Practicing Practical INSPIRE

INSPIRE Conference 2017
Introduction

Architectural Overview with examples

Implementation Issues what went wrong - how we made it right

Discussion / Wrap-up what to do about it
Architectural Overview

with examples
Architectural Overview with Examples

- Application to groundwater monitoring system
- Statistical Viewer
- EF - Bathing Sites Monitoring Facilities
- SYKE’s experiences with GeoServer
- WFS 2.0. direct access across multiple feature types
Architectural Overview with Examples

- Application to groundwater monitoring system
- Statistical Viewer
- EF - Bathing Sites Monitoring Facilities
- SYKE’s experiences with GeoServer
- WFS 2.0. direct access across multiple feature types
Application to groundwater information network

What was the plan?

● Providing groundwater levels in their context
  ○ Observations/measures (groundwater levels)
  ○ + associated features (Boreholes and facilities)
  ○ + links between them and other GeoScience things
Application to groundwater information network

What was the technology used?

WFS AppSchema: GeoServer + Constellation + Deegree

SOS: 52° North

URIs as identifiers for features (boreholes, HydrogeologicUnits, facilities) and observations (ground water levels)

Apache resolver to manage redirections

- http://ressource.brgm-rec.fr/obs/RawSeriePiezo/00463X0036/H1.2-622
Application to groundwater information network

Where were the problems

GetFeature / GetObservations with filters

- WFS AppSchema works well with GetFeatureById. Not so good with filters.
- Better with SOS but still have limitations (eg: not yet possible to query from result value).

Data duplication

- WFS AppSchema: CONCAT & co. are very memory consuming > you need to have database schema very close to the model to have direct mapping
- SOS: It was not possible to have mapping on the fly from raw DB to SOS > we had to build materialized views compliant with 52nSOS expected schema and tables
Application to groundwater information network

How did it come out? (1/2)
Application to groundwater information network

How did it come out? (2/2)

Legend:
- Feature(s)
- Observation(s)
Architectural Overview with Examples

- Application to groundwater monitoring system
- **Statistical Viewer**
- EF - Bathing Sites Monitoring Facilities
- SYKE’s experiences with GeoServer
- WFS 2.0. direct access across multiple feature types
Statistical Viewer

What was the plan:
Create simple viewer for statistical data:
• Statistical Units
• Population Distribution
No INSPIRE compliant data services available
• Implemented utilizing open data from eurostat

What was the technology used:
• Data Transformation: Java code
• Data Provision: GeoServer AppSchema & PostGIS
• Middleware: PHP Filter module
• Web GUI: Open Layers, ajax, jquery
Statistical Viewer

Where were the problems:
- Stored Queries - syntax for complex features not fully documented
- Filtering of distinct values available for specific fields not possible
- Population Distribution features massive (not geo!)
- Various GeoServer bugs (i.e. quite crash after requests for multiple complex features)

How did it come out:
Architectural Overview with Examples

- Application to groundwater monitoring system
- Statistical Viewer
- **EF - Bathing Sites Monitoring Facilities**
- SYKE’s experiences with GeoServer
- WFS 2.0. direct access across multiple feature types
EF - Bathing Sites Monitoring Facilities

**What was the plan:**
Transform bathing sites according to INSPIRE EF meeting requirements:
- Conformity of the transformed GML dataset, served by means of WFS, to INSPIRE data model and GML Specs.
- Provide both deegree and GeoServer web services

**What was the technology used:**
Data Transformation: hale studio
Data Provision:
- GeoServer (using hale studio AppSchema feature)
- deegree
**DEGREE:**

**Issue:** Feature types **ef:ObservingCapability** and **om:OM_Observation** are not present in the feature store as “Feature types” but are present as “Feature collection type hierarchy” and not published as WFS layers.

**Cause:** in the INSPIRE *EnvironmentalMonitoringFacilities.xsd*, encoding of the elements (associations):

1. "featureOfInterest" with data type "gml:FeaturePropertyType"
2. "procedure" with data type "om:OM_ProcessPropertyType"

**Solution:**

The data types for the fields have been changed to "gml:ReferenceType“ in the EF.xsd
EF - Bathing Sites Monitoring Facilities
EF - Bathing Sites Monitoring Facilities

How did it come out?
EF - Bathing Sites Monitoring Facilities

GeoServer & hale studio Appschema feature:

Issues:

- `<null>` namespaces in WFS response (`<null:OM_Observation>` even when not using virtual services)
- hale studio Appschema feature not able to correctly map multiplicity `<ef:ObservingCapability>`

Solution:

- No solution found for first issue
- Edit the appschema file
EF - Bathing Sites Monitoring Facilities

How did it come out?
Architectural Overview with Examples

- Application to groundwater monitoring system
- Statistical Viewer
- EF - Bathing Sites Monitoring Facilities
- **SYKE’s experiences with GeoServer**
- WFS 2.0. direct access across multiple feature types
SYKE’s experiences with GeoServer

What was the plan?

