# Human Centered Ontology Management with HCONE

**Konstantinos Kotis, George A. Vouros**
Department of Information and Communications Systems Engineering
University of the Aegean
Samos, Karlovassi, 83200 Greece
{kkot, georgev}@aegean.gr

## Abstract

The development of ontology management environments that empower communities of knowledge workers to shape their information space by being actively involved in the ontology engineering life-cycle is emerging. This is motivated by the need to incorporate ontology management tasks in knowledge-empowered organizations and the need to populate the semantic web with agents exploiting semantically annotated information items. Towards this target we aim at ontology management environments that focus to the way people interact with their conceptualisations and to the way conceptualisations are formed as part of the day-to-day activities of knowledge workers. The paper points on important issues for the development of human-centered environments for the management of ontologies and presents a prototype system, HCONE (Human Centered Ontology Management Environment), which has been designed with the aim to address these issues.

## 1. Introduction

Ontologies are formal conceptualizations of domains, describing the meaning of (abstract and concrete) domain aspects by means of concepts and their interrelations [Chandrasekaran *et al*, 1999]. Ontologies have been realized as the key technology to shaping and exploiting information for the effective management of knowledge and for the evolution of the Semantic Web. Ontologies establish a common vocabulary for community members to interlink, combine, and communicate knowledge shaped through practice and interaction among community members, binding the knowledge processes of creating, importing, capturing, retrieving, and using knowledge [Staab *et al*, 2001; Benjamins *et al*, 1998].

The incorporation of ontology management tasks in knowledge-empowered organizations can prove to be a hindrance if not done in a way that is seamless to the day-to-day activities of the community members [Vouros 2003]. Traditionally, knowledge engineers develop the ontologies that are required by knowledge workers[1], and provide these ontologies for exploitation. However, due to constant updating, changing, and evolution of ontologies, there must be a close collaboration between knowledge engineers and workers, requiring the active and decisive involvement of the latter in many stages of the ontology management processes [Stojanovic, 2002].

In conjunction to the above, the proliferation of the semantic web aims at explicating the meaning of the worldwide available information. Ontologies constitute the backbone of this effort, providing machine-exploitable semantic information for the Knowledge Web [Fensel *et al*, 2000]. But how could we possibly populate the semantic web, enabling information providers to attach semantic information to every published information item, shaping in conjunction their consensual conceptualisations? Current efforts support the semi-automatic and manual structuring of web pages using ontologies, in conjunction to ontology learning and enrichment (e.g. [Erdman *et al*, 2001; Maedche *et al*., 2000]). In conjunction to these technologies, we need tools that would empower people to develop and manage ontologies in a seamless and transparent way, without bothering with the formalities and symbol-level details of specifications.

In this paper we conjecture that in parallel to research efforts concerning methods and techniques for engineering, learning and enriching ontologies, the need for tools that would empower people to minimize the up-front knowledge engineering effort and accelerate the knowledge processes is great. Towards this aim we adopt a human-centered approach to designing and developing ontology management environments. Such environments must support "knowledge workers" to shape their information space by being actively involved in ontology management tasks throughout the ontology engineering life cycle.

[1] A knowledge worker is any member of an information production-exploitation community. Such communities may involve workers within an organization, or World Wide Web users with common interests.

To further support our conjecture for the need for human-centered tools, let us consider the following ontology management scenarios in a living organization setting:

*Scenario No 1:* Involved in a knowledge retrieval process, a worker is searching for a specific piece of information about best practices concerning the design of a product type. The retrieval tool exploits the ontology concerning product designs, but the worker can neither find the terms that she thinks to be appropriate for querying the system, nor can she get the needed information by any combination of existing terms. She soon finds out that the definitions of some terms must be changed to reflect the information related to the new case at hand. The information is there, but cannot be reached, since the ontology does not reflect the up-to-date practice of the organization. Imagine now the same case happening for five workers per day in a fast changing domain. We suggest that workers must be empowered to shape their information space, possibly working in collaboration with knowledge engineers.

*Scenario No 2:* In a knowledge use process, a worker browses, recalls existing knowledge items, and process them for further use. During this process the worker may produce derivations that should be captured as new knowledge, indexed by new terms, or combinations of existing terms. Capturing derived knowledge is very important. Empowering this worker with the proper tools for describing her conceptions formally, incorporating them in organization's information repository, submitting and sharing this information with co-workers readily, accelerates much the knowledge processes.

*Scenario No 3*: In day-to-day information creation and import tasks, workers are devising business documents, proposals, product reports, best practices, problem/fault reports, etc. Indexing such information using formal ontological commitments should be done in a seamless way by knowledge workers themselves, during authoring, allowing them to devise, expand and update their shared conceptualisations at the same time.

