Chapter 5
Creating and Publishing Semantic Metadata about Linked and Open Datasets

Matias Frosterus, Eero Hyvönen, and Joonas Laitio

Abstract The number of open datasets available on the web is increasing rapidly with the rise of the Linked Open Data (LOD) cloud and various governmental efforts for releasing public data in various formats, not only in RDF. However, the metadata available for these datasets is often minimal, heterogeneous, and distributed, which makes finding a suitable dataset for a given need problematic. Governmental open datasets are often the basis of innovative applications but the datasets need to be found by the developers first. To address the problem, we present a distributed content creation model and tools for annotating and publishing metadata about linked data and non-RDF datasets on the web. The system DATAFINLAND is based on a modified version of the VoiD vocabulary for describing linked RDF datasets, and uses an online metadata editor SAHA3 connected to ONKI ontology services for annotating contents semantically. The resulting metadata can be published instantly on an integrated faceted search and browsing engine HAKO for human users, as a SPARQL end-point for machine use, and as a source file. As a proof of concept, the system has been applied to LOD and Finnish governmental datasets.

5.1 Semantic Metadata Service for Datasets

Linked Data [15] refers to data published on the web in accordance with four rules [4] and guidelines [5] that allow retrieving metadata related to data entities, and linking data within and between different datasets. The datasets and their relations are represented using RDF (Resource Description Framework) and entities are identified by Uniform Resource Identifiers (URIs)[1], which allows using the HyperText

Correspondance author: Matias Frosterus, Aalto University School of Science P.O. Box 11000 FI-00076 Aalto, Finland University of Helsinki P.O. Box 68 00014 Helsingin Yliopisto, e-mail: matias.frosterus@aalto.fi. See the List of Contributors for full contact details.

[1] http://www.w3.org/TR/uri-clarification/

D. Wood (ed.), *Linking Government Data*, DOI 10.1007/978-1-4614-1767-5_5,

Transfer Protocol (HTTP) to retrieve either the resources themselves, useful descriptions of them, or links to related entities [6].

The Linked Open Data community project[2] has collected a large number of datasets and mappings between them. However, little metadata about the datasets is provided aside from short, non-uniform descriptions. As the number of linked datasets [13] grows, this approach does not allow for easy understanding of what kind of dataset are offered, who provides them, what is their subject, how they interlink with each other, possible licensing conditions, and so on. Such information should be available both to human users as well as to the applications of the Semantic Web.

Aside from linked datasets in RDF format, various organizations have also began publishing open data in whatever format they had it in. From a semantic web viewpoint, using the linked data would be an optimal dataformat, but opening data in any form is in general better than not publishing data at all [4]. For example, The governments of the United States and the United Kingdom have been releasing their governmental data in an open format[3] in different data formats (CSV, data dumps, XML etc.) and other governments are following suit. Such datasets are released with varying amounts of heterogenous associated metadata, which creates new challenges for finding and interlinking them to each other and with linked data datasets. The work presented in this chapter aims at

1. setting up a uniform metadata schema and vocabularies for annotating all kinds of datasets on the semantic web, as well as
2. a tool for collaborative distributed production of metadata and
3. an effective search tool that helps developers to find the datasets in order to use them for new applications.

There are search engines for finding RDF and other datasets, such as ordinary search engines, SWSE[4] [17], Swoogle[5] [11], Watson[6] [10], and others. However, using such systems—based on the Google-like search paradigm—it is difficult to get the general picture of the contents of the *whole* cloud of the offered datasets. Furthermore, finding suitable datasets based on different selection criteria such as subject topic, size, licensing, publisher, language etc. is not supported. To facilitate this, interoperable metadata about the different aspects or facets of datasets is needed, and faceted search (also called view-based search) [27, 14, 18] can be used to provide an alternative paradigm for string-based search.

This chapter presents a solution approach along these lines for creating, publishing, and finding datasets based on metadata. In contrast to systems like CKAN[7], a

[2] http://linkeddata.org/

[3] http://www.data.gov/ and http://data.gov.uk/

[4] http://swse.org/

[5] http://swoogle.umbc.edu/

[6] http://watson.kmi.open.ac.uk/WatsonWUI/

[7] http://www.ckan.net/

widely used system for publishing metadata about datasets, our approach is ontology based, using controlled vocabularies with RDF-based semantics. We make use of the Linked Data oriented VoiD[8] (Vocabulary of Interlinked Datasets) metadata schema [2] with some extensions for describing the datasets. Furthermore, many property values are taken from a set of shared domain ontologies, providing controlled vocabularies with clearly defined semantics. Content is annotated using a web-based annotation tool SAHA3[9] [22] connected to ONKI ontology services[10] [33, 31] that publish the domain ontologies. SAHA3 has been integrated with the lightweight multifaceted search engine HAKO[11] [22], which facilitates automatically forming a faceted search and browsing application for taking in and discerning the datasets on offer. As a proof of concept, the system has been applied to describing the LOD cloud datasets as well as the datasets in the Finnish Open Data Catalogue Project[12] complementing the linked open governmental datasets on a national level. The demonstration system called DATAFINLAND is available online[13]—it received the first prize of the "Apps4–Doing Good With Open Data" competition[14] in the company application series in 2010.

We will first present the general model and tools for creating and publishing metadata about (linked) datasets, and then discuss the VoiD metadata schema and ontology repository ONKI presenting a controlled vocabulary. After this, the annotation tool SAHA3 for distributed semantic content creation is presented along with the faceted publication engine HAKO. Finally, we will provide a review of related work in the area of publishing, searching, and exploring metadata about open datasets including the widely used CKAN registry for open data packages.

