Serving Coverage Data in FMI Open Data Portal
Finnish Meteorological Institute

Finnish Meteorological Institute
Roope Tervo, Mikko Visa
FMI Open Data

Finnish Meteorological Institute opened its data in 2013. Basically everything that FMI has property rights was opened. Data is provided in freely in machine readable format.

https://en.ilmatieteenlaitos.fi/open-data
FMI Open Data Portal follows INSPIRE requirements.

The very same data portal works as Open Data and INSPIRE portal.
# Example of Data Sets

<table>
<thead>
<tr>
<th>Data set</th>
<th>Description</th>
<th>Time Interval</th>
<th>Estimated publish date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weather Observations</td>
<td>Temperature, Wind, Humidity, Ground Temperature…</td>
<td>10 min</td>
<td>Open, older data to be added</td>
</tr>
<tr>
<td>Sun Radiation</td>
<td>UV, Short and Long Term Radiation…</td>
<td>1 min</td>
<td>Open</td>
</tr>
<tr>
<td>Marine Observations</td>
<td>Waves, Sea Temperature, Sea Level…</td>
<td>1 h</td>
<td>Open</td>
</tr>
<tr>
<td><strong>Weather Radars</strong></td>
<td><strong>Precipitation Rate, Precipitation Amount…</strong></td>
<td><strong>5 min</strong></td>
<td>Open, older data to be added</td>
</tr>
<tr>
<td>Lightning</td>
<td>Thunder Strikes in Finland</td>
<td>5 min</td>
<td>Open</td>
</tr>
</tbody>
</table>

3.12.2015
# Example of Data Sets

<table>
<thead>
<tr>
<th>Data set</th>
<th>Description</th>
<th>Time Interval</th>
<th>Estimated publish date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Real Time Observations</td>
<td>Real Time Observations from specific location(s)</td>
<td>AWS 2010 – Soundings 1959 – Flashes 1998 – Sea Level 1971 – Waves 2005 –</td>
<td>Open <em>older data will be added</em></td>
</tr>
<tr>
<td>Climatological Observations</td>
<td>Dayly and monthly temperature mean and extreme values from weather stations</td>
<td>1959 -</td>
<td>Open</td>
</tr>
<tr>
<td>Climatological Observations</td>
<td>Monthly temperature and precipitation rate mean values interpolated to grid</td>
<td>1961 -</td>
<td>Open</td>
</tr>
</tbody>
</table>
## Example of Data Sets

<table>
<thead>
<tr>
<th>Data set</th>
<th>Description</th>
<th>Time Interval</th>
<th>Estimated publish date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weather forecast model HIRLAM RCR</td>
<td>Point forecasts and grid data</td>
<td>Latest model run (4 times a day)</td>
<td>Open</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0…54 h</td>
<td></td>
</tr>
<tr>
<td>Sea forecast models</td>
<td>Sea level point forecasts, Wave (WAM) and current (HBM) as grid data</td>
<td>Latest model run (4 times a day)</td>
<td>Open</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0…54 h</td>
<td></td>
</tr>
<tr>
<td>Environmental Monitoring Facilities</td>
<td>Weather observation stations, radars…</td>
<td></td>
<td>2015</td>
</tr>
<tr>
<td>Aviation Observations</td>
<td>METAR</td>
<td>30 min</td>
<td>open</td>
</tr>
<tr>
<td>Ground &amp; mast observations</td>
<td>Special observations from ground and masts</td>
<td></td>
<td>2016 /Open</td>
</tr>
</tbody>
</table>
# Example of Data Sets

<table>
<thead>
<tr>
<th>Data set</th>
<th>Description</th>
<th>Time Interval</th>
<th>Estimated publish date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Air Quality Observations</td>
<td>Air Quality Observations</td>
<td>1h</td>
<td>2015-2016</td>
</tr>
<tr>
<td>Silam Model</td>
<td>Dispersion Model for Air Quality, Forest Fire and Pollen</td>
<td>Latest model run (once a day) 0…96h</td>
<td>2015</td>
</tr>
<tr>
<td>HELMI Ice Model</td>
<td>Ice forecast model</td>
<td>Latest model run (4 times a day) 0…54 h</td>
<td>open</td>
</tr>
<tr>
<td>Soundings</td>
<td>Temperature, Humidity, Pressure, Wind from ground to 25 km height</td>
<td>2 times a day</td>
<td>2015</td>
</tr>
<tr>
<td>Road Weather Observations (LIVI)</td>
<td>Road Weather Observations</td>
<td>10 min</td>
<td>open</td>
</tr>
</tbody>
</table>
View Service (WMS)

- Based on GeoServer
- Only the most common layers published

Catalog Service (CSW)

- Based on GeoNetwork
Download Service (WFS 2.0)

