The HCOME-3O Framework for Supporting the Collaborative Engineering of Evolving Ontologies

George A. Vouros[1], Konstantinos Kotis[1], Christos Chalkiopoulos[1], Nikoleta Lelli[1]

[1] University of the Aegean, Dept. of Information & Communications Systems Engineering,
AI Lab,
83200 Karlovassi, Greece
{georgev, kotis}@aegean.gr
http://www.icsd.aegean.gr/ai-lab

Abstract. Nowadays it is widely accepted that ontologies, the key technology for the realization of the Semantic Web, are artefacts that are collaboratively and iteratively developed/evolved, shared, evaluated and discussed within communities of knowledge workers. To enhance the potential of ontologies to be collaboratively engineered and be consistently evolved within and between different communities, they must be escorted with rich meta-information describing the conceptualisations they realize, implementation decisions, the rationale for their evolution, as well as the evolution itself. To support the collaborative engineering of ontologies within and across different communities, this paper proposes a framework of (meta-)ontologies for capturing the meta-information that is necessary for interlinking, sharing, and combining knowledge among the parties involved in such a process. The framework is being embedded in the HCOME ontology engineering methodology, and can be applied to the design and implementation of ontology engineering tools towards advancing their interoperability.

1 Introduction

Ontologies establish a common vocabulary for community members to interlink, combine and communicate knowledge shaped through practice and interaction, binding the knowledge processes of creating, importing, capturing, retrieving, and using knowledge [11]. The ontology engineering process itself involves knowledge-intensive activities performed by members of specific communities. People participating in such a process need to share a common understanding of the various aspects and issues involved i.e. domain, methodological and tool-related ones. Therefore, (meta-)ontologies can play a major role in interlinking, sharing and combining information among the parties involved in a collaborative ontology engineering process.

We distinguish between domain knowledge and development information involved in the ontology engineering process. Domain knowledge concerns the conceptualization(s) that knowledge workers shape in order to develop a domain-specific ontology. Development information concerns a) the language-specific aspects for formalizing conceptualizations b) the interlinking of the conceptualizations with domain-related resources and collaborating parties, c) the recording of developers' rationale on choosing specific conceptualizations and ways of formalizing them, and d) the ontology evolution i.e. the changes performed on (informal or formal) conceptualizations and the clustering of these changes in different versions of a domain ontology.

This paper focuses on the formal specification of development information in order to support advanced collaborative ontology engineering processes for the specification of continuously evolving domain knowledge.

Recent ontology engineering methodologies (HCOME [5], DILIGENT [10]) emphasize on (a) the incorporation of ontology engineering tasks in knowledge-empowered organizations in ways that are seamless to the day-to-day activities of the organization members and on (b) the active and decisive involvement of the knowledge workers in all stages of the ontology engineering processes. Particularly, the HCOME methodology accentuates the active and decisive participation of knowledge workers in the ontology life-cycle. Doing so, domain ontologies are developed and managed according to knowledge workers' abilities, they are developed individually as well as conversationally, and they are put in the context of workers' experiences and working settings, as an integrated part of workers' "knowing" process. Besides the methodological issues, leveraging the role of knowledge workers in the ontology life-cycle entails the development of ontology engineering tools that provide greater opportunities for them to manage and interact with their conceptualizations in a direct and continuous way, not only by reusing and combining domain/development knowledge but also communicating such knowledge between them effectively.

This paper points that to empower knowledge workers to actively and decisively participate in the ontology life-cycle, we need to establish a common understanding of (or at least make explicit to them) the way(s) that ontologies are being implemented and evolved. Towards this target, this paper proposes a framework of (meta-)ontologies for capturing the development information that is necessary for interlinking, sharing and combining knowledge among the parties involved in a collaborative ontology engineering process. This framework is embedded in the HCOME methodology, advancing the potential for collaborative ontology engineering tasks, and the interoperability of ontology engineering tools by applying it to their design and implementation.

