FINNISH METEOROLOGICAL INSTITUTE

Ubicasting: Delivering Weather Data Using OGC Web Services

Ubicasting Workshop 10.9.2008

Ilkka Rinne, Systems Architect, Development of Services, FMI

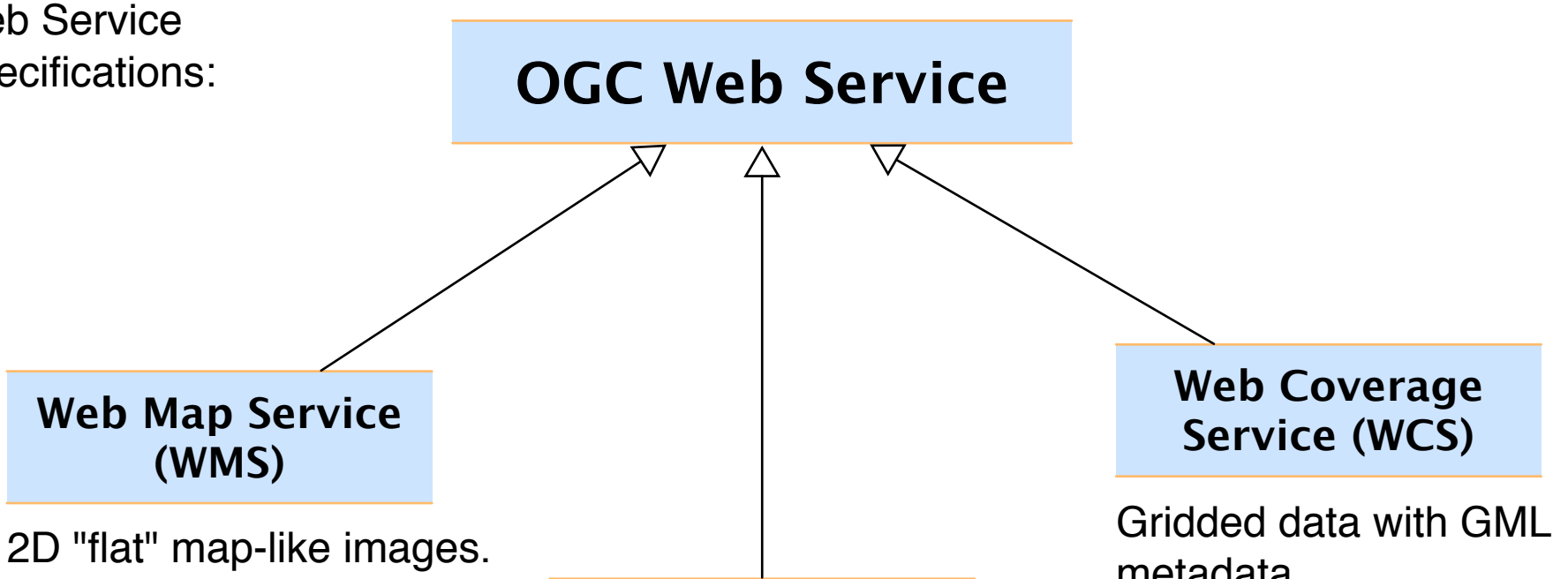
Presentation outline

- What is Open Geospatial Consortium (OGC)?
- An Overview of Some OGC Web Service Standards
- FMI & Open Standard Interfaces: What Do We Want?
- Weather Data Delivery in the Ubicasting Project Using OGC Web Services
- Ubicasting Architecture Status at FMI (09/2008)
- Questions

Open Geospatial Consortium (OGC)

- An international industry consortium of 367 companies, government agencies and universities (09/2008) participating in a consensus process to develop publicly available interface specifications.
- A global forum for the users and the developers of the geospatial information systems (GIS). Aims at interoperability of different GIS systems over the World Wide Web using Open Standards.
- Many essential OGC standards are also pushed through the ISO standardization process.
- An important player in the EU INSPIRE directive implementation.
- Members from Finland: Finnish Geodetic Institute and Vaisala.

Some essential OGC Web Service Specifications:



