VocBench: A Web Application for Collaborative Development of Multilingual Thesauri

Armando Stellato[1]([⊠]), Sachit Rajbhandari[2], Andrea Turbati[1],
Manuel Fiorelli[1], Caterina Caracciolo[2], Tiziano Lorenzetti[1],
Johannes Keizer[2], and Maria Teresa Pazienza[1]

[1] ART Group, Department of Enterprise Engineering,
University of Rome, Tor Vergata, Via del Politecnico 1, 00133 Rome, Italy
{stellato,turbati,fiorelli,lorenzetti,
pazienza}@info.uniroma2.it
[2] The Food and Agricultural Organization of UN (FAO),
Viale delle Terme di Caracalla, 00153 Rome, Italy
{sachit.rajbhandari,caterina.caracciolo,
johannes.keizer}@fao.org

Abstract. We introduce VocBench, an open source web application for editing thesauri complying with the SKOS and SKOS-XL standards. VocBench has a strong focus on collaboration, supported by workflow management for content validation and publication. Dedicated user roles provide a clean separation of competences, addressing different specificities ranging from management aspects to vertical competences on content editing, such as conceptualization versus terminology editing. Extensive support for scheme management allows editors to fully exploit the possibilities of the SKOS model, as well as to fulfill its integrity constraints. We discuss thoroughly the main features of VocBench, detail its architecture, and evaluate it under both a functional and user-appreciation ground, through a comparison with state-of-the-art and user questionnaires analysis, respectively. Finally, we provide insights on future developments.

Keywords: Collaborative thesaurus management · SKOS · SKOS-XL

1 Introduction

SKOS [1] provided public institutions and other organizations with a fast path toward the Semantic Web [2], by allowing them to represent in RDF thesauri and other *knowledge organization systems* (KOSs) [3] traditionally adopted for tasks such as resource indexing, query expansion and faceted search. SKOS proves advantageous [4] for representing concept-based KOSs on the Semantic Web and the Linked Data [5], as it fosters interoperability of resources and the development of distributed applications. Additionally, SKOS-XL [6] provides an extension for describing terms, through lexical relationships and various metadata, concerning aspects such as history notes, editorial workflows and publication status. The SKOS specification is intentionally loose in defining the semantics of the provided modeling, in order to accommodate the variety

of existing practices and guidelines for the compilation of KOSs. Furthermore, many of the constraints that are part of the SKOS specification are not expressed through OWL axioms: verifying the logical consistency of a KOS through OWL-compliant systems is thus insufficient for validating it. *Dedicated editors* should then ensure the consistent use of SKOS (possibly adopting dedicated validators [7, 8]), while at same time implementing useful abstractions over raw data. The maintenance of a SKOS dataset is often beyond the possibility of a single developer, since thesauri tend to be heavy-weight (i.e., composed of many concepts and labels). Moreover, the normative nature of thesauri requires them to be "[…] developed, managed and endorsed by *practice of communities*" [9]. As such, thesaurus development should be a collaborative effort, rather than a top-down process independent from the communities that the thesaurus aims to serve.

In this paper, we present VocBench, a collaborative Web-based multilingual thesaurus editor, which complies with SKOS and its extension SKOS-XL. VocBench allows for collaborative management of the overall editorial workflow, by introducing different roles with specific competencies.

2   Motivations and Requirements

In 2008, the AIMS group of the Food and Agriculture Organization of the United Nations (FAO, http://www.fao.org/) fostered the development of a collaborative platform for managing the Agrovoc thesaurus [10]: the "Agrovoc Workbench". The rising interest in such a platform from other FAO departments and several other organizations motivated its reengineering into a more general thesauri management system: Voc-Bench. Its latest incarnation – VocBench 2, the system presented here – has been developed in collaboration between FAO and the ART group of the University of Tor Vergata in Rome (http://art.uniroma2.it). VocBench 2 has been rethought as a fully-fledged collaborative platform for thesauri management, available free-of-use and open source, offering native RDF support for SKOS-XL thesauri, while retaining from its original version the focus on multilingualism, collaboration and on a structured content validation & publication workflow.

VocBench is meant to satisfy the needs of large institutions and organizations (though may be adopted in smaller settings as well), by matching an assortment of requirements:

R1. Multilingualism. Properly characterizing the data in different natural languages is fundamental, especially for thesauri, due to their use in Information Retrieval.

R2. Controlled Collaboration. Opening up to communities is important, though the development of authoritative resources demands for the presence of some control to be exerted over the resource lifecycle.