5.2 Overview of the Publication Process

Figure 5.1 depicts the generic components and steps needed for producing and publishing metadata about datasets. In the figure, we have marked the tools and resources used in our proof-of-concept system DATAFINLAND in parentheses, but the process model itself is general.

The process begins with the publication of a dataset by its provider (upper right hand corner). Metadata for the dataset is produced either by its original publisher or by a third party, using an annotation tool, in the case of DATAFINLAND the editor SAHA3. A metadata schema, in our case modified voiD, is used to dictate for the distributed and independent content providers the exact nature of the metadata

[8] http://www.w3.org/TR/void/

[9] http://www.seco.tkk.fi/services/saha/

[10] http://www.onki.fi/

[11] http://www.seco.tkk.fi/tools/hako/

[12] http://data.suomi.fi/

[13] http://demo.seco.tkk.fi/saha3sandbox/voiD/hako.shtml

[14] http://www.verkkodemokratia.fi/apps4finland

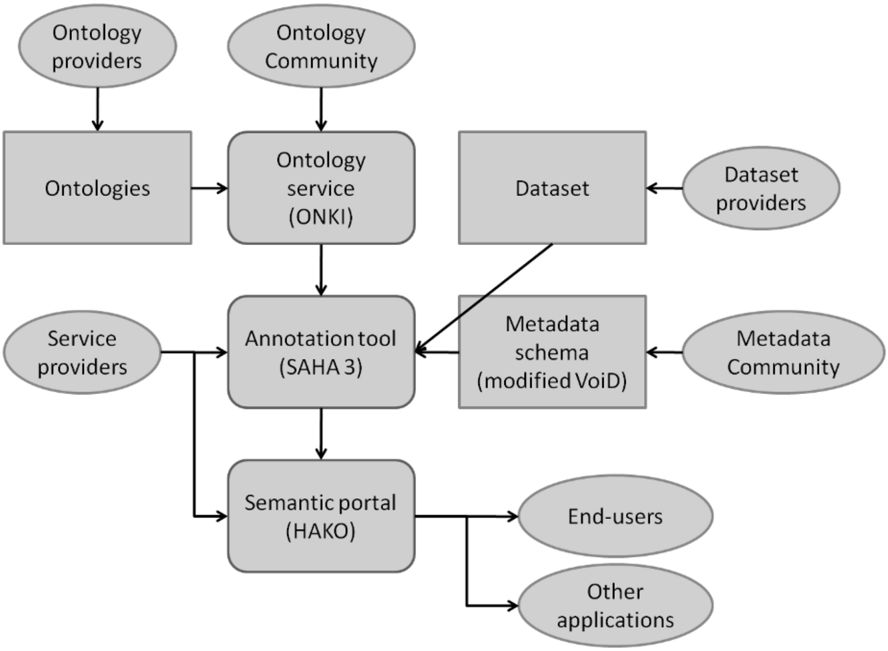


Fig. 5.1 The distributed process of producing and publishing metadata about (linked) datasets

needed according to community standards. Interoperability in annotation values is achieved through shared ontologies that are used for certain property values in the schema (e.g., subject matter and publisher resources are decribed by resources (URIs) taken from corresponding ontologies). The ontologies are developed by experts and provided for the annotation tool as services, in our case by the national ONKI Ontology Service, or by the SAHA3 RDF triple store itself that also implements the ONKI APIs used. Finally, the metadata about the datasets is published in a semantic portal capable of using the annotations to make the data more accessible to the end-user, be that a human or a computer application. For this part, the faceted search engine HAKO is used, with its SPARQL end-point and other APIs. Based on the SPARQL end-point, HAKO can also be used by itself for creating semantic recommendation links from one data instance to another.

5.3 Ontologies

A common practice in community-based annotation is to allow the users to create the needed terms, or tags, freely when describing objects. This facilitates flexibility in annotations and makes it easier for novice users to describe things. On the other hand, in the professional metadata world (e.g., in museums, libraries, and archives) using shared pre-defined thesauri is usually recommended for enhancing interoperability between annotations of different persons, and enhancing search precision and recall in end-user applications. Both approaches are usually needed, and can also be supported to some extent by e.g. suggesting the use of existing tags. For example, in the domain of open datasets, CKAN uses free tagging, but always also suggests using existing ones by autocompletion. This has the benefit that new tags are easy to add but, at the same time, there is a possibility for sharing them. However, the problem with traditional tag-based systems is that it is easy to end up with several different tags that mean the same thing, and in turn a single tag may end up denoting several different things, because the meaning in tags in not explicitly defined anywhere. This is problematic from both a human and a machine use point of view.

The traditional approach for harmonizing content indexing is to use keyword terms taken from shared vocabularies or thesauri [12, 1]. A more advanced approach is to use ontologies [29] where indexing is based on language-free concepts referred to by URIs, and keywords are labels of the actual underlying concepts. Defining the meaning behind the index terms in an explicit way, and furthermore by describing the relations between the different concepts, allows for better interoprability of contents and their use by machines. This is important in many application areas, such as semantic search, information retrieval, semantic linking of contents, and automatic indexing. With even a little extra work, e.g. by just systematically organizing concepts along subclass hierarchies and partonomies, substantial benefits can be obtained [20].

In order to make the use and sharing of ontologies easier, various ontology repositories have been developed [32, 25, 8]. The main idea behind these repositories is to offer a centralized location from which users and applications can find, query and utilize the ontologies. Repositories can also facilitate interoperability, allowing the mapping of concepts between different ontologies and guiding the user in choosing the most appropriate ontology [3].

