BARENTSWATCH - A SOA-BASED SURVEILLANCE SYSTEM AND INFORMATION PORTAL FOR NORWEGIAN WATERS

Frank J. ØYNES

Director, Turnkey Department, Kongsberg Spacetec AS,
Postboks 6244, NO-9292 Tromsø, Norway;
Tel: +47-90103496; Fax. +47-77655859
E-mail: frank@spacetec.no

KEY WORDS: Ocean Monitoring, Intelligent GIS, SOA, OGC Standards, Information portal, Remote Sensing

Abstract: There is a vast amount of information gathered on the oceans and coastal areas both in real time and in historical archives. However, this information is scattered between several governmental and institutional bodies. The individual pieces of information are typically tailored for a large number of different, non-cooperating legacy systems. This makes it hard to gather a complete multi-dimensional recognized picture of the area, whether the application domain is safety at sea, resource management, risk management or environmental monitoring.

Kongsberg Spacetec AS was awarded the contract to develop and implement a harmonized, multi-discipline information portal for the Norwegian Government. The project “BarentsWatch” is a SOA-based Intelligent GIS, and its design was presented during ACRS2011. The first phase of this development, the BarentsWatch Open, is now completed through acceptance testing, and the system is starting to be populated with data from the service providers. The project involves many stake holders and complex development, resulting in many challenges that have all been solved.

This presentation will give an overview of the project’s management and technical lessons learnt up to now, as well as an introduction to the follow-on phase of the project.
Furthermore we will demonstrate how the experience and technical solutions from this project can be utilized in other similar applications.

1. BACKGROUND

1.1 BarentsWatch – an initiative from the Norwegian Government

The BarentsWatch project was established by the Norwegian Government with the purpose of developing a harmonized data integration system for monitoring and warning in the Norwegian waters and coastal areas.

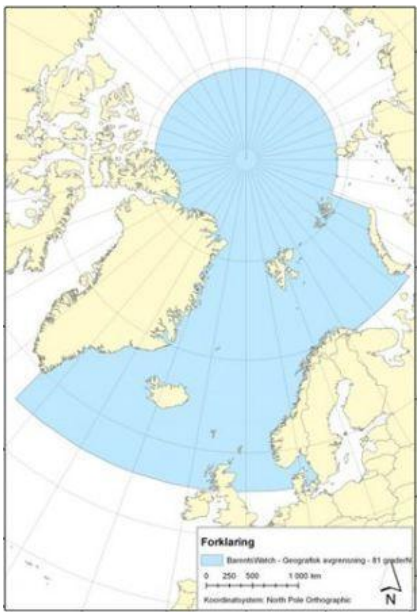

Figure 1: BarentsWatch coverage area

The system enables direct and easy access to high-quality information on climate and environment, marine traffic, marine resources and fishery, oil and gas exploitation and Norwegian sovereignty in the area. All this accomplished through harmonization and integration of data and services from a number of governmental and institutional bodies.

The system gives the authorities a combined, recognized picture of activities in the northern oceans, and enables efficient information exchange and warnings in case of incidents or accidents.

BarentsWatch consists of two different, logical parts; one open part and one closed, or rather; one *channel* for open, public data, and a separate channel also including data and services not available to the general public.

Only the public part is realized at the time of writing. The closed, or restricted, part will most likely be realized as an extension to the existing system.

1.2 BarentsWatch Open

BarentsWatch Open is a portal for the public, both in Norway and internationally. It is a free information channel for the general public, for industry, organizations and the media, as well as a source for information to be used in education at different levels. The portal gives access to data and services from a large number of BarentsWatch partners and service-providers. BarentsWatch Open has interfaces to popular social media, and encourages discussions and information sharing between the users.

