Grundzüge der Geoinformatik (GI)

Kapitel 3

Univ.-Prof. Dr.-Ing. Wolfgang Reinhardt
AGIS / Inst. Für Angewandte Informatik (INF4)
Universität der Bundeswehr München
Wolfgang.Reinhardt@unibw.de
www.agis.unibw.de

Allgemeines

- Interoperabilität in GIS
- GeoWeb Services, Geodateninfratsrukturen (GDI, SDI) - Einführung

➢ Skripten:

http://www.unibw.de/inf4/professuren/geoinformatik/lehre/informatik

Ziele und Inhalte – Vorlesung GI

Ziele: **Grundprinzipien kennen**
Wissen was man mit GIS bearbeiten kann
Einfache Aufgaben durchführen können,

Inhalte: Kapitel 3

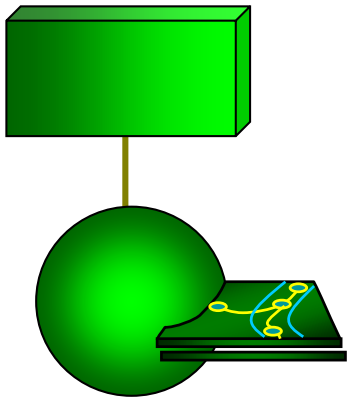
- **Grundproblematik** der Interoperabilität, Überblick über versch. Arten der Interoperabilität, Beispiele verschiedener Modellierung, Aufzeigen der Problematik und Lösungsansätze
- Einführung in GeoWebservices, OGC Services WMS, WFS

Overview

- The initial situation / requirements from the markets
- Short reminder: GI Modelling, Data exchange
- Interoperability in general and related forms, problems and approaches to solve it
  - Syntactic Interoperability
  - Schema based interoperability
  - Semantic interoperability
- Short introduction into Ontologies for Geoinformation

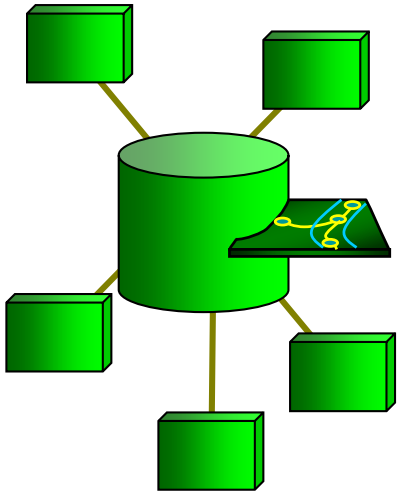
GIS has changed

Desktop



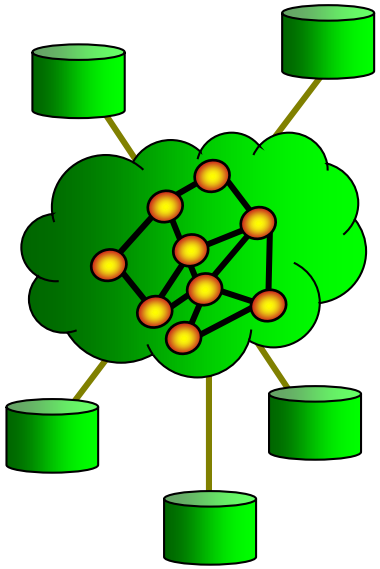
Projects
Stand alone

Client-Server



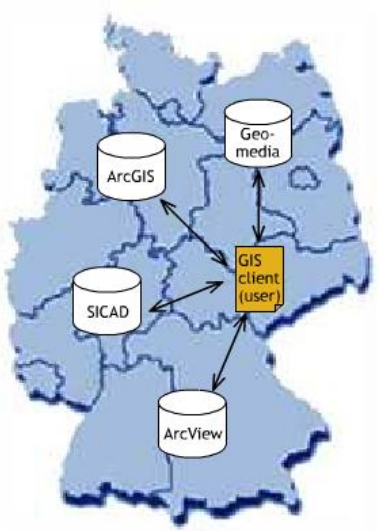
Information-
Management

Web Services



Infrastructures

GIS from different vendors e.g. at state level



Example from Germany

Requirements:

Access to GI data:

- at different locations
- from different domains, **modelled in different ways**
- **stored in different vendor systems**

Semantics + Schema

Challenges in Data exchange

Syntax!

Reminder: content of GI data / models

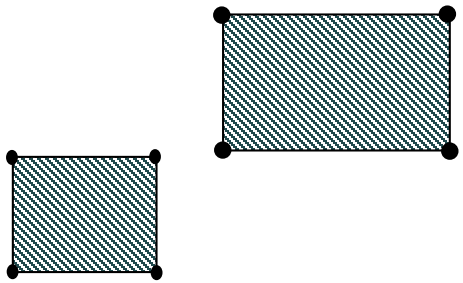
1. Models / Schema information
   - feature classes, structures, relations …meanings of the terms …
   - Basic information (metrics, reference systems …)

2. Features
   - Semantics
   - Geometric properties (Geometry / Topology)
   - Thematic properties (attributes)
   - Graphical representation (partly)

3. Meta data

Example
Feature class: Buildings

Buildings
Geometric properties
Thematic properties



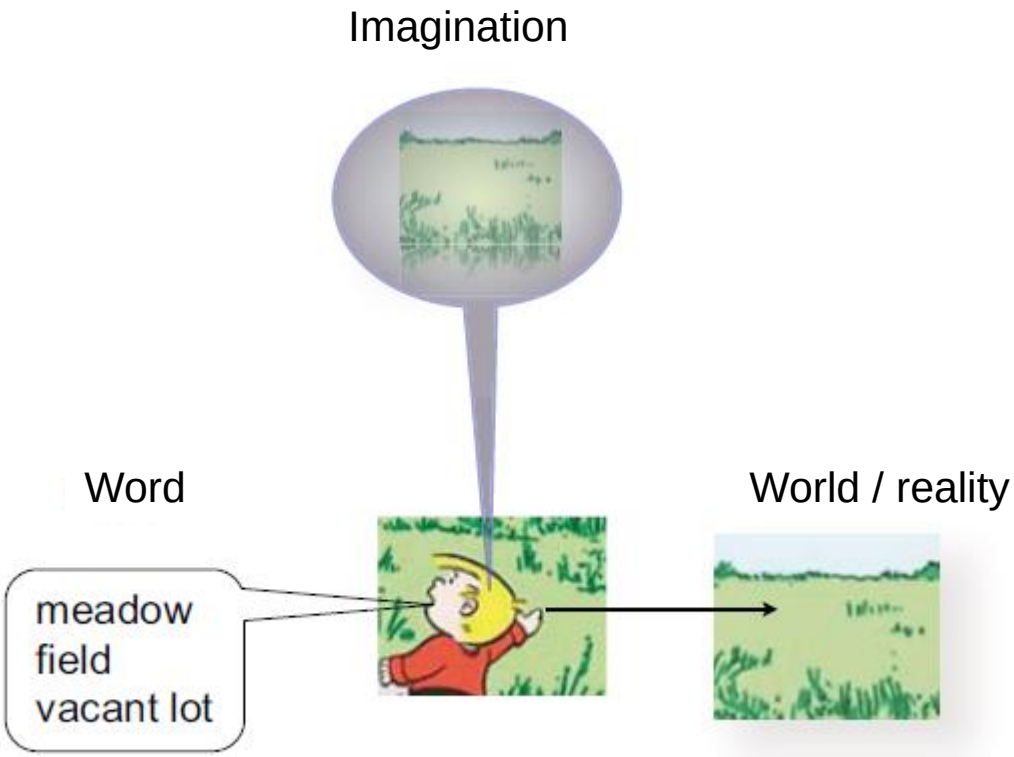
2 instances

Initial Situation

Semantics: different words for the same thing:

"Is this a meadow, a field, or a vacant lot?"

Semantics: a mater of background / domaine and imagination



Semantics: different definitions in different domaines and regions / countries

A river is a "body of inland water flowing for the most part on the surface of the land but which may flow underground for part of its course"

http://www.euwfd.com/html/glossary.html

Semantics -> consequences

Swiss van driver gets stuck up a 'glorified goat track,' blames GPS

http://www.engadget.com/2010/09/29/swiss-van-driver-gets-stuck-up-a-glorified-goat-track-blames/

Semantics -> who cares?

Different names of feature classes

Feature classes can have different names depending on the user domain.

User domain A

Model A

Feature class = lake 

Attribute:

- name
- area
- max. depth
- protected

User domain B

Model B

 Feature class = fish pond

Attribute:

- size
- type of fish

Homonyms - synonyms

Homonyms = One word, but different meanings
e.g. can: to be able to do something or a metal container

Synonyms = Different words, but the same meaning
e.g. screen = monitor

Additionally: Different interpretations in different domains!

For example the German word “ausmachen” has 10 meanings
http://german.about.com/od/vocabulary/a/ausmachen.htm

Different ways of modeling

Features are modeled in different ways in different application domains.

Example: Lake

Application domain A

Model A

Attributes:

- name
- geometry (polygon)
- max. depth
- protected

Application domain B

Model B

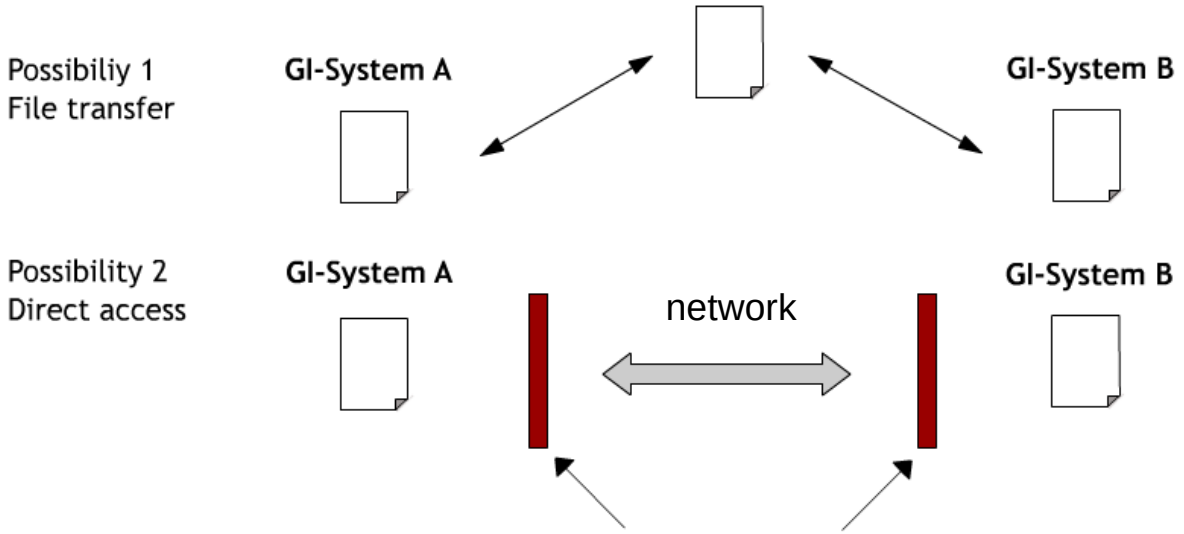
Attributes:

- name
- geometry (point)
- depth
- average temperature
- water quality
- pH value
- vegetation

Problem within data exchange / integration:
Different data models for the same feature class

Requirement of data exchange and access (e.g. through networks / web) without loss of information

Data transfer / file based / -> formats (syntax!)



Data access through interfaces / requires exact definition of the Interface structure

Data exchange: Methods

Data oriented exchange:

- Model information (schema) must be known by the user of the system, which receives the data.
- File format syntax must be known as well.
- With this approach only data is exchanged via specific file formats (DXF etc.).