For DATAFINLAND we used the ONKI Ontology service, which provides a rich environment for using ontologies as web services [21] as well as for browsing and annotation work. ONKI offers traditional web service (WSDL and SOAP) and AJAX APIs for easy integration to legacy applications, such as cataloging systems and search engines, and provides a robust platform for publishing and utilizing ontologies for ontology developers. The simplest way to use ONKI in providing controlled vocabularies for an application is through the Selector Widget. It is an extended HTML input field widget that can be used for mash-ups on any HTML page at the client side with two lines of Javascript code. The widget could be added to, for example, the CKAN web browser based editor, providing then the new possibility of using ontology references as tags in annotations. The ONKI widget provides its user ready-to-use ontology browser functionalities, such as concept finding, semantic disambiguation, and concept (URI) fetching. It can be configured to provide access to a selected ontology or a group of them, possibly on different ONKI-compatible servers [34], to support the use of different languages on the human interface, and so on.

In DATAFINLAND the subject property of a given dataset was connected to the ONKI instance of the General Finnish Ontology (YSO)[15] with some 25,000 concepts, because a major use case of the system is Finnish open datasets. In a similar way, any other ontology such as WordNet[16] or even DBpedia[17] could have been used through ONKI APIs.

5.4 Annotation Schemas

A note on terminology: the word vocabulary can be used to refer to the annotation terms as well as to the annotation schemas in the sense that a vocabulary defines the properties used in the annotations. Here we use the word vocabulary to refer to the terms and the word schema to refer to the annotation structure. Also of note here is that RDF schema (without the capital 'S') refers to a schema made in RDF as opposed to the RDF Schema language.

Aside from using a controlled vocabulary for describing the open datasets, another important consideration is the choice of annotation schemas that are used. If ontologies define the vocabulary, the schemas can be seen as the topics in the description outlining the information that should be recorded. The aim is to provide a concise, machine usable description of the dataset and how it can be accessed and used.

For DATAFINLAND, we chose the Vocabulary of Interlinked Datasets (VoiD), an RDF schema for describing linked datasets [2], as the starting point for our schema. One of the guiding principles behind the design of VoiD was to take into account clear accessing and licensing information of the datasets resulting in efficient discovery of datasets through search engines. Furthermore, VoiD realized effective dataset selection through content, vocabulary, and interlinking descriptions, and, finally, query optimization through statistical information about the datasets.

The basic component in VoiD is the dataset, a meaningful collection of triples, that deal with a certain topic, originate from a certain source or process, are hosted on a certain server, or are aggregated by a certain custodian. The different aspects of metadata that VoiD collects about a given dataset can be classified into the following three categories or facets:

1. *Descriptive metadata* tells what the dataset is about. This includes properties such as the name of the dataset, the people and organizations responsible for it, as well as the general subject of the dataset. Here VoiD reuses other, established vocabularies, such as `dcterms` and `foaf`. Additionally, VoiD allows for the recording of statistics concerning the dataset.
2. *Accessibility metadata* tells how to access the dataset. This includes information about the SPARQL endpoints, URI lookup as well as licensing information so

[15] http://www.yso.fi/onki/yso/

[16] http://wordnet.princeton.edu/

[17] http://dbpedia.org/

that potential users of the dataset know the terms and conditions under which the dataset can be used.

3. *Interlinking metadata* tells how the dataset is linked to other datasets through a linkset resource. If dataset :DS1 includes relations to dataset :DS2, a subset of :DS1 of the type `void:Linkset` is made (:LS1) which collects all the triples that include links between the two datasets (that is, triples whose subject is part of DS1 and whose object is part of :DS2).

In an unmodified state, VoiD supports only RDF datasets, so in order to facilitate annotating also non-linked open datasets, we made some extensions to VoiD. Using VoiD as the baseline has the benefit of retaining the usability of existing VoiD descriptions and making interoperability between the original and the extended VoiD simple.

The most important of these extensions was a class for datasets in formats other than RDF:

- `void-addon:NonRdfDataset` is similar to the `void:Dataset` but does not have the RDF-specific properties such as the location of the SPARQL endpoint.

The addition of this class resulted in modifications to most of the VoiD properties to include `void-addon:NonRdfDataset` in their domain specifications.

Another addition to the basic VoiD in our system was `dcterms:language` that facilitates multi-language applications.

In order to simplify annotation, we also defined two non-essential classes:

- `void-addon:Organization` is for including metadata about the organization or individual responsible for a given dataset, with properties `rdfs:label`, `dcterms:description`, and `dcterms:homepage`.
- `void-addon:Format` is the class for file formats and holds only the name of format under the property of `rdfs:label`.

These classes are useful in that organizations with multiple datasets can refer to the same class instance without the need to describe it again every time they annotate a new dataset. The same is true for format instances.

In order to define the format for the datasets, a new property was needed. One possibility would have been `dcterms:format` but that is normally associated with Internet Media Types and can also include information about size and duration, so we decided on a more limited property:

- `void-addon:format` records the file format of the dataset having `void-addon:Format` as the range of the property.

For annotation work, the property `dcterms:subject` was connected to the ONKI instance of the General Finnish Ontology (alternatively, this could be any other suitable, widely-used ontology) and the property `dcterms:license` to the ONKI instance of Creative Commons licenses. It is also possible for annotators to define their own licenses using the annotation tool SAHA3, which is described in the next section.

5.5 Annotation Tools

This section discusses requirements for a semantic annotation tool for DATAFINLAND and presents a solution for the task.

5.5.1 Requirements

In our application scenario, content creators are not literate in semantic web technologies, and annotate datasets in different, distributed organizations on the web. For this task, a human-friendly metadata editor that hides the complexities of RDF and OWL is needed. The editor should also allow for many simultaneous users without creating conflicts, and the results of annotations, e.g. creating a new organization instance or modifying an existing one, should be instantly seen by every other user. Otherwise, for example, multiple representations and URIs for the same object could be easily created.

