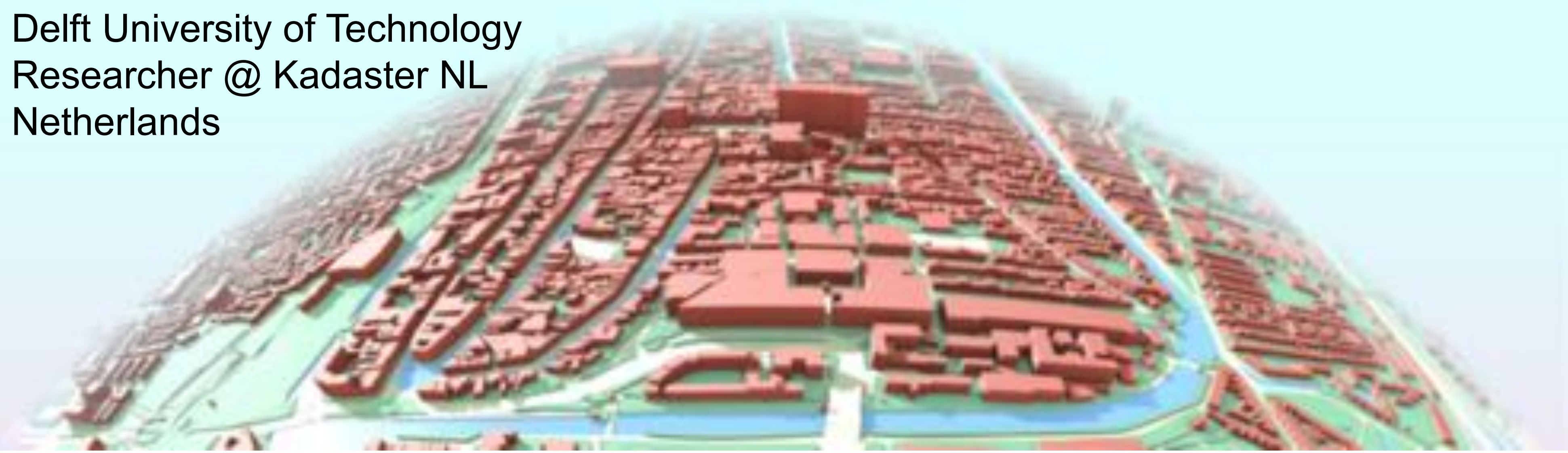
A flawless 4D city modelling information chain

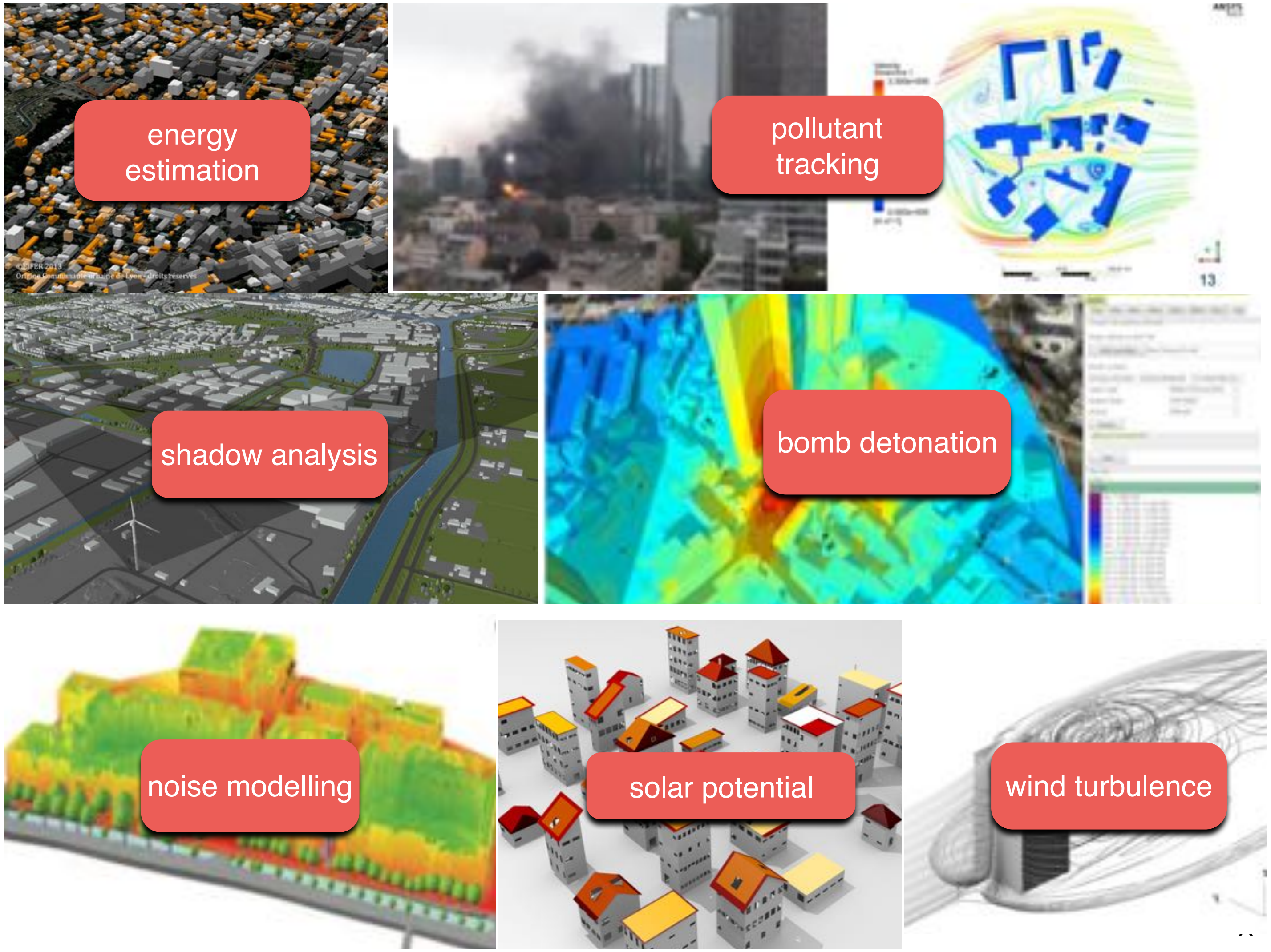
Where do 4D data requirements and 4D data collection possibilities meet?

Jantien Stoter
Professor 3D Geoinformation
Delft University of Technology
Researcher @ Kadaster NL
Netherlands



Relatively easy to reconstruct 3D city models

3D is used in city planning & environmental simulations



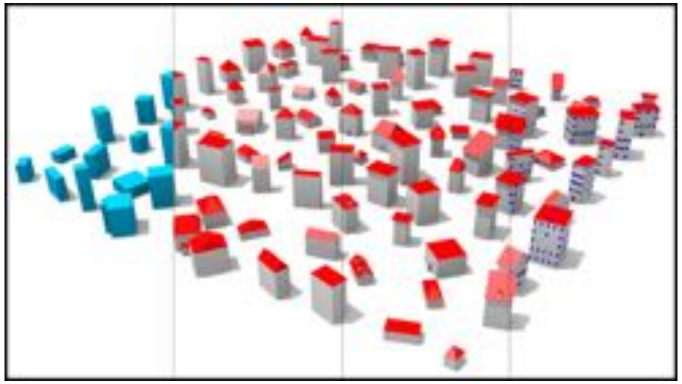
But 3D city models differ a lot, due to

differences in acquisition methods

- generated independently with different reconstruction methods, software and sensor data

differences in applications

- every application requires its own specific semantic and geometric LoD of the 3D data

