A Web-Based Repository Service for Vocabularies and Alignments in the Cultural Heritage Domain

Lourens van der Meij[1,2], Antoine Isaac[1,2], and Claus Zinn[3]

[1] Vrije Universiteit Amsterdam
[2] Koninklijke Bibliotheek, Den Haag
{lourens,aisaac}@few.vu.nl
[3] Max Planck Institute for Psycholinguistics, Nijmegen
claus.zinn@mpi.nl

Abstract. Controlled vocabularies of various kinds (*e.g.*, thesauri, classification schemes) play an integral part in making Cultural Heritage collections accessible. The various institutions participating in the Dutch CATCH programme maintain and make use of a rich and diverse set of vocabularies. This makes it hard to provide a *uniform* point of access to all collections at once. Our SKOS-based vocabulary and alignment repository aims at providing technology for managing the various vocabularies, and for exploiting semantic alignments across any two of them. The repository system exposes web services that effectively support the construction of tools for searching and browsing across vocabularies and collections or for collection curation (indexing), as we demonstrate.

1 Introduction

Cultural Heritage (CH) collections are typically indexed with metadata derived from a range of different vocabularies or Knowledge Organization Systems (KOS, *e.g.*, thesauri, classification schemes, subject lists), such as the Art & Architecture Thesaurus (see http://www.getty.edu/research/conducting_research/vocabularies/aat/), Iconclass (http://www.iconclass.nl/), but also in-house standards. This makes it hard to facilitate uniform access to multiple collections in a semantically interoperable way. The aim to unify the main CH vocabularies into a standard, commonly-accepted vocabulary to use for all—and to migrate all metadata to such a new, overarching standard—is deemed unrealistic. Vocabularies evolved over many years, and will so in the future; also, there are good reasons for domain-, collection-, or institution-specific organisations of CH objects. A *vocabulary matching* approach acknowledges this, and aims at mapping together those concepts of any two given vocabularies that are semantically related to each other. Such vocabulary alignments can then be exploited to facilitate access to multiple collections via the vocabulary of a single one.

L. Aroyo et al. (Eds.): ESWC 2010, Part I, LNCS 6088, pp. 394–409, 2010.

The STITCH project (http://stitch.cs.vu.nl) of the Dutch CATCH programme (http://www.nwo.nl/CATCH) and the European TELplus project (www.theeuropeanlibrary.org/telplus/) investigated methods to support metadata interoperability by automatically identifying inter-vocabulary semantic mappings [1,2]. They showed that the automatic matching of vocabularies can be the basis of various real-world usage scenarios, including support for indexing and re-indexing collection items, inter-collection search and navigation, but also thesaurus management [3,4]. First tools for vocabulary services have been deployed at the National Library of the Netherlands, so that there is indeed an industrial uptake of Semantic Web technology in this context.

Our projects have dealt with a good number of industrial-strength, real-world vocabularies; they have used third party tools, but also developed in-house prototypes to align those vocabularies. A considerable effort was needed to convert the vocabularies' format to make them satisfy the input requirements of the various matching tools. Moreover, the output of some matching tools lacked precise definitions, so that higher-level tools (say, to support indexing) had to rely on interpretations of produced mappings, especially with respect to the *type* of mapping relation used. To support various applications that exploit (or contribute in creating) networks of vocabularies, we felt the need to adopt standardized middleware-level repository services, which we have subsequently developed and which is reported herein. Our middleware for managing vocabularies and their mappings is explicitly targeted at CH technologists, acknowledging the fact that the KOSs of CH institutions, and the application contexts where they are used, share many common features as well as a core of data management requirements.

Research on repository services for Semantic Web ontologies is very active. We focused on selecting and combining those elements from existing APIs and repositories that fit our application scenarios best. Rather than providing a framework for dealing with ontologies in general, we aim at technology to tackle the CH applications at hand. For this, we marry a simpler vocabulary modelling approach with results from the ontology alignment community, taking into account CH issues such as data distribution, scalability and maintenance. We have thus chosen to build our vocabulary and alignment repository services on the basis of a SKOS-based format for vocabularies to give unified, effective and fast access to vocabularies and vocabulary alignments, or their parts. Our semantic middleware is implemented as a distributed web-based architecture. Higher-level tools can now use the services, for instance, to look-up a concept within a vocabulary, to identify the vocabularies where a given concept occurs, to get all related concepts for a given concept within a vocabulary, or across vocabularies by exploiting concept mappings.

In the remainder of the paper, we discuss the repository services in greater detail. Sect. 2 discusses use cases, requirements, and other background, while Sect. 3 introduces data model and design, and lists the main services. Sect. 4 describes the current state of our middleware, and three demonstrator systems we have built with it; Sect. 5 concludes the paper with a discussion.

2 Use Cases, Requirements and Background

While the design of our repository services takes into account general needs of vocabulary experts, its current implementation responds to direct requirements of the STITCH project and its umbrella programme CATCH [4].