R3. Data Interoperability and Consistency. Interoperability of several resources – which is at the basis of SKOS adoption – critically depends on data integrity and conformance to representation standards. However, the flexibility of SKOS translates to
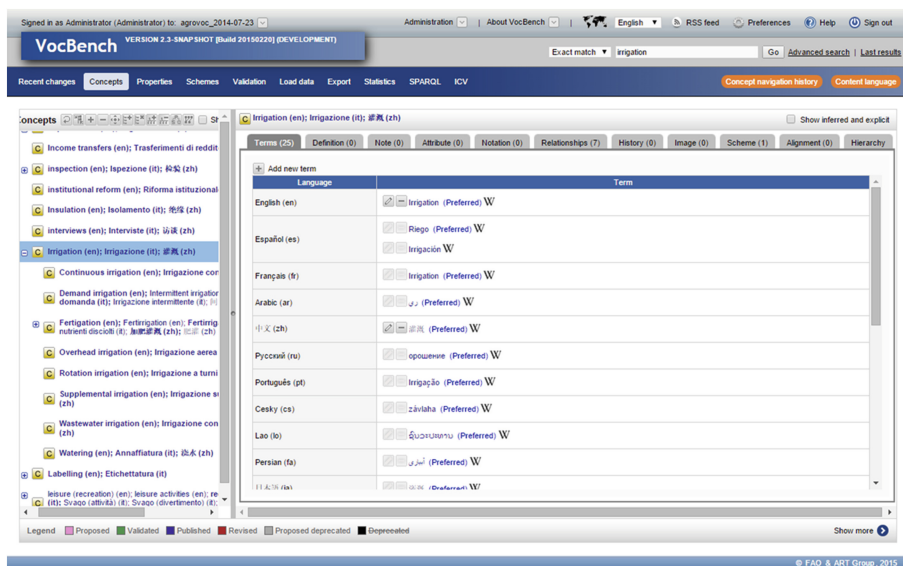
Fig. 1. VocBench user interface showing a fragment of the AGROVOC thesaurus

an underspecified model, at the same time exhibiting formal constraints that are even beyond the expressiveness of OWL. It is thus important that VocBench enforces a consistent use of SKOS, by preventing the editors from generating invalid data. Properly covering the whole family of RDF modeling languages is also part of this requirement, as SKOS actually sits on top of OWL and may benefit from the reuse of OWL vocabularies adding additional domain properties or specific modeling axioms. Finally, support for alignment to other datasets is also a must for the Linked Data World.

R4. Software Interoperability/Extensibility. The system should be able to interact with (possibly interchangeable) standard technologies in the RDF/Linked Data world.

R5. Scalability. The system must deal with (relatively) large amount of data, still offering a friendly environment. User Interface must take that into account.

R6. Under-the-hood data access/modification. While a friendly UI for content managers/domain experts is important, knowledge engineers need to access raw data beyond the usual front-ends, as well as to benefit from mass editing/refactoring facilities.

R7. Ease-of-use for both users and system administrators. This was a particularly important requirement in migrating from the first VocBench (adopted in a close, though large, community) to its second version, released as an open-source free-of-use system.

3   The New VocBench 2

VocBench (also abbreviated as VB) has been conceived as a web application accessible through any modern browser, therefore disburdening end users from software installation and configuration. Many of the limitations of VB1 with respect to the requirements described in the previous section were related to the lack of a real RDF backend. While VB1 was based on the API of Protégé 3 OWL (a non-native OWL wrapper around the legacy Protégé 3 frame-based model), VB2 has been re-designed to rely on the capabilities of Semantic Turkey [11], an RDF management platform already developed and currently maintained by University of Tor Vergata. Semantic Turkey (ST from now on) offers an OSGi service-based layer for designing and developing OWL ontologies and SKOS/SKOS-XL thesauri. A lightweight Firefox interface is available for use as a desktop tool, now complemented by VB, which mainly differentiates for its collaborative nature (and the focus on thesauri).The insight on usability of real thesaurus publishers informed the development. Specially, FAO and its partners provided great support for shaping user interaction and collaboration capabilities, therefore ensuring that VocBench was indeed functioning and meeting its user requirements.

In the rest of this section, we discuss the main characteristics of the software.

User Interface (UI). The user interface consists of multiple tabs, each one associated with specific information and functionalities. A quick exploration of the available tabs is sufficient to discover most of the VocBench functionalities, at least at the user level.

Figure 1 offers a typical view of VocBench, with the concept tree on the left, and the description of the selected concept on the right, centered on the term tab, listing all terms in the different languages available for the resource. Concepts in the tree may be shown through their labels in all of the selected languages for visualization. An option allows to toggle between preferred labels and all labels. Also the multilingual characteristics (requisite R1) of VB are not limited to content management, as the UI is itself localized in different languages, currently: English, Spanish, Dutch and Thailandese.

Controlled Collaborative Editing through Role-based Access Control. A single installation of VocBench may handle multiple independent thesauri. Upon registration, users indicate the thesauri they are interested in and the roles they want to cover; at any time, the administrator may grant additional permissions. VocBench promotes the separation of responsibilities through a role-based access control mechanism, checking user privileges for requested functionalities through the role they assume (req. R2). A completely customizable access policy specifies roles and their assigned privileges. New roles can be created, and existing ones can be modified. The default policy recognizes typical roles and their acknowledged responsibilities: *Administrators*, *Ontology editors*, *Term editors* (Terminologists), *Validators* and *Publishers*.

Formal Workflow and Recent Changes. Collaboration is essential for distributing effort and reaching consensus on the thesaurus being developed. To facilitate collaboration, VocBench provides an editorial workflow in which editors' changes are

tracked and stored for approval by content validators. This workflow management is supported by role-based access control, by providing users with different roles so to enforce the separation between their responsibilities. In a collaborative environment, where users may proactively edit a shared resource, it is important to have means for monitoring the situation. Regarding this aspect, the ability to control recent changes to the thesaurus is useful for detecting hot sections and coordinating with other editors. In VocBench, users can see recent changes both in the Web user interface and as an RSS feed.