Now we can define three levels of interoperability:

Syntax	⇨	Syntactic interoperability
Model / Schema	⇨	Schema based interoperability
Semantics	⇨	Semantic interoperability

General definitions

Syntax: [from Greek - Latin: Compound sentence]

Arrangement of word in sentences, clauses, and phrases, and the study of the formation of sentences and the relationship of their component parts. In a language such as English, the main device of showing the relationship among words is word order [...].
(Source: Encyclopaedia Britannica - Micropaedia, 1974)

Semantics: [from Greek: Significant]

Semantics is the philosophical and scientific study of meaning [...], in particular, of linguistic meaning. [In the field of linguistics semantics is] the study of the elements of a language form the point of view of meaning; and syntax, the study of the formal interrelations that exist between the elements of a language (i.e., sounds, words) themselves.
(Source: Encyclopaedia Britannica - Macropaedia, 1974)

Definition: Interoperability

Examples of definitions:

Interoperability is "the ability of information systems to operate in conjunction with each other encompassing communication protocols, hardware, software, application and data compatibility layers". (Source: Glossary - ANVIL: A Network Virtual Interoperability Laboratory)

Interoperability is "the ability of two or more systems to exchange geospatial information and to make mutual use of the information that has been exchanged". (Source: Bishr, 1997)

Interoperability is a property referring to the ability of diverse systems and organizations to work together (inter-operate). The term is often used in a technical systems engineering sense, or alternatively in a broad sense, taking into account social, political, and organizational factors that impact system to system performance.
(Source, wikipedia, 2009)

- **Syntactic interoperability** If two or more systems are capable of communicating and exchanging data, they are exhibiting syntactic interoperability. Specified data formats, communication protocols and the like are fundamental.

⇨ Syntactic interoperability is ensured, if:

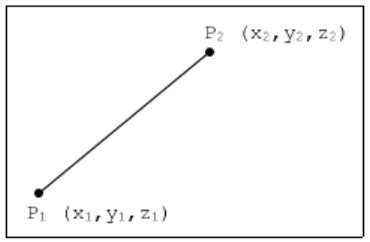
- ⇨ All involved system exchange data via one or more data formats which are supported by these systems
- ⇨ These formats are able to transfer all relevant information (features, geometry, topology …)
- ⇨ The involved systems know the underlying model of the data

⇨ Here only examples of data formats are given, not an examination of there capabilities, as this depends on the specific requirements

Example: DXF-syntax

Example how a syntax looks like:

```
0
SECTION      = begin of section
2
ENTITIES     = entity
0
LINE         = feature
8
0
10           x1
10.0
20           y1
10.0
30           z1
0.0
11           x2
20.0
21           y2
50.0
31           z2
0.0
0
ENDSEC       = end of section
0
EOF          = end of file
```



One other important format:
Shape (ESRI)

Geometry oriented
But extensions for semantic: “ENTITIES”

Example: GML - code

Encoding with GML:

```
...
<property typeName="Hausnr" type="string">8</property>
<gml:Polygon xmlns:gml="http://www.opengis.net/gml"
   srsName="http://www.opengis.net/gml/epsg.xml#4326">
   <gml:outerBoundaryIs>
         <gml:LinearRing>
               <gml:coordinates cs="," decimal="." ts=" ">
                     10.4,10.3 10.4,10.7 10.8,10.3 10.8,10.7
               </gml:coordinates>
         </gml:LinearRing>
   </gml:outerBoundaryIs>
</gml:Polygon>
...
```

- Schema based interoperability often is seen as part of semantic interoperability

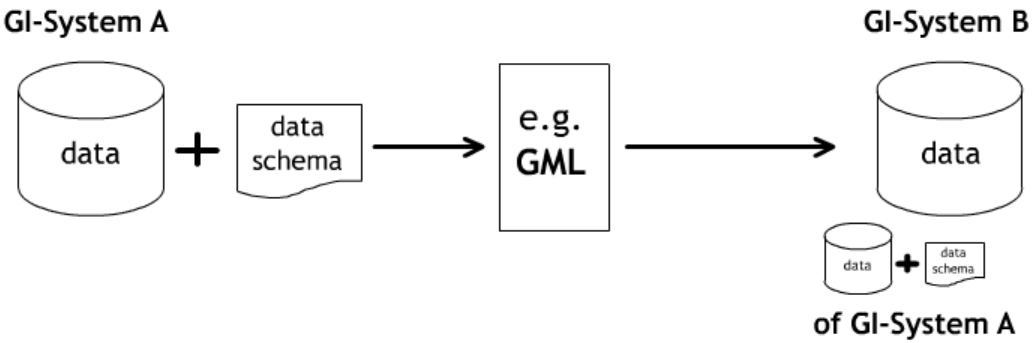
  For reasons of clearness it is categorized separately here

  Two levels can be considered:

- The pure transfer of the schema together with data

- The integration of data with different underlying models

Transfer of data and data schema of the model



In this case GI-System B need not have any information about the data before.
It receives it through the data schema.

```
- <xsd:schema attributeFormDefault="unqualified" elementFormDefault="qualified">   begin of schema
    <xsd:import schemaLocation="http://www.deegree.org/xml/schemas/wcts/feature.xsd"/>   begin of schema declaration
    <xsd:element name="ResultCollection" substitutionGroup="gml:_FeatureCollection"/>
    <xsd:element name="DistrictModel" type="gtp:DistrictModelType"/>
    <xsd:element name="DistrictMember" substitutionGroup="gml:featureMember" type="gtp:DistrictMemberType"/>
    <xsd:element abstract="true" name="_DistrictFeature" substitutionGroup="gml:_Feature" type="gml:AbstractFeatureType"/>
    <xsd:element abstract="true" name="_AbstractGeoFeature" substitutionGroup="gtp:_DistrictFeature" type="gtp:AbstractGeoFeatureType"/>
  + <xsd:complexType name="DistrictModelType"></xsd:complexType>
  + <xsd:complexType name="DistrictMemberType"></xsd:complexType>
  + <xsd:complexType name="AbstractGeoFeatureType"></xsd:complexType>   end of schema declaration
  - <xsd:element name="Way" substitutionGroup="gtp:_DistrictFeature">   begin of feature class: way
    - <xsd:complexContent>
      - <xsd:extension base="gtp:AbstractGeoFeatureType">   abstract geofeature
        - <xsd:sequence>
            <xsd:element name="multiCenterLineOf" type="gml:MultiLineStringPropertyType"/>   geometry of feature
            <xsd:element name="width" nillable="true" type="xsd:float"/>   numerical attribute
          - <xsd:element name="category">   begin of enumerated attribute type: category
            - <xsd:simpleType>
              - <xsd:restriction base="xsd:string">
                  <xsd:enumeration value="street"/>
                  <xsd:enumeration value="state highway"/>
                  <xsd:enumeration value="asphalted way"/>
                  <xsd:enumeration value="footpath"/>
                  <xsd:enumeration value="path"/>
                  <xsd:enumeration value="other"/>
                </xsd:restriction>
              </xsd:simpleType>
            </xsd:element>   begin of enumerated attribute type: category
          </xsd:sequence>
        </xsd:extension>
      </xsd:complexContent>
    </xsd:element>   end of feature class: way
  </xsd:schema>   end of schema
```

Example of the transfer of schema information using GML

→ Explanation see course 5

Features are modeled in different ways in different application domains.

Example: Lake

Application domain A

Model A

Attributes:

- name
- geometry (polygon)
- max. depth
- protected

Application domain B

Model B

Attributes:

- name
- geometry (point)
- depth
- average temperature
- water quality
- pH value
- vegetation

Question:

Can we „integrate“ data from A into B and vice versa from B into A?
Sufficiently?
Which one is more difficult?

There is no general way to do the integration of different models

But in this case, there are possibilities under certain assumptions
Can you describe a possible solution?

The user can do a schema integration

- ⇨ Related to the thematic properties (attributes) this is not a problem (if the new model includes all attributes of both models, which can lead to a necessary additional data acquisition)

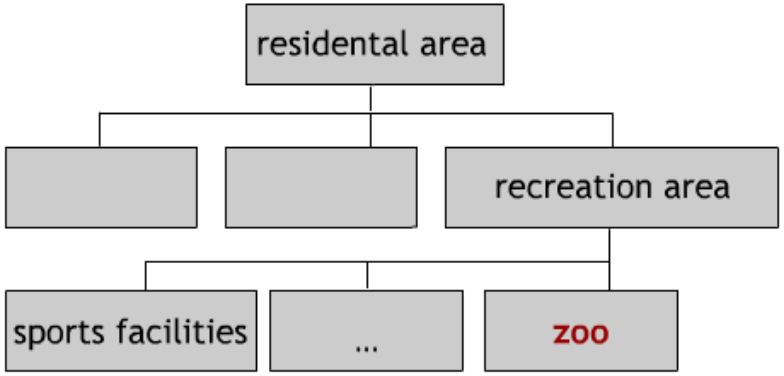
- ⇨ Related to the geometric properties it is more complicated
  - ⇨ If the user chooses the point geometry, for all geometries of model A points have to be calculated (which can be done automatically)
  - ⇨ If he chooses the area geometry, an additional data acquisition has to be performed for all instances of model B

Semantics can be modeled in various ways:

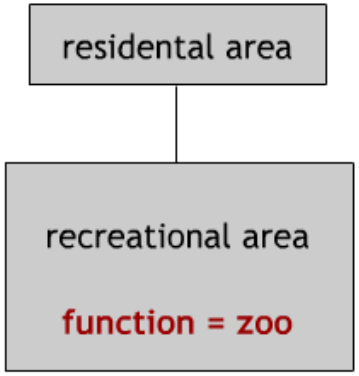
- As a feature class or sub class
- In terms of an attribute of a feature class

An example: Zoo

Topographical data model 1



Topographical data model 2



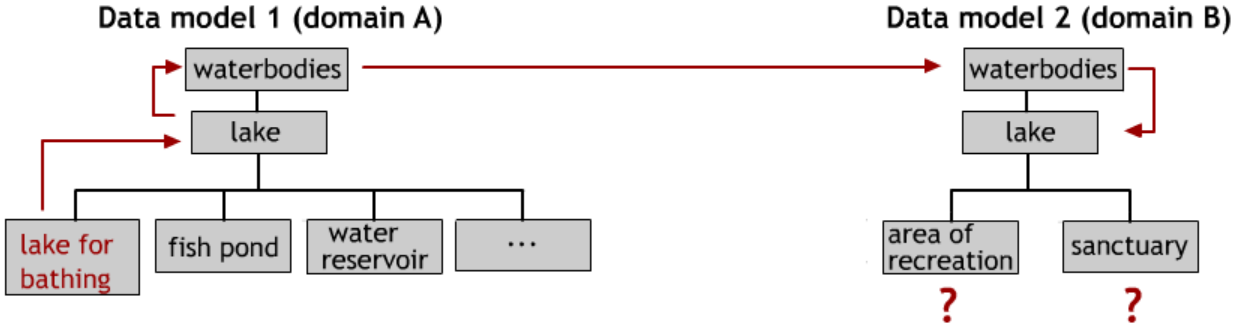
In this case the zoo is modeled as feature class in model 1, and in model 2 it is modeled as feature "recreation area" with an attribute "function", with the value "zoo".
To ensure an interoperable use of both models a "mapping" is needed (e.g. a table), which links the identical information.

Same term (name) but different meanings

Example: Lake

- Data model A: Lake = bathing lake, fish pond, water reservoir, ...
- Data model B: Lake = recreational area, sanctuary

Question: What is the equivalent of lake in data model 1 compared to data model 2?