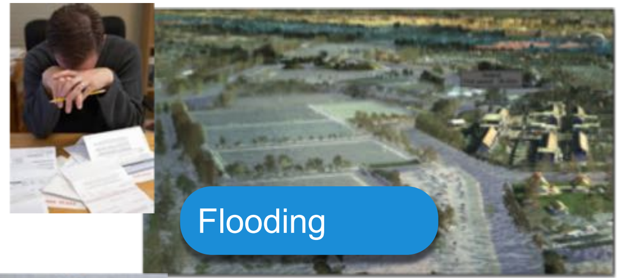


F.Biljecki

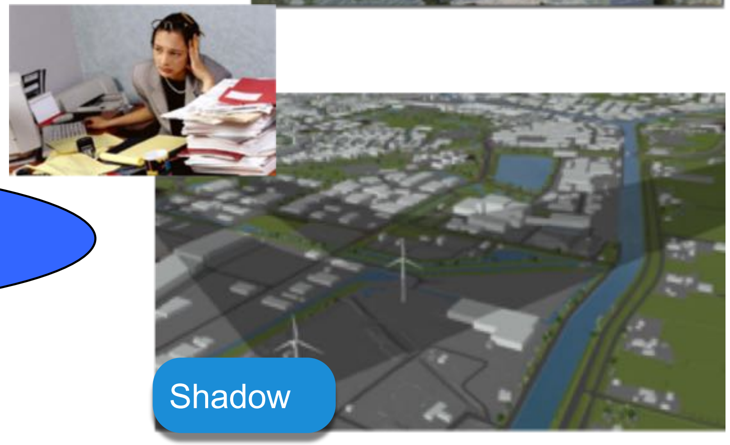
Problems of differences in current 3D city models

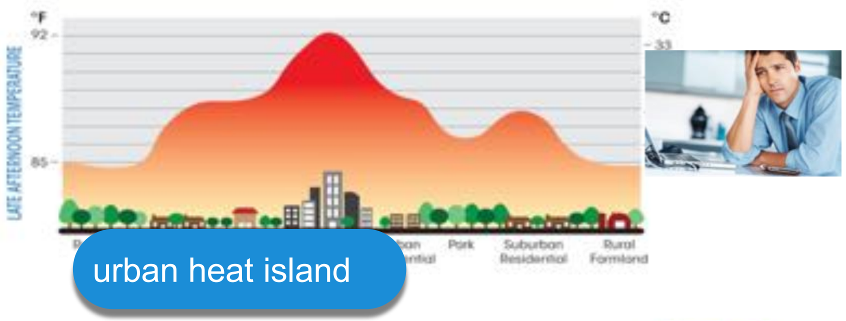
- Non-consistent
- Once collected 3D data for an appl can hardly be reused
- 3D city models often require (interactive) processing to use the data

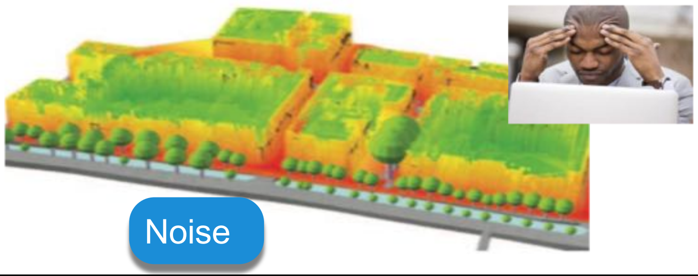
Domain experts spend 70% of their time on 3D data processing

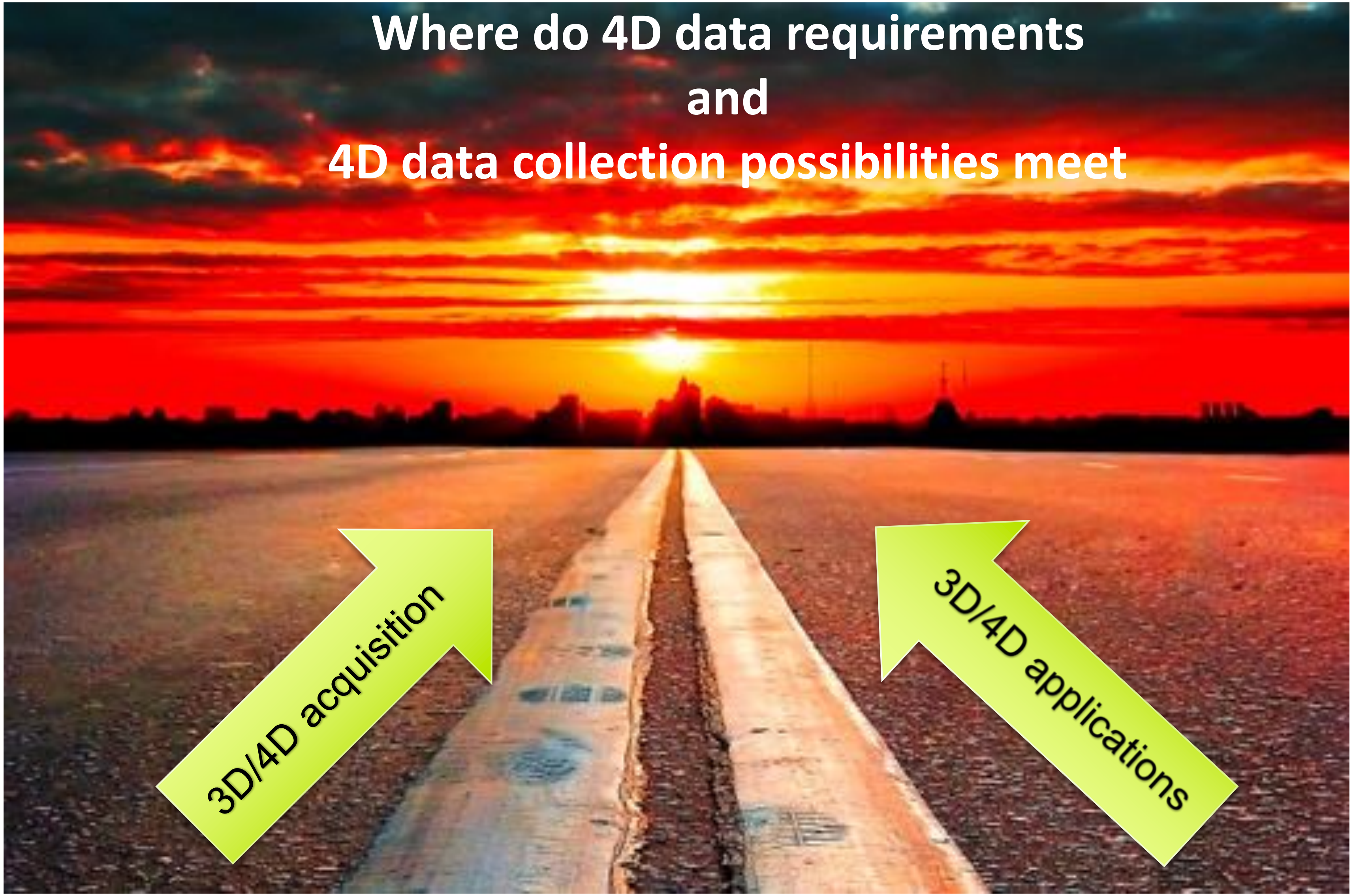


Where do I find useful data








Where do 4D data requirements
and
4D data collection possibilities meet
3D/4D acquisition
3D/4D applications

Where do 4D data requirements and 4D data collection possibilities

Content of my presentation:

- Current 4D modelling practices:
  - what do those imply for the data acquisition process?
- Quality requirements of 3D city models
- Data requirements of 3D/4D applications

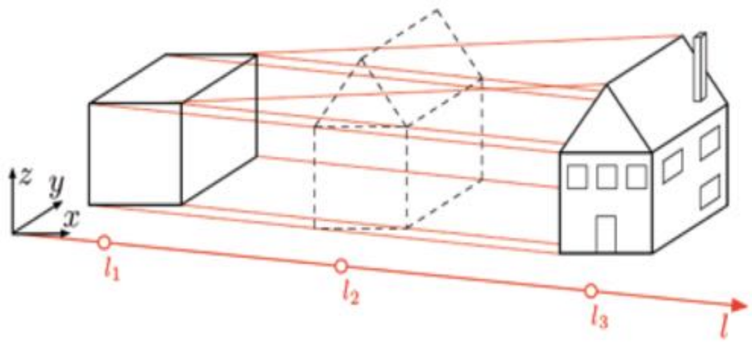
4D: 3D+time models

- Temporal requirements for acquisition:

  – Detect and acquire changes in reality

  – Models remain consistent over time if reality does not change
    - Use of dense image matching PC instead of LiDAR should yield same heights