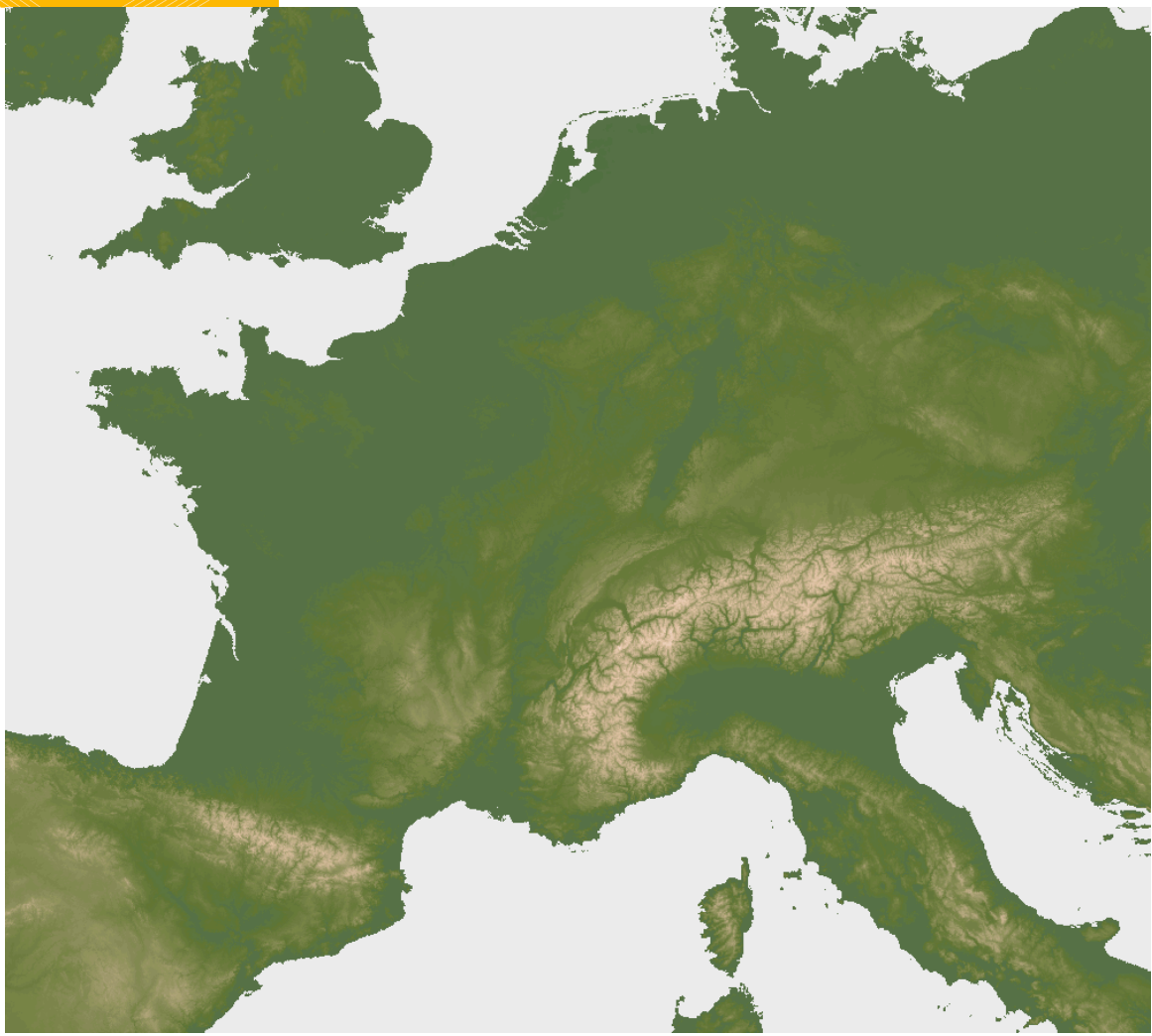
Web Map Service (WMS)

Map-like views of pre-rendered information: HTTP queries to a WMS server return pre-rendered images (layers) from different sources to be layered on top of each other in the client application.

Available layers, projections, and other parameters can be queried with GetCapabilities-request.

A lot of WMS-capable open source and commercial client software and libraries already exist: rapid implementation of end-user applications is expected as soon as interesting data is made available.

Google Earth supports WMS directly



Map
Roads
Satellite
NWM Precipitation
Radar Precipitation

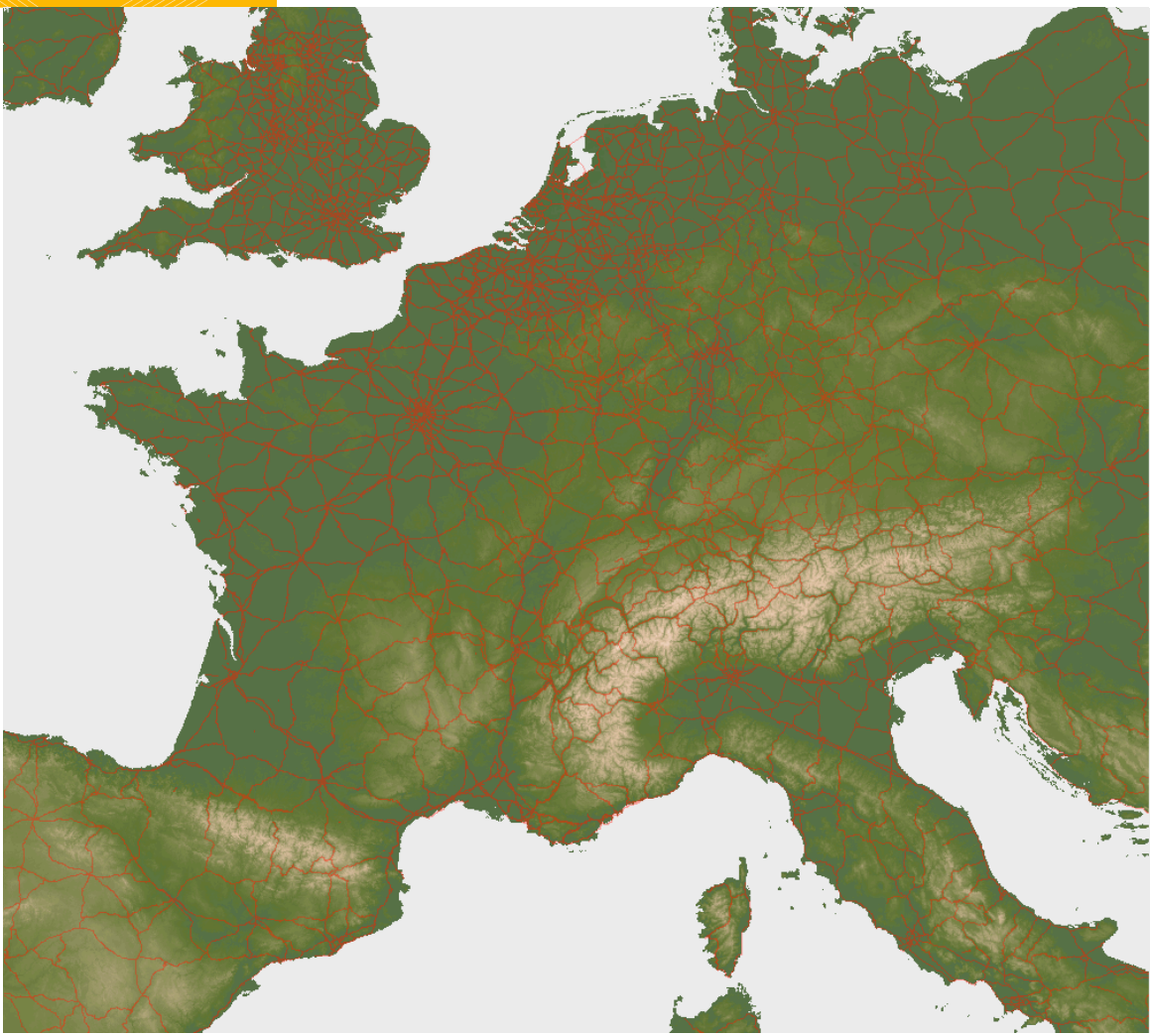
Web Map Service (WMS)

