Journal of Integrative Agriculture
2012, 11(5): 731-740

Available online at www.sciencedirect.com
ScienceDirect

May 2012

REVIEW

A Review and Prospects on Collaborative Ontology Editing Tools

XIAN Guo-jian and ZHAO Rui-xue

Agricultural Information Institution, Chinese Academy of Agricultural Sciences, Beijing 100081, P.R.China

Abstract

Building ontology is a fundamental but also hard work. Collaborative ontology editing tools can make ontology development more efficiently. In this paper, the important features of collaborative ontology development were analyzed, and several tools such as AGROVOC Concept Server Workbench (ACSW), Collaborative Protégé and WebProtégé were studied. Besides, some comparisons among them from several aspects were made and some prospects for the further improvement of these tools were given. Finally, we show it is a good way to build agricultural ontology with these tools collaboratively and simultaneously.

Key words: collaborative ontology editing, agricultural ontology, ACSW, Collaborative Protégé, WebProtégé

INTRODUCTION

Ontology is a formal representation of a set of concepts and the relationships between them. With the growing acceptance of ontology, it plays an increasing important role in addressing issues such as knowledge sharing, system interoperation, information integration, information retrieval, and question answering as well (Yongyuth *et al*. 2008). Building ontology is a fundamental, knowledge-intensive and also time-consuming work. As ontologies are becoming popular within scientific domains, their coverage is becoming so large that no single person or small group of people can construct them effectively (Tudorache *et al.* 2008). Developing ontology is a large project with numerous participants, who are often geographically distributed. So far, most ontology editing tools have focused on the development of ontologies in a stand-alone, single-user mode. With the arising of Web 2.0 and Semantic Web technologies and applications, the web has become a primary platform where people communicate, discuss, and collaborate on projects. Hence, web-based ontology editing tools is the ideal platform to support distributed domain experts, ontology engineers, and even the public to develop ontologies collaboratively. There are already such kinds of tools as Ontolingua Server (Farquhar *et al.* 2003), OntoWiki (Auer *et al.* 2006), NeOn Toolkit (Haase *et al.* 2008), AGROVOC Concept Server Workbench (ACSW) (Sini *et al.* 2007), Collaborative Protégé (Tudorache and Noy 2007), and WebProtégé. The reminder of this paper will give a review and also makes some comparisons on some of these tools, which aims to give some reference while choosing a suitable platform to develop ontology or designing and developing our own ontology editing tools.

IMPORTANT FEATURES OF COLLABORATIVE ONTOLOGY EDITING TOOLS

In order to meet the requirements of supporting col-

Received 28 June, 2011 Accepted 31 August, 2011

XIAN Guo-jian, E-mail: xgj@mail.caas.net.cn; Correspondence ZHAO Rui-xue, E-mail: zhaorx@mail.caas.net.cn

laborative ontology construction, the editing tools should have the following important features:

(1) Various levels of expressiveness. This is the basic requirement to ensure the developed ontologies' completeness and richness. The collaborative tools should ultimately have the same expressive power as a standalone ontology editor. They must support editing all ontology elements such as classes, properties, complex restrictions, and instances.

(2) User management and access control. Users need to have fine-grained access-control based on their roles and privileges, particularly in the development of large ontologies. For instance, some users can make proposals for changes; others can comment and discuss on these proposals; other group of users can perform quality control by reviewing and approving the changes (Finin *et al.* 2008).

(3) Change history tracking. As lots of people contributing to the ontology and the corresponding discussion, it is critical for users to track out where information comes from and what changes happen. Users must be able to see when and who makes specific changes.

(4) Version management and quality control. Version management is very important for ontology maintenance and evolution, such as examining changes from previous baseline versions and accepting or rejecting them before publishing a new baseline. Tools should provide functions for ontology managers to exercise quality control, for instance, conduct consistency checking.

(5) Synchronous and asynchronous editing. In some cases, users want to have their changes seen by everyone as soon as they make them, without the additional step of "checking in" their changes. In other cases, users prefer to have their own "sandbox" to test out the changes they proposed before sharing them with others.

(6) Flexible workflow support. A workflow often includes different tasks that editors are charged with, process for proposing a change and reaching consensus, roles that different users play. Flexible workflows must be an integral part of tools for collaborative development to support different workflow models.

(7) Discussion and annotation. Developing ontology is not an easy task and the developers may disagree on the best way to model concepts in the ontology. Hence, tools that support discussion and chats are essential.

AGROVOC CONCEPT SERVER WORKBENCH

ACSW (also known as VocBench) is originated by the Food and Agriculture Organization of the United Nations. It provides tools for building and structuring multilingual ontology and terminology systems in the area of agriculture with a distributed environment. ACSW aims to create a collaborative reference platform and a "one-stop" shop for a pool of commonly used concepts related to agriculture, containing terms, definitions and relationships between terms in multiple languages derived from various sources.

Architecture of ACSW

ACSW is a web-based working environment to facilitate the collaborative editing of multilingual terminologies and semantic concepts. It consists of three main parts: the Ontological Knowledge management component, the User Management component and the authoring tools. Within the workbench, ontology is kept in OWL format by using MySQL as the persistence repository. Protégé API has been used as an OWL framework to operate data in OWL format such as querying, adding, deleting OWL statement and exporting data. ACSW's architecture is shown in Fig. 1.