2.1 Use Cases and User Requirements

Indexing support. Support CH staff in finding a concept in a given vocabulary, for instance, through term search, including auto-completion, or vocabulary navigation (browsing concept hierarchies). Similarly, return other information attached to concepts such as preferred or alternative labels (possibly across languages), scope notes, or the semantic relationship of the concept in question to other concepts (following broader-than, narrower-than or related-to links). CH staff can then use the result of concept search to annotate literary works, for instance, with appropriate concept labels.

Semantic search and browsing. Support expert and novice users in performing semantic search across multiple collections, e.g., exploiting the object-concept links established by indexing staff. Where a search query returns insufficient hits, replace (some of) its search terms by others that are semantically broader than the given ones. Where CH items of interests are described by concepts of a different vocabulary, replace user-given search terms with equivalent or related terms of the other vocabulary (for the collections and their vocabularies at hand). Moreover, give users browsing access across any two collections using the vocabulary of one or the other, and an alignment between them.

Uniform vocabulary format and storage. Convert the vocabularies of a given CH institution into an electronic and uniform format, and store resulting data in a vocabulary repository; adapt existing CH software to access the new format (supporting the import/export of vocabulary from/to the repository).

Alignment management. Store an alignment between two given vocabularies in a uniform format, and allow users to browse alignment data, add new mappings, and to remove or modify existing mappings. Give access to individual mappings to support the attachment of evaluation marks. Provide support for the combination of existing alignments into a new one, or for selecting an alignment's subset given, say, some confidence threshold. Facilitate the testing of automatic alignment techniques by providing support to compare computed alignments with gold standard reference alignments.

Technical requirements. The vocabulary server should store all data in a commonly accepted format, preferably SKOS, the proposed W3C standard for porting Knowledge Organization Systems on the Semantic Web [5]. It ought therefore to support most of SKOS' constructs such as labels, semantic relations, and documentation aspects. API service functionality should meet the requirements

stemming from the use cases and meet community best practices. Also, the vocabularies shall be restorable from their RDF sources. Moreover, the architecture shall provide authorization methods to facilitate the implementation of various access levels (*e.g.*, to support the updating or versioning of thesaurus data).

An architecture that builds upon a simple "standard" RDF repository is clearly not sufficient to satisfy all requirements at once. SPARQL, for instance, does neither support full text search in labels without appropriate extension nor the advanced sorting of results, which is often required for, *e.g.*, user interaction. It is also hard to constrain SPARQL queries to data subsets (restricted access), or guarantee fast response time when allowing arbitrary rather than optimized SPARQL queries. Consequently, our architecture cannot simply be a front-end to an RDF repository containing uncontrolled RDF triples. Neverthess, it makes use of RDF repositories, but all access is via a simple but standardized interface to (a part of) SKOS.

2.2 Background

There is an increased interest in services and technology around KOS resources. A recent JISC report reviews the state of the art in this area, in particular, with regard to vocabulary types, indicative use cases, best practise guidelines and current research [6]. Services and/or APIs such as [7,8,9,10,11,12,13,14] offer the functionality for accessing common vocabulary features, and are mostly compatible with KOS standards like SKOS, both for serving/exporting or ingesting data.

Following the spirit of the SKOS standard, KOS alignments are expressed in those tools by simple RDF triple statements. A more expressive representation is required, however, to meet realistic application requirements such as the ones we encountered in CATCH [4]. Here, mappings need to have properties of their own, for instance, "confidence measure", "producer", or "evaluated as". This issue is only acknowledged by [10,14]; and the approach of [8] is more motivated by the *provenance* problem rather than guided by the need to serving representations specifically adequate to alignment management.

The design of our middleware is inspired by the influential work of Euzenat [15] for the Ontology Alignment Evaluation Initiative (OAEI, see `http://oaei.ontologymatching.org`), who proposes a representation for alignments that is compatible with KOS (and SKOS) practices. To the best of our knowledge, our middleware is the only one that implements the frameworks introduced in both the KOS and the ontology matching community.

It is important to notice that the design of our middleware was purely driven by practical end-user and application requirements in the context of the STITCH project (and the larger CATCH umbrella). Our services thus cater for CH practitioners who expect computer-supported means for managing simple KOSs following well-established CH usages or workflows.

While ontology portals and services such as NeON [16] or BioPortal [17] also make use of the OAEI framework, they seem to have a more generic approach by supporting fully-fledged formal ontologies; in comparison, we are aiming at

a different level of complexity, lowering the barrier for tool use in practise. This emphasis on simplicity also holds in comparison with other simple KOS-based services such as the aforementioned ones. Rather than providing users with advanced functionality such as calculating semantic proximity between two concepts based on the structure of KOSs [7], we deliberately focused on infrastructural work around core functionality, upon which one can later build more application-specific tools.

3 Service Description

In this section, we describe the data modelling approaches we followed, the overall design of the repository's architecture, and the various vocabulary and alignment services that our middleware provides.

3.1 Data Model

The data model builds upon SKOS and the alignment exchange format used in the Ontology Matching community [15].