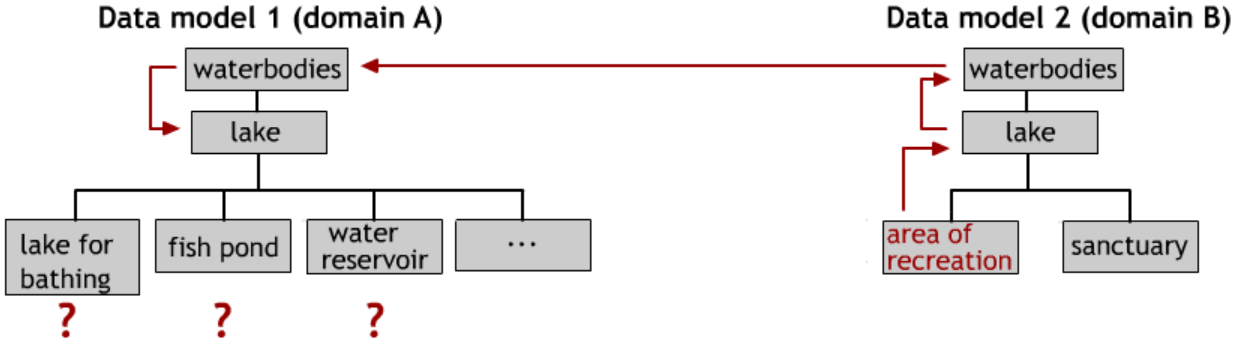
Question: Which object refers to lake for bathing in data model B?

Same term (name) but different meanings

Example: Lake

- Data model A: Lake = bathing lake, fish pond, water reservoir, ...
- Data model B: Lake = recreational area, sanctuary

Question: What is the equivalent of lake in data model 1 compared to data model 2?



Question: Which object refers to "area of recreation" in data model A?

Solution: Agreements on terms or use of ontologies

Problems of different names

Same meaning but different names

Example: House
In the different domains different languages are used.

Question: Which object is the equivalent of "Haus" in data model B?

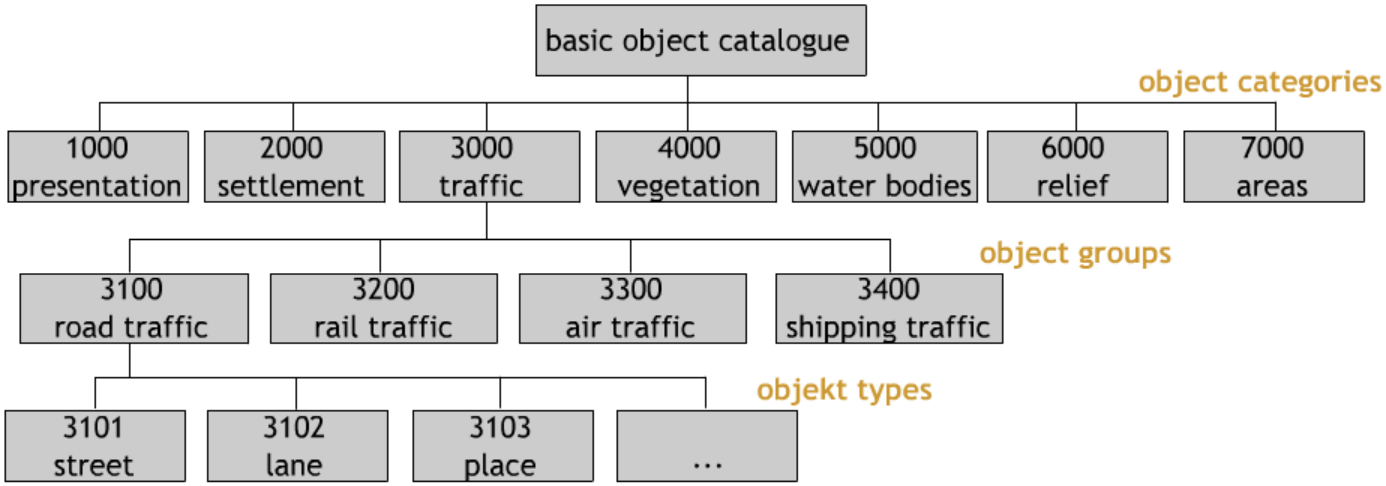
- **Semantic interoperability:** Beyond the ability of two or more computer systems to exchange information, semantic interoperability is the ability to automatically interpret the information exchanged meaningfully and accurately in order to produce useful results as defined by the end users of both systems.

⇨ It is ensured, if both sides agree on a common information exchange reference model. This means an agreement on all terms and meanings is also necessary. It can be achieved using:

- ⇨ Available object catalogues / schemes of the different domaines
- ⇨ Translation tables to link equivalent terms.
- ⇨ Ontologies, semantic web languages, e.g. RDF, OWL (not treated here)

Object catalogue of the German topologic information system

(Source: Arbeitsgemeinschaften der Vermessungsverwaltungen der Länder der Bundesrepublik Deutschland (AdV))



Can be regarded as an application ontology, but in this case there is no formal description of semantics available!

And no relations between classes

OS MasterMap Real World Object Catalogue

Road (surface)

Definition A metalled way for vehicles.
A vehicle is one with wheels on both sides of its body.
Metalling is any artificial (man-made) surface including areas of **asphalt**, **concrete** and **gravel**.

Representation The limits of road metalling are captured. When the driveable carriageway of a road is defined either by a sunken kerb or change in metalling this is not normally be shown. Exceptionally short lengths may be shown when necessary to complete the depiction of the extent of the carriageway, for example at raised junctions.
Roads of any length are captured when they are outside private property. Roads and **drives**, including shared drives, within private property, are only captured when they are over 100 m in length. Exceptionally, short lengths may be shown when necessary to complete the depiction of the extent of the carriageway, for example at raised junctions.
[10 000] Roads are captured to scale or 5.0 m wide if less.
The distinctive names of roads are collected and recorded. Unnamed roads are not normally described.
[2500] [10 000] For Highway Agency road classifications and numbers, see Road Number

Features and Attribute values

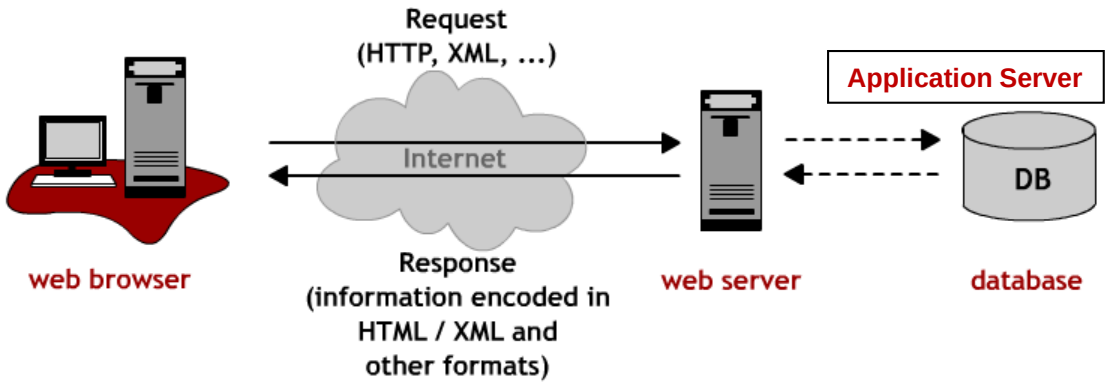
RWO	Function of feature	Feature type	Descriptive group	Descriptive term	Physical presence	Make	Name/Description
Road (surface) Public Road Network	Bounded by	TopographicLine	Road or Track	Public	Edge/Limit	Manmade	<distinctive name only> and/or <classification number> <distinctive name> Rd <distinctive name>
	Extent	TopographicArea	Road or Track			Manmade	
	Text	CartographicText	Roadside				
	Text	CartographicText	Road or Track	Road Name or Classification			
Road (surface) Non-Network	Bounded by	TopographicLine	General Feature		Edge/Limit		
	Extent	TopographicArea	Road or Track			Manmade	
	Text	CartographicText	Roadside				
	Text	CartographicText	Road or Track	Road Name or Classification			

Problem today:
many ontologies availabe, but with different concepts for the same „things“

Geo Web Services (GWS) – Introduction

GWS standardized by the Open Geospatial Consortium (OGC, www.opengeospatial.org) also called OGC Geo Web Services or short OGC Services

Web service architecture



Remember:
In Information Technology (IT) a service in general is understood as encapsulated functionality provided through networks (e.g. web).

⇨ The first approach for Geo Web Services were Map Services

General Remarks

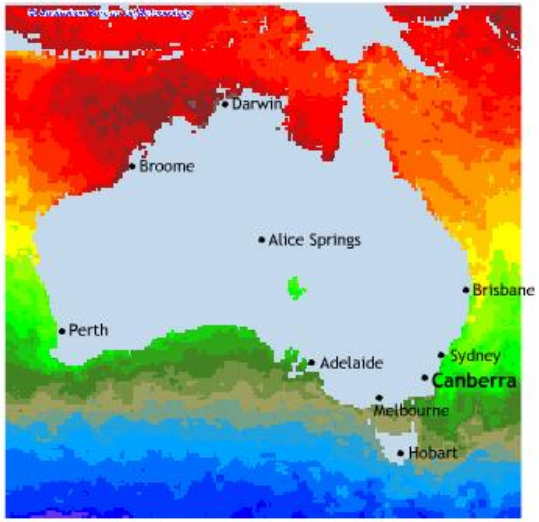
In contrast to analogue maps digital maps can be linked with additional information. These additional information are activated via mouse click on the map.

Therefore these maps are also called "clickable maps".

Types of additional information:

- Texts (e.g. attribute information of objects that are shown on the map)
- Pictures (e.g. photos)
- Videoclips / animations
- Sounds

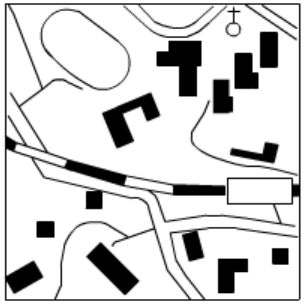
Example:



Data formats for maps:

In general maps can be encoded in raster and/or in vector form:

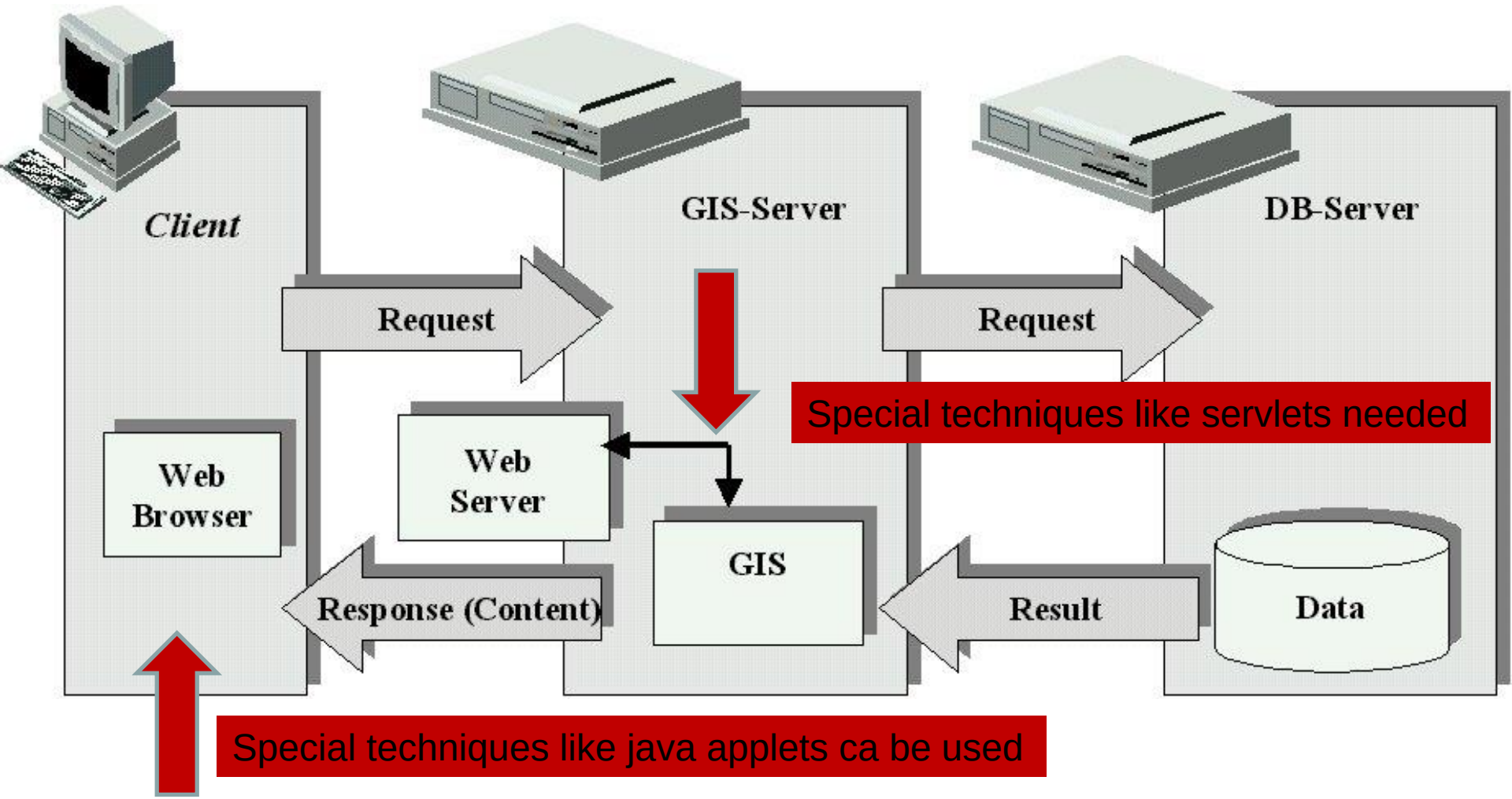
Example of raster formats: GIF, JPEG, PNG, TIFF, BMP