This paper is structured as follows: Section 2 provides the motivation and the work that is closely related to the aims of our work and section 3 presents the proposed framework. Section 4 presents preliminary evaluation of the framework using a collaborative ontology engineering tool, showing its potential to satisfy the stated requirements.

2 Motivation and Related Work

Knowledge workers within and across communities, even if they are interested in the same domain, may not share the same context. The context includes the background knowledge that community members have, their commonly accepted practices, their experiences concerning the domain of interest, their interests and motivation to exploiting ontologies, as well as the ontology exploitation tools/applications they use.

More important to the exploitation and evolution of living ontologies, communities may not have the same view of how and why domain ontologies have been developed and/or evolved in the way they did, and they may not even use the same tool or methodology to engineer them. Therefore, (meta-)ontologies, besides facilitating a common understanding of the issues involved in the ontology engineering task (which is essential for people working with different ontology engineering methodologies to communicate), they also provide a common vocabulary for sharing information concerning the development of domain ontologies (which is essential for different ontology engineering tools to interoperate), and specific information concerning their evolution (which is essential for people to inspect and assess the changes made to domain ontologies).

(Meta-)ontologies must support the sharing, reuse and consistent evolution of domain ontologies within and across communities. This implies the need for the extended sharing of the constructed domain ontologies, together with *formal* specification of meta-information that would support the interlinking, combination, and communication of knowledge shaped through practice and interaction among community members.

Ontologies for the specification of such meta-information must support:

1. The identification of those parties that contribute to the development/evolution of a single ontology.
2. The recording of the conversations towards the commonly agreed requirements and scope of the ontology.
3. The tracking of the arguments towards the agreed (formal or informal) specifications
4. Tracking the change operations performed by individual users
5. Capturing the informal meaning of ontology elements by interlinking formal specifications to other domain resources (e.g. thesaurus, lexicons).
6. The specification of change operations that have occurred between two subsequent ontology versions.
7. Structured argumentation dialogues for the evaluation and further development/evolution of shared ontologies.
8. Integration of versioning and change-tracking information with argumentation dialogues, for the effective sharing of ontologies: This enables tracking the rationale behind individual changes, ontology versions, specification and implementation decisions.
9. The inter-contextual sharing of domain ontologies: Although previous work has emphasized on the sharing of ontologies within specific contexts, meta-information must support the inter-contextual sharing of ontologies, capturing all the detailed aspects involved in the development/evolution of ontologies, either in a personal or in a shared space.

The above requirements for meta-information point to the need of an integrated framework of (meta-)ontologies for the intertwined specification of (a) structured argumentation dialogues, (b) change operations and ontology versions during ontology evolution, (c) administrative information concerning domain conceptualization and ontologies implementations, contributors involved in ontology lifecycle, and relations to other domain-related resources.

Viewing this framework in the context of a specific collaborative ontology engineering methodology, it aims to advance the potential of reusing and consistently evolving formal conceptualizations of domain knowledge. We view this as an essential requirement to the use of such a framework of meta-information as it assures that the framework facilitates the ontology engineering process and advances the understanding of methodological issues involved in the engineering of ontologies (i.e. what does a collaborative ontology engineering process involves, who may participate and what is expected/permitted to contribute, what changes are expected to be made, how versions are being assimilated, the degree to which these changes/versions have to be justified). Specifically, we aim to advance the HCOME methodology by incorporating a framework of (meta-)ontologies to support the collaborative ontology engineering process. HCOME has accentuated the need for advanced functionality for engineering shared and continuously evolving ontologies. HCOME places major emphasis to the conversational development, evaluation and evolution of ontologies. This implies the need for the extended sharing of the constructed domain ontologies, together with meta-information that supports the interlinking, combination, and communication of knowledge shaped through practice and interaction among community members, binding the knowledge processes of creating, importing, capturing, retrieving, and using knowledge.