SKOS. SKOS represents the elements of a KOS as *concepts*, provides them with various kinds of labels and documentation notes, and allows linking them together by three types of semantic relations, namely *broader*, *narrower* and *related*. SKOS represents entire KOSs as *concept scheme* objects, with explicit references to the concepts they contain. SKOS *collections* represent meaningful groupings of concepts within a KOS such as "persons by age." SKOS also provides a mechanism for the creation of RDF data. For instance, concepts are represented as resources with the `skos:Concept` class as type, and the property `skos:prefLabel` is used to indicate a concept's preferred lexicalization.

Our service supports the manipulation of the SKOS RDF constructs for data representation and exchange. The only exceptions are collection-related constructs, since such groupings of concepts were not observed in the thesauri at hand, and in many other KOSs.[1]

Unfortunately, SKOS lacks the ability to represent *concept subschemes* that belong semantically or functionally to a *concept scheme*. The GTT thesaurus used at the National Library of the Netherlands, for instance, is divided into 8 sub-vocabularies: "general subjects", "places", "genres" *etc.* We have therefore introduced the notion of *concept scheme groups* to represent the inclusion of KOSs into larger sets that can be used as concept schemes themselves.

OAEI format. OAEI represents individual mappings between concepts as *mapping cells* that indicate the type of relation that holds between these concepts, as well as a confidence measure. In addition, a cell may contain other metadata, for instance, users' assessment or evaluation of the mapping. OAEI also provides

[1] See the "vocabulary usage" section of the SKOS implementation report, `http://www.w3.org/2006/07/SWD/SKOS/reference/20090315/implementation.html`

explicit representations at the *alignment level*, allowing users to manipulate (*e.g.*, for composing alignments) or evaluate the result of matching efforts on a group basis by attaching metadata to the *alignment* resource itself.

When using the OAEI data model to represent mapping cells, we explicitly reuse the SKOS relations *exactMatch*, *closeMatch*, *broadMatch*, *narrowMatch* and *relatedMatch* as relation types. This illustrates the complementarity between the two models: SKOS does not support the annotations of mappings as we need it; on the other hand, OAEI does not make any commitment with regards to the semantic type of relations that mappings assert. SKOS mapping properties support the appropriate types to fill this gap.

3.2 Design

Fig. 1 depicts the interaction between the components of the vocabulary and alignment services. It considers an example case where alignment data comes from one (local) store, while vocabulary data is accessed from two data stores (one local store and one remote store).

Our vocabulary and alignment services, which have been implemented using Java, provide an abstract data access layer on top of existing RDF repositories. Implementations of the interface `VocabularyAndAlignmentAccess` support, for instance, the plug-in of SPARQL endpoints, and connections to different versions of local or remote RDF stores such as Sesame (see `http://openrdf.org`) via their API. The services can be accessed over SOAP (using the Apache Axis implementation, see `http://ws.apache.org/axis/`), or locally. Local access is being used by dedicated servlets that provide a simple HTTP REST-like access (see Sect. 4.2).

Crucially, a repository service instance can be connected to several vocabulary data sources at the same time, each of them providing its own set of vocabularies. Instances of our `SKOSAccess` interface for vocabularies may indeed connect to a single RDF source or to multiple other instances wrapped around different sources—as for the `SKOSMultiConnection` in Fig. 1. For this, instances of `SKOSAccess` make public the vocabularies or alignments they contain, which allows a repository service aggregating them to distribute data queries to the various sources available, and merge the obtained results. This allows our architecture to easily scale up to dozens of vocabularies. It also makes it easier to maintain updates of vocabularies, or authorization mechanisms, as the owners of vocabularies can choose to implement and control their own components among the sources of a central repository.

We have chosen a similar distributed approach for vocabulary and alignment data. KOS and alignment providers (or consumers) may indeed come from different institutions. Some actors may publish vocabularies without seeking to align them to external sources, while others would establish alignments between vocabularies they do not own. Whenever possible, we thus packaged the vocabulary and alignment functions into two distinct service specifications. This allows interested implementors to focus on just one dimension, while still fitting the wider picture.