Map-like views of pre-rendered information: HTTP queries to a WMS server return pre-rendered images (layers) from different sources to be layered on top of each other in the client application.

Available layers, projections, and other parameters can be queried with GetCapabilities-request.

A lot of WMS-capable open source and commercial client software and libraries already exist: rapid implementation of end-user applications is expected as soon as interesting data is made available.

Google Earth supports WMS directly



Map
Roads
Satellite
NWM Precipitation
Radar Precipitation

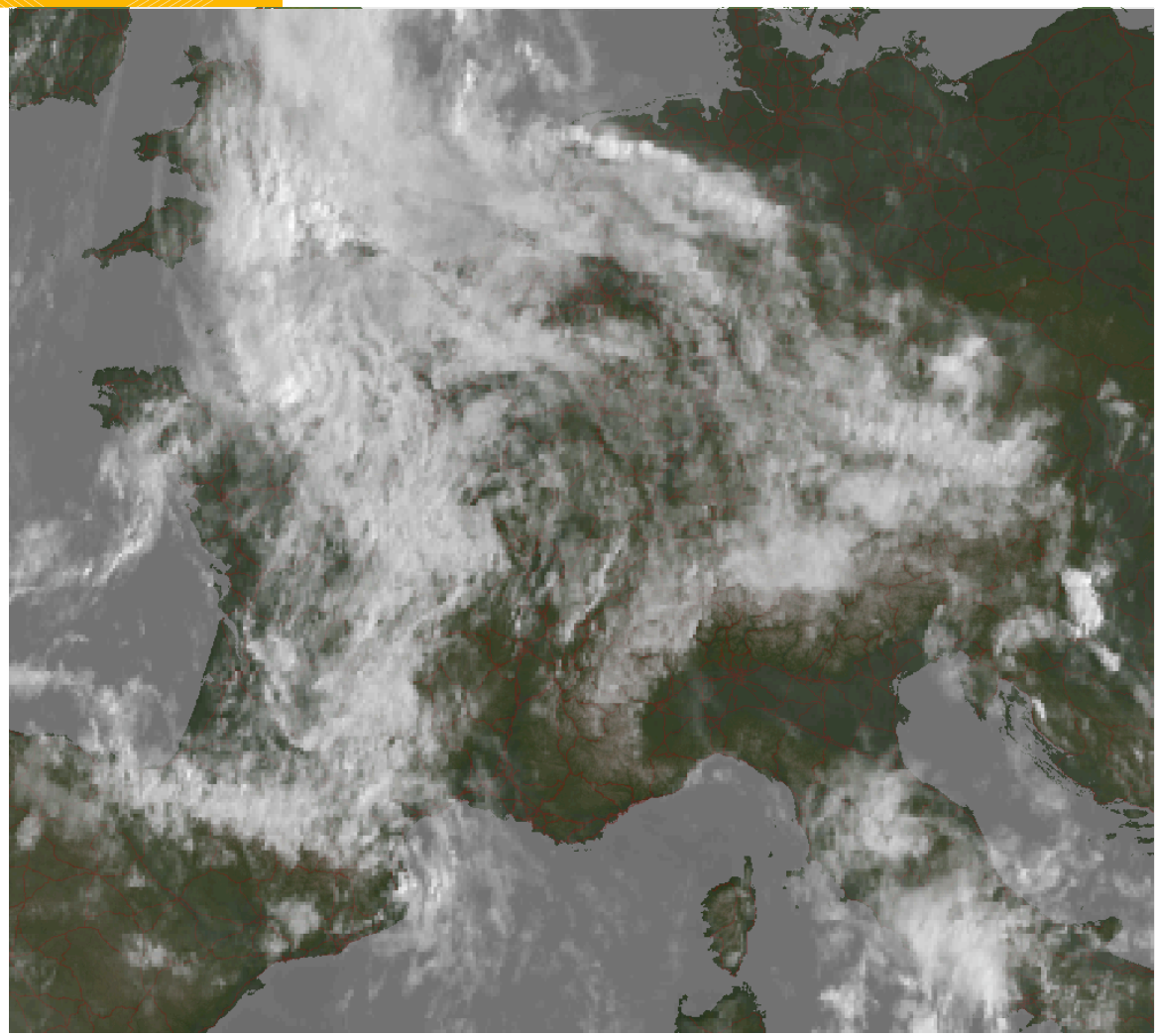
Web Map Service (WMS)

Map-like views of pre-rendered information: HTTP queries to a WMS server return pre-rendered images (layers) from different sources to be layered on top of each other in the client application.

Available layers, projections, and other parameters can be queried with GetCapabilities-request.

A lot of WMS-capable open source and commercial client software and libraries already exist: rapid implementation of end-user applications is expected as soon as interesting data is made available.

