INFORMATICA, 2000, Vol. 11, No. 1, 41–48

Ontological Engineering: Common Approaches and Visualisation Capabilities

Saulius MASKELIŪNAS
Institute of Mathematics and Informatics
2600 Vilnius, Akademijos 4, Lithuania
e-mail: mask@ktl.mii.lt

Abstract. The survey of the current status in ontological engineering is presented: notion, peculiarities, applications, design and evaluation of ontologies. The possibilities of using *The Brain*^{TM}, a personal desktop productivity tool, for visualisation of ontologies are outlined and compared with that of Hyperbolic ontology viewer of Ontobroker.

Key words: knowledge-based systems, knowledge sharing, ontologies, visualisation.

1. Ontological Engineering

1.1. *Notion of Ontologies*

In its *general meaning*, ontology is the theory or study of being as such; i.e., of the basic characteristics of all reality. Though the term was first introduced in the $17th$ century, ontology is synonymous with metaphysics or "first philosophy" as defined by Aristotle in the $4th$ century BC.

Since the beginning of the nineties the usage of the term "ontologies" has become more and more frequent in *artificial intelligence community*[1], i.e., in knowledge sharing (interchange), agent interoperation, common sense knowledge representation, natural language processing, and other fields.

The essence of ontological engineering can be explained by the idea of *knowledge representation levels*. According to Brachman (1979), knowledge can be represented in the following levels:

- *Linguistic, language* (linguistic terms),
- *Conceptual* (conceptual relations, primitive objects and actions),
- *Epistemological, structure* (concept types, structuring relations),
- *Logical, formalisation* (propositions, predicates, functions, logical operations), and
- *Implementation* (memory cells, pointers).

Guarino (1994) has supplemented this view with:

[1] `http://www.cs.utexas.edu/users/mfkb/related.html`

- *Ontological, meaning level*, inserted between conceptual and epistemological levels.

In practical usage, computer-oriented ontological engineering usually covers three levels: conceptual, ontological, and epistemological.

Now there are many *disciplines involved* in and supported by ontological engineering: information (library) science, (computational) linguistics, corporate knowledge management, database technology (conceptual schemas), professional terminological standardisation, knowledge engineering (Breuker *et al.*, 1997), each of them having specific attitude to ontologies. But the most cited and widely accepted *definition* of ontologies is given by Gruber (1993): "An ontology is an explicit specification of a conceptualization". "Conceptualization" here refers to an abstract model of a phenomenon in the world by having identified the relevant concepts of it; "explicit" means that the concepts used, concept types, and the constraints on their usage are explicitly defined. An ideal ontology is easily integrated, shared, collaboratively developed and maintained (Gangemi *et al.*, 1999).

The most commonly used *types of ontologies* are: knowledge representation ontologies, general/common sense ontologies, top-level ontologies, meta-ontologies (also called generic or core ontologies), domain ontologies, linguistic ontologies, task, method and application ontologies (Gómez-Pérez and Benjamins, 1999).

In addition, according to the development level, ontologies can be classified into: (1) informal ontological repositories (catalogues of normalised terms, glossed catalogues, taxonomies), (2) axiomatised taxonomies, and (3) ontology libraries (Gangemi *et al.*, 1999). There is a growing number of publicly available ontology libraries on the Internet, e.g.: The Ontology Server[2], Cycorp's Upper CYC Ontology Server[3], Ontosaurus[4] (Arpírez Vega *et al.*, 1998).

1.2. *Main Features and Applications of Ontologies*

The basic modelling primitives of ontologies are: classes (concepts), relations, functions, distinguished objects (instances), and axioms (Gruber, 1993). The key peculiarities of ontologies are:

- a hierarchical, taxonomic organisation of concepts from general to specialised;
- the list of key attributes associated with each concept, along with restrictions on the type of these attributes, and default values where appropriate; and
- allowed relations between concepts to link the ontology using non-taxonomic connections (Altman *et al.*, 1999).