Advanced Scheme Management. The definition of scheme in SKOS is blurred, as the SKOS reference [1] neutrally defines the scheme as an "aggregation of concepts" while SKOS primer [12] promotes schemes as identifiers for thesauri themselves, though reporting that several issues exist: #secskoscontainment. VocBench allows to manage thesauri organized around multiple concept schemes. Users can switch across schemes by selecting them through the relevant *Schemes* tab in the user interface. The Concepts tab shows the concept hierarchy and filters out concepts not belonging to the selected scheme. Concepts may belong to more than one scheme but must be in at least one, otherwise they are *dangling*, as they cannot be seen in any scheme view. VocBench functionalities are well-behaved with respect to schemes, as actions that would generate dangling concepts are forbidden, detailing the cause of the impediment to the users. In any case, since data can be loaded from pre-existing sources developed outside of VocBench, a fixing utility for dangling concepts is available through the UI. This will be part of a larger section dedicated to Integrity Constraints Validation, providing issue detection and repair actions (thus meeting requirements R3), which is currently available in ST and its Firefox UI and will ported to the UI of VB in the forthcoming VB2.4.

Vocabulary Import and Data Import/Export. The SKOS standard defines a very general, domain-agnostic, meta-model for the representation of KOSes. VocBench allows to import ontology vocabularies (from the web, file system or even a dedicated local mirror), providing additional shared descriptors (e.g. additional properties, which reflect specific conceptual and lexical relations for the domain of interest) for modeling the thesaurus. Data import/export is available for all notable RDF serialization formats. **Metrics & SPARQL Querying**. VocBench supports the computation of several metrics concerning the thesaurus itself and the collaborative workflow. These metrics are grouped with respect to common themes: distribution of labels across different languages, structure of the thesaurus, vocabulary use and workflow statistics. Structural metrics are helpful in assessing the granularity (hierarchy depth) of the thesaurus, its scope (hierarchy width) and its level of uniformity (variance of metrics). Statistics about the use of vocabulary properties help in understanding the completeness of the resource. Finally, workflow statistics support management of the entire editing process.

In addition to statistics and visualizations provided by VocBench, users may formulate SPARQL 1.1 queries to select information precisely, or to perform analytical tasks. The query editor is based on the open source project Flint SPARQL Editor (https://github.com/TSO-Openup/FlintSparqlEditor), which provides syntax highlighting and completion. The Flint syntax completion has been customized to be fed with

information (e.g. the adopted namespaces and their chosen prefixes) originating from the edited thesaurus. Availability of SPARQL updates completes the above in order to fulfill requirement R6.

Alignment. From version 2.3 (latest stable version at the time of writing), VocBench features a dedicated tab in the concept description area, showing alignments to other thesauri. Currently, the creation of alignments can either be performed manually, by inserting URIs as values of the various SKOS mapping properties, or be assisted in case of mappings to other thesauri managed by the same instance of VocBench. In the latter case, a concept-tree browser with advanced search interfaces (which can be manually prompted or automatically populated with the lexicalizations of the local concept) facilitates the identification of the best matching concepts from the targeted datasets.

4   Architecture

VocBench has a layered architecture (Fig. 2) consisting of a presentation and multi-user management layer, a service layer and a data management layer. The first layer is implemented as a Web application, powered by GWT (Google Web Toolkit, http://www.gwtproject.org/). The other layers coincide with the Semantic Turkey RDF management platform, equipped with an extension providing additional services expressly developed for VocBench. VocBench is also in charge of user and workflow management, since these aspects are not covered by Semantic Turkey. User accounts and tracked changes are stored in a relational database accessed through a JDBC connector. The ST backend manages the data and implements all the required editing functionalities. The interface between the frontend and backend consists of a series of lightweight Web services in the spirit of the Web API movement. Semantic Turkey provides core services related to project management, OWL and RDFS ontologies, SPARQL, etc. Furthermore, the adoption of OSGi allows for dynamic plugging of extensions: in particular, other than realizing additional services, different connectors for specific RDF middleware and triple storage technologies can be provided (req. R4). VocBench is currently shipped with a connector for Sesame2 [13], supporting all of its storage/connection possibilities: in memory, native, remote connection and their respective configurations. The remote connection is particularly useful, as it allows VocBench to connect to Sesame2 compliant triple stores (e.g. GraphDB [14]) without need for a dedicated connector. VocBench RDF API are based on OWL ART (http://art.uniroma2.it/owlart/), an abstraction layer supporting access to different triple stores. Different connectors can be implemented from scratch in terms of those API, or by reusing middleware already bridged through other existing connectors. For instance, the Virtuoso triplestore [15] is compatible with the Sesame API, but requires a dedicated client library: it thus needs to be introduced by a specific connector, though its implementation may be largely realized as an extension of the already existing Sesame connector.