- Publish a BETA WFS service providing SYKE’s annex I datasets according to the appropriate *INSPIRE schemas and feature types*:
  - **Natura 2000 sites** = *Protected Sites: ProtectedSite feature type*. The national dataset comprise of both polygons and lines. Some spatial objects comprise of *both polygons and lines*.
  - **Nationally designated areas** = *Protected Sites: ProtectedSite feature type*. The national dataset comprise of polygons.
  - **River network** = *Hydrography: WatercourseLink & Hydronode feature types*. The national dataset comprise of lines and points.
- Publish the reported datasets using one GeoServer instance
- Create separate stored queries for the PS datasets
- Create required dataset (3) and service metadata (1)
SYKE’s experiences with Geoserver

The technology used:

- FME Workbench
- Esri ArcSDE Geodatabase/MS SQL Server
- PostgreSQL/PostGIS
- DB-to-DB conversion
- DB to XML mapping
- Hale Studio
- PostgreSQL/PostGIS
- DB Schema
- Data
- Mapping
- Dataset publication (WFS)
- Manual editing for AppSchema mapping
- Example GML file creation
- A validating XML Editor
- GeoNetwork open source (Finnish National Geoportal)
- CSW harvesting
- Esri Geoportal Server (SYKE Catalog)
- Geoserver with AppSchema + INSPIRE plugins
- PostgreSQL/PostGIS
SYKE's experiences with GeoServer

How did it come out?

- INSPIRE WFS service published (BETA)
  http://geoserver.ymparisto.fi/geoserver/wfs?service=wfs&version=2.0.0&request=GetCapabilities

- One Stored Query per PS dataset

- List of issues encountered, see later slides
Architectural Overview with Examples

- Application to groundwater monitoring system
- Statistical Viewer
- EF - Bathing Sites Monitoring Facilities
- SYKE’s experiences with GeoServer
- **WFS 2.0. direct access across multiple feature types**
WFS 2.0. direct access across multiple feature types

The plan?

Taking one data set, spanning many INSPIRE data themes, and transforming it for homogenized WFS download service

Going from a GIS-centric DB (topologically-correct layers organized in a single coherent GIS dataset) transposing it as INSPIRE GML datasets (4 INSPIRE data themes) & appropriate download services

Have everything validated (3*Metadata Files, 2*GetCapabilities Files, 4*GML datasets): using existing public tools;

Make everything work!
# WFS 2.0. direct access across multiple feature types

## Technology used?

<table>
<thead>
<tr>
<th>Desktop GIS (ArcGIS + QGIS) + RDBMS (Microsoft SQL Server) + ArcGIS SDE</th>
<th>Consolidating &amp; preparing the data</th>
</tr>
</thead>
</table>
|**Proprietary**: Snowflake Go Publisher - Desktop  
**Open source**: Humboldt Alignement Editor| Transforming the data + GML predefined datasets |
|**Proprietary**: Snowflake Go Publisher Server  
**Open source**: Geoserver 2.xx + AppSchema + pgSQL| Serving the data as WFS Download Service |
|**GML validation**: ENVplus, OGC TeamEngine validator;  
**MD validation**: INSPIRE validator| Validating everything: MD + GML + WFS |
|Notepad++, Python scripting, OS-GEO bug-tracker, uncountable technical discussion forums| Misc tools |
WFS 2.0. direct access across multiple feature types

The problems:

**GML Transformations**

**w. Snowflake:**
- leverage SQL views instead of transformers, for performance issues;
- really slow with M$ SQL, a lot faster with pgSQL
- all ETL transformers work with Snowflake Server

**w. HALE + Geoserver (+ AppSchema):**
- HALE transformers limited support in AppSchema
- Plenty of manual mappings done in notepad
- Plenty of database trickery to get the data to work with transformations
- Some untreated bugs, some yet to be identified, performance questions on very large datasets (millions of records)
- Things are improving, support is growing

**INSPIRE MD files**

**MD - GetCapabilities - Service MD:**
o no MD editors for this link > Notepad

Geoserver INSPIRE plugin for GetCapabilities only implements Scenario 1 of the [TG - Services](#)

Manual XML files in 2017??
WFS 2.0. direct access across multiple feature types

The problems:

<table>
<thead>
<tr>
<th>WFS 2.0.0 Services</th>
<th><strong>w. Geoserver:</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>● multiple instances of geoserver must be up &amp; running (for the same FeatureType, having selective, WFS/WMS, testing)</td>
</tr>
<tr>
<td></td>
<td>● multiple geometry types necessary (WMS works best with simple features)</td>
</tr>
<tr>
<td></td>
<td>● a LOT of open-bugs and issues still remain to be closed: OSGEO bugtracker</td>
</tr>
<tr>
<td></td>
<td>● difficult to implement custom Get Capabilities XML response</td>
</tr>
<tr>
<td></td>
<td>● AppSchema limitation: one transformation per FeatureType</td>
</tr>
<tr>
<td></td>
<td>● <a href="">null:element</a>, some WFS requests missbehave, SRS sometimes miss-encoded</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th><strong>w. Snowflake:</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>● easier to have multiple endpoints per dataset</td>
</tr>
<tr>
<td></td>
<td>● performance is database dependent</td>
</tr>
<tr>
<td></td>
<td>● Manually written GetCapabilities XML file only</td>
</tr>
<tr>
<td></td>
<td>● some WFS requests missbehave</td>
</tr>
</tbody>
</table>

