LEONHARD
OBERMEYER
CENTER

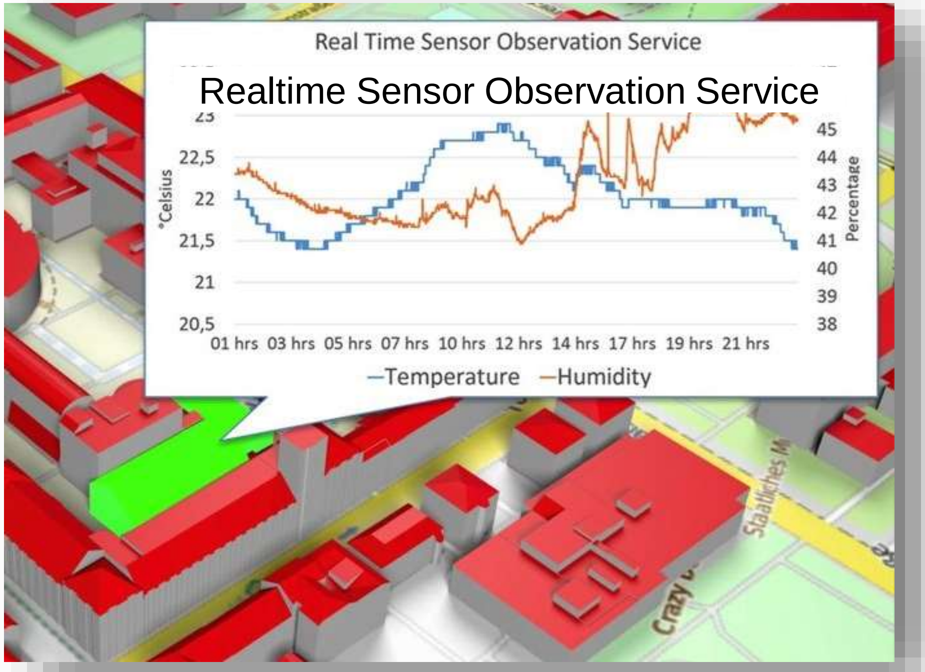
From 3D City Models to Urban Digital Twins

Prof. Dr. Thomas H. Kolbe

Chair of Geoinformatics
Dept. of Aerospace and Geodesy
TUM School of Engineering and Design
Technical University of Munich

https://www.asg.ed.tum.de/gis/

Slovenian Land Surveying Day 2023, Brdo,
15th of Nov. 2023



The Urban Challenge


Image: The Japan Times | KYODO

- More than 50% of all humans live in cities
  - likely to increase to 70% until 2050
- Climate change – energy and environment
  - most energy is consumed in/by cities
  - cities & the urban work and life are responsible for the majority of the emissions of green house gases
  - reduction of emissions urgently required
- Concentration of production and traffic
  - stress of the local environmental (air and water pollution, noise)
  - mobility / traffic flows
- Urban development must consider all these aspects simultaneously – and at the same time getting more efficient

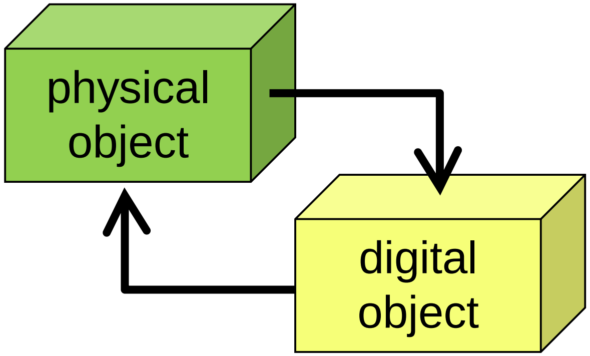
Motivation: Urban planning of the future

- Urban development has always had to take into account **many different aspects**, including
  - economic
  - ecologic
  - energetic
  - mobility
  - urban planning
  - architectural
  - legal
  - sociological
  - security & safety
- **Integrated planning is very difficult** due to the many disciplines involved and the different approaches and information requirements
- **„Urban Digital Twins“** initiatives: Improvement through digital modeling of the relevant real world entities with an explicit bi-directional coupling
  - Promise / hypothesis: basis for the **integrated and well-informed planning and operation of cities**

(Urban)
Digital Twins

Digital Twins

- A **real object** of interest is represented by a **digital counterpart**, the Digital Twin, and data on operating states and changes is accumulated **over the entire life cycle**.
- Concept developed in the context of **Industry 4.0**
- Development / implementation started over 15 years ago
  - first in the fields of space flight, aerospace, mechanical engineering, energy technology, automotive engineering
- A digital twin is connected to its real, physical counterpart via sensor observations and actuators
  - in Industry 4.0 via the **Internet of Things (IoT)**



Digital Twins in Industry 4.0

The digital twin for each specimen of a turbine comprises

- Model data, manufacturing specific data
- Logs of all measurements recorded during operation (operating and performance data)
- Maintenance reports

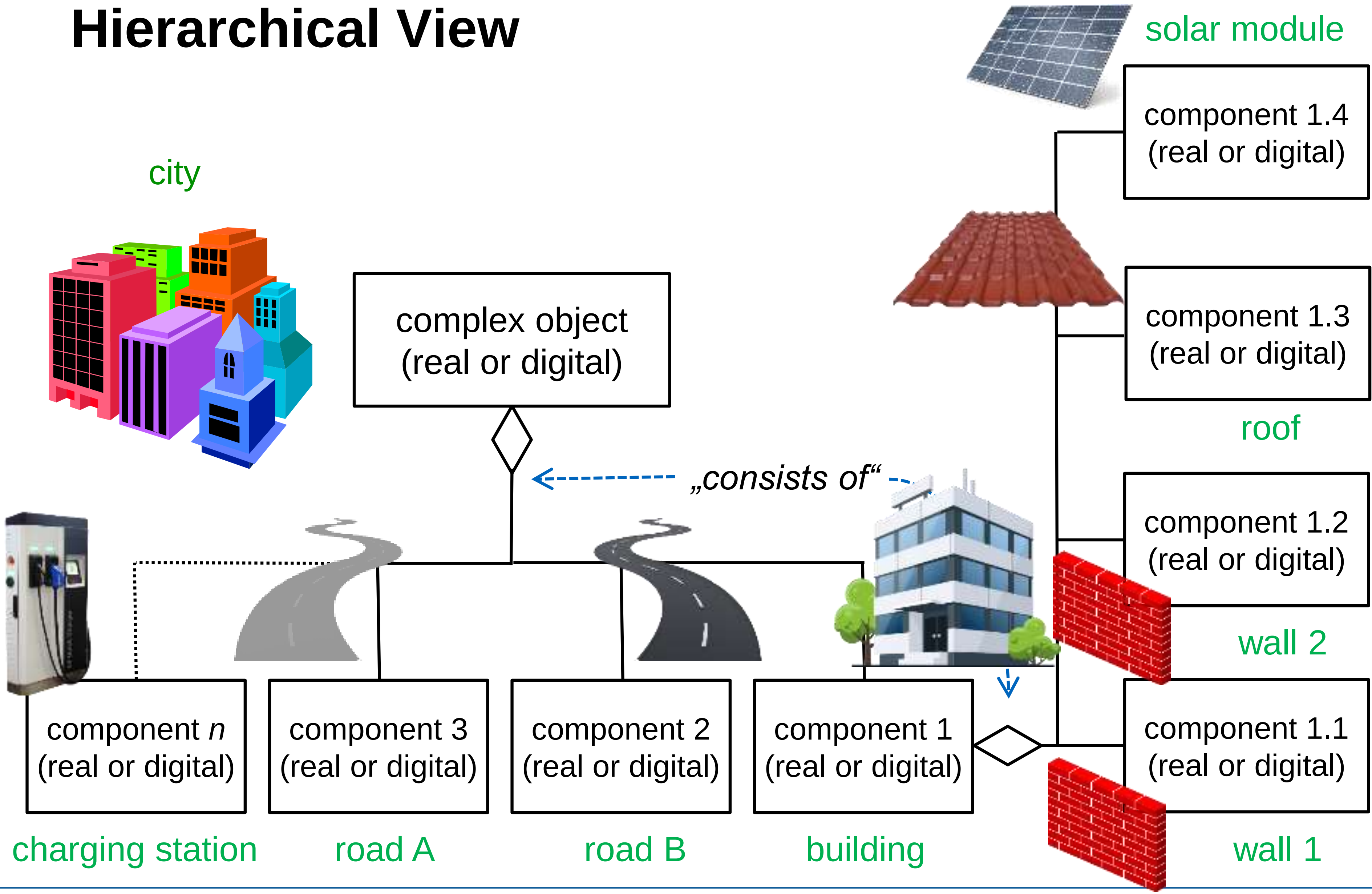
Image source: https://www.ibm.com/blogs/internet-of-things/iot-digital-twin-revolution/

Imagesource: https://new.siemens.com/global/de/branchen/windenergie/equipment/digitalisierung.html

Benefits of Digital Twins

- Comprehensive **cumulative documentation** of real objects
- Calculation / **estimation of the system state** and **system performance**
  - Derivation of indicators based on the data of the digital object
- **Simulation** of real systems on the basis of digital objects
- Applications
  - **Monitoring and control** of physical objects and systems
  - Hazard detection, **detection of exceptional conditions**
  - Operation optimization, **predictive maintenance**
- **"What-if" scenarios**
  - Modification of the digital twin according to the scenarios and estimation of the impact on the system

Hierarchical View



Spatial Digital City Models (→3D City Models)

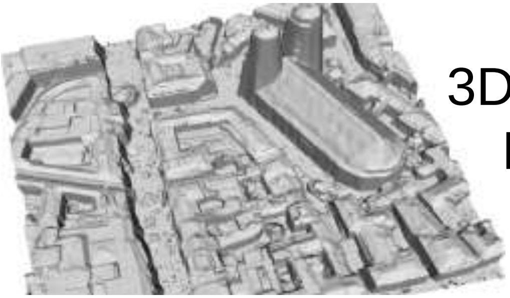
- most **relevant urban objects are physical things**
- physical things **occupy space in the real world**
  - partitioning of the occupied real space → discrete objects
  - spatial properties: place (location & orientation), extent, shape
- **space as a unifying organizing principle** for urban information
- large-scale, area-wide **recording of the spatial configuration** possible **using sensors**
  - from satellites, airplanes, UAVs, vehicles, persons with cameras, laser scanners, tachymeters
  - automatic extraction of urban objects and decomposition into detailed sub-objects
- **intuitive visualization and exploration**
  - 3D depiction is more intuitive for people than 2D maps

Digital 3D Models of the City

- There are different types, for example:

Semantic
3D City Models
e.g. CityGML

Building
Information
Modeling
e.g. IFC



3D Mesh
Models

3D Point
Clouds

- All have certain advantages and disadvantages, and cities nowadays employ more than one type to compensate for the weaknesses of the others

Spatio-Semantic City Modeling

3D Decomposition of Urban Space

- City is decomposed into meaningful objects with clear semantics and defined spatial and thematic properties
  - buildings, roads, railways, terrain, water bodies, vegetation, bridges
  - buildings may be further decomposed into different storeys (and even more detailed into apartments and single rooms)
  - domain specific data are attached to the objects as thematic attributes