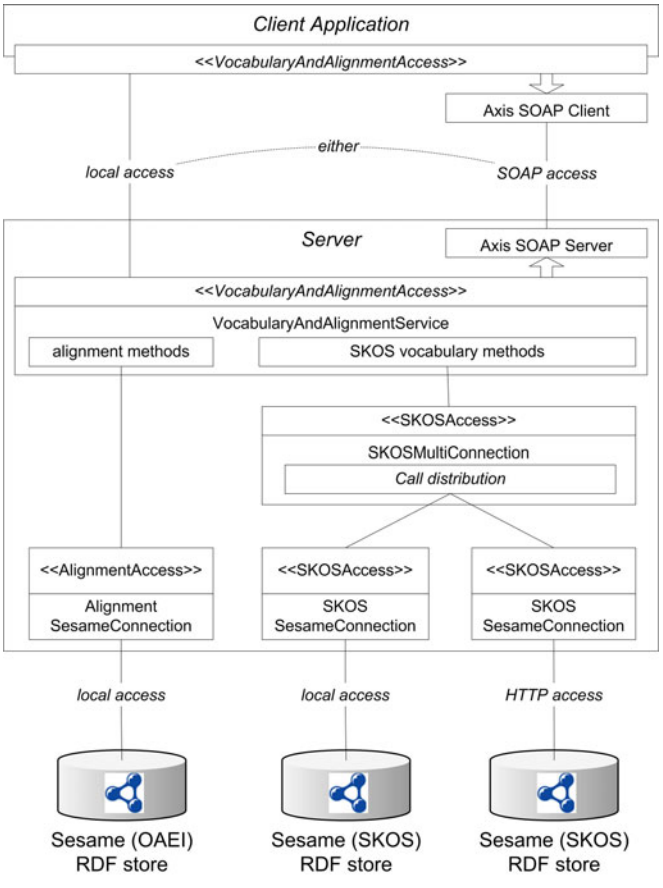


Fig. 1. Service architecture—angle brackets indicate the main interfaces implemented

3.3 Web Services Overview

Vocabularies. Entire KOSs can be imported to and exported from a repository. As required by CH institutions, the import function keeps track of the original information sources, so that the export function is capable of returning the exact copy of the original import (enabling lossless round-trips), leaving out potential later KOS enrichments (*e.g.*, when inference added new statements to a KOS' content).

The services give access to the *metadata* of the concept schemes and their groupings. There is functionality to search for schemes whose labels match a given string. Search can be restricted by specifying the type of label, or a label's language. Search results can be sorted or filtered given user-defined ranges, and are returned either as URI references, structured descriptions (*i.e.*, Java objects)

or simple URI/label couples. Knowing a KOS' URI, there are methods to access either all of the KOS' concepts or only its "top" elements.

Access to KOS data (concepts, relations) is centered on concepts, which can be searched based on their (different types of) labels and notes, with similar options for sorting, filtering and output. There is also a method that returns all concepts which are semantically related to a given concept; here, standard SKOS relations are followed, but this can be easily extended to include other relation types of the KOS in question, if defined. Moreover, unlike traditional KOS approaches, in SKOS it is possible for a concept to belong to more than one concept scheme. Our implementation therefore allows users to specify the scheme to which the sought connected concepts shall belong, thus addressing a concept's potential multiple provenance.

The KOS services do not provide editing functionality. Here, we assume the existence and use of purpose-built tools that fit best each CH institute's existing workflow. Moreover, KOSs are more stable than alignments. Nonetheless, the versioning of KOS updates is an issue, which has not yet been dealt with satisfactorily within the SKOS community.

Alignments. Entire alignments can be imported and exported. All functionality for the creation and management of new alignments was inspired by Euzenat's Alignment API functions [15]. New alignments can result from subjecting existing alignments to *e.g.*, filtering, intersection or union operations, the implementation of which is left to future extensions. A plug-in mechanism supports the integration of automatic alignment methods, in particular, with support for accessing existing KOSs of the vocabulary repository.

There are also methods for comparing alignments. Such comparisons form the basis for evaluation use cases where automatically generated alignments are set against existing reference alignments. The result of such a comparison, which is by default specified as a structured object, can be extended to reflect specific needs—precision, recall, various f-measures for evaluation; overlap and inclusion for more neutral comparison purposes *etc.*

The management of an alignment's metadata follows the standard OAEI format, but there are provisions for custom annotations to fit specific needs. All metadata is searchable to find all alignments that link together any given two vocabularies.

The individual *mapping cells* of an alignment can also be accessed. There is support to iterate through its (indexed) cells, but also to search for cells that match a given combination of concepts, mapping relation type and confidence measure. This facilitates access to cells that are not given a URI (blank nodes). The output can be sorted according to (extensible) presentation strategies.

Alignments are editable. Individual cells can be added to or removed from alignments, or modified. Cell metadata can be accessed and edited in a flexible way, *e.g.*, to reflect specific annotations resulting from a mapping's manual assessment.

4 Current Status

4.1 Repository Instances

At Vrije Universiteit (VU), we have deployed a repository instance hosting SKOS versions of five (groups of) vocabularies: *Iconclass*, a classification to describe images, *RAMEAU*, the subject headings of the French National Library, *Brinkman*, a thesaurus used at the National Library of the Netherlands, the noun subset of *Wordnet* and *RACM Glas*, an archeology KOS to describe glass material. Due to licensing issues, we have also deployed a richer, privately accessible, instance with nine additional KOSs used in the CATCH context. In total, we converted 10 out of the 13 KOSs to SKOS in the STITCH and TELPlus projects; the creation of two others was a joint effort with other research teams. The non-public instance now contains more than 1,700,000 concepts; its underlying (in-memory) RDF stores consume 6 GB of memory. Access to the service and further details, including statistics, are given at `http://stitch.cs.vu.nl/repository`, also see Appendix.

At the time of writing, the two service instances also host 15 alignments between various vocabularies to amount to almost a million mapping cells. In part, they were produced using lexical (using the concepts' labels) or instance-based (using objects annotated with the concepts) matching techniques as investigated in STITCH and TELplus [1,2]; in part they were created manually in the context of other projects.

