ASAM Ideation Workshop

Modelling Roads and the Streetspace using CityGML 3.0 and its relation to OpenDRIVE

July 24, 2024

Christof Beil
Technical University of Munich
Chair of Geoinformatics
christof.beil@tum.de

Semantic 3D City Modeling

- City is decomposed into meaningful objects with clear semantics and defined spatial and thematic properties
  - buildings, roads, railways, terrain, water bodies, vegetation, bridges
  - buildings or roads may be further decomposed into different storeys or sections / intersections (and even more detailed into apartments and single rooms or individual lanes)

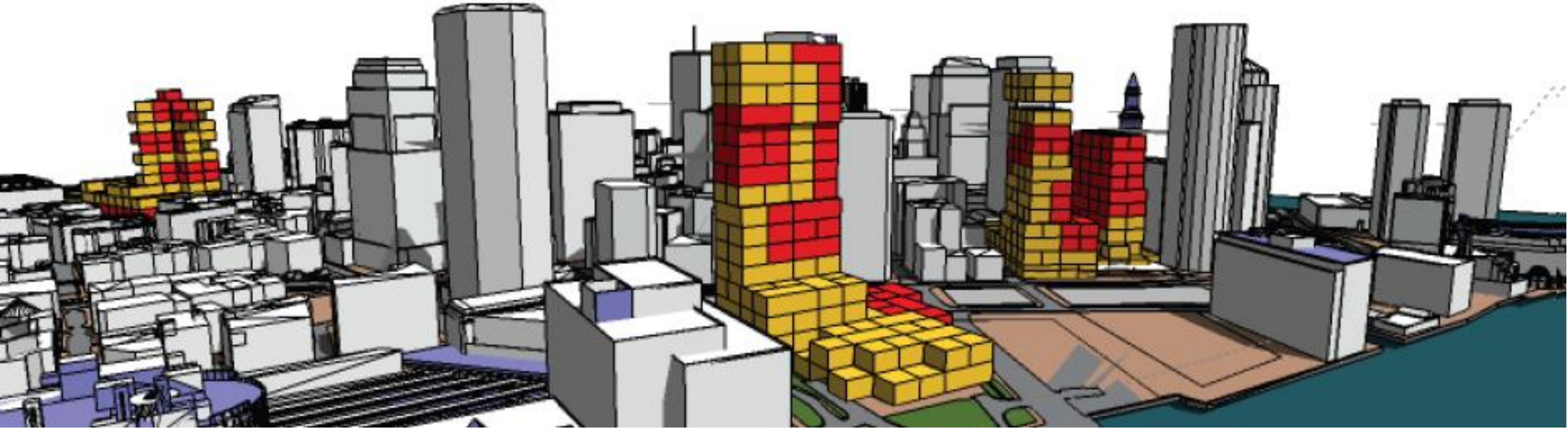
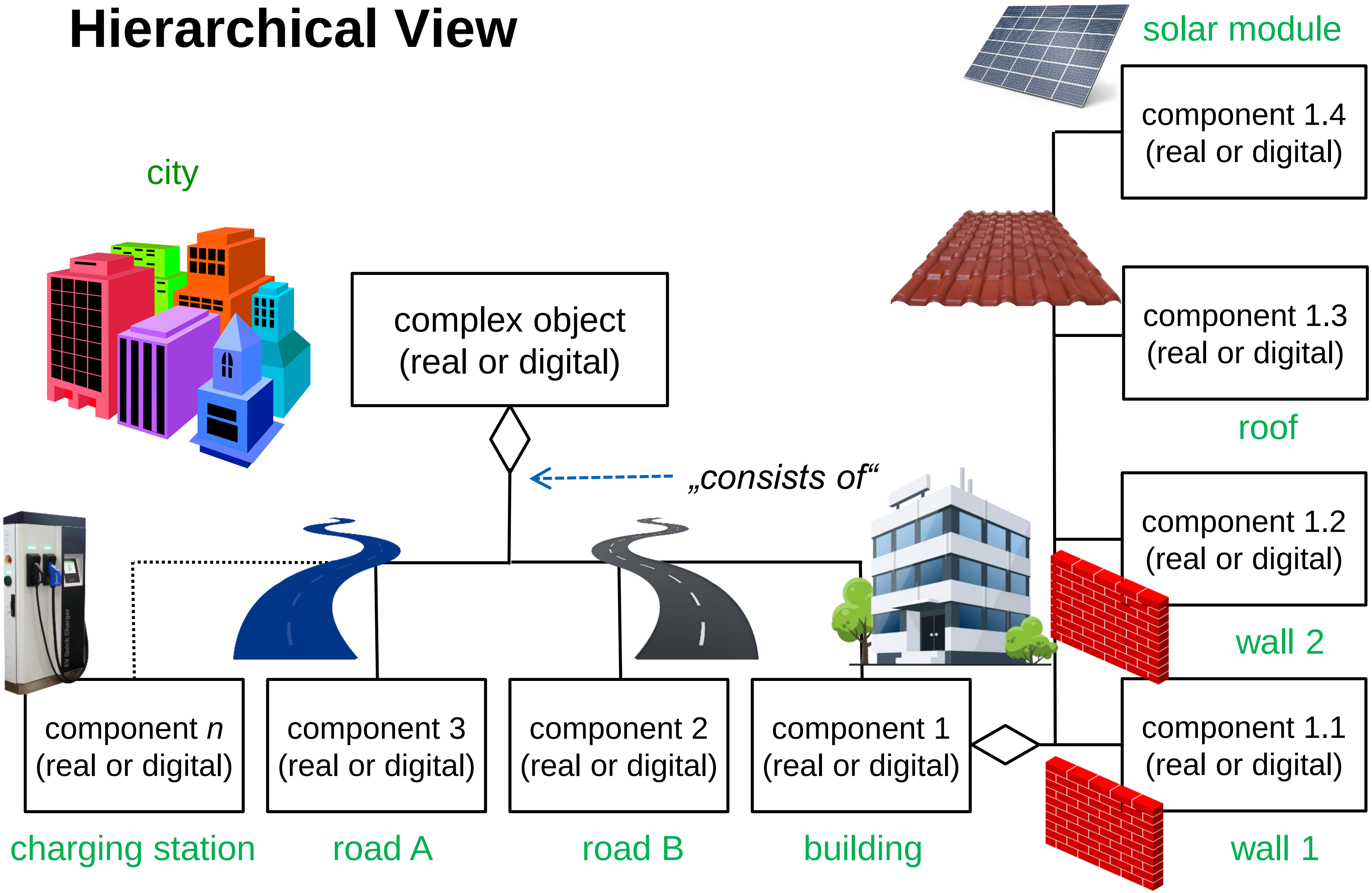
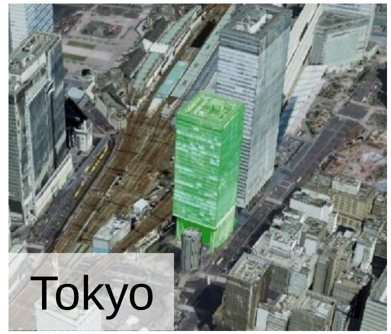


Image: Paul Cote, Harvard Graduate School of Design

Hierarchical View



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
- Testing automated driving functions
- Traffic planning and simulation

City Geography Markup Language – CityGML

Application independent Geospatial Information Model
for semantic 3D city and landscape models

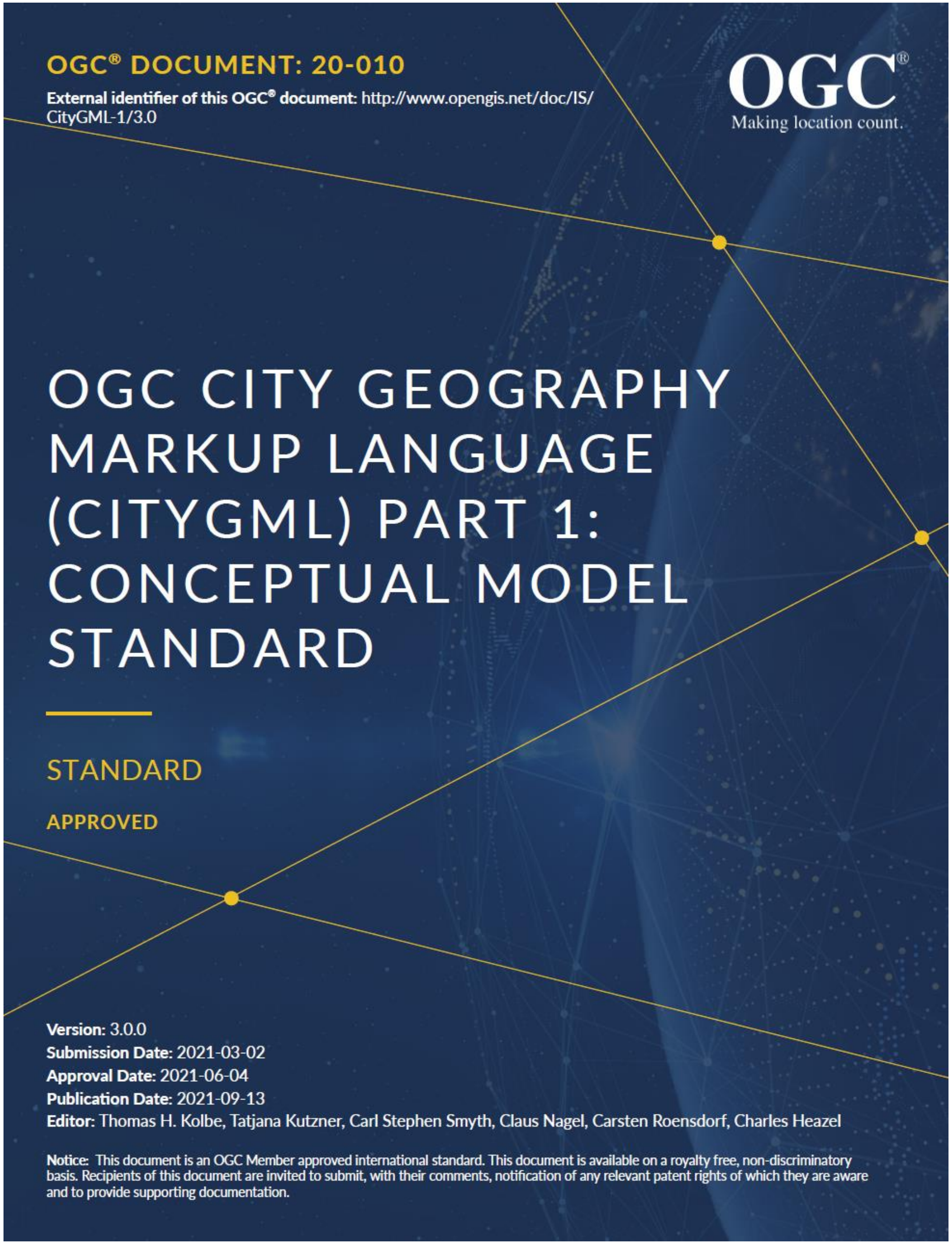
- comprises **different thematic areas** (buildings, roads, vegetation, water, terrain, tunnels, bridges etc.)
- **Data model (UML) + Exchange format**
- Key strengths: **Geo-referenced**, **3D geometries**, **topological** information as well as **semantic capabilities**, **time-dependent properties**, easy **visualization**
- Large scale (entire city) management using **geo-databases** (e.g. 3DCityDB)