Example of vector formats: SVG, DXF, EPS, CGM etc.

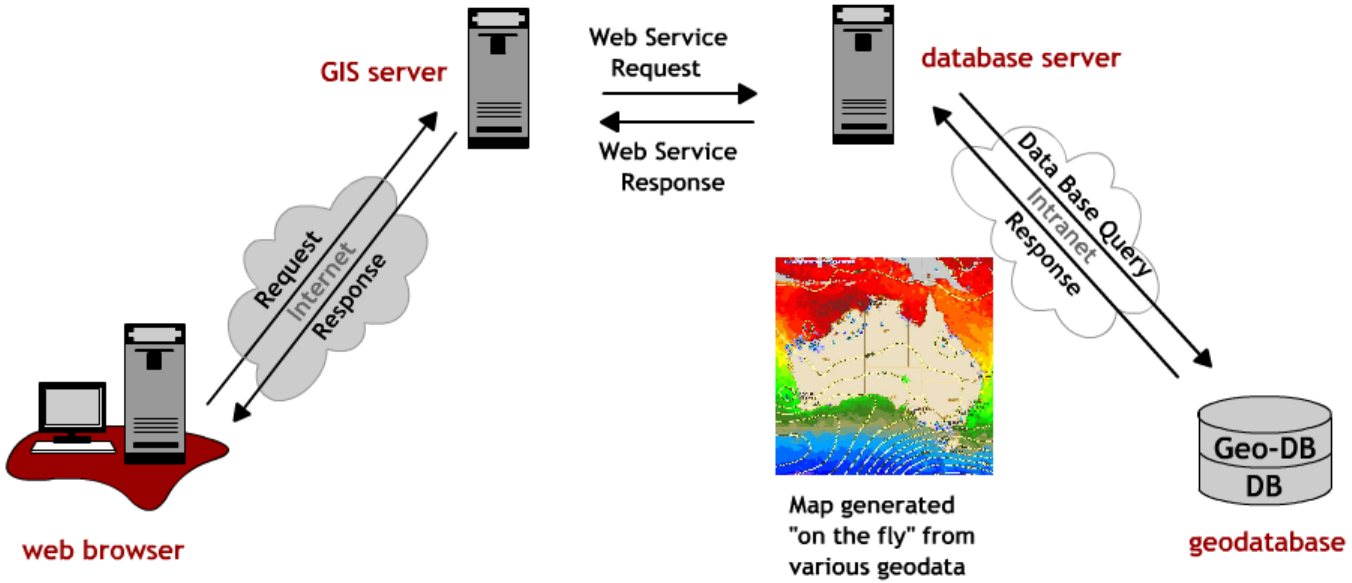
Example of overlay of raster and vector data

Standard formats like JPEG and GIF (and partially SVG) can be viewed using normal web browsers like IE® or Mozilla®. For other formats the browser program has to be extended by means of plug-Ins, activex controls etc.



General architecture of a Internet Map Server

A service that generates maps from a Geo-DB is called Internet Map Service (IMS) or Web Map Service (WMS).



„On the flight“

Difference between format of data stored and data transferred !!!

Advantages of Web Map Services

Web map services based on GIS and Internet technologies offer various advantages:

- A standard technology (Internet / Intranet) is used.
- The handling is easy: **GIS for "everybody"**
- Internet users can access a GIS via browser and use GIS functionality without a GIS software (example: Route planning).
- It is an inexpensive solution (only a client workstation with an Internet access is required).
- GIS functionality and GIS services are available to a wide field of network-based applications in industry, administration and education.
- GIS on the Internet enables participation (e.g. of citizens in planning processes).

- This solution allows for a creation of a network of distributed information.
- Online access to distributed and up-to-date information for a large number of users is provided.
- A spatial data infrastructure (SDI) can be established by means of Web Map Services.

Geo Web Services are the „new“ concept of making using of Geoinformation

There are also very important within SDI / NSDI (National Spatial Data Infrastructures)
Dt.: GDI (Geodateninfrastrukturen)

SDIs are developed at International, national and regional levels

A few remarks on SDI in the next slides

NSDI Definition (FGDC (USA):

Consistent means to share geographic data among all users could produce significant savings for data collection and use and enhance decision making. Executive Order 12906 calls for the establishment of the National Spatial Data Infrastructure defined as the technologies, policies, and people necessary to promote sharing of geospatial data throughout all levels of government, the private and non-profit sectors, and the academic community.

The goal of this Infrastructure is to reduce duplication of effort among agencies, improve quality and reduce costs related to geographic information, to make geographic data more accessible to the public, to increase the benefits of using available data, and to establish key partnerships with states, counties, cities, tribal nations, academia and the private sector to increase data availability.

An important milestone: USA, 1994: „Clinton executive order" (V.P. Al Gore)

The Clinton Executive Order 12906, which states, " *Geographic Information is critical to promote economic development, improve our stewardship of natural resources and protect the environment….*

Geo data

Provider X

Geo data

Provider Y

Geo data

Provider Z

1. Organisation:

- *User, technology and Service provider*
- *Public-private-partnerships*
- Reference (Goals, Standards, access rights,…)
- Business model (allocation of tasks)
- Consence oriented!

User A

User B

Geo data
Provider X

Geo data
Provider Y

Geo data
Provider Z


User A

1. Organisation

2. Standards & regulations

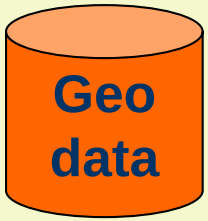
- Standards
  - *Lingua franka, to provide* interoperability
  - *de facto*:
    - W3C, OASIS,…
    - OpenGeoSpatial Consortium
  - *de jure:*
    - ISO, CEN, DIN
  - Regulations & laws
    - Implementation regulations ..
    - Geodata laws, costs, conditions of usage


User B

Provider X

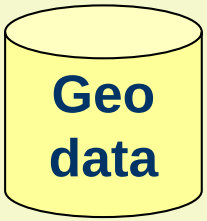
1. Organisationsstruktur
2. Standards & Richtlinien
3. Distributed Geo Services
   - Internet as a base
   - Standards for interfaces and communication, data formats

User A

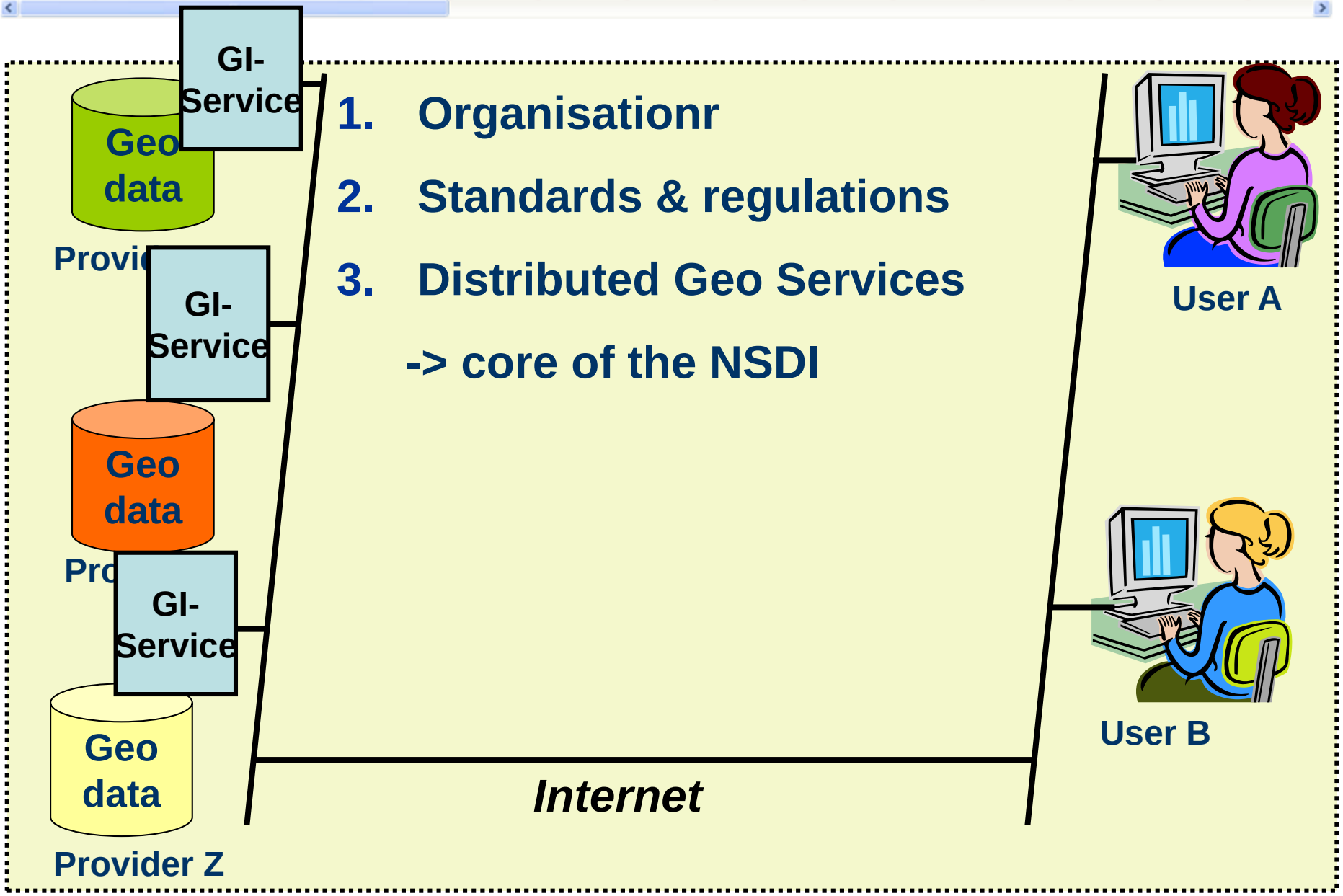


Provider Y

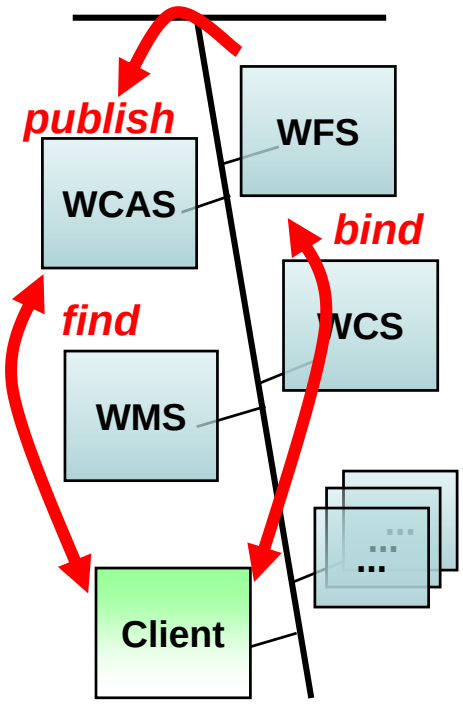
User B



Provider Z


GI-Service
Geo data
Provi
GI-Service
Geo data
Pro
GI-Service
Geo data
Provider Z
1. Organisationr
2. Standards & regulations
3. Distributed Geo Services
-> core of the NSDI
User A
User B
Internet