In the current bibliography, there are works about (meta-)ontologies for supporting specific facets of the ontology engineering process (OMV [1, 2], OntoView [3], CHAO [9], DILIGENT [12]): These works do not provide an integrated solution to the problem of knowledge exchange, so as to support the collaborative engineering and consistent evolution of ontologies within and across different communities of knowledge workers. Although they do specify types of information that need to be captured for describing the conceptualization and development of domain ontologies, they do not specify a unique integrated conceptual framework for capturing and sharing this information, and neither specify how such a framework is embedded within an ontology engineering methodology.

Related work concerning ontology evolution frameworks in specific, has been proposed in [9], using the Change and Annotation Ontology (CHAO). Instances of this ontology represent changes between two versions of an ontology. Changes are linked to annotations. For each change, the change and annotation ontology describes the following information: the type of change; the class, property, or instance that was changed; the user who performed the change; the date and time when the change was performed. Although annotations on changes are being recorded, the arguments supporting and/or being against individual changes are not captured, affecting the effectiveness of the representation for recording the rationale and different views/opinions behind individual changes and/or the issuing of assimilated ontology versions.

Other works [6, 7, and 8] provide information concerning ontology change management in different levels of abstractions (simple or complex changes, collections of changes (versions), changes discovered from similarity measures, etc). However, although annotations on changes are being recorded, arguments are not captured and are not interrelated with other meta-information.

Similarly to the ontology-evolution framework proposed in [9], Figure 1 presents the processes that may be performed by knowledge workers and the meta-information that must be recorded as a by-product of the collaborative ontology engineering processes according to HCOME: As it can be seen, in extend to other frameworks (e.g. in [9]), we require ontologies to be escorted with the meta-information concerning their development and evolution. This meta-information is further enriched via the processes of editing (creating, importing, capturing), exploiting (inspecting, retrieving and using) and arguing about domain knowledge. We further require that when domain ontologies or parts of them are being shared between workers, the relevant meta-information has to be shared as well.

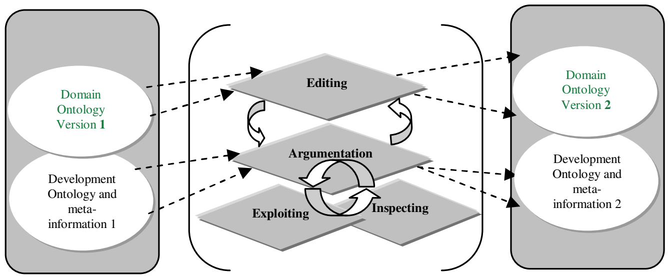


Fig. 1. Processes and meta-information in an ontology evolution cycle: Rectangles denote processes and ovals ontologies. Plain arrows point on the input and output produced: domain ontologies and individuals recorded in (meta-)ontologies.

Summarizing the above, the proposed work aims to advance the state of the art by contributing to the following issues conjunctively:

1. it provides an integrated framework of ontologies for the specification of meta-information,
2. it embeds this framework within the HCOME collaborative methodology for ontology engineering,
3. it examines the implications of adopting this framework to the design of ontology engineering tools.

3 The HCOME-3O framework

According to the stated requirements, this section presents the HCOME-3O framework of three ontologies, which specify meta-information concerning:

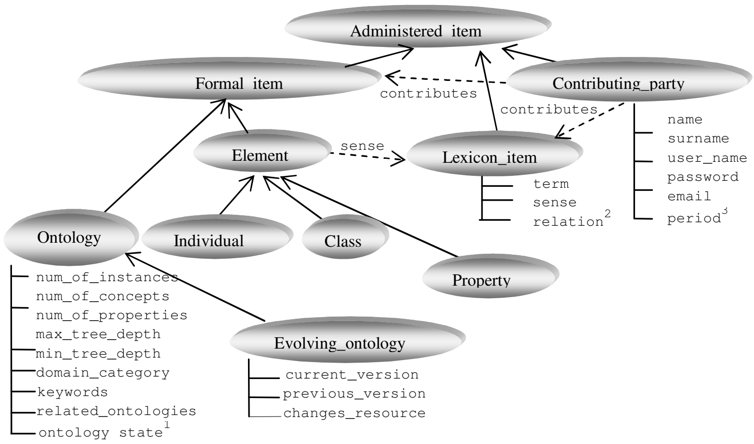
Administrative meta-information: This specifies information concerning the conceptualization, development of domain ontologies, as well as versioning of ontologies.