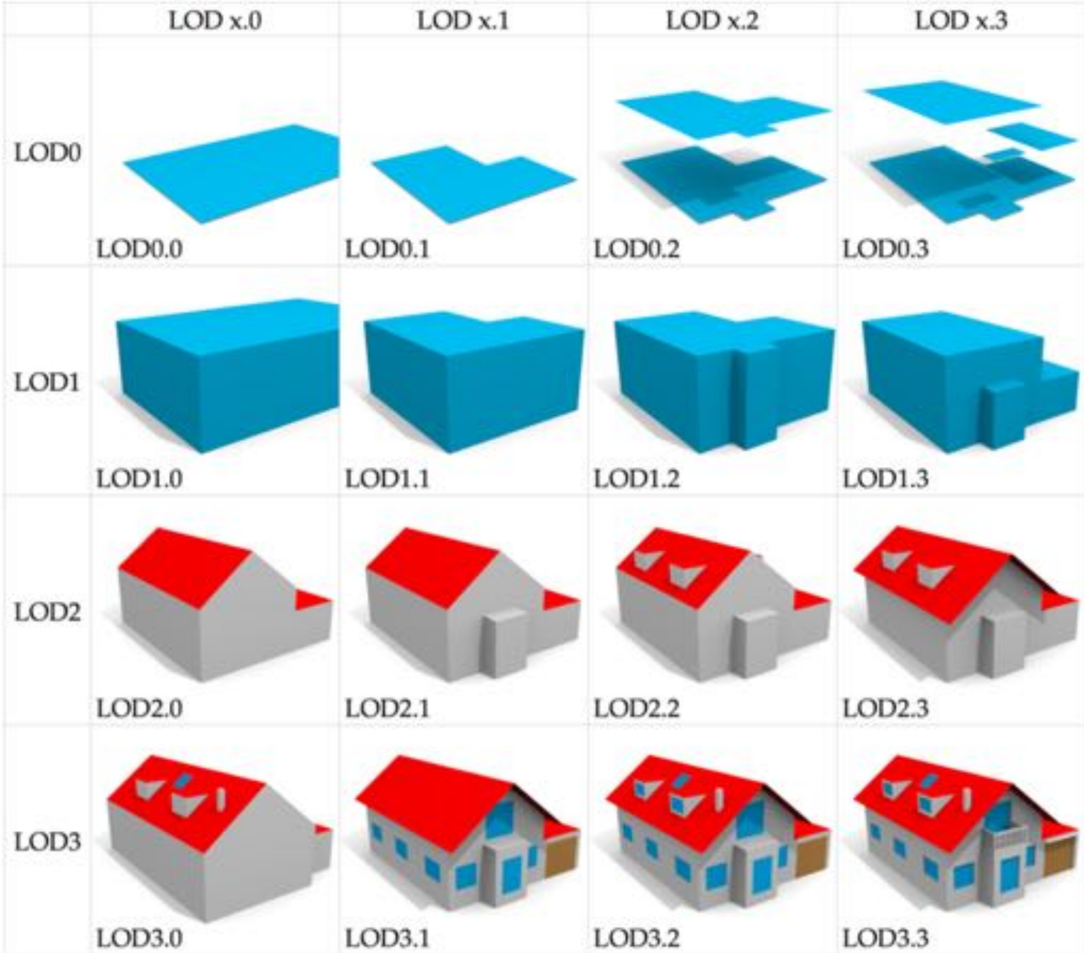
4D: 3D+LoD



- Well-known 5 LoDs for buildings in CityGML

- But what is less known….

Each LoD can have different implementations



Biljecki, 2016

Be precise & refine

– Simply saying “LoD2” is not sufficient

Even LoD1 models have different realisations

- Which height is used for extrusion?
  - Gutter? Maximum height? 2/3, 1/2 of roof height?
  - Application dependent
- How calculated? e.g. max height:
  - Highest point that falls in polygon? Median? Using buffer?
- Often users are not aware of possible differences

Be clear about which height reference and how obtained (preferably more than one)

More standardisation is needed

Biljecki, 2016

INSPIRE
Data Specs for Buildings

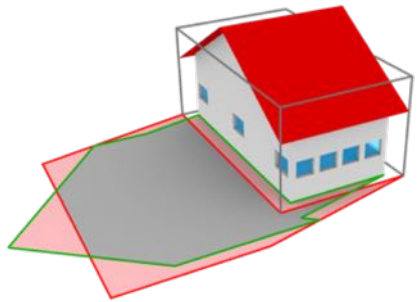
- attribute elevationReference specifies height reference

Figure 15: Examples of elevation references for different kinds of building

"The more detailed, the better"
"Lod2 is more accurate than LoD1"

The effect of acquisition error and level of detail on the accuracy of spatial analyses
Filip Biljecki, G Heuvelink, H Ledoux, J Stoter, Cartography and Geographic Information Science, 45(2): 156-176, 2018.

- Accuracy of acq method more impact on quality of spatial analysis than LoD
- Higher LoDs do not always bring significant improvements
  - LoD1 versus LoD3 for shadow estimation
- 3D CMs can be too detailed

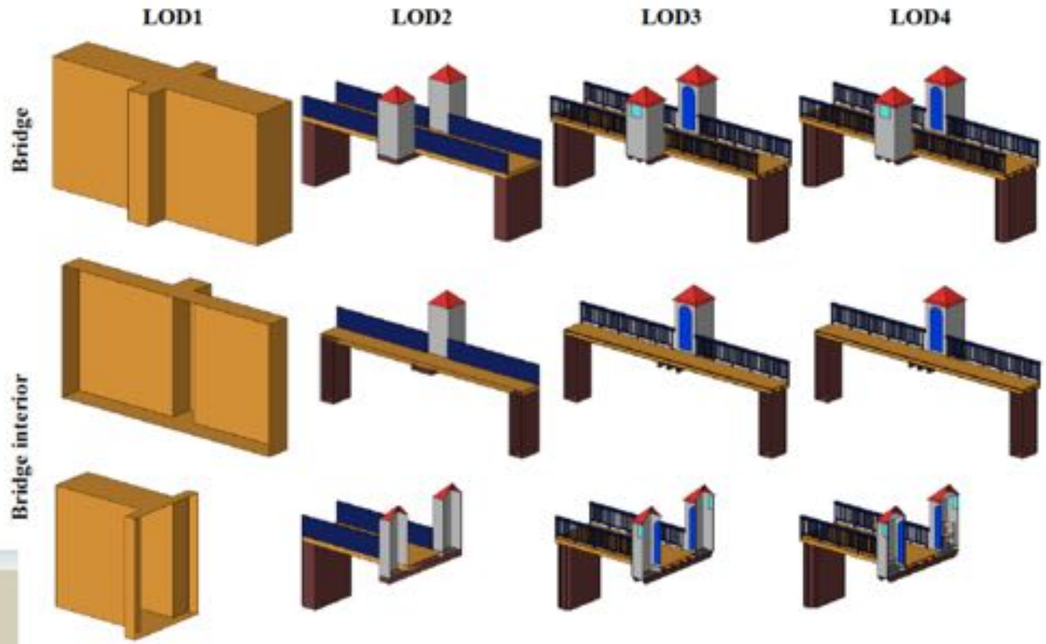


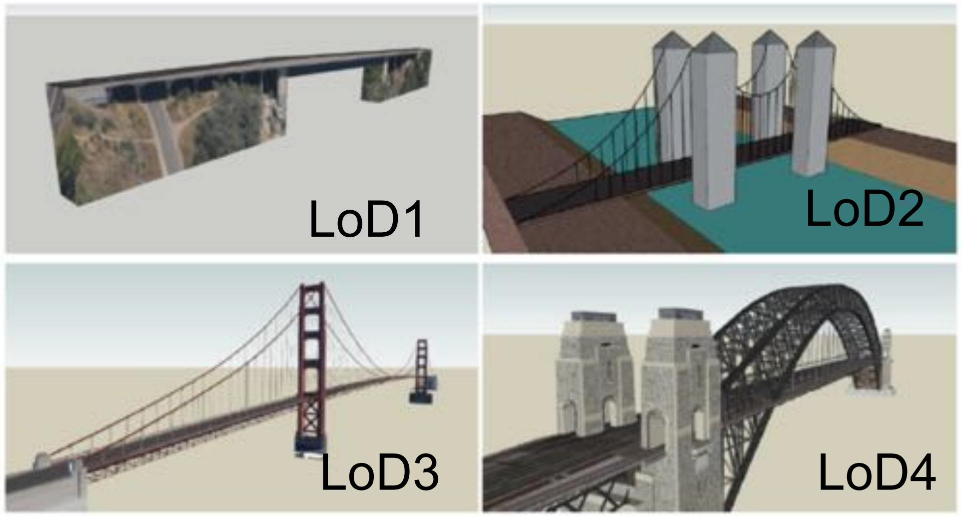
➡ Not always strive for highest LoD, relate it to app