**Validating MD/GML/GetCapabilities:**

|                | ● MD Validator was/is still in development, some validations are based on assumptions not real-life |
|                | ● OGC-based GML validators creates false-positive errors when validating geometry |
|                | ● Impossible to validate very large datasets, sampling data makes validations unreliable |
|                | ● GML is hard to create, harder to validate |
WFS 2.0. direct access across multiple feature types

Outcome - WFS access to homogeneous data: ps:PS, gn:NP, au:AU, br:BR

The best implementation solution seems to be based around a hybrid open-source and proprietary software:

○ proprietary SW usually has less bugs and better SW management features
○ there is little that can be done in the way of fixing bugs and issues
○ open-source solutions require more work and documentation to get going

Geoserver based solution requires quite the customization and bug-fixing, but it is 100% percent achievable

○ some technical issues regarding TG implementation still exist
○ has great Stored Query capabilities allowing for very interesting URL rewrites
○ has come a long way since the early days of INSPIRE and is now a recommendable alternative to proprietary SW such as ArcGIS, Snowflake or others.
Implementation Issues

what went wrong – how we made it right
Implementation Issues (Overview)

• Data Provision
  ○ from the perspective of the data provider the problems faced in creation and provision of the data to the public

• Data Access
  ○ from the perspective of the user the problems encountered in downloading the data

• Data Usage
  ○ from the perspective of the user the problems encountered in using the data
Implementation Issues (Overview)

Intro to the following issues:

- **Data Provision**
  - Issues providing download services (GeoServer, DeeGree)
  - Simplification Options

- **Data Access**
  - Identifier Management and Referencing
  - Stored queries

- **Data Usage**
  - Available client libraries
  - Bits&Pieces
Implementation Issues (Overview)

Intro to the following issues:

- **Data Provision**
  - Issues providing download services (GeoServer, DeeGree)
  - Simplification Options

- **Data Access**
  - Identifier Management and Referencing
  - Stored queries

- **Data Usage**
  - Available client libraries
  - Bits&Pieces
Data Provision: Issues providing download services

- Various technologies being utilized for the provision of INSPIRE Services
- Most have some deficiencies pertaining to INSPIRE
- Knowing about these issues can save a great deal of effort and frustration!
- Workarounds can help to mitigate some of these issues
- Joint funding (crowd-funding among institutions) would be ideal (but politically difficult)
- Knowing who has contracted fixes would be valuable for coordination of efforts
Main issues encountered by BRGM

● GeoServer WFS AppSchema

○ One namespace is linked to one xsd (e.g. gml is associated to GML3.1.1 or GML3.2.1 for all the app schemas based data) > If you have data based on both, then you must have several GeoServer.

○ AppSchemaCache is not automatically updated when XSD change > You have to (think that you will have to) do it manually.

○ AppSchema is mostly reduced to one to one mapping (CONCAT cannot be used for performance reasons).
Main issues encountered in SYKE implementation I

● Hard to meet the „one endpoint per dataset“ Download Services TG requirement 52 with GeoServer
  ○ Strictly speaking, if you want to publish two INSPIRE datasets you need to set up two GeoServer instances and publish them as separate endpoints
  ○ **You can only publish the same feature types having the same namespace once.** In order to provide our PS datasets (Natura, NDA) separately, we would need to set up and one GeoServer instance for each.

● When you publish WFS you automatically also generate WMS (workaround: have a separate GeoServer instance for publishing WFS services only.)

● Solution for these issues: workspace isolation (next page)
Geoserver/AppSchema Improvement: Isolated Workspaces

- Allow publishing the same (complex/simple) feature types (with the same namespace) more than once using a different data source.

- Break the connection with namespaces and the workspaces:
  - Isolated workspace could have an arbitrary prefix.
  - Namespaces and feature types added to an isolated workspace would not conflict with or be visible in other workspaces or in the global services.

- Extend AppSchema to allow more than one complex feature mapping per feature type:
  - Restriction: feature types used for feature chaining can only be mapped once or must be mapped again for each using data store (OK).