Main functionalities of ACSW

The most important component of ACSW is Ontological Knowledge management. There are several main functionalities in this part to support workflows for ontology maintenance, validation and quality assurance.

Concept management This module provides concept navigation in tree view style. Editors can manage concept after selecting any node from the concept tree (Sini *et al*. 2010). They can also add, edit or delete more detail information for each concept in any language, such as the definition, history change, scope note, and editorial note for reminding some important

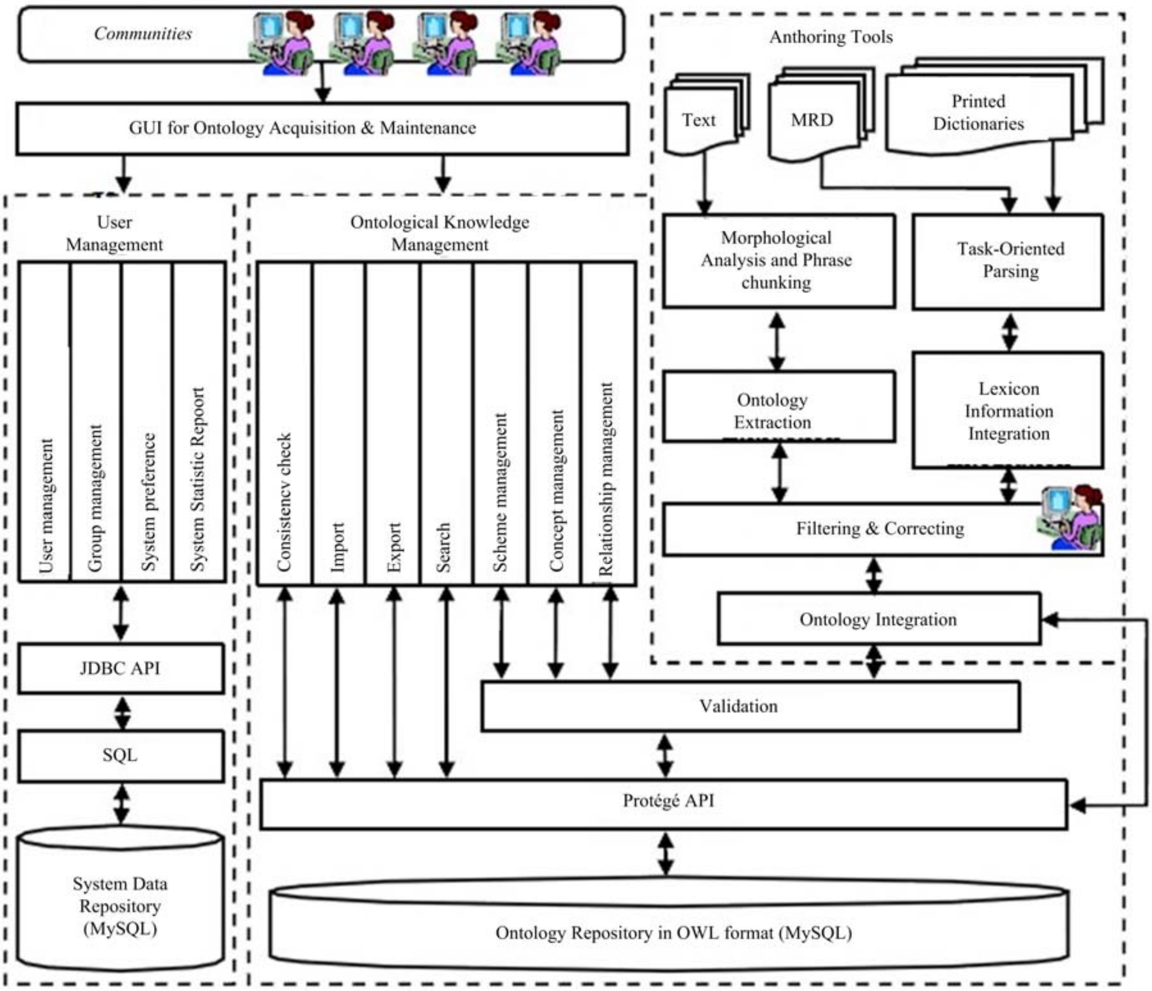


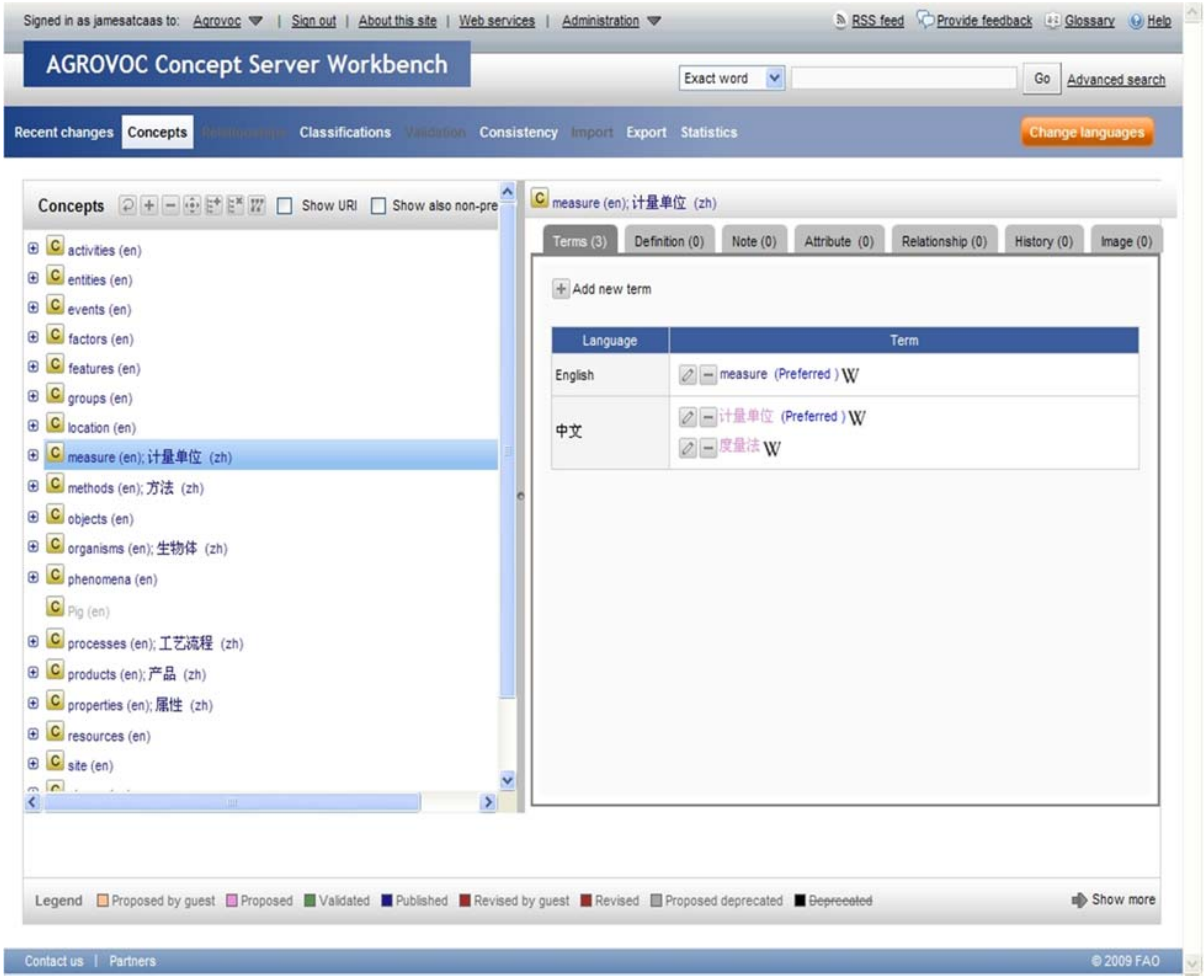
Fig. 1 The architecture of ACSW (Yongyuth *et al*. 2008).

information for sharing with the other editors, terms that related to the concepts for supporting multi-lingual aspect, relationship between the selected concept to other concepts, and also includes the image that associated to the concept.

Relationship management Editors can use this module to add, edit or delete relationships that were used in this system. The relationship hierarchy consists of 2 types of relationship properties (object property and data type property). In case of adding new relationship, they can also add more related information to that relationship, such as the label and definition of the relationship in any language, the properties of relationship (such as symmetric, transitive and inverse functional), the domain and range which explicate the boundary of subject and object of that relationship.