A lot is known about LoD of buildings

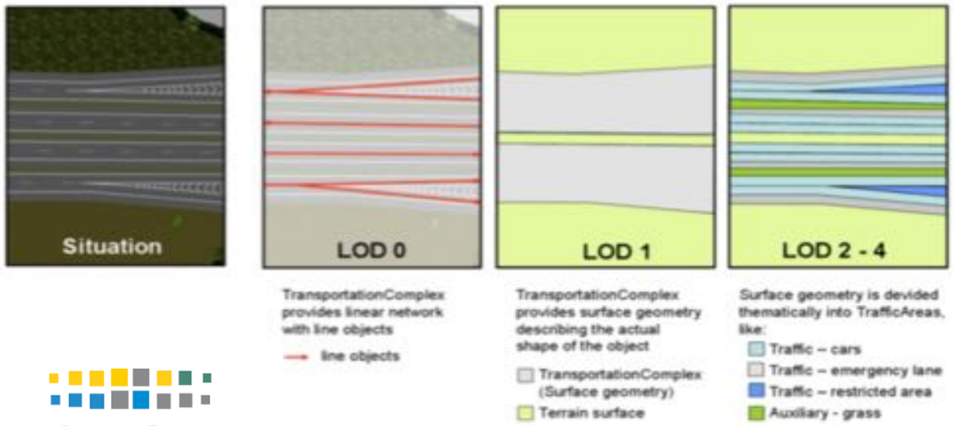
what about other types of features?

Bridges and tunnels

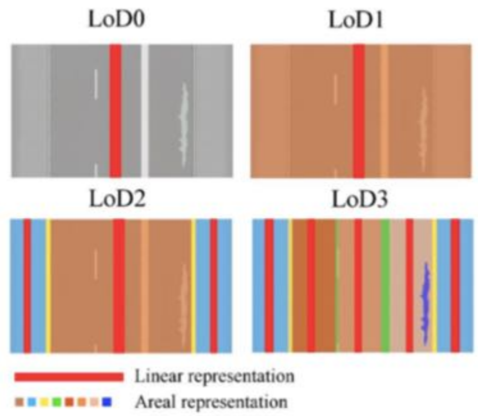




LoD Roads (transport) in CityGML



CityGML



Beil, C. and Kolbe, T. H., 2017

- Network only for LoD0
- Lod1-4 surfaces (no relation with network)

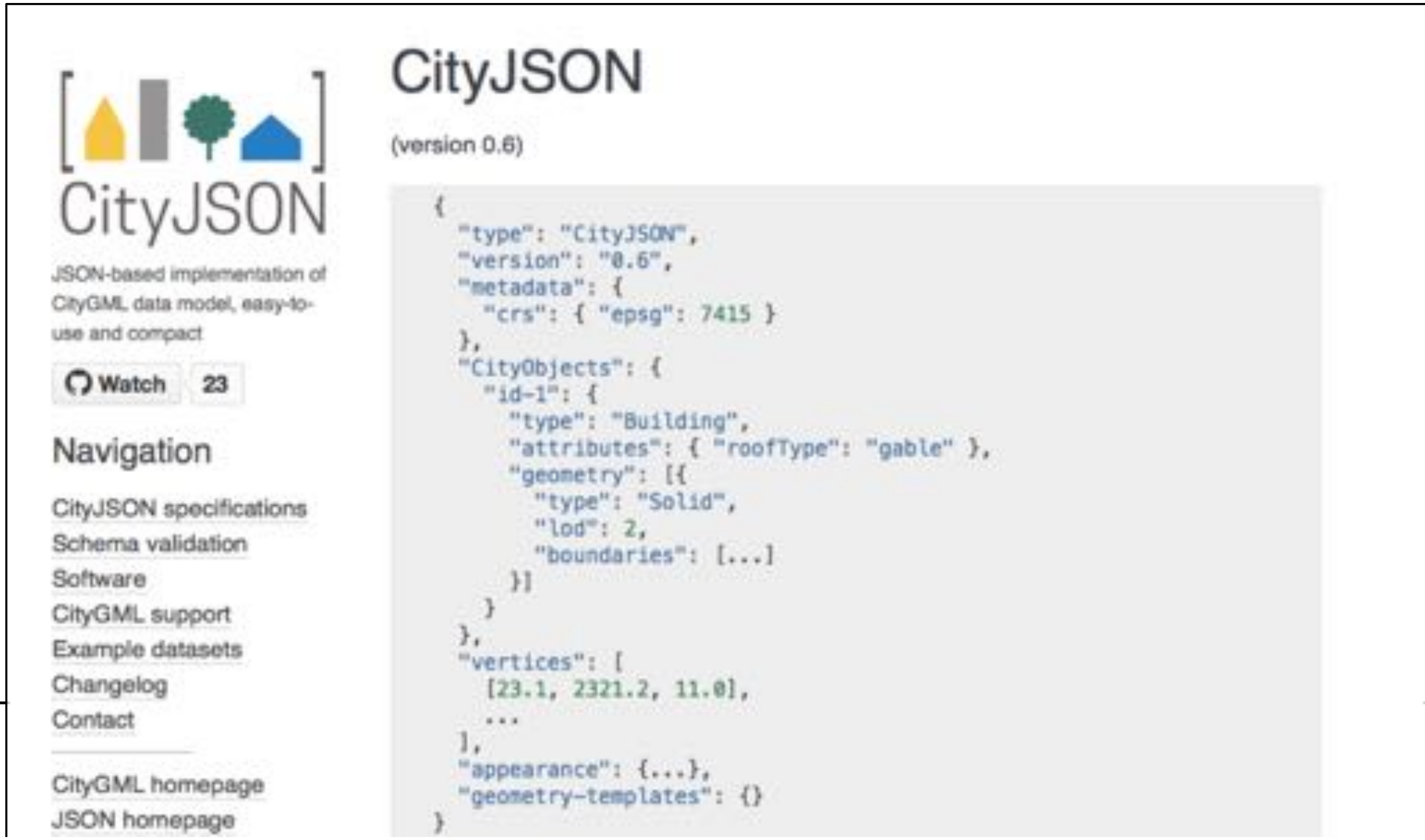
CityGML as a data format

Complete, but verbose & complex, and therefore sometimes difficult to work with

CityJSON encoding

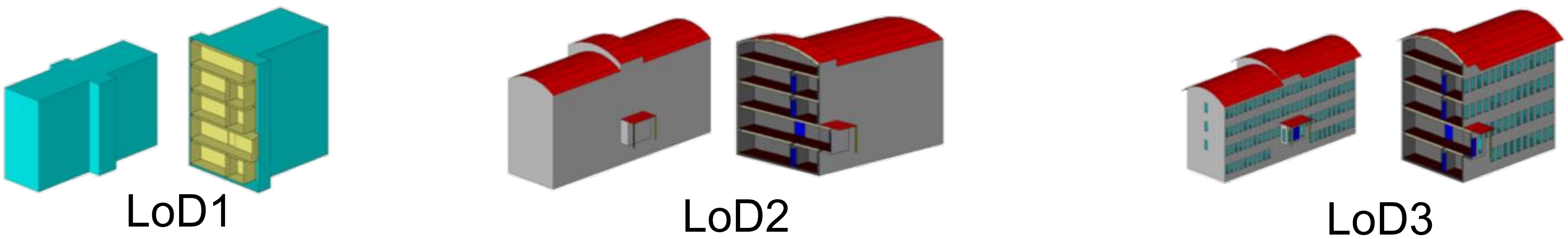
JavaScript Object Notation

– Easy-to-use for developers; compression 7 to 10 x compared to CityGML



CityGML 3.0

- Major revision compared to 2.0
- Support for storeys; versioning
- No LoD4; LoD0-LoD3 for indoor and outdoor
- Distinguish between Conceptual model and GML encoding



