Article manuscript, submitted for review 1 June 2015

Publishing SKOS vocabularies with Skosmos

Osma Suominen, Henri Ylikotila, Sini Pessala, Mikko Lappalainen, Matias Frosterus,

National Library of Finland

Jouni Tuominen, Semantic Computing Research Group, Aalto University

Thomas Baker, Sungkyunkwan University, Seoul

Caterina Caracciolo, Food and Agriculture Organization of the UN

Armin Retterath, Central Contact Point, Spatial Data Infrastructure Rhineland-Palatinate

Abstract

Controlled vocabularies such as thesauri and classifications are published on the web for searching and browsing. With the advent of Linked Data, such vocabularies are starting to be published as RDF data using the Simple Knowledge Organization System (SKOS) model to represent concepts and their labels. General purpose Linked Data publishing tools can be used to expose such vocabulary data on the web, but this brings little benefit for users who rely largely on term-based search. Publishing tools specialized for SKOS vocabularies can provide search, browsing and other features specific to vocabularies, in addition to basic Linked Data publishing. We outline the requirements for such tools and evaluate existing SKOS publishing tools according to those requirements. We then present Skosmos, an open source web-based SKOS vocabulary browser that uses a SPARQL endpoint as its back-end. Skosmos provides a multilingual user interface for browsing and searching the data and for visualizing concept hierarchies. The user interface has been developed by analysing the results of repeated usability tests. A developer-friendly REST API is also available providing access for using vocabularies in other applications such as annotation systems.

1. Introduction

Many libraries, museums and archives develop and publish their own thesauri or other controlled vocabularies. These vocabularies are often searchable and browsable on the web for purposes such as indexing documents or performing literature searches. Typically, each vocabulary comes with its own web application, such as the MeSH browser[1], LCSH browser[2] and STW Thesaurus browser[3]. All such applications provide approximately the same services to users, i.e., term search, indexes and browsing via semantic relations, but users must learn each interface separately. Collectively, these stand-alone browsers represent significant duplication of effort. Terms from multiple vocabularies can sometimes also be searched across vocabularies via discovery services such as terminological registries (d'Aquin & Noy 2011; Golub et al., 2014).

In today's Linked Data context, the concepts of thesauri and other controlled vocabularies are given Web identifiers (URIs), and these identifiers are increasingly used to tag and index resources. The use of concept identifiers for indexing is more precise than the traditional practice of indexing with the words and phrases of the concept labels.

Bringing controlled vocabularies such as thesauri and classifications to the Linked Data world typically involves representing their contents, whether once and for all or by periodic update from source, using Resource Description Framework, a W3C standard data model, and Simple Knowledge Organization System (SKOS), an RDF extension vocabulary specifically for describing concept schemes (Miles & Bechhofer, 2009; Baker et al., 2013) and publishing the result according to Linked Data principles (van Assem et al., 2006; Summers et al., 2008; Neubert, 2009; Lange et al., 2012; Zapilko et al., 2013; Caracciolo et al., 2013).

If concept URIs provide a solid basis for precision indexing, however, humans need to use the related words and phrases when browsing or searching for terms. People need to explore vocabularies when tagging resources for indexing or when formulating search queries.

[1] http://www.nlm.nih.gov/mesh/MBrowser.html
[2] http://id.loc.gov/authorities/subjects.html
[3] http://zbw.eu/stw/

Linked data standards such as SKOS, RDF, and SPARQL provide a basis for general-purpose platforms for publishing and browsing data (e.g., Pubby[4] or Elda[5]), but such tools are generally not well suited for vocabulary data. Existing Linked Data publication tools are particularly weak in the area of searching for resources by the words and phrases used to describe them, because terms are represented as literal values and general purpose RDF tools expect entities to be queried by URIs. These tools also typically lack visualization methods essential for vocabularies, e.g., for displaying concept hierarchies. Only a few specialized tools are available today for publishing concept schemes as Linked Data.

This paper presents Skosmos, a new tool for publishing controlled vocabularies both for human-friendly display and machine access via Linked Data and APIs. The paper begins with general requirements and desirable features for SKOS publication tools, then discusses the specific features of Skosmos as compared with other tools of similar purpose. The following section presents use cases and requirements for publishing controlled vocabularies using SKOS. Section 3 compares existing SKOS publishing tools with respect to those requirements. Section 4 describes the architecture of Skosmos, publication processes, and the results of usability tests, and Section 5 describes how Skosmos is currently being used. Section 6 looks ahead at future work.

2. Requirements for SKOS vocabulary browsers

In our experience, and supported by related work (Tuominen et al., 2009; van der Meij et al, 2010; White et al., 2013), the main use case for using published vocabularies on the web is search and browsing of vocabulary concepts, usually for the purpose of indexing material such as books, articles, artworks and other kinds of documents. Vocabularies can also be referred to in information retrieval tasks for looking up suitable terms to use as search keywords, used as translation aids, or referenced during the development of new vocabularies and mappings between existing vocabularies. We see the following main requirements for vocabulary browsers:

Term search. The most important facility for accessing a controlled vocabulary is by searching for terms, i.e. labels of concepts. An autocomplete feature will give instant feedback to users about the expected

[4] http://wifo5-03.informatik.uni-mannheim.de/pubby/
[5] http://www.epimorphics.com/web/tools/elda.html

search results for a particular query. In many systems, searches can be further restricted to particular types of concepts or specific parts of the vocabulary. For multilingual vocabularies, searches can be targeted by language.

Index displays. Any controlled vocabulary can be displayed alphabetically. In addition, some vocabularies such as classifications and some thesauri have a hierarchical structure that can be used to display a hierarchical index to drill down from general to more specific concepts. Some vocabularies have a thematic or domain-centric structure independent of the hierarchy which can be used as a basis for a thematic index.