1.3 BarentsWatch Closed

The closed part of BarentsWatch will act as an information exchange hub between agencies having a management responsibility in the area. BarentsWatch will here provide unified, coherent and simultaneous information to all agencies, securing that every entity has the same, common evaluated picture. The system will be a decision support system, with means for collaboration and information exchange between its users. This is expected to lead to faster, more efficient and better targeted mitigating actions in case of emergencies, accidents or illegal activities. Relevant users of BarentsWatch Closed are, among others, the military, customs control, the police, rescue-services, the coast guard and pollution control.

The closed version of BarentsWatch will most likely need interfaces, or stubs, to enable data and information exchange with various existing legacy systems.

2. DESIGN RESTRICTIONS

A number of design restrictions were imposed on the BarentsWatch project. They were all there for valid reasons and have ensured that BarentsWatch is extendable, open, and absolutely true to the standards it was set to follow. They did, however, cause significant challenges in the design and development phases of the project. When it came to the criteria listed in this section, we were not allowed to cut any corners – even if it might save significant development time.

2.1 No plug-ins, please

BarentsWatch is designed and developed to be accessible and fully functional through a standard web browser. The system does not depend on any plug-ins, hence feature-rich content using flash, java applets, Google Earth plug-ins or similar is ruled out. We received a waiver for JavaScript, as it is considered an inherent part of HTML(5) by now, and is supported by all major browsers without the need of a plug-in.

There is a clear rationale for this constraint: The number of users on restricted networks is significant. In many cases these users do not have the necessary privileges to install software on their own computers – not even a browser plug-in.

This restriction caused major challenges in BarentsWatch, especially when developing a state-of-the-art map-client to be integrated in the solution. Although good alternatives exist for features like animation and 3D-view, we had to re-implement all required features in plain JavaScript.

The benefit in the end? We now have a highly sophisticated map-client that can be used in all major browsers, on mobile devices as well as computers, and accessing any data conforming to the WxS[1] services defined by OGC.

2.2 No transfer of meta-data ownership

One of the benefits of a SOA architecture or distributed information systems, is that the data or services continue to be maintained by the original owners. When two-tier architectures were predominant, we used to attempt to copy or merge all relevant data into one huge data store under control of the system.
The result was a fast, homogeneous and deterministic system. The downside was that the content soon became out of date, inaccurate or even wrong. We lost the maintainer of the data in the integration process.

People having experienced a few technological changes will nod in recognition at this. Most of us agree that the data need to stay with the maintainer in order to keep the system accurate, quality-controlled and up-to-date.

But what about the meta-data?

The meta-data, data information, or data description, is in a distributed system equally as important as the data itself. Without correct, updated and relevant meta-data you will not be able to query, browse and assemble the information you need. This implies that the meta-data must be considered a part of the data, and not a part of the system. The meta-data must be maintained and updated by the data owner in order to stay in sync with the data it is meant to describe.

This is unfortunately neglected by most existing geo-spatial infrastructures today. It is so much easier to take control of the meta-data within the system than to quality assure meta-data harvested from all data providers.

BarentsWatch did not go for the easy solution. This has been one of BarentsWatch's toughest organizational challenges. The reason, and the solution, is given in sections 3.7 and 3.8.

2.3 No proprietary formats or protocols

The benefits of open standards are obvious when discussing integration and interoperability. These standards are however not always the most efficient ones. A significant speed-up can be achieved by using more compact data formats and protocols between the client and data-/service providers.

The early design phase of BarentsWatch aimed at utilizing this fact. While still providing all information on open standards to other systems, BarentsWatch itself could use proprietary standards for enhanced performance. This idea was abandoned for two reasons; purity of the system design, and maintainability of the system. We may have sacrificed some performance, but we now have a system where all internal and external interfaces are compliant to the standards. Expanding the BarentsWatch system into a system of systems is hence trivial, and a node may be a BarentsWatch node or another standard-conformant geospatial data service centre.

2.4 Thou shalt not copy my data

No data should be extracted from the service providers into the system framework. If the data is not of an adequate quality or in an unsupported format, this has to be solved locally.