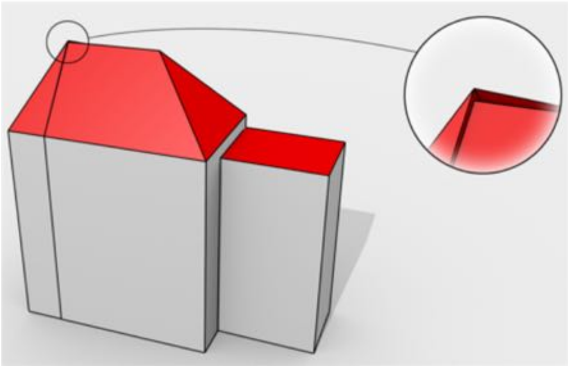
Content

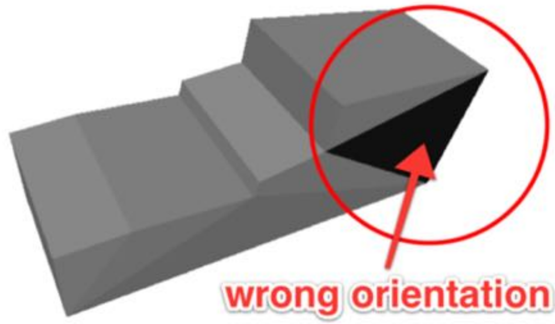
- Current 4D modelling practices:
  - what does that imply for the data acquisition process?
- **Quality requirements of 3D city models**
- Data requirements of 3D/4D applications

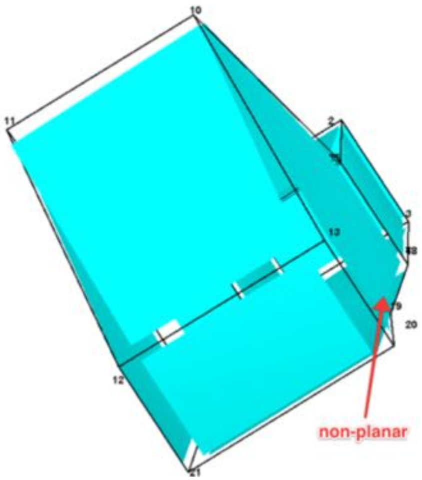
3D model is not a 1:1 model of reality

- For 3D applications, we need:
  - Data beyond the “wow” effect
  - Up-to-date
    - Not only acquisition: also maintenance
  - Consistent (4D)
  - Without errors

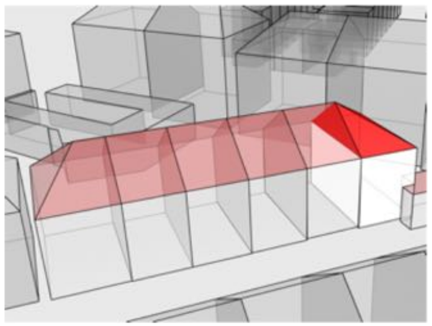
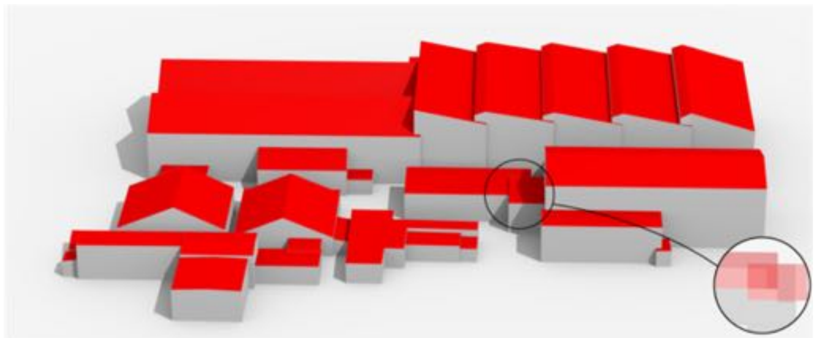
errors = common in 3D





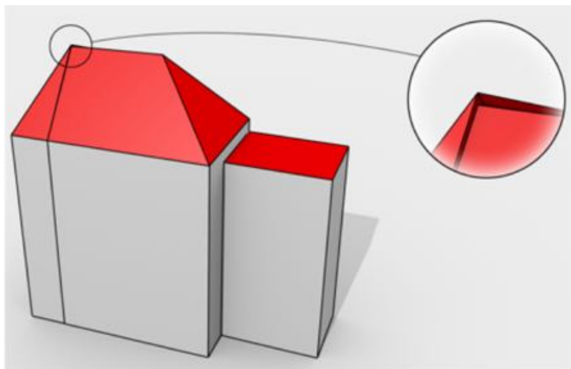




Errors in 3D models

- Not visible-> users are not aware
- May give no problems in specific software or applications
- But not possible to reuse 3D data in other software and applications



Software to validate 3D data

- Validates geometries according to international standards (ISO19107 & OGC)
- Web interface: http://geovalidation.bk.tudelft.nl
- Reads CityGML

***Val3dity: validation of 3D GIS primitives according to the international standards**. Hugo Ledoux. Open Geospatial Data, Software and Standards 3 (1), 2018, pp. 1*

Software to validate 3D data

To understand quality of existing 3D data sets

- Applied to 37 datasets in 9 countries
  - 3.6m buildings
  - 16m 3D primitives
  - 40m surfaces

Conclusion
validating existing 3D city models

The most common geometric and semantic errors in CityGML datasets
Filip Biljecki, Hugo Ledoux, Xin DU, Jantien Stoter, Kean Huat SOON, Victor KHOO
ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci., IV-2/W1: 13-22, 2016.

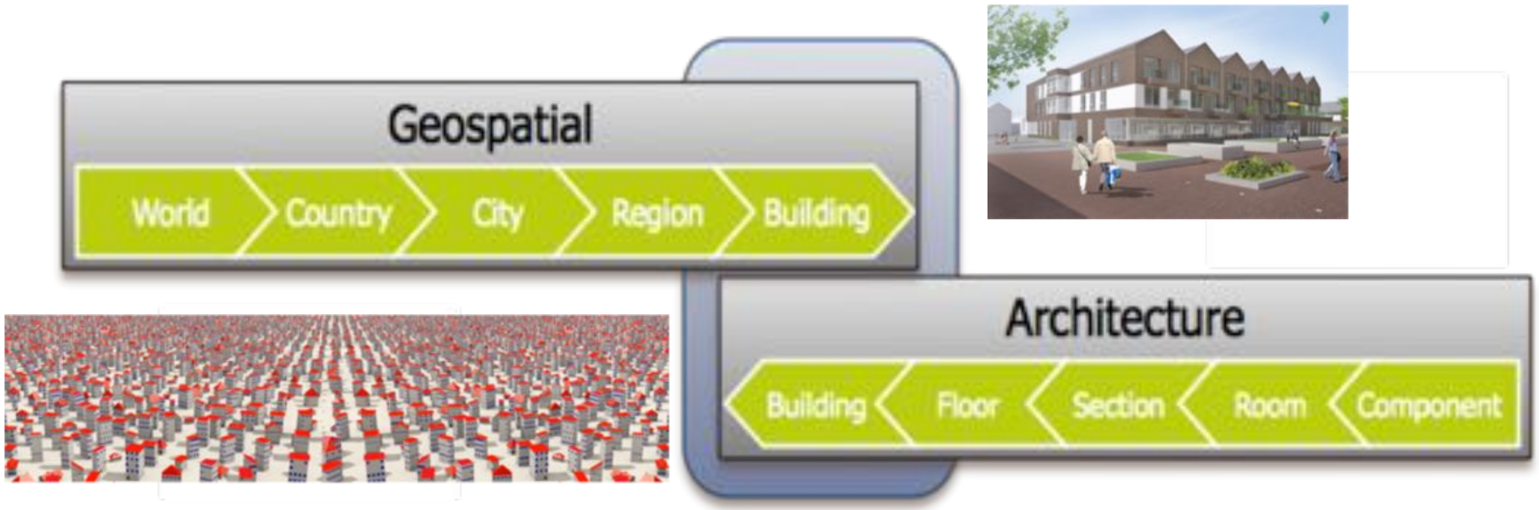
- CityGML data without errors are rare
- Most valid models are LoD1 models
- Many errors can be automatically fixed or prevented:
  - missing faces; geometries not properly snapped; orientation of surfaces; non planar faces (often caused by deviations of few cm only)

→ ***Reconstruct valid 3D models, if you want your 3D data to be (re)used***

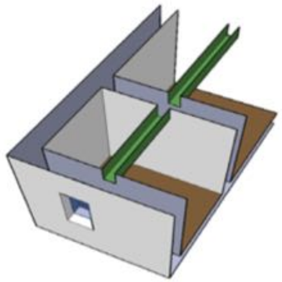
Content

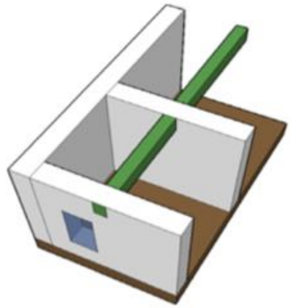
- Current 4D modelling practices:
  - what does that imply for the data acquisition process?
- Quality requirements of 3D city models
- **Data requirements of 3D/4D applications**