To make sure that the ontology within ACSW has high quality and no logical errors, this module also provides Consistency check function for checking whether some ontology parts are inconsistency, and Validation function is used to help ontology expert (may be validator or publisher) to approve the actions made by ontology editors before releasing the updates to the public.

In addition, there are also other very useful modules. For example, Search function is used to help users to locate the target concept; Import function enables to import external ontology in OWL format that has the same schema compared to the system. The system will alert in case of duplication; Export function enables to export ontology to RDF, SKOS; Scheme management function is used for grouping concept. Fig. 2 is the screenshot of ACSW after logged in as ontology editor.



- activities (en)
- entities (en)
- events (en)
- factors (en)
- features (en)
- groups (en)
- location (en)
- measure (en); 计量单位 (zh)
- methods (en); 方法 (zh)
- objects (en)
- organisms (en); 生物体 (zh)
- phenomena (en)
- Pig (en)
- processes (en); 工艺流程 (zh)
- products (en); 产品 (zh)
- properties (en); 属性 (zh)
- resources (en)
- site (en)

Language	Term
English	measure (Preferred) W
中文	计量单位 (Preferred) W 度量法 W

Fig. 2 The user interface of ACSW.

COLLABORATIVE Protégés

Collaborative Protégé was developed by the Protégé team at the Stanford Center for Biomedical Informatics Research, USA. Collaborative Protégé is an extension of the existing Protégé system (http://protege.stanford.edu/index.html). It enables users to hold discussions or chat, and annotates ontology components and changes – all as an integral part of the ontology-development process.