Google Earth supports WMS directly



Map
Roads
Satellite
NWM Precipitation
Radar Precipitation

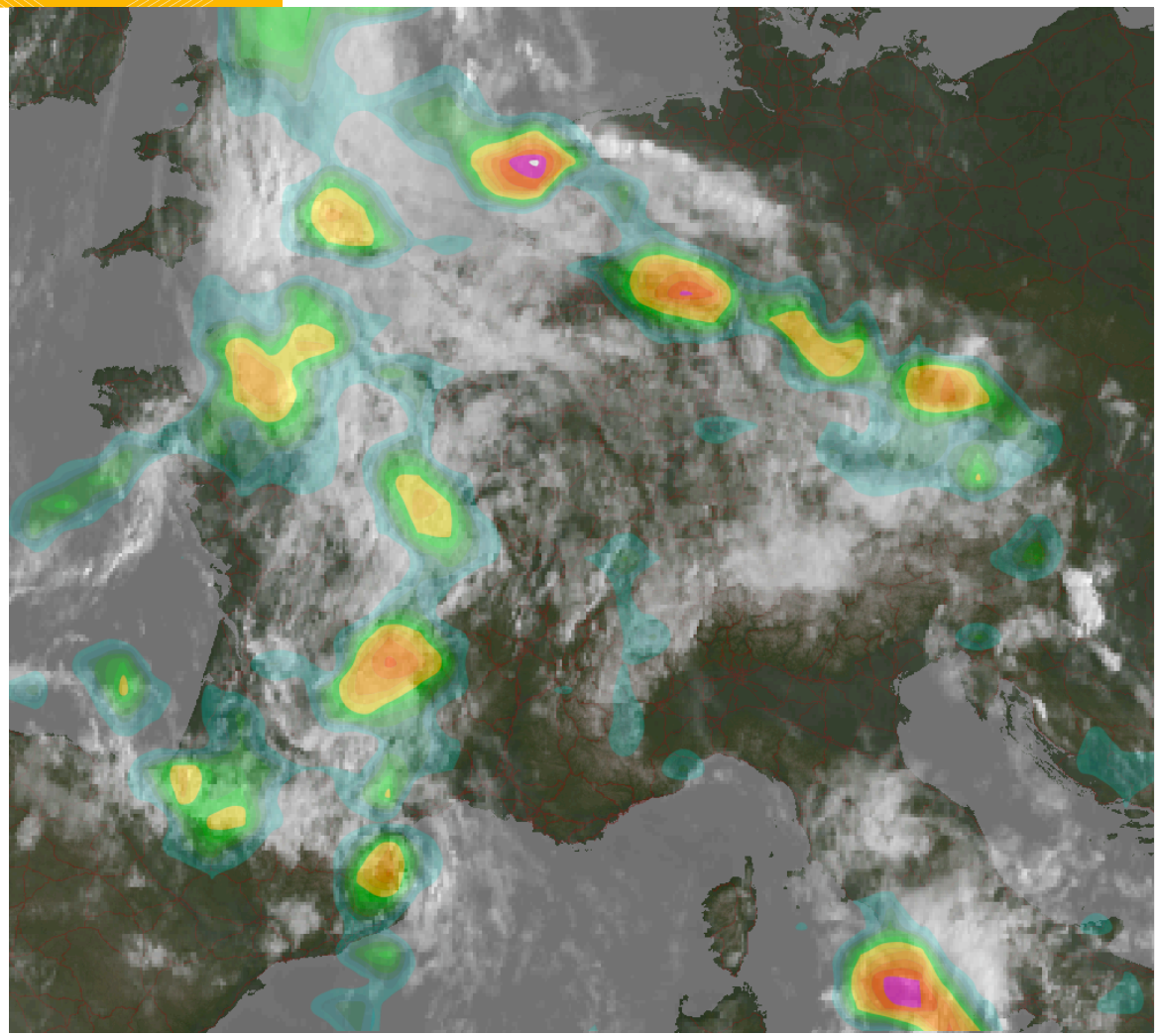
Web Map Service (WMS)

Map-like views of pre-rendered information: HTTP queries to a WMS server return pre-rendered images (layers) from different sources to be layered on top of each other in the client application.

Available layers, projections, and other parameters can be queried with GetCapabilities-request.

A lot of WMS-capable open source and commercial client software and libraries already exist: rapid implementation of end-user applications is expected as soon as interesting data is made available.