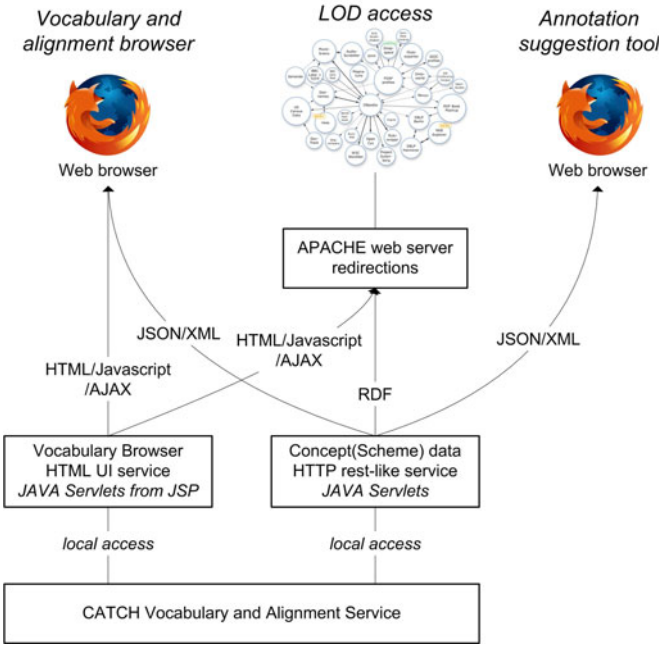


Fig. 2. Three deployments accessing the vocabulary and alignment services

4.2 Service Demonstrators

We have implemented three demonstrators, which are connected to our repository service instances as shown in Fig. 2.

Vocabulary and alignment browser. The *vocabulary and alignment browser*, shown in Fig. 3, consists of servlets that connect to a service instance to produce content (HTML, Javascript, AJAX) in response to browser-based user requests.

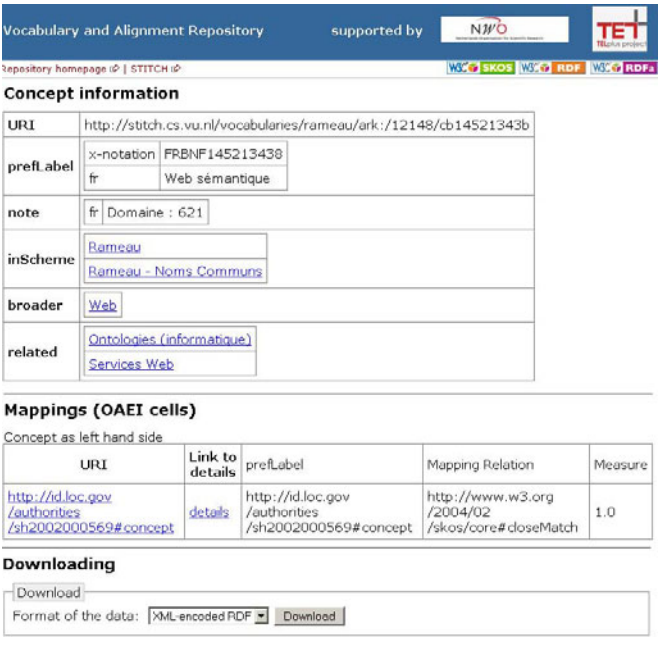
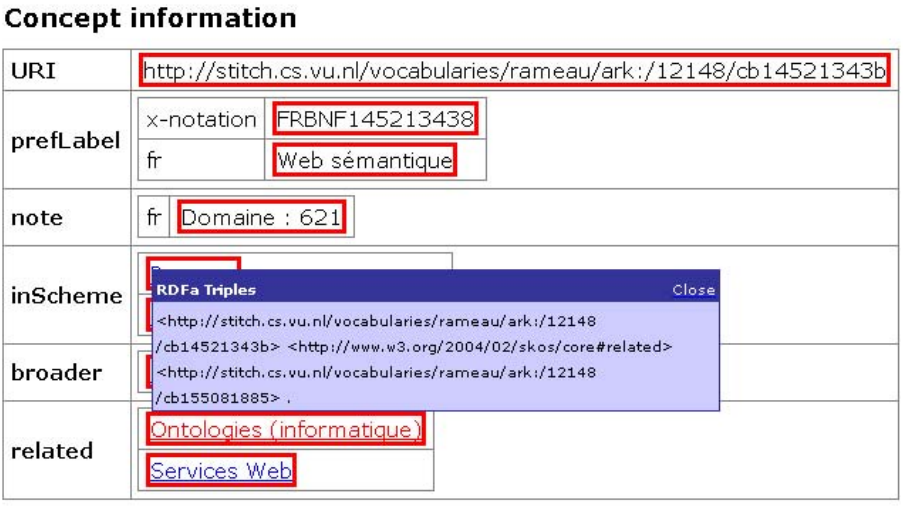


Fig. 3. Concept Information Service, HTML view

The demonstrator features an autocompletion function that helps users to search for concepts by partially typing concept labels. It supports the generation of RDF, JSON, and UI-oriented XML data for all elements viewed (see Fig. 4(b)). Also, RDFa markup [18] is included in all generated pages, see Fig. 4(a). Elementary data access functionality is implemented by a specific set of servlets, which then provide a HTTP REST-like access interface to services.

