Geography Markup Language (GML)
An introduction

Exam appointment

aktuelle(s) LV/Fach | gleiche(s) LV/Fach

Nr.	Sem.	Art	Titel	Prüfer*in	Datum	Ort	Universität	Anmelde-fenster
856.161	23W	VO	Design of Geospatial Data Models	Belina G (P)	**22.03.2024** 15:00 - 17:00	Bauteil 14, GI-Lecture (SC30OG1.107)	PLUSonline	24.01.2024 00:00 20.03.2024 12:00
856.161	23W	VO	Design of Geospatial Data Models	Belina G (P)	**01.03.2024** 14:00 - 16:00	Bauteil 14, GI-Lecture (SC30OG1.107)	PLUSonline	22.01.2024 00:00 28.02.2024 12:00
856.161	23W	VO	Design of Geospatial Data Models	Belina G (P)	**02.02.2024** 14:00 - 16:00	HS T02 (JAK2OG2.01)	PLUSonline	22.01.2024 00:00 31.01.2024 12:00

- Be on time – we start right at time (i.e. punctual)
- Bring paper and pencil
- Closed book

What is GML? – Scope

- The Geography Markup Language is
  - a markup language for geographic information based on XML
  - designed for the web and web-based services
  - To transport and store spatial data in an non proprietary fashion
  - Originally defined by OGC to facilitate data exchange between GI services and applications

What is GML? – The status

- GML is an OpenGIS® Implementation Specification
  - The current version is 3.2.1 (with corrigendum 3.2.2)
  - Version 3.3 supplements version 3.2.1 by extended schema and encoding rules
  - See: https://www.ogc.org/standards/gml
- GML is also a work item of ISO/TC 211 and is published as ISO 19136 (also: GML 3.2.1)
- The work is carried out by a Joint Working Team of OGC and ISO/TC 211

What is GML? – Characteristics

GML

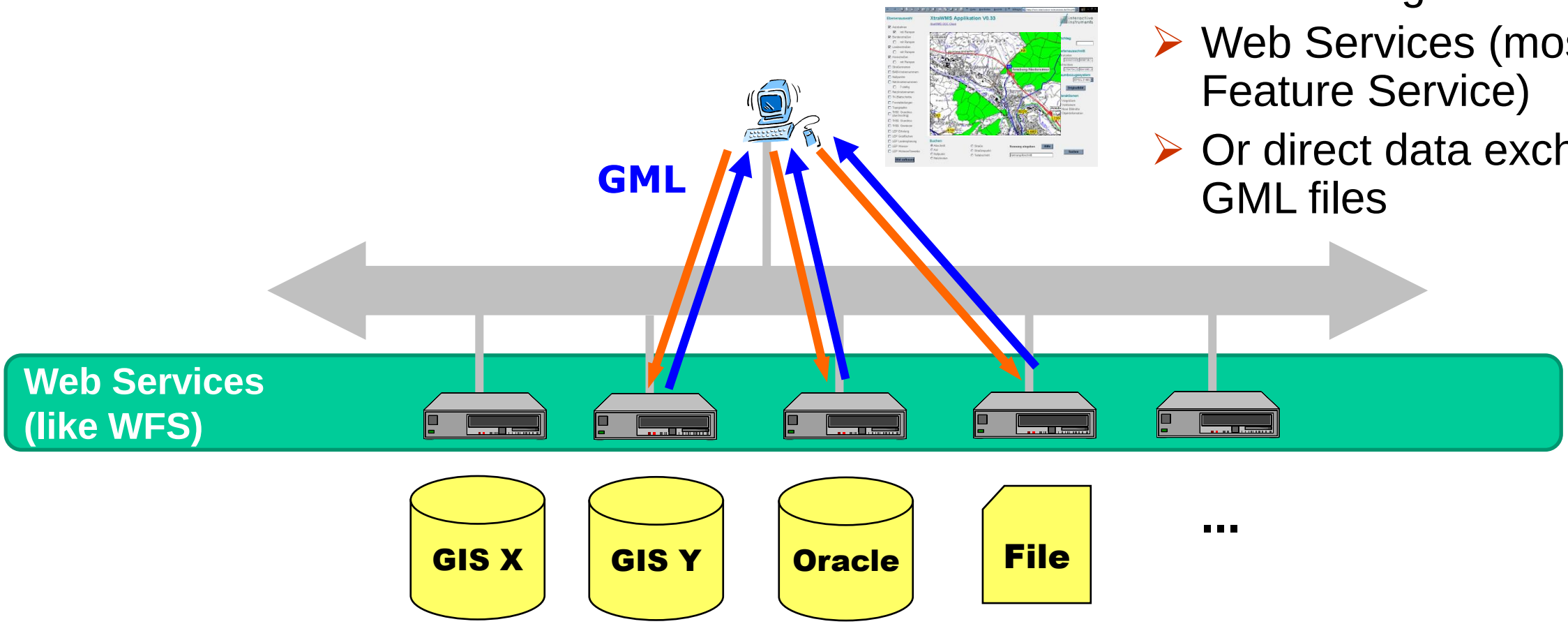
- is based on XML technologies (W3C)
  - XML, XML Namespaces, XML Schema, Xlinks
  - Extensible
- Describe **features** in XML (see also the UML class before Christmas)
  - Describes and supports spatial and non-spatial properties of objects
  - is open and vendor-neutral
- implements concepts of the ISO 19100
- supports the definition of profiles (proper subsets) of the full GML capabilities
  - Can be adopted according to specific needs

What is GML? – Characteristics

GML

- supports the description of geospatial ***application schemas*** for ***information communities***
- enables the creation and maintenance of ***linked*** geographic application schemas and datasets
- supports the ***transport and storage*** of application schemas and data sets
- increases the ability of organizations to ***share*** geographic application schemas and the information they describe
- leaves it to implementers to decide whether application schemas and datasets are stored in native GML or whether GML is used only for schema and data transport