Desirable features of ontologies are: explicit taxonomy with subsumption among concepts, semantic explicitness, modularity of namespace, stratified design of the modules, absence of polysemy within a module, interface between ontology namespace and set(s)

[2] `http://www-ksl.stanford.edu/knowledge-sharing/ontologies/index.html` and `http://www-ksl-svc.stanford.edu:5915/`
[3] `http://www.cyc.com`
[4] `http://www.isi.edu/isd/ontosaurus.html`

of lexical realisations, linguistically meaningful naming policy (cognitive transparency), rich documentation, minimal axiomatisation to detail the difference among sibling concepts, explicit linkage to concepts and relations from generic theories, meta-level assignments to distinguish among the formal primitives assigned to concepts, supporting languages and implementations, possibility of collaborative modelling (Gangemi *et al.*, 1999). All characteristics of ontologies can be classified into: identifying features (on ontology, main developers, and distributors), descriptive features (general, scope, design, requirements, cost, usage), and functional features (Arpírez Vega *et al.*, 1998).

The most commonly used *languages to build ontologies are* Ontolingua, CycL, Loom and FLogic (Gómez-Pérez and Benjamins, 1999). The formality of ontology representation languages can vary from highly informal, structured-informal to semi-formal and rigorously formal (Uschold and Jasper, 1999). A greater formality of ontologies restricts possible interpretations, reduces ambiguity.

Ontology *applications* can be described according to the following dimensions (Uschold and Jasper, 1999):

1) intended purpose and benefits (communication, inter-operability, software engineering benefits),
2) role of ontology,
3) actors necessary to implement ontology application scenarios,
4) supporting technologies,
5) maturity level,
6) formality of term meaning representation,
7) architecture for sharing vs. exchange of information.

The main categories of ontology applications are: (1) neutral authoring (authoring ontologies, authoring operational data), (2) common access to information (human communication, data access via shared or mapped ontologies, shared services), (3) indexing for concept based search (Uschold and Jasper, 1999).

Most popular application areas of ontologies are: knowledge management, natural language generation, enterprise modelling, knowledge-based systems, ontology-based brokers, interoperability between systems (Gómez-Pérez and Benjamins, 1999).

1.3. *Design of Ontologies*

Design criteria for ontologies are: (1) clarity and objectivity, (2) coherence, (3) maximum monotonic extendibility, (4) minimal encoding bias, (5) minimal ontological commitment (Gruber, 1995). Gómez-Pérez and Benjamins (1999) have supplemented this list with a set of principles that had been proved useful in the development of ontologies: ontological distinction principle (i.e., classes in an ontology should be disjoint), diversification (multiple inheritance) of hierarchies, modularity, minimisation of the semantic distance between sibling concepts, standardisation of names.

The best known *methodologies* for building ontologies are: Uschold's methodology (based on the experience building the Enterprise Ontology), Grüninger and Fox's

(TOVE) methodology, the METHONTOLOGY framework, KACTUS methodology, and SENSUS-based methodology. Their comparative analysis is presented in (Fernández López, 1999). Here, the criteria for analysing methodologies are: inheritance from knowledge engineering, the level of details, knowledge formalisation, ontology building strategy (application dependent, semi-dependent or independent), strategy for identifying concepts (bottom-up, top-down or middle-out), recommended life cycle, recommended techniques, usage.

The most popular *ontological engineering tools* – Ontolingua, WebOnto, ProtégéWin, OntoSaurus, ODE, KADS22 – are compared and contrasted in general, ontology and co-operation aspects in (Duineveld *et al.*, 1999).

Four main *steps in developing ontologies* are: requirement specification, conceptualisation, implementation, and evaluation (Gómez-Pérez *et al.*, 1996).

Two ontologies that represent the same problem area can be: alternative, truly overlapping, equivalent but with vocabulary mismatches, overlapping and with disjoint domains, homonymically overlapping; and can have various *levels of interoperability*: mediation (weak interoperability), alignment (with at least a partial conceptual integration), unification (with a principled conceptual integration) (Gangemi *et al.*, 1999).