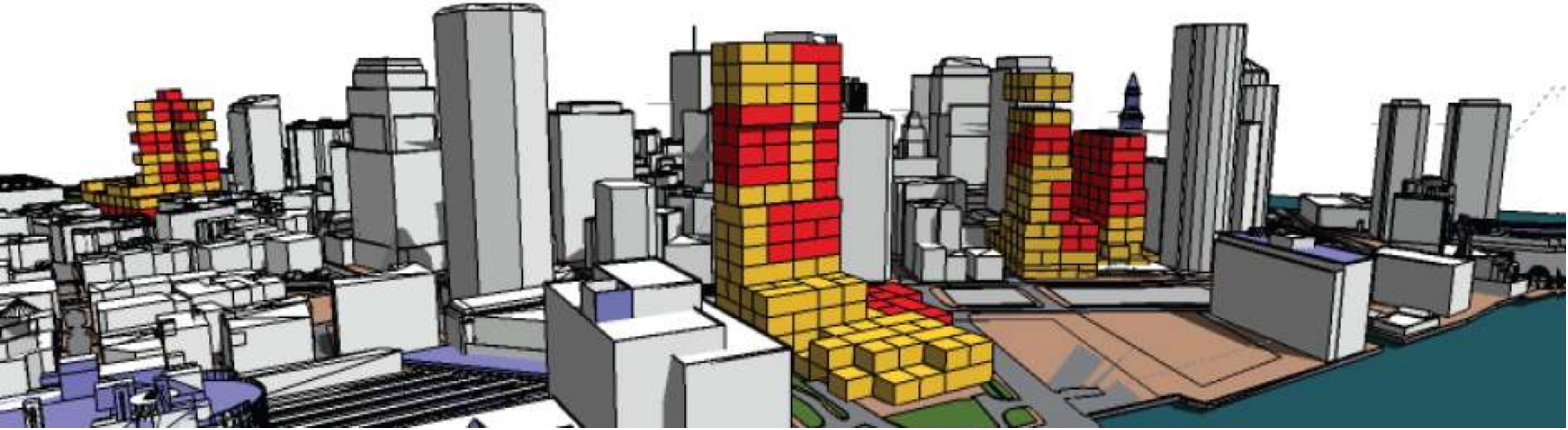
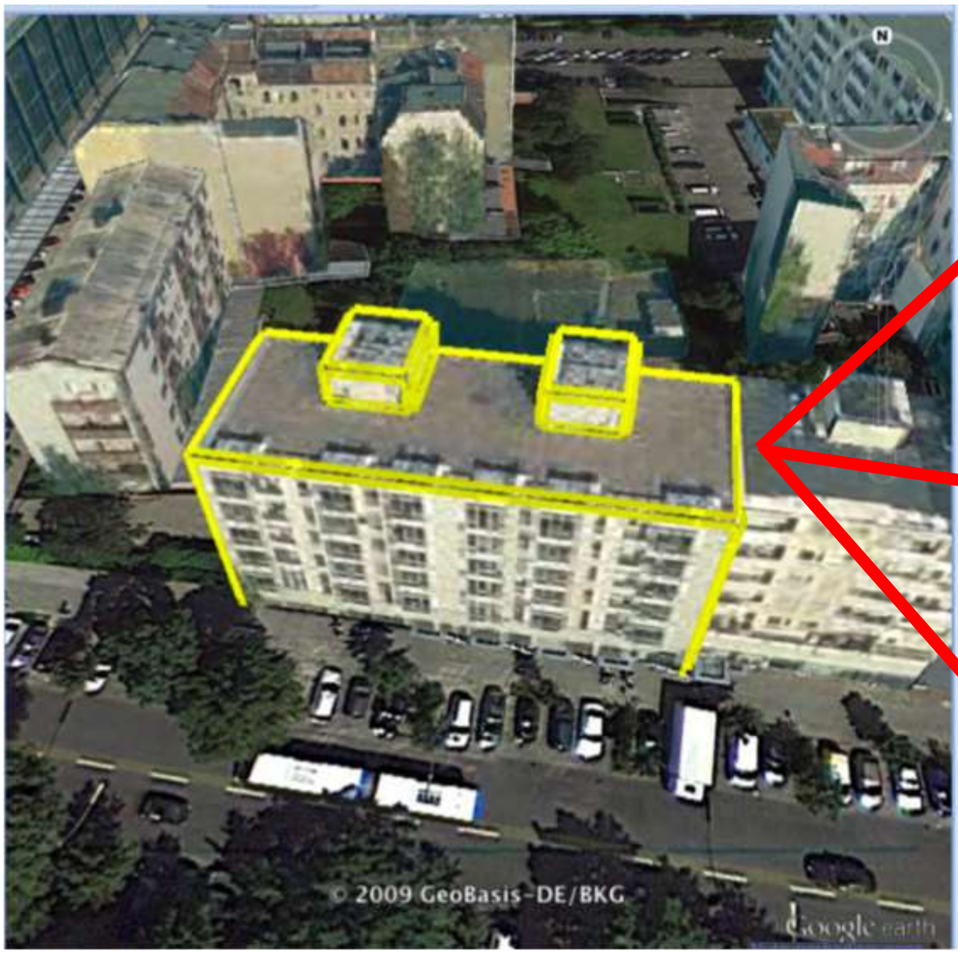


Image: Paul Cote, Harvard Graduate School of Design

Information Integration via 3D City Model Objects



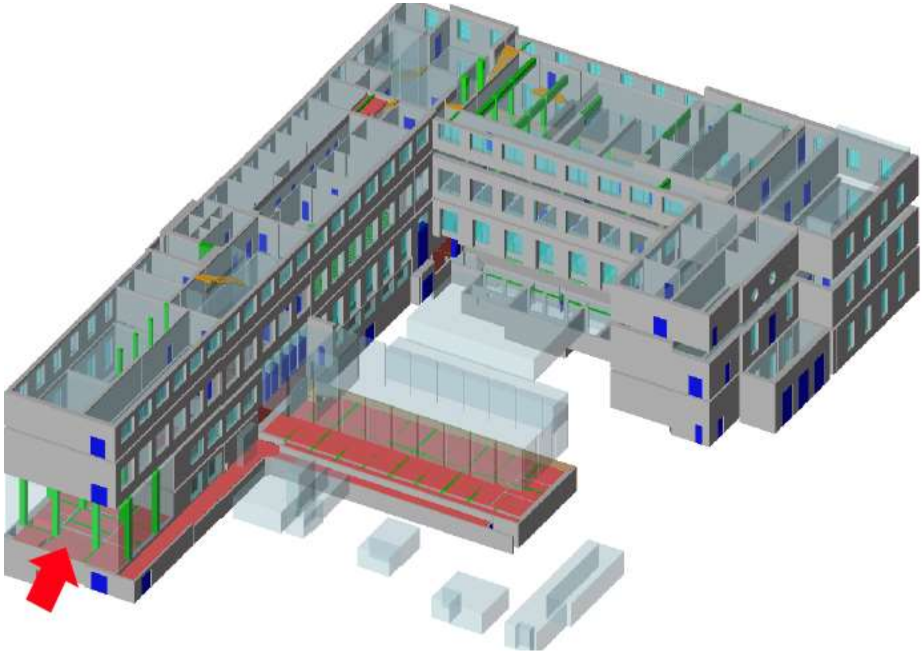
Energy
Heat energy demand
Energy demand for warm water
Electric power consumption

Noise immission
Noise levels on the facade
Number of inhabitants

Economy
Assessed real estate value
Available rental support

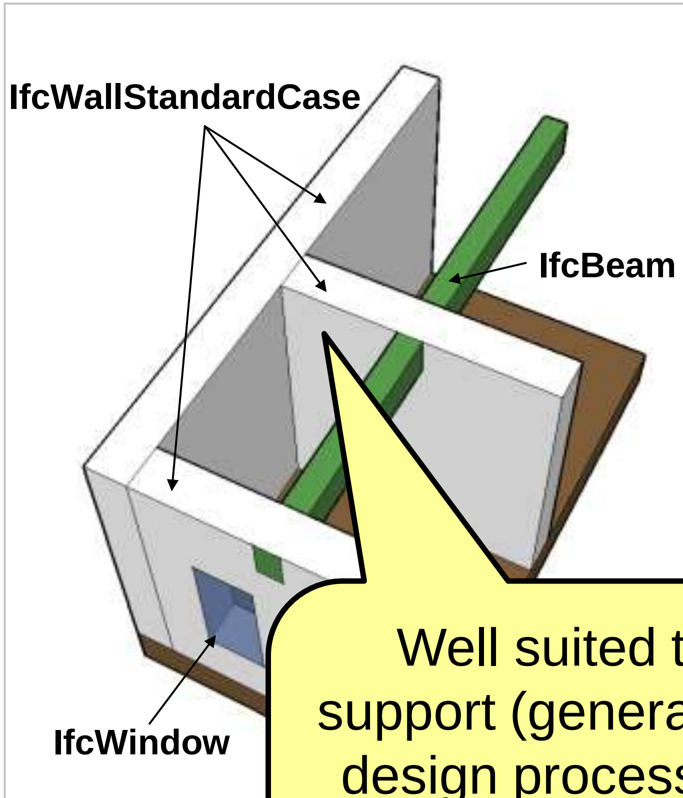
Semantic 3D Models of the Built Environment

- On the scale of individual sites:
  Building Information Modeling (BIM)
- On the scale of city quarters up to entire regions:
  Semantic 3D City Models (*Urban Information Models*)



Differing Object Modeling Paradigms

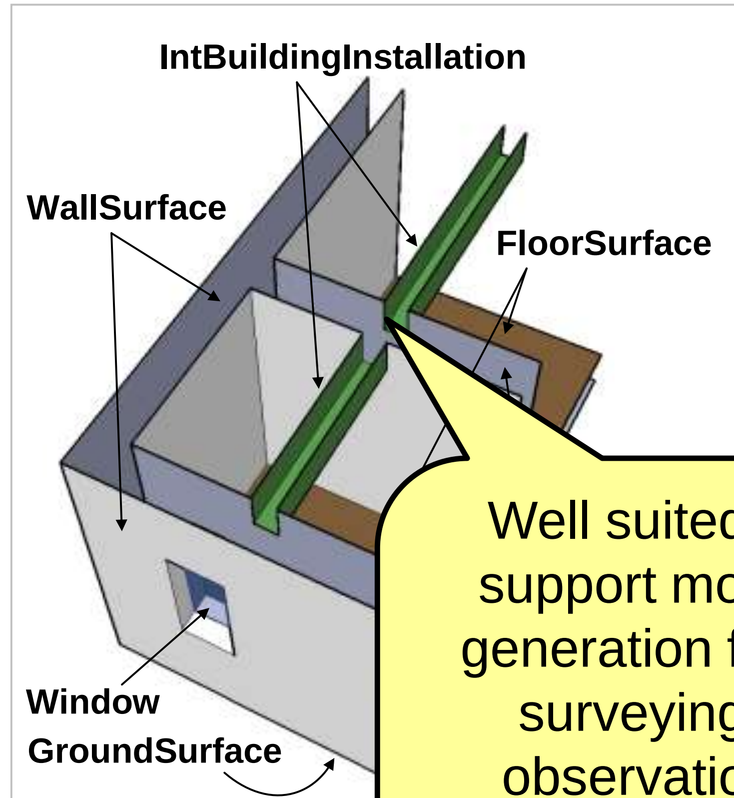
BIM (e.g., IFC)
Constructive Solid Geometry



Volumetric, p
representing th
o

Well suited to support (generative) design processes, hence, to create **models for planned objects**

3D GIS (e.g., CityGML)
Boundary Representation



Accumulation of obs
topograph

Well suited to support model generation from surveying / observation, hence, to create **models for existing objects**

Standardized Access to Semantic City Models

Mapping the state of a city at time t_i → CityGML ← Virtually carrying out planned actions by changing the city model accordingly