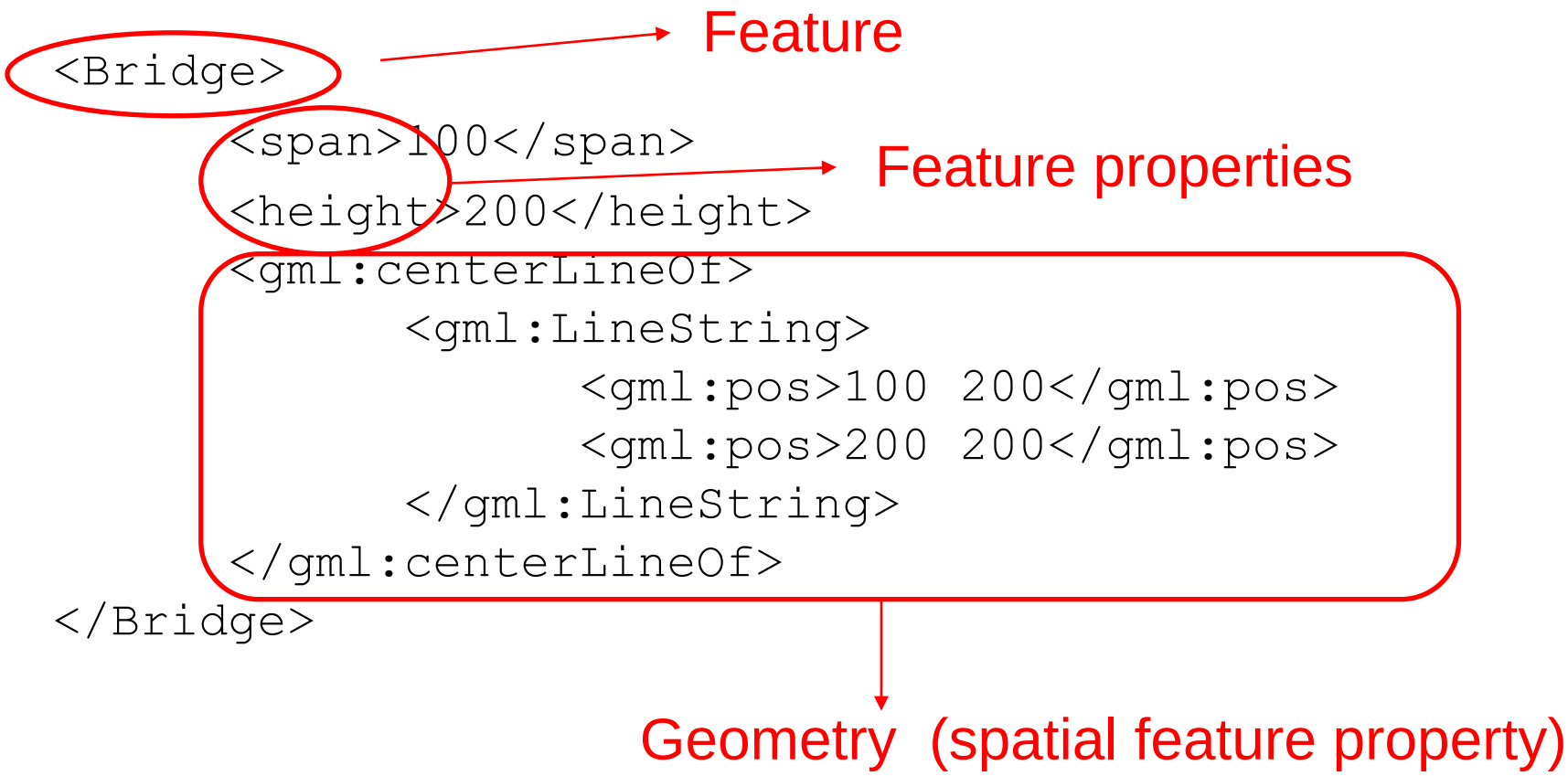
GML vision: a vendor-neutral exchange of spatial data



Data exchange via:

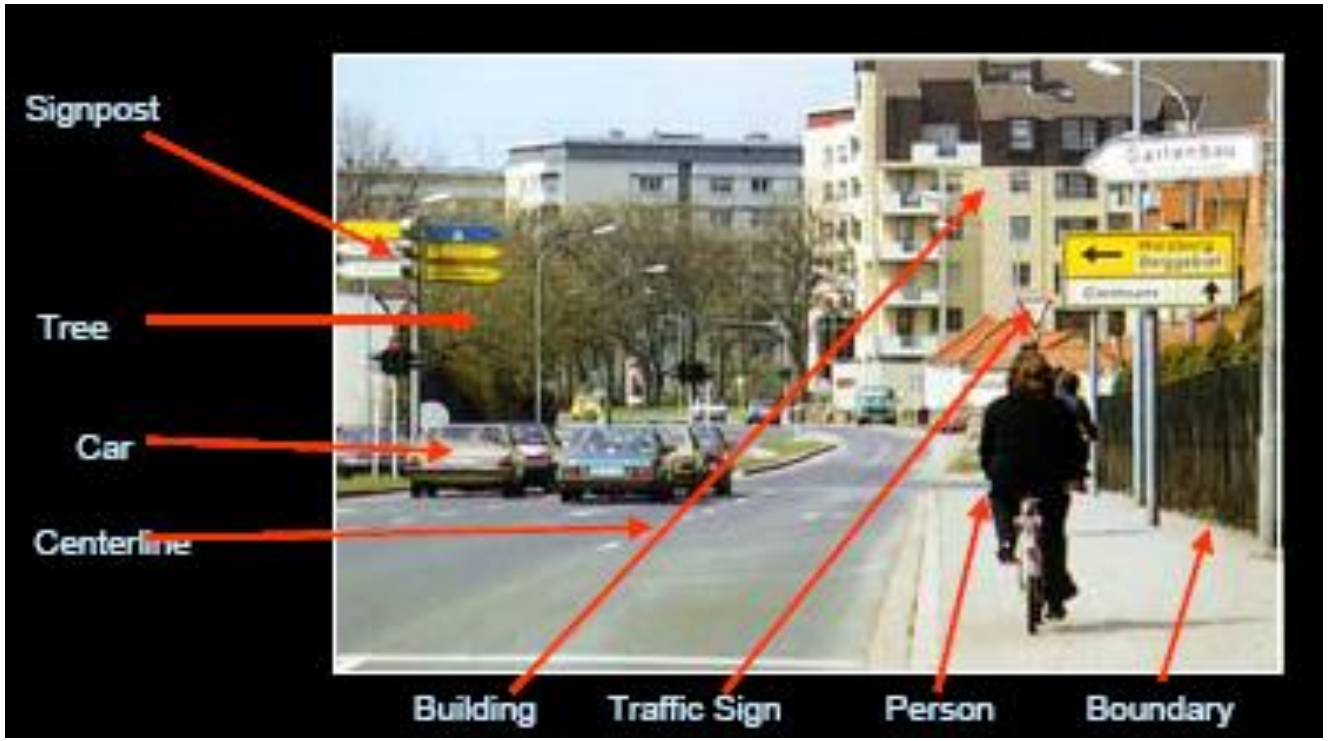
- Web Services (mostly Web Feature Service)
- Or direct data exchange of GML files