International Standard of the **Open Geospatial Consortium**

- Version 2.0.0 was issued in 3/2012
- **Version 3.0 was issued in 2021**

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

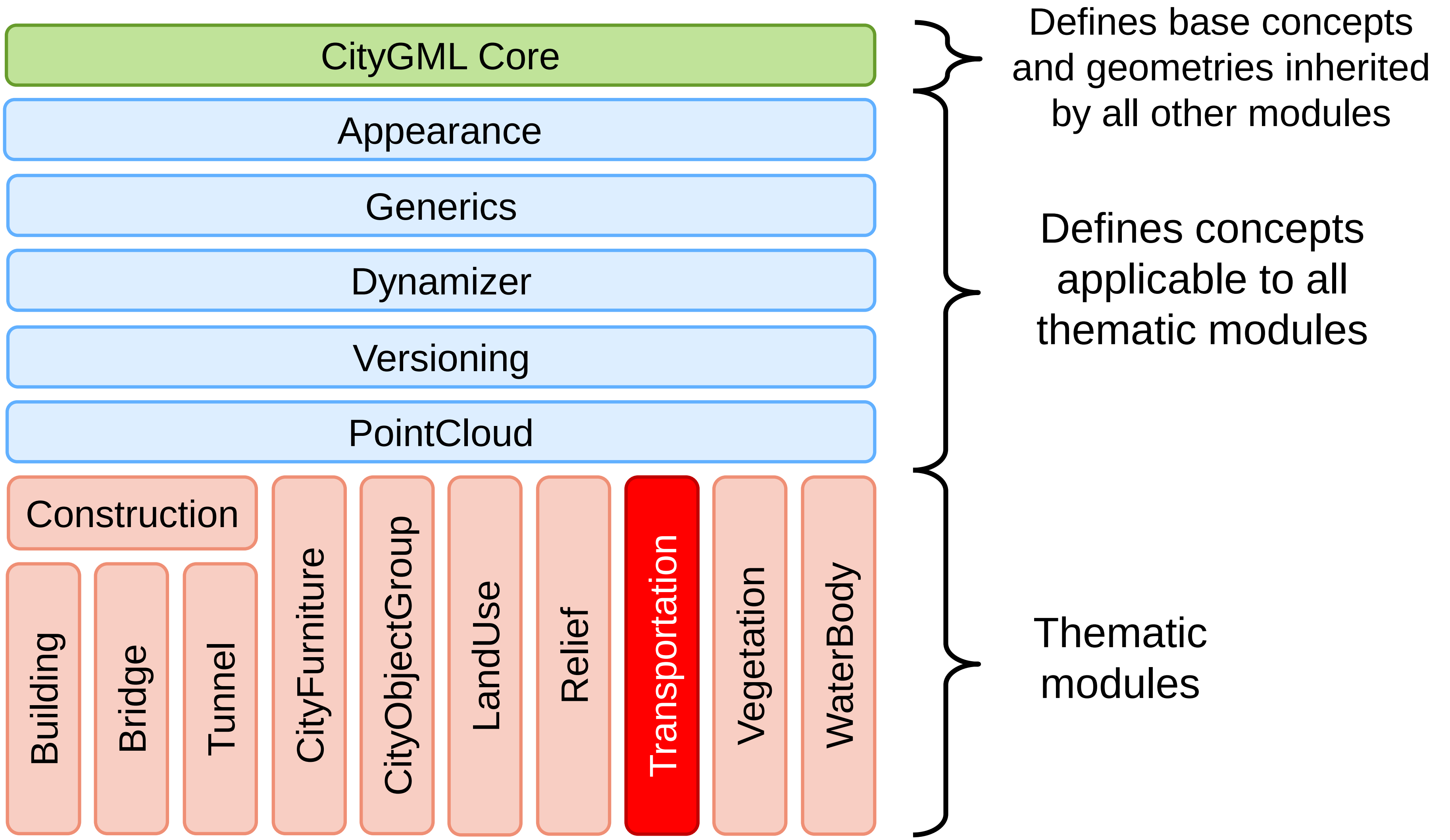
Version: 3.0.0
Submission Date: 2021-03-02
Approval Date: 2021-06-04
Publication Date: 2021-09-13
Editor: Thomas H. Kolbe, Tatjana Kutzner, Carl Stephen Smyth, Claus Nagel, Carsten Roensdorf, Charles Heazel

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.

CityGML Usage

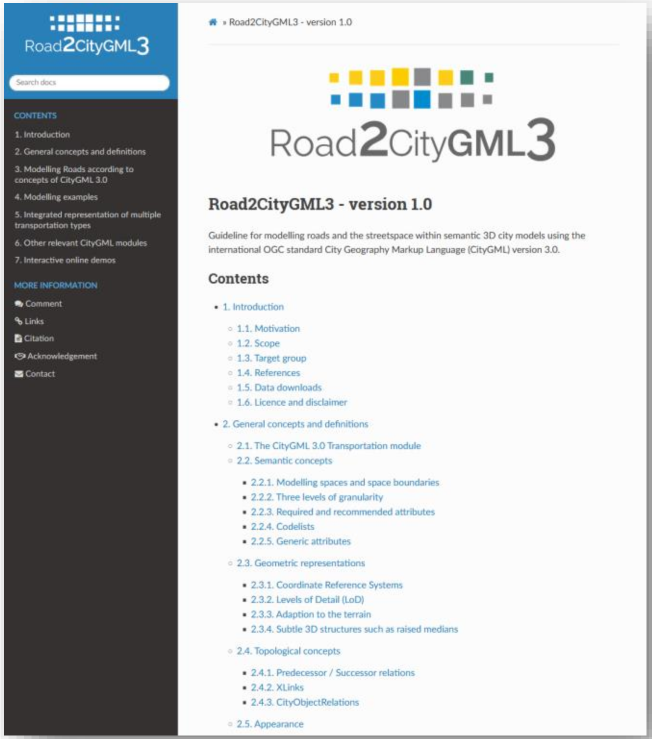
- CityGML is already successfully used at national scale
  - The official national and municipal 3D geoinformation standards of Germany and the Netherlands based on CityGML 1.0/2.0
  - Japan published 3D city models for >100 cities based on CityGML and the i-Urban Revitalization Application Domain Extension
- 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

Modular structure of the CityGML standard



Guideline Road2CityGML3

- https://tum-gis.github.io/road2citygml3
- Concepts of the CityGML 3.0 for modelling the streetspace
  - Semantics
  - Geometries
  - Topology
  - Appearance
- Examples
  - E.g. NYC, Melbourne, Ingolstadt, Munich
- Data downloads



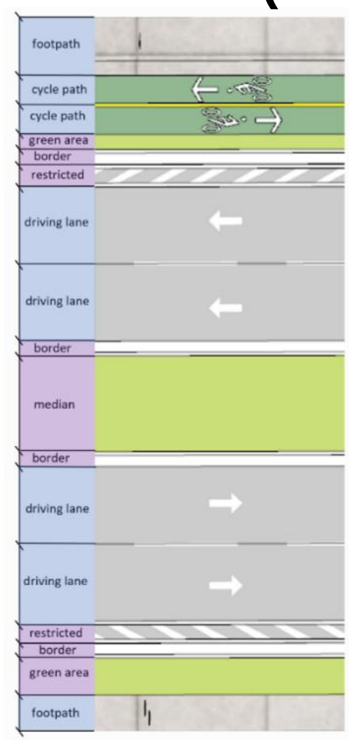
Representation of Road Infrastructure

- Real world objects are represented by geographic features as defined in **ISO 19109**
- Each object has a **unique identifier** (gml:id attribute)
- **Hierarchical structure** / decomposition into individual objects

Road

Section

Traffic Areas (Lanes)

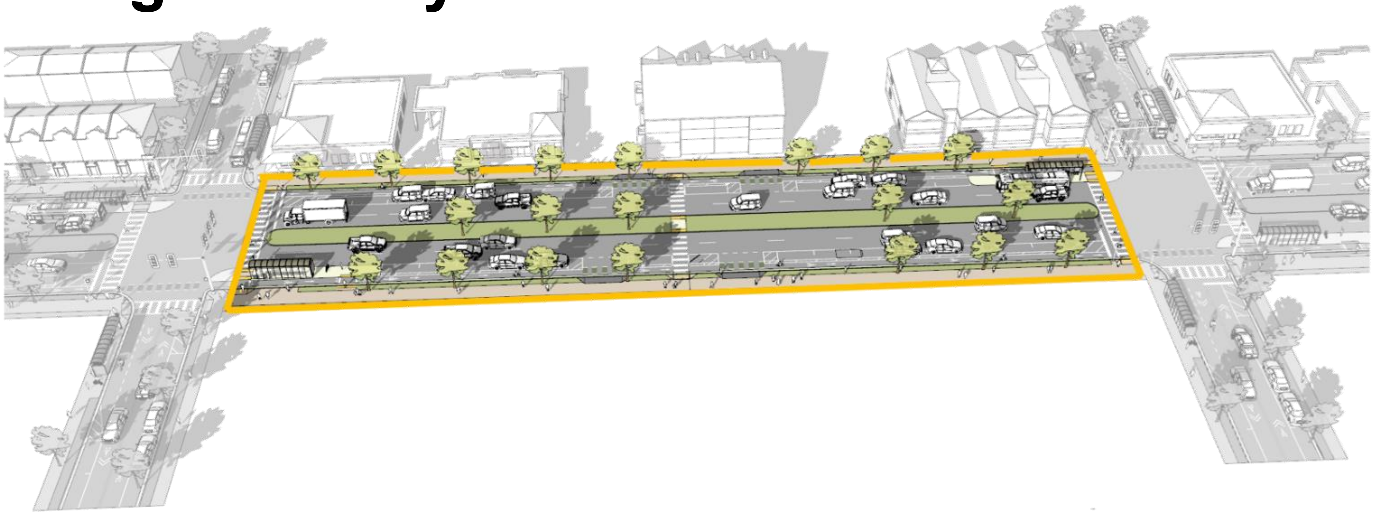


Representation of Roads

- A **Road** is a transportation space used by vehicles, bicycles and/or pedestrians.
- Usually **Roads** should be distinguished by individual names