For DATAFINLAND we used and developed further the SAHA3 metadata editor [22], which is easily configurable to different schemas, can be used by multiple annotators simultaneously, and works in a normal web browser, therefore needing no special software to be installed. The support for multiple annotators is made in a robust way with synchronization and locks which guarantee that the annotators don't interfere with each other's work. The tool also includes a chat channel, if online dicussions between annotators is needed. SAHA3 is available as open source at Google Code[18].

5.5.2 Initialization Process

The initialization process for a new SAHA3 project consists of two parts. First, the project is created by importing a metadata schema along with any available initial data conforming to that schema. The structure of the schema is important, since the behavior and views of SAHA3 are based on the RDFS and OWL constructs found in the schema. The classes defined serve as the types of resources that are annotated, the properties in the domain of those classes are offered in the annotation forms by default, and range definitions control what type of values a resource property field accepts. In general, the SAHA3 interface makes use of any applicable schema construct it can recognize and acts accordingly, and is often ready to be used without any further configuration. Additional RDF files can later be imported to the project —they are simply appended to the project's existing RDF model along with any schema information the new model might contain.

[18] http://code.google.com/p/saha/

Second, the SAHA3-specific configuration of the project is done through a separate configuration view. This is for configuration aspects that only concern SAHA3 and not the data in general, such as the order in which the properties are listed, the external ontology services used by certain properties and if a property should be hidden from the interface. Since this information is not usually interesting outside the context of SAHA3, it is not included in the project's RDF model but is rather stored in a separate XML configuration file. This kind of configuration is class specific, so distinct classes can have different orderings for their properties even if the properties themselves are the same.

5.5.3 Annotation Process

The annotation process is simple using SAHA3. When a SAHA3 project has been initialized and configured, the annotator is shown the main view of the project[19], giving a general overview of the annotation project. On the left side, there is a list of all metadata items that can be created, in this case format types, license types, LOD datasets, non-RDF datasets, and organizations. On the schema level, these types are represented as classes (i.e., instances of the meta class owl:Class). After the class type, one can see a number in parantheses representing the count of how many instances of that class exist in the project. In the figure, for example, metadata descriptions of 88 LOD datasets have been created. The instances can be viewed or new ones created by clicking on the corresponding type name opening up the list of instances shown.

When clicking on the name of a metadata instance, such as a dataset in our case, an item page is opened showing the basic overview of an annotated resource. Such a resource can be, for example, the metadata about the Linked Open Data dataset BBC Music[20]. In an item page like this, all property values of the resource are listed, except those that are configured to be hidden. The page also contains an [edit] button under the main title: clicking on it takes the user to the annotation page, in which the metadata can be edited.

When editing a metadata item (i.e., an instance of an owl:Class instance in the schema) on an annotation page, the annotator is provided with a number of editable fields that correspond to the properties of the class instance at hand (i.e., properties whose domain matches the class of the instance in the underlying schema are made editable) —for example, the annotation page corresponding to the BBC Music dataset[21]. Depending on the range of a given property in the schema, the field takes in either free text or instances of classes. In the latter case, the instances can be either ones defined internally in the current SAHA3 project or chosen from external resources on the web implementing the ONKI ontology web service API [31].

[19] http://www.seco.tkk.fi/linkeddata/datasuomi/saha_mainviewWithList.png

[20] http://www.seco.tkk.fi/linkeddata/datasuomi/resource_view.png

[21] http://www.seco.tkk.fi/linkeddata/datasuomi/saha.png

In DATAFINLAND, ontologies in the National Ontology Service ONKI[22] are used. Also the massive RDF triple store of the CultureSampo semantic portal [24], including e.g. DBPedia, could be used here via the ONKI API. In all cases, (semantic) autocompletion[19][16] can be used to aid the annotator.

A nice feature of the SAHA3 editor is its generality, based on RDF(S) and OWL standards, and independence of application domain. Basically, one can put in any simple RDF/OWL schema, conforming to certain generic contraints of SAHA3, and the end-user inferface and other services are automatically created online. For example, the source RDF of the project becomes available for download online, a HAKO search engine application can be created by the push of a button [22], and APIs, such as a SPARQL end-point and ONKI API are created automatically. The end-user interface can then be modified interactively according to an application's specific needs, such as ordering the metadata fields in a certain order, or hiding internal properties from the annotator.

If the metadata has already been recorded elsewhere, the resulting RDF can also be easily uploaded into an existing SAHA3 project by an interactive tool, or by simply making a union of the RDF decription files. This means that the metadata can be collected from various sources and as long as it can be transformed to conform to the shared schema in use, all the harvested data can be included for publishing.

5.5.4 Implementation

On an implementation level, we have paid special attention to the design of the underlying search index system, with performance and simplicity as the main goals. Especially global text searches and other operations that require extensive lookup can be quite slow even with contemporary triple store systems. To speed them up, the index is divided into two parts: a regular triple store (based on Jena TDB[23]) that houses all of the actual RDF data, and a full-text index (based on Apache Lucene[24]), to which most of the search operations are made. The system scales up to at least hundreds of thousands of instances.

The architecture and data flow of SAHA3 can be seen in Figure 5.2. The user interface uses both the full-text index and the triple store through page loads and asynchronous Direct Web Remoting (DWR)[25] calls, while external APIs, such as the SPARQL end-point based on Joseki[26], directly query the triple store. Business logic between the UI and the indices controls the search logic and index synchronization.

