SUBLIMA - an open-source software for creating and maintaining topic-driven portals

Author: Anette Munthe
Affiliation: University of Oslo Library, Library of Medicine and Health Sciences
Oslo, Norway
Presenting author: Anette Munthe, anette.munthe@ub.uio.no

Background and project

SMIL is a successful topic-driven portal, containing links to quality assessed web pages on medical information understandable for laypersons. SMIL is created and maintained by a group of medical libraries in Norway, Sweden and Finland.

After years without any technical upgrading, the time had come to look for a new software for our portal. The Public Library of Oslo was in the same position, using the same software for their portal, and together we applied for funding from The Norwegian Archive, Library and Museum Authority.

We argued that it is clearly more cost-effective to develop software that can be used by many, than to fund several individual portal projects, to which the Authority agreed. A project was soon established, and the aim of the project was to develop an open source software that could be used by anyone who wants to create and maintain topic-driven portals.

A small project group was appointed, a set of specifications made, and without going into further details about the process of finding someone to develop the program, we finally signed a contract with a Norwegian company called Computas.

Key requirements for a new software

1) **Open source and open standards**
   Norway's public sector has been keen on adapting open standards in governmental organizations. This is reflected in numerous projects, including this project which is a part of The Norwegian Archive, Library and Museum Authority's program: Norwegian Digital Library.

2) **Flexibility and expressiveness in the modeling**
   In the case of our portal SMIL, we wanted a controlled vocabulary that – as far as possible - corresponded with MeSH. Other portals have other needs, so the requirement for flexibility in modeling is essential. There should be no limits in the number of levels in the vocabulary, and every portal should be able to define all necessary relationship types between concepts.

3) **Combining search and navigation**
There should be multiple ways to approach the information in the portal, and we require quality in both searching and navigation. By navigation we mean browsing by using different kinds of relationships between the topics.

4) **Full free text and metadata search capabilities, also in combination**

5) **Full multi-lingual support**
It should be possible to add as many interface languages as you want, in all kinds of characters.

6) **Faceted filtering of the search results.**
This is an important feature in a modern portal. In our old portals there were no ways to limit the search results. In Sublima it will be possible to limit the search results in as many ways as there are datafields.

7) **All types of resources on the web**
It should be possible to link to all different filetypes: text, sound, pictures, videos etc.

8) **Interactivity**
We want to give our users the possibility to subscribe to RSS-feeds, to give feedback on certain resources, suggest new resources etc.

9) **Deep search**
It should be possible to index and search data in the registered web pages, not only the metadata in the portal.

10) **Continued search**
It should be possible to use the original search argument in other sources like Google and Wikipedia.

11) **Semantic interoperability**
We wanted some kind of semantic interoperability for the data in our portals.

Semantic web and semantic interoperability

Our call for tenders did not include requirements concerning choice of technology, as we wanted to leave that open for discussion with the potential bidders. Computas, being specialists in semantic technology, offered a solution compliant with the Semantic Web recommendations of the W3C. [1]

The semantic web has been described as a web of data, as opposed to a web of hypertexts. In the web of data the links are between data or "things" described by RDF.[2] RDF stands for the Resource Description Framework, and this framework provides a standard way to organize information into semantic units. All "things" described by RDF expressions are called *resources,* and URIs are used as names for these "things". And because every "thing" has a name, everything – not just every page – is directly addressable on the (semantic) web.

In Sublima, RDF is supplemented by standard vocabularies like Dublin Core[3] and SKOS. [4] Dublin Core is used to describe the metadata of the "thing": Title, contributor, description etc. SKOS is used to describe the relationship between "things", that is the thesaurus, taxonomy, ontology, classification scheme, subject-heading system, or any other type of structured controlled vocabulary.

The purpose of using standards like these is to achieve semantic interoperability on the web, and Sublima portals can be searched by anyone (or any machine) using SPARQL [5], which is the query language for RDF. SPARQL can be used to express queries across diverse data sources.

Sublima also supports the SRU-protocol [6], although at a very basic level. SRU is an acronym for Search/Retrieval using URL. It is a standard developed by Library of congress. SRU-clients are data programs which are searching the indexes of a remote computer (SRU-server) by sending a single URL. The SRU-server returns an XML stream with data which can be used by the SRU-client.