RAMEAU subject headings as linked data. In the TELplus project we converted the RAMEAU vocabulary of the French National Library to SKOS and ingested the result into our repository. We then implemented the recipes of [19] to have HTTP requests for RAMEAU concept URIs redirected so as to provide either HTML or RDF representation of these concepts, following the Linking Open Data (LOD) principles (see `http://linkeddata.org/`). This



(a) RDFa markup—highlight with RDFa Bookmarklet

```
<rdf:RDF>
  <skos:Concept rdf:about="http://stitch.cs.vu.nl/vocabularies/rameau/ark:/12148/cb14521343b">
    <skos:prefLabel xml:lang="x-notation">FRBNF145213438</skos:prefLabel>
    <skos:prefLabel xml:lang="fr">Web sémantique</skos:prefLabel>
    <skos:note xml:lang="fr">Domaine : 621</skos:note>
    <skos:inScheme rdf:resource="http://stitch.cs.vu.nl/vocabularies/rameau/autorites_matieres"/>
    <skos:inScheme rdf:resource="http://stitch.cs.vu.nl/vocabularies/rameau/noms_communs"/>
    <skos:broader rdf:resource="http://stitch.cs.vu.nl/vocabularies/rameau/ark:/12148/cb13319953j"/>
    <skos:related rdf:resource="http://stitch.cs.vu.nl/vocabularies/rameau/ark:/12148/cb144109034"/>
    <skos:related rdf:resource="http://stitch.cs.vu.nl/vocabularies/rameau/ark:/12148/cb155081885"/>
    <skos:related rdf:resource="http://stitch.cs.vu.nl/vocabularies/rameau/ark:/12148/cb155081885"/>
    <skos:related rdf:resource="http://stitch.cs.vu.nl/vocabularies/rameau/ark:/12148/cb144109034"/>
  </skos:Concept>
  <rdf:Description rdf:about="http://stitch.cs.vu.nl/alignments/macs/manual_rameau_lcsh">
    <align:map>
      <align:Cell>
        <align:entity1 rdf:resource="http://stitch.cs.vu.nl/vocabularies/rameau/ark:/12148/cb14521343b"/>
        <align:entity2 rdf:resource="http://id.loc.gov/authorities/sh2002000569#concept"/>
        <align:measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">1.0</align:measure>
        <align:relation rdf:resource="http://www.w3.org/2004/02/skos/core#closeMatch"/>
      </align:Cell>
    </align:map>
  </rdf:Description>
</rdf:RDF>
```

(b) Concept Information Service, RDF response

Fig. 4. Serving RDF data

demonstrator, which is available at `http://stitch.cs.vu.nl/rameau`, also features an alignment between RAMEAU and LCSH (see `http://id.loc.gov`, [20]) as manually produced in the MACS project (see `http://macs.cenl.org`). The application of the Linked Data recipes allows users or RDF-consuming agents to seamlessly "follow their nose" from one KOS to the other.

Annotation suggestion tool. As part of the CATCH project, we have subjected a corpus of 250.000 *dually* indexed books of the National Library of the Netherlands (KB) to an instance-based method to derive an alignment between two KOSs, the *Brinkman* thesaurus and the *Biblion* one. We then implemented an alignment-based annotation suggestion tool to support KB employees indexing new, undescribed or singly indexed books. Library indexers get access to a list of subject suggestions, which they can accept or reject on a subject per subject basis [21]. For concepts they feel are missing, the tool gives browsing access to the vocabulary service. Feedback from KB staff is very positive as the annotation tool greatly supports the indexing task; the quality of automatically obtained rules reached a precision of 72.7% with a recall of 47.9%—and many mistakes are in fact near-matches. Further, the browser-based UI—dynamically generated HTML using XSLT—is perceived as more user-friendly than previously used software and paperwork for this task.

At the time of writing, the tool is only accessing the vocabulary service; we are working on adapting the alignment services to better fit the requirements of KB staff. In this respect, mappings between *combinations* of concepts from each KOS need to be exploited—for example, {'travel guides' + 'Spain' → 'Spain; travel guides'}. While the OAEI format supports many-to-many mappings, our current solution does not correspond yet to established practice in SKOS. This clearly points to further work determining representation and access means that fit existing (or anticipated) use cases and community best practices.

5 Discussion

RDF stores often result from conflating complex data models into huge sets of RDF triples. Making them accessible via SPARQL endpoints (or as Linked Data) leaves much freedom to application developers; on the other hand, the developers are on their own with constructing complex SPARQL queries to retrieve the data they require. A repository with well-defined services has many advantages for data consumers (implementors) but also for data providers and the community as a whole.

The implementation of the aforementioned three demonstrators, for instance, was greatly facilitated by the availability of our well-defined and fast repository services. The implementation of the autocompletion feature of the vocabulary and alignment browser, for instance, was helped by an appropriate service version of concept search via labels that only returns a lightweight representation of URI/label pairs. We hope that other application designers will profit from this and our other middleware services as well.