Summarizing the above, this paper emphasizes on devising environments that focus to the way people interact and shape their conceptualisations and to the way conceptualisations are formed as part of knowledge workers communities' day-to-day activities [Tennison et al, 2002]. The paper presents important issues for such environments and presents the overall architecture and functionalities of a human-centred environment for ontology engineering (HCONE), which has been designed and implemented, based on these requirements.

## 2. Management of ontologies

As it is widely argued and shown in the above scenarios, ontologies explicate conceptualizations that are shaped and exploited by humans during *practice*. Being part of knowledge that people possess, ontologies evolve in communities as part of *knowing* [Cook and Brown, 1999].

Ontology management in the context of communities of knowledge workers involves the development, evaluation and exploitation of conceptualizations that emerge as part of *knowing*. In particular it involves:

- *The development of individual ontologies.* People develop their own conceptualizations that may either explicate (e.g. by formalizing concepts, by taking notes about their meaning or just by naming them) or not (by storing them in the background of their minds). In their day-to-day activity people develop their conceptualizations either by improvising, by specializing/generalizing/aggregating existing concepts based on their experiences and on interaction with other community members, or by synthesizing existing conceptualizations.
- *The development of commonly agreed group ontologies*. Developing commonly agreed and understandable ontologies is a very difficult and resource-demanding task that requires members of the communities to work synergistically towards shaping the information they exploit. Working synergistically, workers map others' conceptualizations to their own and put them in the context of their own experiences. This leads to a conversation whose back-and-forth, as it is pointed in [Cook and Brown, 1999], not only results in exchanging knowledge but also in generating new knowledge.
- *The evaluation and exploitation of ontologies.* Exploitation and evaluation of ontologies as part of the day-to-day practice of communities can be considered only as part of knowing. Conceptualizations are put in practice or in the criticism of community members who, as already pointed, have to compare them with their own conceptualizations and put them in the context of their own experiences. Evaluation can result in new meanings since concepts are seen under the light on new experiences and evolving contexts.

Impediments for knowledge workers to participate actively in these tasks include their unfamiliarity with formal languages and knowledge engineering principles/methods, as well as with methods and techniques for constructing and synthesizing ontologies. Most of the existing ontology management environments have been designed and implemented for the knowledge engineer, concentrating mostly on the ontology development process at the symbol level. This implies that the deployment of these environments in organizations with limited experience in ontologies is essentially prohibited, leading communities to develop semantically-poor thesauruses for their domains, or even abandoning the trial for semantically annotating their resources, since in most of the cases they are not willing to pay the costs implied by employing knowledge engineering resources.

Leading tools for ontology engineering [ONTOWEB, 2002] employ powerful methods for manipulating ontologies but mostly at the symbol level. To empower knowledge workers to participate actively in the ontology engineering process in collaboration with knowledge

engineers, tools must enable people to improvise new ontologies, synthesize and map existing ontologies, and collaboratively develop ontologies with their co-workers, in ways that are natural for them to interact with their conceptualizations. This must happen in the background of the day-to-day knowledge intensive activities of workers, in a seamless way to their working practices, and so that the semantic validity of specifications is assured. Tools that are close to our aims at supporting knowledge workers to develop/manage ontologies, possibly in the absence of knowledge engineers, include APECKS [Tennison et al, 1999] and WebODE [Arpirez et al, 2001]. APECKS supports "living" ontologies, allowing discussion and evaluation of the evolving domain conceptualizations. WebODE offers an extensible workbench to ontology management, based on a well-defined methodology for ontology development. However, both tools lack important facilities for managing the evolution of ontologies, such as version management and ontologies libraries.

In a greater extend than existing tools, we emphasize at providing greater "opportunities" for systems and tools to be used by members of knowledge management organizational units in their day-to-day activities, proposing the design of integrated ontology management environments using human-centered design principles [Hoffman *et al*, 2002].

The paper presents a human-centered ontology management environment (HCONE) that provides advanced functionalities for editing, viewing, managing the evolution, reasoning, integrating ontologies, and mapping ontologies to upper-level (top) ontologies and lexicons, putting emphasis on the way humans interact with their conceptualisations and on the way groups reach to an agreement on domain conceptualisations. Viewing ontology management from this point, we reveal*" opportunities for using the technology that we never considered"* [Clancey, 1999], providing also insights for technological advances in managing and engineering ontologies.