- Web Feature Service (WFS) 2.0 Simple Profile
- Based on stored queries
  - Predefined data sets with possibility for additional parameters (i.e. time and area)
- In-house production
Data Models

- Observations and point forecasts as GML
  - The same data is published in:
    - MultiPointCoverage
    - MeasurementTimeSeries
    - SimpleFeature

- Gridded data is provided in appropriate binary format (Grib, NetCDF, GeoTiff…)
  - WFS members contains the metadata ‘envelope’ with a link to a actual data
Download Service

Data Services

- Supported data formats: *querydata (FMI internal)*
  - Ready tools for converting other formats (*grib, netcdf, hdf…*) to *querydata*
- Memory-maps the data from NFS
- Serves both point data grid data
  - Supports both spatial and temporal interpolation and nearest point selection
Download Service
Download Plugin (WCS-like)

• Provides grid data as binary data
• Supported output formats: GRIB1, GRIB2, NetCDF and Querydata,
• Supports all proj.4 projections (depends on output format support)
• Supports slicing by
  • area (bbox)
  • elevation (pressure and/or model level)
  • time (start time, end time and origin time)
• Possibility to define grid resolution by
  • selecting every Nth grid point to x and y direction
  • grid size → data is interpolated to new grid points
Q3 (WPS-like)

- Provides service to process the data and return output as data or image
- Input: LUA scripts
- Output formats
  - Matrix as text
  - Matrix as binary (querydata)
  - JSON
  - Contoured images: svg, png, jpeg, pdf

```lua
local param = T
local limit = 0

local r, err = HIR{ hybrid=true, params={param,Z,P} }
assert(r, err)

-- Iterate levels from down to up
-- Store height and pressure when >= 'limit' (last will remain)

local m_Z = matrix() -- heights collected; originally all 'nan'
local m_P = matrix() -- pressures collected
local m_v = matrix() -- value at such positions (not needed)

for g in grids_by_level(r) do
  for pos, v in points(g[param]) do
    if v >= limit then
      m_Z[pos] = g.Z[pos]
      m_P[pos] = g.P[pos]
      m_v[pos] = v
    end
  end
end
return m_Z, m_P, m_v
```
Producing INSPIRE Data Products

Point Forecasts

Memory mapped data

Server provides logic for
interpolation the data for
requested area and time.

On-the-fly formatting
(based on template)

INSPIRE compliant
XML response

3.12.2015
Producing INSPIRE Data Products

Grid Forecasts 1/2

File System

BS Data Server

Server provides relevant meta data to fetch the data content

BS Data Server WFS Plugin

On-the-fly formatting (based on templates)

INSPIRE compliant XML response containing meta data and link to binary data

Forecast model data
Producing INSPIRE Data Products

Grid Forecasts 2/2

File System

BS Data Server

BS Data Server Download Plugin

Memory mapped data

Server provides logic for interpolation the data for requested area and time.

Format the data to requested format and projection.

Binary data

Forecast model data

3.12.2015
Producing INSPIRE Data Products

Radar Images 1/2

PostGIS DB

BS Data Server

BS Data Server WFS Plugin

GeoServer Image Mosaic plugin database

Server provides relevant meta data to fetch the data content

On-the-fly formatting (based on templates)

INSPIRE compliant XML response containing meta data and link to binary data
Producing INSPIRE Data Products

Radar Images 2/2

PostGIS DB → GeoServer → GeoServer WMS

GeoServer Image Mosaic plugin database

Raw black and white GeoTiff images so that there’s no information lost. (User may still request images as colored png)
Some Experiences

At the moment about 7200 registered users

And a little over 430,000 data downloads per day (5 req/s)
Data Models Popularity Comparison

- Measurement Timeseries: 80
- MultiPoint Coverage: 19.8
- Binary Grid: 0.2

Based on one month data (04/2014)

All data sets combined
Some Experiences

Although standards are followed, there’s a gap between provided data model and clients’ capabilities.

INSPIRE is a long project. Better to look forward than backward.
Converting everything to one data format is complicated, time consuming and expensive.

Multiple data formats are required
Handling parameters is one of the hardest parts.

Names, units, levels, time intervals…
Hard Parts

- Multidimensional data is hard to handle

Data is often 5 dimensional with irregular grid and time intervals
There’s no always support for HTTP 1.1. (in clients, proxies, load balancers…)

Data need to be transferred in chunked encoding (requires HTTP 1.1)
Lessons learned

Supporting several projections for the data is vital.

For example, many weather models are calculated in rotated lat-lon. Grids are often irregular.
Lessons learned

There have to be some way for clients to check if new data exists.

Data is updated often but not regularly.
Lessons learned

It might be a good idea to require some pub/sub functionalities in INSPIRE contexts.

New standard but quite simple to implement.
http://www.slideshare.net/tervo/
https://en.ilmatieteenlaitos.fi/open-data