GML - A simple example

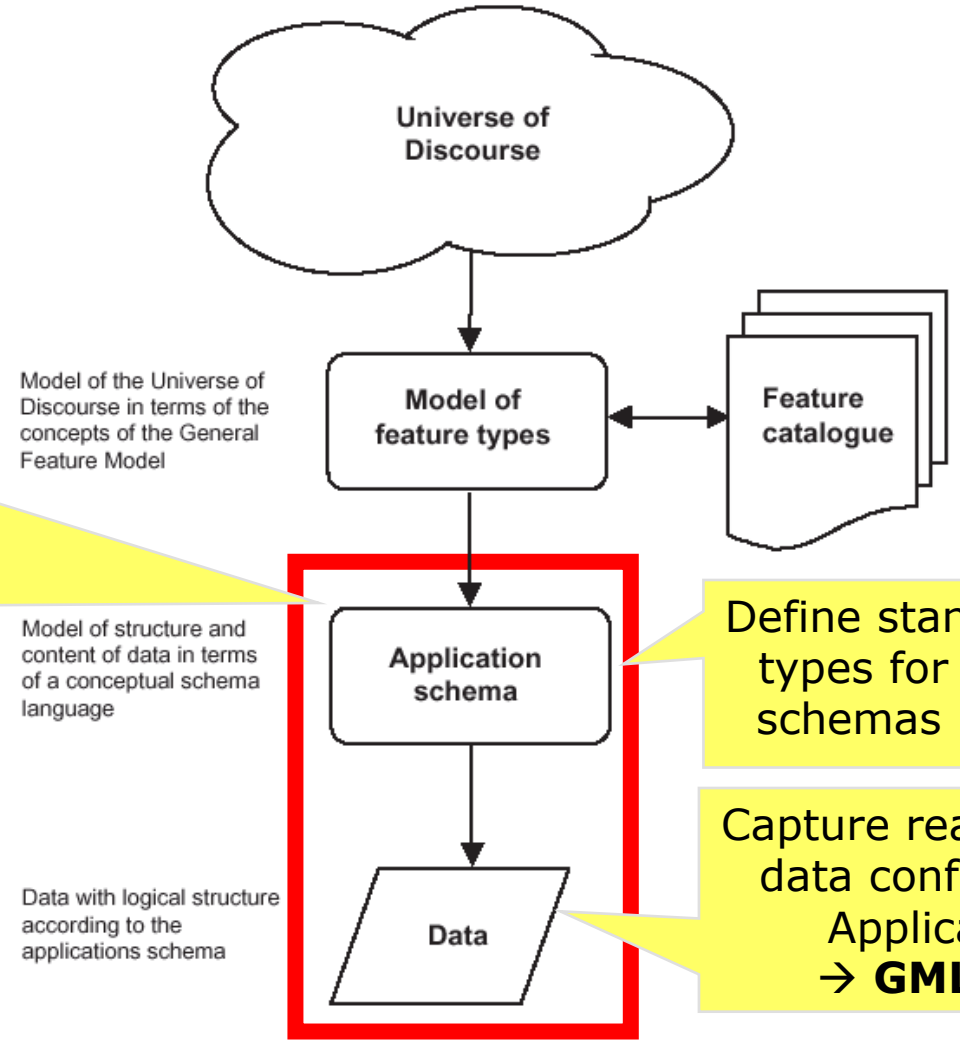


Modelling Feature Types

- The core concept of GML is the feature (see also the class materials related to the UML Application schema).
- A feature is the abstraction of the phenomenon in the real world.
- Every feature has a feature type. A feature type in GML is a named classification of a fact of the real world.



GML Base Schemas, Application Schemas and Documents



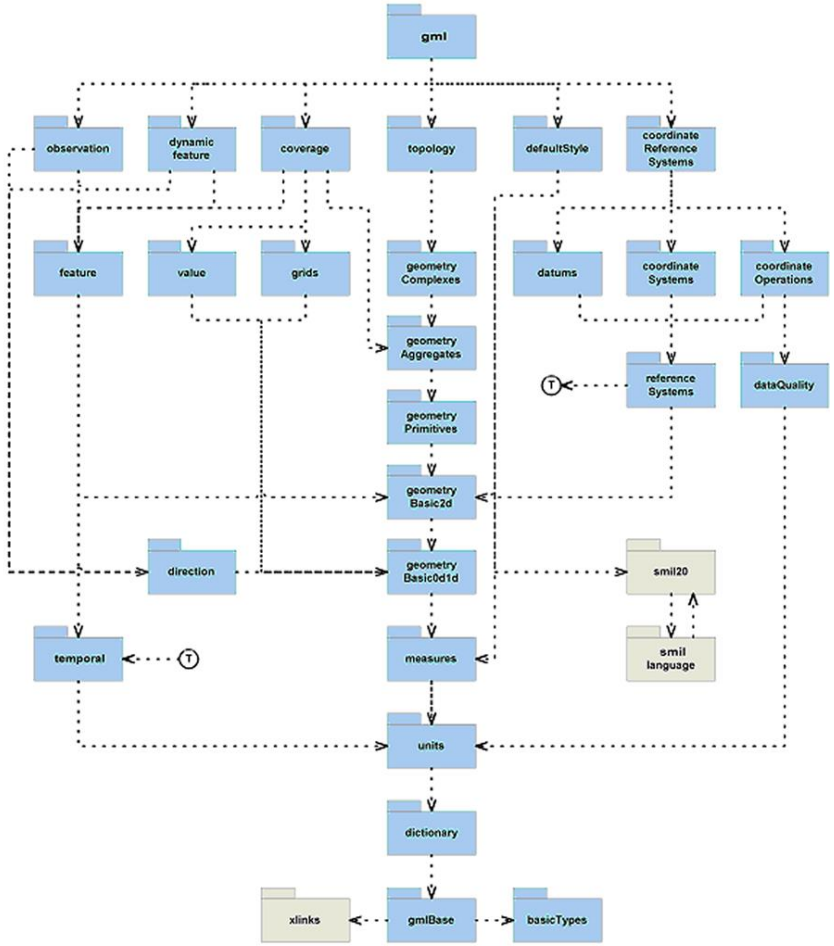
Use a schema language to model geographic information in a **UML Application Schema** and define rules for such schemas

Define standard elements and types for use in application schemas → **GML schemas**

Capture real-world objects as data conforming to a GML Application Schema → **GML Documents**

GML Base Schemas

- ***GML Base Schemas*** are **horizontal and not focused on a specific application domain**
- But they can **provide common constructs** and concepts which may be used by all the different application domains
- It tailors XML to the GML vocabulary



You find the gml base schema under:
https://schemas.opengis.net/gml/

GML Base Schemas

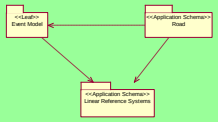
- Base schemas, general syntax, feature model, metadata mechanisms
- Basic geometry (0d, 1d, 2d)
- Additional geometric primitives (0d, 1d, 2d, 3d)
- Geometric composites
- Geometric aggregates
- Coordinate reference systems
- Topology
- Temporal information and dynamic features
- Definitions and dictionaries
- Units, measures and values
- Directions
- Observations
- Coverages
- Default styling

You find the gml base schema under: https://schemas.opengis.net/gml/

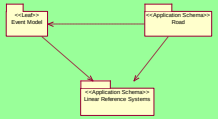
GML Technology Stack

GML Application Schemas

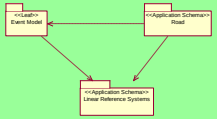
Cadastre, Land Use



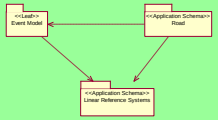
Traffic And Transport



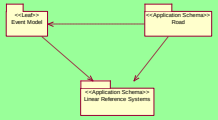
Telecom



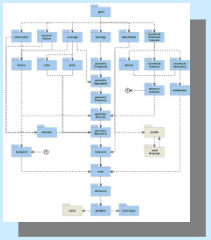
Environ-ment



...