## 3. Human-centered Ontology Management

Human-centered computing (HCC) is a viewpoint intent on helping us achieve the full potential of computing so that might to maximize the value of computing to society in a new age of human and machine symbiosis [Hoffman *et al*, 2002]. Joint research efforts on cognition, ergonomics, psychology, social sciences and information technologies contribute to the emergent of human-centered systems, leading to a new world of applications and services. In this new era, systems' designers have to really understand the use of systems in conjunction with the *cognitive abilities*, *contextual constraints* and *activities of users.*

It has been argued that the human-centered development of systems must take the triple of "people-machine-context" as the unit of analysis [Hoffman *et al*, 2002]. This involves studying people capacities and goals, computational mechanisms and interface capabilities, all within their appropriate contexts. Context involves organizations, activities, practices, norms and constraints that are inherent in these activities, policies, procedures, as well as devices and media that people use for communicating and doing their work.

Viewing people activities for managing conceptualizations of domains within the context of emergent communities we shall not focus on object-centered representation formalisms or on other technological issues, but put special emphasis on the physical way of interacting with these conceptualizations and on the way conceptualizations are formed by means of people interacting among themselves and practicing.

This section points on important issues for the development of human-centered environments for the management of ontologies. The list is by no means complete. However it points on issues that we consider to be the most important:

- *Provide the tools needed for the management of ontologies in an eclectic way.* Members of the communities must be allowed to follow any approach or combination of approaches for the development of ontologies, which better fits their practice, their working norms and constraints: They may improvise by integrating concepts in a conceptual system, provide concepts with informal definitions attaching information items to them, compare, merge and refine/generalize existing ontologies. In this respect, version management and libraries of ontologies must provide people with the ability to keep track of changes in their conceptualizations.
- *Support people to communicate their conceptualizations without dealing with low-level implementation details.* The major issue for human-centered ontology management is that low level implementation details must be hidden from people who do not understand knowledge representation formalisms' terminology. People must be given the power to express subtle ontological distinctions in a way that is natural to them but satisfies the formal constraints of the specifications too.
- *Provide powerful reasoning services for checking specifications' consistency and coherency.* This is important for people to develop well-formed ontologies (i.e. ontologies without inconsistencies among the defined concepts, with coherency, and well organized) and to compare/map existing ontologies. Reasoning services must perform in the background, providing sufficient information to people about the well formed-ness of their ontologies. It is very important to notice that these services must not prohibit people to follow any approach to ontology development. On the contrary, feedback from the reasoning services must be provided in the form of help and advice about the validity and consistency of specifications, with respect to people cognitive capacities.
- *Provide facilities for exchanging and creating ontologies conversationally*. The aim is to support conversations between individuals in order to enable criticism and