Data providers are in danger of running out of computing resources when making accessible a SPARQL endpoint with no restrictions. Here, consumers could easily formulate and submit queries that are far from optimized or tractable. Queries, for instance, that request the description of a vocabulary, including all the concepts that belong to this vocabulary (that is, following inbound `skos:inScheme` statements that have the concept scheme as object) are expensive, and it is more efficient to serve the same data via well-defined (and efficiently implemented) repository services.

Designing high-level repository services (and making them available to others) also pushes a community forward in terms of agreeing with common application requirements and best practises, but also in sharing expertise. Consider the example of Binding and Tudhope [22] on query expansion. Such mechanisms capitalize on a significant amount of existing research. Their implementation can be really tedious in any back-end engine, but it is hard, if not impossible, to reproduce it *via* SPARQL queries. It is not surprising thus that functionality of this kind is often provided at higher levels than SPARQL [23,17].

Sharing data at an appropriate level of abstraction is important for the CH institutions we work with. What makes our repository services unique with regard to others is that institutions can ingest and maintain their own KOS RDF data sources along with rich semantic alignments, a great help for institutions with little IT or expertise in this area.

Our future work will focus on scalability, improved speed and robustness. This includes the fine-tuning of services' description to reduce server/client communication. Also, we are currently investigating the provision of a local access API to complement the current network-based one. Having a local instance of the repository services would eliminate network bandwidth and yield significant better access times. Once there are locally-run repositories, the need may arise to synchronise their data with centrally-run repositories. The versioning problem, however, has not been tackled so far. Vocabularies (and alignments) evolve, and there is a strong requirement from CH practitioners to have our middleware handling this aspect. Unfortunately, we found that CH institutions employ rather *ad hoc* than systematic and easily implementable procedures for versioning vocabularies. Here, we would like to learn a lesson from more generic ontology repository systems and their versioning control.

Our web-based repository services for vocabularies and alignments are available at `http://stitch.cs.vu.nl/repository`. The webpage also gives access to the JavaDoc API. We would like to encourage interested parties to access and use it, and also to provide us with their feedback to improve the services.

Acknowledgements

This work was funded by the Dutch NWO CATCH programme (STITCH) and the eContentPlus programme of the European Union (TELplus). We are indepted to Stefan Schlobach, Shenghui Wang, Henk Matthezing, Frank van Harmelen and Hennie Brugman for valuable discussion time and advice.

References

1. Isaac, A., van der Meij, L., Schlobach, S., Wang, S.: An empirical study of instance-based ontology matching. In: Aberer, K., Choi, K.-S., Noy, N., Allemang, D., Lee, K.-I., Nixon, L.J.B., Golbeck, J., Mika, P., Maynard, D., Mizoguchi, R., Schreiber, G., Cudré-Mauroux, P. (eds.) ASWC 2007 and ISWC 2007. LNCS, vol. 4825, pp. 253–266. Springer, Heidelberg (2007)
2. Wang, S., Isaac, A., Schopman, B., Schlobach, S., van der Meij, L.: Matching multi-lingual subject vocabularies. In: Agosti, M., Borbinha, J., Kapidakis, S., Papatheodorou, C., Tsakonas, G. (eds.) ECDL. LNCS, vol. 5714, pp. 125–137. Springer, Heidelberg (2009)
3. Isaac, A., Schlobach, S., Matthezing, H., Zinn, C.: Integrated access to cultural heritage resources through representation and alignment of controlled vocabularies. Library Review 57(3), 187–199 (2008)
4. Isaac, A., Wang, S., Zinn, C., Matthezing, H., van der Meij, L., Schlobach, S.: Evaluating thesaurus alignments for semantic interoperability in the library domain. IEEE Intelligent Systems 24(2), 76–86 (2009)
5. Miles, A., Bechhofer, S.: SKOS Reference. W3C Recommendation (2009), `http://www.w3.org/TR/skos-reference/`
6. Tudhope, D., Koch, T., Heery, R.: Terminology Services and Technology – JISC state of the art review. Technical report, University of Glamorgan and UKOLN and University of Bath (September 2006)
7. Binding, C., Tudhope, D.: SKOS-based semantic web services: experiences from the STAR project. Presentation at the ISKO-UK KOnnecting KOmmunities Seminar: Sharing Vocabularies on the Web via SKOS (July 21, 2008)
8. Hillmann, D., Sutton, S.A., Phipps, J., Laundry, R.: A Metadata Registry from Vocabularies Up: The NSDL Registry Project. In: International Conference on Dublin Core and Metadata Applications (DC), Mexico (2006)
9. Jupp, S., Bechhofer, S., Stevens, R.: A Flexible API and Editor for SKOS. In: Aroyo, L., Traverso, P., Ciravegna, F., Cimiano, P., Heath, T., Hyvönen, E., Mizoguchi, R., Oren, E., Sabou, M., Simperl, E. (eds.) ESWC 2009. LNCS, vol. 5554, pp. 506–520. Springer, Heidelberg (2009)
10. Macgregor, G., McCulloch, E., Nicholson, D.: Terminology server for improved resource discovery: analysis of model and functions. In: International Conference on Metadata and Semantics Research, Corfu, Greece (2007)
11. Neubert, J.: Bringing the "thesaurus for economics" on to the web of linked data. In: WWW Worskhop on Linked Data on the Web (LDOW), Madrid, Spain (2009)
12. Sini, M., Lauser, B., Salokhe, G., Keizer, J., Katz, S.: The AGROVOC Concept Server: rationale, goals and usage. Library Review 57(3), 200–212 (2008)
13. Tuominen, J., Frosterus, M., Viljanen, K., Hyven, E.: ONKI SKOS Server for Publishing and Utilizing SKOS Vocabularies and Ontologies as Services. In: Aroyo, L., Traverso, P., Ciravegna, F., Cimiano, P., Heath, T., Hyvönen, E., Mizoguchi, R., Oren, E., Sabou, M., Simperl, E. (eds.) ESWC 2009. LNCS, vol. 5554. Springer, Heidelberg (2009)
14. Vizine-Goetz, D., Childress, E., Houghton, A.: Web services for genre vocabularies. In: International Conference on Dublin Core and Metadata Applications (DC), Madrid, Spain (2005)
15. Euzenat, J.: An API for Ontology Alignment. In: McIlraith, S.A., Plexousakis, D., van Harmelen, F. (eds.) ISWC 2004. LNCS, vol. 3298, pp. 698–712. Springer, Heidelberg (2004)