Change operations meta-information: This concerns changes that have been made in domain ontologies.

Argumentation meta-information: It concerns rationale and arguments related to individual changes and ontology versions.

Although ontologies such as the ones presented in section 3 could be incorporated into the framework, we have only consulted these ontologies in the engineering of the proposed framework, so as to specify the minimum meta-information that must be captured in a modular but intertwined manner, according on the stated requirements.

3.1 Administrative meta-information



1. It specifies whether an ontology is personal, shared or agreed
2. It specifies the semantic relation between the term that lexicalizes the concept and the lexicon/thesaurus term entry (e.g. synonym, more specific, more general)
3. A period starts when a personal ontology is send to the shared space and ends when a version of this ontology is in the agreed state.

Fig. 2. *A*dministrative meta-information

The Administration ontology provides a schema for representing meta-information about administered items and contributing parties. `Administered items` can be either ontologies, ontology elements (classes, properties, individuals), or items that informally describe the meaning of terms that lexicalize properties or classes in the domain ontology. All types of items are identified by a resource identifier. `Formal items` and `lexicon items` are contributed by `contributing parties`. Lexicon items may also be automatically assigned by mapping algorithms. Contributing parties may contribute to the development/evolution of a personal, shared or agreed `ontology`, or may contribute to the specification of a `class`, `property` or `individual`. Also, an ontology can have several uniquely identified versions, which result from the changes made and recorded during ontology development/evolution.

The administrative ontology distinguishes between the informal and formal conceptualization of a domain by linking items to the informal (lexicon-based) description of their meaning: This distinction is further supported by linking items to argumentation items (of the argumentation dialogue) that provide arguments for the conceptualizations/specifications made. In this way, administrative meta-information is documentary and extensible, and supports the interlinking with other domain-specific resources. The entities of the implemented proposed schema and their relations are depicted in Figure 2.