- Energy Demand & Production Estimation
- Noise Immission Simulation & Mapping
- Real Estate Management & Urban FM
- Vulnerability Analysis & Disaster Management

CityGML

City Geography Markup Language – CityGML

Application independent Geospatial Information Model

for semantic 3D city and landscape models

- comprises **different thematic areas** (buildings, vegetation, water, terrain, traffic, tunnels, bridges etc.)
- **Data model (UML) + Exchange format** (based on GML3)

CityGML represents

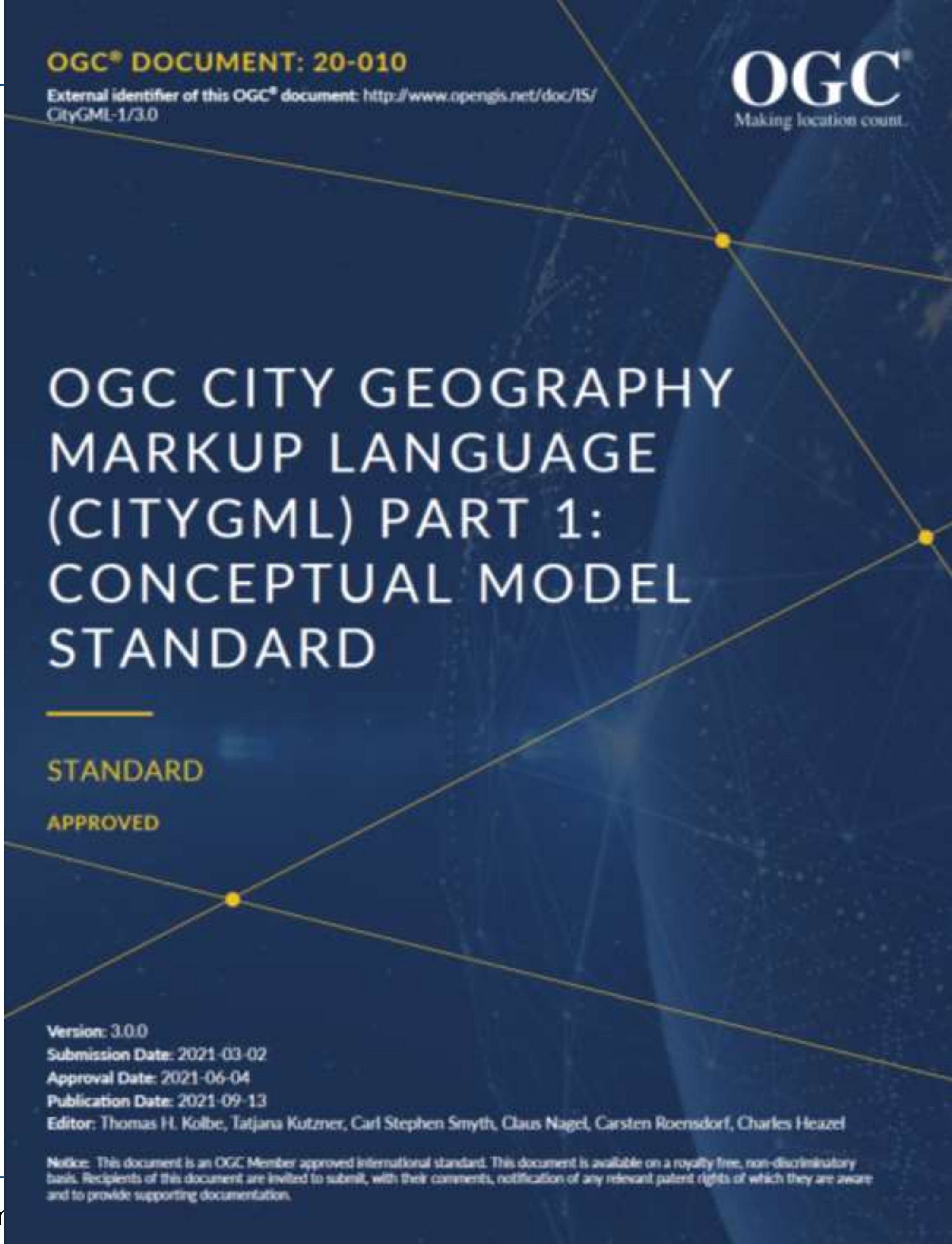
- 3D geometry, 3D topology, semantics, and appearance
- in 5 discrete scales (Levels of Detail, LOD)

International Standard of the Open Geospatial Consortium

- Version 2.0.0 was issued in 3/2012
- **Version 3.0 was issued in 2021 (Part 1) and 2023 (Part 2)**

The recent version CityGML 3.0

- On 13 September 2021, after eight years of development, the OGC published version 3.0 of the international standard CityGML
- The new version can be downloaded here: https://docs.ogc.org/is/20-010/20-010.html


OGC® DOCUMENT: 20-010

External identifier of this OGC® document: http://www.opengis.net/doc/IS/CityGML-1/3.0

OGC
Making location count.

OGC CITY GEOGRAPHY MARKUP LANGUAGE (CITYGML) PART 1: CONCEPTUAL MODEL STANDARD

STANDARD

APPROVED

Version: 3.0.0
Submission Date: 2021-03-02
Approval Date: 2021-06-04
Publication Date: 2021-09-13
Editor: Thomas H. Kolbe, Tatjana Kutzner, Carl Stephen Smyth, Claus Nagel, Carsten Roensdorf, Charles Heazel

Notice: This document is an OGC Member approved international standard. This document is available on a royalty free, non-discriminatory basis. Recipients of this document are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

Five (respectively four) defined Levels of Details

LOD0 – Regional, landscape model **+ interior**
- 2.5D Digital terrain model

LOD1 – City, regional model **+ interior**
- Prismatic buildings without roof structures

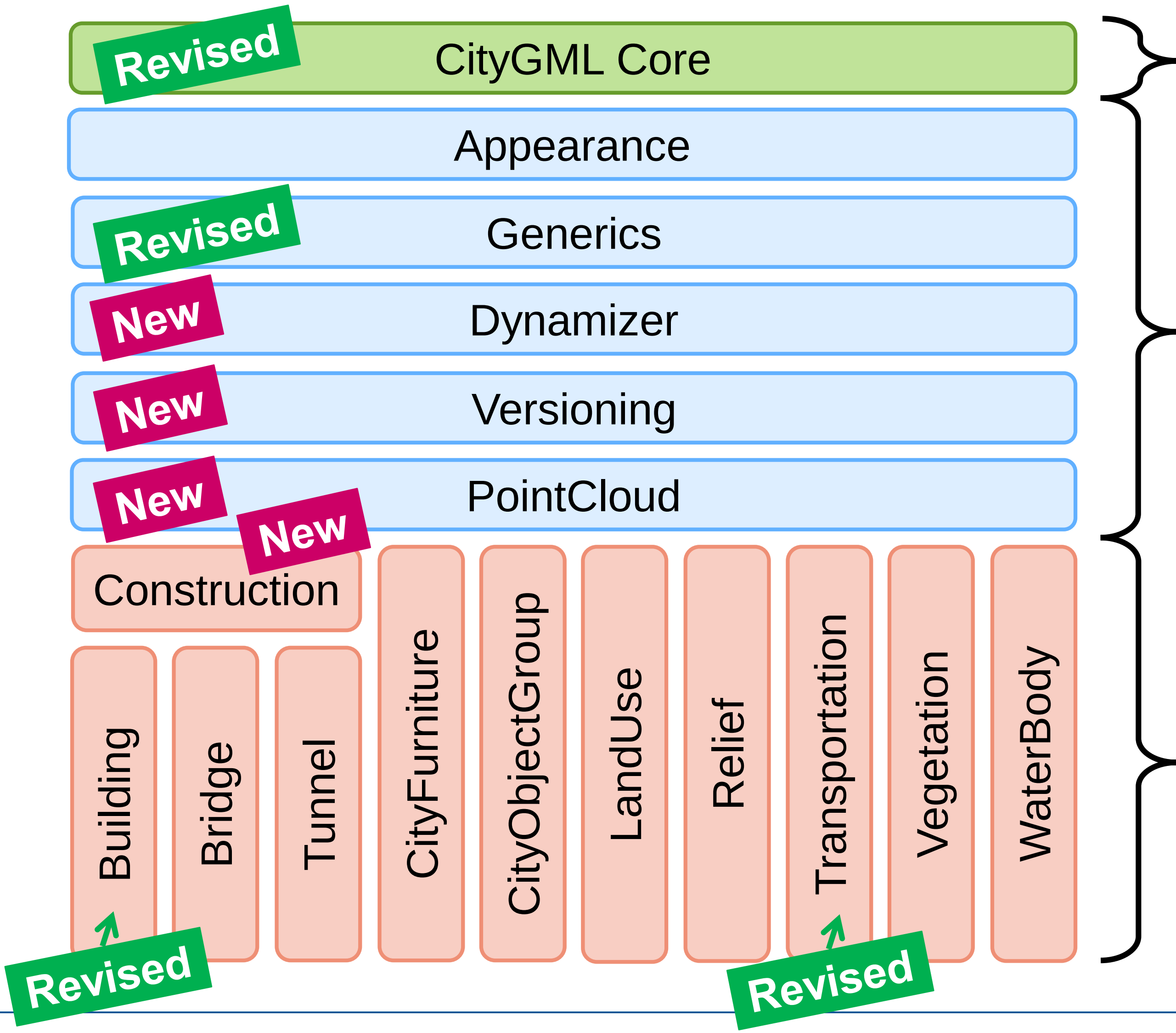
LOD2 – City districts, site model **+ interior**
- Simple buildings with detailed roof structures

LOD3 – Architectural models (exterior) **+ interior**
- Detailed architectural models

LOD4 – Archi tura models (interi r)
- “Walkable” arc itectu models

- CityGML 3.0 allows for representing the interior of buildings, tunnels and bridges in LODs 0-3 as well.
  → E.g., the exterior can now be modelled in LOD1, whereas the interior is represented in LOD2 or 3
- Supports the use of 3D city models in applications which require detailed representations of the indoor, but not necessarily of the outdoor, e.g. indoor navigation, energy applications, smart homes

