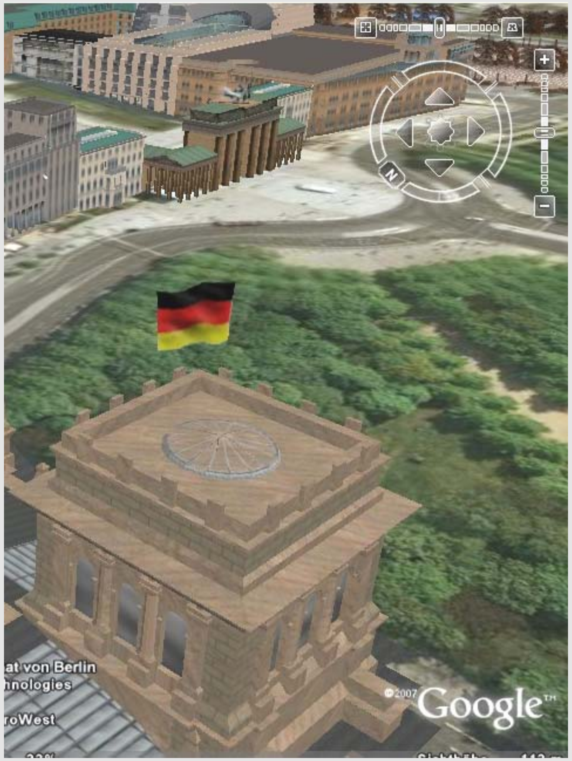
CityGML Tutorial



Prof. Dr. Thomas H. Kolbe

Institute for Geodesy and Geoinformation Science
Berlin University of Technology
kolbe@igg.tu-berlin.de

27th of August, 2007

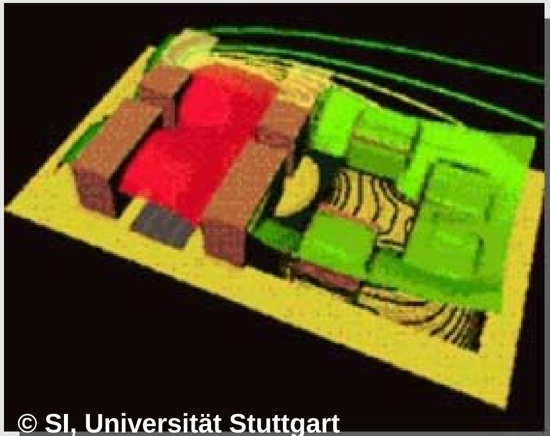
1st Joint Workshop on the Sino-Germany Bundle Project
“Interoperation of 3D Urban Geoinformation“ in Urumqi, China

Overview

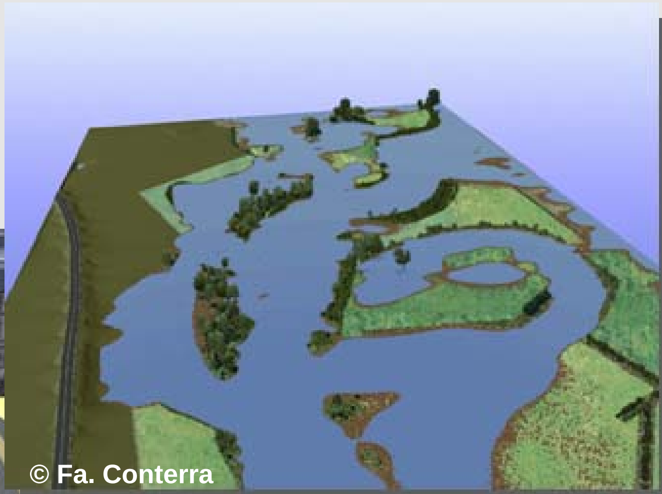
- Introduction: Urban Information Modelling
- CityGML overview and status
- OGC Geography Markup Language (GML)
- CityGML details
- Extending CityGML
- Application examples
- Relations to other standards
- Summary

Urban Information Modelling

Applications of Virtual 3D City Models


© SI, Universität Stuttgart


© IKG, Universität Bonn


© Fa. Conterra


© T-Mobile


© Rheinmetall Defence Electronics

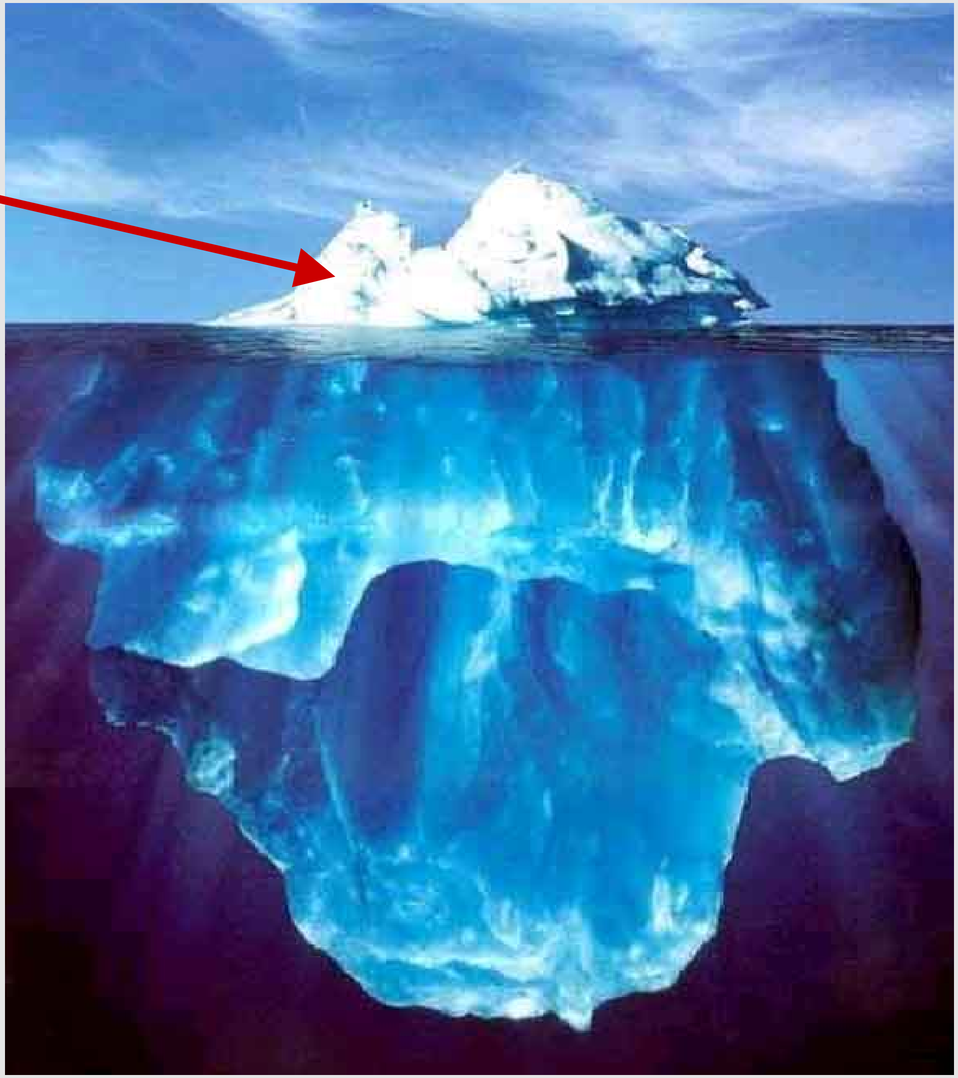

© Sony Corporation

3D City Modelling

… is far more than
the 3D visualization
of reality

In fact, the **geometry** and its **appearance** are **only one aspect** of an entity!

Key issue:
Semantic Modelling



Geospatial Information Modelling

- Ongoing paradigm shift in spatial modelling:
  - **from geometry / graphics** oriented models
  - **to representation of well-defined objects** with their properties (among them spatial and graphical ones), structures, and interrelationships
- Concerning 2D data: long tradition in European cadastres
  - Germany: ALKIS/ATKIS/AFIS (AAA)
  - UK: Ordnance Survey Mastermap
  - Netherlands: Top10NL
- Concerning **3D data: often seen as being identical with 3D graphics models** of the respective region
  - Google Earth [KML, COLLADA], X3D, 3D PDF, 3D Studio Max
- However: numerous **applications beyond 3D visualization**

3D City and Landscape Models

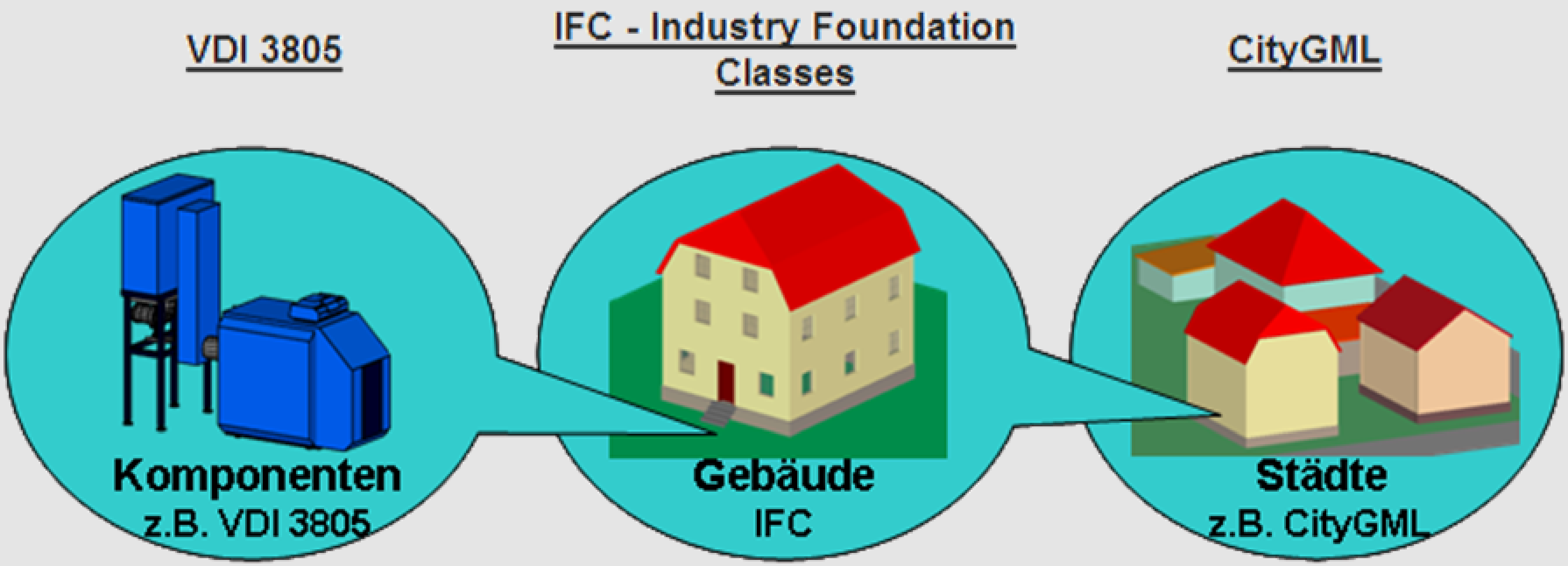
- are **a product family on their own** (like Building Information Models, BIM, are a product family)
- with specific applications (differing from BIM)

Characteristics

- complete representation of city topography / structures **‘as observed‘** (typically **not ‘as planned‘**)
  - often full spatial coverage of a city or district
  - built-up environment (buildings, infrastructure)
  - natural features (vegetation, water bodies, terrain)
- 3D geometry, topology, semantics, and appearance
- homogeneous data quality (at least on the same scale)

Information Modelling at Different Scales

- Model content, structure, and employed modelling principles depend on
  - Scale
  - Scope (application contexts)



Taken from the Homepage of the Helmholtz Research Center Karlsruhe, © Karl-Heinz-Häfele

CityGML

Overview & Status

Application independent Geospatial Information Model

for virtual 3D city and landscape models

- comprises **different thematic areas** (buildings, vegetation, water, terrain, traffic etc.)
- **data model (UML)** acording to **ISO 191xx** standard family
- exchange format results from rule-based mapping of the UML diagrams to a GML3 application schema
- ongoing standardisation process in OGC

CityGML represents

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

CityGML Development

Department of Geoinformation Science

Originator: SIG 3D of the Initiative Geodata Infrastructure North-Rhine Westphalia in Germany **GDI NRW**

- **Open group** of more than 70 parties / institutions working on technical and organizational issues about virtual 3D city models
- T-Mobile, Bayer AG, Rheinmetall Defence, Environmental Agencies, Municipalities, State Mapping Agencies, UK Ordnance Survey, 11 Univ.

CityGML was brought into **Open Geospatial Consortium** for international standardisation by the end of 2004

- Handled by the **3D Information Modelling Working Group (3DIM WG)**
- Current status: OGC Best Practice Paper [since July 2007]
- Roadmap: International Standard [December 2007]

Goals of CityGML (I)

Establish **high degree of semantic** (and syntactic) **interoperability**

- enabling multifunctional usage of 3D city models
- definition of a **common information model (ontology)**
- „3D geo base data“ (in the tradition of most European 2D digital landscape models, cadastre models)

Representation of **3D topography** as observed

- explicit 3D shapes; mainly surfaces & volumes
- identification of **most relevant feature types** usable in a **wide variety of applications**
- limited inclusion of functional aspects **in base model**

Goals of CityGML (II)

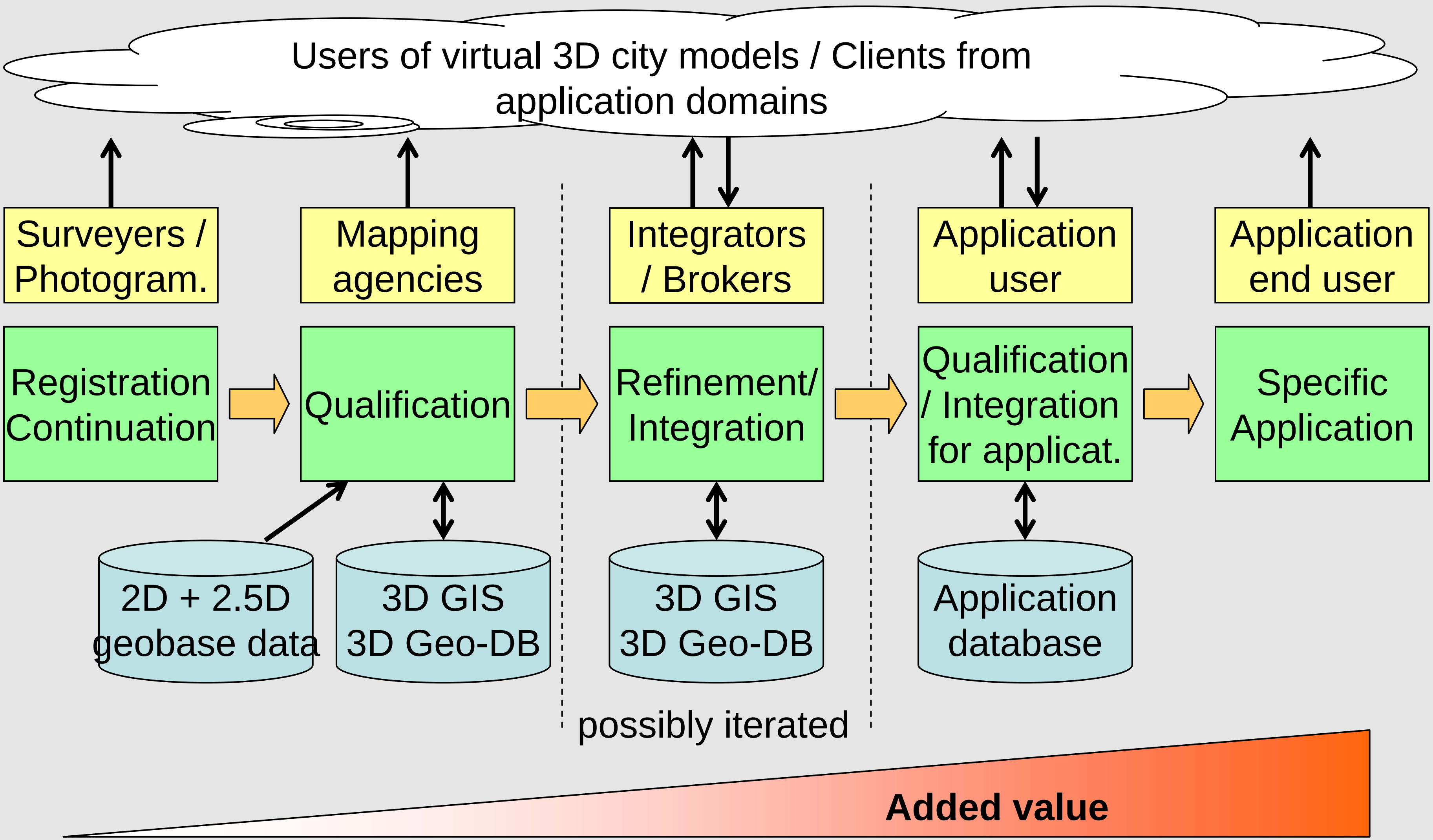
Suitability for **Spatial Data Infrastructures**

- mapping to appropriate exchange format -> **GML3**
  - needs high degree of expressivity wrt. OO models
  - must be usable in the context of OGC Web Services
- possibility to **link any CityGML feature** to more specialised, functional models / external data sources

Must be **simple to use** for applications

- **well-defined semantics** for feature types; however semantic structure not too fine-grained
- subset of GML3 geometries (no curved lines, surfaces)
  - **Boundary representation** with absolute coordinates
  - advantage: **directly manageable** within **3D GIS / geo DB**

CityGML along the Processing Chain



Difficulties along the Processing Chain

- **Diverse qualities** of **3D models** in the different steps
  - different degree of fidelity of geometry, topology, appearance
  - from simple structured objects to complex application models
- Until now: often **change of data models** and **exchange formats** inbetween the processing steps
  - loss of data because of limited modeling powers / expressivity of models and formats
  - difficult preservation of object identities
- **Missing back links / references** to original data of preceding processes
  - causes problems with updates / continuations
- **CityGML can be used along the full processing chain**

CityGML Status

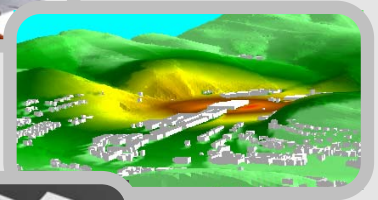
- The **new version (0.4.0)** of the specification document has been adopted as an **OGC Best Practice Paper**
  - at the recent OGC TC Meeting in Paris, July 2007
- New version deprecates version 0.3.0
  - Version 0.4.0 is downloadable from the OGC Homepage (section „Documents“, subsection „Best Practice Papers“)
- Version **0.4.0 is backwards compatible** to V 0.3.0
- Changes + new features: see next slide

Changes from previous version 0.3.0

Department of Geoinformation Science

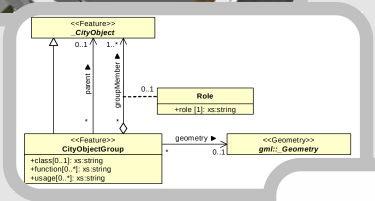
Introduction of a new appearance model

Introduction of Application Domain Extensions (ADE)