Architecture of Collaborative Protégé

Collaborative Protégé is based on client-server architecture, as shown in Fig. 3.

On the Protégé server side there is an ontology repository that contains all the ontologies, Protégé clients can edit collaboratively. The repository uses ChAO (Changes & Annotation Ontology) (Noy *et al.* 2006) knowledge bases for each domain ontologies, which represent the changes and the annotations for the corresponding ontology. Different ontologies can share the same ChAO knowledge base.

The server side also provides Java APIs. The changes API supports getting the structured log of ontology changes in detail; the Annotations API is used for adding annotations to ontology components and changes; the Ontology Components API provides common methods for the Changes and Annotations APIs and also supports the access to the ontology components kept in the ChAO knowledge bases; the Ontology API supports accessing and changing the content of the ontologies and knowledge bases.

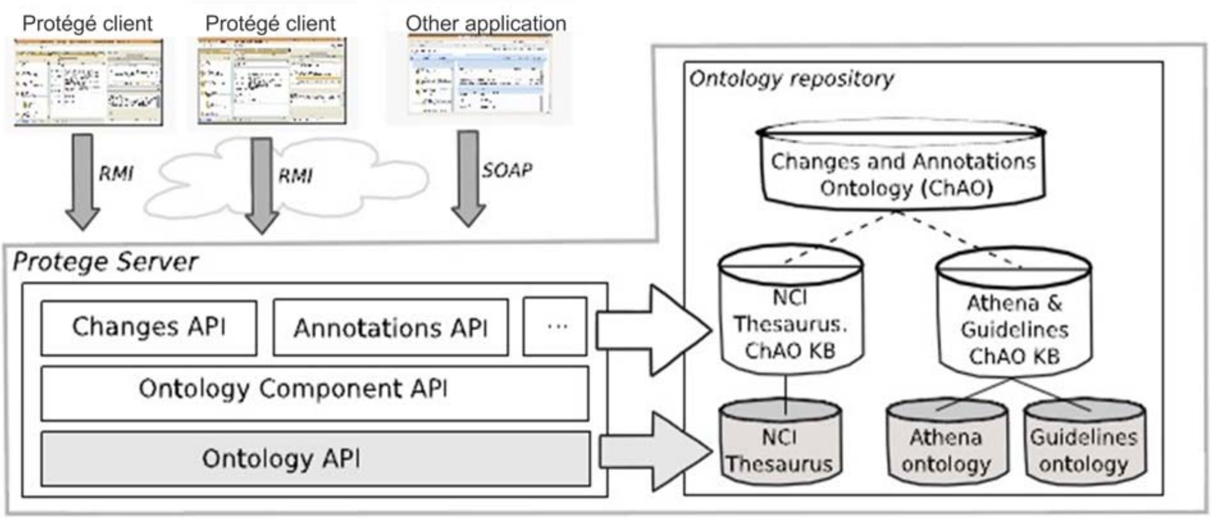


Fig. 3 The architecture of Collaborative Protégé (Tudorache *et al*. 2008).

As an extension to the Protégé, Collaborative Protégé can run in a client-server environment. Ontologies are stored in a central Protégé server. Ontology editors access the ontologies in the server to browse and edit them through desktop or web Protégé clients. The client-server mode uses the Remote Method Invocation (RMI) mechanism of Java. Other applications such as WebProtégé use the Simple Object Access Protocol (SOAP) to interact with the server side.

Main functionalities of Collaborative Protégés

As an extension of the existing Protégé system, Collaborative Protégé inherits many useful functions from Protégé, such as supports browsing, adding, editing, deleting for the ontology components (e.g., class, property, instance, etc.). Besides, it also has some special functionalities:

Metaproject management, which is a frame-based ontology that models permissions. Several classes are designed such as User, Group, Operation, GroupOperation, and Project (http://protegewiki.stanford.edu/wiki/Protege_Client_Server_Tutorial_Configuration#The_Metaproject). It contains information about which projects are available to Protégé clients, which users have access to these projects, what operations users can make. Administrators can manage the metadata in the metaproject, such as adding new projects, creating users and permission groups (roles), authorizing users with certain roles and projects.

Annotation module supports browsing the annotations that are attached to the selected class, property or individual, and creating annotations of specific type (Comment, Question, Example, and Proposal, etc.) to these components or the ontology as a whole. Annotations may be filtered and searched based on different criteria such as author, date, type, and body of annotation.

Discussion module, much like the annotation function, has a similar user interface and features, but the annotations in this part are attached to ontology itself rather than to ontology components. For example, editors can discuss modeling patterns, or name conventions that are broader in scope and that should apply to the whole ontology.

Change tracking module supports tracking change history of particular ontology component. Users can get the change details (author, date, sub-changes, etc.), they can create comment and discussion threads related to a certain change. Furthermore, they can also search all annotations based on different criteria.

Chat module supports exchanging live messages among users connected to the same Protégé server, sending HTML format messages, and also can send internal links that point to ontology components. Other users can click on the internal link and see the definition for the class mentioned in the chat.

Fig. 4 is the main window of Collaborative Protégé while building ontology collaboratively.