CityGML 3.0 Module Overview

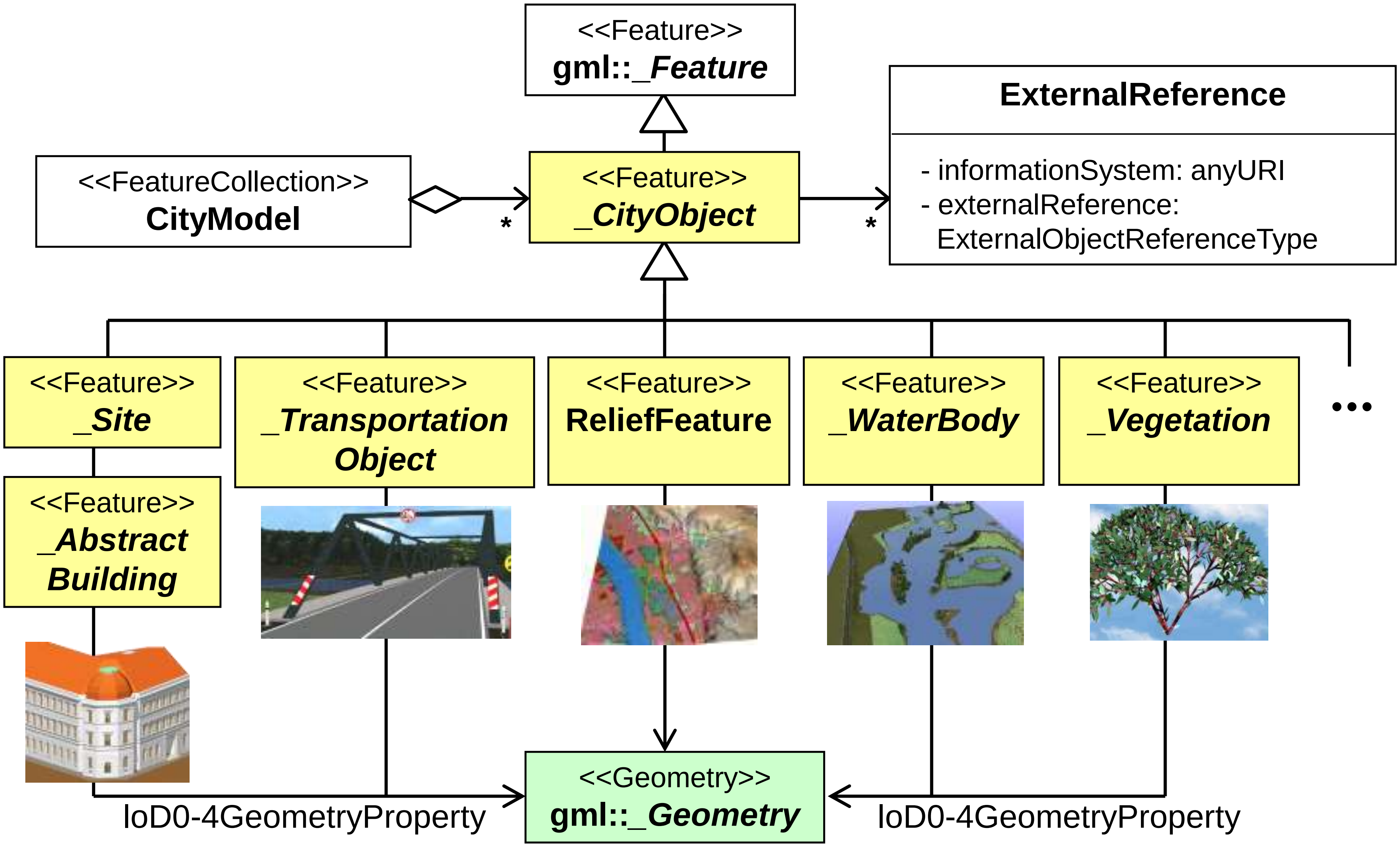


Defines base concepts and geometries inherited by all other modules

Defines concepts applicable to all thematic modules

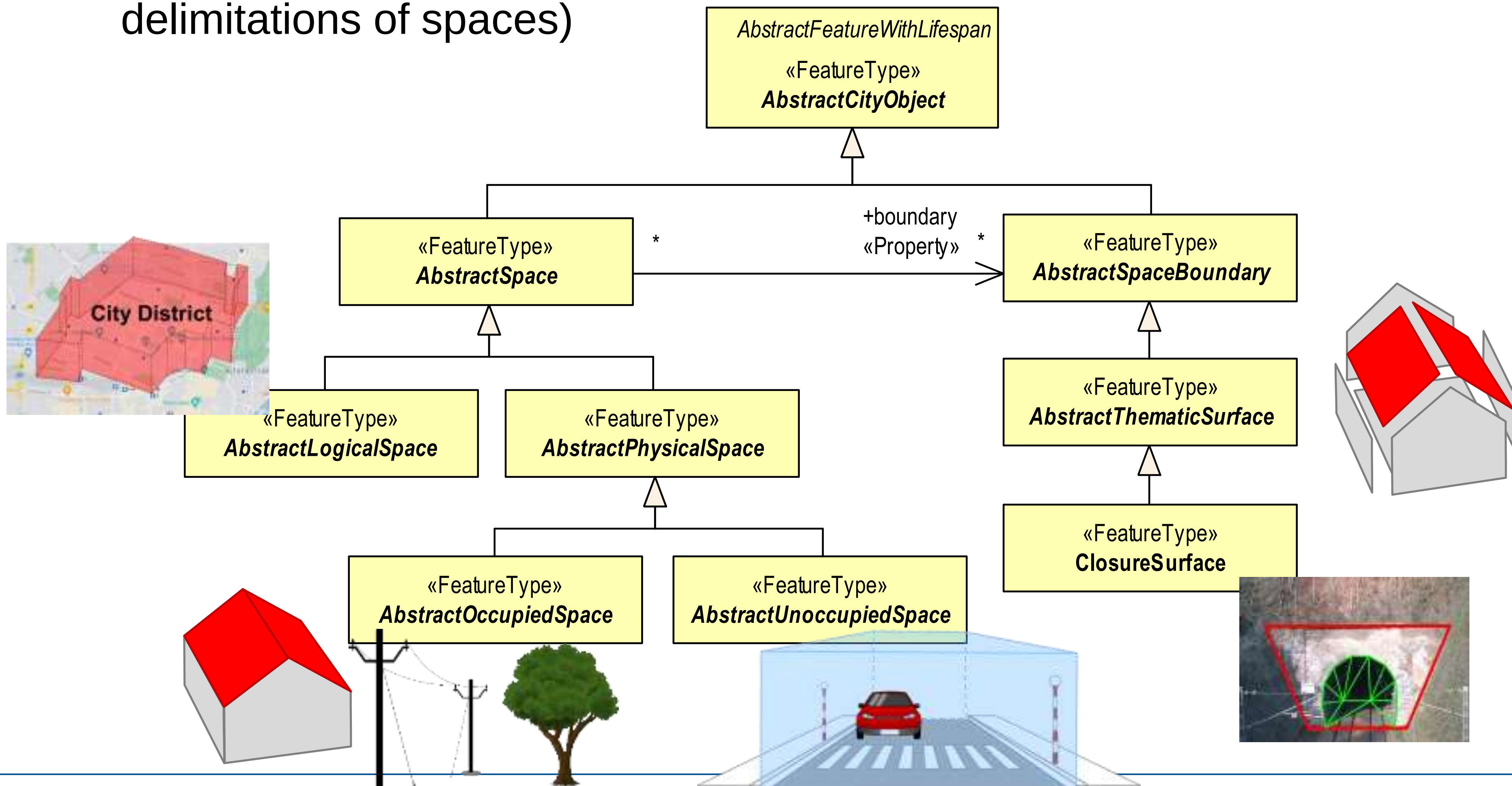
Thematic modules

Thematic Modeling in CityGML



CityGML 3.0: Introduction of a Space Concept

- All thematic objects are now either categorized as **spaces** (objects with volumetric extent in the real world) or **space boundaries** (areal delimitations of spaces)