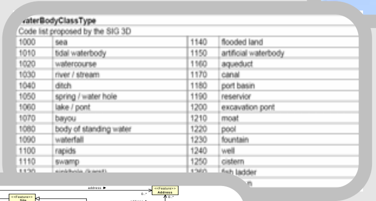
Minor changes to the building model

parent association in *CityObjectGroup*



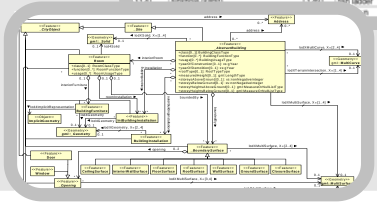
Terrain Intersection Curves (TIC) added to city furniture

Revision of external code lists



1000	sea	1140	flooded land
1010	tidal waterbody	1150	artificial waterbody
1020	watercourse	1160	aqueduct
1030	river / stream	1170	canal
1040	ditch	1180	port basin
1050	spring / water hole	1190	reservior
1060	lake / pont	1200	excavation pont
1070	bayou	1210	moat
1080	body of standing water	1220	pool
1090	waterfall	1230	fountain
1100	rapids	1240	well
1110	swamp	1250	cistern

Revision of UML diagrams



Geography Markup Language

Geography Markup Language (GML)

- GML is an **International Standard** for the **exchange and storage of geodata**
- Issued by the **Open Geospatial Consortium (OGC)**
- Version 3 was released in 2003
  - CityGML is based on (current stable) version 3.1.1
  - Specification freely downloadable from www.opengeospatial.org
- Further development jointly by OGC & ISO: GML 3.2.1 will be published as **ISO Standard 19136**
- Several national topography and cadastre models are already based on ISO 191xx and GML
  - e.g. in Germany, United Kingdom, Netherlands

Design Goals of GML3

- **Open, vendor independent** framework for the definition of **spatial data models**
- **Transport and storage** of schemas and datasets
- Support for the specification of **application schemas**
  - **GML is a meta format;** i.e. concrete exchange formats are specified by GML application schemas (like CityGML)
- Support of **distributed** spatial application schemas and datasets (over the Intra-/Internet)
- Possibility to create **profiles** (subsets of GML3)
- **Facilitate Interoperability** in the handling of geodata

GML3 Overview

- **Object oriented modelling** capabilities
  - Generalisation / specialisation & aggregations
- **Simple and complex geometries**
  - 0D: points
  - 1D: straight lines, splines, arcs
  - 2D: planar surfaces, nonplanar surfaces (spline, NURBS, TINs)
  - 3D: volumes by using Boundary Representation (B-Rep)
  - Composed geometries
- **Topology** (with or without associated geometry)
- **Coordinate** and **time reference systems**
- **Coverages** (regular and irregular rasters, TINs, maps)

Difference to other GIS exchange formats

- Object oriented; facilitates **semantic modelling**
  - In contrast to pure geometry models (like CAD formats or VRML) or geometry oriented GIS models (like Shapefiles):
  - **Identifiable objects** (with ID)
  - Spatial and nonspatial properties
  - **Specialization hierarchies** (taxonomies)
  - **Aggregation hierarchies**
  - **Associations** / relations between objects
- **Mixed usage** of **different spatial reference systems** within the same dataset possible
- **XML based**

CityGML

Details

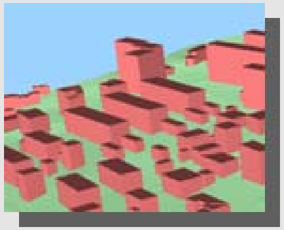
Multi-scale modelling: 5 levels of details

LOD 0 – Regional model

- 2.5D Digital Terrain Model

LOD 1 – City / Site model

- “block model“ w/o roof structures

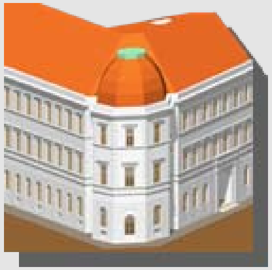


LOD 2 – City / Site model

- textured, differenciated roof structures

LOD 3 – City / Site model

- detailed architecture model

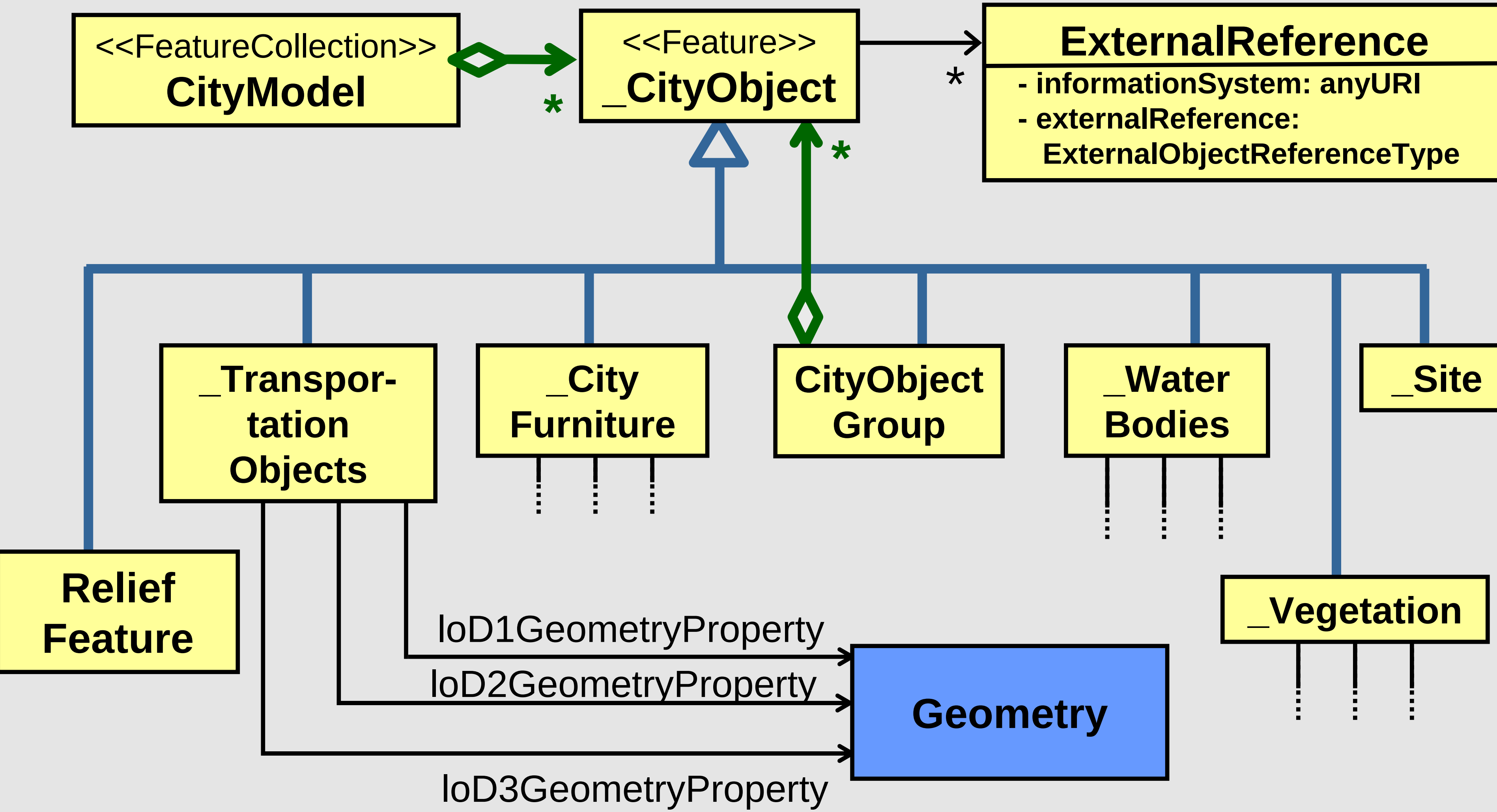


LOD 4 – Interior model

- “walkable“ architecture models

Thematic Modelling in CityGML

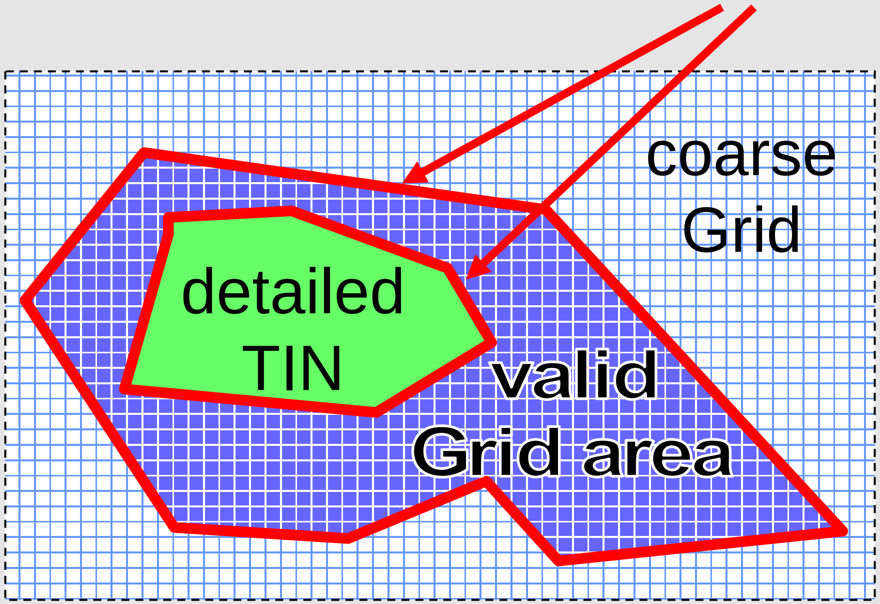
Department of Geoinformation Science



Digital Terrain Models

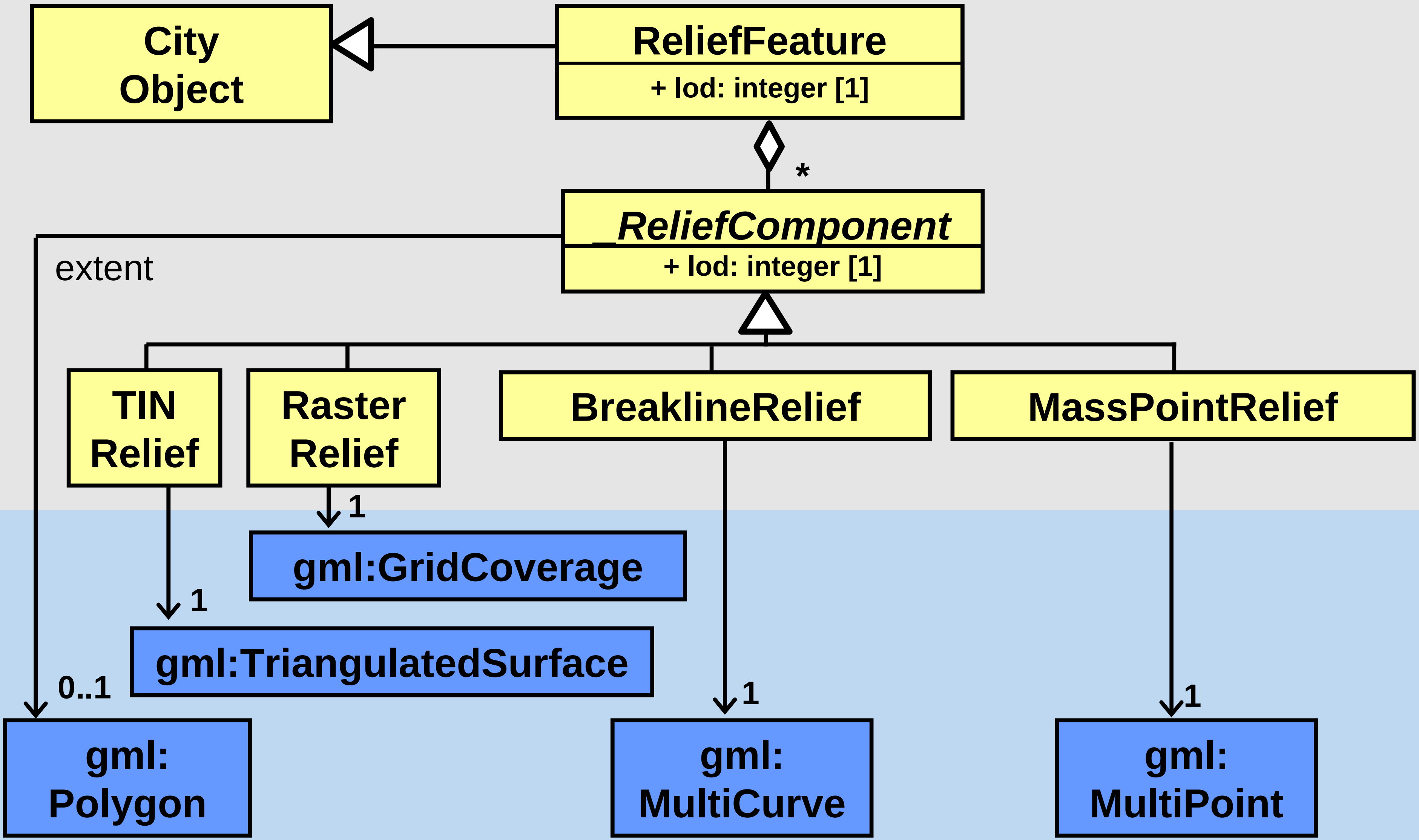
DTM for each Level of Detail can be composed of

- **TINs** (Triangulated Irregular Network), **Grids**, **3D Breaklines**, and **3D Mass Points**
- Each DTM component may be restricted to be valid in a specific region by providing a **validity extent polygon**