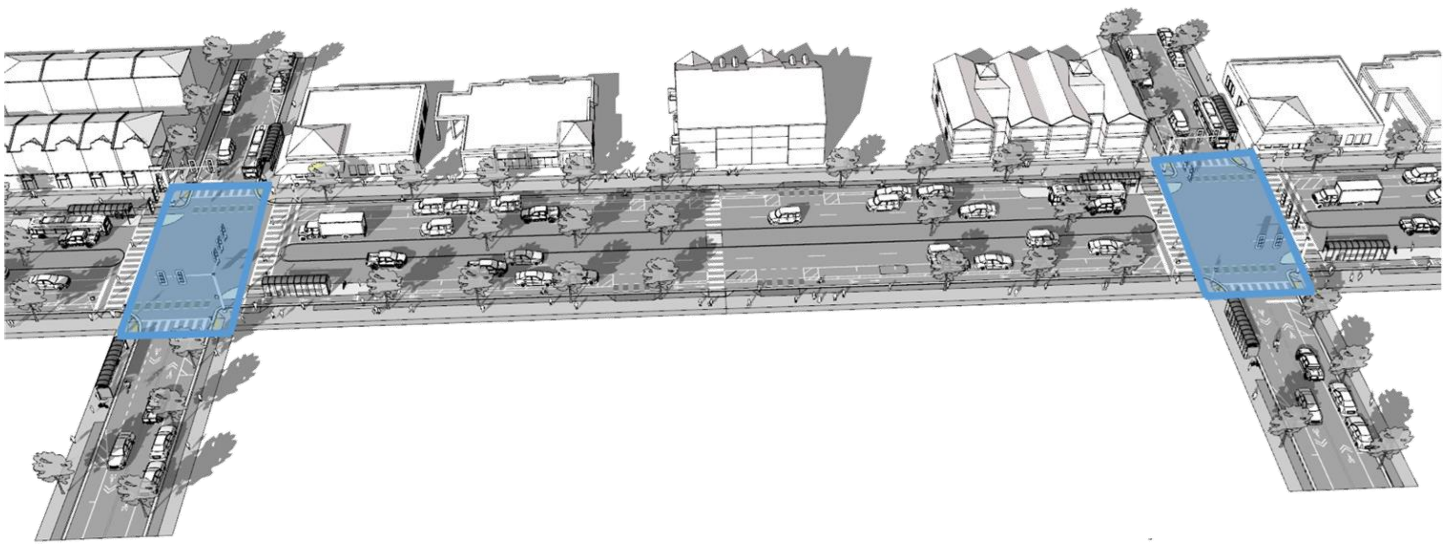
Segmenting road networks into Sections and Intersections

- **Roads** can consist of several **Sections**
- A **Section** is a transportation segment that can clearly be assigned to one **Road** (or **Railway, Track, Waterway**) object.
- **Sections** should cover the entire width of a **Road** and thus directly correspond to the representation of transportation objects in **granularity 'area'.**



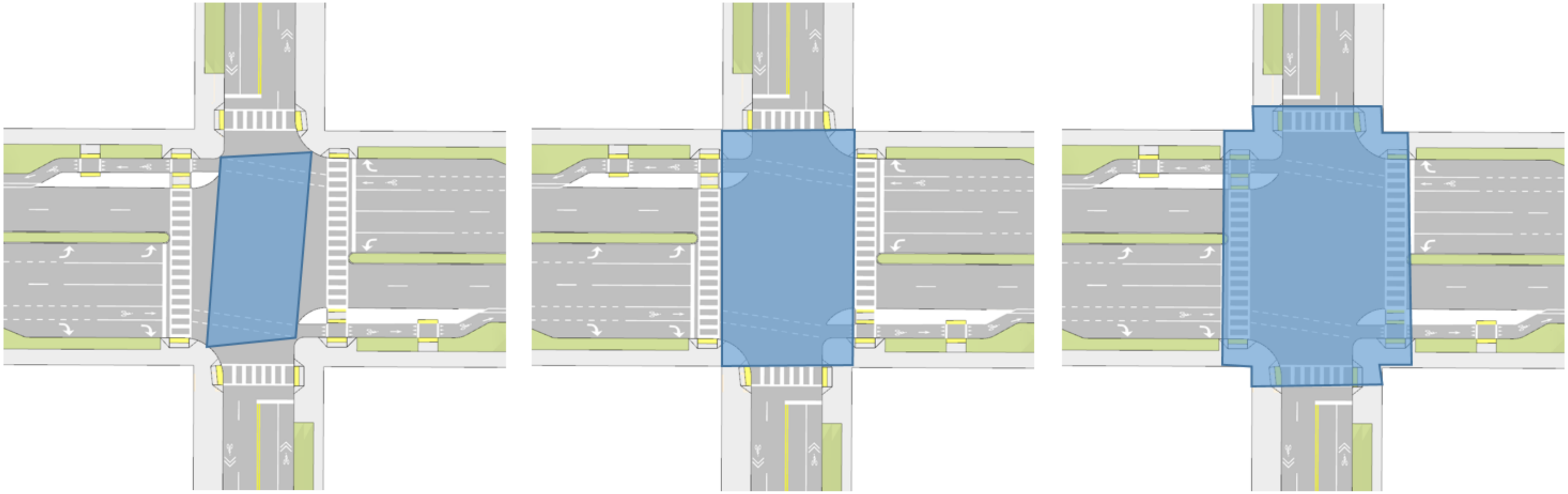
Segmenting road networks into Sections and Intersections

- An **Intersection** is a transportation space that is a shared segment of multiple **Roads** or other transportation objects such as **Railways** (e.g. a crossing of two **Roads** or a level crossing of a **Road** and a **Railway**).
- The link of an **Intersection** to multiple **Roads** can be explicitly modelled (using **XLinks**)



Segmenting road networks into Sections and Intersections

- An **Intersection** is a transportation space that is a shared segment of multiple **Roads** or other transportation objects such as **Railways** (e.g. a crossing of two **Roads** or a level crossing of a **Road** and a **Railway**).
- Different possibilities to define Intersections areas

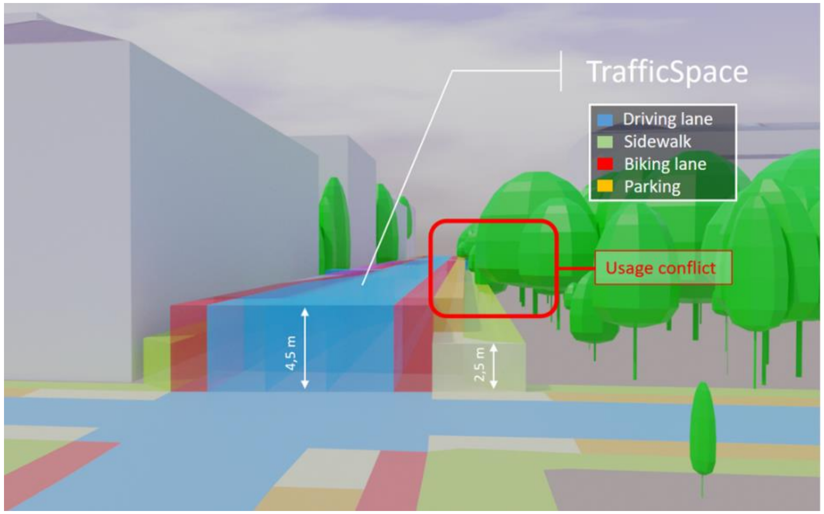


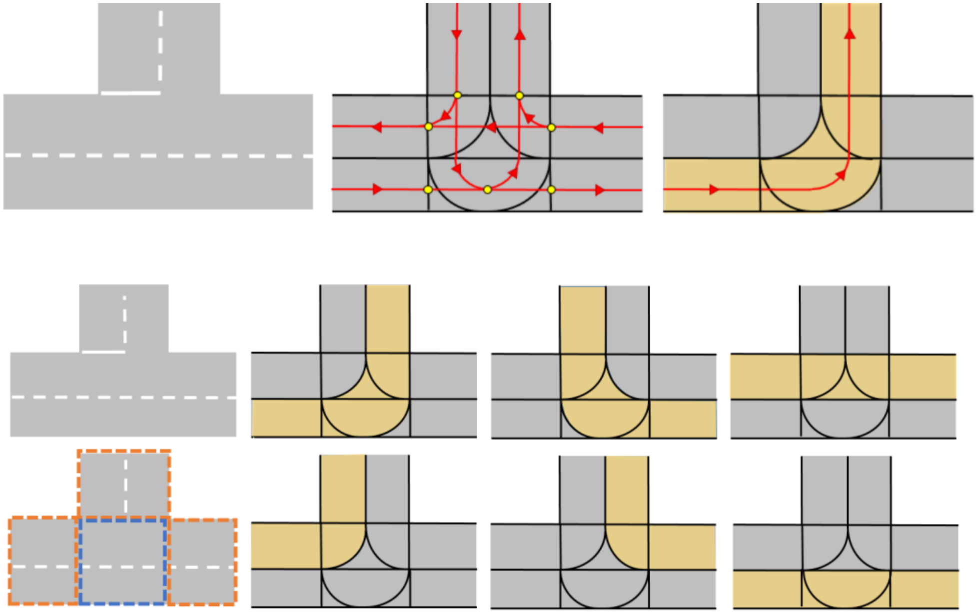
Segmenting road networks into Sections and Intersections

- Example: Melbourne

Transportation Spaces in CityGML

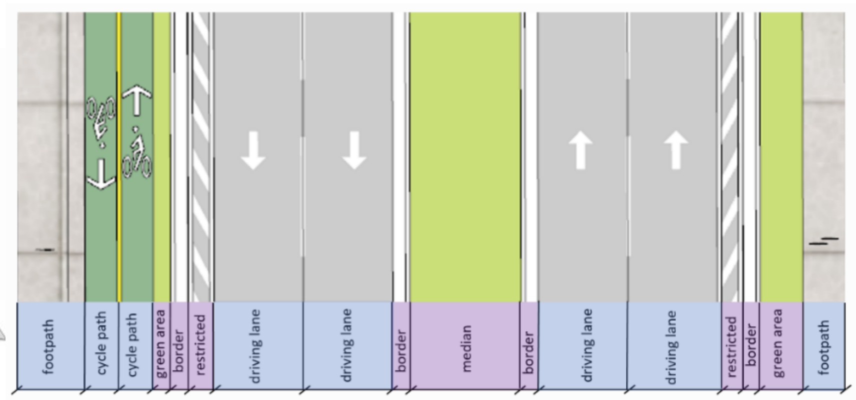
- **TrafficSpace** is a space in which traffic takes place
- Can be represented geometrically using e.g. **linear or volumetric geometries** (or geometric representation can be omitted if not required)
- Ground surfaces of **TrafficSpaces** are **TrafficAreas**





TrafficAreas and AuxiliaryTrafficAreas

- A **TrafficArea** is the ground surface of a **TrafficSpace** upon which traffic actually takes place.
- **AuxiliaryTrafficAreas** are describing further elements of the **Road**, like kerbstones, raised medians, and green areas not intended for direct traffic usage.
- Standardized attributes on function, surface material, etc.
- **Example: Section** with two carriageways segmented into **TrafficAreas** and **AuxiliaryTrafficAreas**

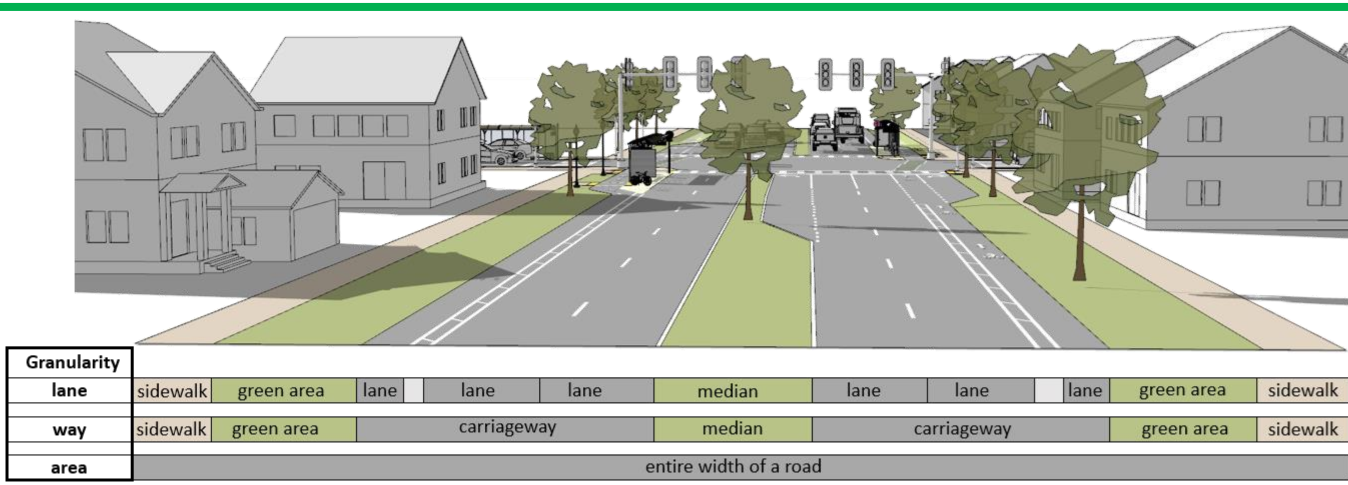


Three levels of granularity

- In **granularity ‘area’** the entire width of a **Road** is modelled with one single object.
- In **granularity ‘way’** individual objects are modelled per surface function (or traffic type). This means one object per carriageway is used.