A major challenge in using multiple indices for the same data is to keep them syncronized bringing both stability and performance concerns. Fortunately, in a manual

[22] http://www.onki.fi/

[23] http://openjena.org/TDB/

[24] http://lucene.apache.org/

[25] http://directwebremoting.org/dwr/index.html

[26] http://www.joseki.org/

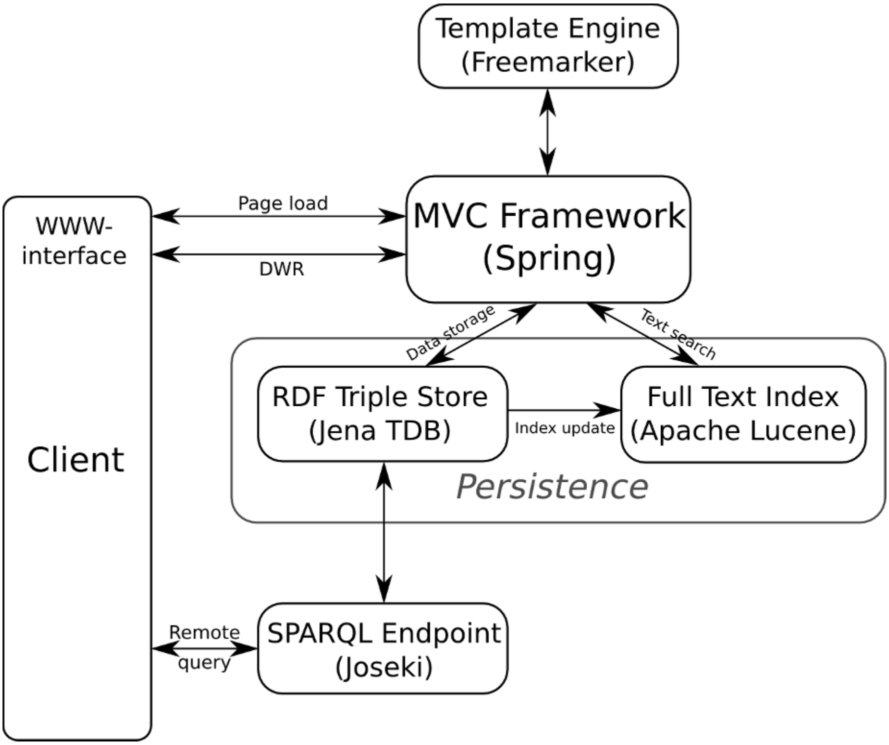


Fig. 5.2 Data flow and system architecture of SAHA 3

editing environment such as SAHA3, most write operations on an RDF graph are simple adds and deletes which are quite efficient in triple-based data. In practice their effect on performance is insignificant when compared to the speed boost given to the read operations by a fast search index.

5.6 Publishing

The final part in the publication process is the actual publishing of the datasets and the associated metadata. Ideally, the metadata about different datasets is gathered into a central repository. Since the datasets have been annotated using RDF properties and ontologies, a natural choice for exposing the metadata to human users is to use a semantic portal [28], featuring powerful semantic search and browsing capabilities, distributed content management, and direct machine-usability of the content.

In DATAFINLAND, we used HAKO, a faceted search engine, to publish the metadata recorded in the SAHA3 project as a readily usable portal [22]. The RDF data produced in SAHA3 is directly available for HAKO, which is integrated over the same index base as SAHA3.

Publishing a SAHA3 project as a HAKO application can be performed easily and interactively. First, the publisher clicks the HAKO button in SAHA3, seen in the upper right corner of the SAHA3 main view. After this, a page is opened[27] where the publisher selects the classes whose instances are to be used as search objects of the final application. All the classes explicitly defined (i.e., used as a subject) in the data model are offered as choices on the left side of the view.

On the same page, the facets corresponding to the properties of the selected classes in the search engine are selected. Similarly to the classes, all object properties (defined by being of type `owl:ObjectProperty`) present in the data are offered as facet choices. After selecting the instances and facets, clicking the link 'start hako' on the right end in the figure starts the application.

The result is a semantic portal for human end-users supporting faceted search (cf. Figure 5.3, the facets are on the left). In addition, a complementary traditional free text search engine is provided. The search input field and button is seen in Figure 5.3 below the title DataSuomi (Data in Finnish).

For machine use, SAHA3-HAKO has a SPARQL endpoint[28] which can be used to access the metadata from the outside as a service, in addition to accessing the HAKO portal via the human interface. The SPARQL interface can be used also internally in SAHA3 for providing, e.g., semantic recommendation links between data objects on the human interface.

Since HAKO and SAHA3 are built on the same index, the search engine is fully dynamic: changes made to the data in SAHA3 are immediately visible in HAKO. This is useful for making real time changes to the data, such as creating new search instances, or updating old metadata, such as the license of a dataset or its other descriptions.