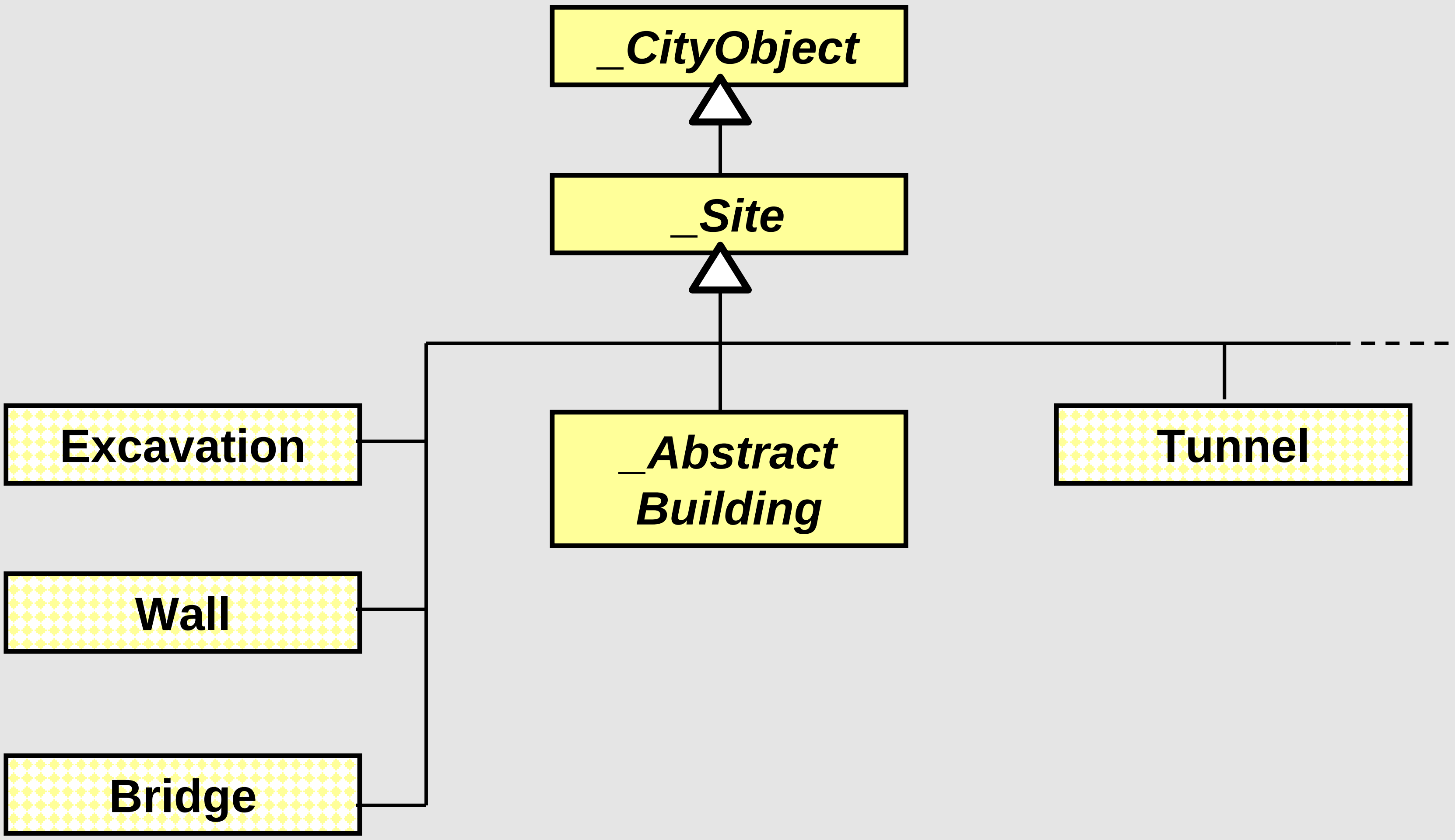


Validity extent polygon can have holes which allow **nested DTMs!**

Digital Terrain Model: UML Diagram



Site Model

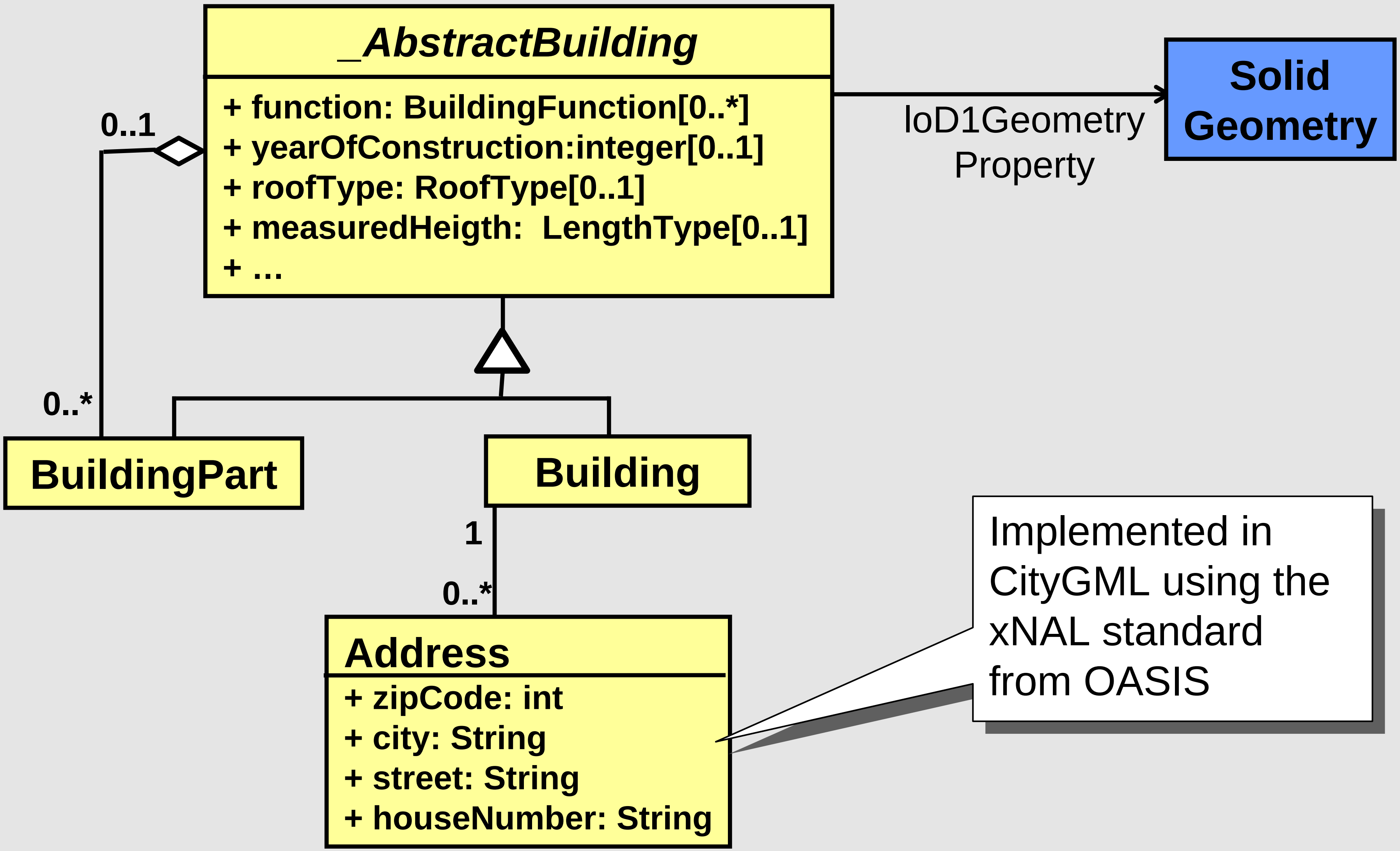

_CityObject
_Site
Excavation
_Abstract Building
Tunnel
Wall
Bridge

Building Model

Department of Geoinformation Science

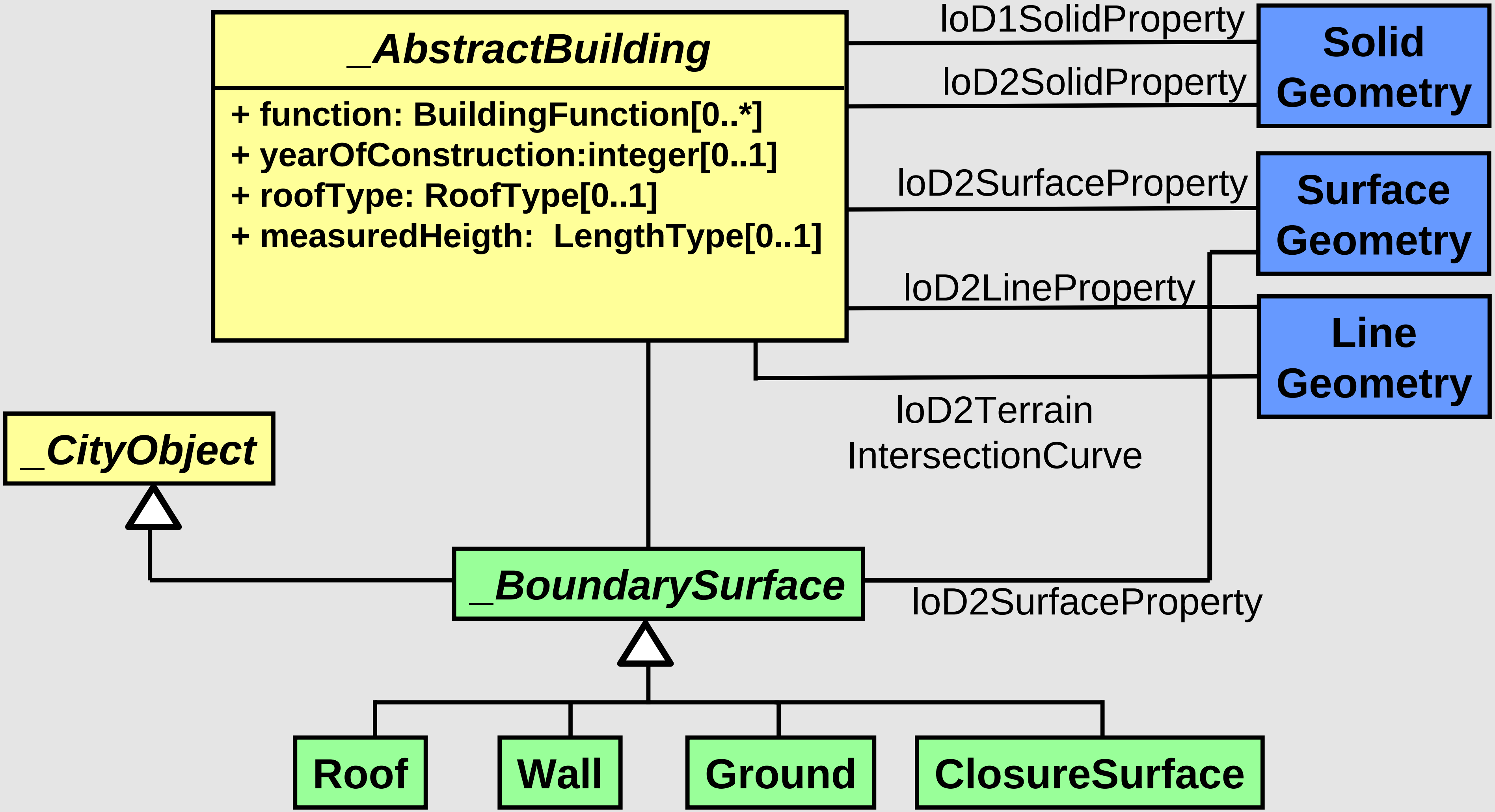
- Coherent aggregation of spatial and semantical components
  - (recursive) composition of **building parts**
  - **thematic surfaces** (roof surface, wall surface, etc.) [from LOD2]
  - **building installations** like dormers, stairs, balconies [from LOD2]
  - **openings** like doors and windows [from LOD3]
  - **rooms** and **furniture** [in LOD4]
- Components contain relevant **thematic attributes**
  - name, class, function, usage, construction and demolition date, roof type, address
  - no. of storeys above / below ground, storey heights

Building Model in LoD1: UML Diagram

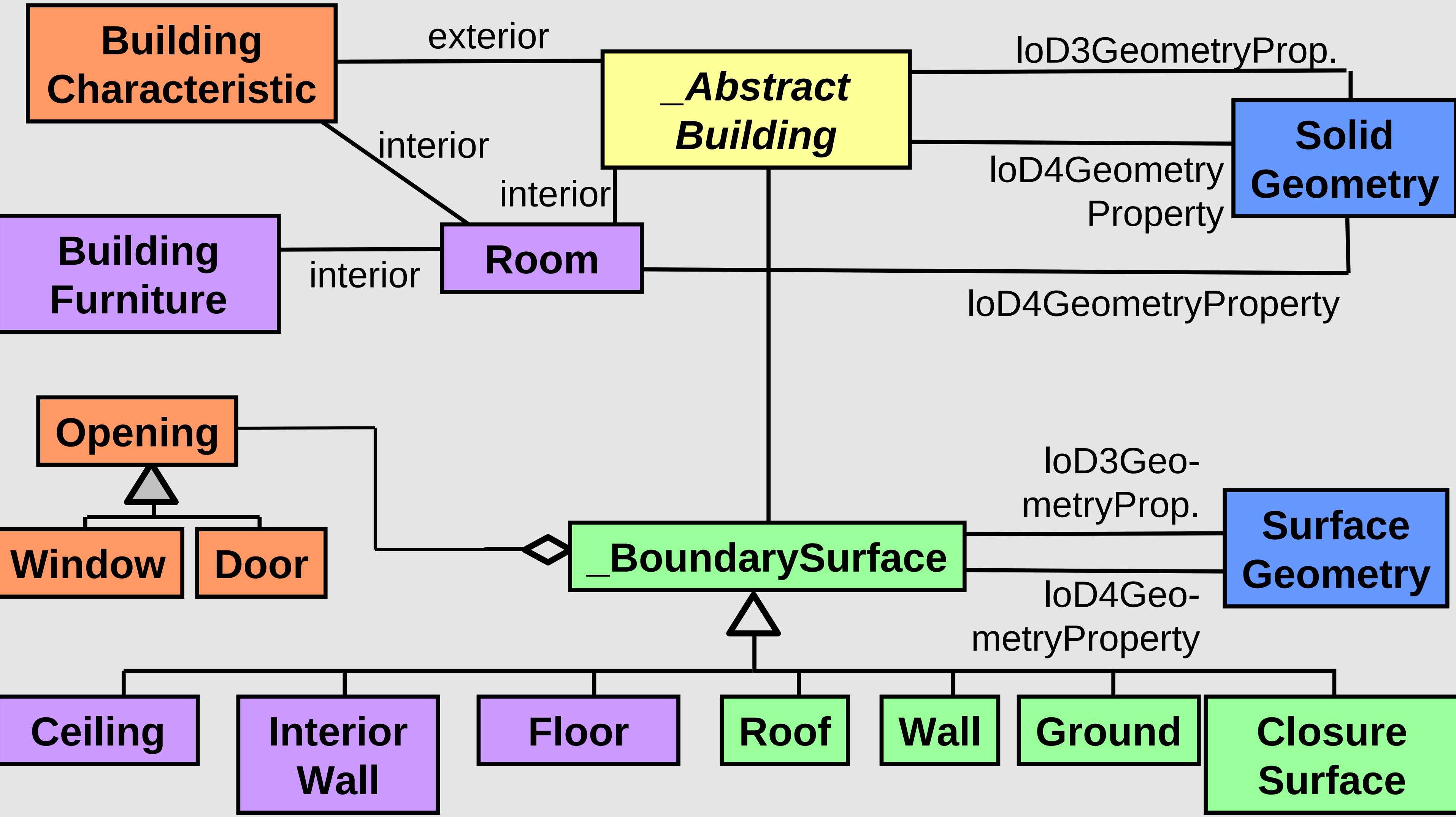


Building Model in LoD2

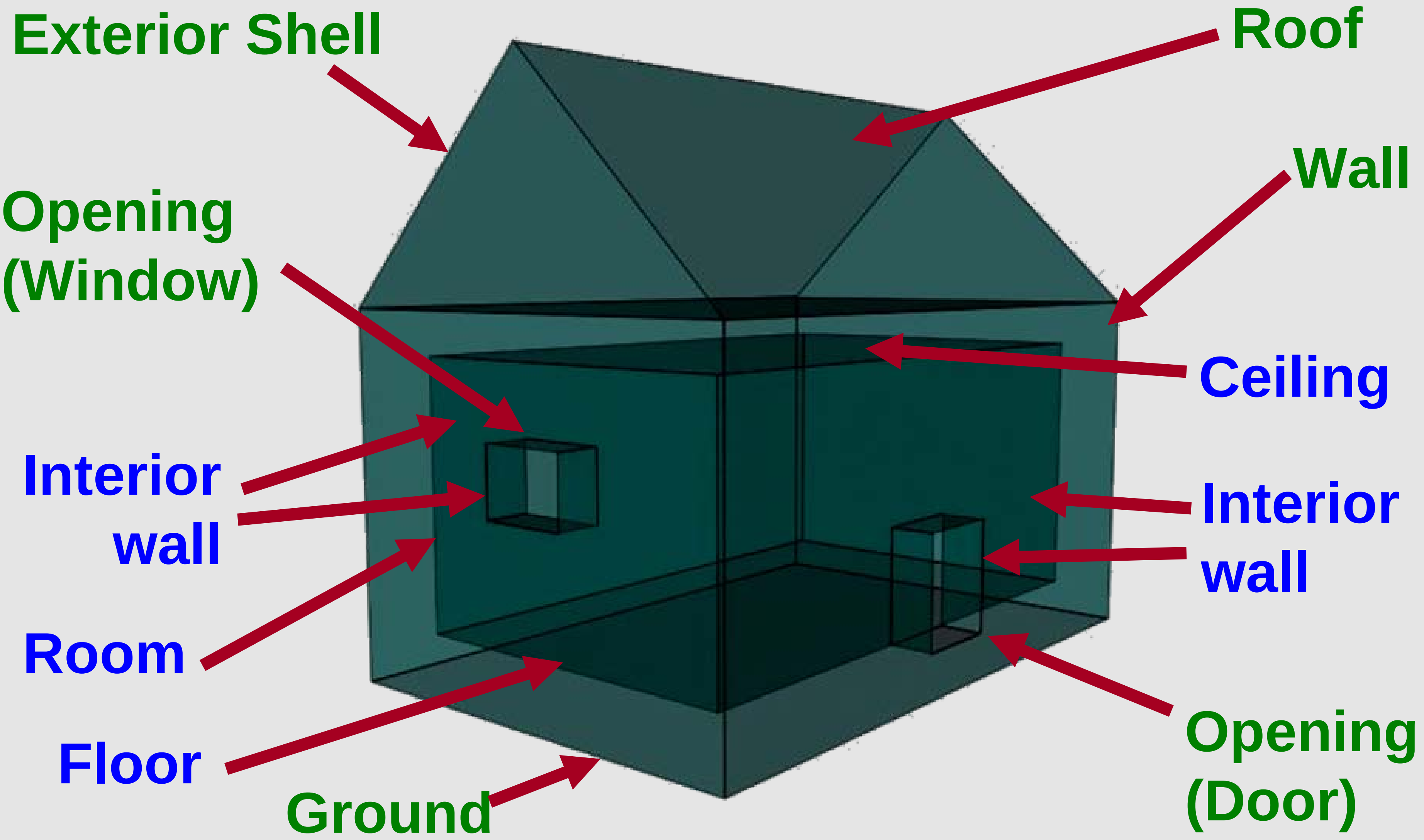
LoD1 LoD2



Building Model in LoD3+4



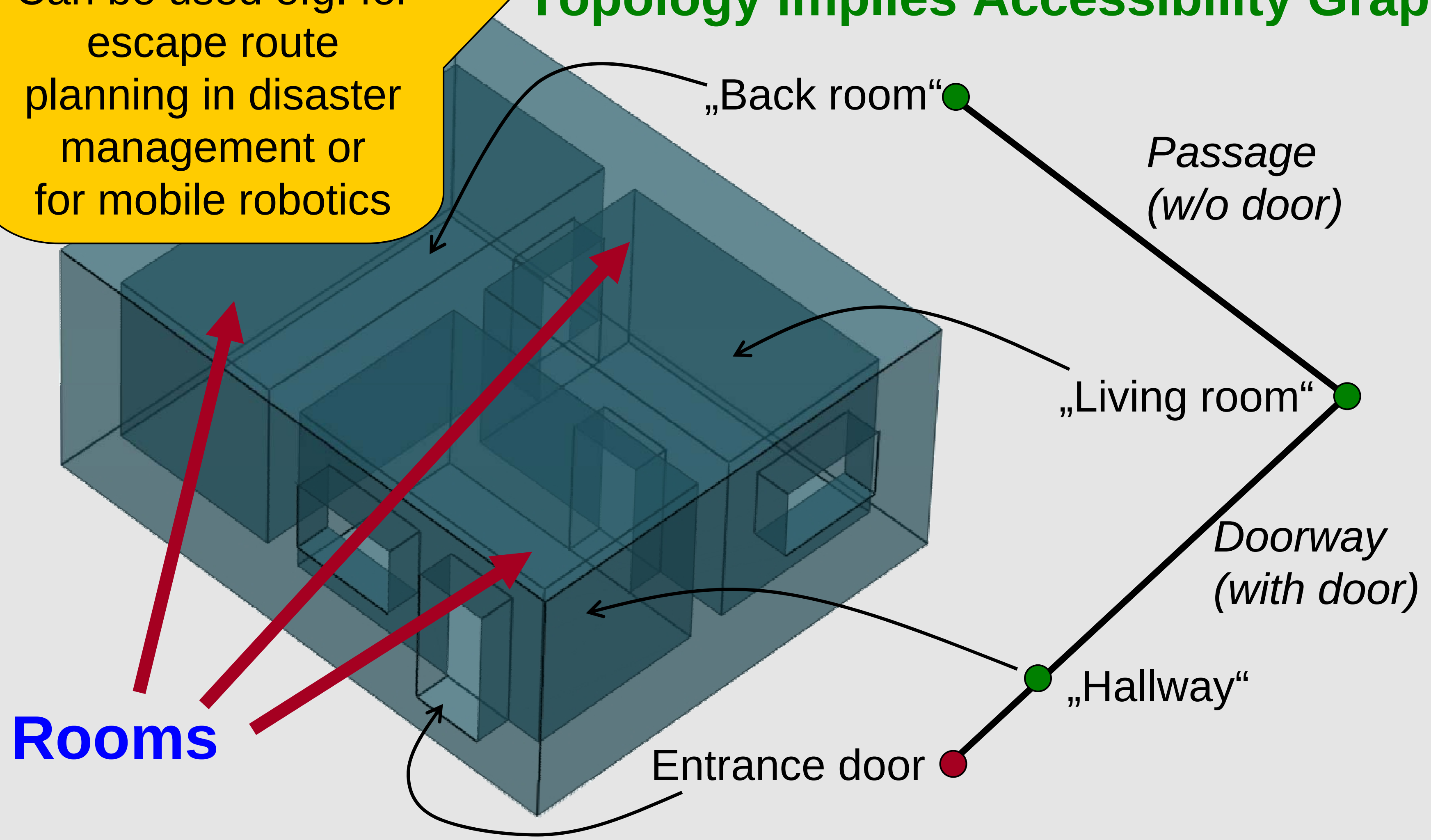
Building Features in LoD4



Building in LoD4 – Interior Model

Can be used e.g. for escape route planning in disaster management or for mobile robotics

Topology implies Accessibility Graph !