Geo Web Services within a NSDI



Examples:

- WebMapService (WMS)
- WebFeatureService (WFS)
- WebCoverageService (WCS)
- WebCatalogueService (CSW)

- GI-Services
  - are Web Services (WEB/HTTP).
  - follow ISO/OpenGIS Standards.
  - GI-Services are permanently available
  -
  - Follow the *publish-find-bind* Paradigma.
  - Can be connected „ad-hoc“ (*service chaining).*

SDI architecture – Example from europe

Appl. Layer

Application and Geoportals

Service Bus

Service Layer

GeoRM layers

Network Services

Registry Service

Discovery Service

View Service

Download Service

Transf. Service

InvokeSD Service

Data Layer

Registers

Service Metadata

Data Set Metadata

Spatial Data Set

Thematic DS

TWG | TWG | TWG

Framework for harmonized DS

Metadaten

Data Spec.

Important:
Kind of Services!

INSPIRE:
Infrastructure for Spatial Information in the European Community

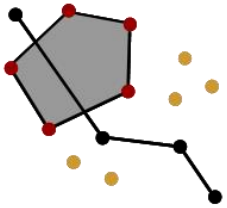
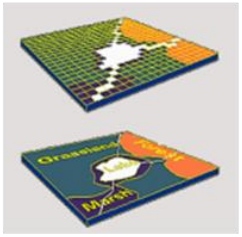
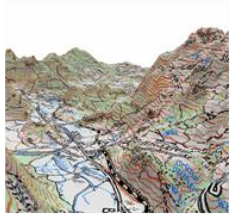
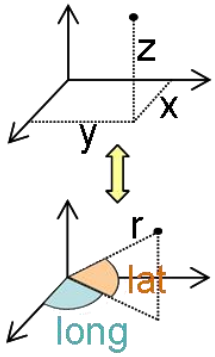
GOALS

- Reduce effort for different agencies
- Improve Quality
- Reduce costs of goegraphic information
- Provide data for various societal tasks
- Fast access to Geoinformation

Terms used: NSDI – SDI – GDI (german)

- **In brief: SDI = geodata + networks + services + standards**

Types of geodata

Maps	Features	Coverages	Terrain Views	Metadata	Coordinate Transformation
					

Corresponding OGC web services

Web Map Service (e.g. JPEG)	Web Feature Service (GML)	Web Coverage Service (e.g. GeoTiff)	Web Terrain Service (e.g. JPEG)	Web Catalogue Service	Web Coordinate Transformation Service

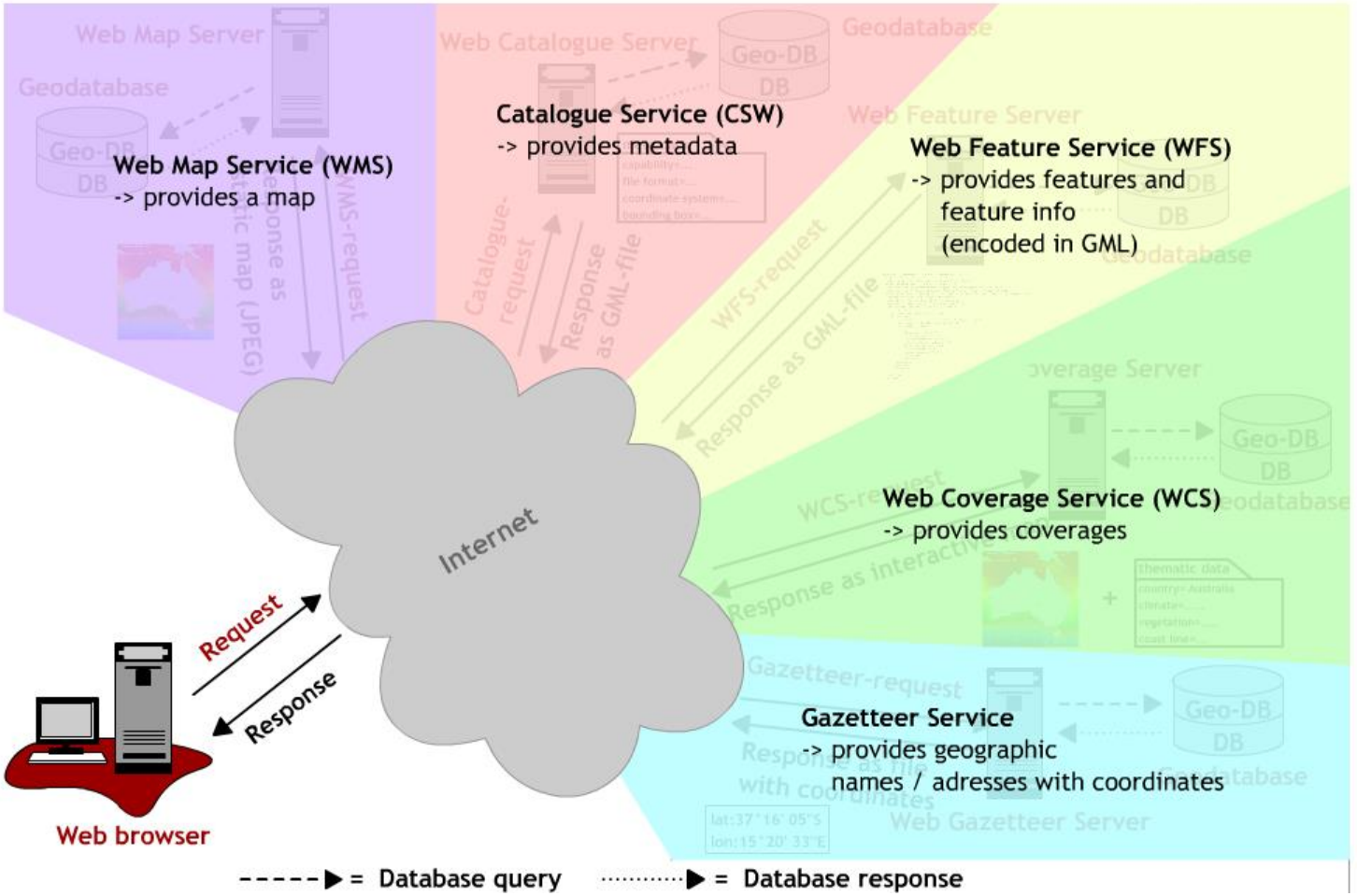
Source: FGDC Geospatial Applications and Interoperability Working Group

OGC web service "Implementation Specifications" for transfer and access of geodata - an overview:

- Web Mapping Service (WMS): The WMS transmits geodata in the form of maps (e.g. in JEPG-format). The WMS generates these maps (in raster format) from several layers, which can even come from different distributed web servers.
- Web Feature Services (WFS): The WFS provides vector data (features / geo-objects), which are encoded in GML (Geography Markup Language -> see course 5).
- Web Coverage Services (WCS): The WCS provides coverages (e.g. aerial images, TINs, matrix data).
- Gazetteer Service: The Gazetteer Service provides geographical names with their spatial reference (coordinates). e.g. addresses
- Catalogue Service (CSW): The Web Catalogue Service provides metadata belonging to available geodata.

These are examples! There are many other ones like WPS, SWE …

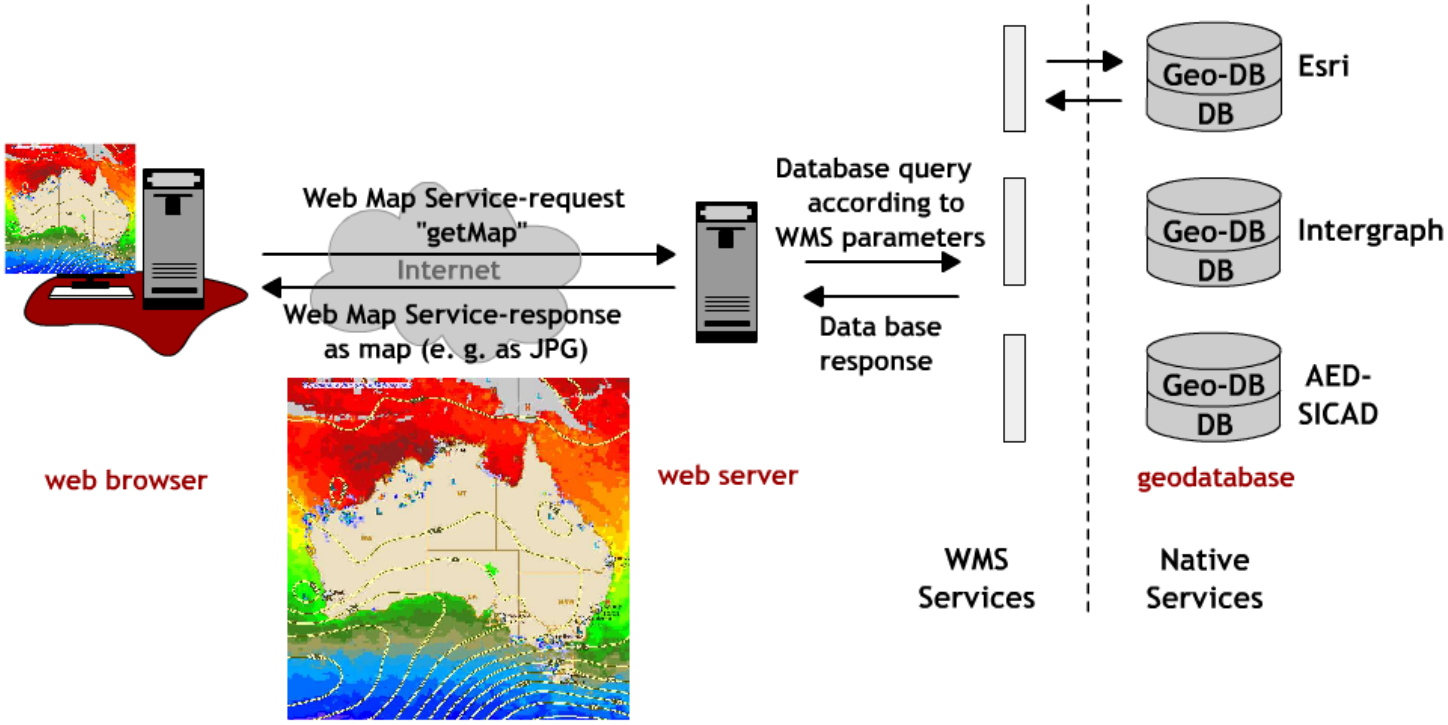
Overview of OGC Web Services



There are other many other OGC services available, e.g. SOS to connect with sensors!