There are three alternative ways for *integration of ontologies*: (1) building a new ontology *reusing (composing)* other available ontologies, (2) *merging* different ontologies on the same subject into a single one that "unifies" all of them, (3) *integration* of ontologies *into applications*. The characterisation of each of them is presented in (Pinto *et al.*, 1999).

1.4. *Evaluation of Ontologies*

The main terms for evaluation of knowledge sharing technologies (in general) and ontologies (in particular) are: *evaluation* (i.e., judge against a reference framework), *verification* (i.e., guaranteeing of correctness with respect to a reference framework), *validation* (i.e., checking of correspondence to the systems that they are supposed to represent), and assessment (i.e., the usability and usefulness when they are reused or shared in applications) (Gómez-Pérez, 1999). The evaluation of ontologies refers to the correct building of the ontology content, and includes evaluation of: (1) each individual definition and axiom, (2) collection of definitions and axioms that are stated explicitly in the ontology, (3) definitions that are imported from other ontologies, and (4) definitions that can be inferred using other definitions and axioms (Gómez-Pérez, 1999).

Criteria for ontology comparison are: *Epistemological adequacy* (clarity, intuitiveness, relevance, completeness, discriminative power), *Operationality* (encoding bias, coherence, computationality), and *Reusability* (task-and-method reusability, domain reusability) (Visser and Bench-Capon, 1997).

Intended *benefits* of using ontologies are: *interoperability* (i.e., mapping between different concepts in different components), *browsing and searching* (i.e., assisting an intelligent search engine in query processing, e.g., automatical generalisation of the query to find nearest partial matches), *reuse* (e.g., using a Component X which exists in someones else's library, instead of rebuilding it), and *structuring* (i.e., using the conceptualisations to assist in structuring the knowledge of a new domain) (Menzies, 1999).

On the other hand, the circularity, partition, redundancy, grammatical, semantic, and incompleteness *errors* are rather frequent in developing of ontologies (Gómez-Pérez, 1999). The common *problems of* existing *terminologies* requiring a solution are: Lack of hierarchies, Ambiguous hierarchies, Informality, Lack of modularity, Polysemy, Uncertain semantics, Prototypical descriptions, Ontological opaqueness, Lack of a (minimal) set of axioms, Confusing lexical clues, Awkward naming policy, 'Remainder' and 'Exception' partitions, Terminological cycles, Meta-level soup (i.e., no distinction among different kinds of concepts), and Low maintenance capabilities (Gangemi *et al.*, 1999).

2. Visualisation of Ontologies; Using *The Brain*^{TM} as Ontology Viewer

Ontologies usually are large interconnected hierarchies of concepts. In more complicated cases, it is important for the users to have a possibility of overview of the whole hierarchy, quick and easy navigation from one class in the hierarchy to another. However, the visualisation of ontologies is not a widely discussed problem yet. The best known approach to ontology visualisation is the Hyperbolic ontology viewer of Ontobroker[5] (Fensel *et al.*, 1998), which is a graphic hyperbolic interface based on a Java-profiler[6].

On the other hand, various personal information management tools make it posible to visualise hierarchical and networked information, too. One of most powerful tools of this kind is *The Brain*^{TM} of Natrificial Software Technologies (Natrificial LLC, 1998), that runs on Windows 95, 98, or NT 4.0. Sample applications of it include project management, decision support, knowledge management, collaboration, information portals, business intelligence, and presentations.

The Brain is a personal desktop productivity tool which lets users create navigable associative maps of personal ideas, file systems, Websites, etc. for fast and powerful surfing. I.e., *The Brain* is a graphic user interface that allows to create associations between items of many types and navigate through them in a way that mimics human thought. It is an alternative to the ordinary knowledge tree hierarchy. Instead of folders inside endless folders, *The Brain* lets users put files, documents, and Web pages (each one is considered a Thought) into collections (called Brains), and create almost infinite associations between different Thoughts. In addition, *The Brain* has an ability to add notes (such as a page summary or product description) to any Thought, a History list, a Properties list that details any activated Thought, and the ability to search by keyword for Thoughts in any recently opened Brain. Switching between Brains is very easy.

