INSPIRE – What if?

Thorben Hansen, Head of Division
Danish Agency for Data Supply and Efficiency
What if INSPIRE needs to be future proofed?

Coexistence of multiple frameworks for data sharing
Spatial Data Infrastructure and data sharing

Global Spatial Data Infrastructure Association (GSDI) defines spatial data infrastructure (SDI) as:

“the technology, policies, standards, and human resources necessary to acquire, process, store, distribute, and improve utilization of geospatial data”

GSDI Cookbook, 2009

The definition originates from Federal Geographic Data Committee, 1994

INSPIRE is based on the national spatial data infrastructures (NSDI) in the member states and establishes a framework for sharing relevant spatial data within the European Community for users and applications in the environmental domain
INSPIRE and other data sharing frameworks

• On the national level INSPIRE is one of several drivers that support development of a national (spatial) data infrastructure supporting multiple domains

• Examples of other drivers are initiatives within e-government, emergency management, and the military

• Each of these initiatives have their own requirements for a data-sharing framework

• We need a solid approach and best practice recommendations for data sharing in a (spatial) data infrastructure that allow for multiple frameworks for data sharing to coexist in a cost-effective manner
Different views on the same data

- A framework for data sharing represents a specific ‘view’ on the content of the spatial data infrastructure
- INSPIRE is one view on data – other domains have other views on data
The Danish model for implementation

- GML application schemas (or GeoJSON schemas) managed independently from each other
## INSPIRE extension vs. Danish model

<table>
<thead>
<tr>
<th>Pros</th>
<th>Danish model</th>
<th>Cons</th>
</tr>
</thead>
<tbody>
<tr>
<td>• European standard</td>
<td>• National standard</td>
<td>• High level of complexity</td>
</tr>
<tr>
<td>• Based on pan-European needs</td>
<td>• Based on national needs</td>
<td>• INSPIRE skills are needed</td>
</tr>
<tr>
<td>• Well-described methodology</td>
<td>• Terms that have meaning for the users in the domain are used</td>
<td>• Certain values in code lists are not clearly defined semantically</td>
</tr>
<tr>
<td>• Possibility of reuse</td>
<td>• Model can immediately be part of the Basic Data Model</td>
<td>• Dependency on any future changes in INSPIRE data specs</td>
</tr>
<tr>
<td>• Possibility of exchanging nationally defined data via a European format</td>
<td>• Possibility to refer to classes from other models in the Basic Data Model</td>
<td>• INSPIREs code lists must be used</td>
</tr>
</tbody>
</table>

Taken from Danish Road Directorate considerations about which model to use
What if application developers didn’t know how to use our data?

Development environments that satisfy both geo-professionals and mainstream ICT developers
What is the challenge?

• INSPIRE defines the web-services that must be available for data sharing, our traditional community of geo-professionals have no problem using these web-services when building applications, and the development environment to do so is both flexible and powerful.

• However, when you get outside the inner circle of the geo-domain, development against web-services is an issue, and we often hear objections about our web-services being too complex to handle. Mainstream ICT offers more approachable development environments for occasional and less geo-savvy developers (e.g. JavaScript APIs such as Google Maps API or OpenLayers API).

• INSPIRE (and other data sharing frameworks) must find a way to offer a development environment that mainstream developers are more familiar with – and preferably an environment that can coexist with the current.
The first SDFE attempt – ‘Vis Stedet’

**Goal:**
- Give users a visual service and ease access to presentations on maps across government in an **open source** environment
- Make it easy to **overlay own information** on top of shared government base maps and geodata
‘Vis Stedet’ functionality

• Based on OpenLayers / OpenLayers API
• Tool box with HTML/JavaScript fragments
• Present WMS and WFS web-services with maps and geodata in a browser
• Overlay maps and geodata with own point or area based information stored in XML (e.g. KML)
• Find area of interest via geo-keys such as address, parcel id, place name, administrative unit, etc.
• Access to general tools such as map/geodata selector, pan and zoom
Ex. 1: Simple map

Minimum code in order to include a map in a browser
Ex. 2: AJAX drop-down with event handler

AJAX input field with drop-down auto-completion. Shows how the different DIV tags can be named and how a dedicated function can be programmed to react on a specific selection in the drop-down.
Ex. 3: AJAX drop-down wired to map

Place name search coupled directly to a map and to a specially coded event handler that handles clicks on the map and displays the corresponding coordinates.
Ex. 4: Formatted WFS look-up (KML, Atom, ...)

Address look-up combined with formatted WFS look-up resulting in KML or (as here) Atom format
What is missing?

- Shared guidelines for building HTML/JavaScript fragments
- Credible open source communities around HTML/JavaScript fragments – including somebody willing and able to take the lead
- A well understood and accepted funding regime around creation and maintenance (including support) of HTML/JavaScript fragments
- Communication about the development environment – both the API itself and about available HTML/JavaScript fragments
- Training in how to use the development environment (e.g. documented examples, hackathons, presence at the Stack Exchange network, …)
- …?