Extensibility. While SKOS can be used to represent the content of many controlled vocabularies, specialization is sometimes necessary, and should therefore be supported in publishing tool. SKOS itself, as an RDF vocabulary, was designed for extensibility (Baker et al., 2013). For example, custom properties can be defined extending the SKOS properties, and also specific types of concepts can be defined as subclasses of SKOS Concept. The SKOS eXtension for Labels (SKOS XL) enables stating facts about lexical labels in the vocabulary, at the cost of some added complexity. Further extensions to SKOS and SKOS XL are defined in ISO 25964 (ISO, 2011), including specific types of hierarchical relationships (generic, partitive and instantial) and the distinction between two subtypes of SKOS Collections: *arrays* of concepts, which can be used to subdivide the set of sibling concepts using node labels such as “milk by source animal” and “milk by fat content”, and *groups* of concepts, which can be used for other types of groups, including thematic or domain-centric groups.

RDF representations. Like all Linked Data publishing tools, SKOS vocabulary publishing tools can be expected to expose the underlying RDF triples expressed in various syntaxes such as RDF/XML or Turtle. If these serializations are served using HTTP content negotiation when vocabulary URIs are accessed, there is no need for a separate Linked Data publishing infrastructure.

Access through APIs. In addition to serving basic Linked Data, a vocabulary publishing tool can also expose the vocabulary data using other styles of APIs such as Web Services or REST-style APIs. These can be used to expose the term-centric search functionality available in the publishing tool. Such APIs can be used for accessing vocabularies from other applications, or as a basis for user interface widgets that provide vocabulary services (Viljanen et al., 2008). A detailed review of SKOS-oriented APIs is given by Binding and Tudhope (2010).

Open source. Using an open source tool for publishing vocabularies has several benefits in addition to being free of licensing costs. The source code of the tool can be inspected and new features and local customizations may be implemented without necessarily needing to consult the original authors. On the other hand, applying an open tool may require more technical skills than using a commercial product which is supported by a company.

3. Related work on vocabulary publishing tools

In order to update our knowledge of the current landscape of tools for publishing Linked Data controlled vocabularies, we performed a literature search and examined web resources such as the comprehensive listing of SKOS-related tools on the W3C SKOS wiki[6].

We limited the comparison of related vocabulary publishing tools using three criteria:

1. We only included tools that are not tied to a single vocabulary and support SKOS as an input format. This excludes many ontology publishing tools and ontology library systems (d’Aquin & Noy, 2012), which are based on ontology languages including OWL and OBO.
2. We only included tools that support some kind of search functionality (e.g. term-based search) on the SKOS vocabulary data instead of only providing access via URIs and/or SPARQL queries. This excludes general purpose Linked Data publishing tools such as Pubby, Elda and implementations of the Linked Data Platform (2015) which can be used for any RDF data but cannot perform searches.
3. We excluded purely commercial tools not available as open source software (one of the requirements outlined above) such as Lexaurus Bank, Mondeca Intelligent Topic Manager, PoolParty Thesaurus Editor, and TopBraid Enterprise Vocabulary Net. Moreover, commercial tools have generally not been described in academic literature, it would have been difficult to characterize their features strictly on the basis of published material.

[6] http://www.w3.org/2001/sw/wiki/SKOS

Based on a review of literature and targeted web searches, we found four comparable tools that can be used to publish any SKOS vocabulary. Each tool is briefly described in the following subsections, followed by a summary of the tools as a whole.

3.1. HIVE

HIVE (Helping Interdisciplinary Vocabulary Engineering) is a system that assists in indexing of documents (Greenberg et al., 2011; White et al., 2013). The system is pre-loaded with multiple SKOS vocabularies. The web user interface of HIVE provides concept search and browsing facilities as well as an automated indexing tool which, when given a text document, suggests concepts from a user-selectable subset of the vocabularies. The main focus of the tool is on the automated indexing aspect, which is implemented using the KEA++ algorithm (Medelyan, 2009). In the following, we will only concentrate on the vocabulary browsing aspect of HIVE. HIVE is available as open source software.

3.2. iQvoc

iQvoc[7] is a vocabulary management tool that supports editing and publishing of SKOS datasets (Bandholtz et al., 2010; Bandholtz et al., 2011). Originally built for the maintenance of German environmental vocabularies, it has since evolved into a general purpose multi-user SKOS and SKOS XL editing tool. For unauthenticated users, it provides search and browsing access to concepts in the vocabulary. In the following, we will only concentrate on the publishing aspect of iQvoc. The system is developed by innoQ Deutschland GmbH, a commercial company, but available as open source software.

3.3. CATCH

The CATCH demonstrator (van der Meij, 2010) is a SKOS-based vocabulary and alignment repository. It consists of middleware providing vocabulary-oriented access services and APIs. One of the components is a vocabulary and alignment browser that can be used to search for concepts, for example to support manual indexing of documents. In the following, we concentrate on the browsing tool of CATCH. At the time

[7] http://iqvoc.net

of writing this article (May 2015), the online demonstrator of the tool was not functional, and the source code is not available for local testing, so our description of CATCH is only based on the publication.

3.4. ONKI

ONKI (Viljanen et al., 2009; Tuominen et al., 2009) is an ontology repository with support for SKOS vocabularies. It has been used for publishing the Finnish Linked Open Ontology Cloud KOKO (Frosterus et al., 2013) and several international vocabularies. ONKI has a web user interface for browsing the vocabularies and searching for concepts, with support for querying multiple vocabularies simultaneously. ONKI has a strong focus on facilitating manual content annotation by providing an autocompletion widget and several APIs. It also supports information retrieval tasks with ontology-based query expansion facilities.

3.5. Summary of publishing tools

There are relatively few general purpose SKOS publishing tools. Of the tools we surveyed, only ONKI is directly positioned as a vocabulary publishing tool. All the other tools have a different focus but also provide vocabulary publishing and access facilities. A summary of some of their features related to publishing is given in Table 1, which also includes our tool Skosmos for comparison.