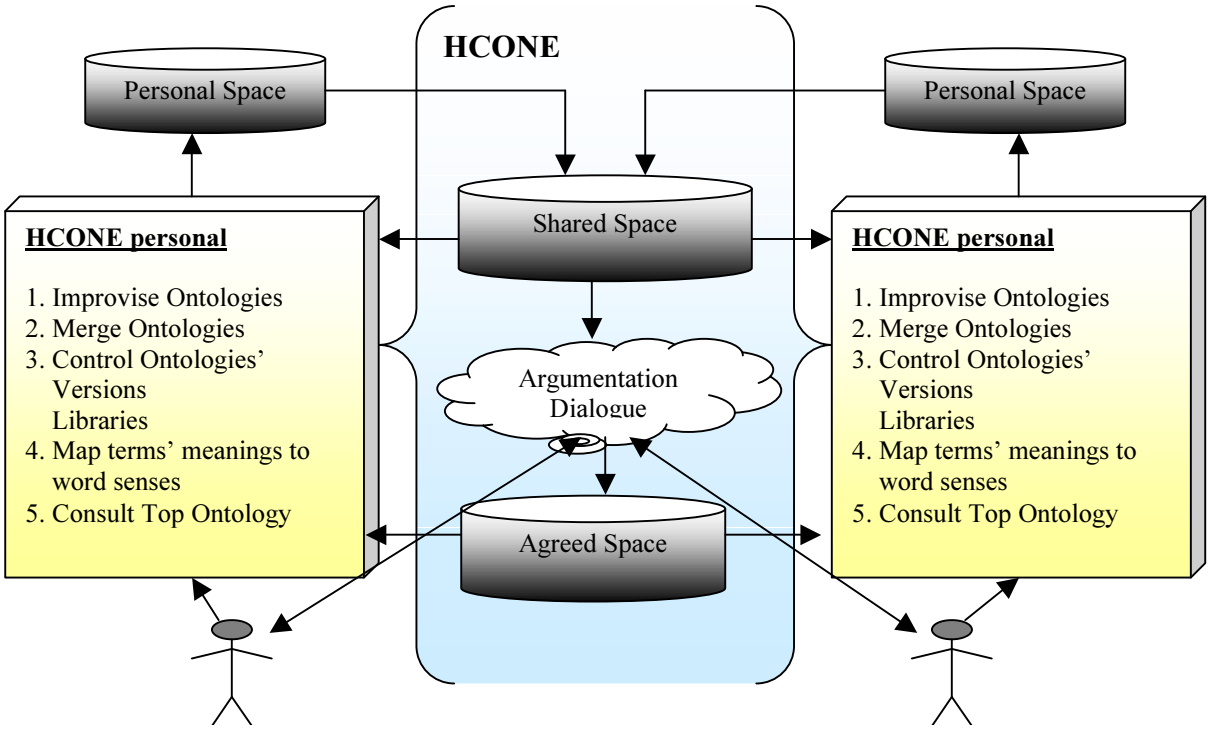


**Figure 1**. HCONE Architecture

encourage feedback among community members to ontological specifications. These facilities must provide the means for detecting new opportunities for collaboration, as well as for getting out of deadlocks within problematic situations that may arise during collaboration.

- *Provide facilities for bridging different perspectives via concepts' meaning disambiguation.* This is important for the development of commonly agreed conceptualizations especially in communities where people from different disciplines may use the same term with different meanings or use different terms for the same concept.
- *Provide facilities for the management of multilingual ontologies.* Multilingualism in this context aims to support ontological specifications in different languages. Doing so, members of organizations may specify an ontology using their native language and may evaluate and exploit an ontology originated in another language, in their native language too.

The development of human-centered ontology management environments is challenging. Not only due to the expectations it creates, but also in that it requires the synergistic deployment of many technologies: *intelligent and collaborative interfaces for editing and deploying knowledge bases, object-centered knowledge representation systems with powerful reasoning services, language-engineering technologies for terms' disambiguation*, to mention a few. HCONE (Human Centered ONtology management Environment) is a prototype environment that has been designed and developed taking into account the above-pointed issues.

## 4. HCONE

HCONE (Human Centered ONtology Environment) follows a decentralized model to ontology engineering that is shown in Figure 1. According to this model people can create their own ontologies stored in a personal space. Ontologies can be later publicized and shared among groups of workers that jointly contribute to ontologies development, with the aim to reach an agreement in conceptualizing their domain. During this process, workers may evolve ontologies by improvising in their personal space, map and synthesize their conceptualizations with the conceptualizations of their co-workers and discuss their arguments, objections and positions within the group. During collaboration, workers follow a structured argumentation process in which they may raise issues, propose solutions via positions, provide arguments for or against a position etc. Agreed ontologies are stored in a virtual space and can be further shared, evolved in workers' personal space and so on.

HCONE is a modular and integrated environment, providing access to any integrated tool in any HCONE point. Doing so, workers are free to combine their own method for using the environment, following an eclectic way to ontology engineering. For instance, a worker may construct an ontology in her personal space while receiving comments on a previous version of the same ontology that has shared with co-workers. In the meantime, she is trying to comply with generic ontological commitments that the group has agreed to comply with, while in another slice of her work she is trying to merge her ontology with an

ontology issued by a co-worker. Towards these targets, HCONE provides facilities for (a) users to improvise their conceptualizations, (b) consult generic ontologies that provide important semantic distinctions, (c) manage different versions of their ontologies, tracking the differences between the versions, (e) track the generalization/ specialization of an ontology during ontology development, (d) get proper consultation from machine exploitable/ readable lexicons by mapping concepts' meaning to word senses, (e) merge ontologies and further manipulate merged conceptualizations, and (f) share their ontologies with groups of co-workers, following a structured conversation towards agreeing in domain conceptualization.

HCONE supports people to interact with their conceptualisations hiding low-level implementation details, enabling them to express subtle ontological distinctions, complying at the same time with formal constraints of specifications. Formal specification of ontologies in HCONE is done in the NeoClassic Description Logic [Patel-Schneider *et al*, 1996].