3.2 Change operations meta-information

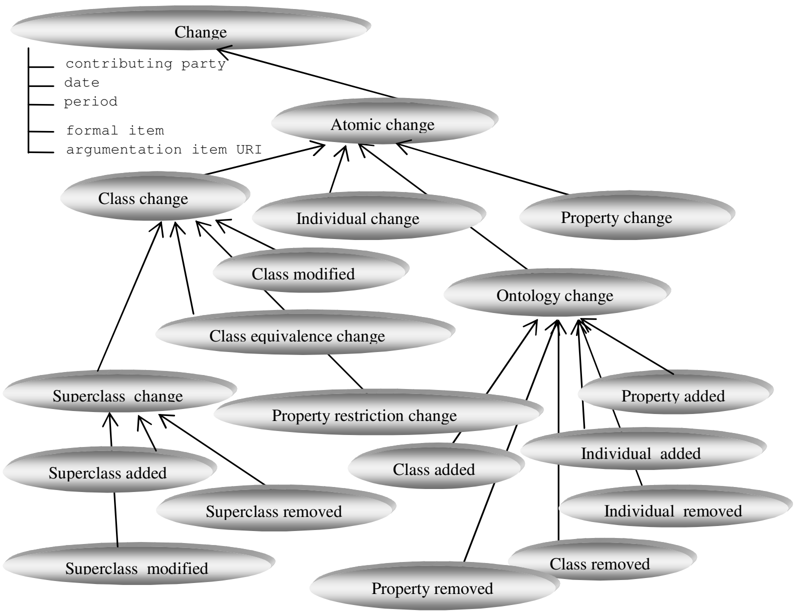


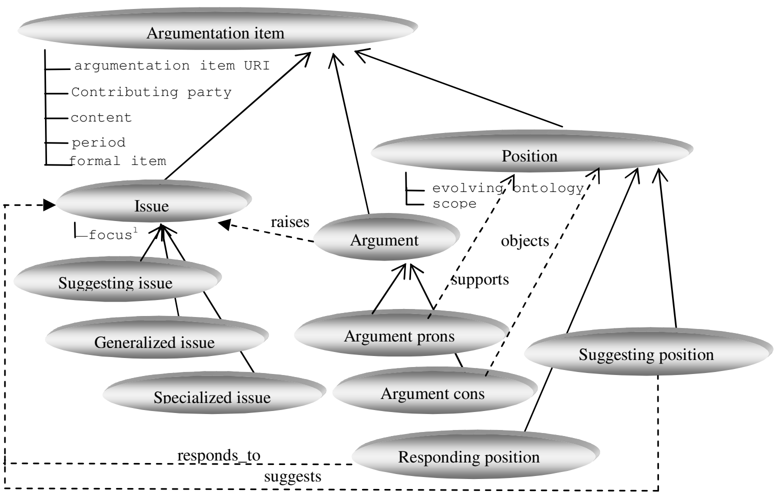
Fig. 3. Meta-information concerning changes that occur during the editing of ontologies

The change operations (meta-)ontology provides a schema for representing information about the `changes` that contributing parties can make to the ontology elements during the evolution of a domain ontology. It also supports the reporting of differences between two versions of a single ontology.

This ontology currently specifies only atomic changes: Any atomic change to the specification of a formal element (Class, Property, and Individual) made during the editing of an ontology is recorded together with the rationale behind it. The relations between a change made by a contributed party, the argumentation items (if any) behind this change, and the element that has been changed, are specified by means of the `Atomic change` class properties (contributing party, argumentation item, formal item).

Figure 3 depicts only a part of the ontology. Change operations that can apply to individuals and properties are missing due to space restrictions.

3.3 Argumentation meta-information



1. It specifies whether the issue concerns the conceptualization of the domain or formal aspects of ontology implementation
2. It specifies whether the position concerns the ontology version or the change of an ontology element

Fig. 4. Information captured in an argumentation dialogues.

The argumentation ontology provides a schema for representing meta-information about *`issues`*, *`positions`*, and *`arguments`* that contributing parties make during an argumentation dialogue upon the collaborative evolution of shared ontologies.

Specifically, an argument may raise an issue that either suggests changes in the domain conceptualization, or questions the implementation of the conceptualized entities/properties. Based on this issue, a collaborative party may respond by publicizing a position, i.e. a new version of the ontology, or by suggesting the change of a specific ontology element. A new argument may be placed for or against a position, and so on. Issues may be generalized or specialized by other issues. The connection of the recorded arguments with the ontology elements discussed by specific contributing parties and with the changes made during a period (Figure 3) is performed through the argumentation item and position classes' properties (formal item, contributing party, period, evolving ontology).

The argumentation ontology supports the capturing of the structure of the entire argumentation dialogue as it evolves among collaborating parties within a period. It allows the tracking and the rationale behind atomic changes and/or ontology versions. It is generic and simple enough so as to support argumentation on the conceptual and on the formal aspects of an ontology.

The entities of the implemented proposed schema and their relations are depicted in Figure 4.

4 Preliminary Evaluation

Early evaluation of the proposed framework has been performed by embedding it in a prototype version of HCONE tool [4]. This version was designed by taking into account the requirements of the proposed framework in addition to the methodological requirements of HCOME methodology. Having said that, it must be clearly stated that in this paper we do not point on the value of a collaborative engineering methodology itself. The contribution and importance of collaborative engineering of ontologies has been studied in other related works [5, 10] and evaluated in [13, 14].