Example for CityGML file structure


```
<?xml version="1.0" encoding="UTF-8"?>
<CityModel xmlns="http://www.citygml.org/citygml/1/0/0“ ...further namespaces omitted>
   <gml:name>Cologne</gml:name>
   <gml:boundedBy>
        <gml:Envelope
                 srsName="urn:ogc:def:crs,crs:EPSG:6.12:31466,crs:EPSG:6.12:5783">
              <gml:pos>  5659800.0   2561800.0   15.9  </gml:pos>
              <gml:pos>  5662200.0   2564200.0   95.7</gml:pos>
        </gml:Envelope>
   </gml:boundedBy>
   <!-- now come the CityObjects like Buildings, DTM, Roads etc. -->
   <cityObjectMember>
        <Building gml:id=“Building0815">
        <!-- shown on following slides -->
        </Building>
   </cityObjectMember>
   <!-- more CityObjects here -->
</CityModel >
```

Combined horizontal and vertical CRS

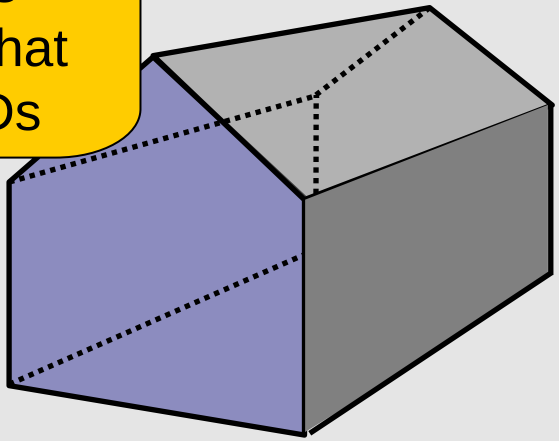
Bounding volume of the whole city model

Example: Simple Building in CityGML 1/3


```
...
  <Building gml:id="Building0815">
      <gml:name>My nice building</gml:name>
      <externalReference>
          <informationSystem>http://www.adv-online.de</informationSystem>
          <externalObject>
              <uri>urn:adv:oid:DEHE123400007001</uri>
          </externalObject>
      </externalReference>
      <function>1012</function>
      <yearOfConstruction>1985</yearOfConstruction>
      <roofType>3100</roofType>
      <measuredHeight uom="m">8.0</measuredHeight>
      <lod2Solid>
          <!-- geometry (for Level of Detail 2) see next slide -->
      </lod2Solid>
  </Building>
...
```

Example: Simple Building in CityGML 2/3

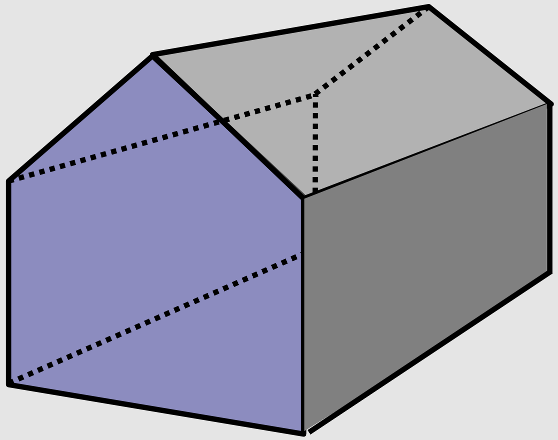
Please note that geometries are objects that can have IDs



```
<!– continued from previous slide -->
<lod2Solid>
    <gml:Solid gml:id=“solid0815" >
        <gml:exterior>
            <gml:CompositeSurface>
                <gml:surfaceMember>
                    <gml:Polygon gml:id=“polygon4711">
                        <gml:exterior>
                            <gml:LinearRing>
                                <gml:pos> 5660398.399  2562509.711  41.79 </gml:pos>
                                <gml:pos> 5660402.019  2562514.546  41.79 </gml:pos>
                                .........................
                            </gml:LinearRing>
                        </gml:exterior>
                    </gml:Polygon>
                <gml:surfaceMember>
        <!-- further surfaces of the solid; closing tags omitted due to limited space -->
</lod2Solid>
```

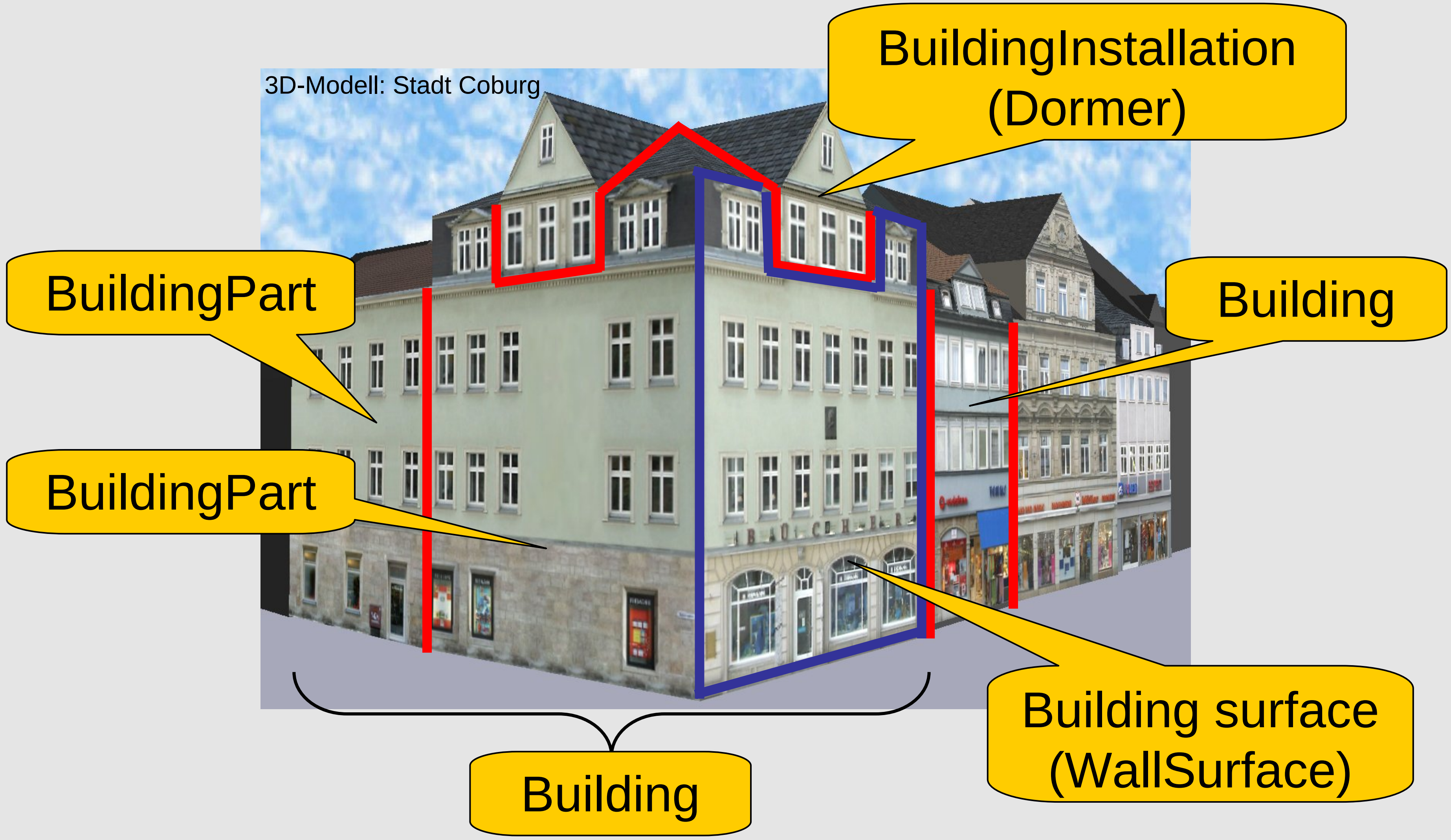
Example: Simple Building in CityGML 3/3


```
<Building gml:id="Building0815">
  <lod2Solid>
    <gml:Solid>
      <gml:exterior>
        <gml:CompositeSurface>
          <gml:surfaceMember>
            <!-- front surface as in
                 previous slide -->
          </gml:surfaceMember>
          <gml:surfaceMember>
            <!-- side surface -->
          </gml:surfaceMember>
            <!-- here come side, back, roof, and ground surfaces -->
        </gml:CompositeSurface>
      </gml:exterior>
    </gml:Solid>
  </lod2Solid>
</Building>
```