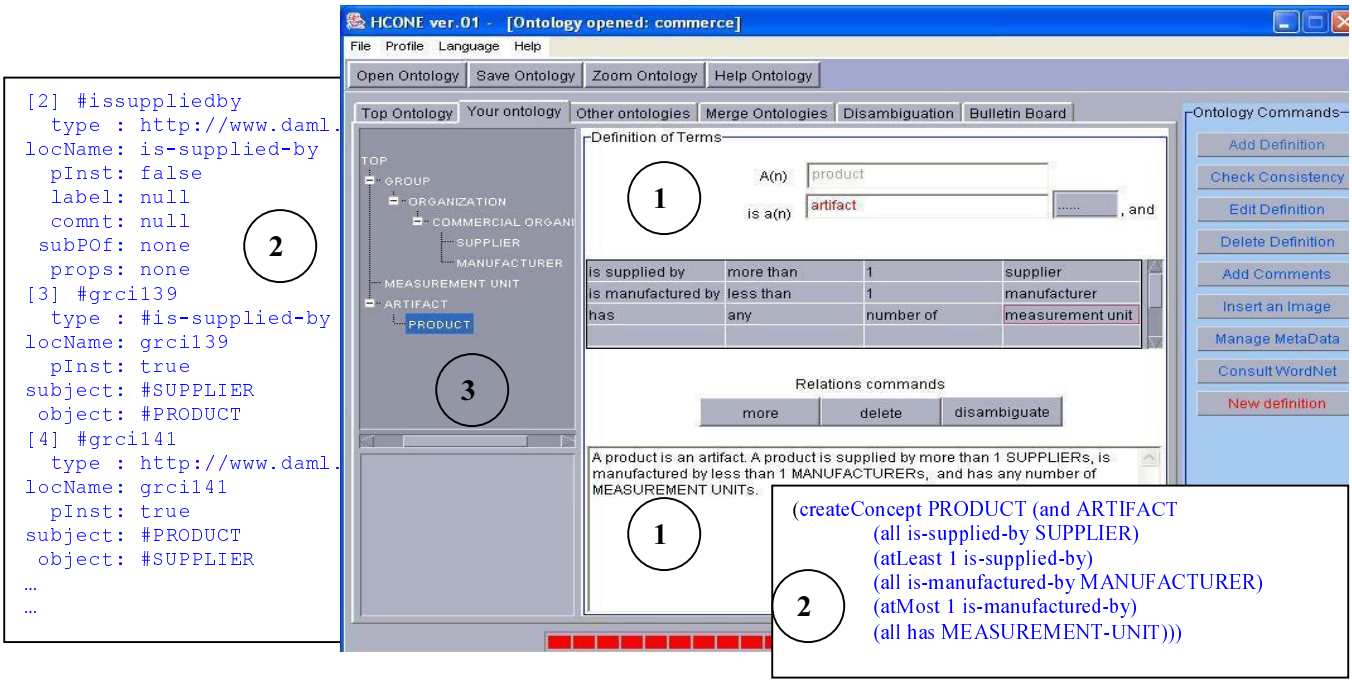


**Figure 2**. HCONE concept specification: 1) natural language, 2) formal, 3) graphical representation

In particular, workers communicate their conceptualizations using the full expressive power of NeoClassic, without dealing with low-level implementation details. This is achieved by following canned natural language dialogues based on a slight variation of the What-You-See-Is-What-You-Meant (WYSIWYM) [Power *et al*, 1997] knowledge specification paradigm. As figure 2 depicts, while users specify the definition of a concept, they get a feedback text that reflects the definition of the corresponding concept. HCONE provides seamless access to reasoning services provided by Description Logics. These services include automatic concepts' classification, concepts' definitions consistency checks (e.g. between a concept and its subsumers) and detection of concepts' definitions differences. Feedback from these reasoning services to users is constantly provided during ontology development/ management and is of high significance. For instance, while tracking the differences between two versions of the same ontology or during merging, reasoning services identify semantically equivalent definitions allowing proper handing of versions, and disallowing semantic errors.

Critical to the ontology specification process is the lexicons consultation process. Through lexicon consultation users are guided to the consensual definition of terms, guided to follow well-established norms and practices in the community they are exercising their practice (e.g. by consulting a terminological lexicon) or in the wider context (e.g. by selecting the appropriate lexicalizations of their conceptions). In HCONE, lexicon consultation can be supported in any of the following three ways: (a) by mapping concepts definitions to word senses in a machine readable/exploitable lexicon through the disambiguation process, (b) by formally complying with generic ontological commitments of top level ontologies or (c) by simply consulting lexicons and other ontologies.

HCONE uses the Latent Semantic Indexing (LSI) technique [Deerwester *et al*, 1990] for mapping terms' meaning to word senses. LSI is a vector space technique for