Google Earth supports WMS directly



Map
Roads
Satellite
NWM Precipitation
Radar Precipitation

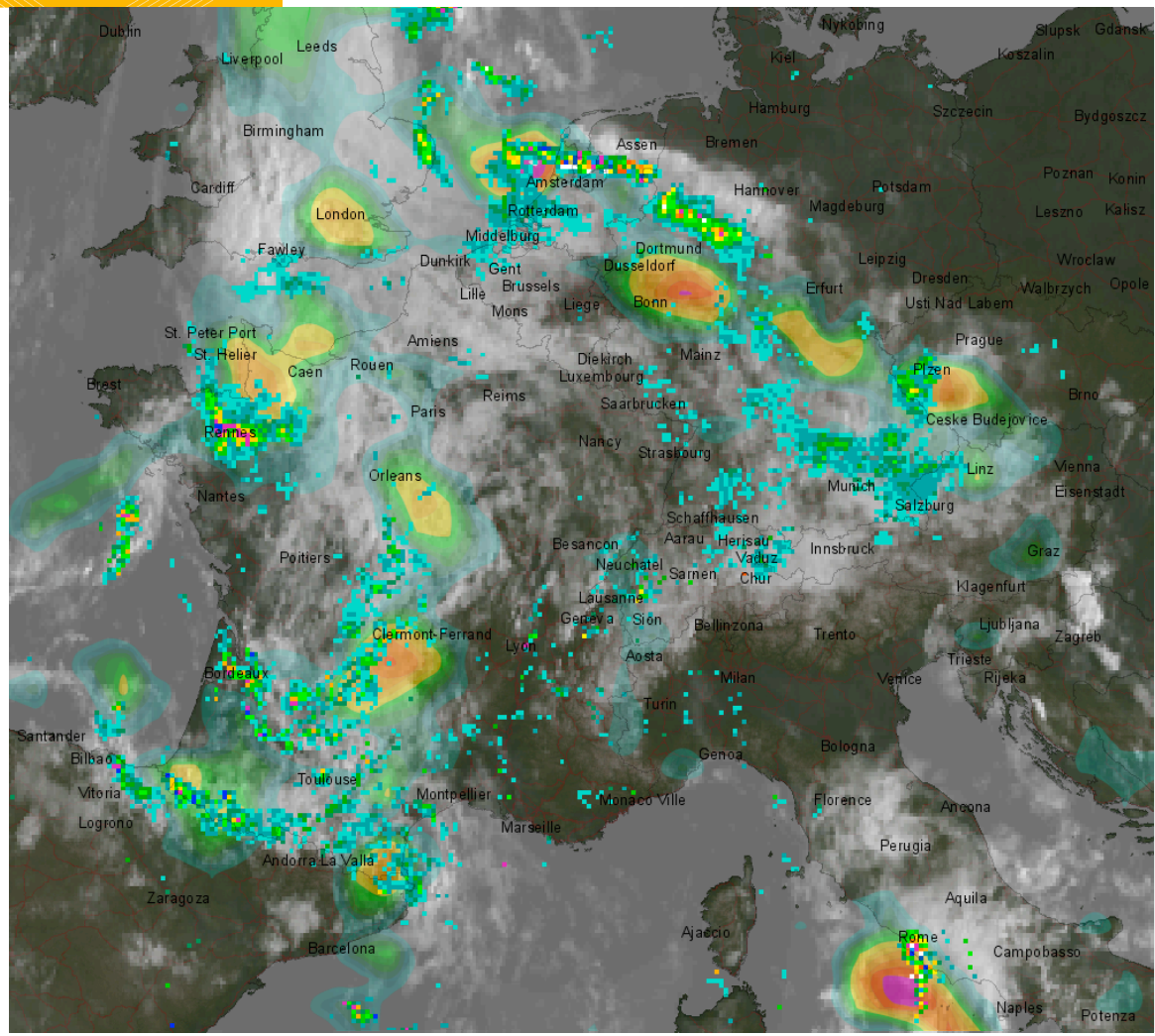
Web Map Service (WMS)

Map-like views of pre-rendered information: HTTP queries to a WMS server return pre-rendered images (layers) from different sources to be layered on top of each other in the client application.

Available layers, projections, and other parameters can be queried with GetCapabilities-request.

A lot of WMS-capable open source and commercial client software and libraries already exist: rapid implementation of end-user applications is expected as soon as interesting data is made available.

Google Earth supports WMS directly



Map
Roads
Satellite
NWM Precipitation
Radar Precipitation

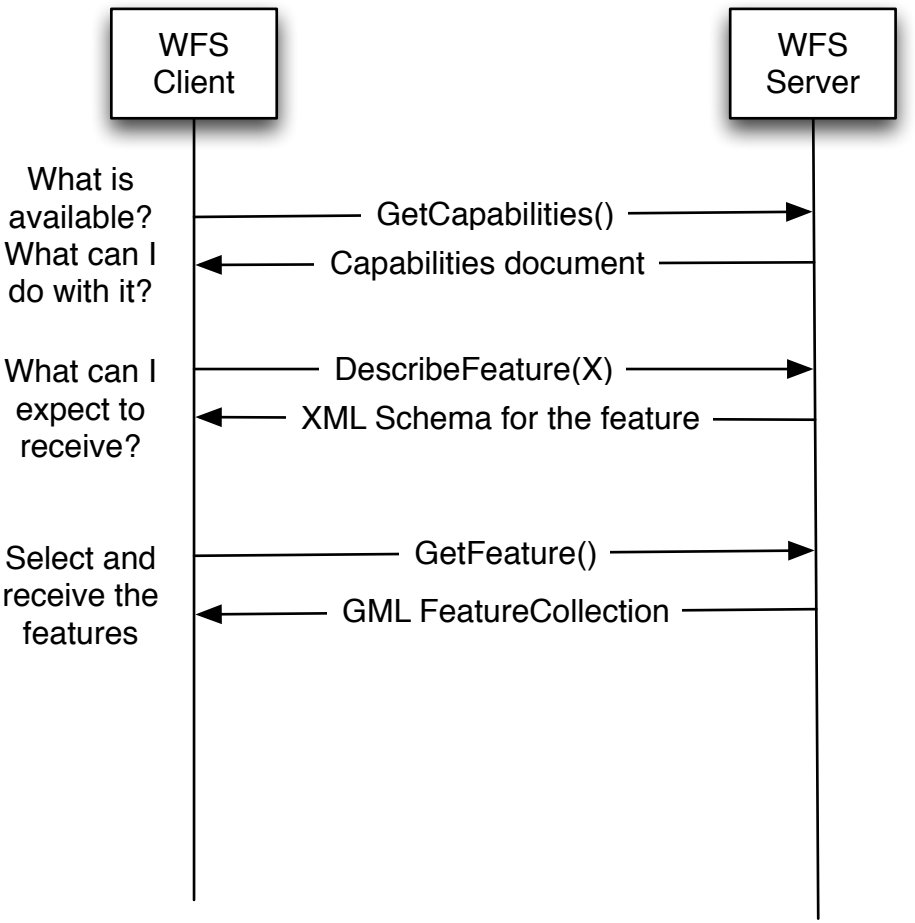
Web Feature Service (WFS)

Web Feature Service defines protocols for requesting as well as (optionally, WFS-T) inserting and modifying GML Features (Geospatial objects) through HTTP requests.