	3D GIS	BIM
Focus area:		



Concepts:	Geographical concepts: space, outdoor shell	Constructions





Geometry:	Measured	Designed

GeoBIM integration: to reuse data

“easily” reconstructedwith point clouds

rather rare in practice, because costly and complex to acquire data

- To be realised with IFC

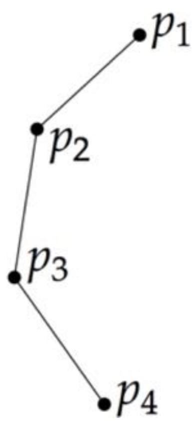
Industry Foundation Classes

IfcActuatorType
IfcAirTerminalBoxType
IfcAirTerminalType
IfcAirToAirHeatRecoveryType
IfcAlarmType
IfcAnnotation
IfcBeam
IfcBoilerType
IfcBuildingElementPart
IfcBuildingElementProxy
IfcBuildingStorey
IfcCableCarrierFittingType
IfcCableCarrierSegmentType
IfcCableSegmentType
IfcChillerType
IfcCoilType
IfcColumnType
IfcCompressorType
IfcCondenserType
IfcControllerType
IfcCooledBeamType
IfcCoolingTowerType
IfcCovering
IfcCurtainWall
IfcDamperType
IfcDistributionChamberElementType
IfcDistributionControlElement
IfcDistributionElement
IfcDistributionFlowElement
IfcDoorType
IfcDuctFittingType
IfcDuctSegmentType
IfcDuctSilencerType
IfcElectricApplianceType
IfcElectricFlowStorageDeviceType
IfcElectricGeneratorType
IfcElectricHeaterType
IfcElectricMotorType
IfcElectricTimeControlType
IfcElementAssembly
IfcEnergyConversionDevice
IfcEvaporativeCoolerType
IfcEvaporatorType
IfcFanType
IfcFastenerType
IfcFilterType
IfcFireSuppressionTerminalType
IfcFlowController
IfcFlowFitting
IfcFlowInstrumentType
IfcFlowMeterType
IfcFlowMovingDevice
IfcFlowSegment
IfcFlowStorageDevice
IfcFlowTerminal
IfcFlowTreatmentDevice
IfcFooting
IfcFurnishingElement
IfcFurnitureType
IfcGasTerminalType
IfcHeatExchangerType
IfcHumidifierType
IfcJunctionBoxType
IfcLampType
IfcLightFixtureType
IfcMechanicalFastenerType
IfcMemberType
IfcMotorConnectionType
IfcOpeningElement
IfcOutletType
IfcPile
IfcPipeFittingType
IfcPipeSegmentType
IfcPlateType
IfcProtectiveDeviceType
IfcPumpType
IfcRailing
IfcRamp
IfcReinforcingBar
IfcReinforcingMesh
IfcRoof
IfcSanitaryTerminalType
IfcSensorType
IfcSite
IfcSlab
IfcSpace
IfcSpaceHeaterType
IfcStackTerminalType
IfcStair
IfcSwitchingDeviceType
IfcSystemFurnitureElementType
IfcTankType
IfcTransformerType
IfcTransportElementType
IfcTubeBundleType
IfcUnitaryEquipmentType
IfcValveType
IfcWall
IfcWasteTerminalType
IfcWindowType

1000+ in total

Many more IFC geometric classes than GML (point, curve, surface and solid)

Curves/wires



IfcCircle
IfcEllipse
IfcLine
IfcEdge
IfcOrientedEdge
IfcEdgeLoop
IfcPolyLoop
IfcPolyline
IfcCompositeCurve
IfcTrimmedCurve

Faces

- IfcArbitraryClosedProfileDef
- IfcArbitraryProfileDefWithVoids
- IfcRectangleProfileDef
- IfcRoundedRectangleProfileDef
- IfcRectangleHollowProfileDef
- IfcTrapeziumProfileDef
- IfcCircleProfileDef
- IfcCircleHollowProfileDef
- IfcEllipseProfileDef
- IfcFace

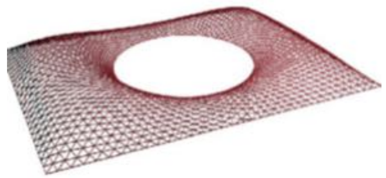
IfcCShapeProfileDef
IfcLShapeProfileDef
IfcIShapeProfileDef
IfcTShapeProfileDef
IfcUShapeProfileDef
IfcZShapeProfileDef
IfcDerivedProfileDef

Volumetric shapes

- IfcExtrudedAreaSolid
- IfcExtrudedAreaSolidTapered
- IfcConnectedFaceSet
- IfcCsgSolid
- IfcBlock
- IfcBooleanResult
- IfcSphere
- IfcRectangularPyramid
- IfcRightCircularCylinder
- IfcRightCircularCone
- IfcTriangulatedFaceSet
- IfcHalfSpaceSolid

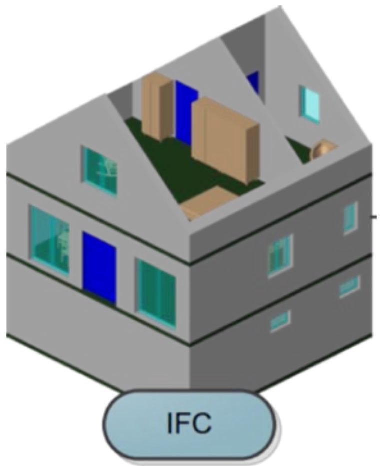
Abstract shapes

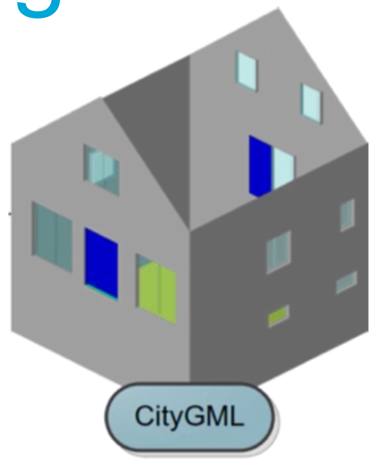
IfcRepresentation
IfcGeometricSet
IfcShellBasedSurfaceModel
IfcManifoldSolidBrep
IfcMappedItem
IfcFaceBasedSurfaceModel



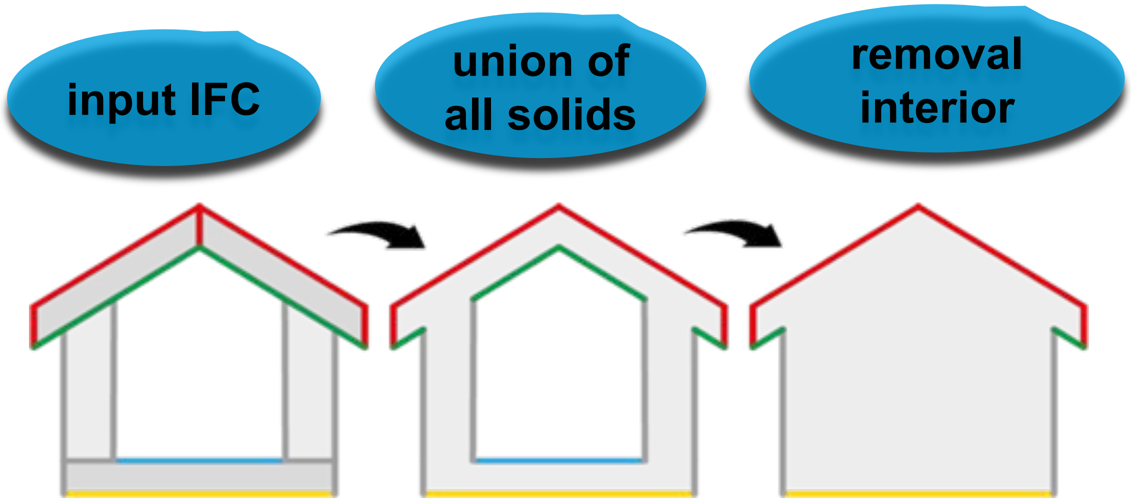
IFC -> CityGML
Not just conversion of geometry

Model with high detail and many objects





One object as closed volume; no “thick” walls



S Donkers, H Ledoux, J Zhao, J Stoter: ***Automatic conversion of IFC datasets to geometrically and semantically correct CityGML LOD3 buildings****. Trans. GIS 20(4): (2016)*

Conversion IFC-> CityGML

- Works on “academic” and “clean” IFC models:
  - Modelled as expected
  - Without errors

- In practice, conversion of real models is difficult, in practice:
  - IFC files are not “standard” and vary a lot in their structure and classes used
  - IFC models contain errors, because support of main softwares is missing




Making location count.