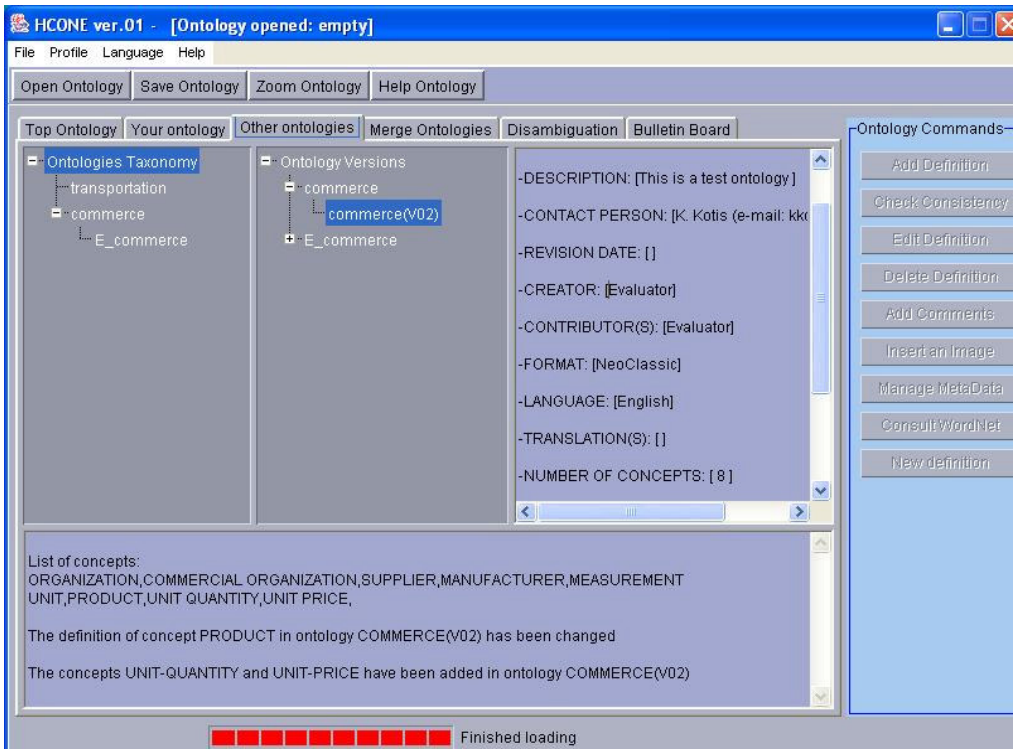


**Figure 3**. HCONE versions management

information retrieval and indexing. It takes a large matrix of term-document association data and constructs a semantic space wherein terms and documents that are closely associated are placed near one another. Currently HCONE exploits the WordNet [Miller *et al*, 1990] lexical database to match descriptions of terms provided by the user with word senses in WordNet. LSI in this case constructs a semantic space of terms and word senses [Agirre and Rigau, 1996]. Word senses that closely match the given terms are provided to the user in an ordered way.

On the other hand, knowledge workers may follow a deductive approach to terms specifications by elaborating a generic top ontology. In this case, concepts definitions can be checked for their semantic validity against generic conceptualisations by means of the consistency checking mechanisms provided by the representation and reasoning system. In doing so, the construction of domain specific ontologies is speed-up and guided by important semantic distinctions specified in generic ontologies. HCONE, in its current status, provides the option of importing any generic ontology, given that it has been implemented in NeoClassic. We have contacted experiments with a generic ontology for terminological resources, which incorporates EuroWordNet and SIMPLE semantic commitments [Vouros and Eumeridou, 2002].

Having mapped terms to word senses, concepts in different ontologies can be associated among themselves, resulting in effective merging of ontologies. Merging functionality in HCONE is performed in "batch mode". The result of this automated process is a merged ontology whose creation is driven by heuristics, a mapping algorithm and the reasoning processes of NeoClassic. As figure 4 shows, HCONE presents the steps of the merging process to the end user as a feedback report, describing the actions the system has followed to the merged ontology. The resulted merged ontology can be further manipulated according to users preferences.

Furthermore, HCONE exploits the association of concepts to WordNet word senses for computing equivalent terms in other European languages.

Finally, a very important built-in HCONE functionality concerns sharing ontologies to group members and supporting group members' participation in structured conversations about conceptualizations. This is a built-in, rather than a patched-on facility, since it has been designed in order to support people to discuss ontological aspects and incorporate their suggestions / positions to specifications, rather than a generic argumentation or discussion facility. This facility enables criticism, identifying possible opportunities for members' collaboration, as well as overcoming deadlocks within problematic situations that arise in ontologies specification.

## 5. Concluding Remarks

HCONE provides a modular environment supporting an eclectic approach to ontologies management, aiming to provide and multiply the opportunities of exploiting ontology management tools in working places. We conjecture that this target can be achieved via designing human-centred ontology management environments.