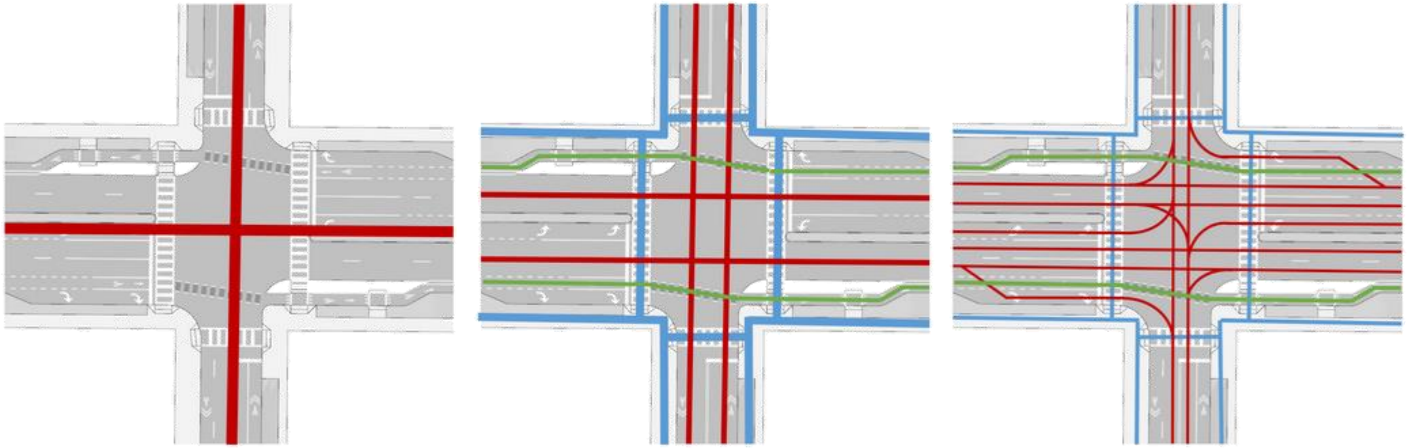
Granularity of OpenDRIVE lane data

- In **granularity ‘lane’** each individual lane (including driving lanes for vehicles) are modelled separately.



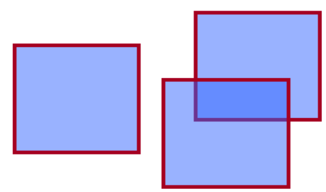
Three levels of granularity

- In **granularity 'area'** the entire width of a **Road** is modelled with one single object.
- In **granularity 'way'** individual objects are modelled per surface function (or traffic type). This means one object per carriageway is used.
- In **granularity 'lane'** each individual lane (including driving lanes for vehicles) are modelled separately.



Geometry

- ISO 19107:2003 Geographic information - Spatial schema
- Geometry Types most relevant for Transportation Module
  - GM_Point
  - GM_MultiPoint
  - **GM_MultiCurve**
  - **GM_MultiSurface**
  - GM_Solid
  - GM_Object



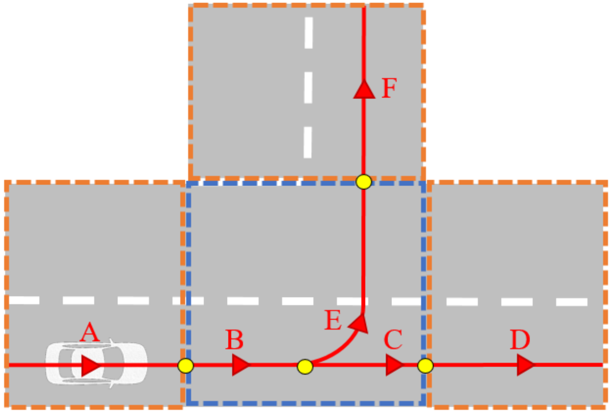
<<CodeList>>
GM_CurveInterpolation

+ linear
+ geodesic
+ circularArc3Points
+ circularArc2PointWithBulge
+ elliptical
+ clothoid
+ conic
+ polynomialSpline
+ cubicSpline
+ rationalSpline

- Nearly all geometries use 3D coordinates
- Coordinates always have to be given with respect to a coordinate reference system (CRS)

Topology

- Predecessor / Successor Concept
- Traffic direction attribute
- Useful for routing / navigation applications

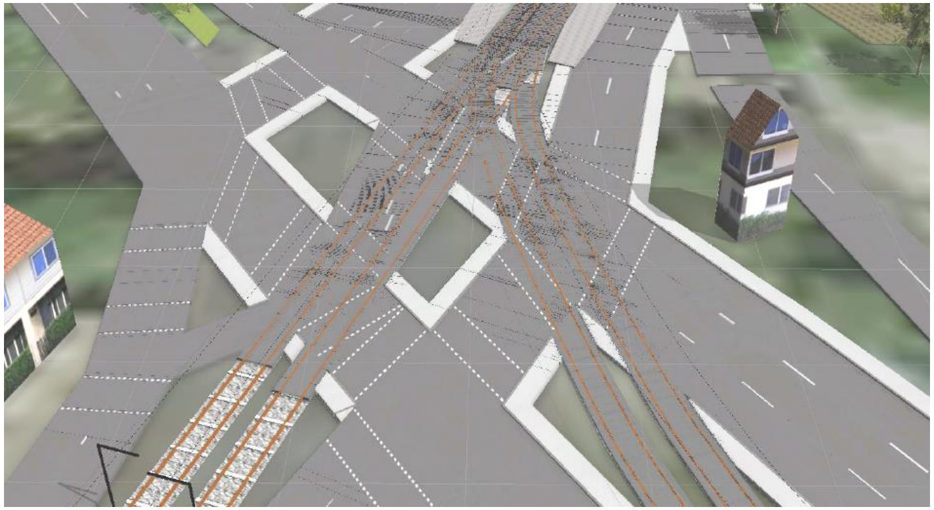


TrafficSpace	Predecessor	Successor(s)
A		B
B	A	E, C
C	B	D
D	C	
E	B	F
F	E	

- Linking concept for non-redundant representations (e.g. XLinks)
  - Geometry shared by different semantic objects only needs to be represented once

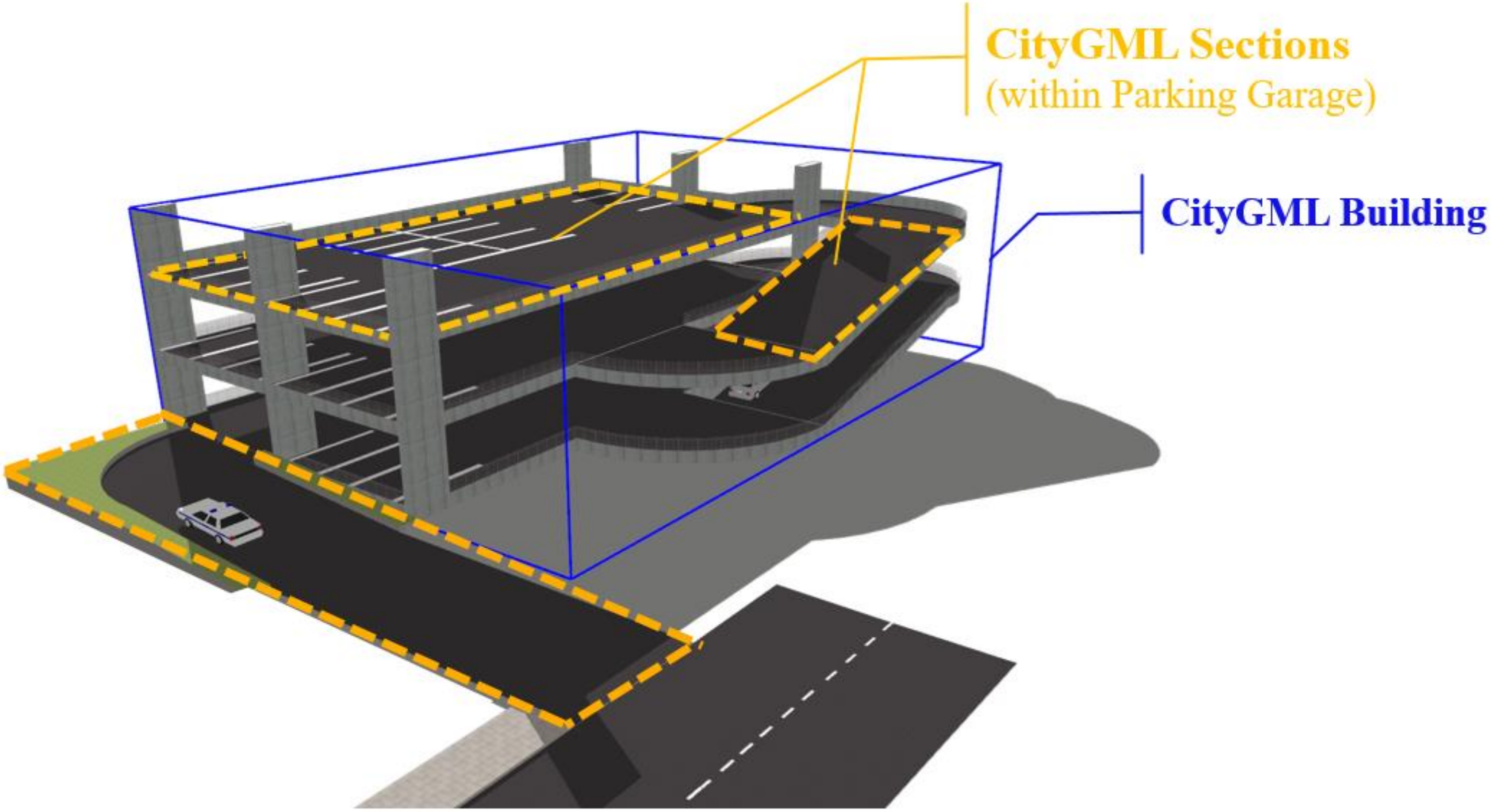
Integration of multiple Transportation types

- Concepts apply to Road, Railway and Waterway infrastructure
- Combined and non-redundant geometric and semantic modelling of multiple transportation infrastructure possible
- **Consistent**, **integrated** and **complete 3D city model**