Future City Pilot-1: Using IFC/CityGML in Urban Planning Engineering Report

Publication Date: 2016-10-03

Approval Date: 2017-08-17

Posted Date: 2017-06-27

Reference number of this document: OGC 16-097

Reference URL for this document: http://www.opengis.net/doc/PER/FCP1-UPrules

Category: Public Engineering Report

Editor: Mohsen Kalantari

Title: Future City Pilot 1: Using IFC/CityGML in Urban Planning Engineering Report

OGC Engineering Report

Main conclusion:

"*integration was not possible due to inconsistent coding of IFC elements that made transformation to CityGML complicated*"

-> "*a clear set of specification needs to be set for the preparation of IFC files*"

We're making specific recommendations for geo-ready IFC data

Instead of throwing data over the fence, enable downstream use of the data

1. How to construct valid volumetric objects
2. How to avoid self-intersections
3. Where IfcSpaces should be used
4. Which Ifc classes should be used
5. How to correctly georeference

EuroSDR GeoBIM project

EuroSDR

- Lantmateriet Sweden
- GUGiK Poland
- NLS, Finland
- Kartverket, Norway
- ADSE, Denmark
- Kadaster, NL
- Swisstopo, Switzerland
- Ordnance Survey, UK
- Ordnance Survey, Ireland
- IGN, France
- ICGC, Catalonia

Use case 1:
From design to construction

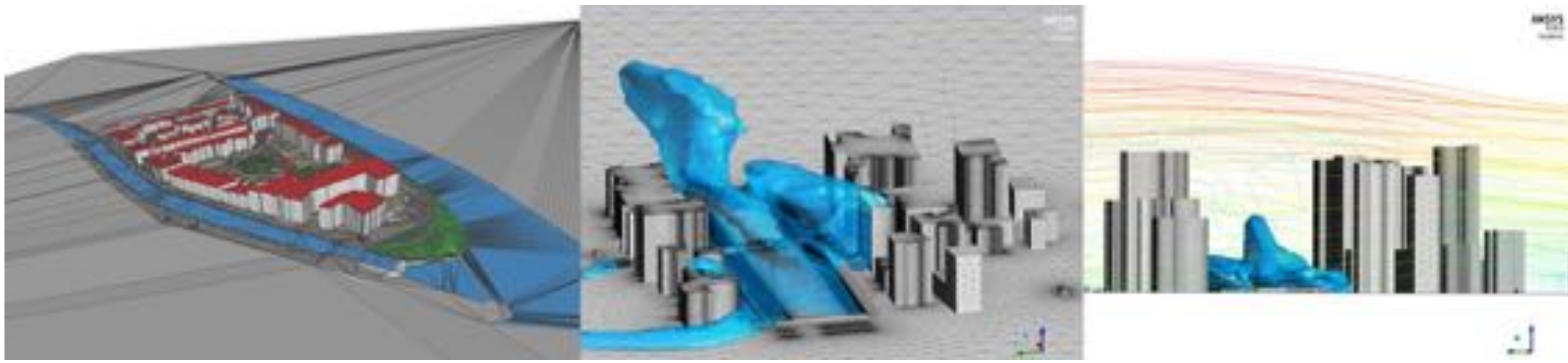
Use case 2:
Lifecycle support in AM

Applications

- Few words about other applications

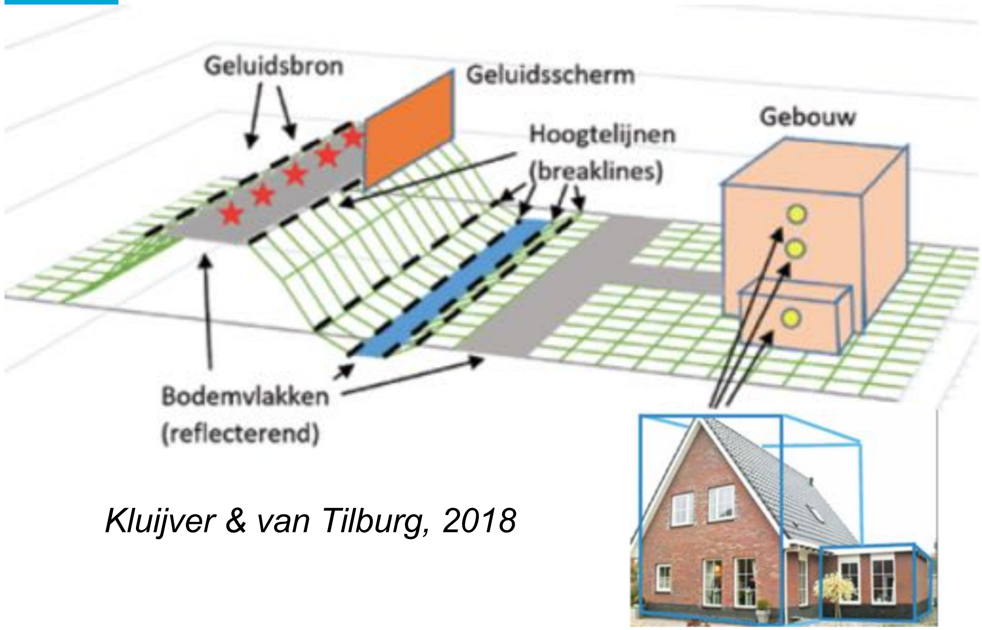
3D data for Simulation - CFD

- Computer fluid dynamics modelling (wind, air quality, temperature)
- Application specific requirement of CFD modelling:
  - LoD1 model (max height)
  - should be 100% closed