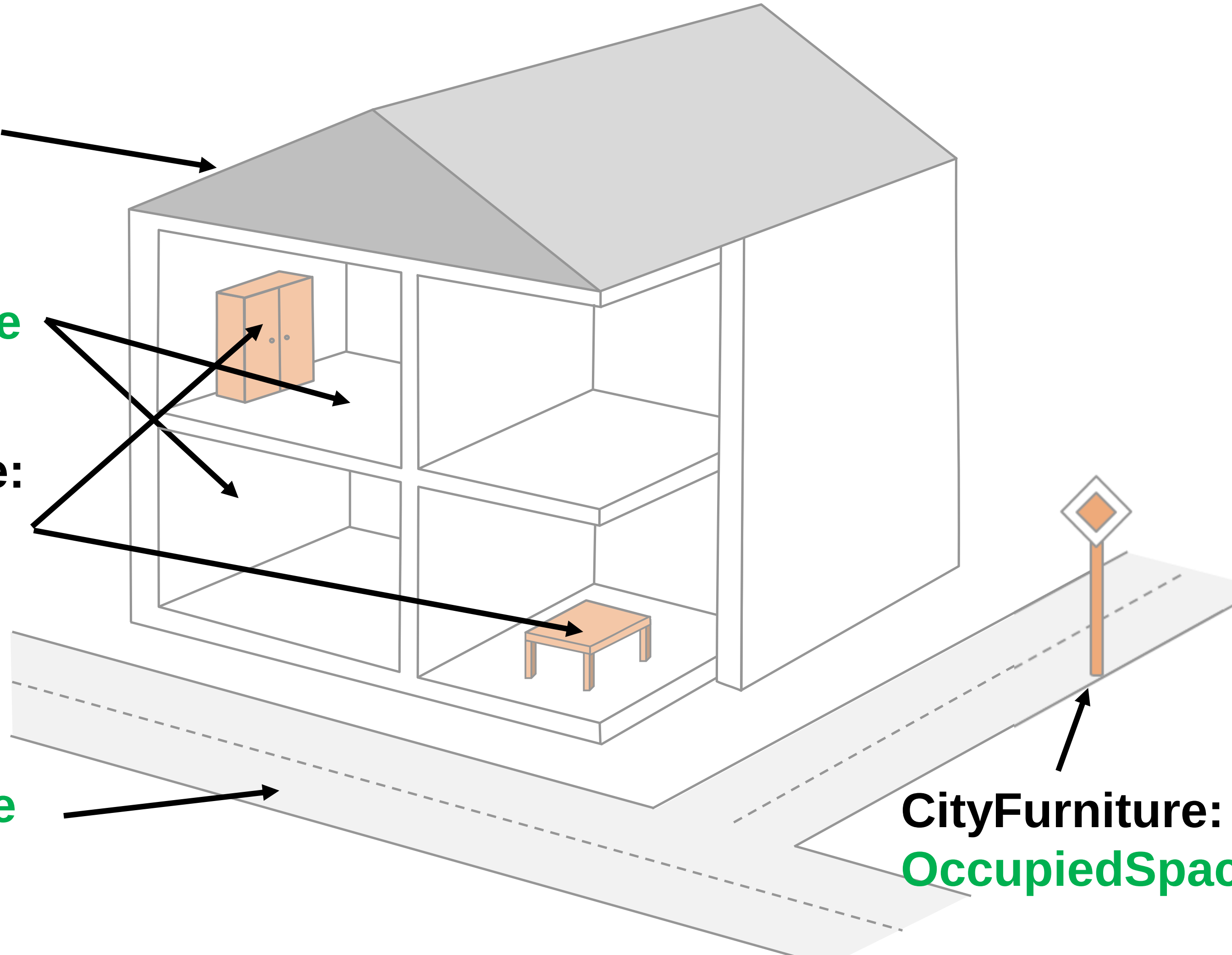
Examples for the Space Concept

Building:
OccupiedSpace

BuildingRoom:
UnoccupiedSpace

BuildingFurniture:
OccupiedSpace

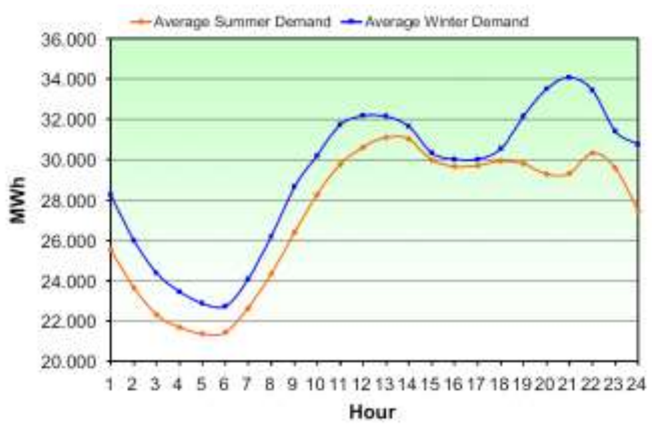
Road:
UnoccupiedSpace



CityFurniture:
OccupiedSpace

Changes in the context of semantic 3D city models – Highly dynamic changes

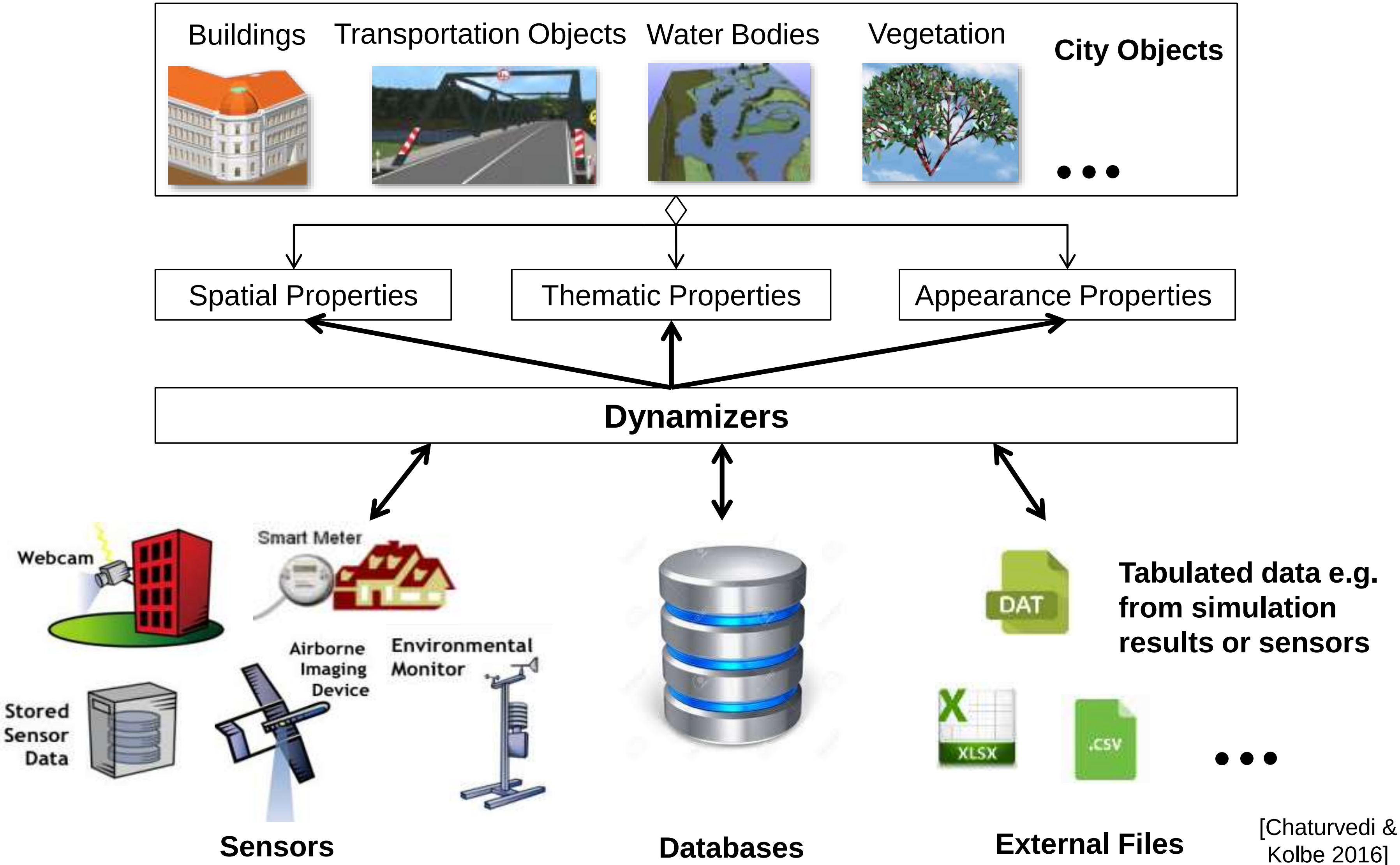
- **Variations of spatial properties**: change of a feature's geometry, both in respect to shape and to location (e.g. moving objects)
- **Variations of thematic attributes**: changes of physical quantities like energy demands, temperatures, solar irradiation
- **Appearance**: e.g. raster images showing air quality
- **Variations with respect to sensor or real-time data**



Source:C. García-Ascanio and C. Maté, "Electric power demand forecasting using interval time series: A comparison between VAR and iMLP,"*Energy Policy*

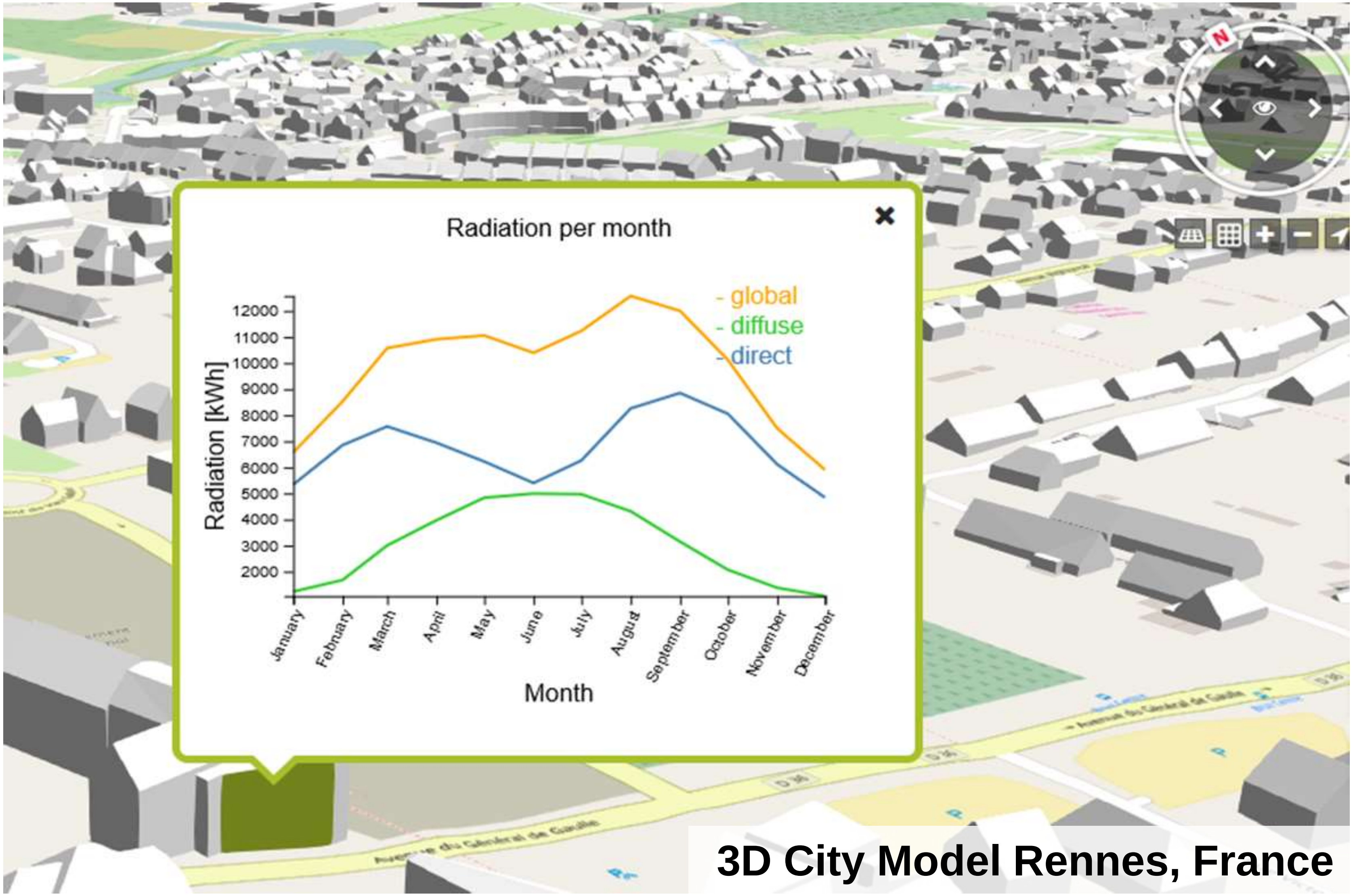
Source: MOREL M., GESQUIÈRE G., "Managing Temporal Change of Cities with CityGML". In UDMV (2014)

New CityGML 3.0 Dynamizer module



[Chaturvedi & Kolbe 2016]

Visualization of Time-dependent Solar Potentials



CityGML 3.0 Streetspace Modeling Example 1

- Streetspace model of Ingolstadt automatically generated from OpenDRIVE data
- Open-source OpenDRIVE to CityGML 2/3 converter r:trån
https://github.com/tum-gis/rtron

CityGML 3.0 Streetspace Modeling Example 2

- Project: Urban Digital Twin Munich
- Representation of the streetspace and dynamic streetspace activities

CityGML Usage (mostly Version 1.0/2.0 so far)

- CityGML is already successfully used at national scale
  - The official national and municipal 3D geoinformation standards of Germany and the Netherlands base on CityGML 1.0/2.0
  - Japan published 3D city models for >100 cities based on CityGML and the i-Urban Revitalization Application Domain Extension
- Many cities worldwide use CityGML for their 3D city models
- List of Open CityGML Datasets: https://github.com/OloOcki/awesome-citygml
  - from 18 countries, in different Levels of Detail, most data are officially maintained 3D city models
- CityGML extensions like the Energy ADE and the Noise ADE are used internationally

Applications of Urban Digital Twins

Applications – 4D Traffic Simulation Exploration

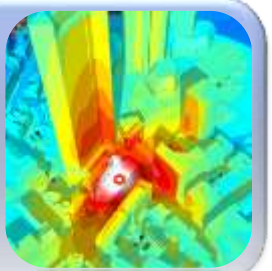
Urban Simulations using 3D City Models: Situation Assessment + 'What-if' Scenarios

Flooding

River flooding
coastal flooding
Tsunami

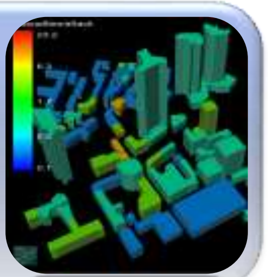
Blast

Detonation
pressure wave
propagation



Vulnerability

Risk-based
vulnerability
assessment



Crowd

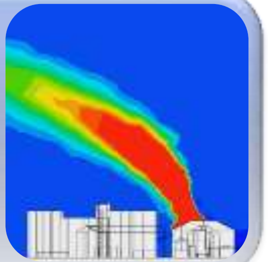
Crowd dynamics
for events and
evacuations