---

[1] http://www.w3.org/2001/sw/

[2] http://en.wikipedia.org/wiki/Resource_Description_Framework

[3] http://en.wikipedia.org/wiki/Dublin_core

[4] http://en.wikipedia.org/wiki/SKOS

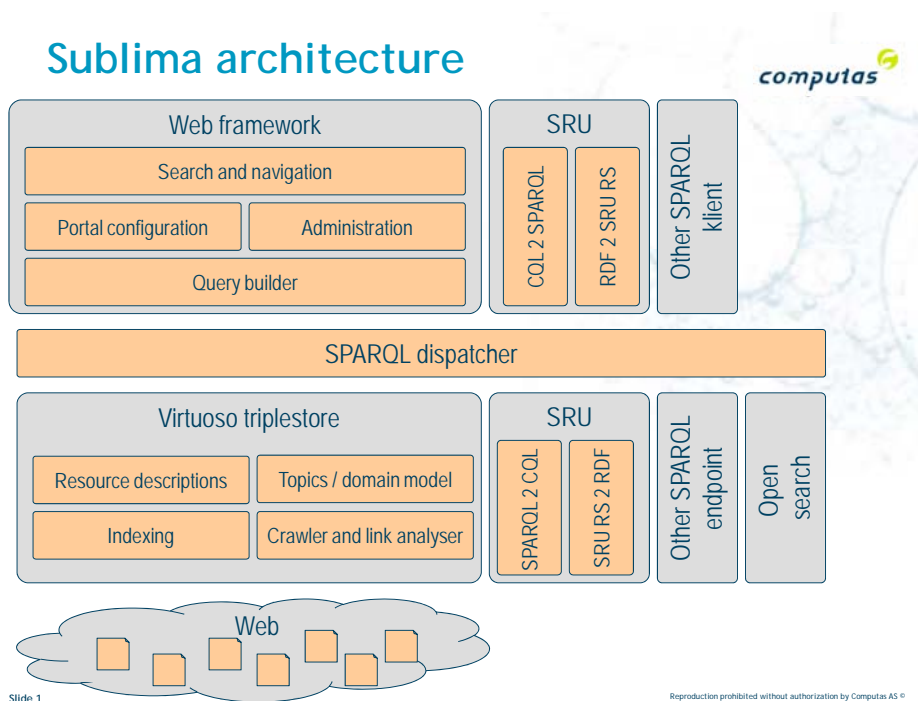[5] http://en.wikipedia.org/wiki/Sparql

[6] http://techessence.info/node/48

Architecture

Sublima is written in Java and runs on a Linux platform. What makes Sublima unique is the use of Open Link Virtuoso[7] as a database tool. This is an overview of the Sublima architecture:

Sublima architecture — computas

Web framework		SRU		Other SPARQL klient
Search and navigation		CQL 2 SPARQL	RDF 2 SRU RS	
Portal configuration	Administration			
Query builder				

SPARQL dispatcher

Virtuoso triplestore		SRU		Other SPARQL endpoint	Open search
Resource descriptions	Topics / domain model	SPARQL 2 CQL	SRU RS 2 RDF		
Indexing	Crawler and link analyser				

Web

The user interface

As mentioned earlier, Sublima is a very flexible tool, and a lot of configuration is done during the setup of the portal. Additionally it will be possible to change the interface and design by using Cascading Style Sheets (CSS) and XSLT.

There are several ways to retrieve information from the portal:

  a. Simple search

  b. Advanced search

  c. An alphabetical topic index

  d. Selected themes on the start page

The topic index gives access to all the topics in the portal – represented by preferred and alternative terms. It is also possible to add hidden labels – misspellings and so on – but these are of course hidden and do not appear in the alphabetic index.

---

[7] http://virtuoso.openlinksw.com/

As default a simple search is truncated and the logical operator between multiple words is AND. The user can however turn of the truncation and change logical operator. It is also possible to switch between different sorting criteria.

In the result page both resources and topics are presented. Two different views will be available: Short view and full view. Every portal using Sublima will of course be able to decide which fields to be shown in the different views.

The result page also displays which facets can be used to refine the result list. Possible refinement categories include: Topics, Languages, Publishers, Media type, Audience etc. Again it is the portal owner who decides which categories/data fields to use.

When entering a "topic-page", all information about that topic is displayed:

   a. Synonyms

   b. Description of the topic (if any)

   c. Related topics

   d. Related resources

There are three default relationship types: Broader, Narrower and Related, but as mentioned, you can create as many different relationship types as you like.

Administration

The administration module includes administration of: Topics (including relationship types), Resources, Users (including role types), Publishers, Link check, Database export and import, and Statistics.

Like all the data in Sublima, the topics and publishers are represented in RDF with a URI as a name. This model supports authority control, and makes revision of all information about "things" – like topics or publishers - very efficient. All together, the administration module is a great improvement compared to the administration module of our old portals.

Installation and configuration

First time installation of Sublima can be rather cumbersome as several components have to be installed before installing Sublima itself. Components needed are: Subversion, Maven, Tomcat, Java and Open Link Virtuoso.

Of these, Open Link Virtuoso, which is used as a database tool, is probably the most experimental component.

Sublima was originally developed using Postgres as a database tool. However we did have problems with the performance, and at a relatively late stage it was decided to migrate to Open Link Virtuoso, which is a software specially developed for the semantic web.

As Sublima is a very flexible tool, the configuration also involves quite a bit of work, setting up a search index, search parameters, a basis-url and database access. When this is done the application must be restarted and deployed to the Tomcat-server. Then you populate the database (if you already have data) and build the search index.

Installing new versions involves less work, of course.

Summary

If you are planning to create a portal or you already have one, Sublima might be an interesting option. With Sublima you get an open source application developed for the Library, Archive and Museum community. Sublima allows your users to access the information through searching or navigation, and to limit the search results by criteria like topics, audience, media types etc. For administrators it is a flexible tool, which allows you to create a portal for you own need. Additionally, the semantic web technology used in Sublima supports interoperability with the rest of the web.

Sublima is not yet openly available, but do not hesitate to send me an email if you are interested in more information. The plan is to publish links and instructions on the web page of The Norwegian Archive, Library and Museum Authority as soon as possible. We are hoping for a large Sublima-community and further development of Sublima in the future.

Contact

For those who want updated information, please send me an email and I will contact you as soon as Sublima is ready for use.

anette.munthe@ub.uio.no

Acknowledgement