16. Duc, C.L., d'Aquin, M., Barrasa, J., David, J., Euzenat, J., Palma, R., Plaza, R., Sabou, M., Villazón-Terrazas, B.: Matching ontologies for context: The NeOn Alignment plug-in. Deliverable 3.3.2, NeOn project (2008)
17. Noy, N.F., Griffith, N., Musen, M.A.: Collecting community-based mappings in an ontology repository. In: Sheth, A.P., Staab, S., Dean, M., Paolucci, M., Maynard, D., Finin, T., Thirunarayan, K. (eds.) ISWC 2008. LNCS, vol. 5318, pp. 371–386. Springer, Heidelberg (2008)
18. Adida, B., Birbeck, M., McCarron, S., Pemberton, S.: RDFa in XHTML: Syntax and Processing. W3C Recommendation (2008), `http://www.w3.org/TR/rdfa-syntax`
19. Berrueta, D., Phipps, J.: Best Practice Recipes for Publishing RDF Vocabularies. W3C Working Group Note (2008), `http://www.w3.org/TR/swbp-vocab-pub/`
20. Summers, E., Isaac, A., Redding, C., Krech, D.: Lcsh, skos and linked data. In: International Conference on Dublin Core and Metadata Applications (DC), Berlin, Germany (2008)
21. Isaac, A., Kramer, D., van der Meij, L., Wang, S., Schlobach, S., Stapel, J.: Vocabulary matching for book indexing suggestion in linked libraries – a prototype implementation & evaluation. In: Bernstein, A., Karger, D.R., Heath, T., Feigenbaum, L., Maynard, D., Motta, E., Thirunarayan, K. (eds.) ISWC 2009. LNCS, vol. 5823, pp. 843–859. Springer, Heidelberg (2009)
22. Binding, C., Tudhope, D.: KOS at your Service: Programmatic Access to Knowledge Organisation Systems. J. Digital Information 4(4) (2004)
23. Daltio, J., Medeiros, C.B.: Aondê: An ontology web service for interoperability across biodiversity applications. Inf. Syst. 33(7-8), 724–753 (2008)

Appendix

Fig. 5 gives an overview of all vocabularies that are stored in the STITCH/CATCH/TELplus vocabulary repository ("source" column), and indicates their use of SKOS constructs (e.g., `skos:ConceptScheme`, `skos:Concept`, `skos:note`). For the complete table, and more details, please consult `http://www.cs.vu.nl/STITCH/repository/stats.html`.

Source name	ConceptScheme	hasTopConcept	Concept	prefLabel	altLabel	broader	related	note	scopeNote
Brinkman thesaurus	4	17169	12505	12505	1921	4704	2013	0	463
Mandragore voc.	1	13	16233	16390	2211	19097	0	2948	0
NBC (Dutch basic class.)	1	59	2143	6429	4286	2084	0	0	663
GOO thesaurus	9	78223	65297	104478	24418	26518	7559	0	5622
Glas voc.	11	141	337	337	0	240	0	0	0
Iconclass	2	14	24315	97209	1031867	26175	5462	0	0
GTAA thesaurus	6	0	160921	160921	1597	11658	42117	0	54566
Wordnet (nouns)	0	0	79689	79689	141691	81857	27243	0	0
SWD subject headings	9	0	805017	818364	1072714	292512	30042	230749	286816
Rameau subj. headings	7	0	154974	309979	196499	127161	59114	118467	27140
NBD/Biblion thesaurus	6	57968	49171	49171	5612	24820	2207	362	3067
Regiothesaurus KB	1	17	16410	20304	3174	16401	5560	0	9523
LCSH (lcsh.info)	13	0	340557	340557	310320	247112	21460	0	11118

Fig. 5. Statistics: vocabularies and usage of SKOS constructs (excerpt)