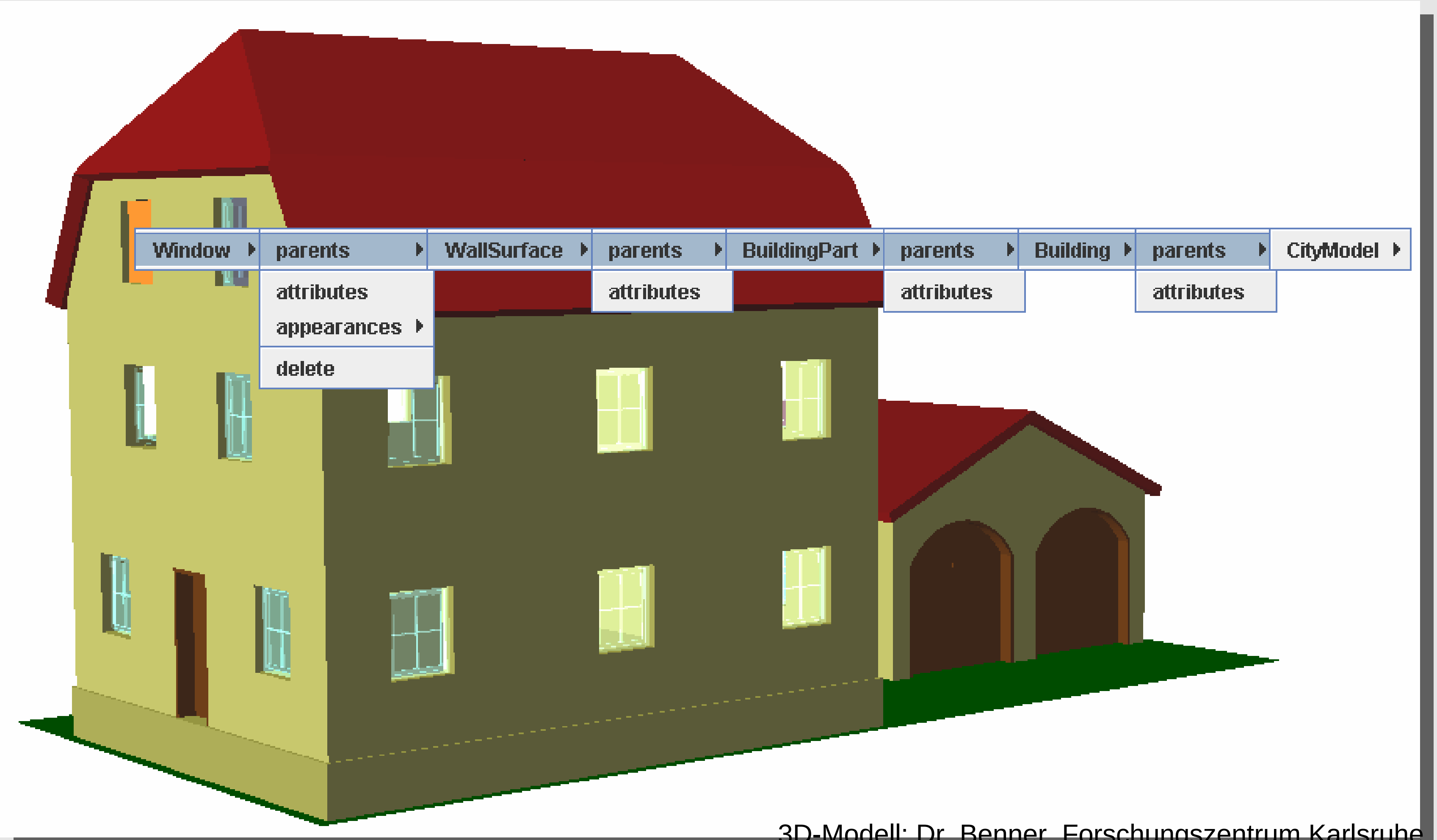


Spatio-semantical Composition

Department of Geoinformation Science

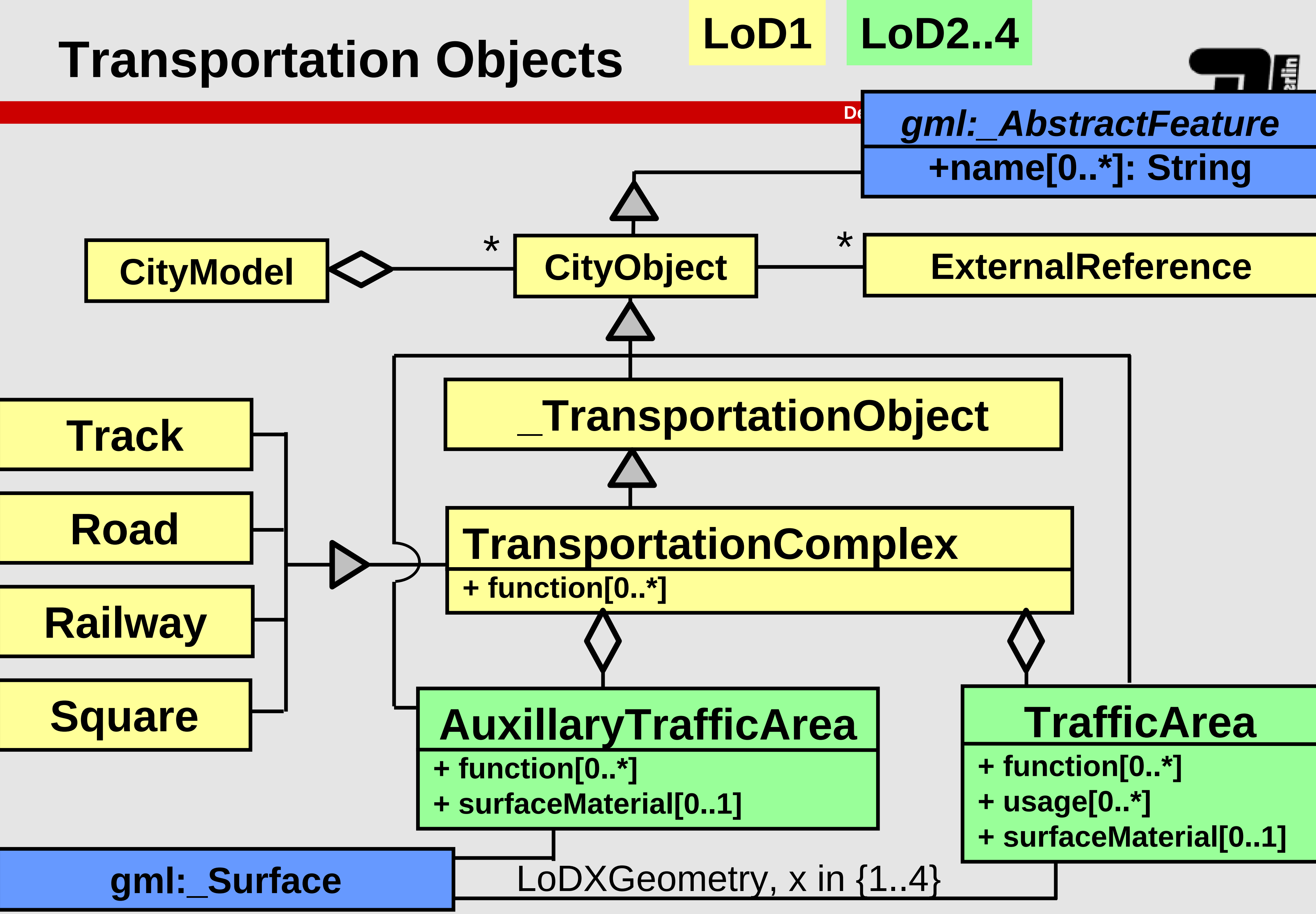


Coherent Building Model in Level of Detail 3

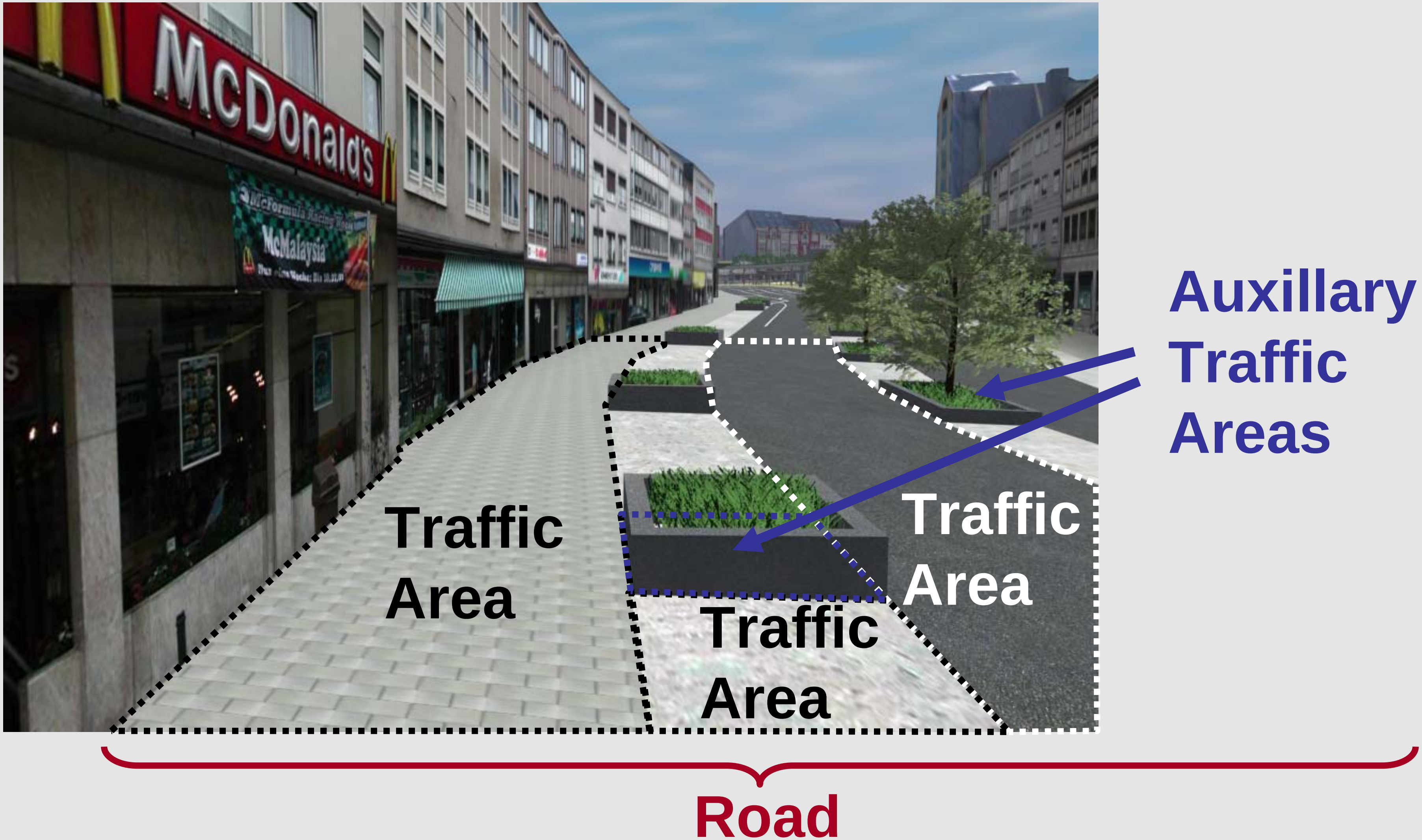


3D-Modell: Dr. Benner, Forschungszentrum Karlsruhe

Transportation Objects



Example: Transportation Model in LoD2



Water Bodies

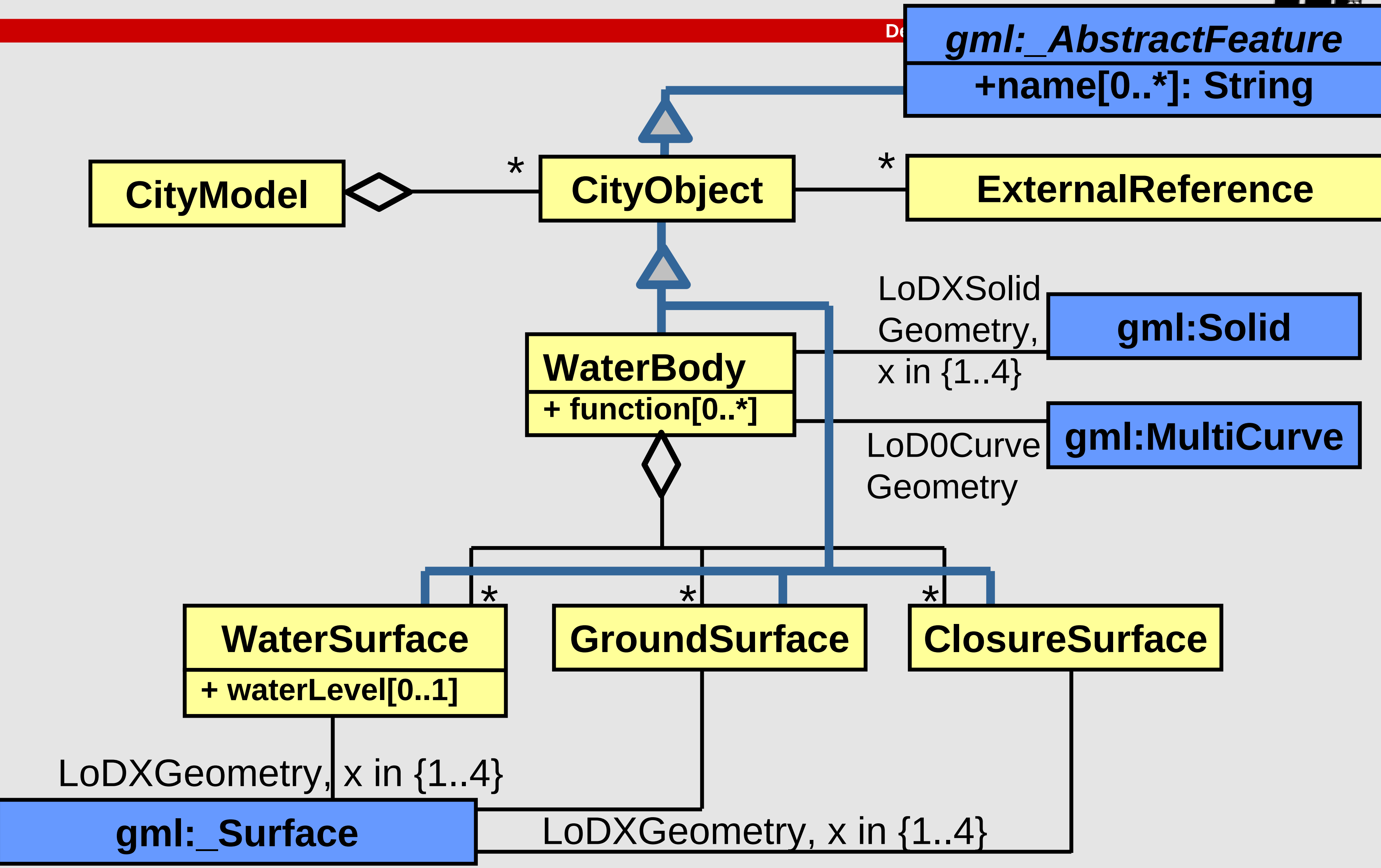
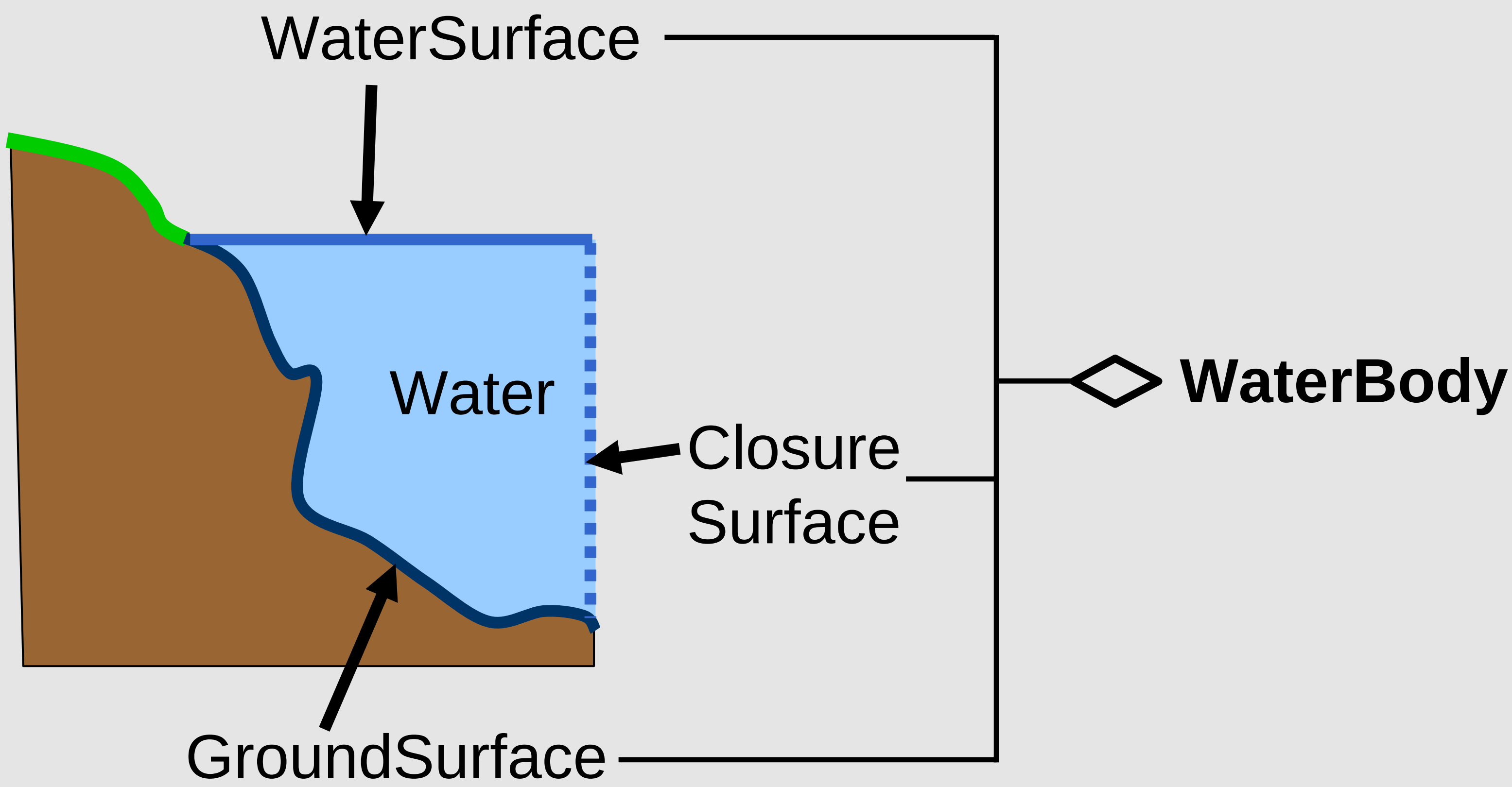


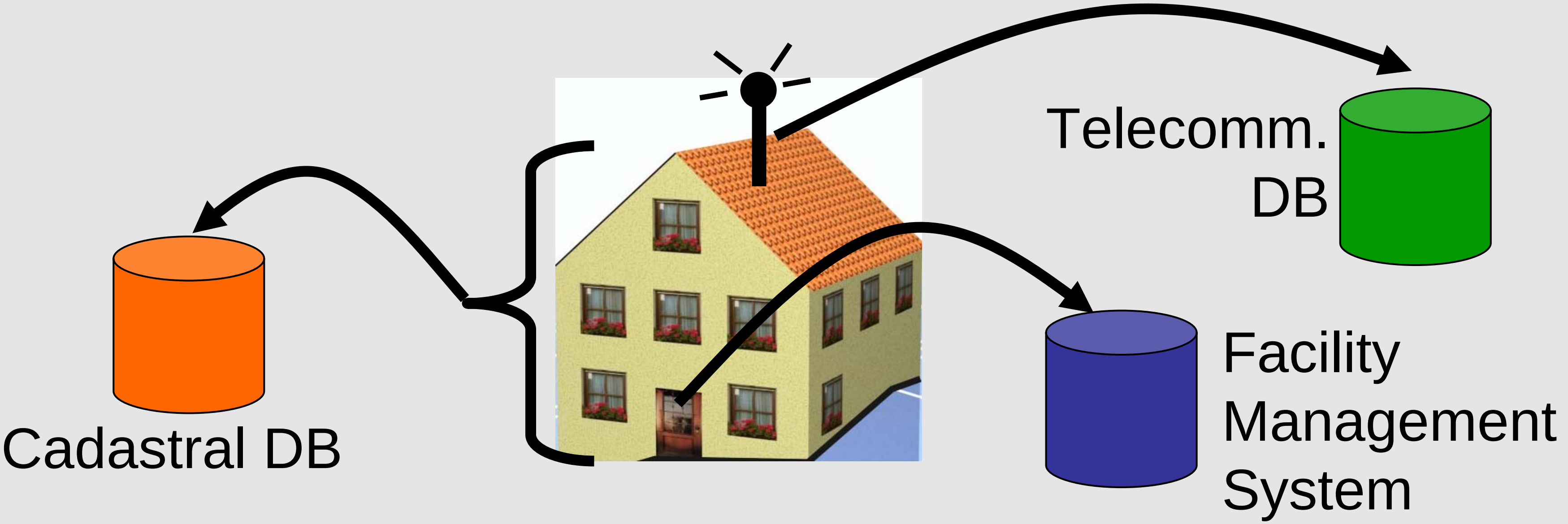
Illustration of a Water Body



Further CityGML Concepts

- Support for **generalization of 3D data**
  - Generalized objects are linked to the original objects on the larger scale
- Explicit **linking**
  - Every CityGML object can have an arbitrary number of links to external resources (files, objects, database entries)
- Object **history**
  - Objects may have a lifespan (incl. termination date)
- Support for spatial homogenization / integration
  - e.g. **Terrain Intersection Curves** (for integration of 3D objects with the terrain)

External References



Every object (part) may have **references** to **corresponding objects** from **external resources**

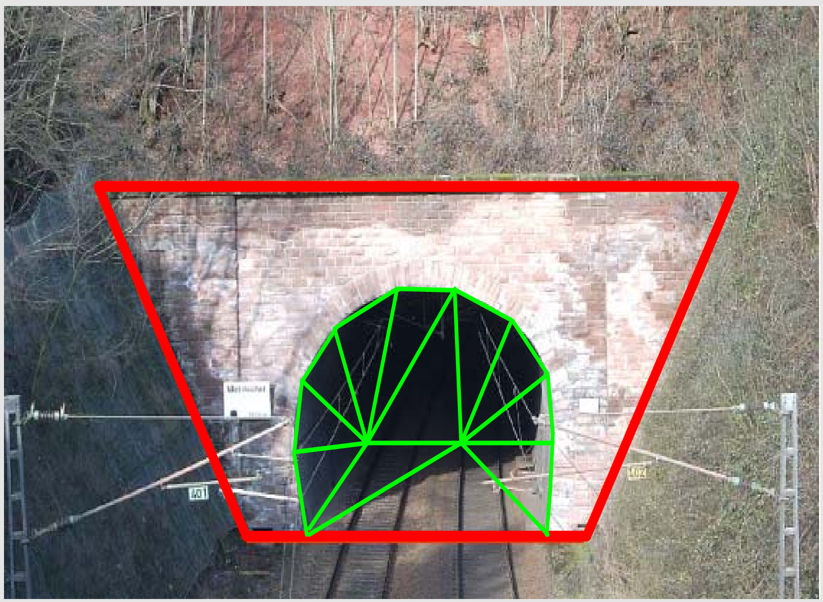
Connection with external information, e.g.:

- building: link to cadastre, owner's contact information
- door, antenna: link to facility management systems

Terrain Intersection Curve (TIC)

„Interface between 3D objects and the terrain“

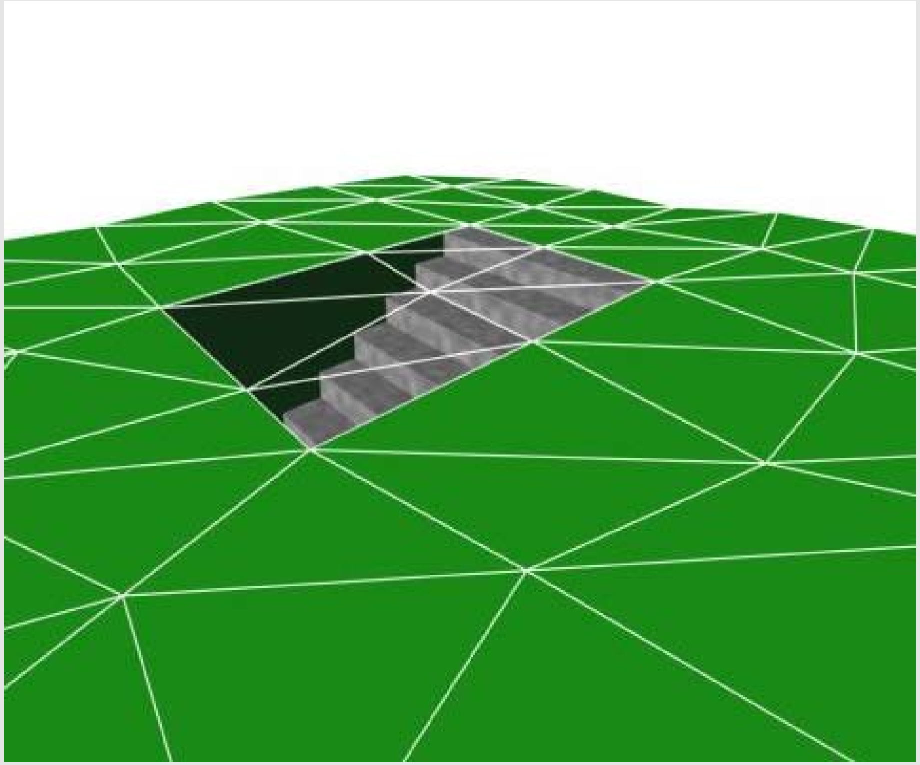
- ensure matching of object textures with the DTM
- DTM may be locally warped to fit the TIC



Closure Surfaces

„Seal open 3D objects“

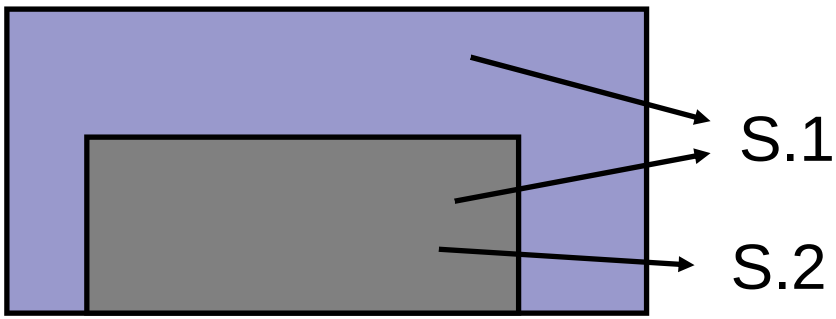
- in order to be able to compute their volumes



recursive aggregation

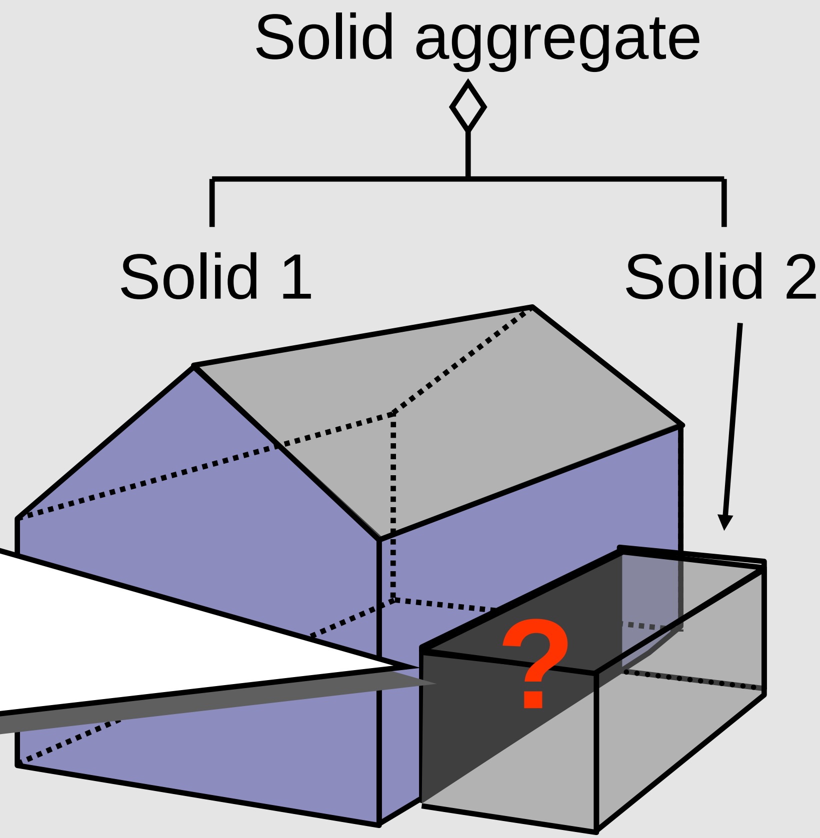
➔ arbitrary depth

- Wall face should be partitioned into 2 faces



➔ explicit topol. connection

- but: goes beyond B-Rep



“Backdoor Topology“

How to allow for **flexible usage of topology?**

- until now, most 3D city models do not consider topology
- need to represent city models with geometry only

Topology model of GML3 sophisticated, but complex

- would make it necessary to implement 2 options for the representation of spatial properties

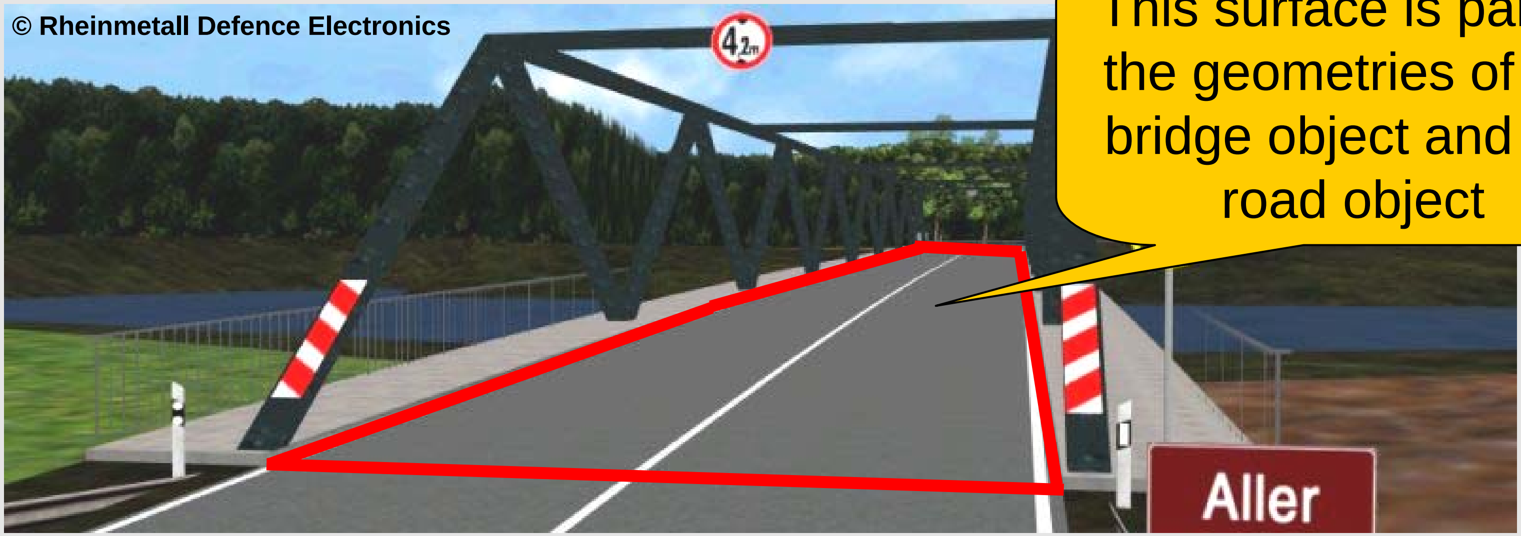
Approach in CityGML:

- **topological connections** are represented by **Xlinks**
- GML3 geometries are objects; composites/aggregates can include subgeometries by value or by reference;
- references express topological relations

Semantic Relations by Topology

Multiple referencing of geometry (components) by distinct geospatial features (from different feature classes)

- realizes topological, but also semantic relations
- redundancy free description of space and surfaces possible, thus no overlaps occur



This surface is part of the geometries of the bridge object and the road object

CityGML Extensions to Standard GML3

Surface Materials

- Colors, Textures (adopted from X3D & COLLADA)
- Appearance information can be assigned to any surface

Implicit geometries (Prototypic shapes)

- Shape of a 3D object in local coordinates
- Instancing at anchor points (+ further transformations)

Both are concepts used in scene graphs

- directly transformable to VRML, X3D, U3D etc.
- however **only simple & limited extensions**
- tailored to the demand of 3D city models
- easy to support by exporting / importing applications

Prototypic 3D Objects

Department of Geoinformation Science

3D city models often contain large numbers of geoobjects of identical shape but at different locations

- Examples: trees, traffic lights, street lamps, benches, etc.

in GML3, all geometries have absolute coordinates

- every copy / instance would have to be explicitly represented

CityGML: **Implicit Geometries**

- Separation of shape definition and georeferencing (anchor point + transform.)
- Comparable to scene graph concepts

Grouping of CityObjects

Feature type **CityObjectGroup**

- has **arbitrary CityObjects** as members

CityObjectGroup is a CityObject

- can become again member of another group
- every member can denote its role in a group

usable for **user-defined aggregations**

- e.g. results of classifications or selection

usable also to **group** CityObjects **wrt.** some **function or area**, e.g.