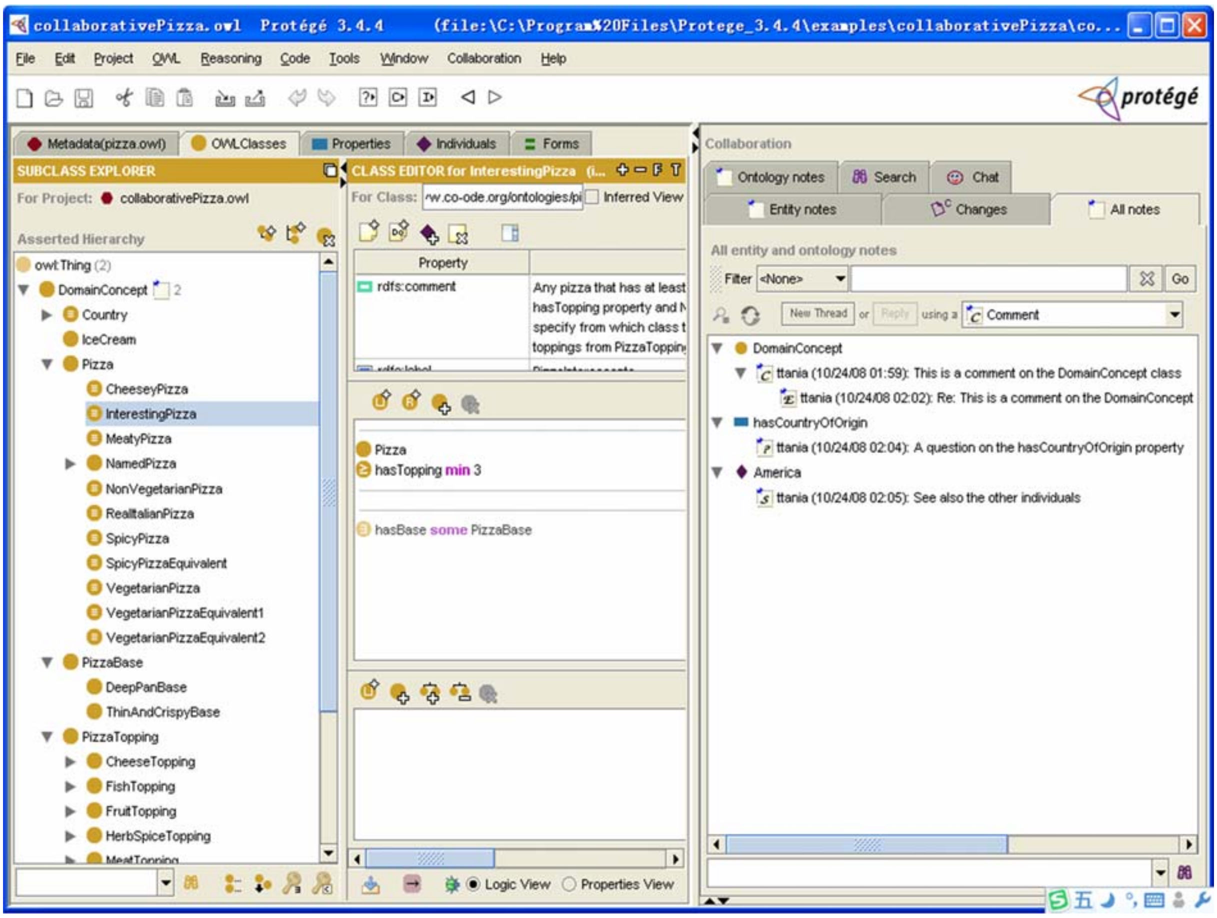
WebProtégé

WebProtégé is an open source, lightweight and web-based ontology editor (http://protegewiki.stanford.edu/wiki/WebProtegeUsersGuide), and is also developed by the Protégé team at the Stanford Center for Biomedical Informatics Research, USA. It is a web-client for Collaborative Protégé and aims to better support the collaborative development process in a web environment.

Architecture of WebProtégé

WebProtégé takes advantages of the existing components of Collaborative Protégé to support for simultaneous editing, but it is based on Browse/Server modules rather C/S, as shown in the Fig. 5.

The Remote Procedure Calls (RPC) server module listens to and makes response to the request from the client side. The server provides ontology access services such as reading and writing OWL ontologies through the Ontology API. It also provides support for collaboration services, such as annotation of ontology components and change tracking through Collaboration API and ChAO. Another important task is to keep track of the changes in the ontology with version number and to manage conflicts when different clients make changes to the same ontology. The client side includes the user interface implemented with Google Web Toolkit (GWT), an internal model of the ontology that represents the ontology view of the client, and the RPC client module to communicate with the server. The Server side is completely implemented in Java and makes API calls to Protégé backend, client side is developed in Java, and compiled into JavaScript at run-time. The communication between server client is finished with GWT RPC (JSON).



ttania (10/24/08 01:59): This is a comment on the DomainConcept class

ttania (10/24/08 02:02): Re: This is a comment on the DomainConcept

ttania (10/24/08 02:04): A question on the hasCountryOfOrigin property

ttania (10/24/08 02:05): See also the other individuals

Fig. 4 The user interface of Collaborative Protégé.