In order to test visualisation capabilities of *The Brain* and to compare it with the hyperbolic ontology viewer of Ontobroker, the *Research-topic ontology*[7] was selected. This ontology is used at $(KA)^2$ project[8], and describes research groups, topics, products, etc.

[5] `http://www.aifb.uni-karlsruhe.de/WBS/broker`
[6] `http://www.physics.orst.edu/bulatov/HyperProf/index.html`
[7] `http://www.swi.psy.uva.nl/usr/richard/ka2/research-topic.html`
[8] `http://www.aifb.uni-karlsruhe.de/WBS/broker/KA2.html`

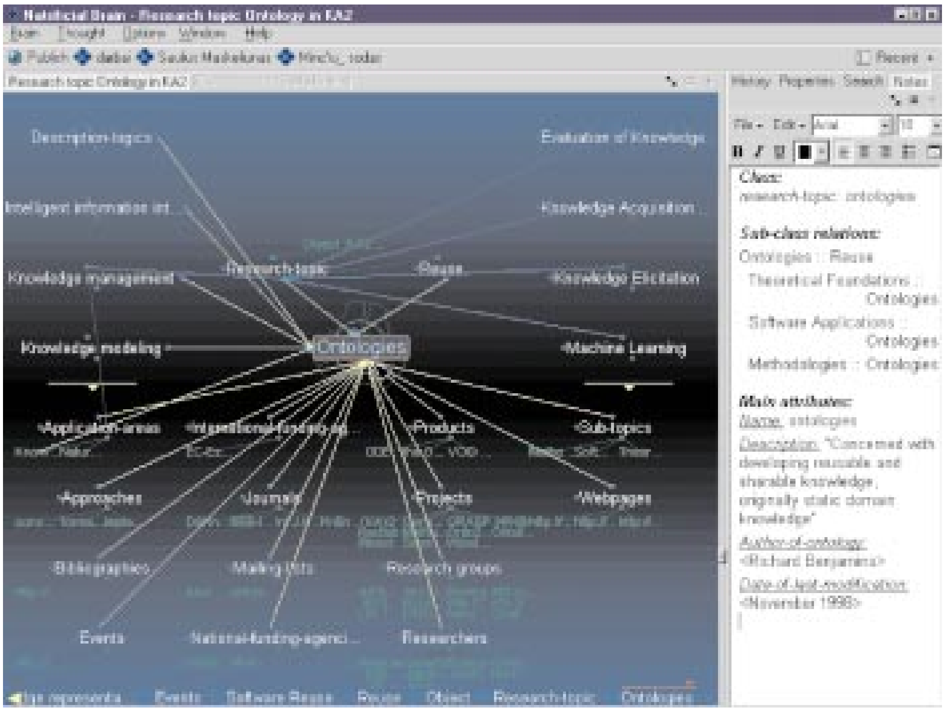

Fig. 1. The view of instance "Ontologies" from the "Research-topic ontology of $(KA)^2$" presented by *The Brain*.

of the knowledge acquisition community. Here, "$(KA)^2$" means "Knowledge Annotation Initiative of the Knowledge Acquisition Community" (Benjamins *et al.*, 1998).

An example of the ontology view by means of *The Brain* is given in Fig. 1. Here the view of the "Ontology" item (which is a component of "Research-topic" item) with the links to "parent", "child", "jump" items and the corresponding notes is presented.

Navigation in *The Brain* space is very similar to that of Hyperbolic ontology viewer of Ontobroker. However, the visualisation power of *The Brain* viewer is much stronger, more convenient for the user:

- links of each Brain thought (i.e., represented ontology item) are clearly grouped into parent (possibly multiple), sibling, child, and jump thought sets. Parent thoughts are presented above the active thought, sibling thoughts – on the right side, child thoughts – below, and jump thoughts – on the left side of active thought;
- there are many options what to represent in the graphical view;
- not only web pages, but also shortcuts to files of any type and needed graphic icons can be attached to each thought;
- editing of ontology representations by proper mouse clicks and draggings is natural and convenient;
- there is an easy possibility to publish the created ontologies on the Internet, make thoughts (i.e., items of a created ontology) private or public.