- Contractor: GeoSolutions (funded by SYKE/Envibase project) -> benefit to all
Other issues encountered in SYKE implementation II

- AppSchema restricted the HALE mapping (workaround: editing of mapping documenting by hand, for example SWE/FI names)
- We could not create optimal GMLs with the GeoServer solution, only with Atom for PS using FME or HALE alone, as AppSchema cannot handle MultiGeometry objects (workaround: lines and polygons were split up into separate spatial objects)
- GDAL interpreted the CRS in incorrectly when providing them in an INSPIRE compliant way, that is in URI-form, not URN. This has been reported and fixed.
- Open search support in GeoNetwork is not fully working
- ESRI Geoportal Server INSPIRE metadata templates are not fully according to INSPIRE requirements
- WMS Portrayal issue: according to IR code list values should be used in WMS layer names, however if we use our national code list extensions, then we fail in the name validations...
<table>
<thead>
<tr>
<th>ID</th>
<th>Problem Description</th>
<th>Workaround</th>
<th>Version</th>
<th>Date Reported</th>
<th>Reported by</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Unique Endpoint per Dataset</td>
<td>Should Apaches reviving functionality. For geoprocessing the namespace specific URI should be used for geoprocessing the request URI should be revivied with the namespace excuded.</td>
<td>2.9</td>
<td>03/17/2017</td>
<td>Kathy Schmidt</td>
<td>Finnish Environment Institute</td>
</tr>
<tr>
<td>2</td>
<td>Stored Queries (Some filtering of the data through Stored queries are not possible on complex features)</td>
<td>Set up simple features and define features on these for the id, then request the correct feature by id.</td>
<td>2.9</td>
<td>03/17/2017</td>
<td>Kathy Schmidt</td>
<td><a href="https://osgeo.org">OSSGeo Issue Link</a></td>
</tr>
<tr>
<td>ID</td>
<td>Problem</td>
<td>Description</td>
<td>Works round</td>
<td>INSPIRE Thema s Impacted</td>
<td>Version</td>
<td>Issue Link on the Forum</td>
</tr>
<tr>
<td>----</td>
<td>---------</td>
<td>-------------</td>
<td>-------------</td>
<td>--------------------------</td>
<td>---------</td>
<td>-------------------------</td>
</tr>
<tr>
<td>1</td>
<td>Support not provided for OGC HTTP URI CRS references</td>
<td>the http URI encoding for CRS in GML to be single step currently not supported by deegree. <a href="https://github.com/deegree/deegree-3/issues/711">https://github.com/deegree/deegree-3/issues/711</a></td>
<td>*</td>
<td>*</td>
<td>3.3.18</td>
<td><a href="https://github.com/deegree/deegree-3/pull/840">https://github.com/deegree/deegree-3/pull/840</a></td>
</tr>
<tr>
<td>2</td>
<td>Omission of deeply nested fields</td>
<td>when creating the tables in the DB from application schema, deegree does not include 'deeply' nested field i.e. when the element structures extend a certain complexity (with nesting and recursion) deegree simply omits them</td>
<td>manually edit configuration files</td>
<td>*</td>
<td>3.3.18</td>
<td><a href="https://sourceforge.net/p/deegree/mailman/message/12578671/">https://sourceforge.net/p/deegree/mailman/message/12578671/</a></td>
</tr>
<tr>
<td>3</td>
<td>WFS response does not show those application schema properties corresponding to empty fields in the data source</td>
<td>WFS response does not show those application schema properties corresponding to empty fields in the data source tables (derived of course from relevant empty elements in the GML file which feed the DB). This behaviour turns into an issue when no relevant information exists in the source data (because the provider does not have or simply does not want to share) for target application schema elements which are <strong>mandatory and not voidable</strong> (e.g. <code>&lt;ps:geometry&gt;</code>).</td>
<td>*</td>
<td>*</td>
<td>3.3.18</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>xlink:title is being removed on WFS (using a memory store from GML 3.2)</td>
<td>Using Deegree 3.3.20 server with an in memory storage (GML 3.2 file), there is a problem when a request for the feature is made using the WFS request. The response is translated correctly in the WFS format, but all the xlink:title are removed.</td>
<td></td>
<td></td>
<td>3.3.20</td>
<td><a href="https://github.com/deegree/deegree-3/issues/841">https://github.com/deegree/deegree-3/issues/841</a></td>
</tr>
</tbody>
</table>
# Data Provision: Issues providing download services - examples GeoServer

<table>
<thead>
<tr>
<th>Problem</th>
<th>Workaround</th>
<th>Funding</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unique Endpoint per Dataset</td>
<td>Utilize Apache's rewriting functionality</td>
<td>SYKE</td>
</tr>
<tr>
<td>Stored Queries on Complex Features</td>
<td>Documentation Issue, works, not properly documented</td>
<td></td>
</tr>
<tr>
<td>WMS doesn’t work on gml:MultiSurface</td>
<td>Create simple feature</td>
<td></td>
</tr>
<tr>
<td>WFS-T doesn't work with complex features</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Requests for multiple complex features crashes Geoserver</td>
<td>Request features individually</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Data Provision: Issues providing download services

● Extend approach for other technologies:
  ○ Deegree (in progress)
  ○ GeoNetwork
  ○ ...

● Options for formalizing current excel sheet solution
  ○ Current approach quite “fragile”
  ○ Open to general edits

● Integrate content into existing INSPIRE platform?
Data provision: the pushy idea (BRGM)

• Do we (really) need data servers?

• What about building and storing all possible requests results, store them and expose them?