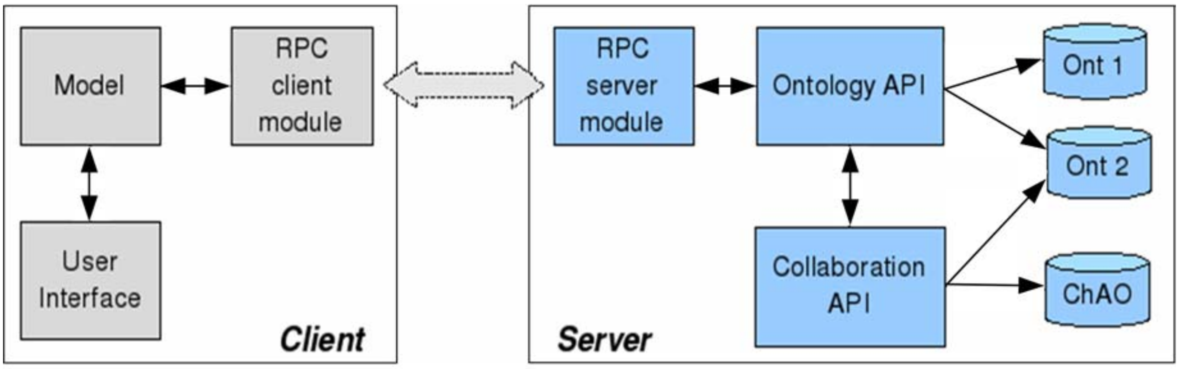


Fig. 5 The architecture of WebProtégé.

Main functionalities of WebProtégé

As a lightweight and web-based ontology editor, WebProtégé has some basic functionalities:

Browsing ontologies on the web The layout is similar as the Protégé, a navigating tree for classes, properties and individuals on the left, and more detail information of the selected ontology components on the right region. It is very convenient to view ontology with internet browsers, not so many limitations as the Protégé client.

Lightweight ontology editing Users can create and delete classes, properties and individuals after logging in. They can also track the changes, and add, reply and delete notes to these components.

WebProtégé has some other features. It is designed using a portal metaphor: the user interface is composed of reusable components, including tabs (e.g., Classes tab, Property tab, etc.) and each tab contains several portlets (e.g., the class tree, property values, restriction view, notes, etc.). Users can customize the layout such as enable or disable tabs, resize and rearrange the portlets in a tab by drag-n-drop.

Fig. 6 shows the operation window of WebProtégé in web browser.

COMPARISONS

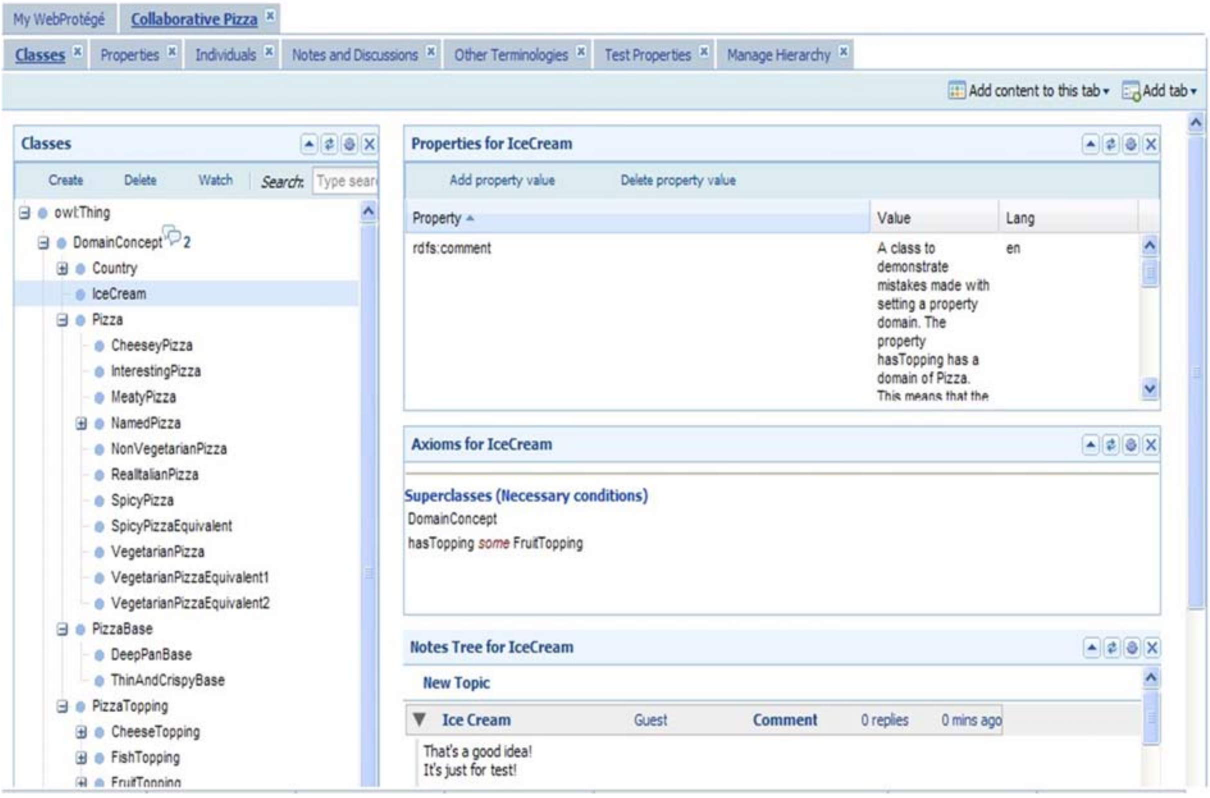
The comparisons among the three tools from several aspects are shown in the Table.