The only exception to this rule is the ability to cache wms image representations of certain semi-static data. Data such as base-maps and long term statistics is cached within BarentsWatch, but only as image tiles representing the given data. No data-sets should be imported into BarentsWatch; data maintenance shall always remain at the service provider.

This requirement raised the necessity for fringe-projects that had to be undertaken by the service providers. A number of BarentsWatch partners have upgraded their geospatial infrastructure to a level where they now can deliver OGC-conformant WxS services of acceptable quality.

2.5 Support all major browsers

[1] Web Map Service, Web Feature Service and Web Coverage Service

BarentsWatch should run on "all major browsers". A pre-study was executed to gather statistics on browsers used by the expected BarentsWatch users, and a (rather long) short-list of browsers was compiled.

Without going into details, this caused some challenges for the developers. CSS and JavaScript must be carefully tested on all platforms, and in some cases browser-switches must be introduced in the code. Older Microsoft Internet Explorer versions proved to cause most problems. IE-7 support was eventually rejected, while we managed to get a good user-experience on IE-8.

3. CHALLENGES

In addition to the strict, but clear and well-founded, design constraints in the previous section, we encountered a number of challenges in the integration process. The users' perception of the BarentsWatch service is strongly dependent on the quality of service provided by the project partners. Given the distributed architecture, can BarentsWatch still speed up the data delivery or improve the data quality, or are we restricted to only negotiate contact between the user and the providers?

This section sheds light on some of the issues we had to address in order to improve the responsiveness and information quality exposed to the end users.

3.1 To tile or not to tile

The responsiveness of an interactive map client mainly depends on the latency between the client's request for data and the completed rendering of the received data in the client. This latency will depend partly on bandwidth limitations between the client and the data provider, and even more on the processing speed of the servers. Tiling the data, i.e. requesting smaller imagettes in order to gradually build up the picture can in many cases improves the user's perception of the responsiveness. When panning and roaming in the image, only the new, missing tiles need to be fetched, instead of requesting the full image.

Hence, in general, tiling is preferred for WMS data. It does however depend on the individual server. If the server holds a large dataset that is poorly indexed, the processing time for a small imagette may be almost as long as for the full image. Tiling will in this case result in multiple processing requests each taking almost the time of requesting a single tile covering your viewport, and the responsiveness of the client is significantly reduced.

It can be argued that this is a problem that should be solved server-side, by proper indexing and re-packing of the data. In BarentsWatch we had to implement a tuning parameter to ensure maximum responsiveness. Effectively this parameter gives the system administrator the opportunity to define non-standard tile-sizes to be used for certain server/layer combinations.

The BarentsWatch map client will consult this parameter and attempt to minimize overall data latency for each server.

3.2 My style or your style?

The WMS standard includes the possibility to request specific styling of the received data. This styling can define color-tables, icons and legends to be used for the various data types.
This Styled Layer Descriptor (SLD) possibility is barely used by the service providers relevant for BarentsWatch. Instead, the styling is fixed on the server for each layer.

In BarentsWatch this caused problems when combining layers from different providers. The data-sets with same physical parameter from different providers were not coherently displayed. In addition - different physical parameters in some cases had overlapping symbol- and color-tables. Both of these effects were causing ambiguous data interpretation.

BarentsWatch solved this in two ways – one short-term and one long-term solution.
The long-term solution included the addition of a meta-data parameter describing the SLD to be used for rendering – an optional parameter for each data layer. This parameter will be utilized as soon as the various service providers upgrade their infrastructure to support SLD.

The short term solution was to enforce a change in styling on the server side, resulting in a more coherent representation of the data. In cases where this proved impossible, data-sets with conflicting styling were simply removed from the BarentsWatch meta-data catalogue, as long as alternative data-sets for the same information existed.

3.3 Having too many layers makes it hard to find the essential information

When adding meta-data for services from a growing number of providers, the mere volume of data layers soon makes navigation difficult for the general user. A simple thing such as borders and fisheries-zones in the BarentsWatch coverage area may be 10-15 different layers. The separation of the pieces of information makes sense from a data maintenance point of view, but makes it difficult for the data user.