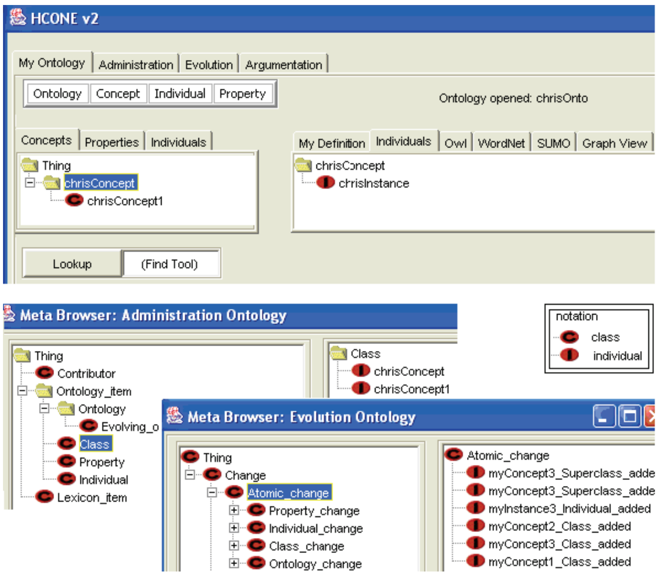


Fig. 5. The HCONE GUI and the meta-browsers windows for exploring the recorded meta-information

An early snapshot of the new HCONE tool is depicted in Figure 5, showing the supported functionalities. In the same figure, the meta-browser windows for navigating through the recorded administrative and evolution meta-information (populated meta-ontologies) are also depicted.

Embedding the proposed framework in HCONE tool allows the recording and presentation of ontologies' development information. This information is recorded as instances of OWL[1]-implemented (meta-)ontologies and is stored in a triples-like RDF[2] store (JENA[3]). The value of the framework in the engineering of shared and evolving ontologies can be measured by the capability of the proposed-framework-based tool to record and present ontologies' development information. Figure 5 shows a snapshot of the recorded meta-information, as it can be explored using the HCONE tool ontology-browser tools.

Prototype implementation has demonstrated that the proposed framework is capable of recording and presenting the following meta-information:

1. Meta-information concerning the parties that contributes to the development/evolution of a single ontology. Such information is recorded as individuals of class "Contributor" in the Administration (meta-)ontology. "Contributor" individuals are related to ontology development and their evolution ("Formal Item" class of Administation meta-ontology) through "contributes" property. Individuals are returned from queries executed over the Administration (meta-)ontology e.g. "*Find all contributors (individuals of "Contributor" class) which contribute to ("contributes" property) the development of "myOntology" ontology (individual of Ontology class)* ".
2. Meta-information concerning the recording and tracking of the conversations. Such information is recorded as individuals of "Argumentation Item" class (specifically, individuals of its subclasses) of the Argumentation (meta-)ontology. "Argumentation Item" individuals are related to a specific ontology development ("Formal Item" class of Administation meta-ontology) through their "formal item" property in the Argumentation (meta-)ontology. Individuals are returned from queries executed over the Argumentation (meta-)ontology e.g. "*Find all the "Argumentation items" (individuals of all subclasses of "Argumentation item" class) which are related to a specific ontology element ("Formal item" property)*".
3. Meta-information concerning the recording of the interlinking between conversations and ontology evolution (versions of a domain ontology). Such information is recorded as a value of the "evolving ontology" property of the "Position" class of the Argumentation (meta-)ontology. This value represents the ontology version of a domain ontology that a contributor has developed in his personal space, and which is uploaded in the shared space for inspection by other contributors.
4. Meta-information concerning the tracking of change operations performed on specific ontology items by individual users. Such information is recorded as individuals of class "Change" (specifically, individuals of its subclasses) of the Evolution

[1] http://www.w3.org/TR/owl-ref/
[2] www.w3.org/**RDF/**
[3] http://jena.sourceforge.net/

(meta-)ontology. Individuals of Class "Change" are related to a specific ontology development ("Formal Item" class of Administration (meta-)ontology) through their "Formal item" property in the Evolution (meta-)ontology. Individuals are returned from queries executed over the Evolution (meta-)ontology "*Find all the changes (individuals of "Change" class subclasses) which are related to a specific domain ontology element ("Formal item" property)*".