All the surveyed tools provide term search facilities and all except HIVE also have an autocomplete feature that suggest concepts when a partial term is typed. In all tools search results can also be limited by other criteria, but the choice of criteria varies. HIVE only offers restricting the search by vocabulary. iQvoc allows limiting by type (either Concept or Collection) and by membership in a SKOS Collection. ONKI allows limiting search to a particular part of the hierarchy by selecting a parent concept, while also allowing restrictions on concept type, group (collection membership), vocabulary and language.

Most tools offer an alphabetical listing of concepts. In HIVE, the alphabetical listing also incorporates some of the hierarchy, by making it possible to open a concept and display its children. iQvoc can display the topmost concepts in the hierarchy and allows opening and closing branches. It also has a separate listing of expired concepts. ONKI has a hierarchical index showing the topmost concepts and their subconcepts. In addition, ONKI supports a thematic index of concept groups, a feature which is used, e.g., in the Finnish General Thesaurus YSA to group concepts by domain.

All tools provide machine access in the form of APIs alongside the human-oriented search and browsing facilities. The specifics vary, but generally API access is implemented either as SOAP-based Web Services API or as a REST API. In some cases both technologies are supported.

In summary, the existing tools for publishing SKOS datasets are not easy to deploy for simply publishing one or more SKOS vocabularies. Only HIVE and iQvoc are available as open source software, but these tools are not primarily aimed at publishing of vocabularies. HIVE is geared towards automatic text indexing, does not have a multilingual user interface, and offers few search facilities beyond simple term search. iQvoc offers an integrated solution for vocabulary management, but is not well suited for scenarios where the vocabulary is maintained using another system (for example, periodically converted from a legacy format to SKOS) and only the publishing aspect is needed.

Table 1: Comparison of SKOS publishing tools.

	HIVE	iQvoc	CATCH	ONKI	Skosmos
Purpose	Machine-aided indexing with multiple controlled vocabularies	Thesaurus editor with publishing functionality	Middleware for accessing SKOS vocabularies and alignments	Ontology and SKOS vocabulary publishing tool	SKOS vocabulary publishing tool
Implementation technology	Java	Ruby on Rails	Java	Java, PHP	PHP
Database technology	Sesame triple store, Lucene index, custom indexes	Relational database	Sesame triple store or SPARQL endpoint	Custom, based on Apache Jena and Lucene	SPARQL 1.1 triple store with optional text index
Search functionalities	Term search, limit by vocabulary	Term search, automatic suggestions, limit by type or collection/group	Term search, autocomplete, type of label, language	Term search, autocomplete, limit by type, group, parent, vocabulary, language	Term search, autocomplete, limit by type, group, parent, vocabulary, language
Index displays	Combined alphabetical and hierarchical	Alphabetical, Hierarchical, Expired concepts	?	Alphabetical, Hierarchical, Thematic groups	Alphabetical, Hierarchical, Thematic groups,
Collections as arrays, e.g. “milk by source animal”	No?	No	No	Yes	Yes
SKOS XL support	No	Yes	No	No	No
Multilingual user interface	1 language	2 languages	?	3 languages	5 languages
API access	REST API	Linked Data, Web API	Web Services, REST API	Web Services, HTTP/REST API, Linked Data	REST API, Linked Data, (SPARQL)
Open source	Yes	Yes	No	No	Yes

4. The Skosmos tool

The previous section showed that there are few general purpose tools for publishing SKOS vocabularies. This is a problem for vocabulary publishers, who either need to develop their own custom web applications for publishing their vocabularies or offer only basic Linked Data without vocabulary-oriented search access. From a data consumer perspective, having only Linked Data access to SKOS datasets makes it difficult to integrate support for controlled vocabularies, such as autocomplete-enabled form fields for indexing, into other systems.

This section presents Skosmos[8], a controlled vocabulary publishing tool which is based on current Web standards, supports many vocabularies, and is both user and developer friendly. Skosmos provides a multilingual user interface for browsing and searching the data and for visualizing concept hierarchies. The user interface has been developed by analysing the results of repeated usability tests. A developer-friendly REST API is also available for accessing vocabularies in other applications such as annotation systems.

Skosmos was built on the basis of prior work on developing vocabulary publishing tools in the FinnONTO (2003–2012) research initiative at the Semantic Computing Research Group[9], which aimed at developing a national level semantic web ontology infrastructure in Finland. The ONKI Ontology Library Service (Viljanen et al., 2009), one of its key results, has been operated as a living lab service since 2008 for serving ontologies for human and machine use. A core part of the service is the ONKI SKOS server (Tuominen et al., 2009) for publishing SKOS vocabularies and similar lightweight RDF ontologies. ONKI uses custom APIs for accessing the vocabularies in the RDF databases of the backend servers.

The development of Skosmos was undertaken a more lightweight successor system to ONKI within the FinnONTO research project. The first step was ONKI Light (Suominen et al., 2012), a prototype for a vocabulary browser on top of a SPARQL endpoint. In 2013, the development of ONKI Light moved to the National Library of Finland where it was renamed Skosmos in 2014, marking the shift from a research prototype to a mature, general purpose SKOS publishing tool.

[8] http://skosmos.org
[9] http://www.seco.tkk.fi

.