GML Features are geospatial objects semantically describing real-world entities, like Meteorological objects (fronts, precipitation areas), weather observation readings, observation stations or roads.

WFS can be used as a client-server data retrieval, server-to-server data synchronization or as a common access interface on top of heterogenous geospatial data stores.

Typical call sequence in WFS client requesting a GML Feature (or set of Features):



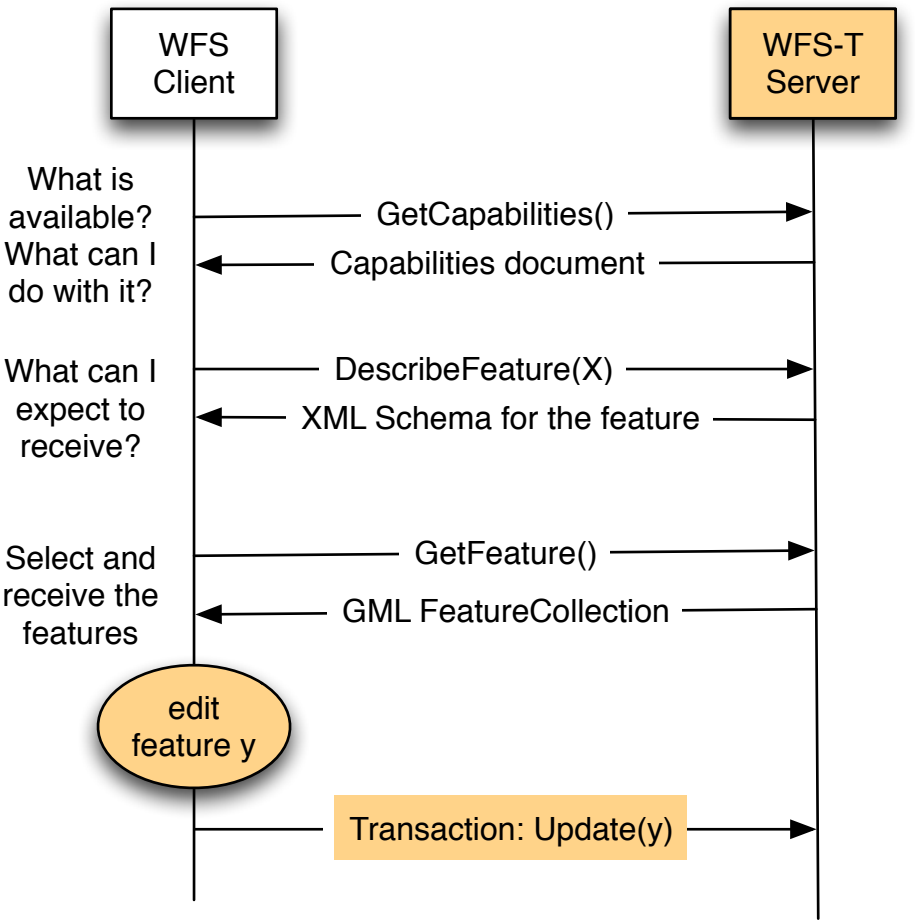
Web Feature Service (WFS)

Web Feature Service defines protocols for requesting as well as (optionally, WFS-T) inserting and modifying GML Features (Geospatial objects) through HTTP requests.

GML Features are geospatial objects semantically describing real-world entities, like Meteorological objects (fronts, precipitation areas), weather observation readings, observation stations or roads.

WFS can be used as a client-server data retrieval, server-to-server data synchronization or as a common access interface on top of heterogenous geospatial data stores.

Optional Transaction requests (WFS-T) enable inserting, updating, deleting and locking GML Features:



Web Coverage Service (WCS)

Similar to WMS and WFS, but instead of features (=objects) deals with coverages: mappings from grid points into parameter (temperature, pressure,..) values = grid data.

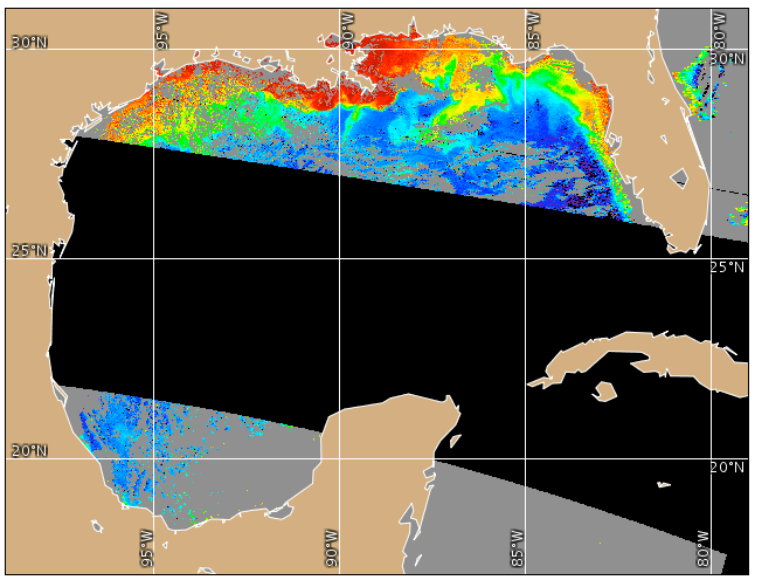
Analogically to WFS, the client requests the WCS server for available data (GetCapabilities), asks it to describe the interesting coverages (DescribeCoverage) and create and return the requested subset of the available coverage data (GetCoverage). Optionally only the HTTP URLs, not the actual data sets can be returned to the client.