5. Meta-information concerning the integration of versioning and change-tracking information with argumentation dialogues. Further, to enable tracking of the rationale behind individual changes, ontology versions, specification and implementation decisions, meta-information concerning the discussions upon specific ontology elements is recorded. Individuals of class "Change" are related to a specific period of discussions upon a specific ontology element through their "period" property of the Evolution (meta-)ontology. The correspondent value of "period" property of the Evolution (meta-)ontology is currently obtained in a rather mediated manner: it is obtained from the argumentation ontology and represents the period that a specific ontology element has been discussed (i.e. related to a specific argumentation item) and this ontology element has been involved into a change operation. Thus, a specific individual change will not be interlinked to an argumentation dialogue unless the ontology element that this change concerns is interlinked to an argumentation item.
6. Meta-information concerning the capture of all the detailed aspects involved in the development/evolution of ontologies, either in a personal or in a shared space. When a knowledge worker fetches an ontology version from the central ontology store, she/he gets all the related meta-information: the previous version, the change operations, the argumentation items related to these versions, administrative meta-information. This allows him/her to inspect the evolution history and decide on the exact contributions he/she has to make. To meet this requirement we have designed and implemented a central ontology repository which stores both domain and meta-information ontologies in a triple-like RDF store (Relational database). When a domain ontology version is accessed using the HCONE environment, several queries are executed against all the stored information in the database in order to load individual values of meta-ontologies concerning this particular ontology. The linking property between all the related information of a specific domain ontology version that is retrieved by these queries is the "Formal item" property which represents a unique identifier (URI) for a specific ontology or ontology element resource. A domain ontology is personal i.e. only one contributor can manage it (its creator) until it is uploaded to an argumentation dialogue for discussion. In this case the property "ontology state" of class "Ontology" in the Administration (meta-)ontology takes the value "shared". If all contributors that have joined the specific dialogue "agree" on the shared ontology, the "ontology state" property is assigned the value "agreed". An "agreed" or "shared" ontology is accessible and manageable by all its contributors. HCONE utilizes the Administration meta-information in order to manage contributors' rights on accessing domain ontologies.

The preliminary evaluation of the proposed framework embedded in HCONE tool has been conducted with test ontologies in an experimental networked setting of a small group of collaborating users. Important issues such as scalability and usability of the prototype tool have been taken into account during tool design. For instance, persistent storage technology at the server-side has been used for handling the possibility of large scale ontologies. A large scale evaluation of the HCONE tool with real-world collaborative ontology engineering tasks has been planned in the near future.

5 Conclusions

To enhance the potential of evolving ontologies to be collaboratively engineered within and between different communities, we have proposed an integrated framework of three (meta-)ontologies that provide information concerning the conceptualization and the development of domain ontologies, atomic changes made by knowledge workers, long-term evolutions and argumentations behind decisions taken during the lifecycle of an ontology. This framework has been proposed in the context of HCOME collaborative engineering methodology and suggested for advancing the functionality of ontology-engineering tools, pointing to specific design issues.

Further work concerns the implementation of further advanced functionalities in the HCONE tool that will also uncover new implications as far as the HCOME-3O framework potential is concerned. More specific, meta-information that is not yet recorded and presented through the HCONE implementation concerns the capturing of the informal meaning of ontology elements by interlinking formal specifications to other domain resources (e.g. thesaurus, lexicons). Also, we must provide a more sophisticated mechanism for interlinking individual changes of ontology elements with specific argumentation items of a discussion period that have actually suggested and influence a change, not with the whole discussions and certainly not with items that have been related with a change for some reason but eventually they did not influence the change at all. Finally, we could advance the changes operations and extend the ontology to represent more complex changes i.e. composite changes that influence more than one ontology element (atomic change).