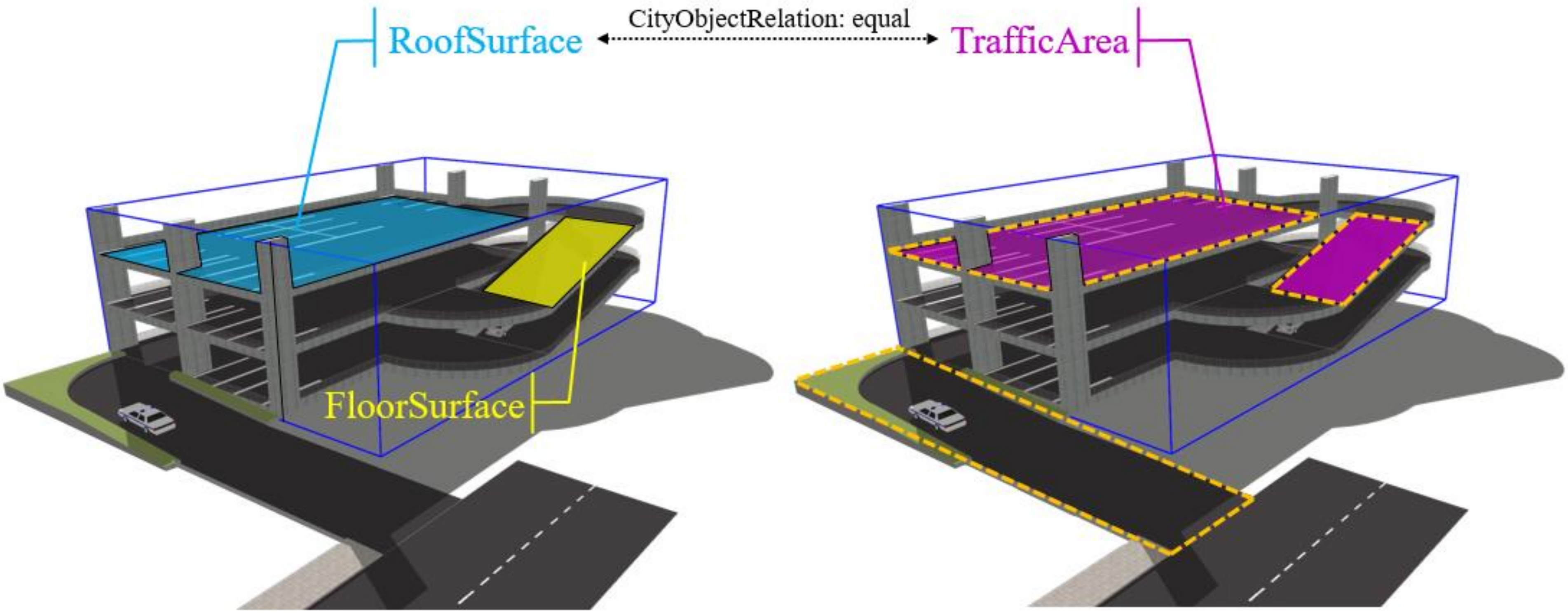
Roads within Buildings (Parking garage)

- Transportation networks and *Roads* can reach into *Buildings* (e.g. within a parking garage)



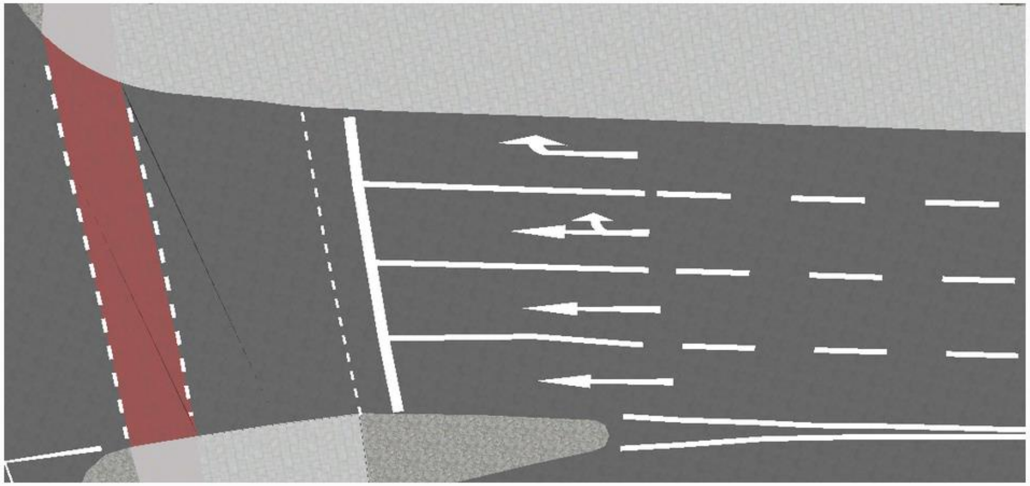
Roads within Buildings (Parking garage)

- Transportation networks and *Roads* can reach into *Buildings* (e.g. within a parking garage)



Other relevant CityGML objects

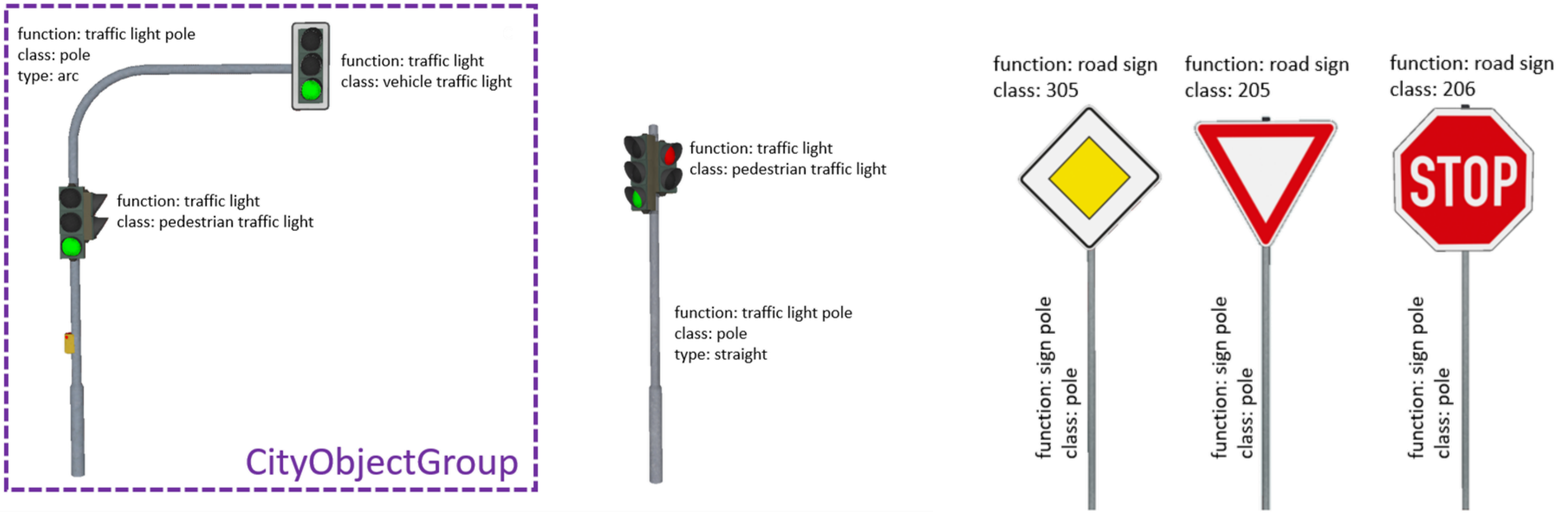
- **Markings**
  - Visible pattern on a transportation area relevant to the structuring or restriction of traffic
  - E.g. using explicit (polygonal) geometries
  - Each marking can be related to a specific lane
  - Attributive information on function (e.g. stop line, arrow straight, etc.) can be contained



Other relevant CityGML objects

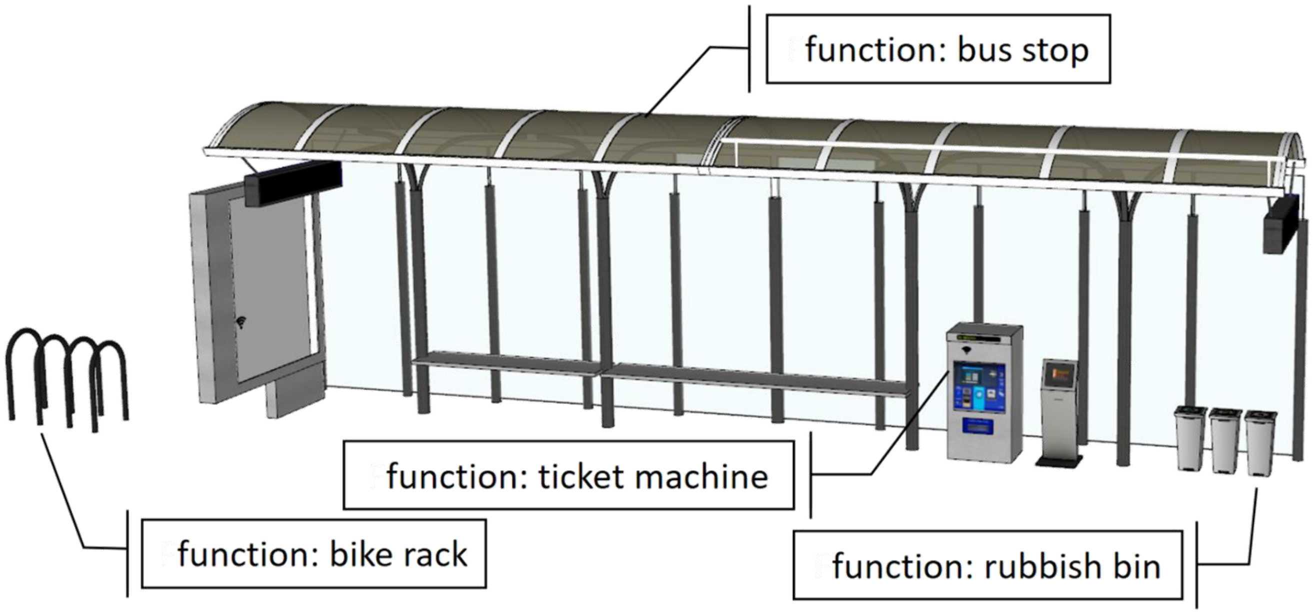
▸ CityFurniture

- Object or piece of equipment installed in the outdoor environment for various purposes (e.g. street signs, traffic signals, street lamps, benches, trash bins, bike racks, etc.)
- Traffic signs / lights can be related to individual lanes
- Implicit or explicit geometries (point / surfaces geometries or actual 3D models)



Other relevant CityGML objects

- **CityFurniture**
  - Object or piece of equipment installed in the outdoor environment for various purposes (e.g. street signs, traffic signals, street lamps, benches, trash bins, bike racks, etc.)