Particular attention has been paid to system scalability (req. R5), both on performance and maintenance aspects. To this end, information is provided to the frontend as much as possible in an incremental fashion (e.g., each level of the concept hierarchy, as
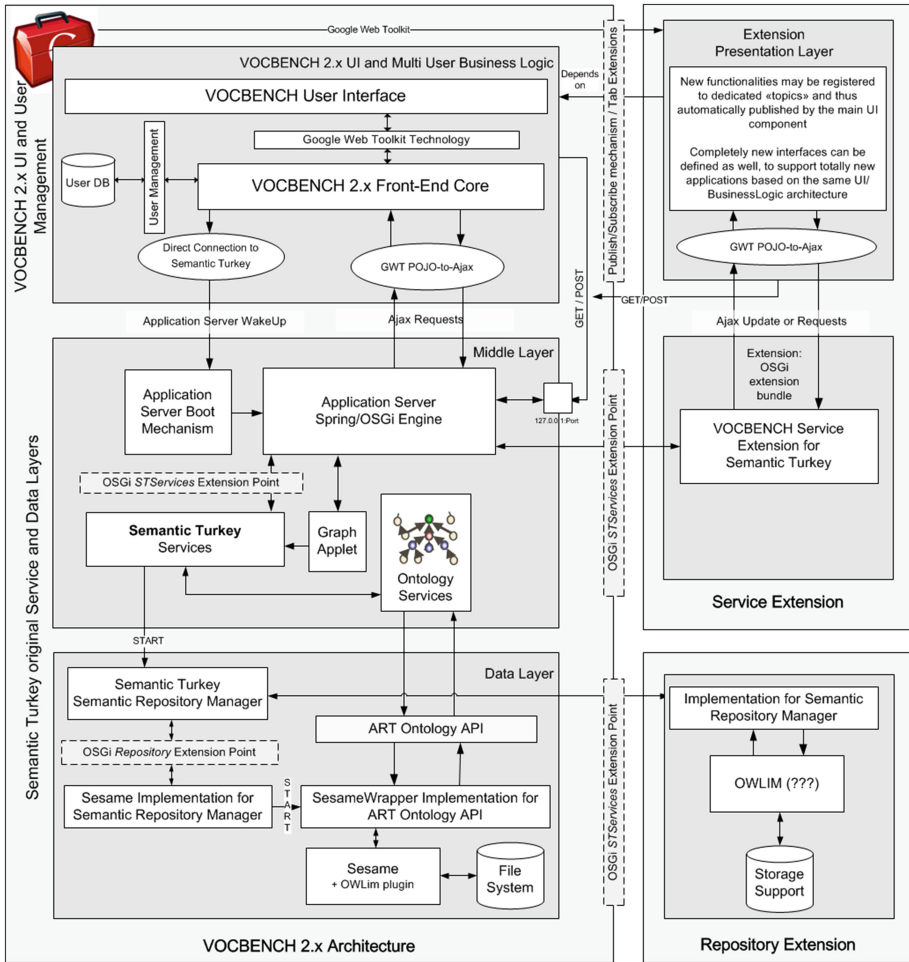
Fig. 2. VocBench 2 extensible architecture

nodes are expanded). Also, though we tried to maintain a meaningful core set of RDF services, many functionalities (especially in the user interface) require the composition of several calls. We thus provided both per-service ad hoc solutions (heavy weight single services realizing specific functionalities) and general development facilities for the injection of additional information into common API calls (e.g. the rendering of RDF resources is available as an extension point, with different implementations being dynamically injectable into the SPARQL queries of several services).

A continuous check-on-start life cycle satisfies requirement R7: VB technically never recognizes itself as installed/deployed, rather at each application startup it checks that the complete set of pre-requisites for a correct start is satisfied. Whenever a new VB version is installed, if new features have been introduced, or mandatory configuration

options added, or the database requires update batches, the system will identify these needs and react accordingly, eventually interacting with the user upon necessity.

5   Related Tools

In this section, we survey other thesaurus editors that we will later compare to VocBench. We analyzed the latest versions of the systems (unless differently reported, as in the case of SKOSed) by asking evaluation licenses when necessary, as in the case of proprietary tools. Even though our survey is certainly incomplete with respect to existing tools (e.g. we have never received the license we requested for Topbraid EVN, http://www.topquadrant.com/products/topbraid-enterprise-vocabulary-net/), we believe our sample is representative of existing technologies.

WebProtégé, http://webprotege.stanford.edu/ [16] is an open source web based system for collaborative ontology development. Unfortunately, WebProtégé has not a dedicated support for SKOS (it covers editing of OWL/OBO), however it has been included in the survey due to extensive support to collaboration, which is an important aspect in our review. WebProtégé is available as a locally installable web application, also offered as a free service via a public portal. It has a clean user interface, organized in a collection of tabs, which in turn contain widgets showing different types of information. The user interface is completely configurable: users (even at runtime) can add, remove or reposition the widgets within a tab as well as add/remove tabs themselves.

WebProtégé relies on the collaboration plugin for Protégé 3 [17], providing change tracking, inline discussions and notifications. It also features an access control mechanism for user groups, based on configurable policies enforced at various granularities. It has a plugin architecture, which supports the development and deployment of additional functionalities. Integration with other applications is also possible through the API provided by the service and backend layers.

PoolParty, http://www.poolparty.biz/ [18] is a proprietary Web based editor for thesauri utilizing Linked Data. It exists in different editions, possibly bundled with other tools supporting semantic tagging and semantic search. Buying options include both on-premises installation and hosted solutions. For our analysis, we obtained a free evaluation account for PoolParty Advanced Server version 4.5.1 (rev 5429).