Table shows that these tools all support collaborative ontology editing in different ways. Both Collaborative Protégé and WebProtégé are open source, and can be downloaded and deployed locally for free, ACSW can only be accessed online. Compared to the Collaborative Protégé's C/S architecture, ACSW and WebProtégé are more flexible for them based on B/S mode. As for administration management, such as managing users, roles are conducted by administrators at back in ACSW, while both Collaborative Protégé and WebProtégé have the metaprojects to realize that, the difference is that Collaborative Protégé supports managing metaprojects itself and changes will take effect immediately, and WebProtégé is built on Protégé's core APIs and the changes will take effect only after restart the web server container (e.g., Tomcat).

As for managing ontology components, ACSW supports the browsing, creating, updating, and deleting of Concept, Relationship and Schema (for grouping concepts) and their detail information as well, it has some important features such as supporting multilingual; Collaborative Protégé has the richest functions because it inherits most features from protégé; WebProtégé has limited function just for light-weight ontology editing. They all adopt some quality control strategy such as Consistency check. ACSW has one more measure-validation (performed by validator and publisher before releasing any updates to the public).

At the aspect of collaborative editing, ACSW only allows users make some notes for each element; both Collaborative Protégé and WebProtégé adopt ChAO Knowledge base as to save typed annotations to ontology components and changes, Collaborative Protégé also supports changes tracking and online chatting. In addition, the two latter provide supports for simultaneous editing, which means that a change made by a user is immediately seen by the other users. To some extends, none of them have clear and flexible workflow.

Both ACSW and Collaborative Protégé support to visualize ontology components' architecture and relationships graphically. Collaborative Protégé has the advanced query function with Queries tab and Annotation filtered search. ACSW supports Basic search and Advance search. WebProtégé only supports the basic type string search function. ACSW supports importing external ontology in OWL format and exporting to RDF, SKOS; Collaborative Protégé uses Ontology repository manager to import the external ontologies, which so far supports five kinds of ontology repositories: Local folder, Relative folder, Local file, Relative file, HTTP repositories, Database repositories (http://protege.stanford.edu/doc/owl/owl-imports.html), and export ontology to N-TRIPLE, N3, OWL, RDF schema.



Property	Value	Lang
rdfs:comment	A class to demonstrate mistakes made with setting a property domain. The property hasTopping has a domain of Pizza. This means that the	en

That's a good idea!
It's just for test!

Fig. 6 The user interface of WebProtégé.

Table Comparisons among ACSW, Collaborative Protégé and WebProtégé

Features	ACSW	Collaborative Protégé	WebProtégé
Availability	No	Yes	Yes
Architecture	Browser/Server	Client/Server	Browser/Server
Administration	User, group, role	Metaproject	Metaproject
Basic operations for ontology Components	Browse, create, update, delete (Concept, relationship, scheme, classification)	Browse, create, update, delete (class, property, individuals)	Browse, create, delete (class, property, individual)
Annotations and change history	Notes, history	ChAO knowledge base, change tracking	ChAO knowledge base
Discussions	No	Discussion thread, online chat	Notes and discussions
Quality control	Consistency check, validation	Consistency check	Consistency check
Simultaneous edit	No	Yes	Yes
Search	Basic and advance search	Queries tab, Annotation filtered	Simple search
Visualization	TouchGraph	OWLVizTa, TGVizTa, OntoTGVizTa	No
Import	OWL	Ontology repository manager	No
Export	RDF, SKOS	N-TRIPLE, N3, OWL, RDFS	No

USERS' FEEDBACK AND COMMENTS

So far, about 33 users have registered in ACSW after its' newest release VocBench 1.1 in April 2011 (http://agrovoc.mimos.my/vocbenchv1.1i/), including 26 users for English language and 3 for Chinese, the new version has better user experience and responds more quickly than ever before. It supports a growing set of user communities, including the global, distributed group of terminologists who manage AGROVOC. More than 50 comments or suggestions have been posted by users and developers.

As Protégé has large user community, e.g., 176 656 registered users, more and more ontology developers begin to use its extension Collaborative Protégé. Many users think that this extended part is very useful and is an excellent work. WebProtégé 0.5 beta was released on May 28, 2011. WebProtégé has been used to support World Health Organization (WHO) to revise the International Classification of Diseases (ICD-11) (Falconer *et al.* 2010). Participants in the project learned how to use the platform quickly and they indicated that the current functionality is easy to use. However, users expect more import features, such as change tracking, better hierarchy management and the multilingual support.