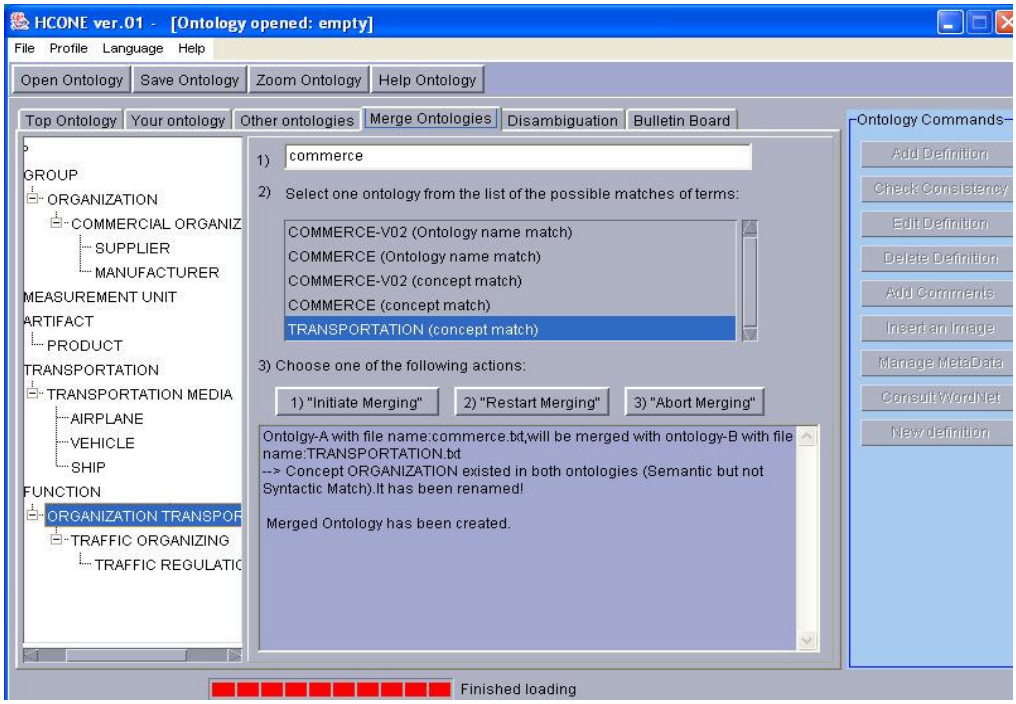


**Figure 4**. HCONE merge process

The fast emergent areas of the semantic web and knowledge management push researchers to new development efforts, but to our view no specific principles have been raised for evaluating the impact of tools on making the foreseen targets achievable. In the latest EON-Workshop[2], assessments of the most well known ontology engineering environment have resulted to some very important conclusions:

a) ontology engineering environments' development criteria cannot be considered in isolation from situations in which the ontology-based tools will be used. Criteria need to be connected to scenarios of use. Such scenarios need to be explained, further analyzed and be connected to activity models. Furthermore, there must be a balance between usage and technology, and between formality and informality [Giboin *et al*, 2002]. HCONE, following the human centered design approach, emphasizes on integrating ontology engineering environments with knowledge providers/consumers' practices considerably, enabling knowledge workers to interact with their conceptualizations at a high-level.

b) There are two types of ontology engineering environments: (1) Environments for developing ontologies and (2) environments for mapping, aligning, or merging ontology [Noy and Musen, 2000]. Mapping environments may identify potential correspondences between concepts, provide the user with guidelines for defining these correspondences, or both. HCONE follows an eclectic approach to ontology management, integrating techniques for ontology management in a toolset.

c) The development of ontologies must be seen as a dynamic process that in most of the cases starts with an initial rough ontology, which is later revised, refined and filled in with the details. Ontology evolution has to be supported through the entire lifecycle, resulting to a living ontology [Stojanovic and Motik, 2002]. HCONE incorporates functionalities (including development, versioning, argumentation, merging) for supporting the management of continuously evolving ontologies.

## References

[Agirre and Rigau, 1996] E. Agirre and G. Rigau. Word sense disambiguation using conceptual density. In Proceedings of COLING-96, 1996.

[Arpirez *et al*, 2001] J.C. Arpirez, O. Corcho, M. Fernandez Lopez, and A. Gomez-Perez. WebODE: a scalable workbench for ontological engineering. In Proceedings of the First International Conference on Knowledge Capture (K-CAP) Oct. 21-23, 2001, Victoria, B.C., Canada, 2001.

[Bechhofer *et al*, 2001] S. Bechhofer, I. Horrocks, C. Goble, and R. Stevens. OilEd: A reason-able ontology editor for the semantic web. In KI-2001: Advances in Artificial Intelligence, LNAI 2174, pages 396-408. Springer, 2001.

[2] http://km.aifb.uni-karlsruhe.de/eon2002

[Benjamins *et al*, 1998] Benjamins R., Fensel D., & Perez A. G. Knowledge management through ontologies. In Proceedings of the 2nd International Conference on Practical Aspects of Knowledge Management (PAKM 98), Basel, Switzerland. October, 1998.