PoolParty supports by default SKOS and has an optional add-on for SKOS-XL. SKOS compliance includes concept lists and collections; PoolParty does not explicitly attach concepts to schemes, but the sole connection with a scheme lies in the reachability from one of its top concepts (this is in contrast with the specification of nonentailment of scheme containment along concept hierarchies, specified in Sect. 4.6.4 of the SKOS Reference [1]). PoolParty supports custom modelling vocabularies expressed in RDFS or OWL, either locally edited or imported from external sources.

Version Tracking is supported, as the system performs access control to some extent. An add-on further enables an approval workflow based on the existing role based access control mechanism. Editing history is shown both at project level and at entity level.

PoolParty supports the lookup over resources published as LOD, either to gather additional information or to create mappings. Similarly, different projects can be linked together, for instance, to enable concept mapping. Additionally, PoolParty publishes a SPARQL endpoint, dereferenceable URIs, and a wiki with limited editing capabilities.

Depending on the specific settings, quality criteria are enforced interactively (i.e., illegal operations are blocked), or violations are simply recorded in a quality report.

PoolParty uses Sesame2 as an abstraction layer over different RDF triple stores, possibly supporting inference. APIs for integration with other applications are available, ranging from basic synchronization up to text mining and indexing applications.

TemaTres, http://www.vocabularyserver.com/, is an open source web application for the management, publication and sharing of controlled vocabularies. TemaTres adopts a term-based meta-model for the representation of thesauri and controlled vocabularies in general. While vocabularies are inherently monolingual, a form of multilingualism is supported through alignments between vocabularies (on the same instance of TemaTres, or remotely accessible through a dedicated web service interface). It is possible to export the data in several formats as well as to import from SKOS and tabular representations. Due to the term-based nature of the model, the export to SKOS is often confusing as, for instance, two terms bound as synonyms are actually exported as two different skos:Concepts. Each vocabulary is associated to a single skos: ConceptScheme.

TemaTres has a rigid access control mechanism based on user roles (administrator, editor, guest). It also features workflow management, which is based on the transition of terms from the candidate status to either accepted or rejected. Editing of a term changes the last modification date, but it is not subjected to further approval. In other words, once a term is approved, changing it does not revert its status from accepted.

Facilities for data quality include metrics and a flexible reporting generator.

TemaTres exposes an API for integration with other systems, such as a thesaurus publishing interface, and a WordPress plugin. A TemaTres add-on, TemaTres Keywords Distiller, supports the automatic categorization of unstructured content.

SKOSEd, https://code.google.com/p/skoseditor/ [19]. An open source plugin for Protégé 4.x for editing SKOS thesauri, SKOSEd represents an exception in our survey as, differently from the aforementioned systems, it is not a web application but a desktop tool, which however we consider worth being mentioned. Being embedded into an ontology editor, SKOSEd allows interweaving SKOS and OWL constructs, and inherits from the hosting environment various capabilities: reasoning, usage search and various rendering options (enhanced through SKOS labelling properties).

We have evaluated version 1.0-alpha(build04) on Protégé 4.1 as, unfortunately, the more recent version 2.0-alpha has a bug related to scheme management: once a scheme has been created, it is no longer possible to create new concepts.

SKOSEd adds to Protégé a dedicated tab, offering tree visualization of concept hierarchies, as well as an input form tailored to the SKOS model. However, the system adopts the same form for concepts and concept schemes; consequently, a user can easily assert that a concept scheme is a top concept of another concept scheme. The hosting environment allows creation and import of additional RDFS and OWL

vocabularies. Despite this overall flexibility, the SKOS view is somewhat rigid, since the widget for asserting related concept is not aware of possible refinement provided by additional vocabularies. In fact, these properties are only accessible as other properties.

SKOSEd supports plugging of external reasoners to determine whether the thesaurus being edited is consistent with respect to the OWL definition of the SKOS model.

As for PoolParty, the concept tree visualization is only based on the membership of topconcepts to a given scheme, not filtering out narrower concepts not belonging to it.

Being an extension of Protégé 4.x, SKOSEd may not be used in conjunction with the collaboration framework developed for Protégé 3.x.

6   Functional Evaluation

In this section, we compare VocBench to the previously reviewed tools with respect to dimensions expressing interesting and useful features (Table 1).

The first consideration is that VocBench is open source and free to use. This fact is particularly unique among the most accredited thesaurus editors (e.g. PoolParty or Topbraid EVN), which are typically proprietary. The open source nature is advantageous, since it allows wide customizability for specific uses, as well as the possibility to add features to the mainstream distribution. TemaTres seems to depend on a term-based representation of thesauri, which can be exported to many formats, including SKOS. The downside of this approach is the somewhat approximated and limited support to SKOS constructs. VocBench is the only editor natively supporting the SKOS-XL specification (followed by PoolParty with its dedicated SKOS-XL addon).

Support for concept schemes is practically inexistent in TemaTres (each thesaurus is a scheme), while PoolParty and SKOSEd suffer from the same issue with improper entailment of scheme membership inherited from topConcepts. Conversely, VocBench fits better the intended semantics of concept schemes in SKOS with its Advanced Scheme Management features.