• Hub’EAU approach with solR
Data Provision: Simplification Options

Problems:

- Complexity of common INSPIRE types (i.e. Geographical Names, Addresses, Related Party...) causes implementation and usage issues
- Workarounds can be dangerous; lead to unstandardized standardization (i.e. putting the entire geographical name into the GN delivery point element)

*(Currently also being discussed in MIG)*

Options:

- Software driven flattening vs. community driven schema simplifications (GeoSciML Lite, EarthResourceML Lite,...).
- Simple Feature Representations/Mapping
  - Potential of APIs exposing simplified features as an alternative solution
- Simplification through alternative encodings (JSON, RDF)
  - Also as a way towards more INSPIRE based linked open data, ldproxy, Sensor Things
GDI-DE Data Specification Analysis
Results and Recommendations
Keep it simple - nordic view
Implementation Issues (Overview)

Intro to the following issues:

• Data Provision
  ○ Issues providing download services (GeoServer, DeeGree)
  ○ Simplification Options

• **Data Access**
  ○ Identifier Management and Referencing
  ○ Stored queries

• Data Usage
  ○ Available client libraries
  ○ Bits&Pieces
Data Access: Identifier Management and Referencing

There are no requirements/recommendations for INSPIRE identifiers of the data provided. Identifiers are critical for feature referencing, and in reporting activities of MS.

INSPIRE data is usually reference data which belongs in European-wide registries of features. We don’t call the 8490 km-long E40 road, by it’s id: 27C59F82-5208-4C70-AEAC-6A8E172D95CD

We also call addresses, admin. units, rivers, geographical places by their names/abbreviations

Identifiers in INSPIRE should mirror this, since it makes working with data much easier/natural

Identifiers allow users to pin-point from very a very large data-pool the exact feature they need

In short, we need intelligent identifiers, since they make sure the data is not a mess, and make it work for everyone.
Data Access: Identifier Management and Referencing

• Identifier management loosely specified in INSPIRE, various non-aligned options available:
  ○ base:inspireId (which provides the local identifier inside a namespace, and versioning )
  ○ gml:id (default for WFS, useful for getting just the exact feature needed, restrictions++)
    ■ doesn’t allow a number of characters, many SW generate IDs randomly
  ○ gml:identifier (alt. identifier in GML, freer version of gml:id, not useful in feature filtering)

• Standard WFS GetFeatureById stored query (SQ) references gml:id
  ○ filtering the data is based on an identifier element that has many restrictions
Data Access: Identifier Management and Referencing

- What does INSPIRE TG mandatory `GetSpatialDataSet` SQ reference ???
  - unclear if this SQ is anything else than an alternative way of getting ALL the features from a WFS, or something more

- How to access a specific feature by the inspireId?
  - Where is `GetFeatureByInspireID` SQ? What about versioned data (dataset time series)?
Data Access: Identifier Management and Referencing

● How to reference specific features? First try: WFS URI including query

Problems:
  ○ URI changes with SW versions
  ○ Long and ugly URI

● Rewriter approach - provider level:
  ○ Configure Apache to rewrite simple URIs to current WFS
  ○ Simple URI used for referencing and in xlinks

● [http://ressource.brgm-rec.fr/data/Piezometre/06512X0037/STREMY.2](http://ressource.brgm-rec.fr/data/Piezometre/06512X0037/STREMY.2) vs.

Redirect + rewrite approach - European Level (nginx):

- **URI1:** RO - Ministry of Environment - Protected Areas Data Set - nat. prot. areas:
  
  ```
  http://inspire.biodiversity.ro/geoserver/ows?service=wfs&version=2.0.0&request=GetFeature&typename=ps:ProtectedSite&featuredid=ROSCI0135
  
  http://gmlid.eu/RO/ENV/PADS/PS/ROSCI0135
  ```

- **URI2:** RO - National Cadaster Agency - Administrative Units Dataset - admin. units:
  
  ```
  
  http://gmlid.eu/RO/ANCPI/UAT/AU/1.29.11940
  ```

Where MS authority abbreviated come from INSPIRE registries, dataset abbreviations are MD namespaces for that dataset, and identifiers make human readable sense
Data Access: Stored queries

- Syntax for stored queries on complex features slightly different from simple features, not very well documented.
  - For simple features element name in the fes:ValueReference sufficient
  - For complex features relative XPath must be provided, examples:
    - gml:name doesn’t work
    - ./gml:name works
    - ./ps:DesignationType/ps:designationScheme/@xlink:href
    - /cdda:DesignatedArea/ps:siteDesignation/ps:DesignationType/ps:designationScheme/@xlink:href
Data Access: Stored queries

Implementations must keep track of the following **RECs** and **REQs**:

- **Req 49**: Predefined SQ available for predefined datasets
- **Req 50**: All combinations of CRS/DataSetIdCode/DataSetIdNamespace/language available as Predefined SQ
- **Req 51**: Following parameter names must be used: CRS, DataSetIdCode, DataSetIdNamespace and Language
- **These do not really apply for WFS-based GML features**
Data Access: Stored queries