3. Conclusions

The field of ontological engineering is as if a new generation (generalisation) of database conceptual modelling: adapting it to new possibilities of using Internet – Intranets – Extranets (i.e., networked collaboration), world-wide data and knowledge sharing, software agent interoperability, etc. Besides, the revolutionary influence of XML standards[9] on modern information technologies will manifest itself only in combination with well developed ontologies in the near future.

Current ontological engineering tools lack more powerful visualisation capabilities. *The Brain* of Natrificial Software Technologies is rather a prospective tool for this: it surpasses the famous Hyperbolic ontology viewer of Ontobroker.

References

Altman, R.B., M. Bada, X.J. Chai, M. Whirl Carillo, R.O. Chen, N.F. Abernethy (1999). RiboWeb: an ontology-based system for collaborative molecular biology. *IEEE Intelligent Systems*, **14**(5), 68–76.

Arpírez Vega, J.C., A. Gómez-Pérez, A. Lozano Tello, H.S.A.N.P. Pinto (1998). (ONTO)^{2}Agent: An ontology-based www broker to select ontologies. In A. Gómez-Pérez and V.R. Benjamins (Eds.). *Proc. of the Workshop on Applications of Ontologies and Problem solving Methods, held in conjunction with ECAI-98*, Brighton, England, pp. 16–24.
`http://delicias.dia.fi.upm.es/WORKSHOP/ECAI98/accepted-papers.html`

Benjamins, V.R., D. Fensel, A. Gómez-Pérez, S. Decker, M. Erdmann, E. Motta, M. Musen (1998). Knowledge annotation initiative of the knowledge acquisition community (KA)^2. In *Proc. of the 11th Workshop on Knowledge Acquisition, Modeling and Management, KAW'98*, Voyager Inn, Banff, Alberta, Canada.
`http://ksi.cpsc.ucalgary.ca/KAW/KAW98/KAW98Proc.html`

Benjamins, V.R., B. Chandrasekaran, A. Gómez-Pérez, N. Guarino, M. Uschold (Eds.) (1999). *Proc. of the IJCAI-99 workshop on Ontologies and Problem-Solving Methods (KRR5)*, Stockholm, Sweden.
`http://SunSITE.Informatik.RWTH-Aachen.DE/Publications/CEUR-WS/Vol-18/`

Brachman, R.J. (1979). On the epistemological status of semantic networks. In N.V. Findler (Ed.), *Associative Networks: Representation and Use of Knowledge by Computers*. Academic Press.

Breuker, J., A. Valente, R. Winkels (1997). Legal ontologies: a functional view. In *Visser and Winkels*, pp. 23–36.

Duineveld, A.J., R. Stoter, M.R. Weiden, B. Kenepa, V.R. Benjamins (1999). WonderTools? A comparative study of ontological engineering tools. In *Proc. of the 12th Workshop on Knowledge Acquisition, Modeling and Management, KAW'99*. Voyager Inn, Banff, Alberta, Canada.
`http://sern.ucalgary.ca/KSI/KAW/KAW99/papers.html`

Fensel, D., S. Decker, M. Erdmann, R. Studer (1998). "Ontobroker: or how to enable intelligent access to the WWW" and "How to use Ontobroker". In *Proc. of the 11th Workshop on Knowledge Acquisition, Modeling and Management, KAW'98*, Voyager Inn, Banff, Alberta, Canada.
`http://ksi.cpsc.ucalgary.ca/KAW/KAW98/KAW98Proc.html`

Fernández López, M. (1999). Overview of methodologies for building ontologies. In *Benjamins et al.*, pp. 4-1 – 4-13.