GML Base Schemas



XML Technologies / W3C

Complexity of GML

- GML can be very complex – support the full spectrum of the GFM
- Complexity can cause a problem for the data exchange
  1. Dynamic feature support
  2. Inclusion of time dimension
  3. Complex geometries
- The complex aspects are not always supported by COTS solutions (COTS = commercial off the shelf)

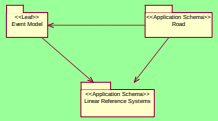
GML - Profiles

- GML Profiles help to limit complexity of GML
- Profiles do select only those aspects of GML which are required (based on the GML standard schemas)
- Creation of GML application schema is based on the profile(s)
  OGC defines profiles for GML like:
  - The „Simple Feature Profile“ (3 levels: SFP0, SFP1, SFP2)
  - **Be aware that most GIS COTS do support only the Simple Feature Profile Level 0!!!**
- You can define your own GML profile (e.g. for your organization)
- The profile is defined as part you your GML application schema definition

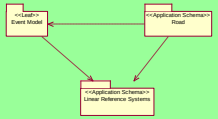
GML Technology Stack with GML Profiles

GML Application Schemas

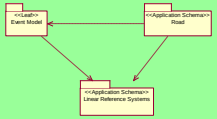
Cadastre, Land Use



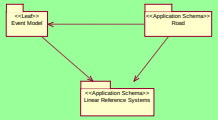
Traffic And Transport



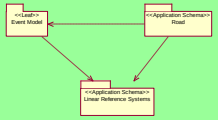
Telecom



Environ-ment

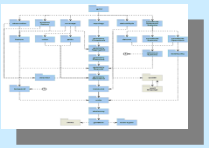


...



GML Profiles

GML Base Schemas



XML Technologies / W3C

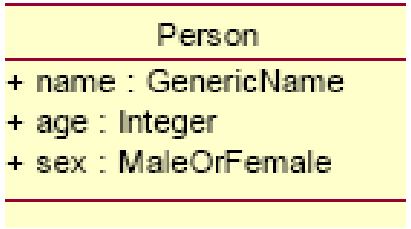
GML Key Concepts

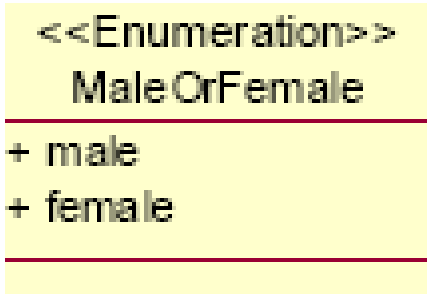
- The state of a ***feature*** is described by a set of properties, in which every property is in principle represented by a triple

  {name, type, value}.
- Spatial properties are those properties that have a geometric object as their value (e.g. a point, a linestring, …).
- Properties may be local values or references to remote objects (stored in other GML documents on other web servers)
- The GML Schema specifies a number of pre-defined types (for example a number of geometry types).

GML Key Concepts

- Features with a similar characteristic are grouped to feature types, those features will share a similar set of properties. This structure is specified in a GML Application Schema.






```
<Person gml:id = “p1”>
    <gml:name>Bob</gml:name>
    <age>10</age>
    <sex>male</sex>
</Person>
```

GML Key Concepts

Three ways to represent a relationship between two features:

```
<Person gml:id="p1">
   <owns xlink:href="#c1"/>
</Person>
<Car gml:id="c1">
   <!-- ... -->
</Car>
```

```
<Person gml:id="p1">
   <owns>
      <Car gml:id="c1">
         <!-- ... -->
      </Car>
   </owns>
</Person>
```

```
<Person gml:id="b1">
   <owns xlink:href="http://www.someserver.com/cars.xml#c1"/>
</Person>
```

- The feature is either a child element of the property or referenced by an xlink:href attribute in the property element
- The xlink:href attribute is interpreted in the way that the value of the property is the feature referenced in the link
- The referenced feature can be part of the same GML document or anywhere in the internet/intranet

GML Example

```
▼<gml:featureMember xmlns:gml="http://www.opengis.net/gml" xmlns:tiger="http://www.census.gov">
  ▼<tiger:poi gml:id="poi.1">
    ▼<tiger:the_geom>
      ▼<gml:Point srsName="urn:ogc:def:crs:EPSG::4326" srsDimension="2">
          <gml:pos>40.689167 -74.044444</gml:pos>
        </gml:Point>
      </tiger:the_geom>
      <tiger:NAME>Statue of Liberty</tiger:NAME>
    </tiger:poi>
```

- The feature „*poi*“ as defined by the Tiger Census data is described by the following properties:
  - *the_geom*
    - Described as a GML Point based on WGS84 coordinates
  - *NAME*
    - Name of the poi feature

GML Application schema - example

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
           xmlns:gml="http://www.opengis.net/gml/3.2"
           xmlns:app="http://example.com/application"
           targetNamespace="http://example.com/application"
           elementFormDefault="qualified">

  <!-- Import GML and specify its namespace -->
  <xs:import namespace="http://www.opengis.net/gml/3.2" schemaLocation="http://schemas.opengis.net/gml/3.2.1/gml.xsd"/>

  <!-- Define the Building type -->
  <xs:element name="Building" type="app:BuildingType"/>

  <xs:complexType name="BuildingType">
    <xs:complexContent>
      <xs:extension base="gml:AbstractFeatureType">
        <xs:sequence>
          <xs:element name="geometry" type="gml:PolygonPropertyType" minOccurs="0"/>
          <xs:element name="height" type="xs:double"/>
          <!-- Add other properties specific to buildings -->
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