- Standardized theme specific stored queries would be valuable for data users
  - Most systems will not allow users to specify their own stored queries, so dependent on existing ones
  - A good complement to data specifications
  - Alignment across systems essential for cross-border applications
  - Discussion of potential stored query types/options for standardization

- Deficit of WFS Filters - no select distinct!
  - Essential for GUI development, which features to select
  - Otherwise App must first access all features using GetPropertyValue, filter redundancies

- Security issue
  - It seems we can delete them while not connected (need configuration to avoid that)
Implementation Issues (Overview)

Intro to the following issues:

- **Data Provision**
  - Issues providing download services (GeoServer, DeeGree)
  - Simplification Options

- **Data Access**
  - Identifier Management and Referencing
  - Stored queries

- **Data Usage**
  - Available client libraries
  - Bits&Pieces
Data Usage: Available client libraries

- Various libraries are available for the implementation of client software. These will be discussed, together with their strengths and weaknesses
  - GDAL GMLAS driver (http://www.gdal.org/drv_gmlas.html)
  - QGIS GML application schema toolbox
  - QGIS V3 - will anything ever work again since V3 will make existing vital plugins unusable?
  - Resolving xlinks

In January 2017 the “application/gml+xml; version=3.2” MIME Type was registered at IANA, and WFS changed text/xml into this new MIME type, making WFS response not readable in the browser anymore.
Clients - QGIS GML Application Schema Toolbox

- QGIS Plugin for WFS with complex features
- Can download GML from WFS2 services
- Convert GML App Schema files in PostGIS and SQLite format
- Works with QGIS3+
- GitHub: https://github.com/BRGM/gml_application_schema_toolbox

Developed by:

- BRGM - BRGM is involved for a long time in the definition of interoperability standards especially linked to OGC and the European INSPIRE directive initiatives.
- European Union's Earth observation programme Copernicus, as part of the tasks delegated to the European Environment Agency
Clients - QGIS GML Application Schema Toolbox

Env. Monitoring Facility

GroundWater raw levels
Clients - QGIS GML Application Schema Toolbox
## Clients - QGIS GML Application Schema Toolbox

### Borehole View (points) - Attributs d'entités

<table>
<thead>
<tr>
<th>Element</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>@gml:id</td>
<td>BSS001:IREWW</td>
</tr>
<tr>
<td>gml:description</td>
<td>Borehole description</td>
</tr>
<tr>
<td>gml:id</td>
<td><a href="http://ressource.brgm-rec.fr/data/BoreholeView/BSS001:IREWW">http://ressource.brgm-rec.fr/data/BoreholeView/BSS001:IREWW</a></td>
</tr>
<tr>
<td>@codeSpace</td>
<td><a href="http://www.ietf.org/rfc/rfc2616">http://www.ietf.org/rfc/rfc2616</a></td>
</tr>
<tr>
<td>gml:name</td>
<td>Porage BSS001:IREWW</td>
</tr>
<tr>
<td>gml:id</td>
<td><a href="http://ressource.brgm-rec.fr/data/BoreholeView/BSS001:IREWW">http://ressource.brgm-rec.fr/data/BoreholeView/BSS001:IREWW</a></td>
</tr>
<tr>
<td>@link:title</td>
<td>lev&amp;© hydrique&amp;© Cologique, gestion de l’Eau</td>
</tr>
<tr>
<td>gml:status</td>
<td><a href="http://resource.europe-geology.eu/vocabs/BoreholeStatus/drillingCompleted">http://resource.europe-geology.eu/vocabs/BoreholeStatus/drillingCompleted</a></td>
</tr>
<tr>
<td>@link:title</td>
<td>drilling completed</td>
</tr>
<tr>
<td>@link:title</td>
<td>hydraulic rotary drilling</td>
</tr>
<tr>
<td>gml:operator</td>
<td>BRGM (PIEZOMETRIE)</td>
</tr>
<tr>
<td>@href</td>
<td>INTRAFOR-COFOR</td>
</tr>
<tr>
<td>gml:drillDate</td>
<td>1974-11-30Z</td>
</tr>
<tr>
<td>gml:startDate</td>
<td><a href="http://resource.europe-geology.eu/vocabs/BoreholeStartDate/naturalLandSurface">http://resource.europe-geology.eu/vocabs/BoreholeStartDate/naturalLandSurface</a></td>
</tr>
<tr>
<td>@link:title</td>
<td>natural land surface</td>
</tr>
<tr>
<td>gml:inclinationType</td>
<td><a href="http://resource.europe-geology.eu/vocabs/BoreholeIndicationType/vertical">http://resource.europe-geology.eu/vocabs/BoreholeIndicationType/vertical</a></td>
</tr>
<tr>
<td>@href</td>
<td>vertical</td>
</tr>
<tr>
<td>gml:material</td>
<td>unknown</td>
</tr>
<tr>
<td>@href</td>
<td><a href="http://resource.europe-geology.eu/vocabs/BoreholeMaterialCustodian/unknown">http://resource.europe-geology.eu/vocabs/BoreholeMaterialCustodian/unknown</a></td>
</tr>
<tr>
<td>gml:length_m</td>
<td>23.0</td>
</tr>
<tr>
<td>@uom</td>
<td><a href="http://qudt.org/vocab/unit/M">http://qudt.org/vocab/unit/M</a></td>
</tr>
</tbody>
</table>
Clients - QGIS GML Application Schema Toolbox
## Clients - QGIS GML Application Schema Toolbox