4.1. Architecture

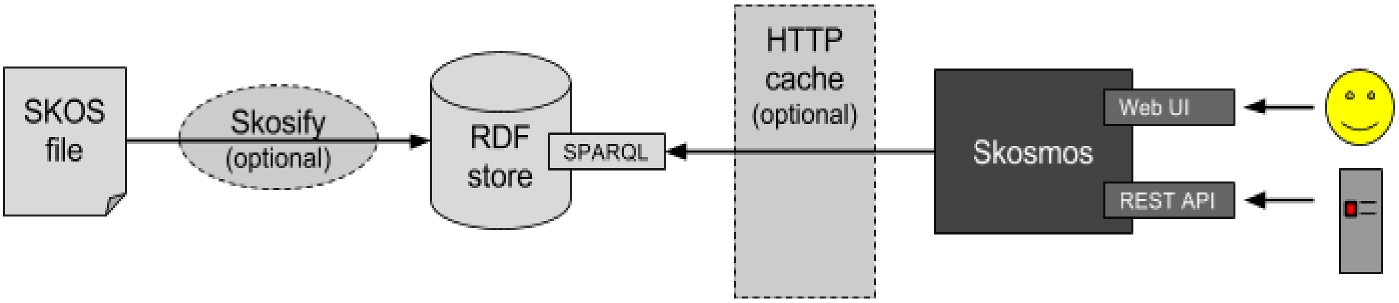


Figure 1: Skosmos system architecture

Skosmos relies on a SPARQL endpoint as its back-end data store and is written mainly in PHP. The Skosmos software provides both a web user interface and API access. The Skosmos web user interface is rendered using the Twig templating engine. Some added functionality is provided by client side JavaScript code. As an example, some content is loaded by asynchronous queries to avoid time intensive full page reloads. However, the basic interface is HTML-based and all content is accessible to web search engines.

The main benefits of using a SPARQL endpoint for data storage is that the data provided by the service is always up to date. This allows fast update cycles in vocabulary development. On the other hand the performance of a SPARQL endpoint for vocabulary access is not always optimal (e.g., van der Meij, 2010). In particular, SPARQL 1.1 does not provide an efficient way to perform text queries, so term-based searches can be slow. Filters can be used for term searches, but they do not perform well with larger vocabularies. Therefore we recommend using the Apache Jena Fuseki[10] triple store with the jena-text index [11] for large vocabularies. In addition to using a text index, caching of requests to the SPARQL endpoint with a standard HTTP proxy cache such as Varnish[12] can be used to achieve better performance for repeated queries, such as those used to generate index views.

With datasets consisting of tens of thousands of concepts, Skosmos page generation times are usually below 300 ms, when running on a regular Linux virtual server and using a jena-text index. For repeated queries to the same page, faster page generation is possible thanks to HTTP caching of SPARQL query

[10] http://jena.apache.org/documentation/serving_data/
[11] https://jena.apache.org/documentation/query/text-query.html
[12] https://www.varnish-cache.org/

results. Loading of external Linked Data resources may cause extra delays depending on the response time of the external server. Generating some alphabetical index views and vocabulary statistics can take several seconds when accessed for the first time.

4.2. Data requirements

In general, Skosmos should work with any well-formed SKOS Core vocabulary. Skosmos is not limited to SKOS Core: there is support for many Dublin Core properties, some extensions to SKOS defined in ISO 25964, and also any custom classes and properties which have been defined in RDF data. We recommend using the Skosify[13] tool (Suominen & Mader, 2014) to pre-process SKOS vocabularies and correct potential problems before publishing them with Skosmos.

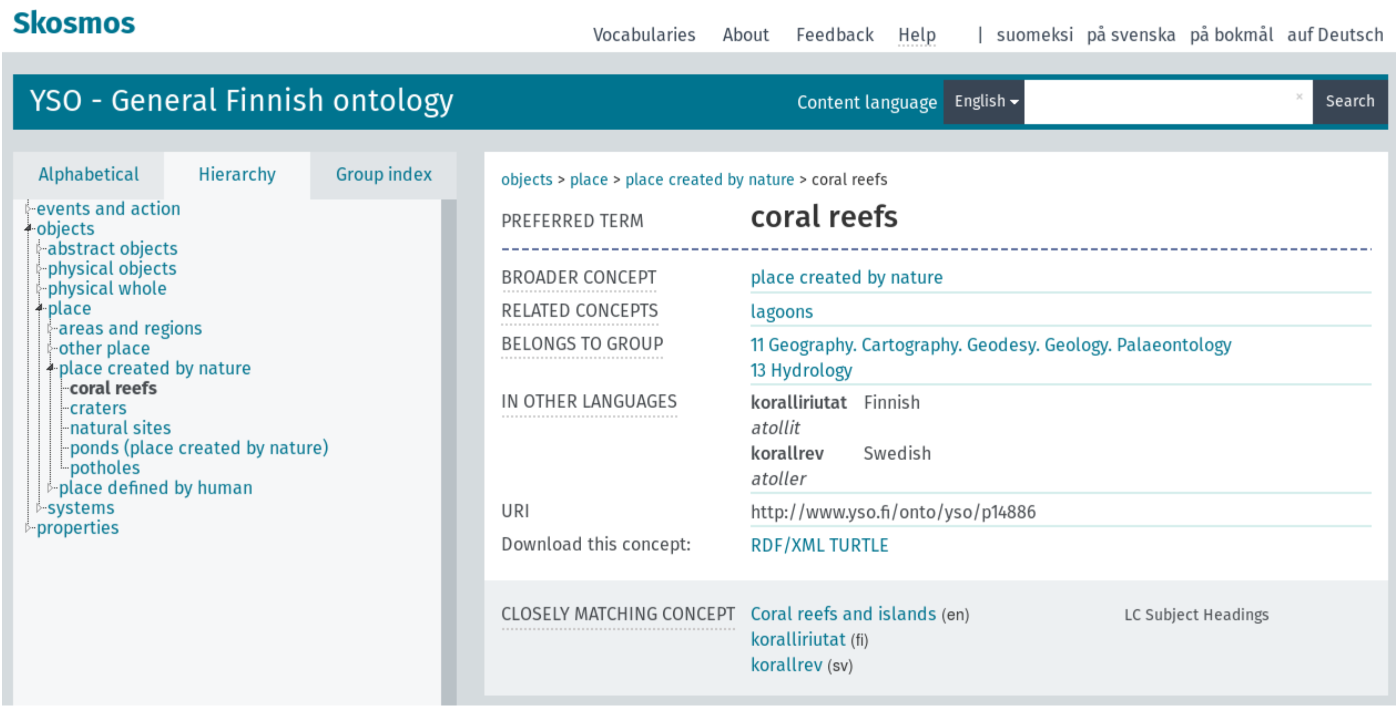
4.3. Vocabulary publishing process

Publishing a SKOS vocabulary with Skosmos involves three steps. First, the vocabulary should be represented as a SKOS dataset. This may involve converting the vocabulary from a legacy format and checking the quality of the resulting data using tools such as Skosify and qSKOS (Suominen & Mader, 2014). Second, the vocabulary is stored in a triple store such as Apache Jena Fuseki, using facilities provided by the store. Third, Skosmos is configured for the vocabulary by editing a small configuration file in Turtle syntax. The file is used to configure details such as the vocabulary name, the SPARQL endpoint, and vocabulary-specific options such as supported languages, hierarchy display options and the use of SKOS collections.