- city districts, building storeys, or evacuation areas

The new appearance model (since V 0.4.0)

Department of Geoinformation Science

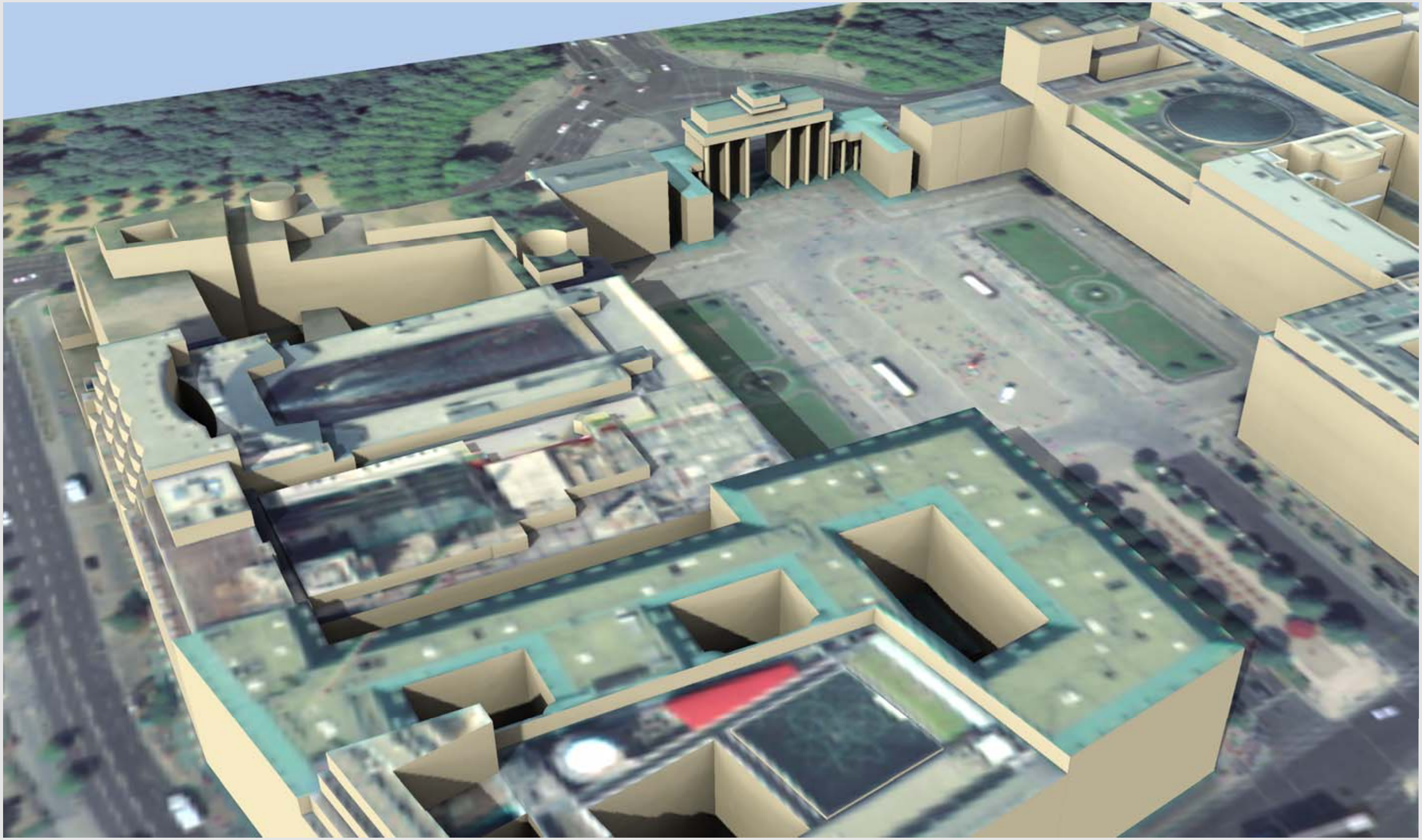
Reasons

- Extension of the GML3 geometry model by class *TexturedSurface*
- Textured terrain unsupported
- Georeferenced textures unsupported
- Material model limited to a single visual surface property

Consequences

- Material model of CityGML 0.3.0 still supported but marked as deprecated
- Introduction of georeferenced and parameterized textures; multiple appearances per object
- Lossless conversion to new appearance model possible
- Existing CityGML 0.3.0 instance documents are still valid

New: Georeferenced Textures



New: Parameterized Textures

Georeferenced Photography:

Projected onto 3D surfaces:

using *worldToTexture* parameterization

(Some)
CityGML Implementation Issues

CityGML Implementation Issues (I)

- **(City)GML files** become **very large** (several GB for bigger cities)
  - file sizes can be effectively reduced by gzip compression (≈10%)
  - but: XML validation and processing can be problematic (classical DOM parsing not feasible due to main memory limitations)
  - WFS access might have to be realized in an asynchronous way in order to avoid timeouts
- **Complex data model**
  - extensive use of OO modeling -> puts **considerable demands on the modelling power** of processing and storage components
  - Aggregation hierarchies: nested objects
  - Specialization hierarchies: inheritance of object properties

CityGML Implementation Issues (II)

- **XLinks**
  - Complex objects can be represented inline, in a self-contained way
  - But: **sub-objects may be also distributed** over different files (even Web Services) and only referenced by their parent objects
  - GML object referencing employs the XLink standard of the W3C
- **Topology**
  - topological relations are realized by reusing (partial) geometries;
  - reusage: referencing the same geometry from different objects
  - referencing uses XLinks, referenced geometries need to have IDs
- **Geometry Model**
  - See next slide

Geometry Model of ISO 19107 / GML3

- **3D GML geometries** are represented as **B-Rep** with absolute (world) coordinates (but always **with CRS**!)
  - no scene graph concepts like transformation nodes
  - the CRS is (one) key to the integration of different spatial datasets
- **No generative modeling** principles like CSG, Sweep Repr.
  - Very few implicit (parametric) shape definitions (e.g. Box, TIN)
- Reusability of geometry within a dataset is limited
  - However useful to express topological connectivity of different features or semantic relations between them
- **Advantages** of the GML3 geometry model
  - easy to spatially index and manage within spatial databases and GIS; native support by Oracle, PostGIS, MySQL etc.
  - visualization (transformation to X3D) is immediate

Extending CityGML

Extending CityGML

Department of Geoinformation Science

1. Generic Attributes & *GenericCityObjects*

- every ***CityObject*** can have an arbitrary number of **extra attributes**
  - allows to extend objects like Buildings, Roads, etc. without the need of new application schemas
- **GenericCityObjects** can have arbitrary geometries (and generic attributes) for every LOD
- “extension during runtime“

2. Application Domain Extensions (ADE)

- extra XML schemas referring to the CityGML XML schema (defined by **information communities**)
- extensions to be **formally specified in XML schema**

Generic CityObjects and Attributes

- Explicitly modeled feature types have the advantage of well-defined object semantics, attributes, and relations
  - basis for semantic interoperability between different actors
- However, often concrete models comprise additional **attributes or features not covered by the model**
- Incorporation of **generic CityObjects and attributes**
  - **every CityObject** can have an **arbitrary number of** additional **generic attributes** (string, int, real, date, URI)
  - **GenericCityObject** is subclass of CityObject
  - arbitrary GML3 geometry for each LOD
- shall only be used, if there is no appropriate concept provided by CityGML (problematic wrt. semantic interop.)

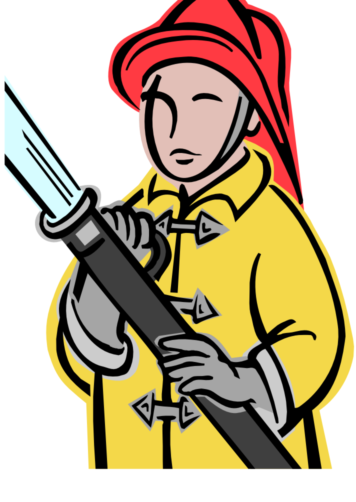
Example for Generic Attributes


```
<Building gml:id=“Building0815">
    <!-- other properties of feature type “Building” -->

    <stringAttribute name=“BuildingOwner">
        <value>Mr. Smith</value>
    </stringAttribute>

    <doubleAttribute name=“Value">
        <value>3500000</value>
    </stringAttribute>

    <!-- further properties of feature type “Building” -->
</Building>
```

- Available data types:
  integer, real (double), string, date, URI

3D Information Communities

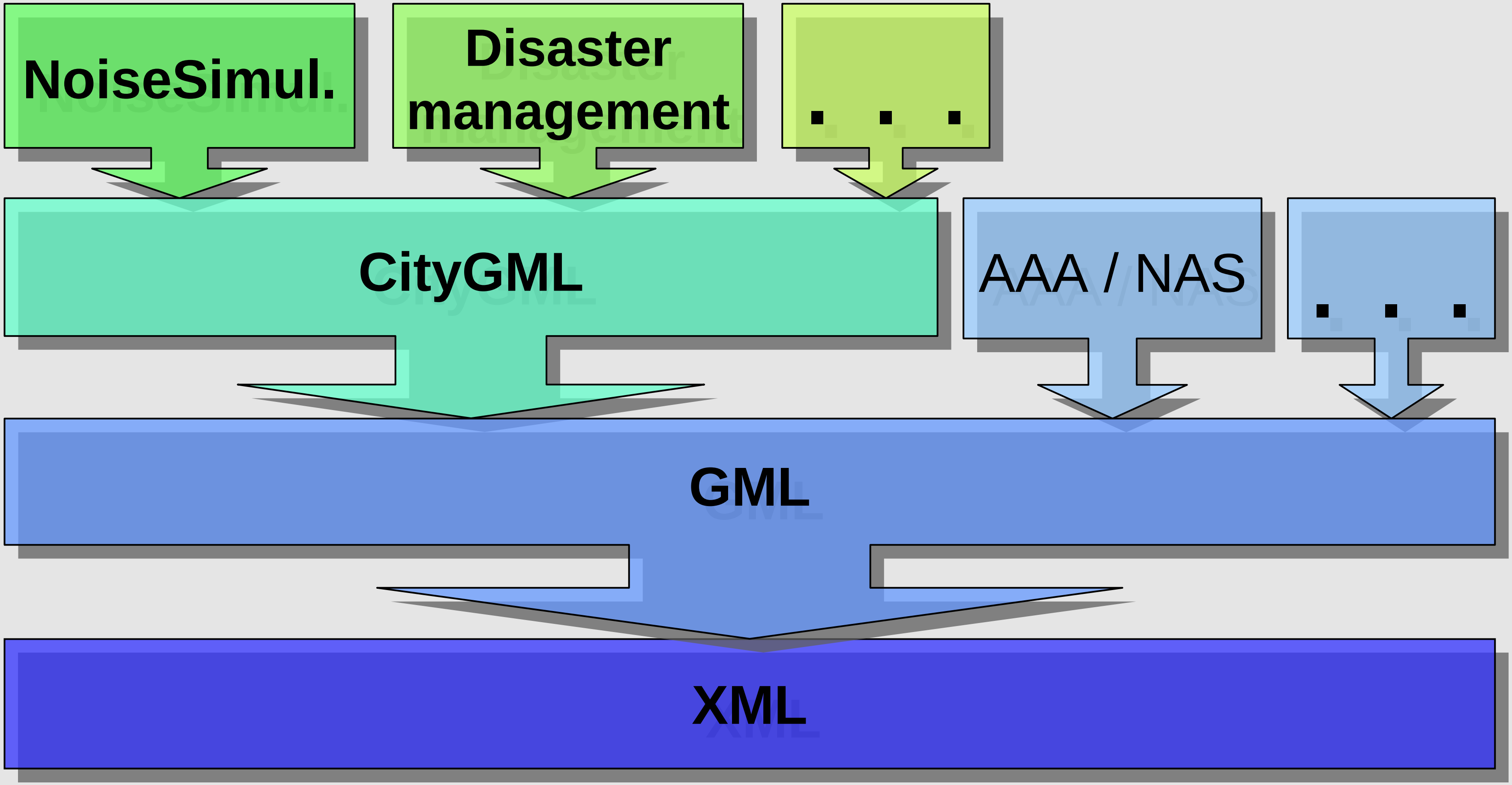
Extending CityGML
for specific application domains



Application Specific Extensions to CityGML

- CityGML should be considered a **base information model** for virtual 3D city models
- **But:** Specific applications need specific extra information
  - typically in close interaction with CityGML base information
- Examples
  - **Environmental simulations** like noise immission mapping need information about noise absorption of surfaces
  - **Cultural heritage** needs to augment objects by their heritage and history, and has to consider the development along time
  - **Utility networks** need to represent pipes, pipe tunnels, connectors, transforming devices

Application Domain Extensions (ADE)



Extension Considerations (I)

- **Information Communities** should be able to define extensions on their own
  - they must be able to associate new attributes to concrete CityGML feature types
  - formal definition of new properties / feature types in XML schema
  - similar situation to the specification of GML application schemas
- Different extensions should be usable **simultaneously**
  - e.g. CityGML Building features extended both by properties from real estate and noise pollution simulation
  - Requires **combinable application schemas**
- What about non-schema aware CityGML readers?

Extension Considerations (II)

Generally two types of domain specific extensions:

- **Extension of existing CityGML feature types** by
  - additional spatial and non-spatial attributes
  - additional relations / associations
- **Definition of new feature types**
  - preferably based on CityGML abstract base class *CityObject*
- Both are typically covered by the subclassing / inheritance mechanism of XML schema
  - *Create subclass of a CityGML feature type and add new properties to this class*

Typical Extension Approach

- create a **new feature type** by deriving the feature type from an (abstract) CityGML feature type like e.g. *_CityObject*, or
- **extend an instantiable feature type** by deriving a subtype from the concrete CityGML feature type and add new properties to this class
  - the extended CityGML class has to receive a new element name like *BuildingWithNoiseProperties*
  - **Problem: how to combine this with other extensions?**
  - **Problem: non-schema aware readers are not able to detect that a <BuildingWithNoiseProperties> is basically a <Building> element with some extra properties**

Application Domain Extensions (ADE)

Extension of the CityGML XML Schema declarations:

```
<xsd:complexType name="Building" …>
      ……
      <xsd:element ref="_GenericApplicationPropertyOfBuilding"
                   minOccurs="0" maxOccurs="unbounded"/>
      ……
</xsd:complexType>

<xs:element name="_GenericApplicationPropertyOfBuilding"
            abstract="true" type="xs:anyType"/>
```

… will allow to inject further XML structures into CityGML feature types at a later point in time (hooks for ADEs).