Other relevant CityGML objects

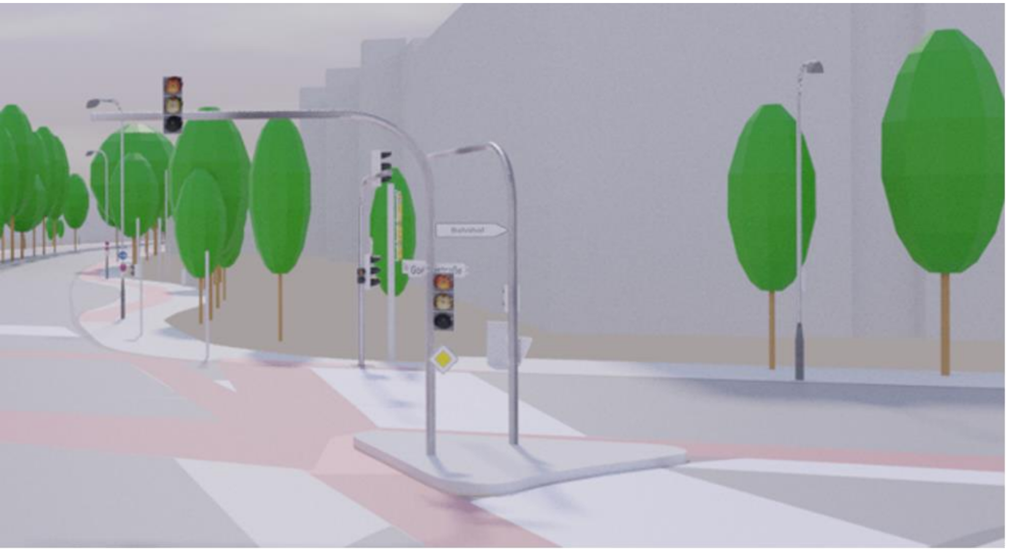
- **Holes**
  - Openings in the surface of a street, road, or plaza, such as road damage, manholes, or gullies

Other relevant CityGML objects

- **Squares**
- Transportation space for unrestricted movement of vehicles, bicycles and/or pedestrians
- Includes plazas as well as large sealed surfaces such as parking lots or gas stations

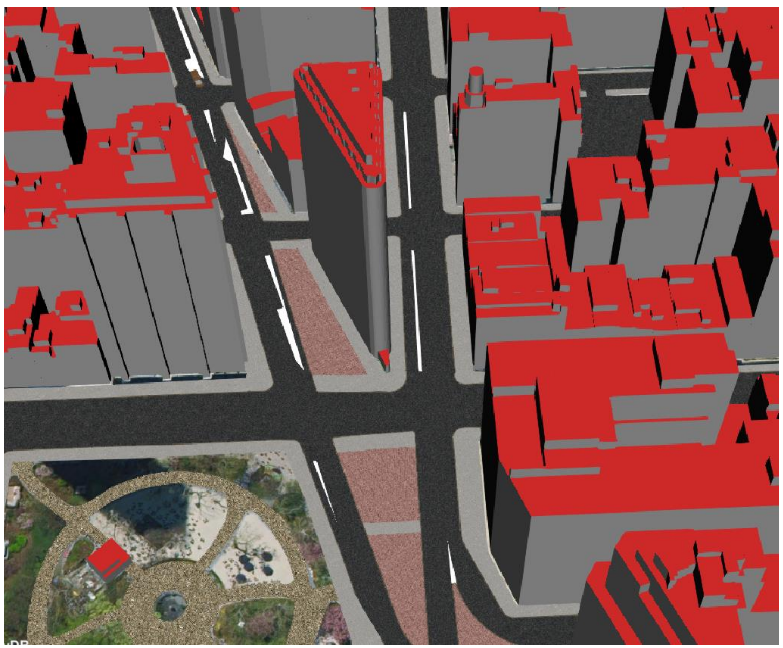
Other relevant CityGML obejcts

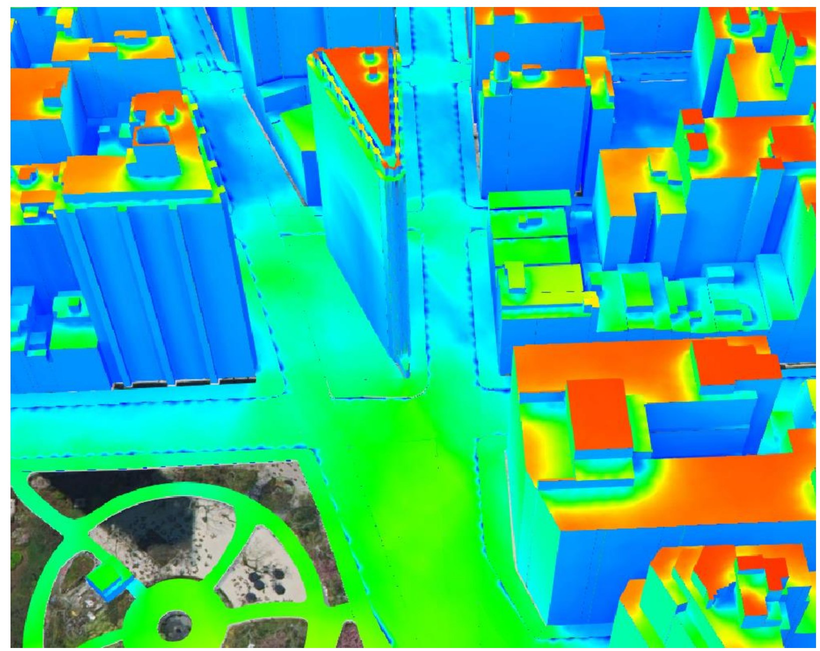
- **Vegetation**
  - Solitary vegetation objects, such as trees, bushes and ferns, or vegetation areas that are covered by plants of a given species or a typical mixture of plant species, such as forests, steppes and wet meadows



Appearance

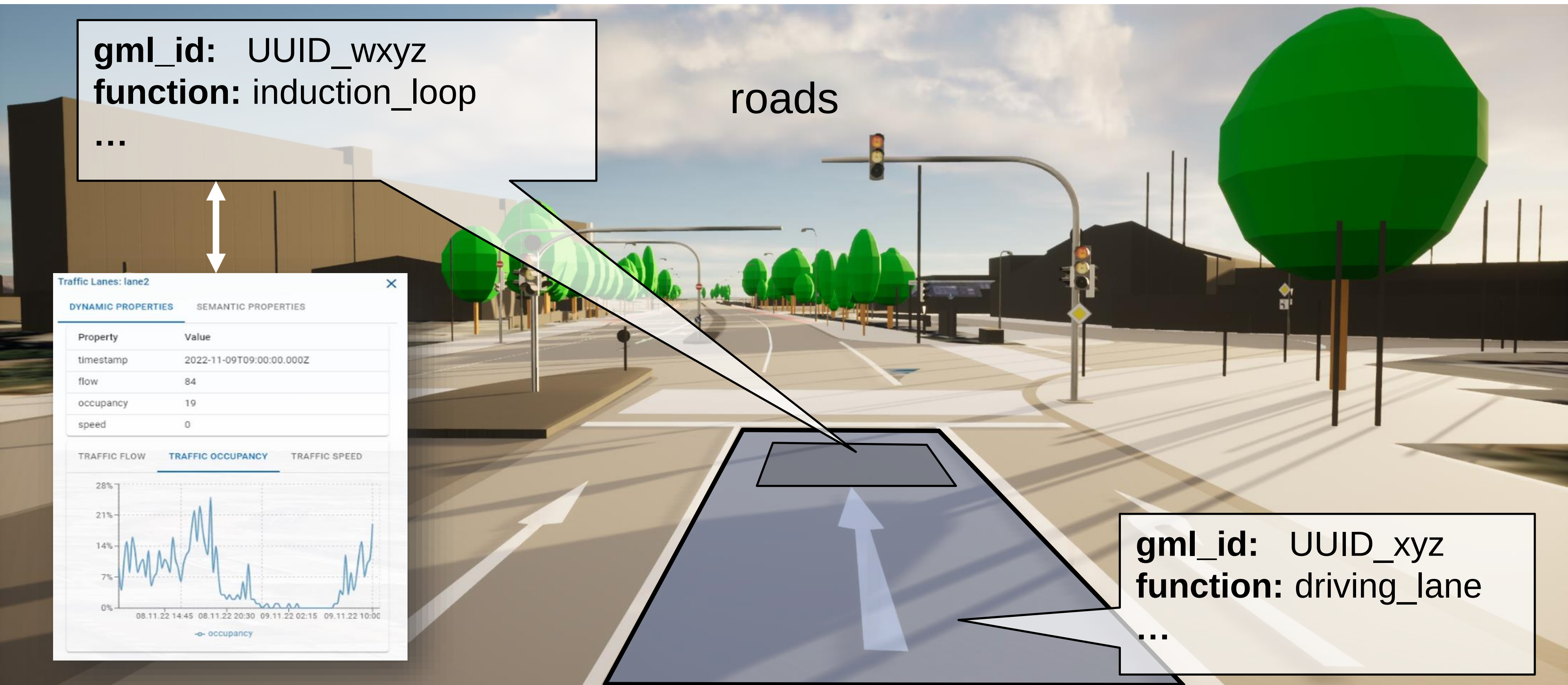
- Colors / textures can be associated with city objects
- E.g. material definitions from the X3D and COLLADA standards
- Single surface geometry object may have surface data for multiple themes (arbitrary categories, e.g. solar irradiation)





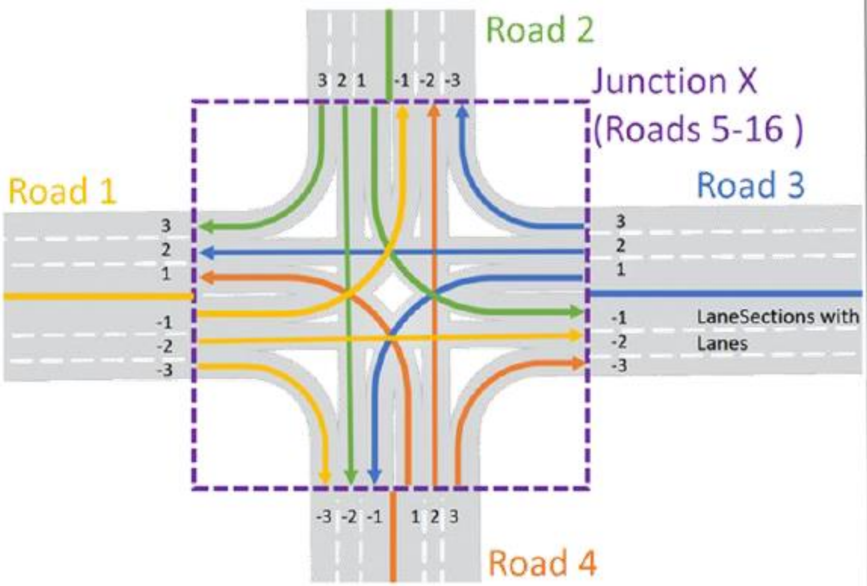
Representing time-depentent properties using Dynamizers

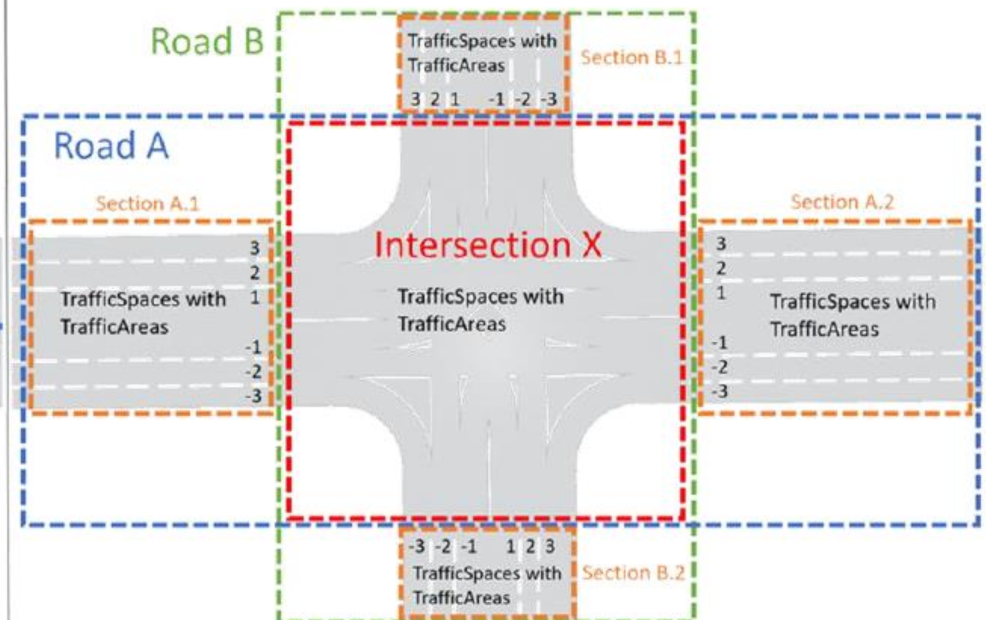
- **Dynamizer**
- Concepts that enables representation of time-varying data for city object properties as well as for integrating sensors with 3D city models.