</xs:schema>
```

Import GML schema

Feature Type definition

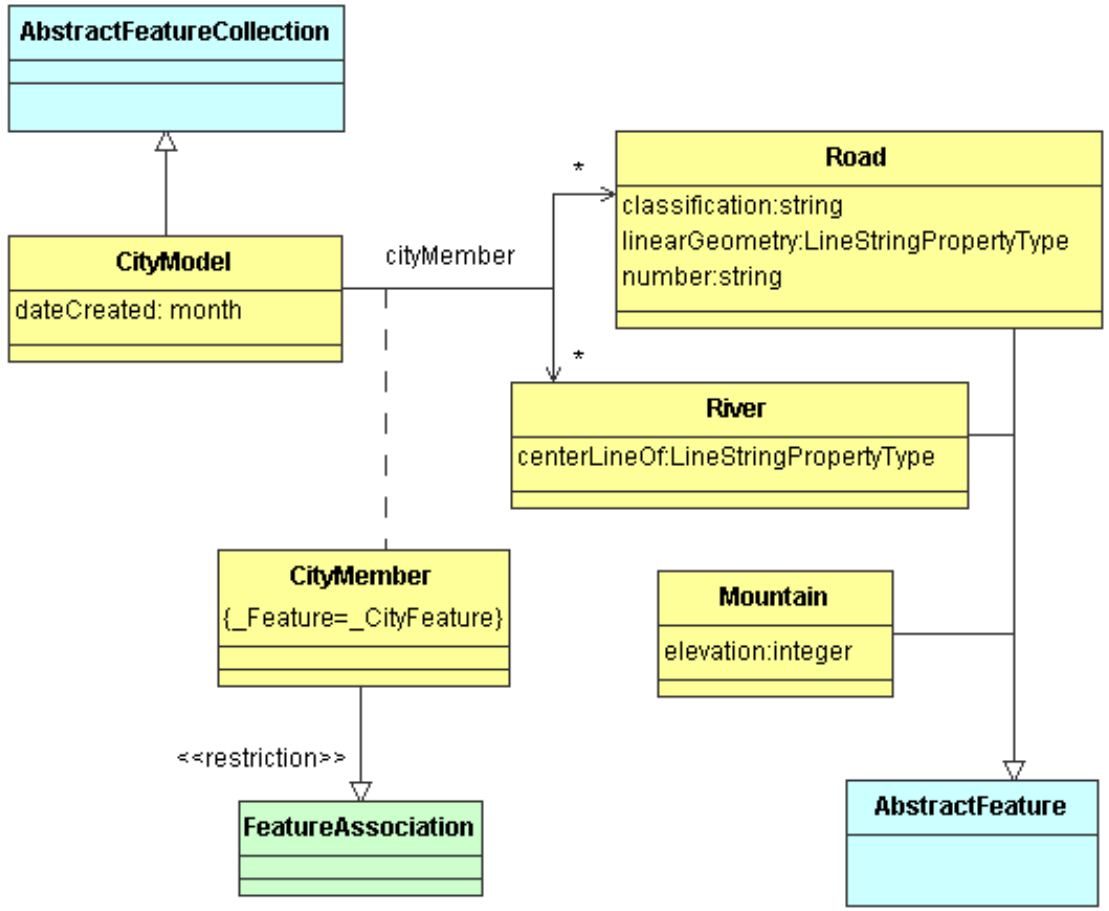
GML instance document

Referncing the GML application schema (definition: see previous slide)

Unique ID

```
<?xml version="1.0" encoding="UTF-8"?>
<app:Building xmlns:gml="http://www.opengis.net/gml/3.2"
              xmlns:app="http://example.com/application"
              gml:id="building1">
  <!-- Geometry of the building -->
  <app:geometry>
    <gml:Polygon>
      <gml:exterior>
        <gml:LinearRing>
          <gml:posList>0 0 0 10 10 10 10 0 0 0</gml:posList>
        </gml:LinearRing>
      </gml:exterior>
    </gml:Polygon>
  </app:geometry>
  <!-- Height of the building -->
  <app:height>20.0</app:height>
  <!-- Other properties specific to buildings can be added here -->
</app:Building>
```

UML Application schema
GML Example Application Schema



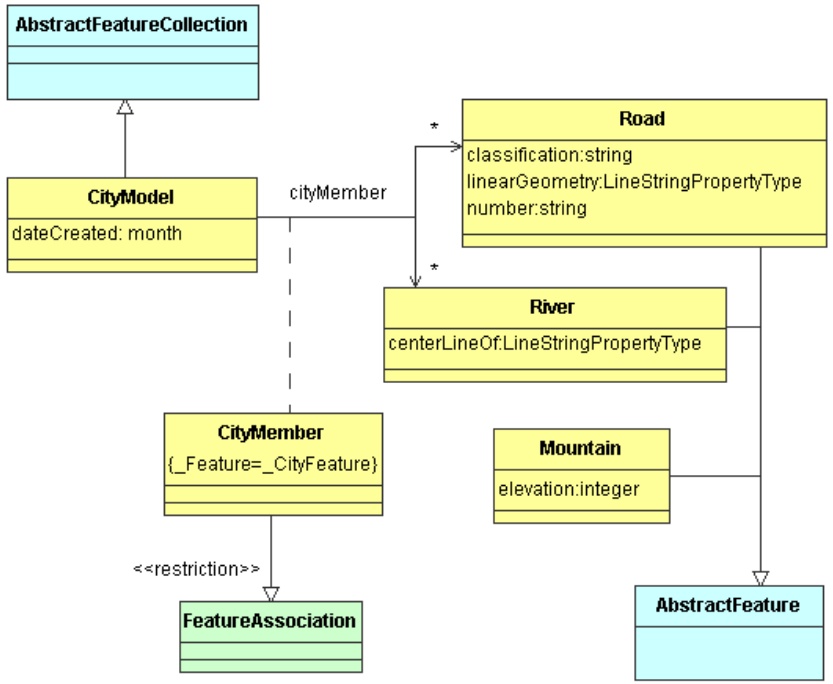

```
<complexType name="CityModelType">
    <complexContent>
      <extension base="gml:AbstractFeatureCollectionType">
        <sequence>
          <element name="dateCreated" type="gYearMonth"/>
        </sequence>
      </extension>
    </complexContent>
  </complexType>

<complexType name="CityMemberType">
   <complexContent>
      <restriction base="gml:FeaturePropertyType">
        <sequence>
          <element ref="ex:_CityFeature"  minOccurs="0"/>
        </sequence>
        <attributeGroup ref="gml:AssociationAttributeGroup"/>
      </restriction>
    </complexContent>
</complexType>

<complexType name="RoadType">
    <complexContent>
      <extension base="gml:AbstractFeatureType">
        <sequence>
          <element name="linearGeometry" type="gml:LineStringPropertyType"/>
          <element name="classification" type="string"/>
          <element name="number" type="string"/>
        </sequence>
      </extension>
    </complexContent>
 </complexType>
```

Data Type Definition

Feature Encodings
GML Example Application Schema




```
<complexType name="CityModelType">
    <complexContent>
      <extension base="gml:AbstractFeatureCollectionType">
        <sequence>
          <element name="dateCreated" type="gYearMonth"/>
        </sequence>
      </extension>
    </complexContent>
  </complexType>

<complexType name="CityMemberType">
   <complexContent>
      <restriction base="gml:FeaturePropertyType">
        <sequence>
          <element ref="ex:_CityFeature"  minOccurs="0"/>
        </sequence>
        <attributeGroup ref="gml:AssociationAttributeGroup"/>
      </restriction>
    </complexContent>
</complexType>