3D data for noise simulations

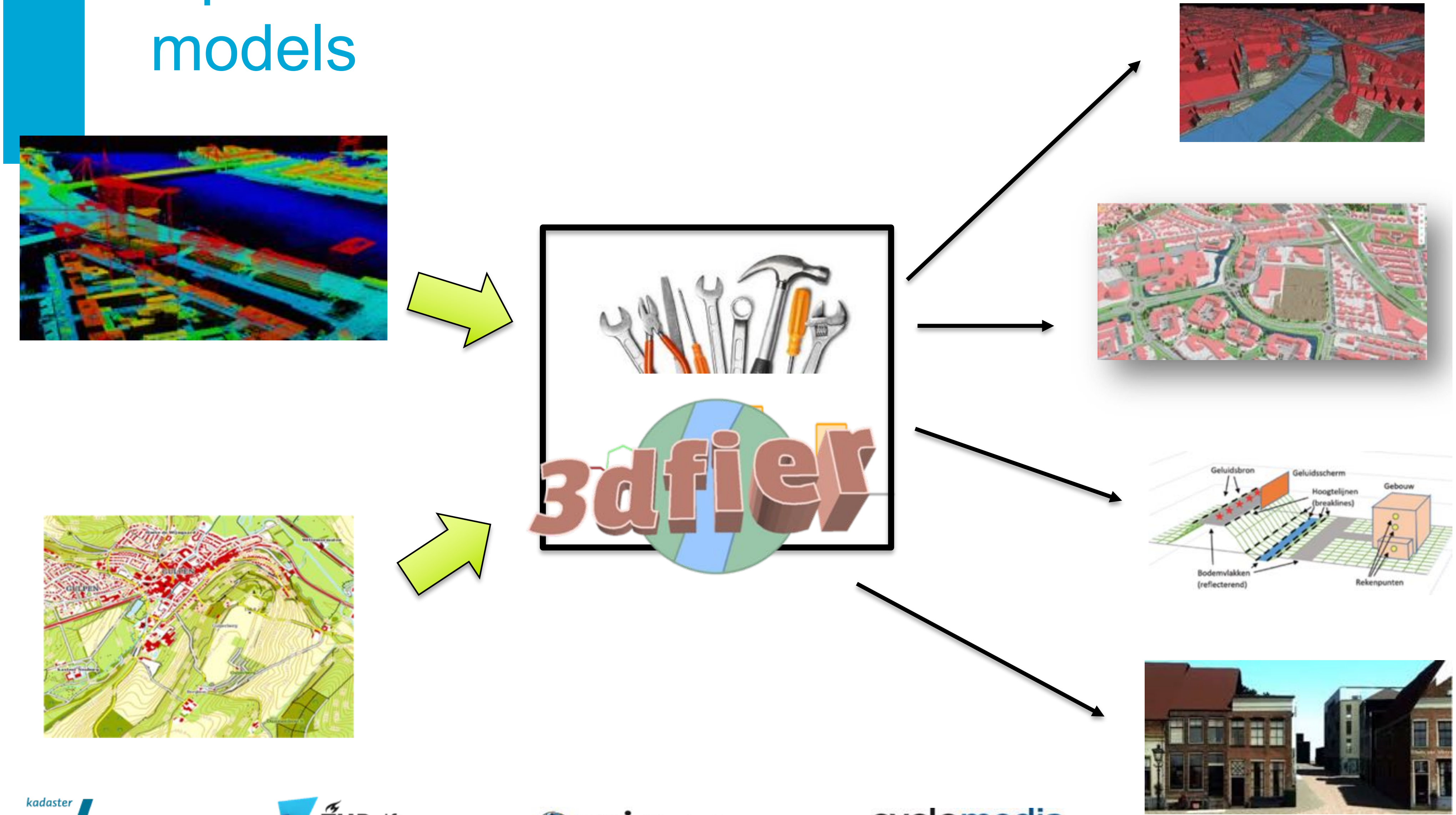
European Environmental Noise Directive



Kluijver & van Tilburg, 2018

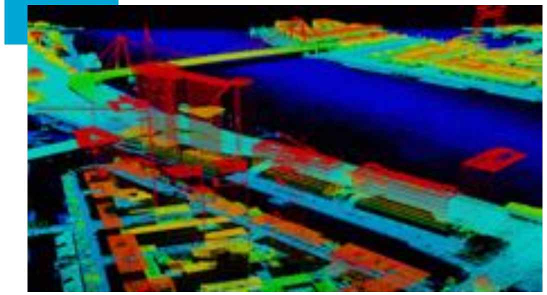
- Most remarkable 3D data requirements:
  - block models are sufficient (max height)
  - block models should model varying heights; even for one footprint
  - height differences with as few height lines as possible (no isolines)

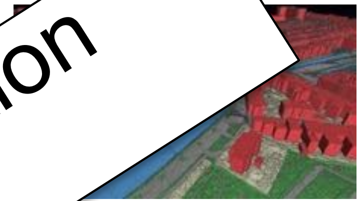
Open software to reconstruct 3D models

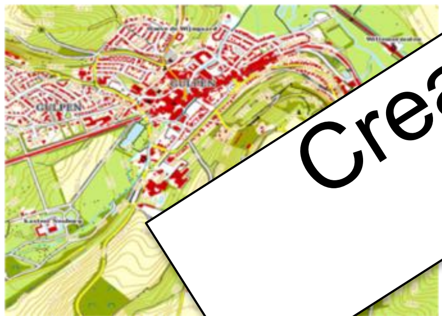


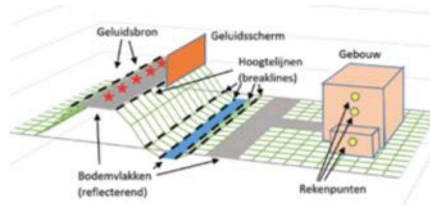
kadaster · TU Delft 3Dgeoinfo · esri Nederland · cyclomedia

Reconstruction of country-wide, application specific 3D data









Creates up-to-date, valid and application specific 3D data

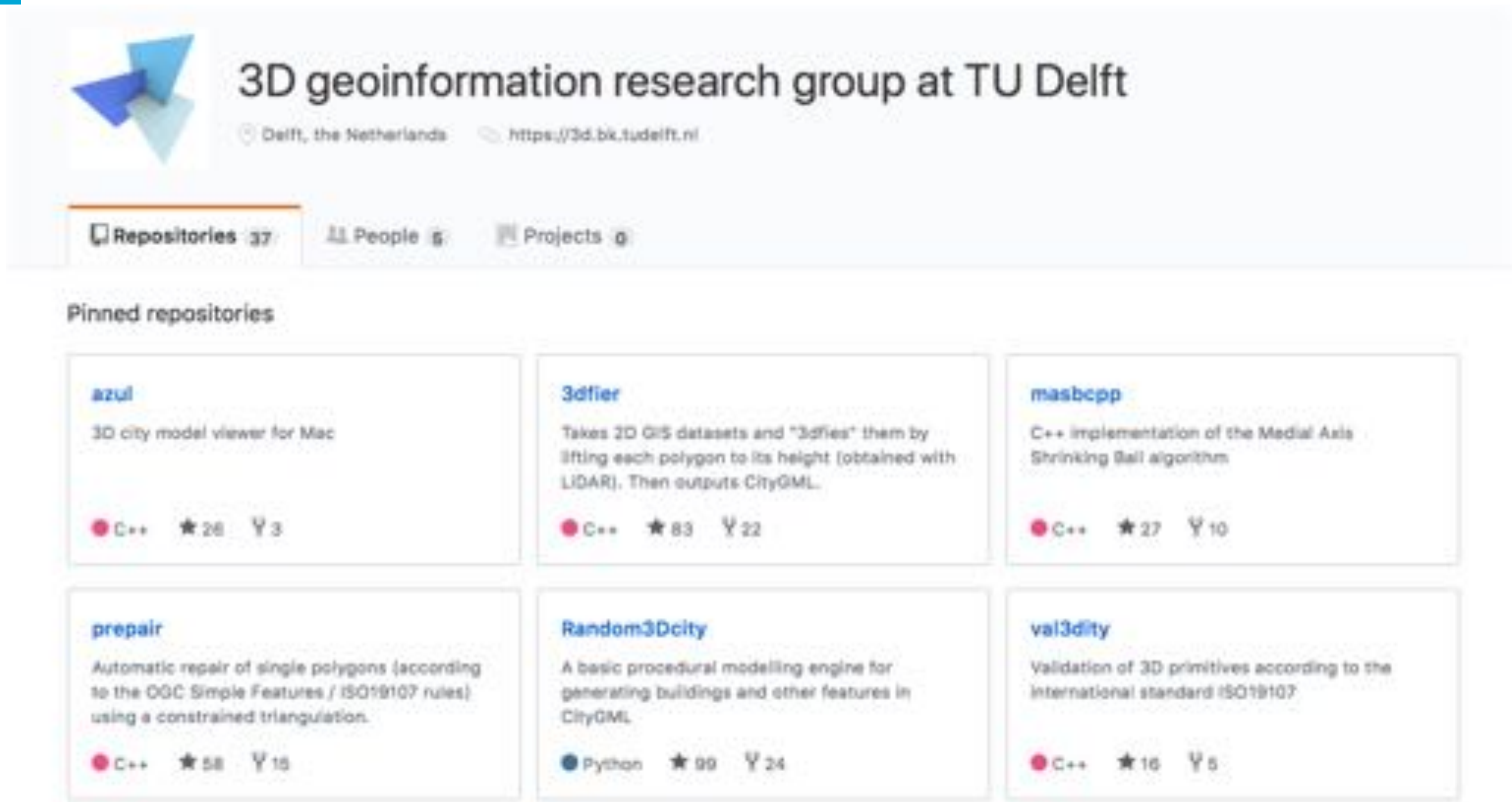
kadaster · TUDelft 3Dgeoinfo · esri Nederland · cyclomedia

Conclusions: bridging the gap between 4D acquisition and 4D applications

- Enables reuse of once captured 3D data models in other applications and software
- Solves current inconsistencies of 3D CM
- Recommendations:
  - Be precise in defining 3D data specifications: “LoD2” is not enough
  - Highest LoD is not always best
  - Different apps need different LoDs (not only buildings)
  - Important to create valid 3D city models

Open source software

github.com/tudelft3d



TU Delft kadaster 3D geoinformation

Thank you!

j.e.stoter@tudelft.nl

For more information, visit 3D.bk.tudelft.nl

Acknowledgements:
Thanks to my colleagues of 3D Geoinformation research group @ Delft University of Technology and the 3D team of Kadaster