The grounding of SKOS in a specific domain/application or editorial environment is realized by the adoption of other RDF vocabularies. SKOSEd is the most advanced with respect to the creation capability, as it is embedded in the ontology editor Protégé. The downside of this power is lesser control on the data being edited/created. VocBench, on the other hand, though not providing the full OWL editing capabilities of Protégé, still allows limited property editing and supports owl:importing external OWL vocabularies.

Obviously, all the systems support import/export of the edited thesaurus. TemaTres has an extensive support for different formats, not limited to RDF. PoolParty and TemaTres are also able to import data from tabular representations, such as spreadsheets, based on a set of statically defined conventions for their format. VocBench has not such a built-in feature in its User Interface. However, we have already developed a highly flexible converter, Sheet2RDF (http://art.uniroma2.it/sheet2rdf/), and made it available for the Firefox interface of Semantic Turkey. It is possible to use the Firefox UI over the same ST instance that is backing VocBench, thus making the whole

Table 1. Comparison of thesaurus management tools

Name	VocBench	PoolParty	WebProtégé	TemaTres	SKOSEd
License	GNU GPL v3 (web application), Mozilla Public License MPL (Semantic Turkey)	Proprietary	Mozilla Public License (MPL)	GNU General Public License version 2.0 (GPLv2)	GNU Lesser GPL
Free to use	Yes	No	Yes	Yes	Yes
Deployment	Web application	Web application	Web application	Web application	Desktop application
Data Models	SKOS-XL, SKOS through offline scaling tool	SKOS SKOS-XL add-on	OWL 2, OBO	Term based thesaurus organization	SKOS
Import/Export	SKOS(-XL), versatile spreadsheet import (through ST Firefox UI)	SKOS(-XL), static spreadsheet import	OWL	MADS, SKOS-Core, Zthes, Others Import from: Skos-Core, tabulated or tagged text file	SKOS
Scheme Management	Yes	Only top concepts	Not applicable	One scheme per vocabulary	Only top concepts
Custom Relations	Import, use	Creation, use	Creation, use	Creation, use	Creation, use
Reasoner	Depends on triple store	Depends on triple store	No, external reasoning possible	No	Depends on available plugins
Data quality	Metrics	Metrics Validation rules	Metrics	Metrics, Reports	KB consistency
Extendibility / Interoperability	API, shared backend, pluggable	REST API	API, shared backend,pluggable	API	Pluggable
ACL	Yes	Yes	Yes	Yes; limited	No
Workflow Management	Yes	Yes (add-on)	No	Yes; limited	No
Collaboration, Content Validation	Change feed, validation	History, versioning, validation	Discussion, watching, changes feed	Limited validation	No
RDF Middleware	OWL ART API (connectors to others: Sesame2 bundled)	Sesame SAIL API	OWL API	No RDF Middleware, SKOS RDF/XML available only as an export	OWL API (used by Protégé 4)
RDF Backend	provided by Sesame2, or other connectors	provided by Sesame2	provided by Protégé 3	Relational database (MySQL by default)	provided by Protégé 4 (OWL API)
SPARQL Querying	Yes	Yes	No	Not native, no realtime, can export data to a SPARQL endpoint through ARC2 (RDF library for PHP)	Yes (inherited from Protégé 4)
Semantic Integration	assisted (browse&search) linking of resources from other projects / manual linking of LOD resources. Extensions for RDF lifting from unstructured content	Linking Text Mining & Entity Extraction, Search function	linking to BioPortal	Linking between vocabularies, Entity Extraction (via addon)	N/A

process require no export/import nor adaptation of data. Sheet2RDF integration inside VocBench UI is also under development.

The RDF framework supports the automatic inference of implicit facts from the explicitly represented knowledge. Reasoning might be useful to materialize redundant information in SKOS thesauri, e.g. skos:broader/narrower relationships, or their transitive closure through skos:broader/narrowerTransitive, or even more elaborated facts determined by axioms defined in the domain vocabularies. VocBench and PoolParty exploit the reasoning capabilities provided by the implementations of the knowledge base, while SKOSed and WebProtégé generally assume that reasoning is performed not in real-time, but by an external component connecting to the backend holding the data.

In traditional ontology development, reasoning is important to formally validate the ontology, by verifying its logical consistency: this is not the case for SKOS thesauri, since most assumptions about the use of SKOS are not explicitly encoded as formal OWL axioms. Therefore, assessing and improving the quality of SKOS thesauri requires dedicated solutions. PoolParty supports different sets of validation rules, which can be enforced during editing or used to generate quality reports. VocBench enforces the consistent use of SKOS constructs, such as the already described constraints on concept scheme management or the uniqueness of preferred labels in a given language, by providing both in line validation and fixing utilities for ingested non-orthodox data.

Another feature relevant for data quality is the possibility to compute metrics and generate various types of reports. Tools differ from each other in terms of the metrics they are able to calculate.

WebProtégé stands up for its support to coordination, by providing history, watching and discussion facilities. VocBench and PoolParty do support history as well, and in addition they support change validation, with VocBench distinguishing more life cycle states than PoolParty. TemaTres has an even more limited set of states, and, as said, once a term has been accepted, subsequent editing does not cause revert the state back from accepted. In both VocBench and PoolParty, validation leverages the role-based access control mechanism. PoolParty has a couple of roles, while VocBench has a more fine-grained and flexible mechanism, which is based on primitive permissions associated with specific actions. Then, specific roles are defined as an assignment of these permissions. VocBench provides by default roles commonly found in thesaurus development processes; nonetheless, it is possible to create new roles as desired. Thus, VocBench allows matching each role to a specific set of competences and duties.