<complexType name="RoadType">
    <complexContent>
      <extension base="gml:AbstractFeatureType">
        <sequence>
          <element name="linearGeometry" type="gml:LineStringPropertyType"/>
          <element name="classification" type="string"/>
          <element name="number" type="string"/>
        </sequence>
      </extension>
    </complexContent>
  </complexType>
```

Feature Definition

```
<element name="CityModel" type="ex:CityModelType"/>
<element name="cityMember" type="ex:CityMemberType" substitutionGroup="gml:featureMember"/>
<element name="Road" type="ex:RoadType" substitutionGroup="ex:_CityFeature"/>
<element name="River" type="ex:RiverType" substitutionGroup="ex:_CityFeature"/>
<element name="Mountain" type="ex:MountainType" substitutionGroup="gml:_Feature"/>
<element name="_CityFeature" type="gml:AbstractFeatureType" abstract="true" substitutionGroup="gml:_Feature"/>
```

GML encoding of a data model

- UML defined according to ISO 19109 can be *automatically transformed* into a GML application schema (an XML Schema encoding).
- GML = Geography Markup Language / ISO 19136
  - GML is an extensive XML-based implementation of the ISO TC211 standards, particularly around spatial types, coordinate reference systems etc.

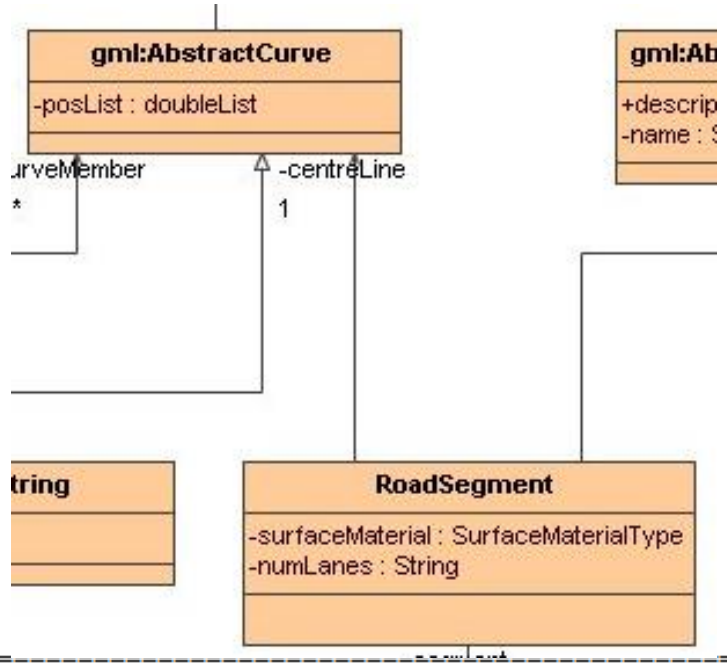
GML encoding of a data model (cont.)

Table D.2 — Implementation of types from the ISO 19100 series of International Standards

UML class	GML object element	GML type	GML property type
GM_Object	gml:AbstractGeometry	gml:AbstractGeometryType	gml:GeometryPropertyType
GM_Primitive	gml:AbstractGeometricPrimtive	gml:AbstractGeometricPrimtiveType	gml:GeometricPrimtivePropertyType
DirectPosition	—	—	gml:DirectPositionType
GM_Position	—	—	gml:geometricPositionGroup (group)
GM_PointArray	—	—	gml:geometricPositionListGroup (group)
GM_Point	gml:Point	gml:PointType	gml:PointPropertyType
GM_Curve	gml:Curve	gml:CurveType	gml:CurvePropertyType
GM_Surface	gml:Surface	gml:SurfaceType	gml:SurfacePropertyType
GM_PolyhedralSurface	gml:PolyhedralSurface	gml:PolyhedralSurfaceType	*anonymous property type*[a]
GM_TriangulatedSurface	gml:TriangulatedSurface	gml:TriangulatedSurfaceType	*anonymous property type*
GM_Tin	gml:Tin	gml:TinType	*anonymous property type*
GM_Solid	gml:Solid	gml:SolidType	gml:SolidPropertyType
GM_OrientableCurve	gml:OrientableCurve	gml:OrientableCurveType	gml:CurvePropertyType
GM_OrientableSurface	gml:OrientableSurface	gml:OrientableSurfaceType	gml:SurfacePropertyType

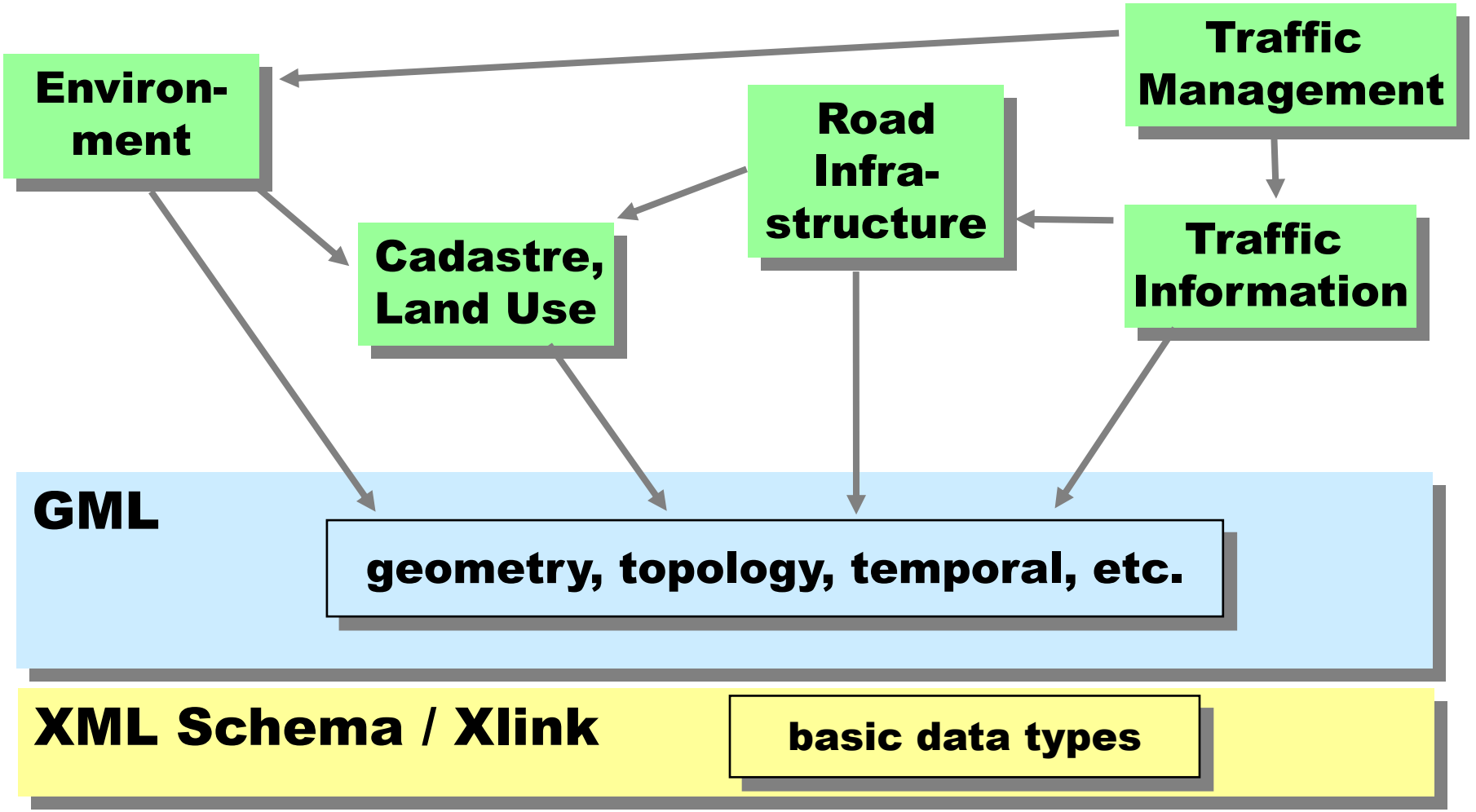
See Annex D of GML 3.2.1 for complete table (several pages long)

Design of application schemas (UML to GML)