Comparison of OpenDRIVE and CityGML

OpenDRIVE Standard Roads / Roads within a Junction || **CityGML** Roads with Sections / and Intersection





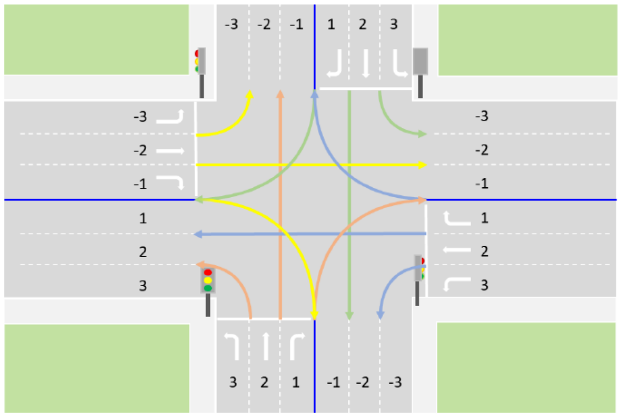
Referenceline-based (parametric) redundant and overlapping geometries and data gaps

Surface-based (explicit coordinates) and non-redundant without overlapping geometries or gaps

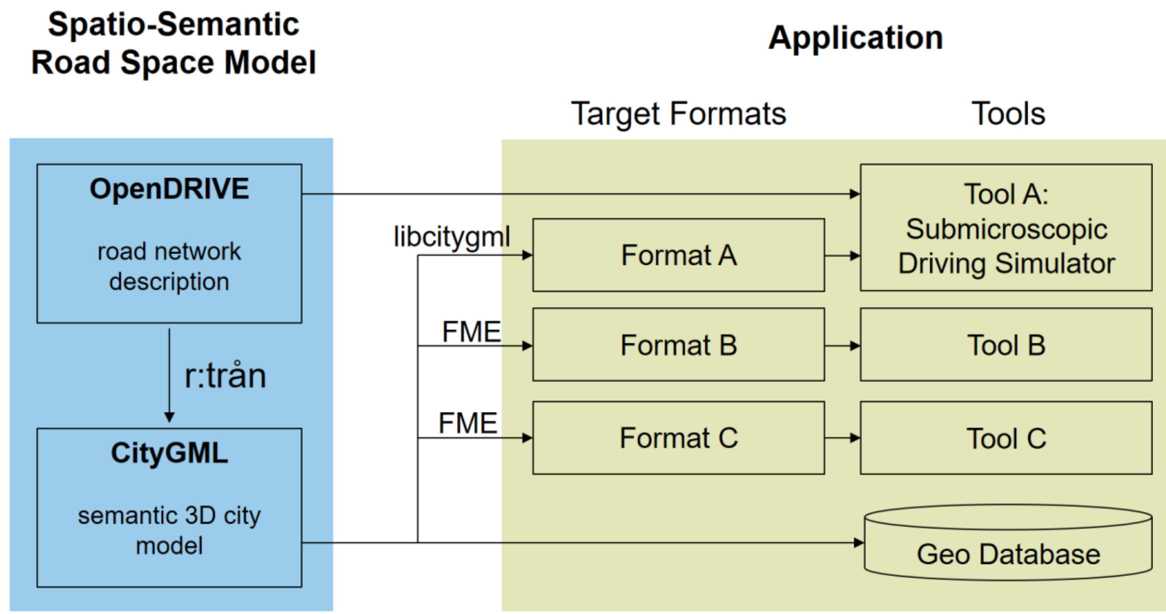
Open source OpenDRIVE to CityGML converter