After these steps the vocabulary becomes immediately available for searching, browsing and API access. Updating the vocabulary can be performed simply by updating the data in the triple store and clearing the HTTP cache, if such a cache has been set up.

[13] https://github.com/NatLibFi/Skosify

4.4. User interface



objects > place > place created by nature > coral reefs

PREFERRED TERM coral reefs

BROADER CONCEPT place created by nature

RELATED CONCEPTS lagoons

BELONGS TO GROUP 11 Geography. Cartography. Geodesy. Geology. Palaeontology
13 Hydrology

IN OTHER LANGUAGES koralliriutat Finnish
atollit
korallrev Swedish
atoller

URI http://www.yso.fi/onto/yso/p14886

Download this concept: RDF/XML TURTLE

CLOSELY MATCHING CONCEPT Coral reefs and islands (en) LC Subject Headings
koralliriutat (fi)
korallrev (sv)

Figure 2: Skosmos user interface (concept page).

Skosmos provides a multilingual user interface for browsing vocabularies (Figure 2). Currently supported user interface languages are English, Finnish, German, Norwegian, and Swedish. However, vocabularies in any language can be searched, browsed and visualized as long as proper language tags for labels and documentation properties have been provided in the data.

All instances of skos:Concept are displayed as concepts in Skosmos. Each concept will have its own page and concepts can be searched for via their labels. Concept information, including labels, semantic relations and documentary notes, as well as any Dublin Core metadata such as timestamps, will be displayed on the concept page. There is special support for formatted collections of narrower concepts such as “milk by source animal”, i.e. arrays of concepts, modelled in ISO 25964 as a subclass of skos:Collection.

Custom types of concepts, i.e. subclasses of skos:Concept, can be defined in RDF data. The custom type will be displayed by Skosmos and searches can be restricted by type. Another way of classifying concepts is to assign a concept to one or more concept groups. Skosmos supports the ConceptGroup type, defined in ISO 25964 as a subclass of skos:Collection, which can be used to create a thematic or domain-oriented classification independent of the main concept hierarchy. Skosmos provides a separate group index which shows groups and their member concepts. Searches can also be restricted by group.

Skosmos recognizes all SKOS mapping relationships and will show them in a separate section on the concept page (see the “Coral reefs and islands” LCSH mapping in Figure 2). Labels for concepts in external vocabularies will be looked up using the Linked Data follow-your-nose principle, i.e. a HTTP request for the concept URI is performed and the concept labels are determined from the returned RDF data.

4.5. Usability tests

Skosmos has undergone three usability test rounds, results of which have pinpointed problems and given direction to the development. Altogether the usability of Skosmos has been formally tested with 12 test subjects: 8 indexers and 4 non-indexers.

In the usability tests, instead of small strict tasks, the users were instructed to find suitable index terms for a book handed to them, using the user interface of Skosmos. Such a broad and open-ended task enabled the users to use Skosmos in their natural way of browsing a vocabulary. Video recordings of the tests were subsequently analyzed to spot bugs and potential enhancements. Additionally, the testers were asked to evaluate the subjective usability of Skosmos with a System Usability Scale questionnaire (Brooke, 1996). The average SUS score was 79 (on a scale of 0 to 100, higher score means better usability), which can be considered a good result. The SUS score has varied very little between the test rounds, so the subjective usability of Skosmos has not seen major changes. In addition to usability testing, we have used the Contextual Inquiry method (Beyer & Holtzblatt, 1995) to gain a better view of the work of indexers.

The usability tests have revealed that users use a variety of strategies to find suitable concepts. When an autocomplete feature is provided, users will often use it to search for many terms that they consider potentially relevant, for example based on the title, cover text and table of contents of a book they are indexing. Repeated searching is much more common than browsing vocabularies, particularly when users feel time-constrained.

During the latest, 3rd, testing round, no critical usability problems were found. Skosmos 1.0 was released[14] soon after completing the tests. Further usability tests will be conducted in the future with indexers and vocabulary developers as test subjects.

4.6. API access

Skosmos provides an easy to use REST API[15] for read only access to the vocabulary data. The return format is mostly JSON-LD, but some methods return RDF/XML, Turtle, RDF/JSON with the appropriate MIME type. These methods can be used to publish the vocabulary data as Linked Data. The API can also be used to integrate vocabularies into third party software. For example, the `search` method can be used to provide autocomplete support and the `lookup` method can be used to convert term references to concept URIs.

4.7. Development process

Skosmos is developed using an open source process. The development is coordinated as a GitHub project [16] where all interested parties can participate. The source code is available under the MIT license. Issues can be reported and code and translations can be, and have been, contributed through GitHub. The development of Skosmos follows a cyclic model, with new versions released several times per year.

5. Impact

Skosmos has already been deployed in several organizations. In this section we present the main installations of Skosmos that we are aware of.

5.1. Finto

The Finnish national thesaurus and ontology service Finto.fi (Lappalainen et al., 2014; Suominen et al., 2014) is built using Skosmos. At the National Library of Finland Finto serves as a publishing platform for the General Finnish Thesaurus YSA[17] and the General Finnish Ontology YSO[18], both used for indexing of

[14] https://lists.w3.org/Archives/Public/public-esw-thes/2015Feb/0012.html
[15] https://github.com/NatLibFi/Skosmos/wiki/REST-API
[16] https://github.com/NatLibFi/Skosmos
[17] http://finto.fi/ysa/en/
[18] http://finto.fi/yso/en/

documents. Many YSO-based domain ontologies have also been published in Finto, as well as the unified ontology cloud KOKO. Experiments have also been made in publishing authority records[19] as linked open data using the RDA data model together with SKOS.