Most of the tools, including VocBench, offers great flexibility for the connection to RDF semantic stores targeting different tradeoffs between requirements. Similarly, these tools tend to support the development of extensions and the integration with other systems. In VocBench, this is achieved by a pluggable architecture and APIs offered to clients. Even the RSS feed can be seen as API to support coordination with other tools, since it contains all the relevant information about each change. Individual editors may subscribe to this RSS feed to be warned of thesaurus changes, which can be considered as a form of watching.

Finally, in some of these tools the aforementioned extendibility supports complex features related to semantic integration beyond thesaurus editing. PoolParty may be integrated with unstructured content analysis systems, as well as with semantic search systems. TemaTres supports the federation of different vocabularies, in order to

establish links between them. VocBench has been equipped with ontology alignment capabilities, currently either by manual data entry or by assisted browsing of other projects internally managed by the application.

7   User Community and Evaluation

VocBench 2.0 was released in November 2013. Thanks to word-of-mouth about the previous VocBench 1.x, and to the insights about the new features and larger flexibility the new version would have brought, it has immediately gathered the interest of a discrete number of organizations (http://aims.fao.org/tools/vocbench/partners).

The current version of the system is VB2.3, released March 2015.

Table 2. USE values

	Usefulness	Ease of use	Ease of learning	Satisfaction
Global	5,34	4,49	5,11	4,93
Experienced	5,58	4,66	5,18	5,02
Inexperienced	4,97	4,19	5,00	4,79

VocBench has a public Web site: http://vocbench.uniroma2.it/. Two mailing lists have been made available to support users (http://groups.google.com/group/vocbench-user) and developers (http://groups.google.com/group/vocbench-developer). To evaluate the appreciation of VocBench among its users, we administered an online questionnaire to the mailing subscribers. We received 11 anonymous responses which have been made publicly available (http://vocbench.uniroma2.it/purl/VocBench-User-Questionnaire_2014-10.zip). The questionnaire is composed of three sections: user profiling, a usability evaluation and features evaluation.

The respondents considered themselves quietly proficient with thesaurus editing, as well as with languages of the RDF family, although in the latter case the answers were more scattered. Users experiences with other tools confirmed our belief in the representativeness of our survey of thesaurus editors.

We adopted the USE questionnaire (http://hcibib.org/perlman/question.cgi?form=USE) to evaluate how VocBench users perceive its usability along four dimensions: usefulness, ease of use, ease of learning and satisfaction. Each dimension is evaluated through a set of Likert-items (with scores ranging from 1 to 7). Table 2 reports the average score regarding each dimension.

The first row of the table represents the average over the entire sample. All averages represent an encouraging result, especially if considering that the highest value was given to Usefulness: this means that users believe that using this tool aids them in their work despite they consider it not very easy to use and to be learned.

We divided the respondents into two disjoint groups based on whether they reported to have adopted other related tools (64 %) or not (36 %). The usability metrics on the experienced group are consistently (and uniformly) higher than those obtained

from the inexperienced one. This is a good indicator as somewhat reflects a good positioning with respect to the state of the art.

The last part of the questionnaire was aimed at surveying the perceived value of some of the most important features of VocBench, in terms of interestingness, effectiveness and easiness of use. For each dimension, a 7-point scale was used. Table 3 shows the average agreement on each dimension and the rows are ordered in decreasing order of how they are perceived as interesting by the users.

Unsurprisingly, collaboration related features are the top rated characteristics. The only negative value in that table (below 4) is the easiness of the triple store connectivity, which is, though, an intrinsically complex feature, negatively affected by the still scarce standardization of triple store connectivity. Users are however interested (average score: 5) in the possibility to plug different stores or even RDF middleware.

Table 3. Feature evaluation

	It's easy to use	It's effective	It's interesting
History	5,38	5,50	6,33
SPARQL querying	4,00	5,40	6,29
Publication workflow Management	5,50	5,63	6,22
Collaborative management	5,75	5,88	6,11
Scheme management	4,83	5,17	5,57
Role-based access control	5,33	5,22	5,40
Reasoning	4,29	4,43	5,38
Triple store connectivity	3,67	4,50	5,00

8   Conclusion and Future Work

VocBench addressed the need of an open-source general-purpose editor of SKOS-XL thesauri supporting a formalized editorial workflow. In this paper, we discussed the features of VocBench and its architecture. Then, we surveyed a representative sample of related tools, to identify important features, and to show that VocBench mostly covers them and in some cases surpasses the state-of-the-art.

A vibrant user community[1] grew around VocBench initially inside various departments of FAO, and later spread across other organizations with analogous needs. Continuous user feedback allowed us to spot bugs and to improve the usability of VocBench.