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>gml:shape</td>
<td></td>
</tr>
<tr>
<td>gml:Point</td>
<td></td>
</tr>
<tr>
<td>@srsDimension</td>
<td>2</td>
</tr>
<tr>
<td>@srsName</td>
<td>urn:ogc:def:crs:EPSG::4326</td>
</tr>
<tr>
<td>@gml:id</td>
<td>gsmpl.shape.BSS001REWV</td>
</tr>
<tr>
<td>gml:pos</td>
<td>46.1909541655103 S:18713262971692</td>
</tr>
<tr>
<td>gsmpl:cored</td>
<td>false</td>
</tr>
<tr>
<td>gsmpl:accessToPhysicalDrillCore</td>
<td></td>
</tr>
<tr>
<td>gsmpl:boreholeUse</td>
<td><a href="http://inspire.ec.europa.eu/codelist/BoreholePurposeValue/groundwaterLevelMonitoring">http://inspire.ec.europa.eu/codelist/BoreholePurposeValue/groundwaterLevelMonitoring</a></td>
</tr>
<tr>
<td>@xlink:href</td>
<td><a href="http://www.opengis.net/def/nil/OGC/0/template">http://www.opengis.net/def/nil/OGC/0/template</a></td>
</tr>
<tr>
<td>@xlink:title</td>
<td>surveillance du niveau de la nappe phA@tique</td>
</tr>
<tr>
<td>gsmpl:detailedDescription</td>
<td></td>
</tr>
<tr>
<td>@xlink:href</td>
<td><a href="http://www.opengis.net/def/nil/OGC/0/template">http://www.opengis.net/def/nil/OGC/0/template</a></td>
</tr>
<tr>
<td>@xlink:title</td>
<td>template:</td>
</tr>
<tr>
<td>gsmpl:geophysicalLogs</td>
<td></td>
</tr>
<tr>
<td>@xlink:href</td>
<td><a href="http://www.opengis.net/def/nil/OGC/0/Unknown">http://www.opengis.net/def/nil/OGC/0/Unknown</a></td>
</tr>
<tr>
<td>@xlink:title</td>
<td>unknown</td>
</tr>
<tr>
<td>gsmpl:geology</td>
<td></td>
</tr>
<tr>
<td>@xlink:href</td>
<td><a href="http://ressource.brgm-rec.fr/obs/RawGeologicLogs/BSS001REWV/">http://ressource.brgm-rec.fr/obs/RawGeologicLogs/BSS001REWV/</a></td>
</tr>
<tr>
<td>@xlink:title</td>
<td>Borehole BSSO01REWV geologic log available.</td>
</tr>
<tr>
<td>gsmpl:groundWaterChemistry</td>
<td></td>
</tr>
<tr>
<td>@xlink:href</td>
<td><a href="http://data/piezometre/06S12X0037/STREMY_2">http://data/piezometre/06S12X0037/STREMY_2</a></td>
</tr>
<tr>
<td>@xlink:title</td>
<td>Des patched to BSS001REWV. Provides link to SensorObservationService of</td>
</tr>
<tr>
<td>gsmpl:rockGeochemistry</td>
<td></td>
</tr>
<tr>
<td>@xlink:href</td>
<td><a href="http://www.opengis.net/def/nil/OGC/0/Unknown">http://www.opengis.net/def/nil/OGC/0/Unknown</a></td>
</tr>
<tr>
<td>@xlink:title</td>
<td>unknown</td>
</tr>
<tr>
<td>gsmpl:poreGasChemistry</td>
<td></td>
</tr>
<tr>
<td>@xlink:href</td>
<td><a href="http://www.opengis.net/def/nil/OGC/0/inapplicable">http://www.opengis.net/def/nil/OGC/0/inapplicable</a></td>
</tr>
<tr>
<td>@xlink:title</td>
<td>inapplicable</td>
</tr>
<tr>
<td>gsmpl:geoTechnicalInfo</td>
<td></td>
</tr>
<tr>
<td>@xlink:href</td>
<td><a href="http://www.opengis.net/def/nil/OGC/0/Unknown">http://www.opengis.net/def/nil/OGC/0/Unknown</a></td>
</tr>
<tr>
<td>@xlink:title</td>
<td>unknown</td>
</tr>
</tbody>
</table>

- **Borehole URI**: http://inspire.ec.europa.eu/codelist/BoreholePurposeValue/groundwaterLevelMonitoring
- **Observations/Geological logs URI**: http://www.opengis.net/def/nil/OGC/0/template
Clients - QGIS GML Application Schema Toolbox