CONCLUSION AND FUTURE WORK

ACSW is a web-based working environment for managing the AGROVOC and featured in collaborative editing of multilingual terminology and semantic concept information. Collaborative Protégé and WebProtégé inherit a lot of features from Protégé, and they have other features such annotations, discussions and simultaneously editing. However, several important improvements need to be made in the future, such as supporting flexible workflows, concurrency control, evolution and version management, scalability for large ontologies, multilingual ontology, ontology mapping and merging, multilingual user interface, and so on.

Agricultural Information Institution (AII) of Chinese Academy of Agricultural Sciences (CAAS) is one of the earliest domestic institutes to carry out related research work on agricultural ontology, such as converted CAT to a light-weight agricultural ontology in RDF(S) and OWL formats, cooperated with FAO on AOS concepts and ideas, mapped the AGRVOC to CAT (Liang *et al.* 2005; Liang and Sini 2006), joined in the whole test process of ACSW and so on. However, we have not yet organized large number of knowledge engineers and domain experts together to build agricultural ontology in large scale collaboratively. It is very necessary that we launch such a significant work with the strong supports of ACSW, Collaborative Protégé, WebProtégé. Besides, it is urgent to design and develop our own collaborative ontology building platform for Chinese users.

Acknowledgements

We would like to give special thanks for Dr. Meng Xianxue, Prof. Qian Ping and Dr. Zhang Xuefu, Chinese Academy of Agricultural Sciences, for their suggestions and comments to this paper. This project was supported by the National Key Technology Research and Development Program of China (2011BAH10B06).

References

Auer S, Dietzold S, Riechert T. 2006. OntoWiki: a tool for social, semantic collaboration. In: *Proceedings of 5th International Semantic Web Conference, ISWC*. vol. LNCS 4273. Springer, Athens, GA.

Falconer S M, Tudorache T, Nyulas C, Noy N F, Musen M A. 2010. WebProtégé: Supporting the creation of ICD-11. [2010-10-22]. http://iswc2010.semanticweb.org/pdf/502.pdf

Farquhar A, Fikes R, Rice J. 2003. The Ontolingua server: a tool for collaborative ontology construction. [2010-10-25]. http://www.agent.ai/doc/upload/200302/farq97_1.pdf

Finin T, Joshi A, Kagal L, Niu J, Sandhu R, Winsborough W, Thuraisingham B. 2008. Rowlbac: role based access control in owl. In: *SACMAT '08 13th ACM Symposium on Access Control Models and Technologies*. June 11-13, 2008. Estes Park, CO, USA.

Haase P, Lewen H, Studer R, Tran T, Erdmann M, d'Aquin M, Motta E. 2008. The neOn ontology engineering toolkit. In: *Demo Session at WWW2008*. April 21-25, 2008. Beijing, China.

Liang A C, Sini M. 2006. Mapping AGROVOC and the Chinese agricultural thesaurus: definitions, tools, procedures. *New Review of Hypermedia and Multimedia*, **12**, 51-62.

Liang A, Sini M, Chang C, Li S J, Lu W L, He C P, Keizer J. 2005. The mapping schema from Chinese agricultural thesaurus to AGROVOC. [2010-11-02]. ftp://ftp.fao.org/docrep/fao/008/af241e/af241e00.pdf

Noy N F, Chugh A, Liu W, Musen M A. 2006. A framework for ontology evolution in collaborative environments. In: *Proceedings of 5th International Semantic Web Conference, ISWC.* vol. LNCS 4273. Springer, Athens, GA, 2006.

Sini M, Lauser B, Salokhe G, Keizer J, Katz S. 2007. The AGROVOC Concept Server: rationale, goals and usage. [2010-09-22]. ftp://ftp.fao.org/docrep/fao/010/ai167e/ai167e00.pdf

Sini M, Rajbhandari S, Amirhosseini M, Johannsen G, Morshed A, Keizer J. 2010. The AGROVOC Concept Server Workbench System: Empowering management of agricultural vocabularies with semantics. In: *IAALD XIIIth World Congress, Montpellier (France), 26-29 April 2010*. [2010-09-22]. http://www.fao.org/docrep/012/al055e/al055e00.pdf

Tudorache T, Noy N F, Musen M A. 2008. Supporting collaborative ontology development in Protégé. In: *Proceedings of 7th International Semantic Web Conference, ISWC 2008*. October 26-30, 2008. Karlsruhe, Germany.

Tudorache T, Noy N F. 2007. Collaborative Protégé. In: *Proceedings of WWW2007*. May 8-12, 2007. Banff, Canada.

Tudorache T, Vendetti J, Noy N F. 2008. Web-protégé: a lightweight OWL ontology editor for the web. In: *Proceedings of 5th OWL Experiences and Directions Workshop (OWLED 2008*). [2010-9-12]. http://www.webont.org/owled/2008/papers/owled.2008eu_submission_40.pdf

Yongyuth P, Thamvijit1 D, Suksangsri T, Kawtrakul A, Rajbhandari S, Sini M, Keizer J. 2008. The AGROVOC Concept Server Workbench: A collaborative tool for managing multilingual knowledge. In: *IAALD AFITA WCCA2008, World Conference on Agricultural Information and IT*. August 24-27, 2008, Tokyo, Japan.

(Managing editor ZHANG Juan)