This has been solved in three different ways in BarentsWatch.

1. In cases where the different pieces of information all come from the same service provider, this can be handled server side by defining “group layers”. These group-layers are meta-layers queryable like any other layer, but they refer to the underlying layers for the actual data. In this way data maintenance is kept as before, but the user is given the possibility to retrieve all related information as if it was a single layer.
2. If the service provider is unable or unwilling to define group-layers, this is done by adding a BarentsWatch specific layer in the BarentsWatch meta-data catalogue, referring to the underlying layers at the service provider.
3. If the pieces of information reside on servers from two or more providers, the BarentsWatch editor can define combined layers – predefined collection of layers stored as Web Map Context (WMC) files.

Please note that the current WMC standard only supports WMS layers, not vector layers. In the future, the WMC standard may be replaced by the newer OWS Context (OWC) once it becomes a ratified standard. OWC will support both tiled and standard WMS, as well as WFS and WCS – among other standards.

3.4 Caching the wrong tiles

Some service providers attempt to speed up their services by setting up cached WMS services. The data is pre-rendered as tiles at different zoom-levels for all supported map projections. This significantly speeds up the processing time for a WMS getMap request, as all the server has to do is to return the correct, pre-rendered image.

The problem with this approach is that the cache needs to be harmonized with the map client in use. The map client must be set up to use the same, pre-defined zoom-levels and tile-size as the cache; otherwise the server will not find pre-rendered tiles matching the request.

Enforcing the same set of zoom-levels/resolutions across all service providers in a large geospatial data service centre as BarentsWatch is not feasible. For this reason, we do not utilize pre-cached services. Instead we set up a local, harmonized cache within BarentsWatch for the most requested static or semi-static data.

3.5 The missing dimensions

When combining layers in BarentsWatch, several interesting aspects were discovered. For instance, *spawning areas for North Atlantic cod* and *seismic activity areas* proved to have a near 100 % geographic overlap. How can this be, when seismic activity is believed to harm cod larvae?

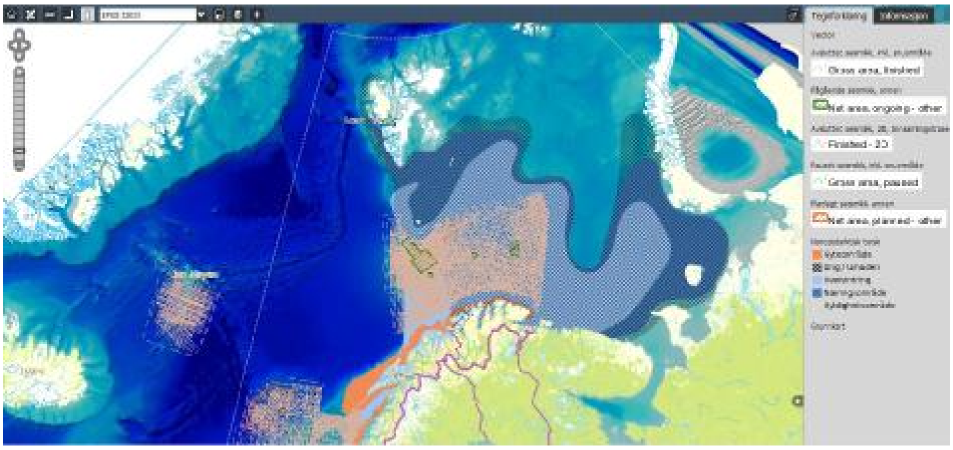

Figure 2: Seismic Activity and Cod spawning/feeding area – geographically overlapping, disjoint in time.

By careful investigation of the underlying data it was clear that the time parameter was missing. The areas were overlapping, but not at the same time of year.