At the National Library Finto is replacing old vocabulary browsers such as VESA[20] that were built with specific vocabularies in mind and provided only limited search functionalities. In the future Finto will be the centralized publishing platform for all controlled vocabularies developed at and used by the Library. Finto also permits a move towards a more modular cataloguing infrastructure: by storing master copies of the vocabularies in Finto and using its APIs for live queries, maintaining local copies of vocabularies in cataloguing system can be avoided. However, much work in this area is still to be done, and current data formats in use such as MARC limit the possibilities of gaining the full benefits of linked open data vocabularies.

Finto is quickly becoming one of the key components of the National Digital Library of Finland[21], which aims to bring together the digital collections from all major cultural organizations in Finland. Within this framework, Finto is currently being integrated into centralized cataloguing systems in the archive and museum sector. It is planned that in the future Finto will serve as the central hub for the shared authority files of all the national memory organizations in Finland.

Finto is also used by indexers at several public sector organizations including the Finnish broadcasting company Yle[22]. Finnish public libraries have also been interested in using Finto’s APIs in their cataloguing systems, and work in this area is underway.

In 2015 Finto is being connected to the Finnish National Data Exchange Layer[23] (Palveluväylä), developed by the Ministry of Finance. The Layer comprises of a set of practices for relaying information in a common way at a low level and provides the needed infrastructure. The development is of limited scope from a technical perspective since Finto is free and open to use by anyone and adheres to the open linked data principles, but the Layer connection is of significant symbolic value. It represents a stamp of approval that

[19] See the Finnish Corporate Names dataset that is based on the Corporate Names used in cataloguing the Finnish National Bibliography: http://finto.fi/cn/en/

[20] http://vesa.lib.helsinki.fi/index.html

[21] http://www.kdk.fi/en

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

[23] http://vm.fi/palveluvayla

Finto is officially a part of the service infrastructure of the public sector and is considered reliable and secure.

Even more importantly, Finto is set to play an important role in the Finnish public sector plans towards semantic interoperability. The Data Exchange Layer solves some of the challenges of integrating various systems and services at a low level. But in order to enable true integration, a layer of semantic interoperability is needed and it is at this level that Finto can be seen as being one important piece in the larger puzzle. A common repository and publication channel for various vocabularies used for annotations is seen as an integral step towards a deeper interoperability between systems in the public sector.

5.2. FAO / AGROVOC

Skosmos is also used by the Food and Agriculture Organization of the UN to provide a search interface to the AGROVOC multilingual agricultural thesaurus[24] (Caracciolo et al., 2013). With the move of AGROVOC to the SKOS/RDF model, a new search and browsing interface was needed, to make possible the update of the AGROVOC web pages. At the time, no browsing and searching tool for RDF/SKOS was publicly available and some effort was put into developing a dedicated tool. Meanwhile, Skosmos appeared, and after some testing it was adopted, mainly because of the very intuitive visualization of concept hierarchies and concept details, and its good performances in terms of speed of search and retrieval of AGROVOC content.

A few issues arose when integrating Skosmos in the already existing infrastructure for AGROVOC. The first point concerns the compatibility of PHP applications and Drupal websites, such as the AGROVOC website. Skosmos was first embedded in the AGROVOC website by means of an IFRAME element, but this choice had adverse effects on the usability of the embedded page, as Skosmos was not designed to be used this way. Then, Skosmos was used as an application on its own, linked from the AGROVOC website. This solution turned out to be rather convenient, and it led us to formulate a more complete set of requirements for fully supporting the publication of multilingual vocabularies in the web. First of all, at the interface level, we experienced a mismatch between the six “FAO languages” of the AGROVOC website (Arabic, Chinese, English, French, Russian and Spanish), and the available user interface languages of Skosmos. This

[24] http://aims.fao.org/agrovoc

prevents users of non-English pages from having a smooth navigation experience, in that after having searched Agrovoc in Skosmos, one may only go back to the English pages of AGROVOC, independently of which language they were browsing. Other issues related to the support of multilinguality concern the actual display of AGROVOC content. In fact, AGROVOC consists of a single hierarchy of some 32,000+ concepts, each available in up to 23 languages. Currently, the hierarchical navigation structure of Skosmos is based on English preferred labels, though in recent versions of Skosmos any language could be used.

Another issues that we experienced concerned the backend infrastructure. AGROVOC is maintained using VocBench, a web-based tool supporting multilinguality and and concurrent uses (Stellato et al., 2015). VocBench may use various triple stores for storing data, but for the case of AGROVOC, OWLIM performs best and is therefore the most convenient. To the public, AGROVOC is exposed by means of the SPARQL endpoint of Allegrograph. In an ideal integration scenario, Skosmos would have been accommodated in the picture, but this was not possible, mainly because text index support was only available for Fuseki.

A different line of issues arose concerning customization. In particular, we found that the entry page to Skosmos is not suitable for a resource such as AGROVOC. The Skosmos entry page displays a number of statistics computed on the vocabulary, such as the number of concepts, the number of labels in various languages etc., but given the size of AGROVOC, the loading of this page required too much time. We disabled the calculation of these statistics, resulting however in sub-optimal visualization of the entry page.

In summary, Skosmos was adopted by AGROVOC as a search & browsing tool. Its adoption benefitted the publication of AGROVOC on the web, in that it offered a very intuitive search and browse interface for the casual user and, despite the duplication of infrastructure (three different triple stores for the same project), it still spared FAO from the burden of converting AGROVOC to the older formats that were required by the previous web interface. Not all the information in AGROVOC is currently displayed by Skosmos, partly because SKO-XL is not supported by Skosmos and also because work on standardization of AGROVOC content is ongoing.