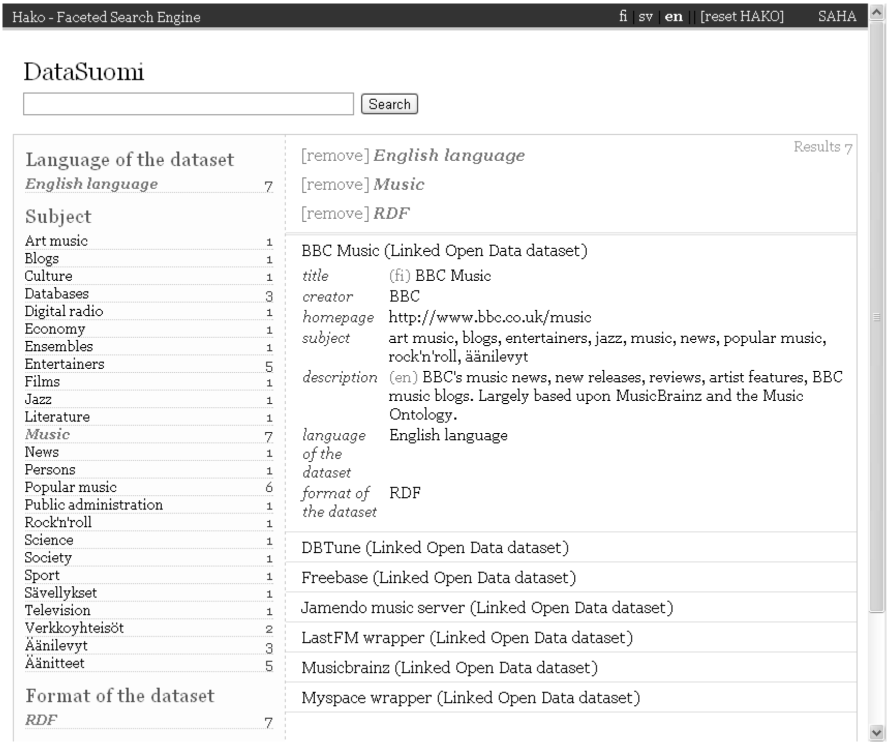
In DATAFINLAND, HAKO is configured to search for both RDF and non-RDF datasets, and to form the search facets based on the license, language, format and subject properties. This way the end-user can, for example, constrain search to cover only Linked Open datasets by choosing the RDF format. In Figure 5.3, the user has selected from the facets on the left RDF datasets concerning music in the English language. Out of the seven results provided by HAKO, the user has chosen BBC Music to see its metadata.

5.7 Discussion

In the section, contributions of the chapter are summarized, related work discussed, and some future directions for development outlined.

[27] http://www.seco.tkk.fi/linkeddata/datasuomi/hakoConf.png

[28] http://demo.seco.tkk.fi/saha/service/data/voiD/sparql?query={query}



DataSuomi

Language of the dataset

English language	7

Subject

Art music	1
Blogs	1
Culture	1
Databases	3
Digital radio	1
Economy	1
Ensembles	1
Entertainers	5
Films	1
Jazz	1
Literature	1
Music	7
News	1
Persons	1
Popular music	6
Public administration	1
Rock'n'roll	1
Science	1
Society	1
Sport	1
Sävellykset	1
Television	1
Verkkoyhteisöt	2
Äänilevyt	3
Äänitteet	5

Format of the dataset

RDF	7

Results 7

[remove] ***English language***

[remove] ***Music***

[remove] ***RDF***

BBC Music (Linked Open Data dataset)

title	(fi) BBC Music
creator	BBC
homepage	http://www.bbc.co.uk/music
subject	art music, blogs, entertainers, jazz, music, news, popular music, rock'n'roll, äänilevyt
description	(en) BBC's music news, new releases, reviews, artist features, BBC music blogs. Largely based upon MusicBrainz and the Music Ontology.
language of the dataset	English language
format of the dataset	RDF

DBTune (Linked Open Data dataset)

Freebase (Linked Open Data dataset)

Jamendo music server (Linked Open Data dataset)

LastFM wrapper (Linked Open Data dataset)

Musicbrainz (Linked Open Data dataset)

Myspace wrapper (Linked Open Data dataset)

Fig. 5.3 HAKO faceted search portal

5.7.1 Contributions

Making data open and public enhances its (re)use and allows for the development of new and innovative applications based on the data. To facilitate this, the data must be made easily findable for the developers. This paper argues that metadata based on ontologies can play here a crucial role for both human and machine end-users. Annotations created in a distributed environment can be made interoperable by using shared ontologies and ontology services, and the resulting RDF repository can be used for semantic search and browsing in human user interfaces, such faceted search in DataFinland. Using SPARQL and other RDF-based APIs means easy access for applications that wish to make use of the metadata.

A major bottleneck for creating such systems is the production, updating and editing of the metadata. It should all be as effortless as possible, and annotation tools should be easy to use, hiding e.g. the complexities of RDF from the annotators, and support collaborative, distributed work.

DataFinland provides an easy-to-use environment facilitating both semantic content creation (SAHA3) and search/browsing (HAKO) into a seamless whole. The system works in common web browsers and does not need the installation of any specialized software. SAHA3 annotation editor can be used easily with RDFS-based metadata schemas, such as the extended VoiD presented in the paper. It allows for both RDF and non-RDF datasets to be annotated in a way that is readily compatible with the existing VoiD metadata. Connected to the ONKI ontology service, SAHA3 provides a powerful annotation environment featuring controlled vocabularies that are always up to date. Finally, HAKO can be used to publish the dataset metadata in a semantic portal that is accessible, with little configuration, to both human and machine users.

5.7.2 Related Work

At the moment, the most widely used system for annotating and publishing datasets is CKAN[29] by the Open Knowledge Foundation[30]. DataFinland differs from CKAN by being based on semantic web technologies, facilitating more accurate machine processable annotations, semantic interoprability in distributed content creation, semantic search and browsing for human end-users, and RDF-based APIs for machines. The free tagging method of annotation used in CKAN, as well as in some commercial dataset repositories such as Infochimps[31] and Socrata[32] leads to various problems such as semantic ambiguity of tags as well as having several tags with the same meaning.

Concurrently to our development of the DataFinland metadata schema, an alternative approach to the schema was taken by Maali et al. for their dcat vocabulary schema [23]. Here existing data catalogues were considered, and the common properties used to describe the datasets in them were identified. Furthermore they evaluated the metadata consistency and availability in each of the data catalogues, and based on this survey, they developed their own RDF Schema vocabulary called dcat. After having defined dcat, they performed a feasibility study proving the cross-catalogue query capabilities of their system. It should be noted, however, that no matter what specific schema is chosen, mapping between different schemas should be relatively straightforward because of the fairly simple application domain, datasets. This means that it is possible to change schemas wihout too much difficulty, and that compatibility across metadata recorded according to different schemas should be fairly easy to achieve.