References

1. Hartmann J., Sure Y., Haase P., Palma R., Suárez-Figueroa M.: OMV -- Ontology Metadata Vocabulary. In Chris Welty, ISWC 2005 - In Ontology Patterns for the Semantic Web (2005)
2. Hartmann J., Sure Y., Haase P., Suárez-Figueroa M., Studer R., Gómez-Pérez A., Palma R.: Ontology Metadata Vocabulary and Applications. In Robert Meersman, Zahir Tari, Pilar Herrero et al., International Conference on Ontologies, Databases and Applications of Semantics. In Workshop on Web Semantics (SWWS), Springer (2005) pp. 906-915
3. Klein M., Fensel D., Kiryakov A., Ognyanov D.: Ontology Versioning and Change Detection on the Web. EKAW 2002, 197-212
4. Kotis K. and Vouros G.: Human Centered Ontology Management with HCONE. IJCAI'03, Ontologies and Distributed Systems Workshop, Acapulco, Mexico (2003) CEUR-WS.org/Vol. 71

5. Kotis K. and Vouros G.: Human-Centered Ontology Engineering: the HCOME Methodology. International Journal of Knowledge and Information Systems (KAIS), (Published Online First: 9 Sept. 2005) Springer (2006) 10(1):109-131
6. Liang Y., Alani H., Shadbolt N. R.: Change Management: The Core Task of ontology Versioning and Evolution. In Proceedings of Postgraduate Research Conference in Electronics, Photonics, Communications and Networks, and Computing Science 2005 Lancaster, United Kingdom. (PREP 2005), pp. 221-222
7. Maedche A., Motik B., Stojanovic L., Studer R., Volz R.: Managing Multiple Ontologies and Ontology Evolution in Ontologging. In Proceedings of the Conference on Intelligent Information Processing, World Computer Congress 2002, Montreal, Canada. Kluwer Academic Publishers (2002)
8. Noy N. F., Klein M.: Tracking Complex Changes During Ontology Evolution. Third International Conference on the Semantic Web (ISWC-2004), Hiroshima, Japan
9. Noy N. F., Chugh A., Liu W., Musen M.: A Framework for Ontology Evolution in Collaborative Environments. 5th International Semantic Web Conference, Athens, GA, (2006)
10. Pinto, H. S., Staab, S., Tempich, C.: DILIGENT: Towards a fine-grained methodology for Distributed, Loosely-controlled and evolving Engineering of oNTologies. ECAI 2004: 393-397
11. Staab S, Studer R, Schnurr H, Sure Y.: Knowledge Processes and Ontologies. IEEE Intelligent Systems, (2001) pp 26-34
12. Tempich C., Pinto H. S., Sure, Y., Staab, S.: An Argumentation Ontology for DIstributed, Loosely-controlled and evolvInG Engineering processes of oNTologies (DILIGENT) In Asunción Gómez-Pérez and Jérôme Euzenat, Second European Semantic Web Conference, (ESWC 2005), volume 3532 of LNCS, pp. 241--256. Springer, Heraklion, Crete, Greece, May 2005.
13. Tempich C., Pinto H. S., Sure, Y., Vrandecic, D., Casellas, N., Casanovas, P.: Evaluating DILIGENT Ontology Engineering in a Legal Case Study In Pompeu Casanovas, Pablo Noriega, Daniele Bourcier, V.R.Benjamins, IVR 22nd World Congress - Law and Justice in a Global Society, no. B4, pp. 330--331.
14. Tempich C., Pinto H. S., Staab S.: Ontology Engineering Revisited: An Iterative Case Study. ESWC 2006: 110-124