```
<!-- ==========================================================
-<complexType name="RoadSegmentType">
 -<complexContent>
  -<extension base="app:AbstractFeatureType">
   -<sequence>
      <element name="surfaceMaterial" type="app:SurfaceMaterialType"/>
      <element name="numLanes" type="integer"/>
      <element name="centreLine" type="gml:CurvePropertyType"/>
    </sequence>
  </extension>
 </complexContent>
</complexType>
<!-- ==========================================================
```

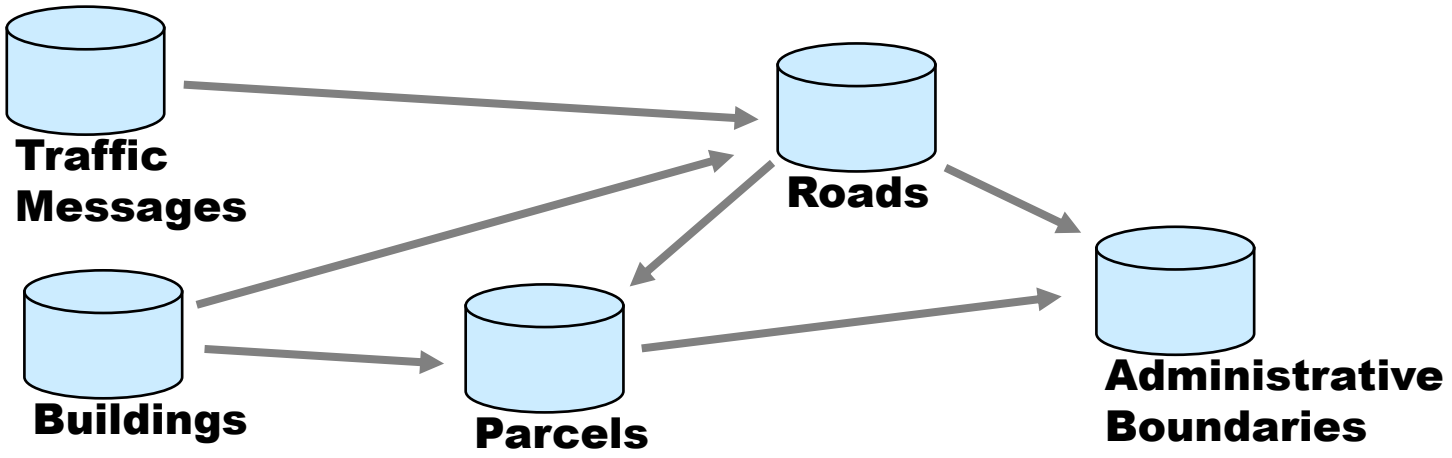
The geospatial web
Linking GML Application Schemas



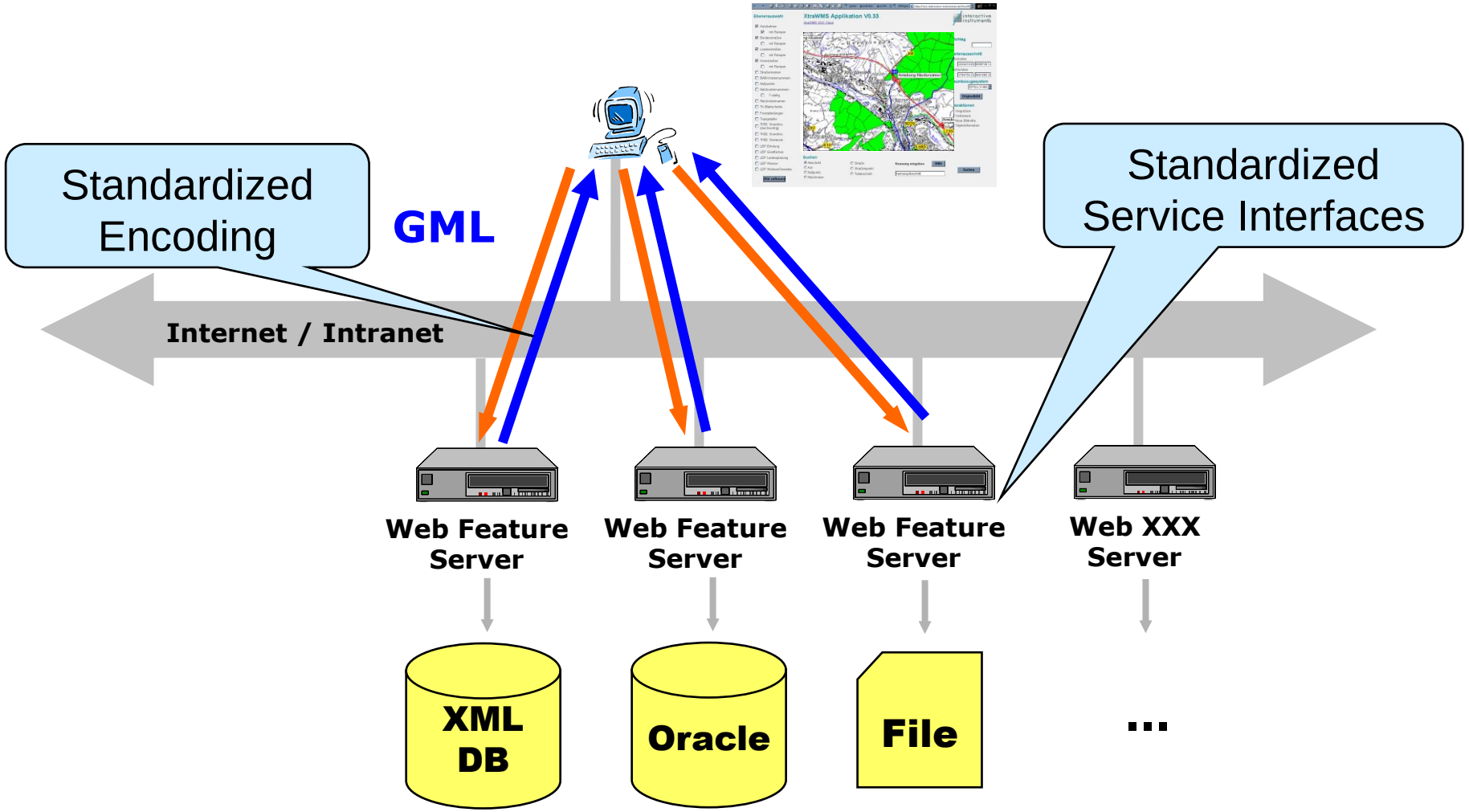
Enabling the geospatial web

- Information Communities publish their Application Schemas (preferably in some sort of registry) so that it can be found, accessed and understood by others
- This enables that also the features can have properties whose values are maintained by other authorities

  → a web of geospatial features is created

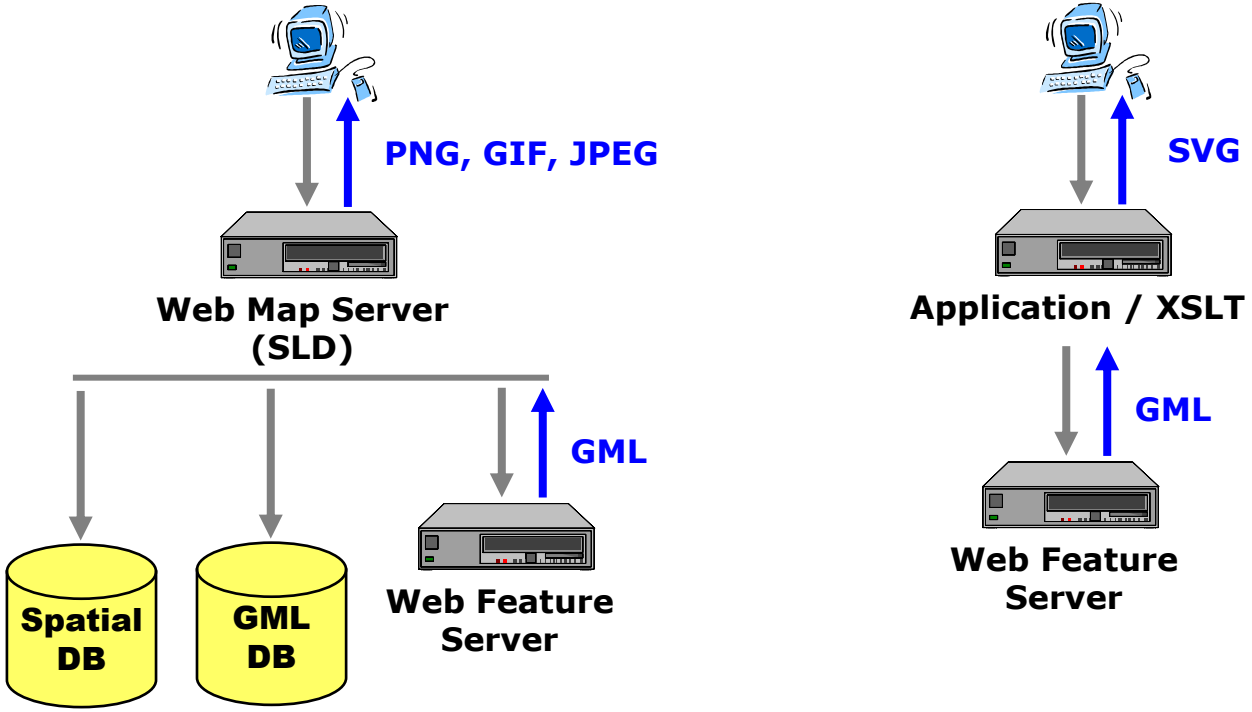


... and use GML as the lingua franca of the geospatial web



3 Mapping GML Data

GML is focused on content and not on visualization!



Support for application schema designers

- Rules for Application Schemas
  - Guidelines for the usage of XML Schema
  - GML documents can be interpreted more easily by software („GML parsers“)
- ISO 19109 “Rules for Application Schemas”
  - Framework / meta model for the definition of application schemas in UML and GML
- Tools to map from UML or other modeling languages to GML (e.g. Shape Change of Interactive Instruments))
  - http://www.interactive-instruments.de/shapechange/
- Using a GML Profile in an Application Schema
  - A declaration of the subset of GML used by an application
  - GML itself includes a simple tool that allows to create such a GML profile automatically

Support for software developers