5.3. Rhineland-Palatinate

The Rhineland-Palatinate spatial data infrastructure initiative[25] in Germany is using Skosmos to publish[26] habitat type classifications (codelists) which were previously only managed as Excel sheets. The provision of these classifications in the form of Linked Open Data allows local German agencies to link their classifications to the European Nature Information System[27] (EUNIS) of the European Environment Agency (EEA). As many other information of the European Union, the classifications of species, habitats and protected sites are already published according to the RDF model. A further idea is to use the Linked Open Data resources, which are published via Skosmos, for the acquisition of spatial data. In future, not the codes from the former codelists, but the URIs of the corresponding concepts should be stored as feature attributes.

Two other use cases, which are currently tested in Rhineland-Palatinate, concern the classification schemata for administrative units (ca. 2400 concepts) and the governmental file plan (ca. 20500 concepts). Both schemata were only available in form of Excel sheets. To publish them with Skosmos, they were exported to CSV and then transformed to SKOS by using simple PHP scripts. The hierarchical structure had to be decompiled from the identifiers, because it was not directly modelled in the Excel representation.

Both data sets, the file plan and the classification of administrative units, are needed in many governmental processes and none of them was available in an interoperable and referenceable way before publishing them as Linked Open Data. There is huge potential for increasing governmental efficiency by opening closed information to the Semantic Web. In 2014 the European Commission has adopted a proposal for establishing a programme on interoperability solutions (ISA2[28]). It covers the period from 2016 to 2020, has a financial envelope of €131 million and may be used to promote the publishing of governmental data in form of Linked Open Data.

[25] http://www.geoportal.rlp.de/portal/en/information/about-us.html
[26] http://www.geoportal.rlp.de/skosmos/
[27] http://eunis.eea.europa.eu
[28] http://ec.europa.eu/isa/isa2/index_en.htm

5.4. Other users

There are pilot installations of Skosmos also at other institutions around the world. The University of Oslo Library are using Skosmos to publish[29] controlled vocabularies including Realfagstermer (a multilingual subject heading list covering natural sciences, mathematics and informatics), Humord (a thesaurus covering humanities and social sciences) and Tekord (a thesaurus mainly covering engineering and science). Compared to their current publishing solutions, a great benefit of Skosmos is that it enables both searching all the vocabularies simultaneously and searching a single vocabulary at a time. It also provides better navigation of the hierarchies, and shows all hierarchical paths as breadcrumbs for concepts belonging to multiple hierarchies. If Skosmos is to be used by end users, however, links out to the library catalogue will have to be added.

The Global Agricultural Concept Scheme (GACS) project, which is an initiative by FAO, CAB International[30] and the National Agricultural Library[31] of the US to create a common agricultural concept scheme, is using Skosmos as a publication platform for agricultural thesauri and concept mappings.

6. Conclusions and future work

Controlled vocabularies can be published on the web as Linked Data using the SKOS model and general purpose Linked Data publishing tools, but such publishing has limited usefulness for users who benefit more from search and browsing. Separate web applications are typically developed for browsing vocabularies, causing duplication of infrastructure and effort. Skosmos is a tool that can both support human browsers in browsing and searching a SKOS vocabulary, and provide Linked Data access to the underlying data together with developer-friendly APIs that support term-based searches.

In the future, we would like to develop better support for extensions, including support for SKOS XL and more aspects of ISO 25964 than are currently supported in Skosmos. Providing more opportunities for local customization, for example linking concept pages to local library catalogs with works described using those concepts, would benefit users and help make library data more linked, accessible and visible. As a practical

[29] http://app.uio.no/ub/emnesok/skosmos/
[30] http://www.cabi.org/
[31] http://www.nal.usda.gov/

matter, adding translations for more languages and other text index implementations than jena-text would help integrating Skosmos into different kinds of websites and infrastructure.

Acknowledgements

We thank Eero Hyvönen for original design ideas for ONKI Light and Alex Johansson for participating in their implementation. Satu Niininen and Kim Viljanen provided very insightful comments on draft versions of this paper, for which we are very thankful. We also thank Dan Michael O. Heggø for participating in Skosmos development and describing the Norwegian application scenario for this paper. The development of Skosmos is performed in the context of the Finto project, funded by the Finnish Ministry of Finance and the Ministry of Education and Culture.

References

d'Aquin, M. & Noy, N.F. (2012). Where to Publish and Find Ontologies? A Survey of Ontology Libraries. *Web Semantics: Science, Services and Agents on the World Wide Web* 11, 96-111.

van Assem, M., Malaisé, V., Miles, A., Schreiber, G. (2006). A Method to Convert Thesauri to SKOS. *Proceedings of the 3rd European Semantic Web Conference (ESWC 2006)*, vol. 4011 of Lecture Notes in Computer Science, Springer, 95-109.

Baker, T., Bechhofer, S., Isaac, A., Miles, A., Schreiber, G., & Summers, E. (2013). Key choices in the design of Simple Knowledge Organization System (SKOS). *Web Semantics: Science, Services and Agents on the World Wide Web*, *20*, 35-49.

Bandholtz, T., Schulte-Coerne, T., Glaser, R., Fock, J., Keller, T. (2010). iQvoc - Open Source SKOS(XL) Maintenance and Publishing Tool. In *Proceedings of the Sixth Workshop on Scripting and Development for the Semantic Web (SFSW 2010).*

Bandholtz, T., Fock, J., Wolff, A., & Schentz, H. (2011). LOD-ready Environmental Terminology with iQvoc. In *Proceedings of EnviroInfo Ispra 2011. Innovations in Sharing - Environmental Observations and Information (part 1)*, pp. 343-352. Shaker Verlag, Aachen.

Beyer, H.R., & Holtzblatt, K. (1995). Apprenticing with the customer. *Communications of the ACM*, *38*(5), 45-52.

Binding, C., Tudhope, D. (2010). Terminology Web Services. *Knowledge Organization* 37(4), 287-298.