WMS

previous page next page



OGC Web Map Service (WMS):

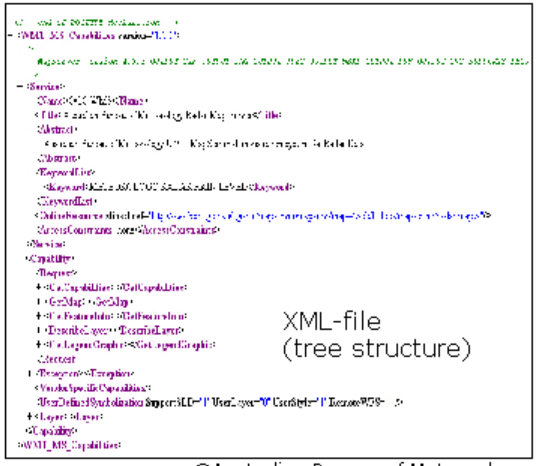
- WMS version 1.1.1 was submitted in January 2002 and became ISO standard with version 1.3.0 in March 2006.
- The WMS interface defines, how maps can be requested by clients. It is possible, to access the proprietary services of different, distributed GIS (based on different vendor software) via WMS.
- WMS provides digital maps in a standardized format, like GIF, JPEG and others.
- A WMS extends a HTTP-request to handle and process WMS requests.
- 3 WMS-requests (GetCapabilities, GetMap, GetFeatureInfo) are defined within the WMS specification.

WMS requests:

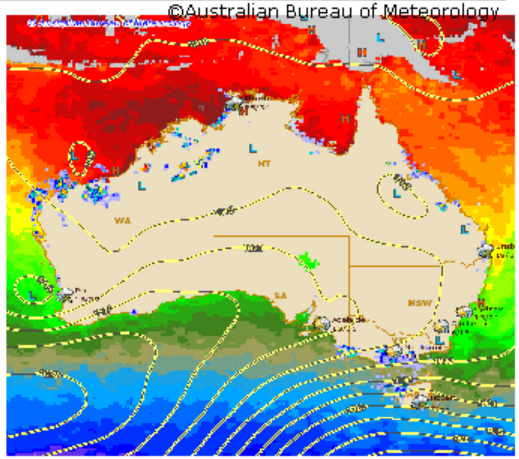
- GetCapabilities: Request for metadata describing the WMS content in order to get information about the capability of the service. This request has to be performed before the Get Map Request, because it delivers input for it, like the available layers, the area etc.

- GetMap: Request for a digital map (see next slide for details)

- GetFeatureInfo: Optional request for information about features, shown on the map. Features are selected by choosing a point on the map.



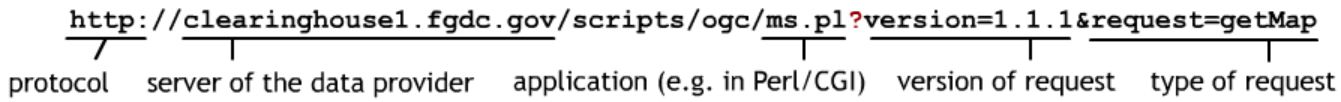
XML-file
(tree structure)



©Australian Bureau of Meteorology

WMS requests:

A WMS request is based on the HTTP common gateway interface (CGI) request, represented by an URL combined with a question mark.



Parameters of GetCapabilities request

VERSION = number of version	Versions# (e. g. "version=1.1.1") - optional
SERVICE = WMS	Which service should be requested - mandatory
REQUEST = GetCapabilities	The name of the request - mandatory
FORMAT = MIME_type	Output format of service meta data - optional

Example of a GetCapabilities-request:

http://ows.bom.gov.au/cgi-bin/mapserver/mapserv?
map=/web/htdocs/mapserver/radar.map&request=GetCapabilities &service=WMS&version=1.1.1

As response on a GetCapabilities-request you receive a XML-document of the capabilities of the web server (see next slide for an example).

Response of GetCapabilities request

Initial request:

http://ows.bom.gov.au/cgi-bin/mapserver/mapserv
?map=/web/htdocs/mapserver/radar.map&request=GetCapabilities &service=WMS&version=1.1.1

Response:

```
- <WMT_MS_Capabilities version="1.1.1">     = version of service
  + <!---->
  - <Service>
      <Name>OGC:WMS</Name>     = name of service
      <Title>Australian Bureau of Meteorology Radar Map Server</Title>     = title of service
    - <Abstract>
        Australian Bureau of Meteorology UMN MapServer demonstration system for Radar Data     = short abstract of service
      </Abstract>
    + <KeywordList></KeywordList>
      <OnlineResource xlink:href="http://ows.bom.gov.au/cgi-bin/mapserver/mapserv?map=/web/htdocs/mapserver/radar.map&"/>
      <AccessConstraints>none</AccessConstraints>
    </Service>
  - <Capability>     = capabilities of service
    - <Request>     = possible requests follow
      + <GetCapabilities></GetCapabilities>     = GetCapabilities request
      - <GetMap>     = GetMap request
          <Format>image/png</Format>     = possible data format of map
          <Format>image/gif</Format>     = possible data format of map
          <Format>image/jpeg</Format>     = possible data format of map
          <Format>image/wbmp</Format>     = possible data format of map
          <Format>image/tiff</Format>     = possible data format of map
        - <DCPType>
          - <HTTP>
            - <Get>
                <OnlineResource xlink:href="http://ows.bom.gov.au/cgi-bin/mapserver/mapserv?map=/web/htdocs/mapserver/radar.map&"/>
              </Get>
            - <Post>
                <OnlineResource xlink:href="http://ows.bom.gov.au/cgi-bin/mapserver/mapserv?map=/web/htdocs/mapserver/radar.map&"/>
              </Post>
            </HTTP>
          </DCPType>
        </GetMap>
      + <GetFeatureInfo></GetFeatureInfo>     = GetFeatureInfo request
      + <DescribeLayer></DescribeLayer>     = other possible request
      + <GetLegendGraphic></GetLegendGraphic>     = other possible request
      </Request>     = end of information concerning the requests
    + <Exception></Exception>
      <VendorSpecificCapabilities/>
      <UserDefinedSymbolization SupportSLD="1" UserLayer="0" UserStyle="1" RemoteWFS="0"/>
    + <Layer></Layer>
    </Capability>     = end of information concerning the capabilities
</WMT_MS_Capabilities>
```

example

Syntax of a WMS request

Get Map request

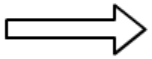
- A WMS request (here: “getMap”) is based on the HTTP common gateway interface (CGI) request, represented by an URL combined with a question mark.

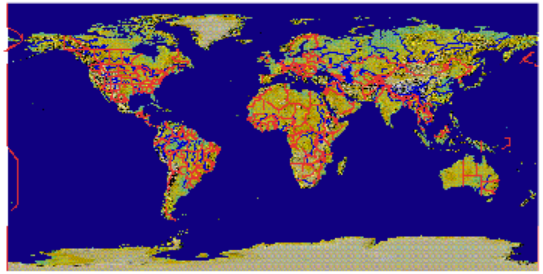
```
http://clearinghouse1.fgdc.gov/scripts/ogc/ms.pl?version=1.1.1&
```

protocol — server of the data provider — application (e.g. in Perl/CGI) — parameter

- After the question mark a list of parameters (key-value pairs) in the format "key = value" follows. If the URL contains several parameters, they are separated by a "&" character.

```
version=1.1.1&
request=getMap&
srs=EPSG:4326&
bBox=-180,-90,180,90&
width=400&height=200&
format=JPEG&
styles=BLACK&
layers=boundary,elevation,lakes,rivers
```





For Explanation of the parameters see next slide!

Mandatory parameters of the GetMap request

previous page next page

VERSION= number of version	protocol versions# (e. g. "version=1.1.1")
REQUEST= getMap	Which request should be retrieved? **Notice**: Some WMS-server only accept "request=Map" instead of "request=getMap"!
LAYERS= layer_list	List of map layers, which should be displayed, seperated by commas (e. g. layers = borders, streets, rivers, lakes)
STYLES= style_list	List of styles (rules for presentation), seperated by commas
SRS= namespace:identifier	Spatial reference system, denotation according to the code list of the EPSG European petroleum survey group (EPSG) >> http://www.epsg.org/ (e. g. Geographical Coordinate System "EPSG=4326")
BBOX= minx,miny,maxx,maxy	Corners of the bounding box (down left, top right) in SRS-units (SRS = Spatial Reference System)
WIDTH= output_width	Width of the map (in pixel)
HEIGHT= output_height	Height of the map (in pixel)
FORMAT= output_format	Definition of the output format (e. g. GIF, JPG, PNG, ...)

Example of a WMS request: GetMap

URL: `http://clearinghouse1.fgdc.gov/scripts/ogc/ms.pl?version=1.1.1&request=map&srs=EPSG:4326&bBox=-180,-90,180,90&width=600&height=300&format=JPEG&layers=boundary,coastline,elevation,lakes,rivers&`

Example:

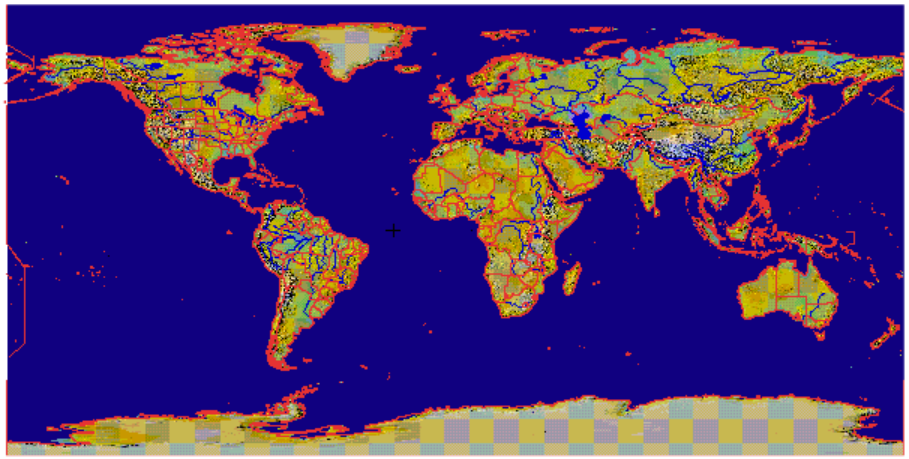
- Map of the World (all layers active)

  multiplying (zoom factor 2)

  hide layer

  zoom & pan

  changing the size of the map



Example of a WMS request: GetMap

URL: `http://clearinghouse1.fgdc.gov/scripts/ogc/ms.pl?version=1.1.1&request=map&srs=EPSG:4326&bBox=-90,-45,90,45&width=300&height=150&format=JPEG&layers=boundary,coastline,elevation,lakes,rivers&`

Example:

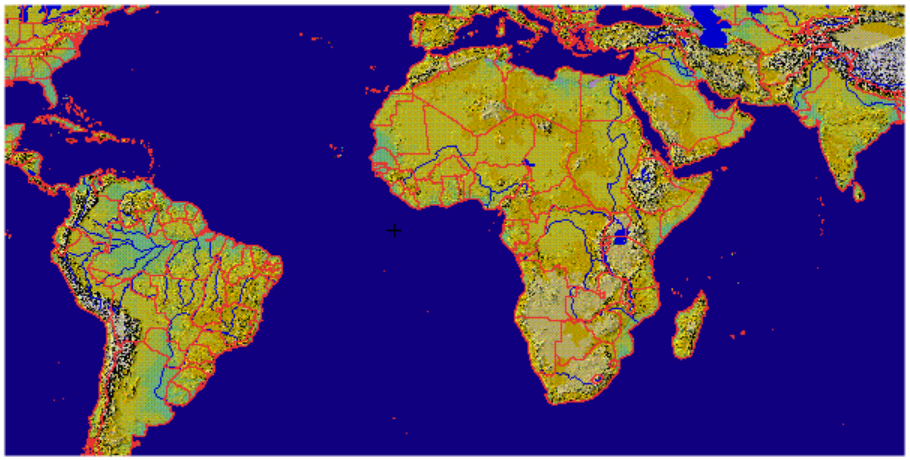
Map of the World (all layers active)