The result of this discovery was an addition of a time-parameter in the map client, enabling the user to view time dependent layers as animations or step-by-step frames in time. Service providers are at this moment updating their meta-data in order to support time-based map queries.

The map client's functions for time-awareness are implemented in pure JavaScript, and still with a strict conformance to data and service standards.

3.6 Access restrictions

Some service providers had access restrictions to their data. Each user was required to go through an authentication process before accessing the services. This was an unwanted feature for BarentsWatch.

We managed to negotiate an agreement where the BarentsWatch system authenticates against the service provider once and for all, and hence all users of BarentsWatch are automatically allowed access to the data.

3.7 Incomplete or missing meta-data

Several of the BarentsWatch service providers were used to being the only user of the same data. They were not used to distributing the data to external parties, hence the meta-data or data descriptions were inaccurate, inadequate or in some cases completely missing.

As the design restrictions states that the meta-data is to be considered the responsibility of the data owner, the only permanent solution to this was to request the service providers to update their meta-data.

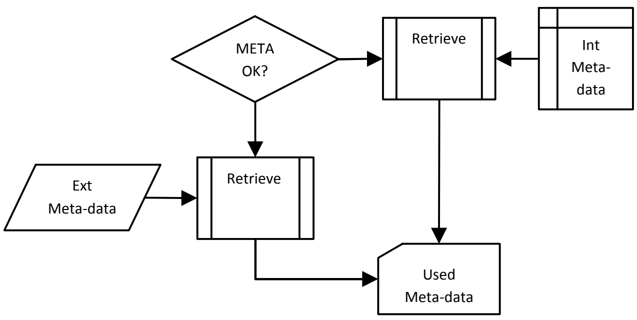

Figure 3: Retrieve internal meta-data if the external source is missing or flagged invalid

As a temporary solution we implemented the possibility for BarentsWatch administrators to add *additional* meta-data for individual servers and layers. This meta-data is used as long as the real meta-data is missing or marked invalid. As soon as the meta-data from the providers is corrected, this is automatically used.

3.8 Feature-info, or feature meta-info?

When viewing data in the BarentsWatch map client, an important feature is the ability to query additional information about the data at a given location. These queries translate into feature-info requests to the servers.

Unfortunately, the response to such queries is poorly standardized. Some servers return a simple text-string, some return nothing at all, and some even return feature-rich html including JavaScript and references only valid on their own servers.

In the BarentsWatch project an elaborate interface document was written, stating the formats and tags to be used in feature-info requests. Most service providers have been able to follow these guidelines. The guidelines are merely a more detailed specification of the format, still well within the specified open standard.

Figure 4: Example of BW-formatted Feature Info with embedded graphic elements

As a remedy for those providers not following the specified format, we are developing specific processing hooks in the client. This makes it possible to reformat the received feature-info in a pre-configured way for each individual server, resulting in a coherent feature-info presentation for all data within BarentsWatch.

4. STATUS AND FUTURE WORK

BarentsWatch was launched on schedule in May 2012. The project was completed on time and on budget. Current work involves a hardening of the system as a preparation for BarentsWatch closed. In addition we are adding functions to the framework to get a better real-time aspect in the system; support for alarms, events and rapidly changing data sources.

A separation of the map (geospatial) and portal (web content) part of BarentsWatch is also ongoing. This will enable a smoother integration in B2B or system-of-systems, and is also a necessary preparation for integration with legacy clients and systems.

The framework for a geospatial data service centre developed under the BarentsWatch project is independent of area and service domain. We are currently reusing the framework in several different projects. Two of the projects resemble the data domain of BarentsWatch, but cover different geographic areas. In another project we are investigating the possibility to use the framework in a more direct sensor-integration system. In this case the data-domain covers a very limited area around critical infrastructure, and the information is gathered from a large number of in-situ sensor systems. The large number of sensors combined with the real-time nature of the project will put the integration framework to test.

We have good hope that the planned extension of the framework to improve real-time capabilities will ensure that this test will be passed.