Brooke, J. (1996). SUS: a 'quick and dirty' usability scale. In *Usability Evaluation in Industry*, pp. 189–194. Taylor & Francis, London.

Caracciolo, C., Stellato, A., Morshed, A., Johannsen, G., Rajbhandari, S., Jaques, Y., & Keizer, J. (2013). The AGROVOC Linked Dataset. *Semantic Web*, *4*(3), 341-348.

Frosterus, M., Tuominen, J., Pessala, S., Seppala, K., Hyvönen, E. (2013). Linked Open Ontology Cloud KOKO—Managing a System of Cross-domain Lightweight Ontologies. In *Proceedings of the 10th Extended Semantic Web Conference (ESWC 2013), Satellite Events*, vol. 7955 of Lecture Notes in Computer Science. Springer, 2013, 296-297.

Golub, K., Tudhope, D., Zeng, M.L., Žumer, M. (2014). Terminology registries for knowledge organization systems: Functionality, use, and attributes. *Journal of the Association for Information Science and Technology,* 65(9), 1901-1916.

Greenberg, J., Losee, R., Pérez Agüera, J.R., Scherle, R., White, H., Willis, C. (2011). HIVE: Helping Interdisciplinary Vocabulary Engineering. *Bulletin of the American Society for Information Science and Technology* 37(4), 23-26.

ISO (2011). *ISO 25964-1:2011 Information and documentation - Thesauri and interoperability with other vocabularies. Part 1: Thesauri for information retrieval.* International Organization for Standardization, Geneva, Switzerland.

Lange, C, Ion, P., Dimou, A., Bratsas, C., Sperber, W., Kohlhase, M., Antoniou, I. (2012). Bringing Mathematics to the Web of Data: The Case of the Mathematics Subject Classification. In *Proceedings of the 9th Extended Semantic Web Conference (ESWC 2012)*, vol. 7295 of Lecture Notes in Computer Science. Springer, 2012, 763-777.

Lappalainen, M., Frosterus, M., Nykyri, S. (2014). Reuse of library thesaurus data as ontologies for the public sector. In *Proceedings of IFLA WLIC 2014, 16-22 August 2014, Lyon, France.*

Medelyan, O. (2009). *Human-competitive automatic topic indexing*. Doctoral dissertation, The University of Waikato.

van der Meij, L., Isaac, A., Zinn, C. (2010). A Web-Based Repository Service for Vocabularies and Alignments in the Cultural Heritage Domain. In *Proceedings of the 7th Extended Semantic Web Conference (ESWC 2010)*, vol. 6088 of Lecture Notes in Computer Science. Springer, 2010, 394-409.

Miles, A., & Bechhofer, S. (2009). SKOS Simple Knowledge Organization System reference. *W3C recommendation, http://www.w3. org/TR/skos-reference*

Neubert, J. (2009). Bringing the “Thesaurus for Economics” on to the Web of Linked Data. In *Proceedings of the WWW Workshop on Linked Data on the Web (LDOW 2009).*

Speicher, S., Arwe, J., & Malhotra, A. (2015). Linked Data Platform 1.0. *W3C recommendation, http://www.w3.org/TR/ldp/*

Stellato, A., Rajbhandari, S., Turbati, A., Fiorelli, M., Caracciolo, C., Lorenzetti, T., Keizer, J., Pazienza, M.T. VocBench: a Web Application for Collaborative Development of Multilingual Thesauri. In *Proceedings of the 12th Extended Semantic Web Conference (ESWC 2015)*, vol. 9088 of Lecture Notes in Computer Science. Springer, 2015, 38-53.

Summers, E., Isaac, A., Redding, C., Krec, D. (2008). LCSH, SKOS and Linked Data. In *Proceedings of the 2008 International Conference on Dublin Core and Metadata Applications (DC-2008).* Dublin Core Metadata Initiative, 25-33.

Suominen, O., Johansson, A., Ylikotila, H., Tuominen, J., Hyvönen, E. (2012). Vocabulary services based on SPARQL endpoints: ONKI Light on SPARQL. In *Poster proceedings of the 18th International Conference on Knowledge Engineering and Knowledge Management (EKAW 2012).*

Suominen, O., & Mader, C. (2014). Assessing and improving the quality of SKOS vocabularies. *Journal on Data Semantics*, *3*(1), 47-73.

Suominen, O., Pessala, S., Tuominen, J., Lappalainen, M., Nykyri, S., Ylikotila, H., Frosterus, M. & Hyvönen, E. (2014). Deploying National Ontology Services: From ONKI to Finto. In *Proceedings of the ISWC 2014 Industry track*.

Tuominen, J., Frosterus, M., Viljanen, K., Hyvönen, E. (2009). ONKI SKOS Server for Publishing and Utilizing SKOS Vocabularies and Ontologies as Services. In *Proceedings of the 6th European Semantic Web Conference (ESWC 2009)*, vol. 5554 of Lecture Notes in Computer Science. Springer, 2009, 768-780.

Viljanen, K., Tuominen, J., & Hyvönen, E. (2008, June). Publishing and using ontologies as mashup services. In *Proceedings of the 4th Workshop on Scripting for the Semantic Web (SFSW2008), 5th European Semantic Web Conference* (Vol. 2008).

Viljanen, K., Tuominen, J., Hyvönen, E. (2009). Ontology libraries for production use: The Finnish ontology library service ONKI. In *Proceedings of the 6th European Semantic Web Conference (ESWC 2009)*, vol. 5554 of Lecture Notes in Computer Science. Springer, 2009, 781-795.

White, H., Willis, C., Greenberg, J. (2013). HIVEing: the effect of a semantic web technology on inter-indexer consistency. *Journal of Documentation,* 70(3), 307-329.

Zapilko, B., Schaible, J., Mayr, P., & Mathiak, B. (2013). TheSoz: A SKOS representation of the thesaurus for the social sciences. *Semantic Web*, *4*(3), 257-263.