A special feature of DataFinland is its use of external ontology services. Aside from ONKI, there is a number of other possibilities for ontology repositories, such as BioPortal [26], which hosts a large number of biomedical ontologies. It features many advanced features, such as comprehensive mappings between ontologies and automatic indexing of the metadata for online biomedical data sets. However, at the

[29] http://ckan.org/

[30] http://okfn.org/

[31] http://www.infochimps.com/

[32] http://www.socrata.com/

moment the domain of the vocabularies is bio-focused and so not suitable for a big part of governmental datasets. Another recently developed ontology repository is Cupboard [9], where the central idea is that user's create ontology spaces which host a number of ontologies that are then mapped to one another. For searching purposes, each ontology uploaded to Cupboard is automatically indexed using the Watson[33] search engine, which has both a human UI as well as access mechanisms for machine use. Finally, there is the Open Ontology Repository project [3] that is being developed but is not yet in use.

There are also various desktop applications suited for browsing and searching for datasets, as an alternative to the web browser based solutions, An example of these is Freebase Gridworks, an open-source application used for browsing tabular datasets annotated according to the dcat schema in [7]. Through some modifications, Gridworks is able to provide a faceted interface and text search in a native UI over the dataset metadata, with tabular datasets being openable.

The approach taken by the Callimachus project[34] is somewhat similar to SAHA3, but even more general: it provides a complete framework for any kind of data-driven application. Through its custom templates and views, a similar system to SAHA3 could be constructed. However, since SAHA3 is more focused in its application area, additional tailor-made features can be utilized, such as the use of external ontology services described above. Additionally, SAHA3 is designed through simplicity to be immediately usable also for less technically oriented annotators.

5.7.3 Future Development

A problem of faceted search with wide-ranging datasets is that facets tend to get very large, which makes category selection more difficult. A solution to this is to use hierarchical facets. However, using the hierarchy of a thesaurus or an ontology intended originally for annotations and reasoning, may not be an optimal facet for information retrieval from the end-user's perspective [30]. For example, the top levels of large ontologies with complete hierarchies can be confusing for the end-users. Our planned solution in the future is to provide the annotators with a simple tool for building hierarchies for the facets as part of the annotation process. Another possible solution would be to use some kind of an all-inclusive classification system as the top level of the facets. There has been some discussion of a classification schema for open datasets in the Linked Data community, but no clear standard has risen yet. A possibility in our case could be using the Finnish Libraries' classification system that is based on the Dewey Decimal Classification.

[33] http://watson.kmi.open.ac.uk/

[34] http://callimachusproject.org/

Acknowledgements

We would like to thank Jussi Kurki for his work on SAHA3, Kim Viljanen and Jouni Tuominen for their work on ONKI ontology repository, Tuomas Palonen for his annotation work on the datasets for the DataFinland demonstration, and Petri Kola and Antti Poikola for fruitful discussions on publishing open datasets. The work on DATAFINLAND was conducted as part of the National Semantic Web Ontology project in Finland[35] (FinnONTO, 2003–2012), funded mainly by the National Technology and Innovation Agency (Tekes) and a consortium of 35 public organizations and companies.

References

1. Aitchison, J., Gilchrist, A. and Bawden, D. *Thesaurus construction and use: a practical manual*. Europa Publications, London, 2000.
2. Alexander, K., Cyganiak, R., Hausenblas, M. and Zhao, Jun. Describing linked datasets - on the design and usage of void, the 'vocabulary of interlinked datasets'. In *Linked Data on the Web Workshop (LDOW 09), in conjunction with 18th International World Wide Web Conference (WWW 09)*, 2009.
3. Baclawski, K. and Schneider, T. The open ontology repository initiative: Requirements and research challenges. In *Proceedings of Workshop on Collaborative Construction, Management and Linking of Structured Knowledge at the ISWC 2009*, Washington DC., USA, October 2009.
4. Berners-Lee, T. 2006. http://www.w3.org/DesignIssues/LinkedData.html.
5. Bizer, C., Cyganiak, R. and Heath, T. How to publish linked data on the web, 2007.
6. Bizer, C., Heath, T. and Berners-Lee, T. Linked data - the story so far. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 2009.
7. Cyganiak, R., Maali, F. and Peristeras, V. Self-service linked government data with dcat and gridworks. In *Proceedings of the 6th International Conference on Semantic Systems*, I-SEMANTICS '10, pages 37:1–37:3, New York, NY, USA, 2010. ACM.
8. d'Aquin, M. and Lewen, H. Cupboard - a place to expose your ontologies to applications and the community. In *Proceedings of the ESWC 2009*, pages 913–918, Heraklion, Greece, June 2009. Springer–Verlag.
9. d'Aquin, M. and Lewen, H. Cupboard - a place to expose your ontologies to applications and the community. In *Proceedings of the ESWC 2009*, pages 913–918, Heraklion, Greece, June 2009. Springer–Verlag.
10. dÁquin, M. and Motta, E. Watson, more than a semantic web search engine. *Semantic Web – Interoperability, Usability, Applicability*, 2011.
11. Finin, T., Peng, Yun, Scott, R., Cost, J., Joshi, S.-A., Reddivari, P. Pan, R., Doshi, V. and Li, Ding. Swoogle: A search and metadata engine for the semantic web. In *In Proceedings of the Thirteenth ACM Conference on Information and Knowledge Management*, pages 652–659. ACM Press, 2004.
12. Foskett, D.J. Thesaurus. In *Encyclopaedia of Library and Information Science, Volume 30*, pages 416–462. Marcel Dekker, New York, 1980.
13. Hausenblas, M., Halb, W., Raimond, Y. and Heath, T. What is the size of the semantic web? In *Proceedings of I-SEMANTICS '08*, 2008.