[Chandrasekaran *et al*, 1999] Chandrasekaran, B., Josephson, J. R., & Benjamins, V. R. (1999, January/February). What are ontologies, and why do we need them?" *IEEE Intelligent Systems*, pp. 20-26.

[Clancey, 1999] Clancey, W. J. An Introduction to Human-Centered Computing. Report, NASA/Ames Research Center, Computational Sciences Division, 1999.

[Cook and Brown, 1999] Cook S. D. N., & Brown, J. S. Bridging epistemologies: The generative dance between organizational knowledge and organizational knowing. Organizational Science, 10:381-400, 1999.

[Deerwester *et al,* 1990] Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, Richard Harshman. Indexing by Latent Semantic Analysis. Journal of the American Society of Information Science, 1990.

[Erdman *et al*., 2001] Erdman M., A.Maedche, H.P.Schnurr, S.Staab. From Manual to Semi-automatic Semantic Annotation: About Ontology-based Text Annotation Tools. ETAI Journal – Section on Semantic Web, Linkoping Electronic Articles in Computer Science Vol. 6. 2001.

[Fensel *et al*, 2000] Fensel D., Harmelen F., Klein M., Akkermans H. On-To-Knowledge: Ontology-based Tools for Knowledge Management.

[Giboin *et al*, 2002] Alain Giboin, Fabien Gandon, Olivier Corby, and Rose Dieng. Assessment of Ontology-based Tools: Systemizing the Scenario Approach. In Proceedings of OntoWeb-SIG3 Workshop at the 13th International Conference on Knowledge Engineering and Knowledge Management EKAW 2002, pages 63-73. Siguenza, Spain, 30th September 2002.

[Hoffman *et al*, 2002] Hoffman, R., Hays, P., and Ford, K. M. The Triples Rule. IEEE Intelligent Systems, pages 62-65, May/June, 2002.

[Maedche *et al*., 2000] Maedche, A., S.Staab, Discovering conceptual relations from text. In Proceedings of ECAI-2000 IOS Press, Amsterdam.

[Miller *et al*, 1995] Miller G., Beckwith R., Fellbaum D., Gross D., Miller K. Five papers on WordNet. CSL Report 43, Cognitive Science Laboratory, Princeton University, 1995.

[Noy and Musen, 2000] Natalya Fridman Noy and Mark A. Musen. Evaluating Ontology-Mapping Tools: Requirements and Experience. In Proceedings of OntoWeb-SIG3 Workshop at the 13th International Conference on Knowledge Engineering and Knowledge Management EKAW 2002, pages 1-14. Siguenza, Spain, 30th September 2002.

[ONTOWEB, 2002] ONTOWEB Consortium for Ontology-based information exchange for knowledge management and electronic commerce. IST-2000-29243, Work package 1, Deliverable 1.3: A survey on Ontology Tools. May, 2002.

[Patel-Schneider *et al*, 1996] Peter F. Patel-Schneider, Merryll Abrahams, Lori Alperin Resnick, Deborah L. McGuinness and Alex Borgida. NeoClassic Reference Manual: Version 1.0. Artificial Intelligence Principles Research Department, AT&T Bell Labs, 1996.

[Power *et al*, 1997] Power R., Scott D., Evans R. What You See Is What You Meant: direct knowledge editing with natural language feedback. ITRI Technical Report No. 97-03, University of Brighton, 1997.

[Staab *et al*, 2001] Staab S, Studer R., Schnurr H., Sure Y. Knowledge Processes and Ontologies. IEEE Intelligent Systems, pages 26-34, January/February, 2001.

[Stojanovic and Motik, 2002] Ljiliana Stojanovic and Boris Motik. Ontology Evolution within Ontology Editors. In Proceedings of OntoWeb-SIG3 Workshop at the 13th International Conference on Knowledge Engineering and Knowledge Management EKAW 2002, pages 53-62. Siguenza, Spain, 30th September 2002.

[Tennison *et al*, 2002] Tennison, J., O'Hara, K., & Shadbolt, N. APECKS: Using and evaluating a tool for ontology construction with internal and external KA support. International Journal of Human-Computer Studies, 56:375-422, 2002.

[Vouros, 2003] George A. Vouros. Technological Issues towards Knowledge Powered Organizations", Knowledge Management Journal, Vol 7, No. 1, 2003

[Vouros and Eumeridou, 2002] George A. Vouros, Eugenia Eumeridou. "Simple and EuroWordNet: Towards the Prometheus Ontological Framework", "Terminology" 8:2 (2002) pp 245-278.