Only data is returned only for the required area and the desired parameters.

Request: GetCoverage

```
<?xml version="1.0" encoding="UTF-8"?>
<GetCoverage xmlns="http://www.opengis.net/wcs/1.1"
xmlns:ows="http://www.opengis.net/ows/1.1"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/wcs/1.1 ../wcsGetCoverage.xsd"
service="WCS" version="1.1.2">
   <ows:Identifier>Cov123</ows:Identifier>
   <DomainSubset>
      <ows:BoundingBox crs="urn:ogc:def:crs:OGC:2:84">
         <ows:LowerCorner>-71 47</ows:LowerCorner>
         <ows:UpperCorner>-66 51</ows:UpperCorner>
      </ows:BoundingBox>
   </DomainSubset>
   <Output format="image/netcdf"/>
</GetCoverage>
```

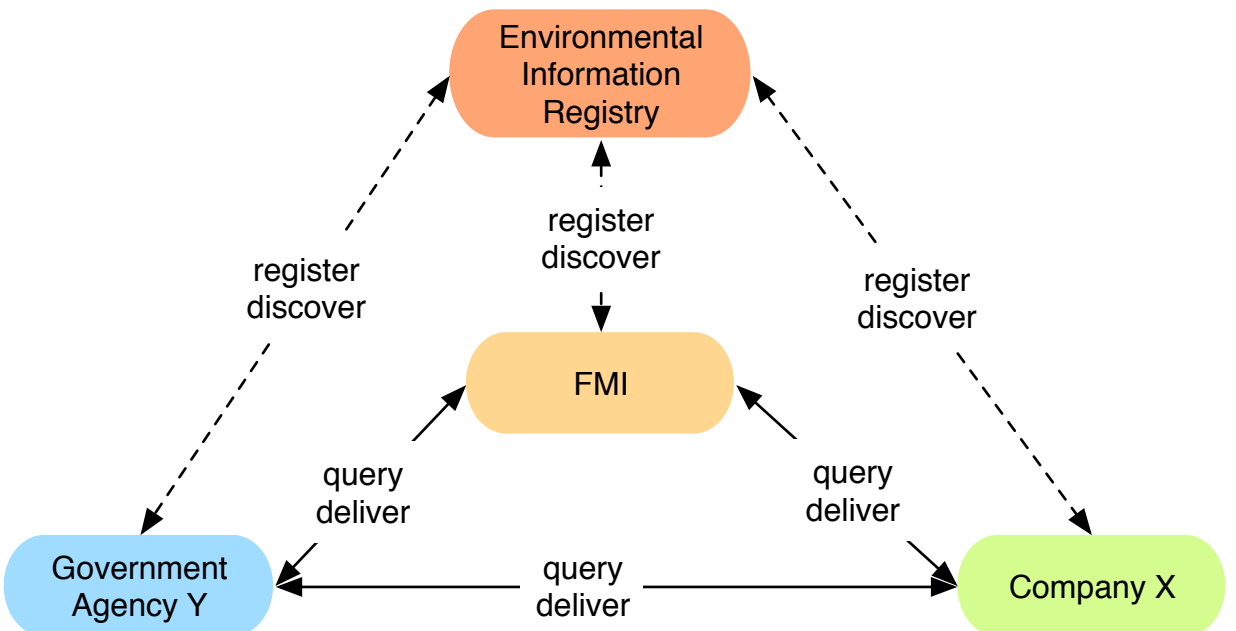
Response: a data grid



FMI Long-term Goal: Environmental Data Interchange Using Open Standard Web Service Interfaces

Need for fast and accurate information exchange of environmental observation and prediction data is increasing all the time.

But building different data delivery solutions for each customer cost a lot for all parties and leads into an maintenance nightmare.



Open Standard Interfaces: only **data availability** needs to be negotiated, not protocols, formats.

All data need not (and cannot) be freely available for everyone: delivery costs, information security issues, trade secrets etc.

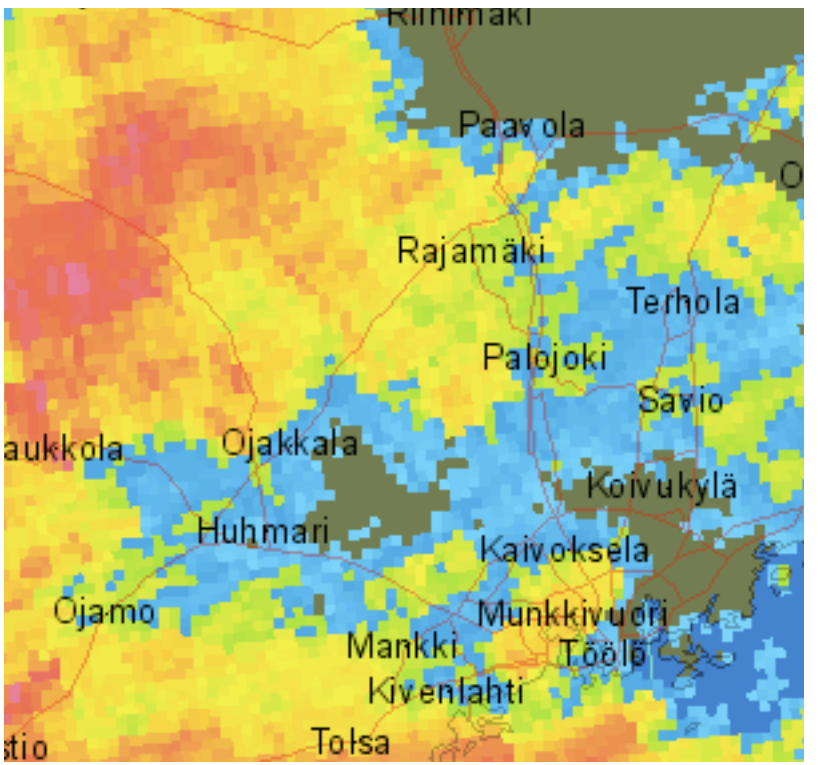
Services made available are registered in a **common registry**.