Wind Field

Turbulent wind
fields and thermal
stratification

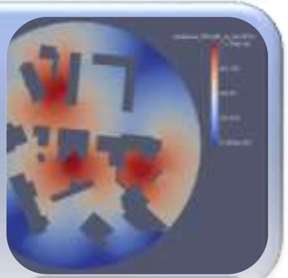
Air Quality

Smoke and
pollution
propagation



Acoustics

3D urban
noise
protection



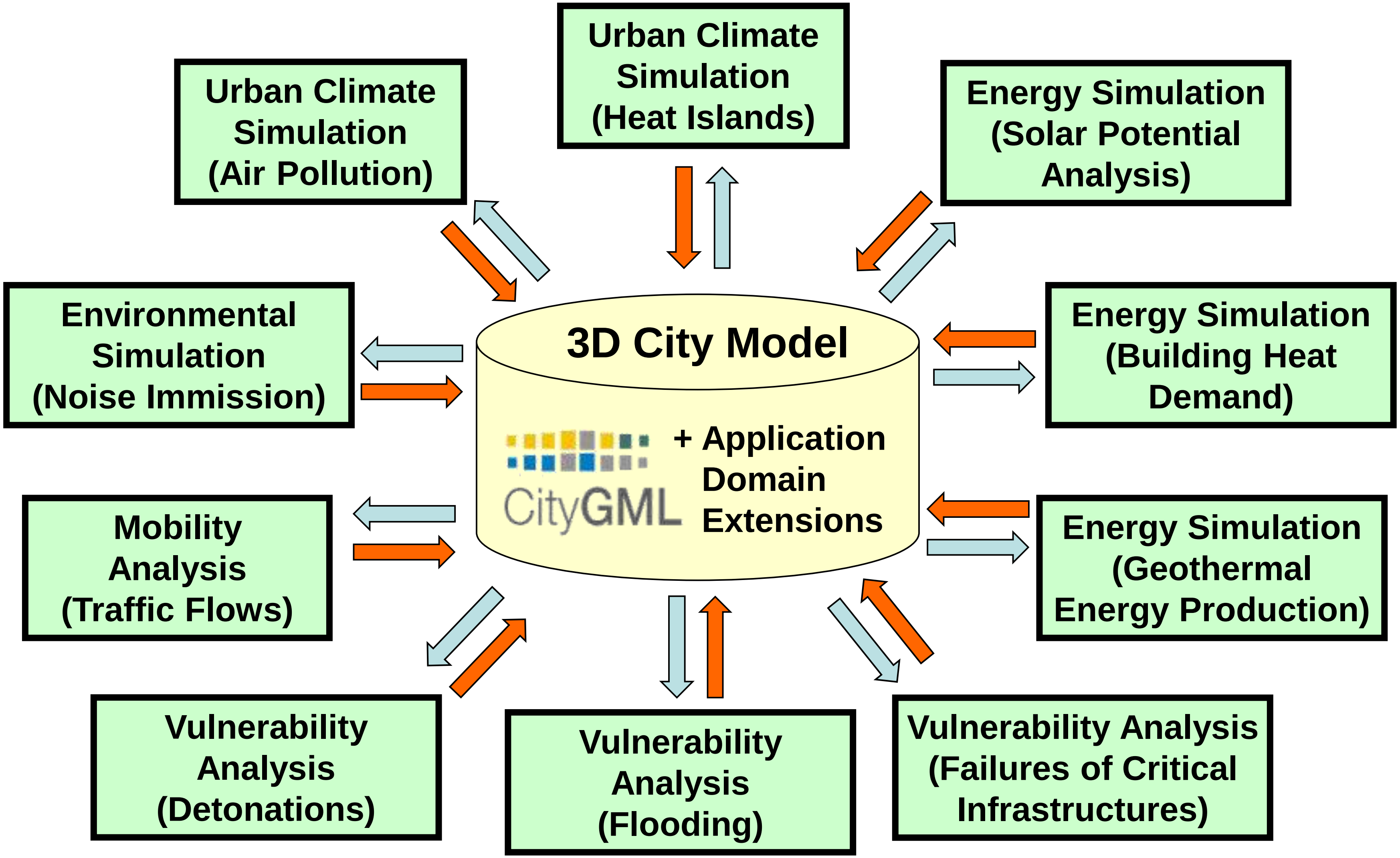
Mobility

Autonomous
driving, traffic
and wireless
communication

Source:

virtualcitySYSTEMS CADFEM

Multi-Simulations with 3D City Models



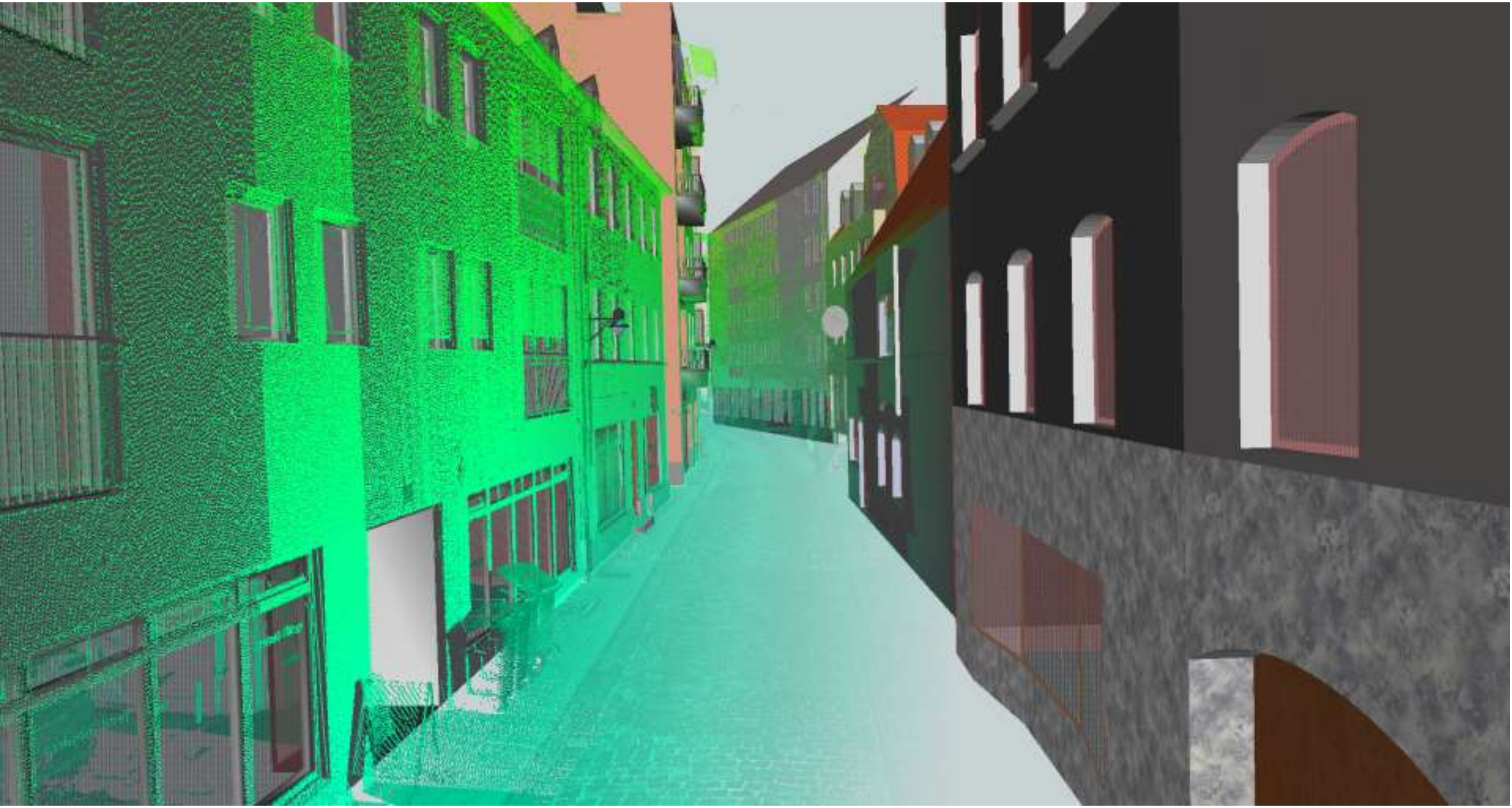
Projects that were carried out by or with participation of my teams so far

Example from current Research Activities

Application: Autonomous Driving

- We use semantic 3D city models to develop realistic simulation environments for the development and testing of autonomous cars
- How?
  - exactly match real sensor readings from RADAR, laser scanning & cameras of cars with city model objects and analyze the signal responses for the different object categories
  - learn patterns from the matched signal responses and draw conclusions on the material types of the different city objects
  - use this information to simulate sensor responses for different car types & sensor configurations and/or for different cities
- Side effect: enrichment of city models by materials

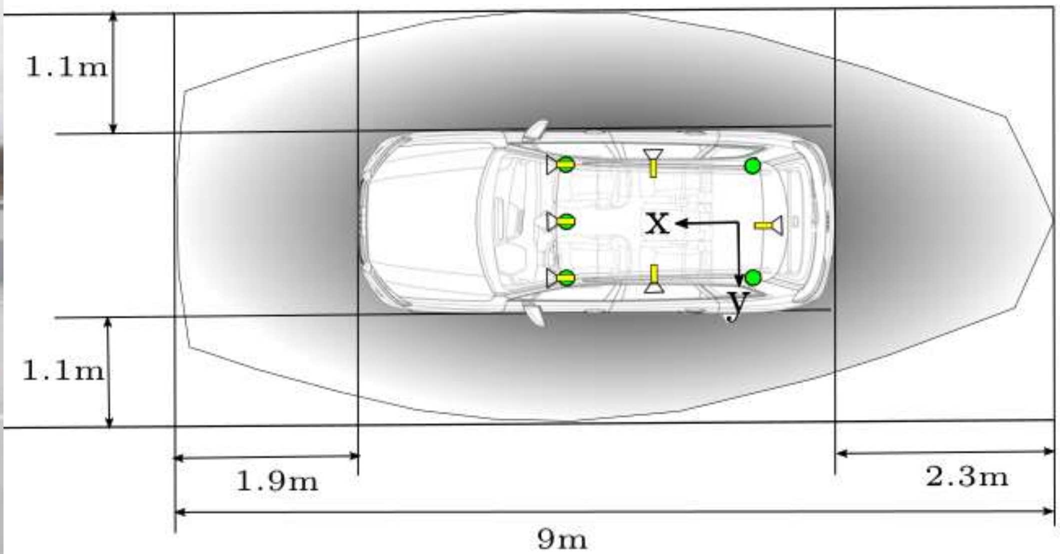
3D Point Cloud → CityGML LOD3 model



- (Manual work so far) Result: highly accurate 3D city model

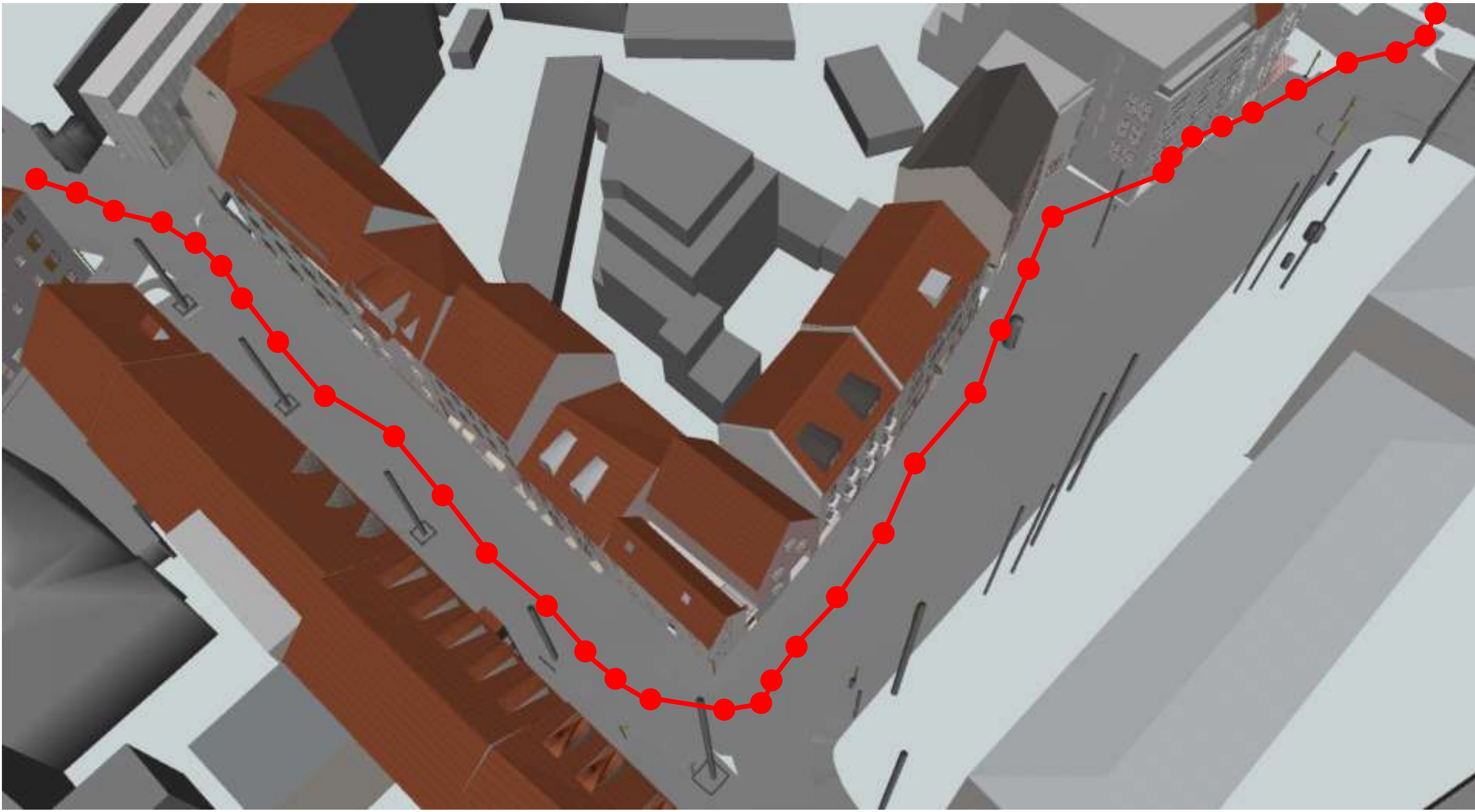
Sensor data from an autonomously driving car