- one hook for each CityGML feature type

Application Domain Extensions (ADE)

Declaration of application domain specific attributes for existing CityGML features (e.g. Building, XML schema)**:**

```
<xsd:element
        name="NoiseReflection"
        type="xsd:string"
        substitutionGroup=
                "citygml:_GenericApplicationPropertyOfBuilding">
</xsd:element>

<xsd:element
        name="BuildingHabitants"
        type="xsd:positiveInteger"
        substitutionGroup=
                "citygml:_GenericApplicationPropertyOfBuilding">
</xsd:element>
```

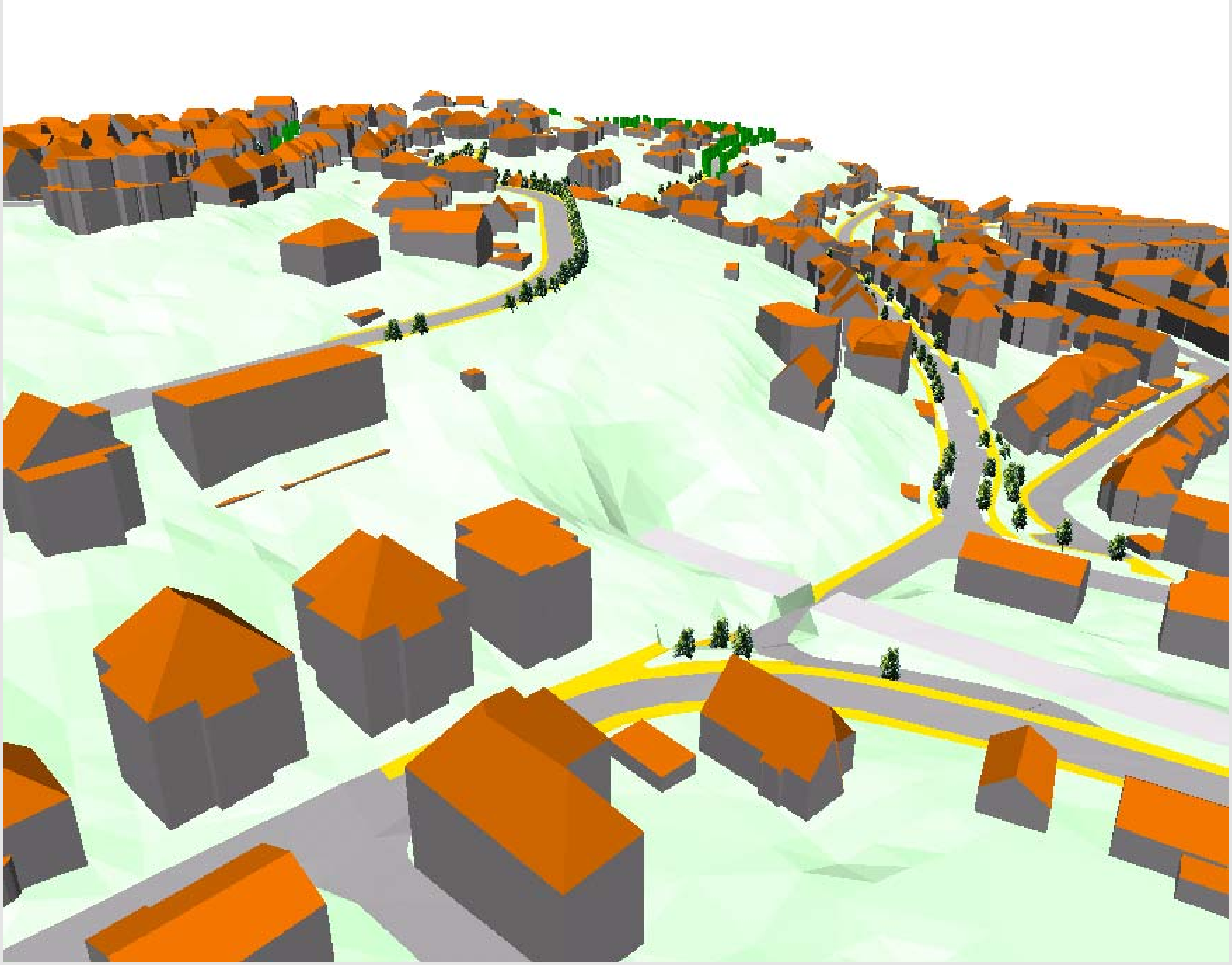
Application Domain Extensions (ADE)

Example for a CityGML **Building feature with** application specific **extra information** (qualified by a namespace):

```
<Building>
        <function>1000</function>
        ……………
        <noise:NoiseReflection>12</noise:BuildingReflection>
        <noise:BuildingHabitants>8</noise:BuildingHabitants>
        ......................
        <lod2Solid> …… </lod2Solid>
</Building>
```

Application Examples

The Official 3D City Model of Stuttgart



Screenshot of administration system (SupportGIS)

LOD2

- Objects have full thematic Information
- texture acquisition ongoing

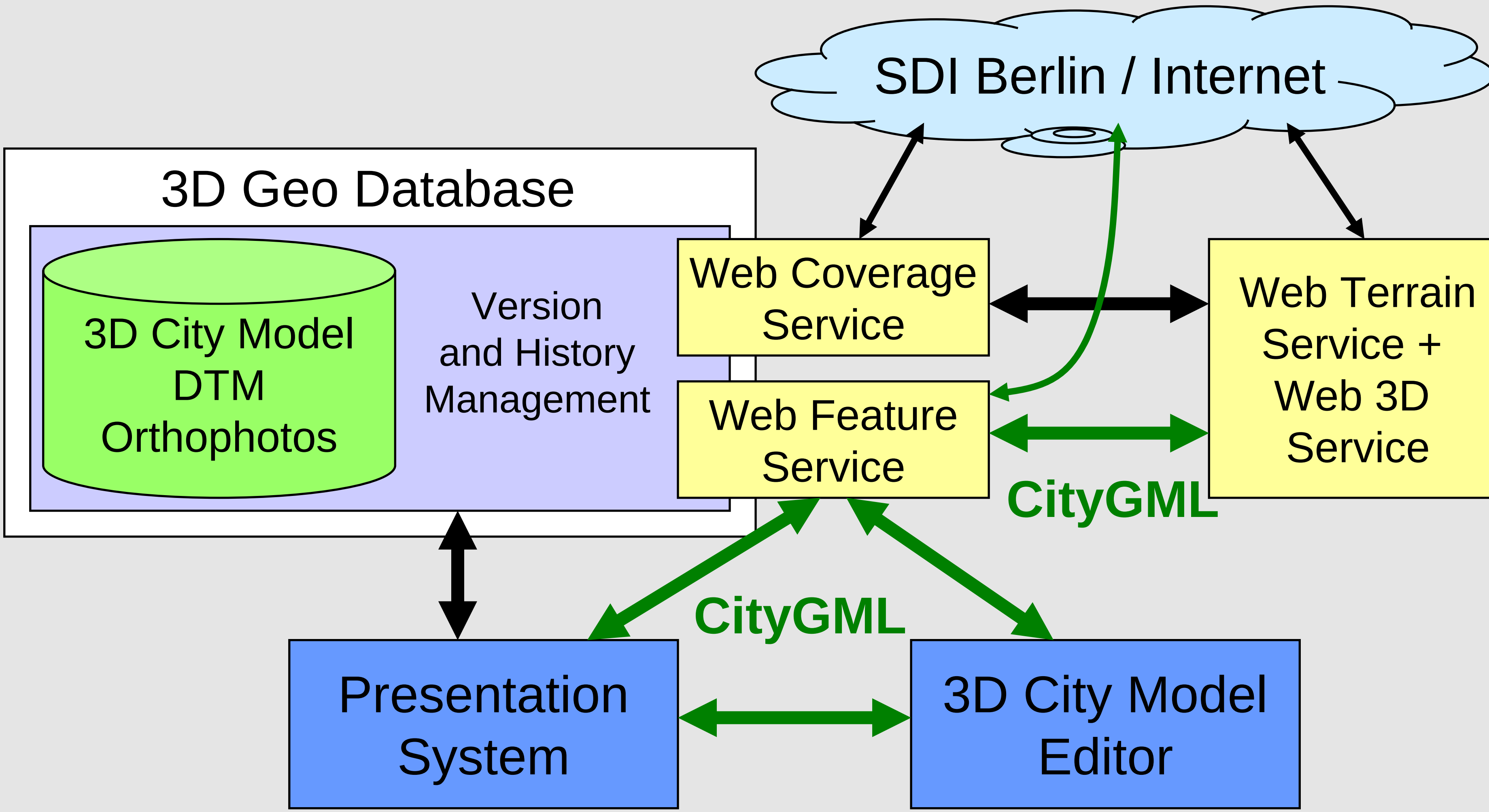
The Official 3D City Model of Berlin

www.3d-stadtmodell-berlin.de

3D visualization is the result of a portrayaling of Berlin's 3D city model

(modeled according to CityGML)

Berlin 3D: Realization with OGC Web Services



Application of CityGML for Noise Mapping

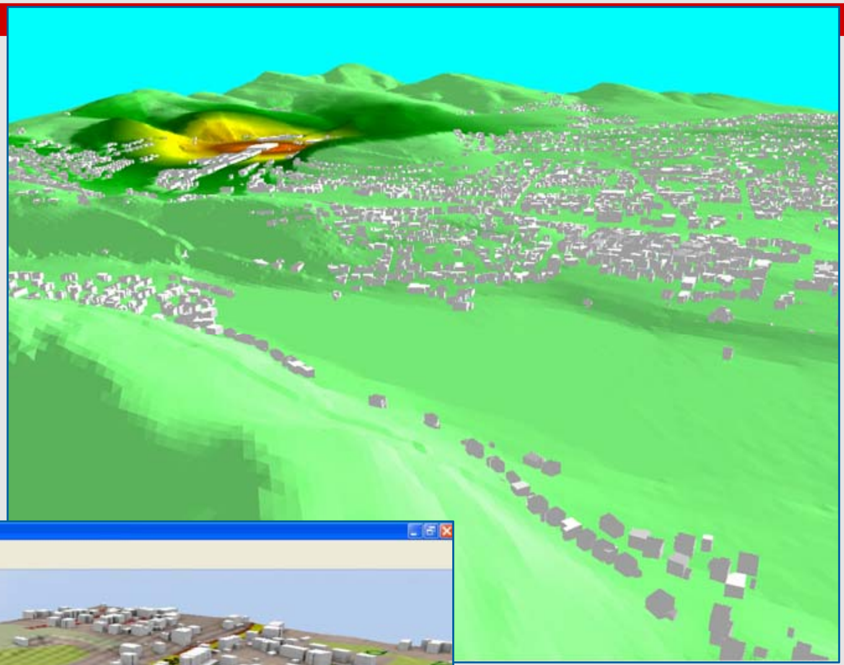
CityGML is applied in an ongoing project in Germany:

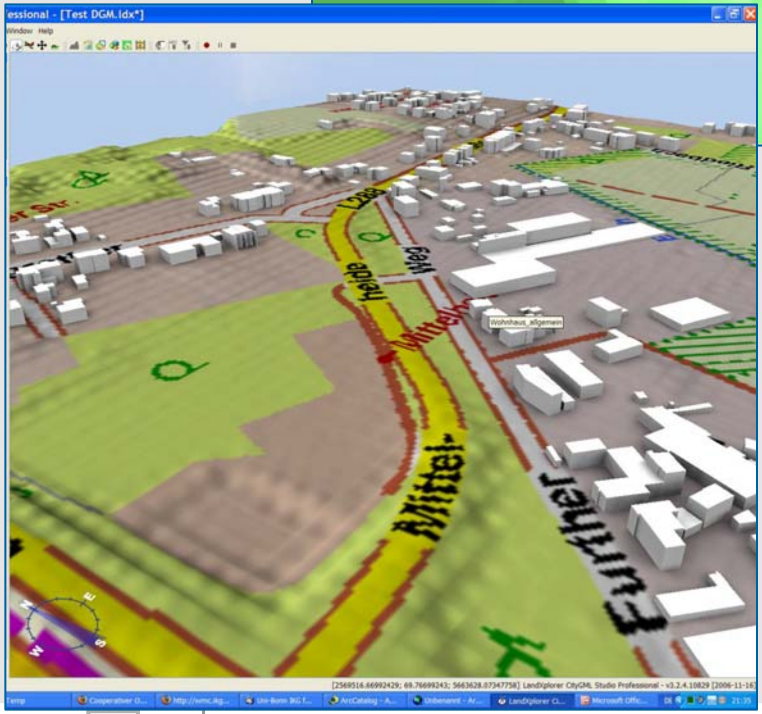
- Computation of **noise pollution maps** in the state North Rhine-Westphalia (18 million citizens)
- **Background: Environmental Noise Directive** from the European Commission
- Spatial Data Infrastructure uses following Web Services: WFS, WMS; Data formats: CityGML, GeoTIFF
- **Estimated savings** (wrt. proprietary systems): **>10 Mio €**
- Extension of CityGML by noise relevant attributes and features: **CityGML Noise ADE**

Illustration of Noise Pollution Mapping

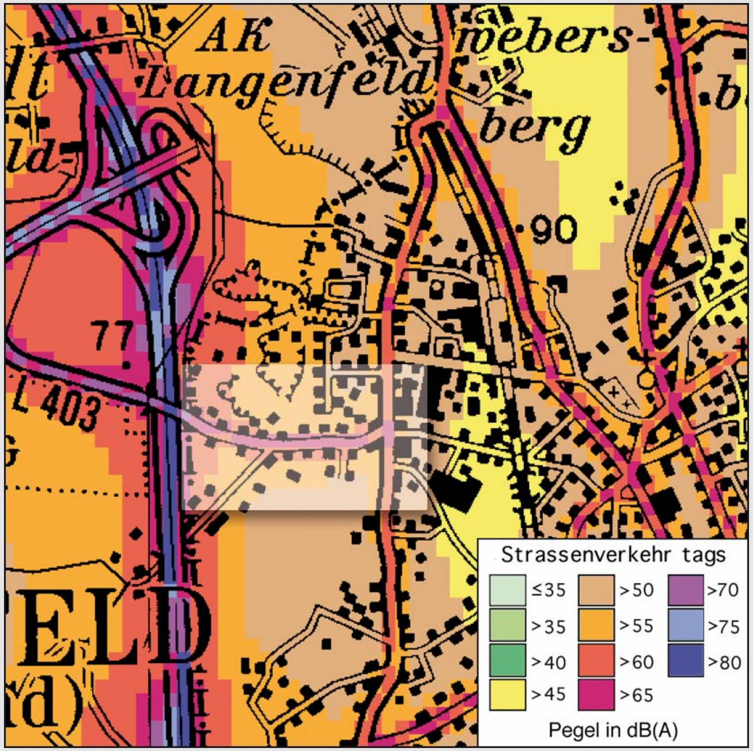
3D block model in CityGML from WFS-T

DTM 10m grid in GeoTiff from WCS





noise immission simulation



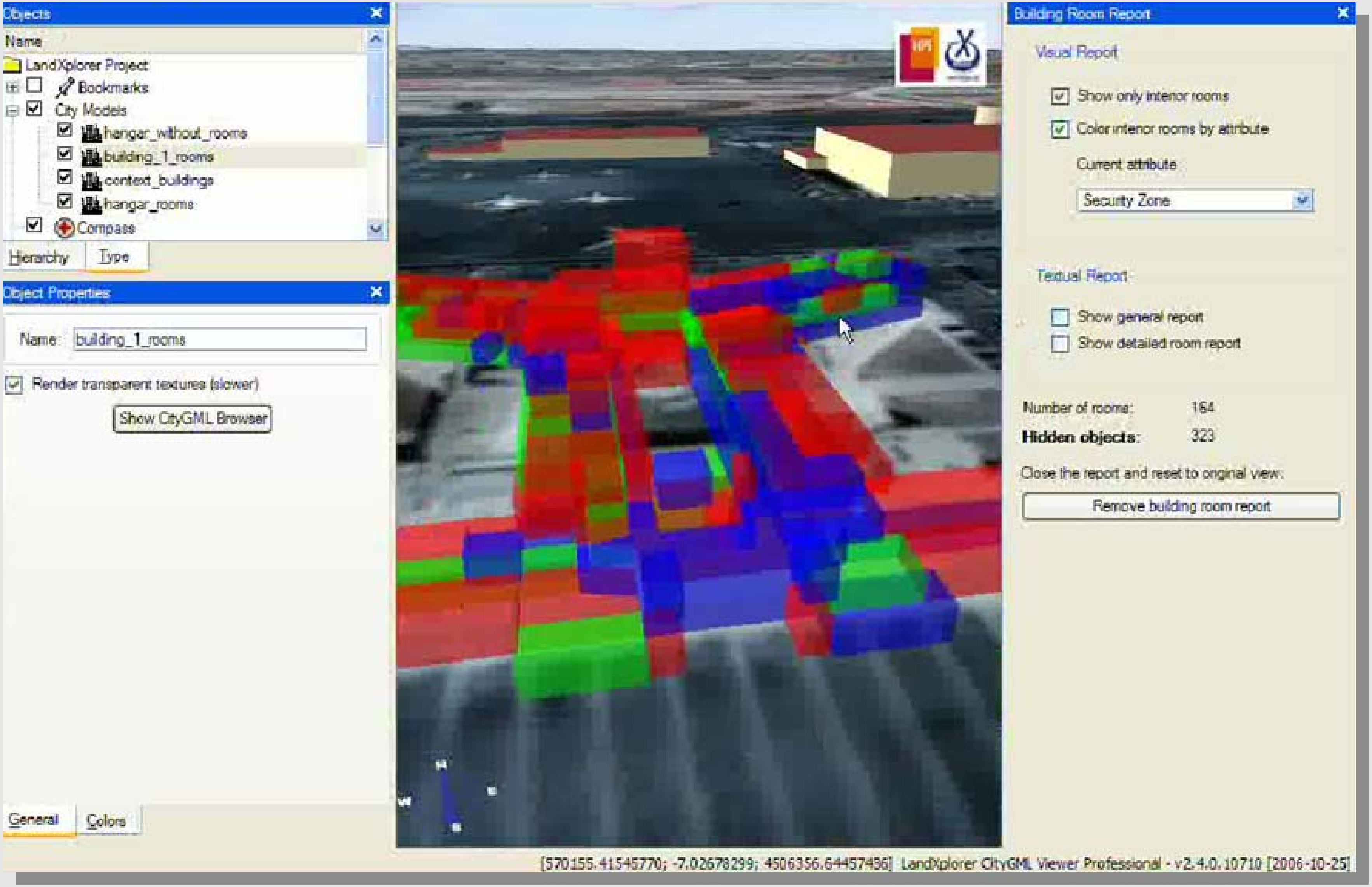
noise pollution maps for European Union reporting (using WMS)

Application Example 4: Homeland Security

Testbed OWS-4 of the Open Geospatial Consortium

- Runtime 6/2006-12/2006
- Fictive Scenario:
  Explosion of a „dirty bomb“ in New York harbour area
- Aim: **Supporting the planning staff** with the installation of a **field hospital**
  - Finding an appropriate location
  - Identification of a suitable building (size, room sizes, air conditioning)
  - Thematic queries & visual inspection
- Coupling of different OGC Web Services and client applications, data formats: **CityGML** and **IFC**

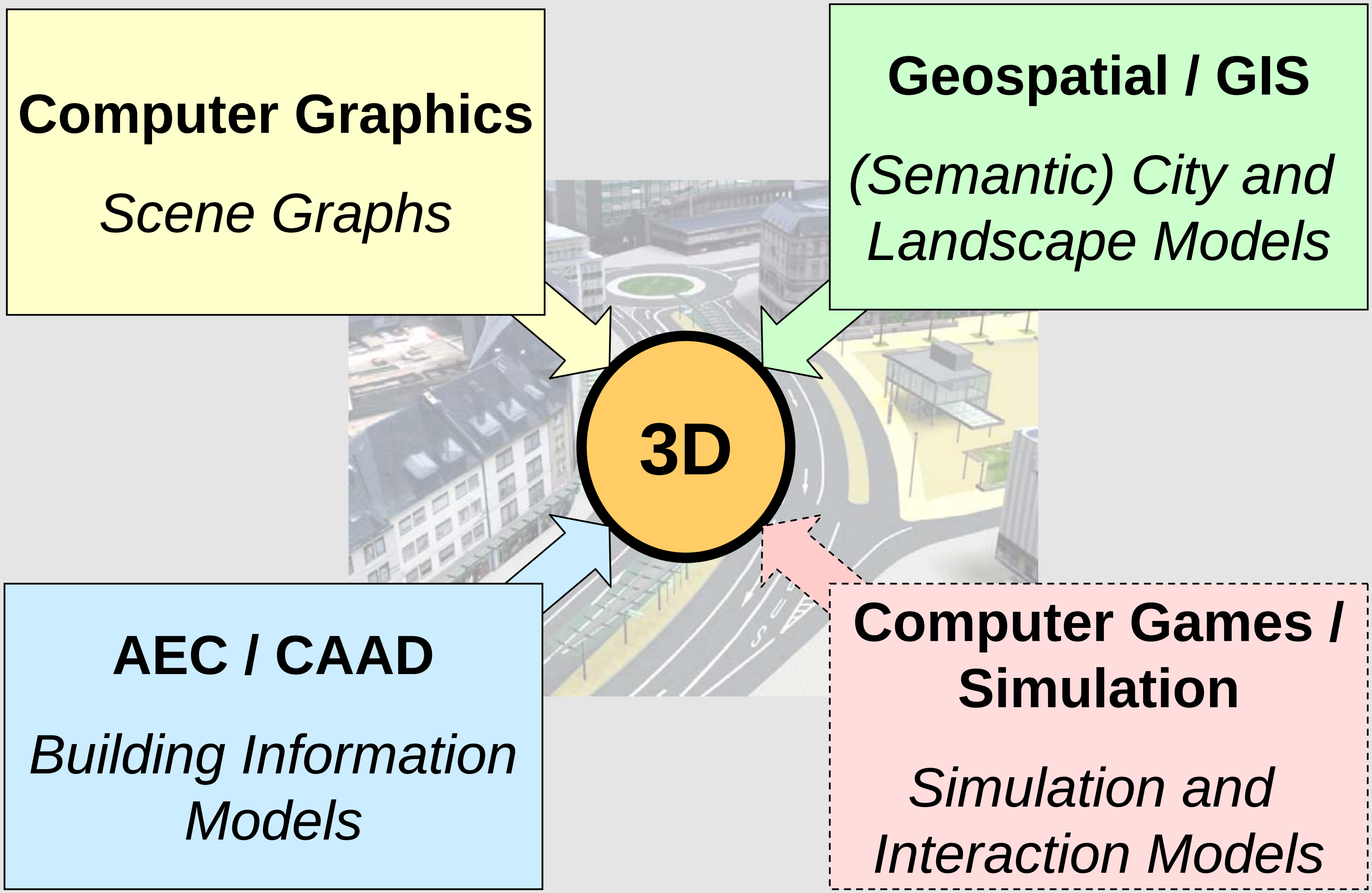
Application Example 4: Homeland Security



Relations to other Standards

Approaches to Virtual 3D City Modelling

Department of Geoinformation Science



Computer Graphics Approach

- **What is modelled?**
  - geometry (parametric primitives; boundary representation)
  - material / appearance
  - limited topology
  - typically no semantic information
  - interaction methods and object behaviour
- all elements are structured within **scene graphs**
  - aggregation using group nodes; transformation nodes
  - allows to define prototypes / reuse object definitions
- some exchange formats support georeferencing
  - GeoVRML, X3D, KML
  - but: models are restricted to cartesian coordinate system

AEC / CAAD Approach

- **What is modelled?**
  - geometry (parametric primitives; boundary representation; constructive solid geometry; sweep volumes)
  - topology
  - limited material / appearance
  - explicit semantics within building information models (BIM) (but not with legacy CAD formats)
- Most important BIM exchange format is IFC (**Industry Foundation Classes**)
  - IFC defines a **product data model for buildings / sites**
- elements of a BIM dataset are aggregated within a **project**
- only the format IFG (IFC for GIS) supports georeferencing
  - but: models are restricted to cartesian coordinate system

Geospatial / GIS Approach

- **What is modelled?**
  - geometry (3D in ISO 19107: only boundary representation)
  - topology
  - semantic information
  - limited appearance / material properties
- Models are based on the notion of **geographic features** (according to ISO 19109); exchange format is **GML**
- **Application schemas** define ontologies, i.e. taxonomies and partonomies of feature types (using OO concepts)
  - Ontology for 3D city models: **CityGML**
- always georeferenced; any 3D coordinate reference system (CRS) can be used (and mixed within the same dataset)
  - all geometries must belong to a CRS; up to now no nesting

Who standardizes (geo)virtual 3D worlds?

