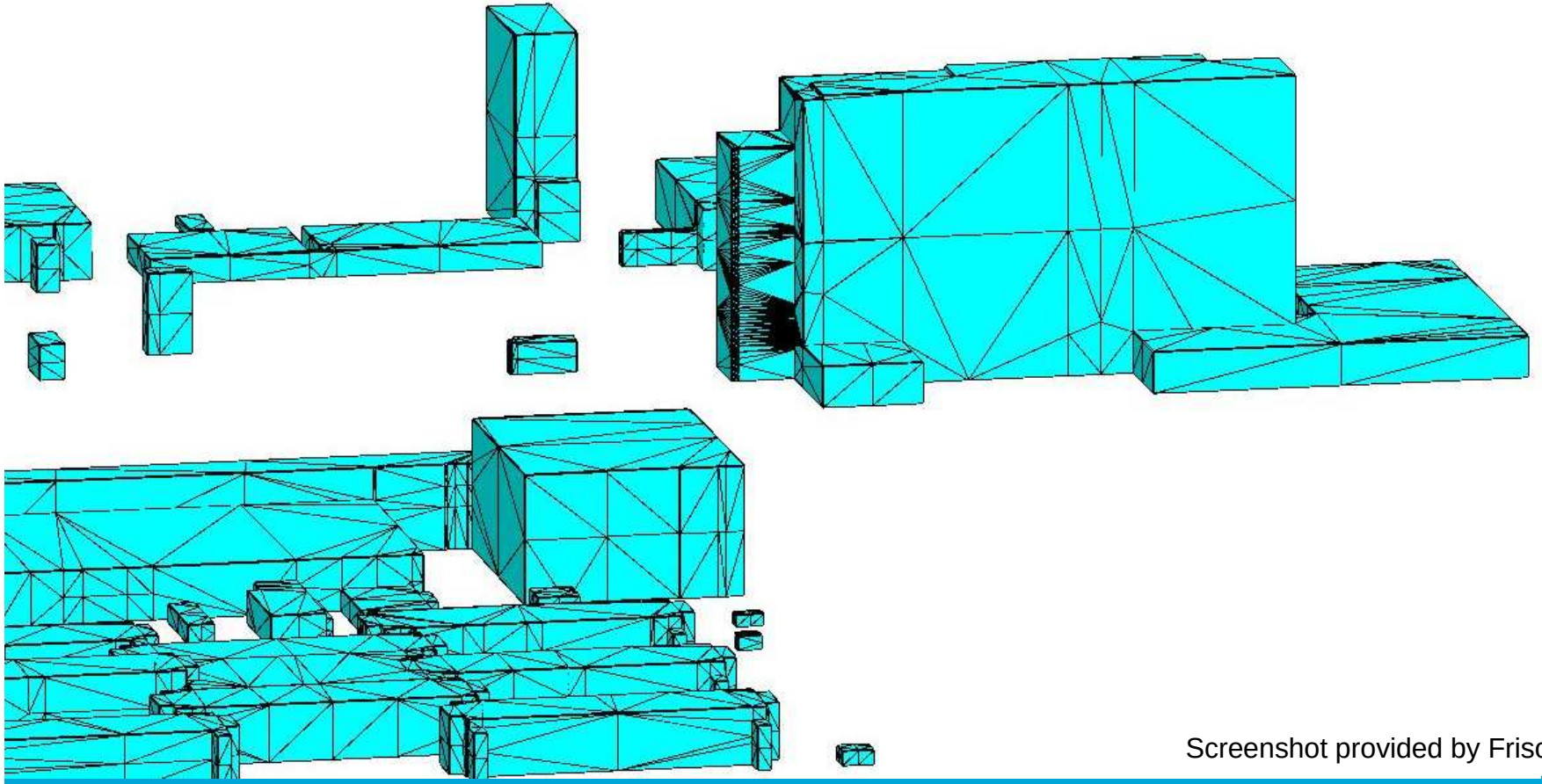
- *Output in autocad 3D polylines for input in database*

- Output in special format for TETGEN
  - No intersections between faces
  - No double points because of 1D surfaces

- Output in autocad 3D faces for SketchUp modelling
  - 3D faces only 3 or 4 points, therefore only walls
  - Simplify geometry for textures

TU Delft

Different data output - Tetgen



Screenshot provided by Friso Penninga

TU Delft