Audi Q7 e-tron with multi-modal sensor setup

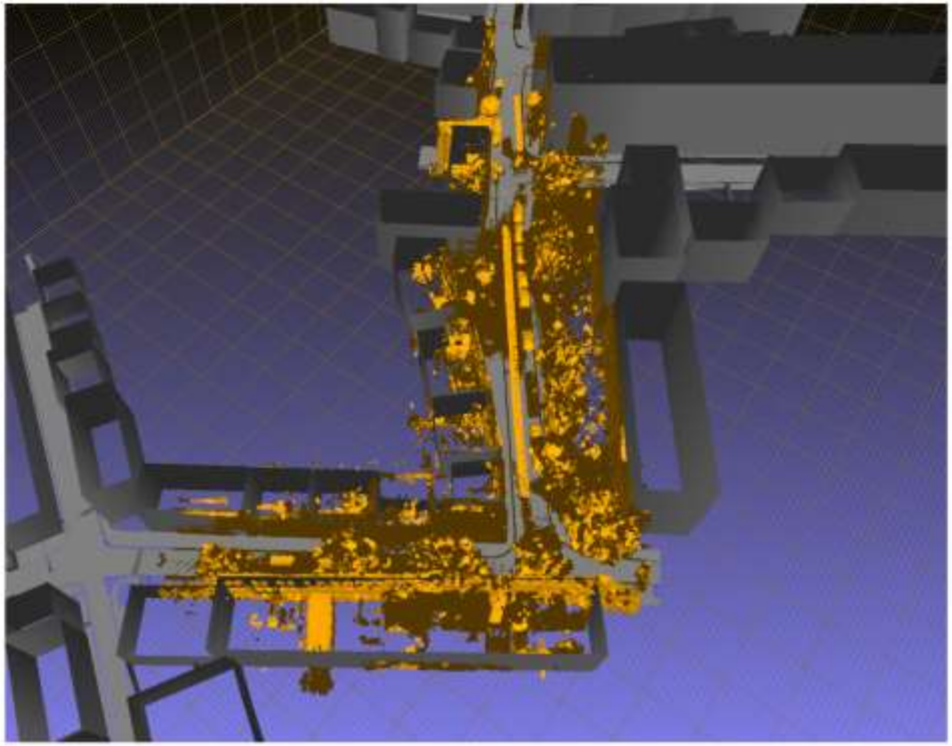


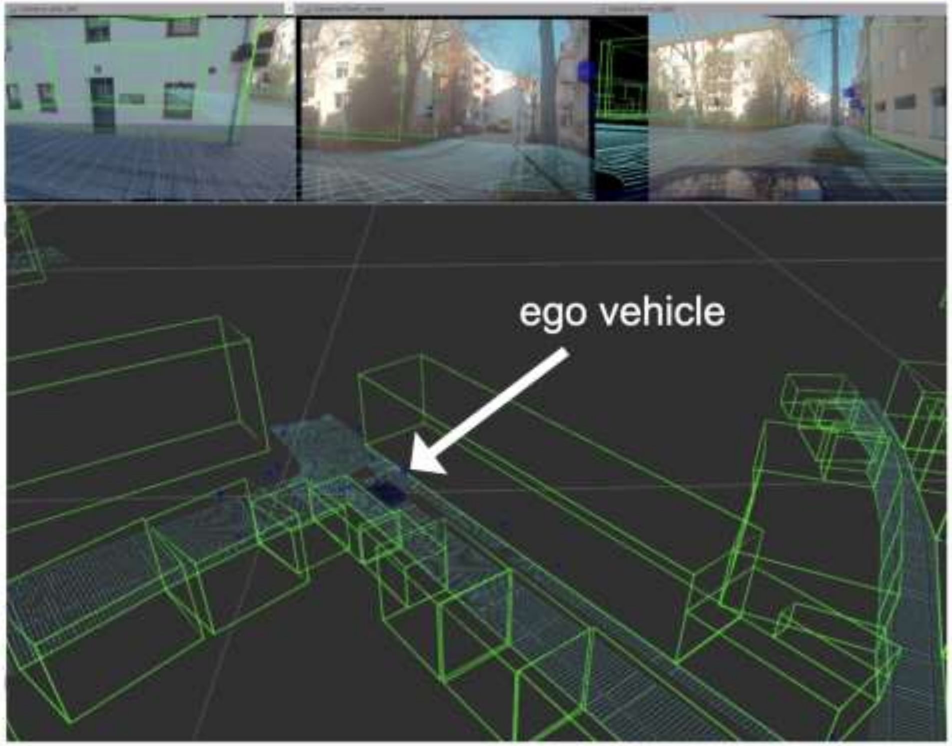
Sensor setup: 6 cameras (yellow) and
5 LiDAR sensors (green)

Taking a Ride: Raw GPS trajectory



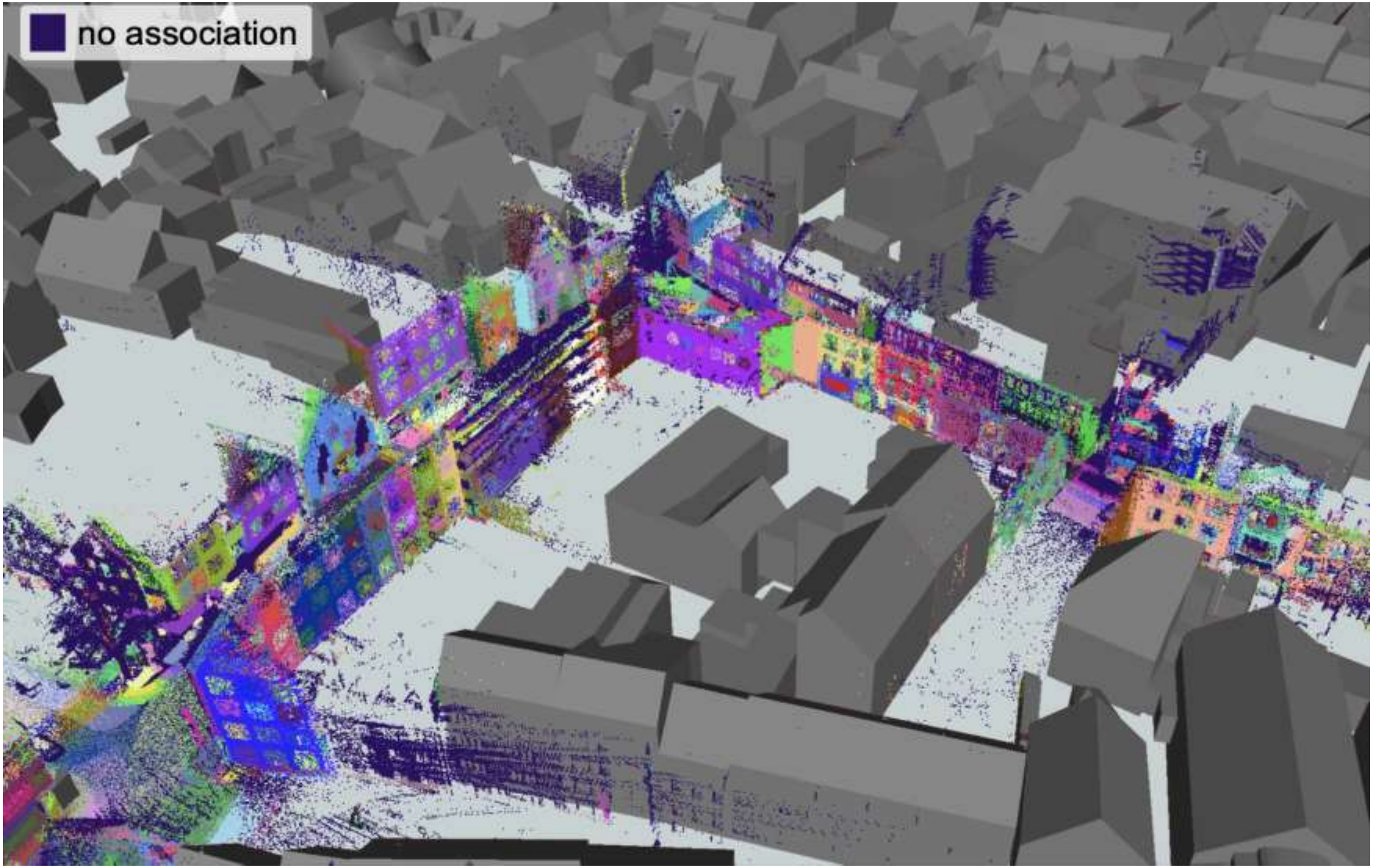
Precise Georeferencing of Car Sensor Data