[35] http://www.seco.tkk.fi/projects/finnonto/

14. Hearst, M., Elliott, A., English, J., Sinha, R., Swearingen, K. and Lee, K.P. Finding the flow in web site search. *CACM*, 45(9):42–49, 2002.
15. Heath, T. and Bizer, C. *Linked Data: Evolving the Web into a Global Data Space*. Morgan & Claypool, San Francisco, USA, 2011.
16. Hildebrand, M., van Ossenbruggen, J., Amin, A., Aroyo, L., Wielemaker, J. and Hardman, L. The design space of a configurable autocompletion component. Technical Report INS-E0708, Centrum voor Wiskunde en Informatica, Amsterdam, 2007.
17. Hogan, A., Harth, A., Umrich, J. and Decker, S. Towards a scalable search and query engine for the web. In *WWW '07: Proceedings of the 16th international conference on World Wide Web*, pages 1301–1302, New York, NY, USA, 2007. ACM.
18. Hyvönen, E., Saarela, S. and Viljanen, K. Application of ontology-based techniques to view-based semantic search and browsing. In *Proceedings of the First European Semantic Web Symposium, May 10–12, Heraklion, Greece*. Springer–Verlag, 2004.
19. Hyvönen, E. and Mäkelä, E. Semantic autocompletion. In *Proceedings of the First Asia Semantic Web Conference (ASWC 2006), Beijing*. Springer–Verlag, 2006.
20. Hyvönen, E., Viljanen, K., Tuominen, J. and Seppälä, K. Building a national semantic web ontology and ontology service infrastructure—the FinnONTO approach. In *Proceedings of the ESWC 2008, Tenerife, Spain*. Springer–Verlag, 2008.
21. Hyvönen, E., Viljanen, K., Tuominen, J., Seppälä, K., Kauppinen, T., Frosterus, M., Sinkkilä, R., Kurki, J., Alm, O., Mäkelä, E. and Laitio, J. National ontology infrastructure service ONKI. Oct 1 2008.
22. Kurki, J. and Hyvönen, E. Collaborative metadata editor integrated with ontology services and faceted portals. In *Workshop on Ontology Repositories and Editors for the Semantic Web (ORES 2010), the Extended Semantic Web Conference ESWC 2010, Heraklion, Greece*. CEUR Workshop Proceedings, http://ceur-ws.org/, June 2010.
23. Maali, F., Cyganiak, R. and Peristeras, V. Enabling interoperability of government data catalogues. In Maria Wimmer, Jean-Loup Chappelet, Marijn Janssen, and Hans Scholl, editors, *Electronic Government*, volume 6228 of *Lecture Notes in Computer Science*, pages 339–350. Springer Berlin / Heidelberg, 2010.
24. Mäkelä, E. and Hyvönen, E. How to deal with massively heterogeneous cultural heritage data—lessons learned in culturesampo. *Semantic Web – Interoperability, Usability, Applicability*, under review, 2011.
25. Noy, N.F., Shah, N.F., Whetzel, P.L., Dai, B., Dorf,M. Griffith, N., Jonquet, C., Rubin, D.L., Storey, M.-A., Chute, C.G. and Musen, M.A. BioPortal: ontologies and integrated data resources at the click of a mouse. *Nucleic Acids Research*, 37(Web Server issue):170–173, 2009.
26. Noy, N.F., Shah, N.H., Whetzel, P.L., Dai, B., Dorf, M., Griffith, N., Jonquet, C., Rubin, D.L., Storey, M.-A., Chute, C.G. and Musen, M.A. BioPortal: ontologies and integrated data resources at the click of a mouse. *Nucleic Acids Research*, 37(Web Server issue):170–173, 2009.
27. Pollitt, A.S. The key role of classification and indexing in view-based searching. Technical report, University of Huddersfield, UK, 1998. http://www.ifla.org/IV/ifla63/63polst.pdf.
28. Reynolds, D., Shabajee, P. and Cayzer, S. Semantic information portals. In *Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters*, WWW Alt. '04, pages 290–291, New York, NY, USA, 2004. ACM.
29. Staab, S. and Studer, R., editors. *Handbook on ontologies (2nd Edition)*. Springer–Verlag, 2009.
30. Suominen, O., Viljanen, K. and Hyvönen, E. User-centric faceted search for semantic portals. Springer–Verlag, 2007.
31. Tuominen, J., Frosterus, M., Viljanen, K. and Hyvönen, E. ONKI SKOS server for publishing and utilizing skos vocabularies and ontologies as services. In *Proceedings of the 6th European Semantic Web Conference (ESWC 2009)*. Springer–Verlag, 2009.

32. Viljanen, K., Tuominen, J. and Hyvöen, E. Ontology libraries for production use: The Finnish ontology library service ONKI. In *Proceedings of the 6th European Semantic Web Conference (ESWC 2009)*. Springer–Verlag, 2009.
33. Viljanen, K., Tuominen, J. and Hyvönen, E. Ontology libraries for production use: The Finnish ontology library service ONKI. In *Proceedings of the ESWC 2009, Heraklion, Greece*. Springer–Verlag, 2009.
34. Viljanen, K., Tuominen, J., Salonoja, M. and Hyvönen, E. Linked open ontology services. In *Workshop on Ontology Repositories and Editors for the Semantic Web (ORES 2010), the Extended Semantic Web Conference ESWC 2010*. CEUR Workshop Proceedings, http://ceur-ws.org/, June 2010.