Gangemi, A., D.M. Pisanelli, G. Steve (1999). An overview of the ONIONS project: Applying ontologies to integration of medical terminologies. *Data & Knowledge Engineering*, **31**(2), 183–220.

Gómez-Pérez, A. (1999). Evaluation of taxonomic knowledge in ontologies and knowledge bases. In *Proc. of the 12th Workshop on Knowledge Acquisition, Modeling and Management, KAW'99*. Voyager Inn, Banff, Alberta, Canada. `http://sern.ucalgary.ca/KSI/KAW/KAW99/papers.html`

Gómez-Pérez, A., and V.R. Benjamins (1999). Overview of knowledge sharing and reuse components: ontologies and problem-solving methods. In Benjamins *et al.*, pp. 1-1 - 1–15.

[9] `http://www.w3.org/XML, http://www.xml.org`

Gómez-Pérez, A., M. Fernández, A.J. de Vincente (1996). Towards a method to conceptualize domain ontologies. In *Proc. of the Workshop on Ontological Engineering*, held in conjunction with ECAI96, Budapest, Hungary. http://wwwis.cs.utwente.nl:8080/kbs/EcaiWorkshop/fullpapers.html
Gruber, T.R. (1993). A translation approach to portable ontology specifications. *Knowledge Acquisition*, **5**(2), 199–220.
http://www-ksl.stanford.edu/knowledge-sharing/README.html#ontolingua-intro
Gruber, T.R. (1995). Toward principles for the design of ontologies used for knowledge sharing. In N. Guarino and R. Poli (Eds.), *Formal Ontology in the Information Technology*, Special issue of the *International Journal of Human-Computer Studies*, **43**(5/6), pp. 907–928.
http://www.ladseb.pd.cnr.it/infor/Ontology/IJHCS/IJHCS.html
Guarino, N. (1994). The ontological level. In R. Casati, B. Smith, G. White (Eds.), *Philosophy and the Cognitive Sciences*, Hölder-Pichler-Tempsky, Vienna.
http://www.ladseb.pd.cnr.it/infor/ontology/Papers/OntologyPapers.html
Menzies, T. (1999). Cost benefits of ontologies. *Intelligence*, **10**(4), 26–32.
Natrificial LLC (1998). *The Brain^{TM}, User Manual*. Natrificial Software Technologies.
http://www.thebrain.com (or:
http://store.thebrain.com/mirror/Brain_Manual.pdf)
Pinto, H.S., A. Gómez-Pérez, J.P. Martins (1999). Some issues of ontology integration. In Benjamins *et al.*, pp. 7-1 – 7-12.
Uschold, M., and R. Jasper (1999). A framework for understanding and classifying ontology applications. In Benjamins *et al.*, pp. 11-1 – 11-12.
Visser, P.R.S., and T.J.M. Bench-Capon (1997). A comparison of two legal ontologies. In *Visser and Winkels*, pp. 37–45.
Visser, P.R.S., and R.G.F. Winkels (Eds.) (1997). *Proceedings of the First International Workshop on Legal Ontologies, LEGONT'97*. University of Melbourne, Law School, Melbourne, Victoria, Australia.
http://www.csc.liv.ac.uk/pepijn/legont-prg.html

S. Maskeliūnas received the Doctors degree in Informatics from the Institute of Mathematics and Informatics (IMI) in 1996. Now he is a researcher at the Software Systems Engineering Department, IMI. His interests include information and knowledge-based systems, qualitative simulation, wokflow automation, knowledge management.

Ontologijų inžinerija: dažniausiai naudojami metodai ir vizualizavimo galimybės

Saulius MASKELIŪNAS

Straipsnyje pateikiama ontologijų inžinerijos dabartinės būsenos apžvalga: ontologijų samprata, pagrindinės savybės ir taikymai, konstravimas, įvertinimas. Nagrinėjama galimybė panaudoti asmeninės informacijos produktyvaus tvarkymo priemonę *The Brain^{TM}* ontologijų vizualizavimui, lyginant su garsia Ontobroker sistemos ontologijų hiperbolinės peržiūros priemone.