https://rtron.io/

Parametric OpenDRIVE

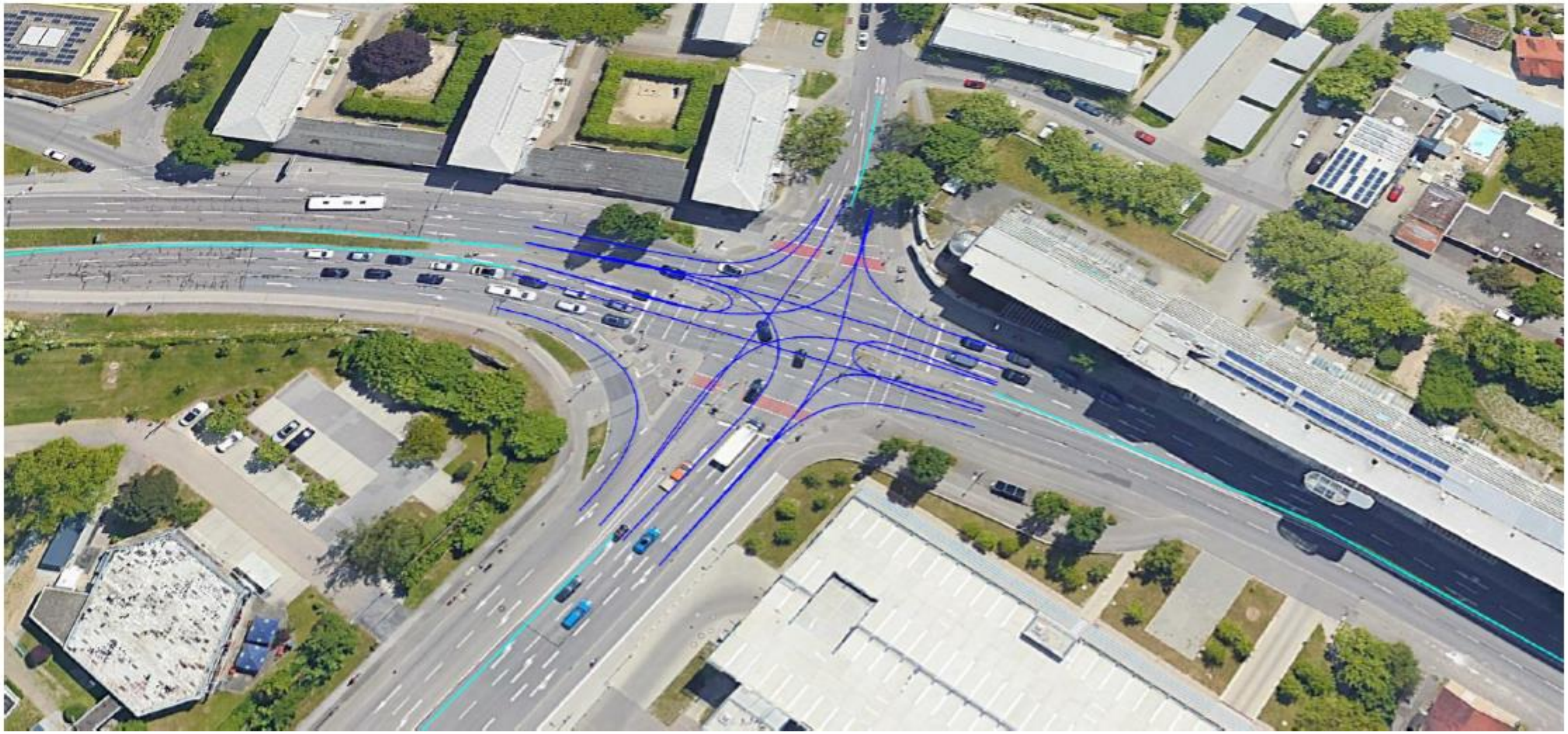


Semantic 3D city model
with explicit coordinates

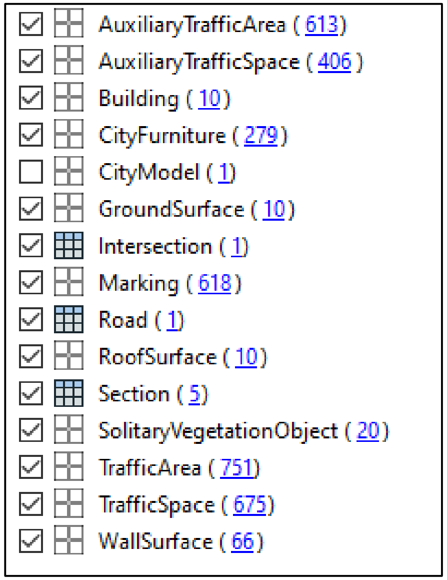


CityGML 3.0 from OpenDRIVE

- OpenDRIVE data created by 3DMappingSolutions GmbH
  Intersection in Ingolstadt

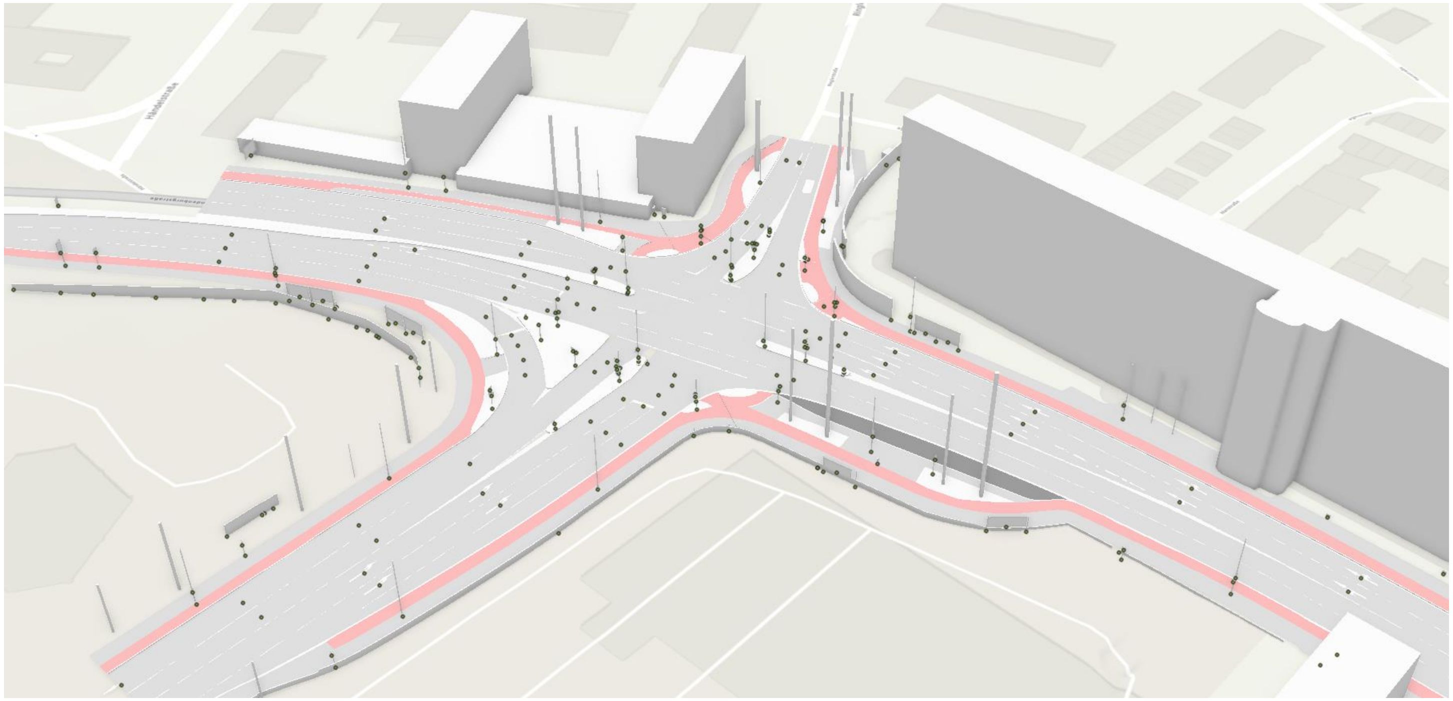


CityGML 3.0 from OpenDRIVE



Converted from OpenDRIVE to CityGML 3.0 using r:tron

Containes overlaps and redundant geometries due to the original data structure



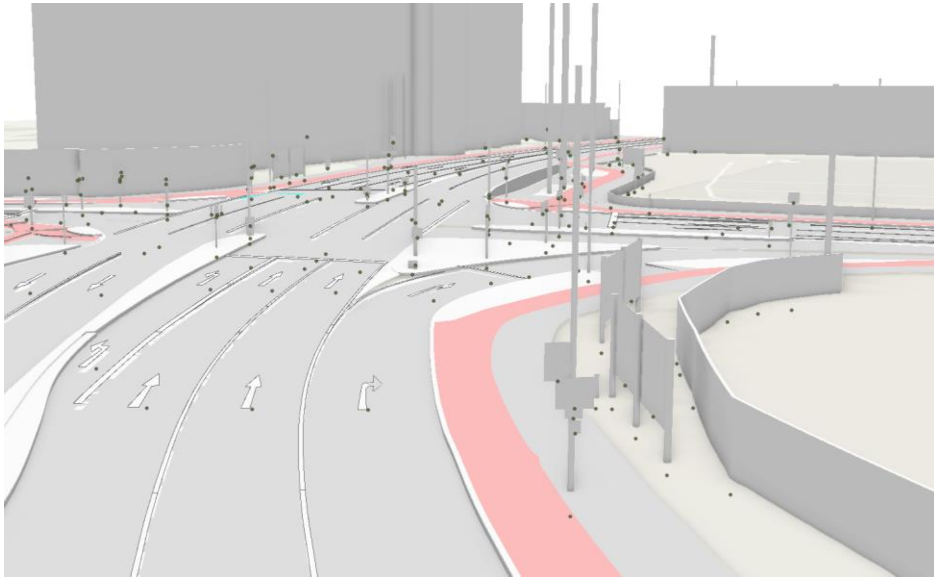
CityGML 3.0 aus OpenDRIVE

CityFurniture
- Poles
- Traffic signs
- Traffic lights
- Barriers
- etc.

Vegetation
- Trees
- Information on height, diameter, etc

Information on signs / traffic lights and trees is (in some cases) abstracted with points containing attributes on type-subtype etc.

This can be substituted with actual 3D sign models etc.



CityGML 3.0 Streetspace Modeling Example 1

- Streetspace model of Ingolstadt automatically generated from OpenDRIVE data and visualized in the UnrealEngine

https://www.youtube.com/watch?v=t5UXEVQcMj4

CityGML 3.0 Streetspace Modeling Example 2

- Digital Twin Munich: Web-based traffic simulation visualization combined with a semantic 3D city model

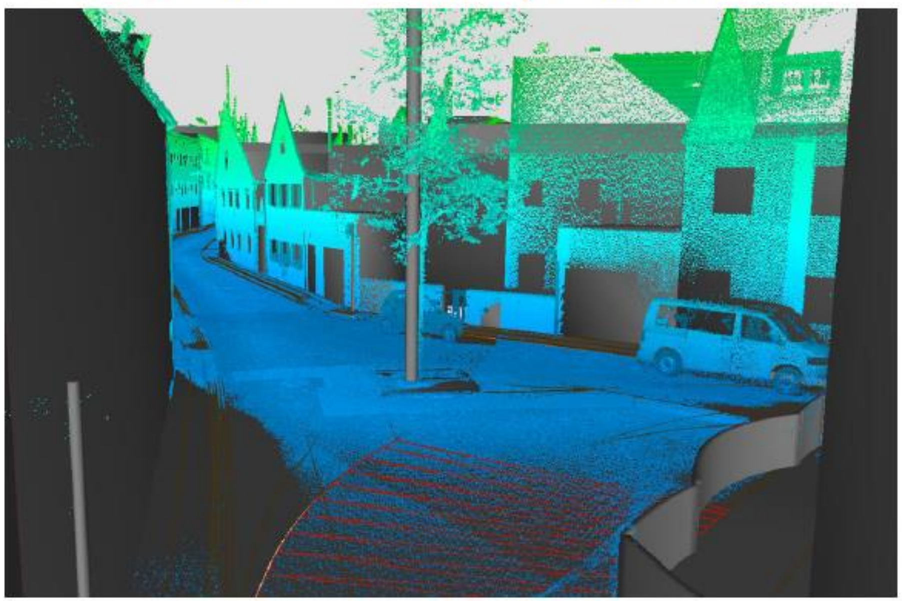
https://www.youtube.com/watch?v=GDf7U8vgyzc

CityGML 3.0 Streetspace Modeling Example 3

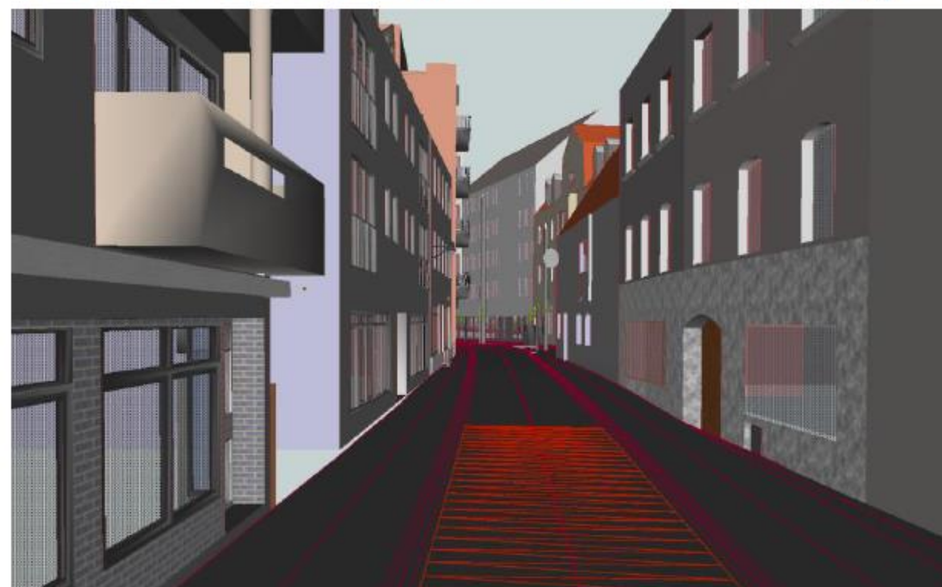
- Dataset examples:
  https://github.com/savenow/lod3-road-space-models

Easy combination of multiple (geo-referenced) datasets

Road model and point cloud



Road model and LOD3 buildings



Common Misunderstandings about CityGML

- ***CityGML is just an exchange format***
  - **NO!** - It is in the first place a conceptual schema on how to represent 3D city models in an interoperable, integrative, and multi-application supportive way
  - Due to historic reasons the normative part of CityGML is its encoding as a GML application schema (there was no standards category for semantic data models in OGC back in 2006 – only abstract or implementation specifications by then)
  - There are alternative mappings of the CityGML conceptual model onto other platforms like spatial databases, JSON, and Graph DB
- ***CityGML is a visualization format***
  - **NO!** – CityGML puts focus on the representation of spatio-semantic characteristics of 3D city and landscape features (Ontology)
  - However, visualizations e.g. in 3DTiles, i3S, X3D, glTF, COLLADA, KML, OpenFlight can easily be generated from CityGML

(Some) Advantages of CityGML compared to (pure) visualization formats

- Extensive semantic, topological, temporal and geometric concepts
- Geo-referenced data (important for large infrastructure objects where the Earth's curvature cannot be neglected)
- Easy combination with other (geo-referenced) data
- Many cities worldwide model their 3D city models according to CityGML
- Database solution available (3DCityDB)
  - Spatial indices on the data
  - Spatio-semantic queries and analysis
- Searchability of objects and semantic information
  - Number intersection types
  - Relations between signs of a certain type next to vegtation
  - Road layout identification
- Association of sensor observations with semantic 3D models

Extending CityGML

- CityGML extension mechanisms
  1. **Generic classes and attributes**
  2. **Application Domain Extensions (ADE)**

     Formally specified augment data model with additional concepts required by use-case (e.g. new properties or object types)

- Models can be enriched with additional information (generic attributes) very easily
  - Reflective information
  - Number of recognitions by vehicles
  - Speed limit
  - Intersection type
  - …

CityGML subsets

- CityGML is application independent and used by multiple disciplines and is useful to several communities
- Thus, CityGML contains many concepts and a powerful but complex data model

However,

- “Implementations may employ a subset of constructs according to their specific information needs.” [CityGML 3.0 specification, 2021]
- Suggestion: Subset of relevant objects and concepts
  - Only used subset required to OpenDRIVE community

Conclusion

- **Revised** and **extended** concepts for modelling transportation Infrastructure in the context of semantic 3D city models
  - Concepts for **geometric and semantic segmentation**
  - Geometric representations: **linear, areal, volumetric or point cloud**
  - Concept for representing city objects using **Spaces**
  - Multiple **levels of granularity** (down to lane level)
  - **Integrated representation** for multiple transportation infrastructure (roads, railways, footpaths, waterways, etc.)
  - Including detailed information bicycle paths and pedestrian areas
- A number of use cases benefit from these concepts

Relevant links

- CityGML3.0
  https://www.ogc.org/standards/citygml
- Guideline explaining CityGML 3.0 Transportation Module
  https://tum-gis.github.io/road2citygml3/
- Interactive Streetspace Demos
  https://wiki.tum.de/display/gisproject/Online+Demo+Collection
- Data Downloads
  https://wiki.tum.de/display/gisproject/Download+Section
- 3DCityDB
  https://www.3dcitydb.org/3dcitydb/
- Open Source Converter r:trån: OpenDRIVE to CityGML
  https://rtron.io/

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

Additional Information

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