- multiplying (zoom factor 2)

hide layer

zoom & pan

changing the size of the map



Example of a WMS request: GetMap

URL: `http://clearinghouse1.fgdc.gov/scripts/ogc/ms.pl?version=1.1.1&request=map&srs=EPSG:4326&bBox=-90,-45,90,45&width=600&height=300&format=JPEG&layers=boundary,coastline,lakes,rivers&`

Example:

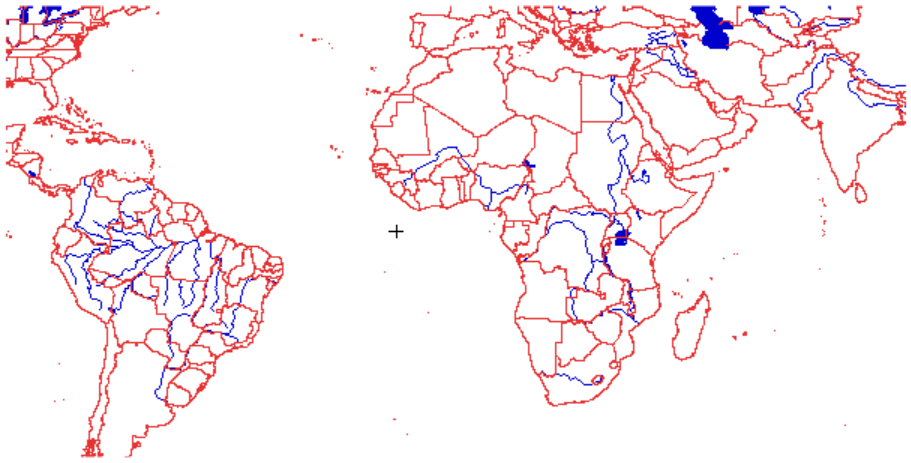
Map of the World (all layers active)

multiplying (zoom factor 2)

- hide layer

zoom & pan

changing the size of the map



Example of a WMS request: GetMap

URL: `http://clearinghouse1.fgdc.gov/scripts/ogc/ms.pl?version=1.1.1&request=map&srs=EPSG:4326&bBox=0,45,12,60&width=600&height=300&format=JPEG&layers=boundary,coastline,elevation,lakes,rivers&`

Example:

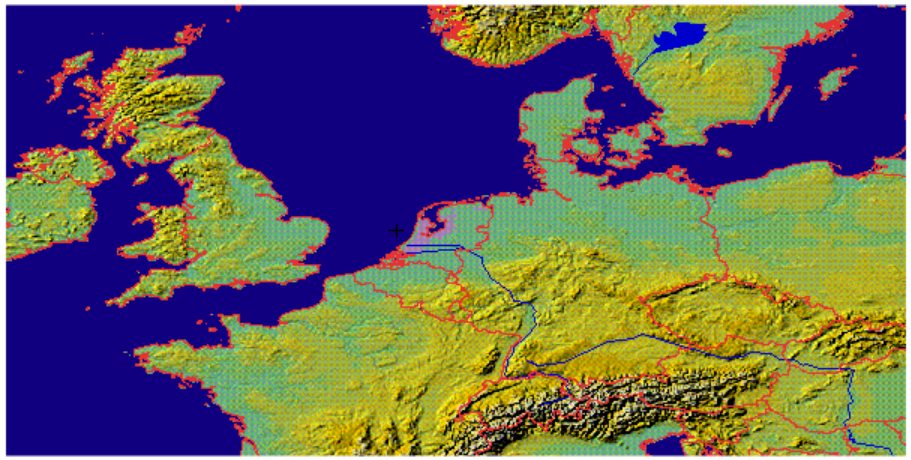
Map of the World (all layers active)

multiplying (zoom factor 2)

hide layer

- zoom & pan

changing the size of the map



Example of a WMS request: GetMap

URL: `http://clearinghouse1.fgdc.gov/scripts/ogc/ms.pl?version=1.1.1&request=map&srs=EPSG:4326&bBox=-90,-45,90,45&width=300&height=150&format=JPEG&layers=boundary,coastline,elevation,lakes,rivers&`

Example:

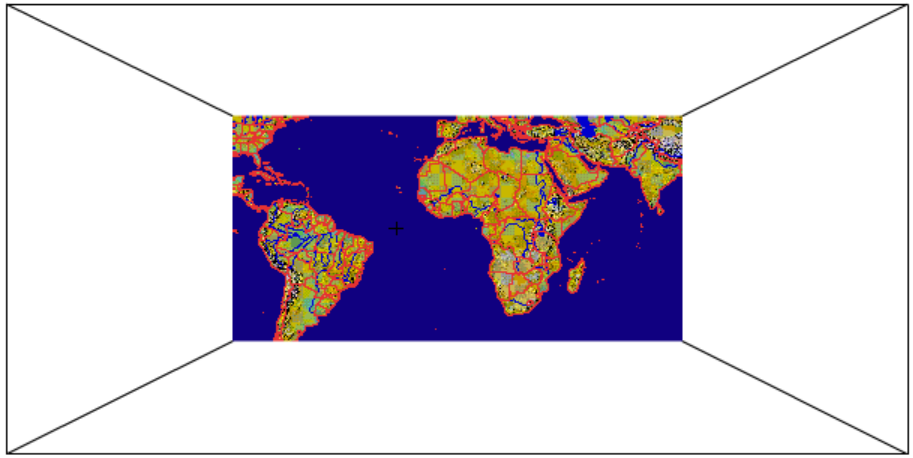
Map of the World (all layers active)

multiplying (zoom factor 2)

hide layer

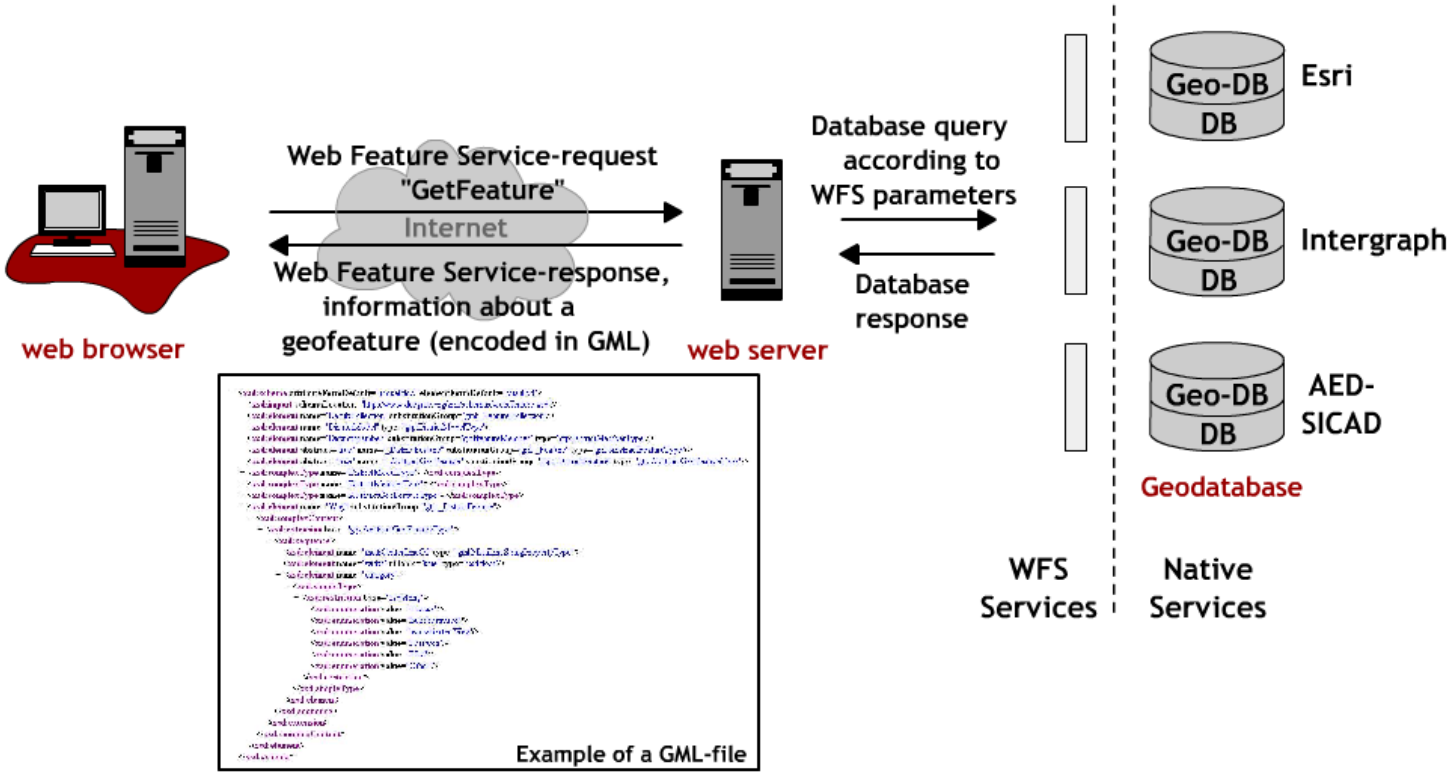
zoom & pan

- changing the size of the map



WFS

previous page next page



OGC Web Feature Service (WFS):

- OGC WFS abstract specification 1.1.0 became ISO standard on May, the 3rd of 2005.
- A WFS provides access to geofeatures via standardized interfaces.
- Features consist of geometries (vector data) and attributes. (see course 5)
- The WFS delivers features encoded in GML (Geography Markup Language). (see course 5)
- The WFS is based on standardized HTTP requests.
- 6 WFS-requests are defined (3 mandatory, 3 optional; see next slides)

Mandatory WFS requests

The 3 mandatory WFS requests:

- GetCapabilities

  Provides metadata for the description of the implemented requests on available feature classes and so on.

- DescribeFeatureType

  Provides a description of the structure of single feature classes, based on XML schema.

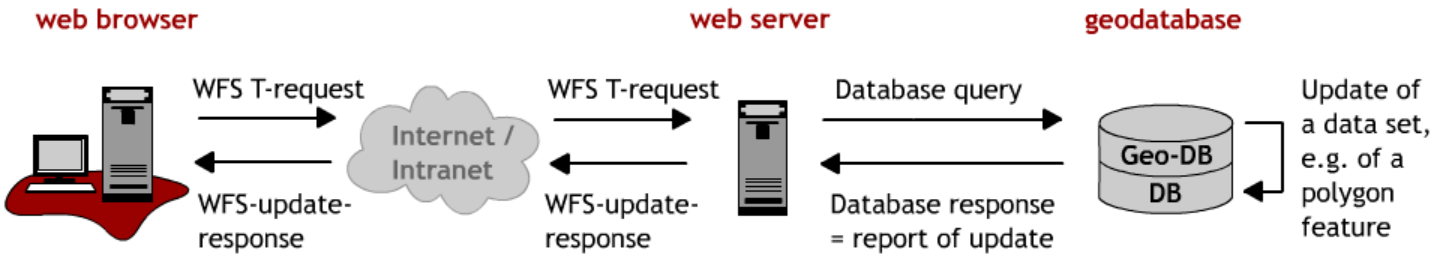
- GetFeature

  Delivers one or more features according to the request.

Update of geodata via WFS-T service

The WFS-Transaction (WFS-T) service is intended to support database updates by offering the following operators:

INSERT - insert a feature into the DB
DELETE - delete a feature in the DB
UPDATE - update attributes of a feature



The 3 optional WFS requests (according to the transaction mode):

- Transaction

  Request to manipulate features. This request uses the database operations INSERT, DELETE and / or UPDATE.