Open Geospatial Consortium (OGC)

- Exchange format GML; CityGML; KML; Web Services: WFS, WTS, W3DS

International Alliance for Interoperability (IAI)

- Product model for AEC/FM: Industry Foundation Classes (IFC)

Web 3D Consortium (W3D)

- Originator of VRML, GeoVRML, X3D

3D Industry Forum (3DIF)

- Graphics format "Universal 3D" (U3D) -> direct embedding in PDF

Khronos Group

- Exchange format COLLADA (used within Playstation, Google Earth)

International "De Jure" Standardisation: ISO

- ISO standards of the 191xx family (≈ OGC Standards), X3D, IFC

Virtual Reality Exchange Formats

Department of Geoinformation Science

	X3D	U3D	KML	COLLADA	IFC	CityGML
geometry	+	+	0	+	++	+
georeferencing	+		+		(IFG) +	++
appearance	+	+	0	++	0	+
topology	0	0		0	+	+
semantics	0			0/+	++	++
linking / embedding	+		++	++		++

Legend: 0 = basic, + = sophisticated, ++ = comprehensive; empty = not supported

What about other data formats? (I)

(Georeferenced) 3D Graphics Standards

- **(Geo)VRML, X3D, U3D, KML**
- focus on geometry & appearance
  - in general, no adequate concepts for semantic feature models
- X3D is extensible, but no common rules for modeling of geographic features, relations, geometry, topology

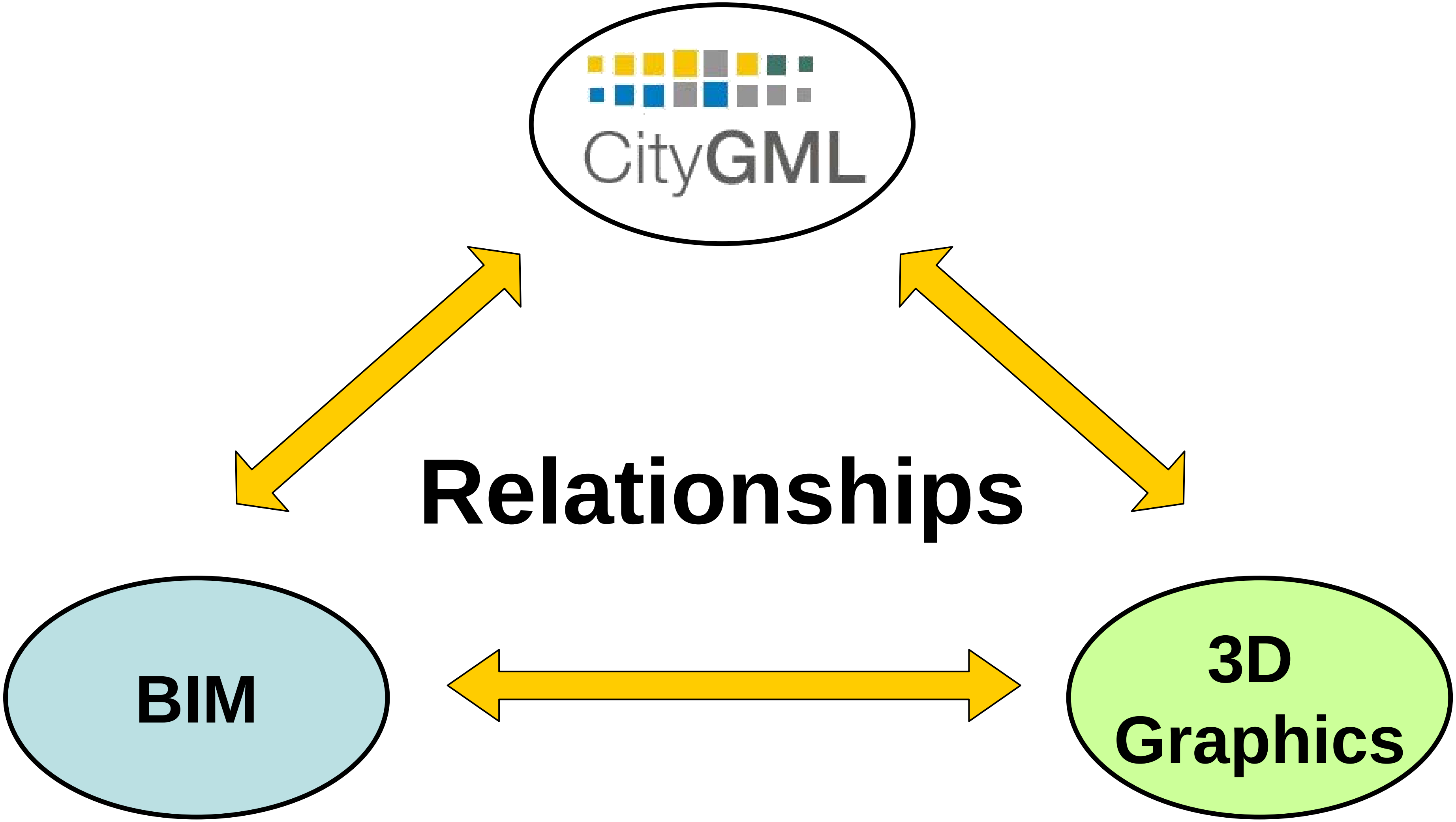
Generic (Proprietary) Exchange Formats

- **3D Shapefile, DXF,** etc.
- limited expressivity wrt. to complex models
- no common information model for 3D city models
  - do not address semantic interoperability

What about other data formats? (II)

Semantic Information Models

- **Industry Foundation Classes** (IFC)
  - good: **objects with well-defined semantics** (product model)
  - however, mostly focused on buildings; few natural features
  - **very complex geometry model** (CSG & B-Rep); no native support by / mapping to spatial datatypes of DBMS
  - developed independently from ISO 191xx and OGC standards
- **LandXML**
  - good: cadastre model / DLM with **well-defined semantics**
  - no buildings; no geometric 3D primitives; appearances?
- Generally **missing features: multiscale modeling, complex DTMs; natural objects**


CityGML
Relationships
BIM
3D
Graphics

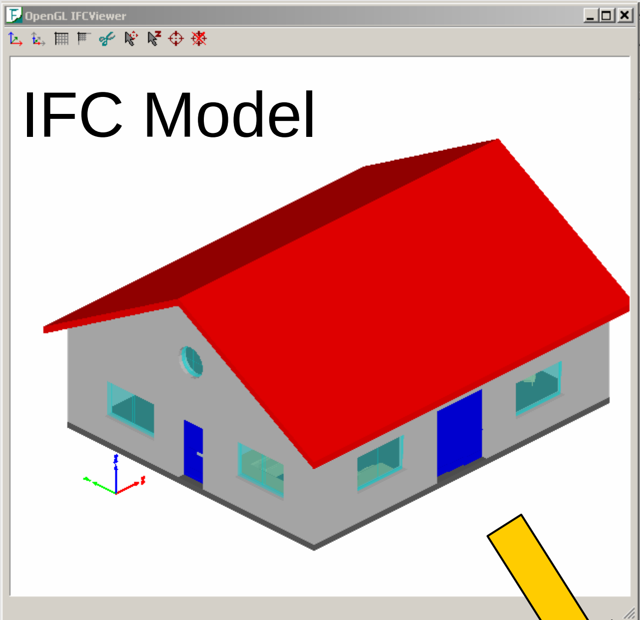
CityGML from the BIM perspective

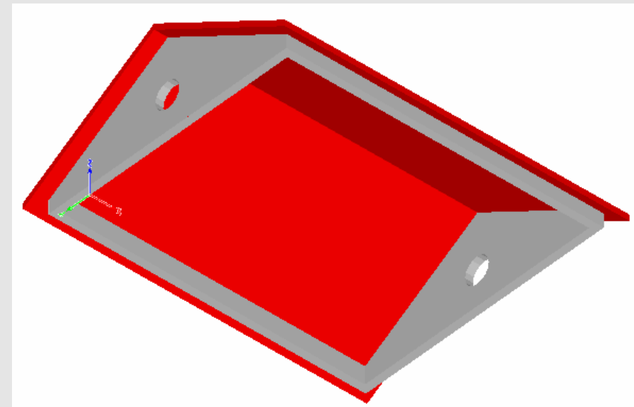
- Provision of **information about the surroundings** / environment of buildings and sites
  - **Embedding of 3D models** into the real world's coordinate frame
  - Analysis and **identification of suitable locations** for construction
  - Querying 3D urban objects with **geospatial selection criteria**
  - Useful for planners, architects, and engineers
- Can be a source format for the creation of Building Information Models from observed data
  - for example CityGML -> IFC
  - CityGML objects already carry semantic information which are helpful in interpretation processes
  - CityGML especially suited for the stepwise reconstruction and refinement of urban objects (coping with different model qualities)

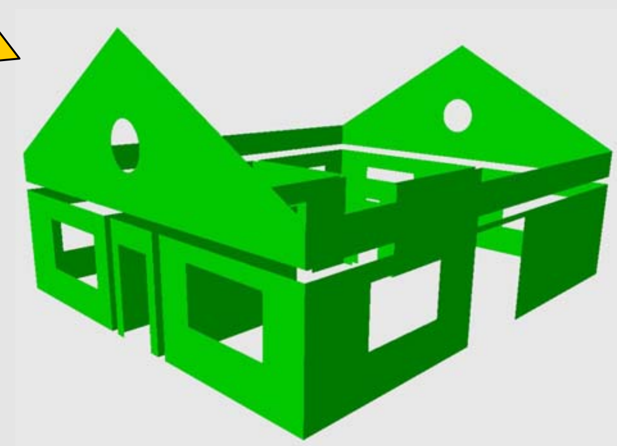
BIM from the CityGML perspective

- Behind **IFC** there is also a **semantically rich information model**
  - In fact, it is more detailed than CityGML
  - However, **lack of other city features**; **limited georeferencing**
- **Source for highly detailed building model data**
  - with respect to geometry and semantics
  - can be used to provide LOD3 and LOD4 models
- CityGML building model adopted some of the conceptual modelings of IFC
  - IFC spaces -> CityGML rooms
  - IFC Property Sets -> CityGML generic attributes, now also ADEs

Deriving LOD4 models from IFC







Current research of Benner, Geiger, Leinemann
Helmholtz Research Center Karlsruhe

Deriving LOD4 models from IFC

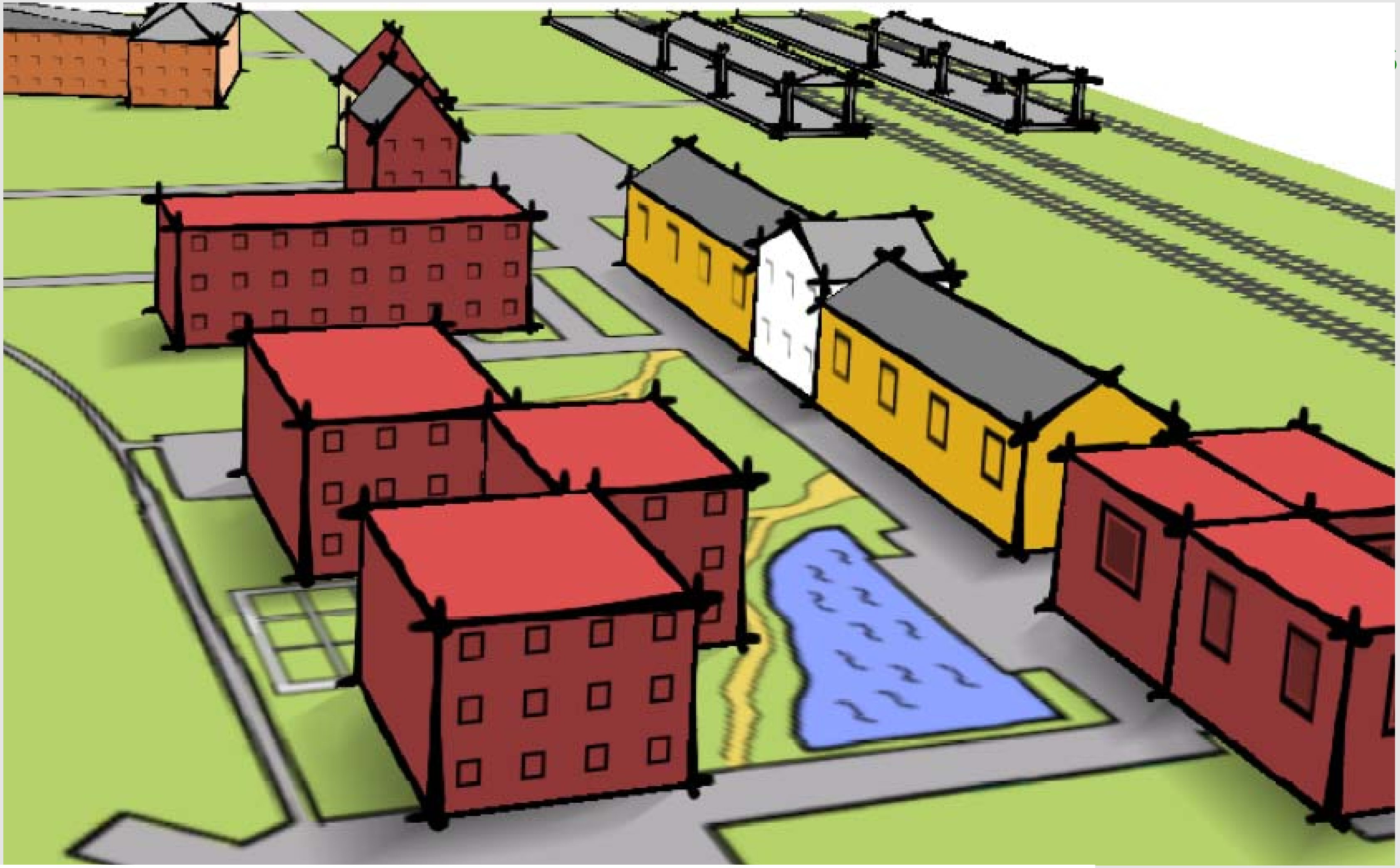
CityGML from the 3D visualization perspective

- **Provision of large amounts of 3D geospatial data**
  - rich attributes and geometric and semantic decompositions
- Not optimized wrt. transfer size and efficient visualization
  - absolute world coordinates (need for projection or transformation)
  - no grouping according to scene graph concepts
  - however: easy to map to 3D graphics as only the Boundary Representation is being used
- No support of more sophisticated appearance properties, shaders, graphical materials, and light sources
  - but: can be derived in many cases from the semantic information of the CityGML features
  - option: definition of a CityGML „High Definition Graphics“ ADE

3D visualization from the CityGML perspective

- **3D visualization** is the **result of a portraying process** applied to a CityGML model
  - **CityGML** is a source structure for visualization processes; **not intended to be used as a 3D graphics format**
- Portraying
  - **simplest form: 1:1 conversion** of geometry and appearance data to a 3D graphics format (incl. coordinate transformations)
  - **more sophisticated: 3D cartographic design**, for example:
  - Text and label placement
  - Symbolization and non-photo realistic rendering
  - Generalization
- Appropriate OGC Web Services for 3D portrayaling: **Web 3D Service** and **Web Terrain Service**

3D visualization from the CityGML perspective



Non-photo realistic rendering. © J. Döllner & M. Walter, 2003

Coming to the end…

Summary

CityGML is a

- **Geospatial Information Model** (based on ISO 191xx)
- and **Exchange Format** for virtual 3D city and regional models (realised as GML3 Application Schema)

CityGML represents **Geometry, Topology, Semantics, and Appearance**

- esp. semantic / structural information is needed for a range of applications

Should be considered as a **rich 3D information** source for the **generation of** (also cartographic) **3D visualizations**

- WFS [CityGML] -> W3DS [X3D and KML / COLLADA]

What is CityGML‘s Use for Research?

- **Base model** / base ontology for
  - geodatabase developments
  - project specific extensions (like relations or new feature types)
- Could be **target model** of 3D extraction methods
  - concerning feature types, aggregation structures
  - 5 discrete scales usable for 3D generalisation
- **Exchange format**
  - lossless information exchange between subsystems / GeoDB
  - increasing number of available implementations / 3D-GeoDB
- Good amounts of **real testdata available**
  - Berlin, Bonn, Bochum, Hamburg, Stuttgart, Recklinghausen, …
  - also 3 freely accessible Web Feature Services delivering CItyGML

Studying 3D City Models (and much more!)

Department of Geoinformation Science

- **International Master's Program** at Techn. University Berlin
  Geodesy and Geoinformation Science
- Duration: **4 terms** (2 yrs.); teaching language: **English**
- Degree: **Master of Science** (MSc.)
- Candidates' prerequisites: qualifying university degree
  - Bachelor or Master of Science (or Diploma) from following fields:
  - Geodesy, Geomatics, Cartography, (Geo-)Informatics, Construction Engineering, Earth Sciences, or related
- **www.igg.tu-berlin.de/master**