- XML Parsers, XSLT processors, etc. are available (including Open Source ones); as XML is popular in general many developers know how to work with and process XML documents
- GML Parsers (i.e. GML-aware XML parsers understanding the GML model and syntax) are emerging
  (e.g. FME, Tatuk GIS, GAIA (carbon project), Snowflake, etc.)
- Most major GIS products have in their latest releases built-in support for GML;
- in addition a significant number of new products providing OGC Web Service interfaces and serving GML via COTS (e.g. Oracle, ArcGIS Sever, etc.)

Summary

- GML 3.2.1 is an OpenGIS® Specification
- ISO 19136 is the ISO specification of GML (GML 3.2.1)
- Most recent OGC Implementation Specifications are linked to GML
  - E.g. IndoorGML, LandInfra/InfraGML
- A number of GML enabled products have been released
- Provides a rich set of predefined types for Application Schemas
- Has an underlying model that makes processing GML documents easier (GML Profiles)
- Separates presentation and content
- Works well in a Web Service environment

→ A building block of the Geospatial Web

Tools & Tutorial

- XML
  - XMLSpy from Altova (widely spread)
    - 30 days evaluation license:
    - https://www.altova.com/xmlspy-xml-editor/download
  - OXyGEN
    - 30 days evaluation license, cheap academic lincensing
    - http://www.oxygenxml.com/
  - XML – Tutorial – W3Schools:
    - http://www.w3schools.com/xml/
- GML
  - https://en.wikipedia.org/wiki/Geography_Markup_Language
  - https://inspire.ec.europa.eu/training/basic-concepts-xml-and-gml
- KML
  - https://developers.google.com/kml/documentation/kmlreference

Mandatory reading assignment

- Use the w3schooools Homepage and work through the following tutorials (please note that the content of those tutorials will form part of the exam)
  - XML – Basics

    http://www.w3schools.com/xml/default.asp
  - XML Schema

    http://www.w3schools.com/xml/schema_intro.asp