- LockFeature

  Locking of one or more features for the duration of an update process. This request ensures the consistency of the database, because a locked feature can only be processed by one user.

- GetFeatureWithLock

  Almost the same like the GetFeature request, but locks the selected features additionally.

Parameters of GetFeature request

VERSION = number of version	Protocol version# (e. g. "version=1.1.0")
SERVICE = wfs	Name of the service, which should be used
REQUEST = getFeature	Name of the request, which should be invoked
PROPERTYNAME = properties_list	List of the attribute names, which should be queried. No indication of the parameter or input of the value "*" means, all attributes are queried.
FEATUREVERSION = ALL "or" n	Version of the features, which should be retrieved. ALL means all versions. No indication of a parameter means the last version.
MAXFEATURES = n	Maximum number of features. No indication of a parameter means all features.
OUTPUTFORMAT = text/xml; subtype= gml/3.1.1	Output format to use for response
TYPENAME = country_borders	List of feature type names to query
RESULTTYPE = results	A web feature service may respond to a GetFeature request in one of two ways. It can either generate a complete response document (RESULTTYPE = results) or it may simply return a count of the number of features that a GetFeature request would return (RESULTTYPE = hits).

The WFS provides GML (Geography Markup Language) - encoded features as result of the request.

Example of a GML - encoded feature:

```
<Road>                                                    begin of feature
  <gml:name>M11</gml:name>
  <linearGeometry>
    <gml:LineString srsName="http://www.opengis.net/gml/srs/epsg.xml#4326">
      <gml:coord>
        <gml:X>0</gml:X>                                  geometry type
        <gml:Y>5.0</gml:Y>
      </gml:coord>
      .....
      <gml:coord>
        <gml:X>80.5</gml:X>
        <gml:Y>60.9</gml:Y>
      </gml:coord>
    </gml:LineString>                                     geometry of feature
  </linearGeometry>
  <classification>motorway</classification>
  <number>11</number>                                     thematic properties
</Road>                                                   end of feature
```

Possible applications of a WFS

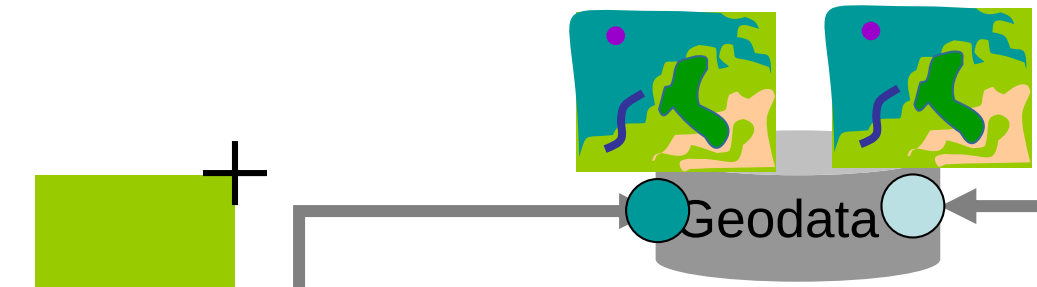
A WFS enables:

- The distribution and access of feature data via the Internet.
- The updating of features in a geodatabase.
- The insertion of newly collected features in the geodatabase.
- The deletion of features in a geodatabase.

WMS + WFS

WebMapService

WebFeatureService



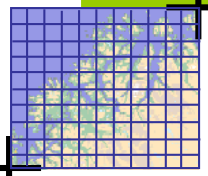
Query (-> data)

ersion to GML

Image generation

Query (->Data)

ureServer

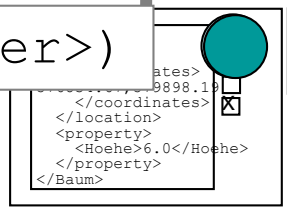


```
<Baum>
  <location>
    <coordinates>
    576834.67,579898.19
    </coordinates>
  </location>
  <property>
    <Hoehe>6.0</Hoehe>
  </property>
</Baum>
```

WebServer

GetFeature(< parameter>)

GetMap(layer,bbox,CRS,…)



WebClient

Scenario Thunderstorm "Lothar" in South-West Germany:

Due to the thunderstorm "Lothar" about 2 Million cubic meter of wood were overthrown in the year 1999 in South-West Germany (state of Baden-Wuerttemberg). Mostly the older tree populations were affected.

Overthrown forest area in Baden-Wuerttemberg, source: Lars Bernard, lecture at the 6. seminar "GIS in the Internet/Intranet", UniBw Munich

For a fast and efficient disaster management the super-regional traffic routes, which may be blocked by fallen trunks, have to be identified.

So the following information may be required:

Show all super-regional road sections, which are located in forest areas, whose age (of the trees) is older than 80 years.

WMS/WFS application scenario - workflow

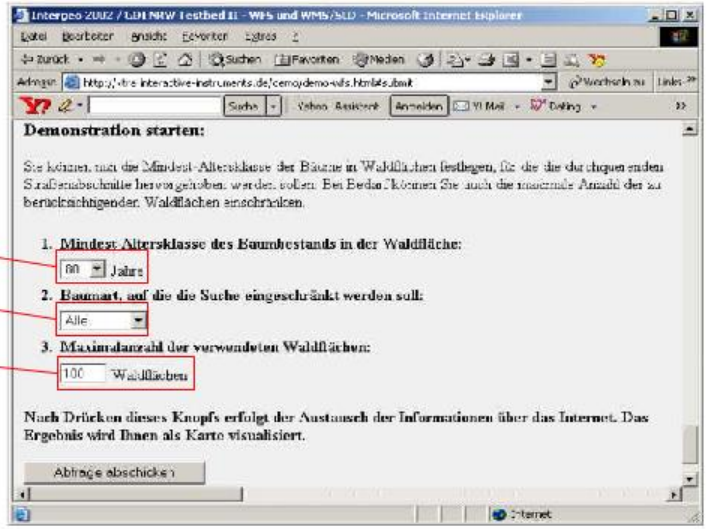
WMF/ WFS example:

Geodata infrastructure in South-West Germany

client

application service Bonn

>Launch of the client belonging to the integrated application service:
http://xtra.interactive-instruments.de/demo/demo-wfs.html#submit

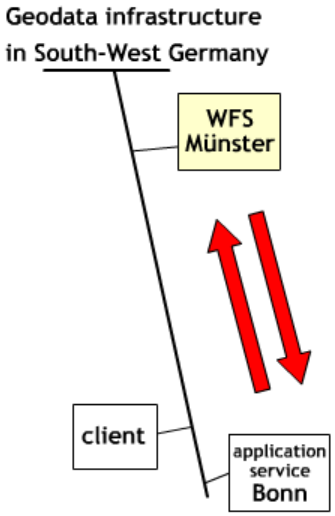


> 80 years

all types of trees

max. 100 forest areas

WMF/ WFS example:

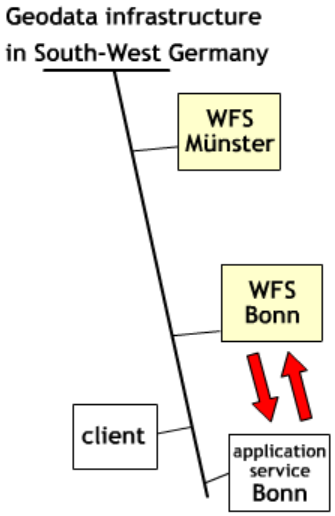


The application service composes three requests:

1. Request to a WFS of the Forest Management in Muenster:

Identify all forest areas (of all kind of trees) with a tree population of an age > 80 years!

WMF/ WFS example:

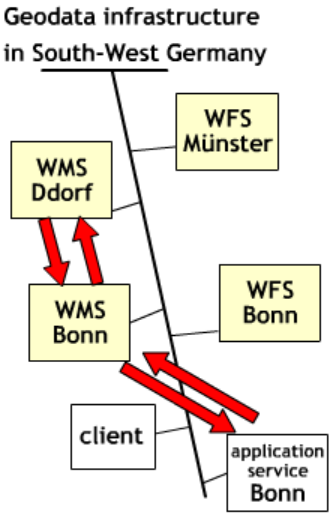


The application service composes three requests:

2. Second request to the WFS of the interactive Road information system in Bonn, using the result of the previous request:

Identify all road sections, which cross the affected forest areas!

WMF/ WFS example:



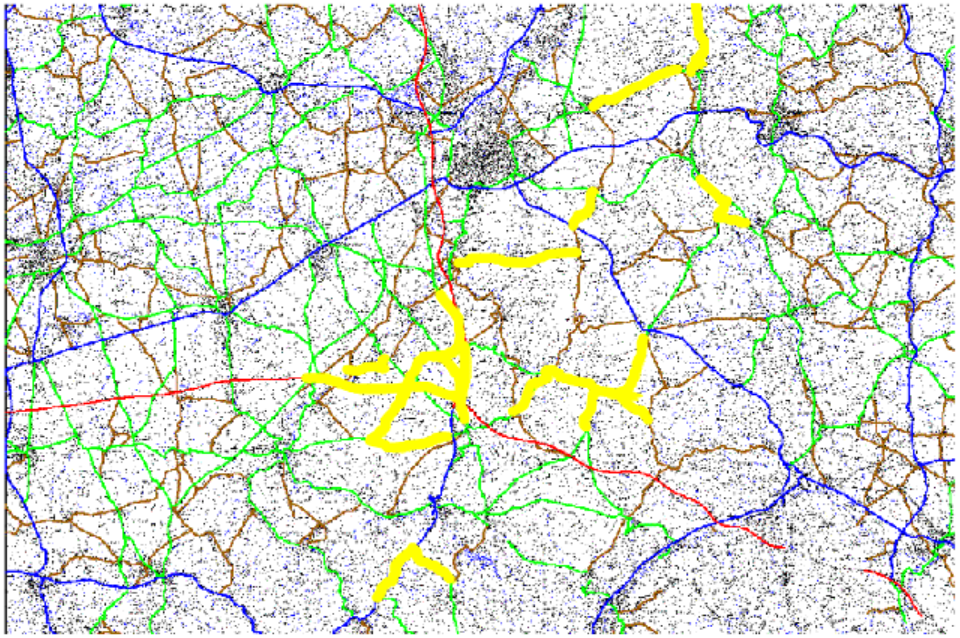
The application service composes three requests:

3. Third request to the WMS of the interactive road information system in Bonn, using the result of the previous requests:

draw map:
- with a topographic map (scale: 1:50000) of the land surveying office in Duesseldorf (WMS in Duesseldorf) as background
- with all roads
- and all affected roads in yellow

WMS/WFS application scenario - result

The result:



This map shows the roads, which can be affected by the storm.

Legend:

- all affected roads
- different types of super-regional roads

Background: Topographic map, scale 1:50 000

Conclusion

- WMS, WFS and other Geo Web Services are widely used
- Geo Web Services are most important within NSDI
- Geo Web Services are one common way to use GI functionality

OGC (2006): Web Map Service Implementation Specification.
OGC 06-042, OpenGIS® Consortium Inc.,
>> http://www.opengeospatial.org/standards/wms

OGC (2005): Web Feature Service Implementation Specification.
OGC 04-094, OpenGIS® Consortium Inc.,
>> http://www.opengeospatial.org/standards/wfs

OGC (2005): Catalog Service Implementation Specification.
OGC 04-021r3, OpenGIS® Consortium Inc.,
>> http://www.opengeospatial.org/standards/cat

OGC (2003): Web Coverage Service Specification.
OGC 03-065r6, OpenGIS® Consortium Inc.,
>> http://www.opengeospatial.org/standards/wcs

Thank you!

Questions?

Wolfgang.Reinhardt@unibw.de