Still a lot of standardization to do: definitions, GML vocabulary etc.

Challenges: Highly Dynamic and Multidimensional Data

Traditional GIS data is relatively stabile: geography, roads, buildings etc.

But environmental GIS data is changing constantly: weather radar data received 15 minutes ago is already old.



9th September 2008 at 07:00z

Everything is temporal:

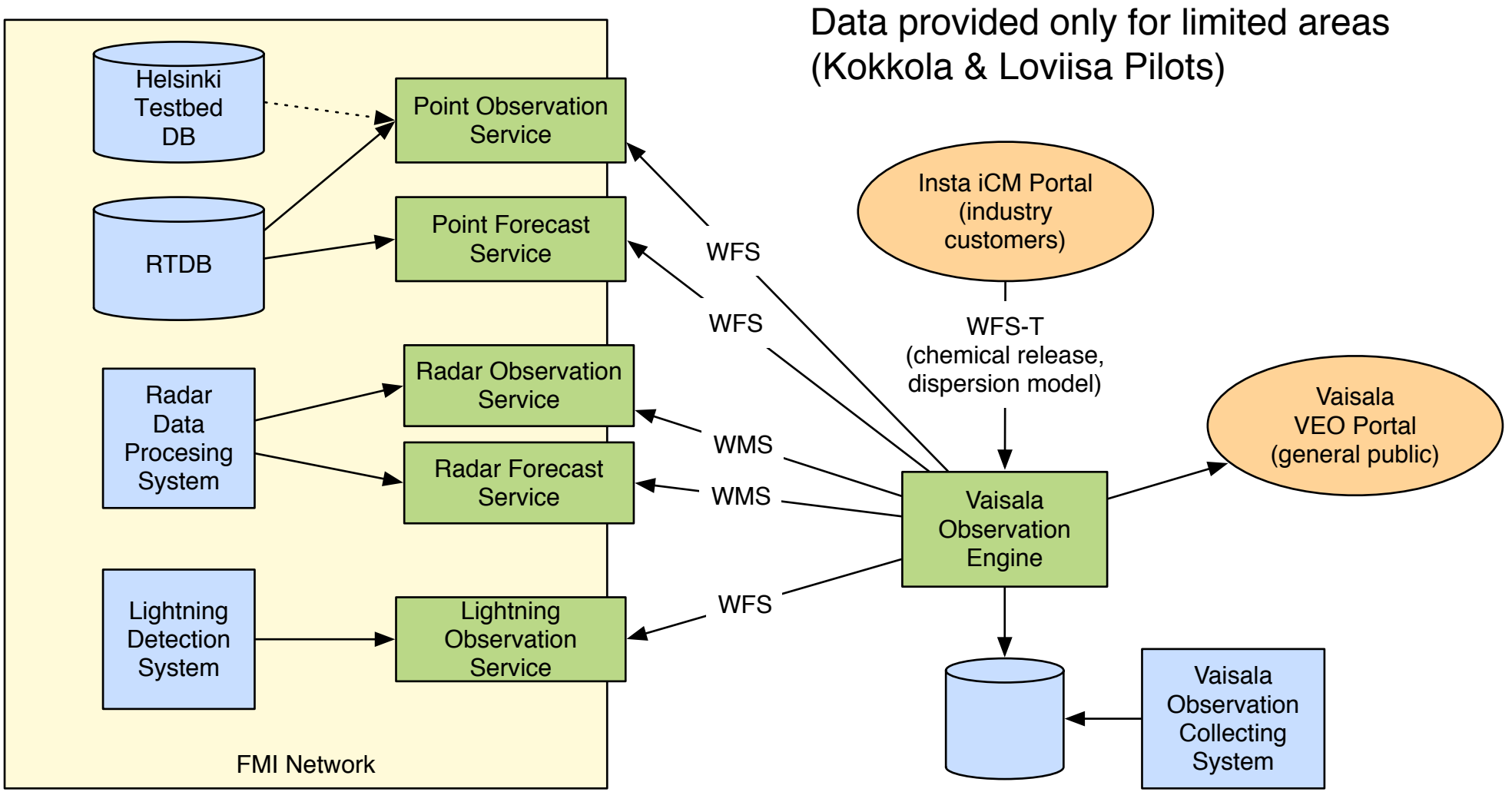
- when was is created?
- what instance of time does it describe or predict?

Handling highly dynamic data is a relative new thing in the GIS community.

Not all OGC Standards are designed with constantly changing data in mind, the software implementations even less so.

WMS does not understand data cross-sections & 3D (+time) data.

OGC Web Services In the Ubicasting Project



Current Status at FMI (09/2008)

- Observation, Forecast and Lightning Services (WFS) is running.
- Radar Observation & Forecast Services (WMS) is running.
- UMN MapServer is the currently used implementation for WMS/WFS.
- Connecting the observation databases directly to WFS servers with Geoserver is under investigation.
- Possibility to use more complex GML schemas for weather observation & forecast data is under investigation.
- Delivering Hirlam, LAPS and radar "raw" data over WCS is under investigation: the "exotic" weather data formats cause some head-scratching.
- The need for a Catalog Service (Registry) becomes more and more evident as the amount of services increase ("what do we have available and where?").