The most important improvement we are working on consists in a more extensive and uniform access to internal and external resources (such as Linked Open Data). This will be particularly useful for improving the alignment user experience, with users browsing both local and LOD resources from within the VocBench interface, performing alignments in a seamless way. Another improvement is towards more complete extensibility: as we previously mentioned, Semantic Turkey has already support

---

[1] See http://vocbench.uniroma2.it/support/ and the related community and mailing lists links.

for extensions, however when it comes to the UI extensions, the GWT framework is rather limited due to its java → javascript compilation phase. We will explore how to overcome this limitation. By following the user evaluation results, we will also add more data connectors for covering the most notable middlewares and triple stores.

References

1. World Wide Web Consortium (W3C): SKOS Simple Knowledge Organization System Reference. In: World Wide Web Consortium (W3C). http://www.w3.org/TR/skos-reference/. Accessed 18 August 2009
2. Berners-Lee, T., Hendler, J., Lassila, O.: The semantic web: a new form of web content that is meaningful to computers will unleash a revolution of new possibilities. Sci. Am. **279**(5), 34–43 (2001)
3. Hodge, G.: Systems of Knowledge Organization for Digital Libraries: Beyond Traditional Authority Files. Council on Library and Information Resources, Washington, DC (April 2000)
4. Pastor–Sanchez, J.A., Martinez–Mendez, F.J., Rodríguez–Muñoz, J.V.: Advantages of thesaurus representation using the simple knowledge organization system (SKOS) compared with proposed alternatives. Inf. Res. **14**(4), 10 (2009)
5. Heath, T., Bizer, C.: Linked data: evolving the web into a global data space. Synth. Lect. Semant. Web Theory Technol. **1**(1), 1–136 (2011)
6. World Wide Web Consortium (W3C): SKOS Simple Knowledge Organization System eXtension for Labels (SKOS-XL). In: World Wide Web Consortium (W3C). http://www.w3.org/TR/skos-reference/skos-xl.html. Accessed 18 August 2009
7. Mader, C., Haslhofer, B., Isaac, A.: Finding quality issues in SKOS vocabularies. In: Zaphiris, P., Buchanan, G., Rasmussen, E., Loizides, F. (eds.) Theory and Practice of Digital Libraries 7489, pp. 222–233. Springer, Heidelberg (2012)
8. Suominen, O., Hyvönen, E.: Improving the quality of SKOS vocabularies with Skosify. In: ten Teije, A., Völker, J., Handschuh, S., Stuckenschmidt, H., d'Acquin, M., Nikolov, A., Aussenac-Gilles, N., Hernandez, N. (eds.) Knowledge Engineering and Knowledge Management. LNCS, vol. 7603, pp. 383–397. Springer, Heidelberg (2012)
9. Shadbolt, N., Berners-Lee, T., Hall, W.: The semantic web revisited. IEEE Intell. Syst. **21**(3), 96–101 (2006)
10. Caracciolo, C., Stellato, A., Morshed, A., Johannsen, G., Rajbhandari, S., Jaques, Y., Keizer, J.: The AGROVOC linked dataset. Semant. Web J. **4**(3), 341–348 (2013)
11. Pazienza, M., Scarpato, N., Stellato, A., Turbati, A.: Semantic Turkey: a browser-integrated environment for knowledge acquisition and management. Semant. Web J. **3**(3), 279–292 (2012)
12. World Wide Web Consortium (W3C): SKOS Simple Knowledge Organization System Primer. In: World Wide Web Consortium (W3C). http://www.w3.org/TR/skos-primer. Accessed 18 August 2009

13. Broekstra, J., Kampman, A., van Harmelen, F.: Sesame: a generic architecture for storing and querying RDF and RDF schema. In: Horrocks, I., Hendler, J. (eds.) ISWC 2002. LNCS, vol. 2342, pp. 54–68. Springer, Heidelberg (2002)

14. Kiryakov, A., Ognyanov, D., Manov, D.: OWLIM – a pragmatic semantic repository for OWL. In: International Workshop on Scalable Semantic Web Knowledge Base Systems (SSWS 2005), WISE 2005, New York City, USA, 20 November 2005

15. Erling, O., Mikhailov, I.: RDF support in the virtuoso DBMS. In: Pellegrini, T., Auer, S., Tochterman, K., Schaffert, S. (eds.) Networked Knowledge - Networked Media. Studies in Computational Intelligence, vol. 221, pp. 7–24. Springer, Berlin Heidelberg (2009)

16. Tudorache, T., Nyulas, C., Noy, N., Musen, M.: WebProtégé: a collaborative ontology editor and knowledge acquisition tool for the web. Semant. Web **4**(1), 89–99 (2013)

17. Tudorache, T., Noy, N., Tu, S., Musen, M.: Supporting collaborative ontology development in protégé. In: Sheth, A., Staab, S., Dean, M., Paolucci, M., Maynard, D., Finin, T., Thirunarayan, K. (eds.) The Semantic Web - ISWC 2008 5318, pp. 17–32. Springer, Heidelberg (2008)

18. Schandl, T., Blumauer, A.: PoolParty: SKOS thesaurus management utilizing linked data. In: Aroyo, L., Antoniou, G., Hyvönen, E., Teije, A., Stuckenschmidt, H., Cabral, L., Tudorache, T. (eds.) The Semantic Web: Research and Applications 6089, pp. 421–425. Springer, Heidelberg (2010)

19. 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.) The Semantic Web: Research and Applications 5554, pp. 506–520. Springer, Heidelberg (2009)