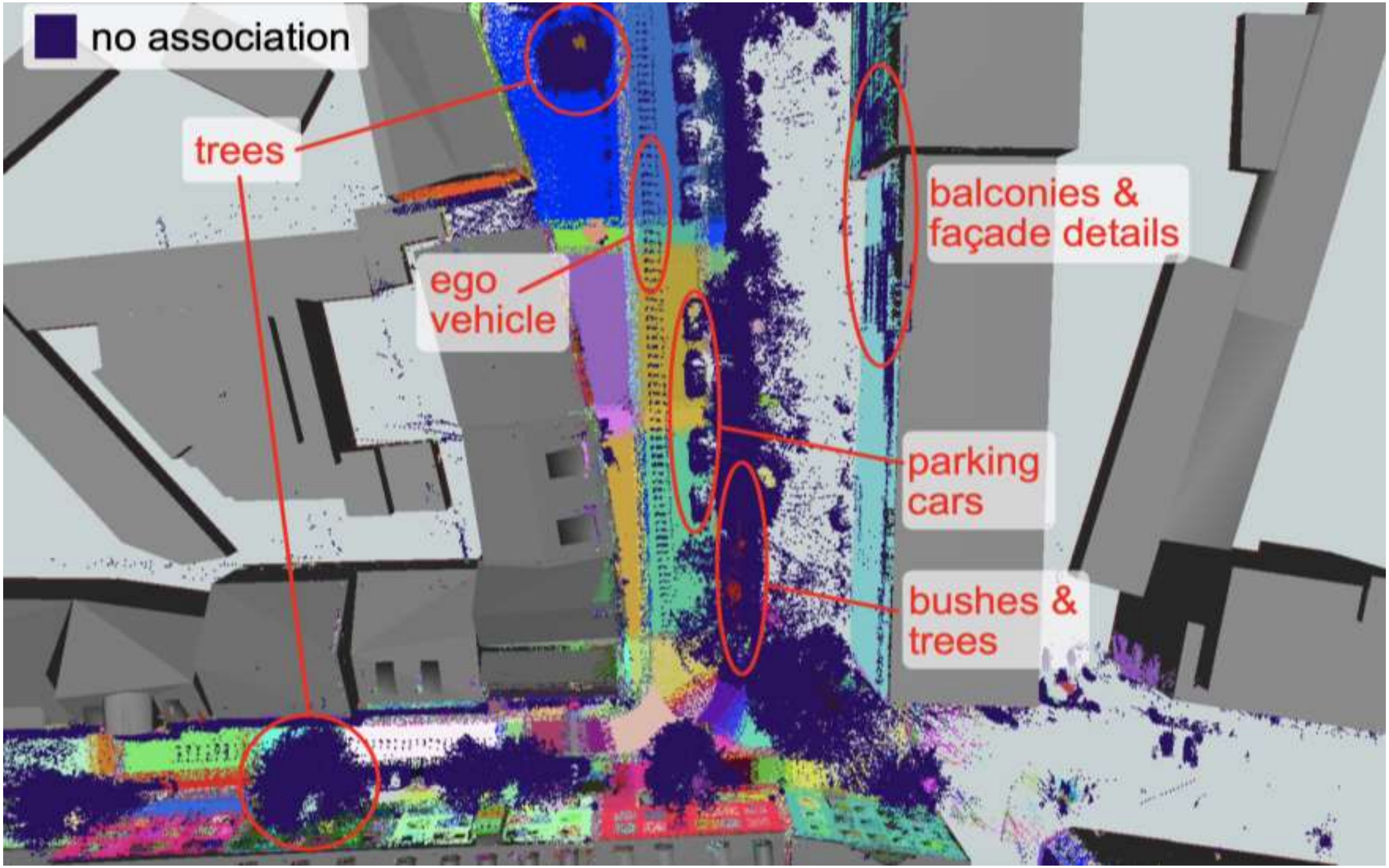


1. Reconstruction of a local map using the graph-based SLAM implementation *Cartographer* on the car's LiDAR sensor data
   - cars' relative position & pose are determined
2. Georeferencing of the generated point cloud by precise alignment with the 3D city model
3. Georeferencing of the vehicle trajectory and pose determination follows implicitly
4. Visualization of the now georeferenced vehicle's sensor data overlayed with features from the CityGML model in the ROS tool *rViz*

Sensor Readings associated to CityGML Objects



Sensor Readings associated to CityGML Objects



Ongoing & Future Work

- Analyzing the collected sensor responses (here: each laser point) for every semantic 3D city object
  - What are the response characteristics for …?
    - Specific building components: wall surfaces, windows, doors, balconies, stairs
    - Specific street space objects: road surfaces, pedwalks, curbs, traffic signs, trash bins, lamp posts
    - Vegetation objects: individual trees
- Also map camera data (and in the future also RADAR data) onto the semantic 3D city objects
- How reliably can we derive the material for each semantic 3D city object from the set of all associated observations?

Coming to

the end…

Summary and Conclusions (1)

- Tackling the urban challenges requires comprehensive digital spatial representations of the environment
- **Different types of city models** are used simultaneously
  - due to different application needs as well as different data acquisition methods
  - semantic 3D city models, BIM, mesh models, 3D point clouds
  - linking the different representations of the same real world objects remains a challenge
- **Semantic modeling is important**
  - required for most applications that go beyond visualizations
  - spatial representations are properties of a semantic object (and not vice-versa, i.e. semantics attached to geometry)

Summary and Conclusions (2)

- CityGML is an international standard of the OGC
  - **geospatial information model** for semantic 3D city and landscape models (based on ISO 191xx)
  - mapped onto different **exchange format** (GML3 aka XML, CityJSON, spatial databases → 3DCityDB, RDF / CityOWL)
  - many cities & entire countries structure their digital models according to CityGML
- **CityGML 3.0** adds **key concepts** required **for digital twins**
  - full historization & alternative versions
  - linking of timeseries data (e.g. from sensors) with city model objects
- CityGML datasets are rich information sources for **urban analyses and simulations,** but also for **AI training**
  - Open Datasets: https://github.com/OloOcki/awesome-citygml

CityGML Ecosystem and (hopefully) useful Links

CityGML Encoding

- CityGML standard specifies two different parts:
  1. **Conceptual Data Model** for the structured representation of the physical objects of the city
  2. **Data format** for mapping such representations to file structures for the purpose of storage and exchange
- In CityGML 2.0 both parts are defined in the same specification. The exchange format is based on the **OGC Geography Markup Language (GML)**, an **XML** application
  - The term "CityGML" thus means both the conceptual model and the exchange format
- There are other encodings that map the conceptual model to alternative formats or storage structures

Alternative CityGML Encodings (1)

- **CityJSON** is a JSON-based exchange format for CityGML city models (www.cityjson.org)
  - issued as an „OGC Community Standard“
  - originally developed by TU Delft
  - there is some software support (e.g. different tools by TU Delft, 3DCityDB, FME, citygml4j, etc.)
- **3DCityDB** is a spatial/relational database schema for the storage and management of CityGML city models (www.3dcitydb.org)
  - developed by the Chair of Geoinformatics at TU Munich in cooperation with the companies virtualcitySYSTEMS and M.O.S.S.
  - Open Source implementation for PostGIS and Oracle Spatial
  - in productive use worldwide in many cities, countries and in research projects; sometimes embedded in commercial systems

Alternative CityGML Encodings (2)

- **OWL / RDF** schemas for the use and analysis of CityGML city models in the Semantic Web / Linked Data. There are two separate developments:
  - Ontology of the Univ. of Geneva (Prof. Gilles Falquet, Prof. Claudine Métral): http://cui.unige.ch/isi/icle-wiki/ontologies and http://cui.unige.ch/isi/onto//citygml2.0.owl
  - **CityOWL** Ontology of the Univ. of Lyon (LIRIS, Group of Prof. Gilles Gesquière): https://github.com/VCityTeam/UD-Graph
- Alternative encodings do not necessarily support the full model scope of CityGML
  - e.g. some limitations with CityJSON, and not directly compatible with OGC services like WFS; but more compact and simpler encoding compared to XML-based encoding, simpler processing e.g. in web browsers

The CityGML Ecosystem

- **CityGML is based on GML** and is thus compatible with many OGC web services standards, e.g.
  - **Web Feature Service** – Access (read, write, modify) entire 3D city models, individual objects and their components
  - **Web Processing Service** – generic interface for encapsulating functions (e.g. data transformations, AI analyses, etc.)
  - **3D Portrayal Service** – Visualization of 3D models (derivation of 3D visualization formats)
- For all services there are different implementations / products (both commercial and Open Source)
- These web services are in turn compatible with the **OGC Catalog Service for the Web (CS/W)** specification
- Comprehensive basis for **Smart City developments**

3D City Database

https://www.3dcitydb.org
https://github.com/3dcitydb

- Open Source software package for the efficient storage, management, and visualization of CityGML-based 3D city models
- **Development cooperation**
  - Chair of Geoinformatics at TUM (Lead)
  - virtualcitySYSTEMS GmbH, Berlin
  - M.O.S.S. Computer Grafik System GmbH, Taufkirchen
- **Development status (**current version: 4.2.0**)**
  - full support of CityGML 1.0.0 and 2.0.0
  - Oracle/PostGIS relational database schema + Import/Export tool with a graphical user interface and a command line interface
  - offers a Web Feature Services 2.0 according to the OGC standard
  - V5.0 currently under development → support of CityGML 3.0

Working with / Visualization of CityGML Datasets

- **FZKViewer** [KIT Karlsruhe, free software]
  https://www.iai.kit.edu/english/1648.php
- **eveBIM** [CSTB France, free software]
  https://www.evebim.fr/telechargement/
- **3DCityDB + 3DCityDB Web Viewer** [TU Munich / VCS / MOSS, Open Source software]
  https://www.3dcitydb.org
- **Azul for MacOS** [TU Delft, Open Source software]
  https://github.com/tudelft3d/azul
- **FME Data Inspector** [Safe Software, commercial]
  https://www.safe.com/fme/fme-desktop/
- **ArcGIS Interoperability Extension** [ESRI, commercial]
  http://www.esri.com/software/arcgis/extensions/datainteroperability

Extract / Transform / Load (ETL) for CityGML

- **HALE Studio** [wetransform, Open Source software]
  https://github.com/halestudio/hale
- **Feature Manipulation Engine** [Safe Software, commercial]
  https://www.safe.com/fme/fme-desktop/

- **GDAL GML Application Schema** [Open Source software]
  https://gdal.org/drivers/vector/gmlas.html
- **Deegree** [lat/lon GmbH, Open Source software]
  https://github.com/deegree/deegree3
- **CityGML tools** [citygml4j, Open Source software]
  https://github.com/citygml4j/citygml-tools

Resources on CityGML

- CityGML 2.0 Standard [Gröger, Kolbe, Nagel, Häfele 2012]
https://www.ogc.org/standards/citygml
- T. H. Kolbe, 2009: Representing and Exchanging 3D City Models with CityGML. In: Lee, Zlatanova (eds.), 3D Geo-Information Sciences, Springer
https://mediatum.ub.tum.de/node?id=1145752
- B. Willenborg, M. Sindram, T. H. Kolbe, 2017: Applications of 3D City Models for a better understanding of the Built Environment. In: Behnisch, Meinel (eds.): Trends in Spatial Analysis and Modelling. Springer
https://mediatum.ub.tum.de/node?id=1348882
- List of worldwide Open Data 3D city models in CityGML („Awesome CityGML"):
https://github.com/OloOcki/awesome-citygml
- F. Biljecki, K. Kumar, C. Nagel, 2018: CityGML Application Domain Extension (ADE): overview of developments. Open geospatial data, softw. stand. 3, 13
https://doi.org/10.1186/s40965-018-0055-6

Resources on CityGML 3.0

- CityGML 3.0 Conceptual Model + UML Diagrams + Discussions
  https://github.com/opengeospatial/CityGML-3.0CM
- CityGML 3.0 XML Schema Files, Test datasets
  https://github.com/opengeospatial/CityGML-3.0Encodings
- IFC → CityGML 3.0 FME Workspace
  https://github.com/tum-gis/ifc-to-citygml3
- T. Kutzner, K. Chaturvedi, T. H. Kolbe, 2020: CityGML 3.0: New Functions Open Up New Applications. PFG – Journal of Photogrammetry, Remote Sensing and Geoinformation Science, 2020, 19
  http://dx.doi.org/10.1007/s41064-020-00095-z
- C. Beil, R. Ruhdorfer, T. Coduro, T. H. Kolbe, 2020: Detailed Streetspace Modelling for Multiple Applications: Discussions on the Proposed CityGML 3.0 Transportation Model. ISPRS International Journal of Geo-Information 9 (10)
  https://www.mdpi.com/2220-9964/9/10/603