Borehole

URI

URI

Observations/ Geological logs

MEL/ GALET/ GRAVIER/ SABLE, ARGILEUX/

ARGILE, JAUNE COMPACT
ARGILE, GRIS COMPACT
ARGILE, BLEU
Data Usage: Codelist Registry & Content Negotiation

Background: codelist contents provided in various formats:
- HTML (Human readable)
- Re3gistry XML
- ISO 19135 XML
- RDF/XML
- JSON
- Atom

Within data, agnostic URI is provided, resolves to human readable HTML page. Example: http://inspire.ec.europa.eu/codelist/AdministrativeHierarchyLevel/1stOrder

Two options for retrieving specific formats & languages:
- INSPIRE Specific URI extensions, Example: http://inspire.ec.europa.eu/codelist/AdministrativeHierarchyLevel/1stOrder/1stOrder.en.iso19135.xml
- Content Negotiation
State of the Art in Informatics is Content Negotiation via Mime Types

<table>
<thead>
<tr>
<th>Format</th>
<th>URI Suffix</th>
<th>Mime Type</th>
<th>Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTML</td>
<td></td>
<td>text/html</td>
<td>HTML Page</td>
</tr>
<tr>
<td>Re3gistry XML</td>
<td>en.xml</td>
<td>application/xml</td>
<td>Re3gistry Encoding</td>
</tr>
<tr>
<td>ISO 19135 XML</td>
<td>en.iso19135xml</td>
<td>application/x-iso19135+xml</td>
<td>ISO 19135 XML Encoding</td>
</tr>
<tr>
<td>RDF/XML</td>
<td>en.rdf</td>
<td>application/rdf+xml</td>
<td>RDF Encoding</td>
</tr>
<tr>
<td>JSON</td>
<td>en.json</td>
<td>application/json</td>
<td>JSON Encoding</td>
</tr>
<tr>
<td>Atom</td>
<td>en.atom</td>
<td>application/atom+xml</td>
<td>Atom Encoding</td>
</tr>
</tbody>
</table>
A further bit to Content Negotiation:  
A new Content Type has been defined for gml (2017-01-09):  
  ● application/gml+xml  

Effect: Browsers no longer display the GML provided, instead make it available as a downloaded file, eg: requesteddata.application file.  

No problem if you’re aware of this, but good to know!
Data Usage: Portrayal

Portrayal rules lead to ugly WMS data services, not really usable

- The least amount of effort went into designing the portrayals of layers
- Some portrayals are just not usable: AU is just a yellow patch
- Labels and scale-dependent styling?

WMS allows for so much more styling to be done, and have really nice maps

There are suggestions on thematic cluster, but not according to INSPIRE Reqs
Data Usage: Bits & Pieces

Many tasks required in implementation of INSPIRE server and client solutions are being duplicated across Europe. Examples:

- **Generic - Codelist Resolution:** at the end of the day, the developer requires a human readable label for the concept URI
- **Specific - Filtering Middleware:** a PD feature provides many values for each spatial object; only one can be displayed. For the creation of a viewer for this data, filtering down to the relevant data via middleware can greatly improve viewer performance
- ...? (brainstorming)
Data Usage: Bits&Pieces

Ideas of what can be done:

- “Map of the state of the INSPIRE implementation puzzle”
  - List existing projects: finished/on the run, technology inside.
  - List people involved in the projects (thus contact points).
  - List past/current issues, tips, workarounds faced and found in the projects.
    - To build a cross-project vision of common objectives, tools, issues
    - To organize action to overcome them

How to make available:

- GitHub: hard to see the forest for all the trees!
- INSPIRE-in-Practice: currently only complete tools, not bits&pieces of helper code
- Could we merge this?

Also see: "Designing a new functionality to help to fill the INSPIRE technical gaps"
Wednesday 14:15h, Room: Amsterdam
Data Usage: Bits&Pieces

Ideas of what can be done:

- We need a way to liaise/link
  - people expressing their IT need (bug-fix, enhancement, ...)
  - and people having the knowledge to solve it (often being paid)

- Some sort of ‘marketplace’ like

  ![Diagram]

  - I have need X on FOSS ‘abc’
  - Me too
  - Me too

  The marketplace

  Data providers

  Identified FOSS dev, SME

  We are core committer of FOSS ‘abc’ we take care of this
Data Usage: Bits&Pieces

Ideas of what can be done:
- The pieces of the puzzle are here, no need for more advanced extra-layer

- Facilitator could be via OsGeo and/or INSPIRE cluster
  - List of know issues already shared (see previous slides)

- The ‘market place’ itself
  - could be another an INSPIRE cluster (tool oriented) and/or the ‘INSPIRE in Practice platform’
  - that should point to FOSS OsGeo mailing list / Ghubs / OGC Europe forum list / ...

⇒ Data providers IT needs solved

⇒ FOSS companies will be more visible and have more contracts
Data Usage: Bits&Pieces

Ideas of what can be done:

- BRGM issues for the MarketPlace from the French Groundwater Information Network

Identified FOSS dev, SME
Discussion / Wrap-up
what to do about it
Discussion...
Thanks for your attention
Providing INSPIRE measurement data

Thanks for your attention!