# Technical Roadmap v1.0

Oscar Corcho, Mariano Fernández-López, Asunción Gómez Pérez
Universidad Politécnica de Madrid
{ocorcho, mfernandez, asun}@fi.upm.es

| | |
|---|---|
| Identifier | Deliverable 1.1 |
| **Class** | **Deliverable** |
| **Version** | **1.0** |
| **Version date** | **30/11/01** |
| **Status** | **Final** |
| **Distribution** | **Public** |
| **Responsible Partner** | **UPM** |

# OntoWeb Consortium

This document is part of a research project funded by the IST Programme of the Commission of the European Communities as project number IST-2000-29243.

**Vrije Universiteit Amsterdam (VU)-Coordinator**
Faculty of Sciences,
Division of Mathematics and Computer Science
De Boelelaan 1081a
1081 HV Amsterdam, the Netherlands
Fax and Answering machine: +31-(0)84-872 27 22
Mobil phone: +31-(0)6-51850619
Contactperson: Dieter Fensel
E-mail: dieter@cs.vu.nl

**Additional contributors:**

Natahalie Aussenac-Gilles, Socorro Bernardos, Olivier Corby, Peter Crowther, Rob Engels, Miguel Esteban, Fabien Gandon, Yannis Kalfoglou, Manuel Lama, Angel López, Adolfo Lozano, Enrico Motta, Natasha Noy, York Sure

# Revision Information

| Revision date | Version | Changes |
|---|---|---|
| 26-11-2001 | 0.1 | First draft |
| 28-11-2001 | 0.2 | New information added |
| 29-11-2001 | 0.3 | New style sheet |
| 30-11-01 | 0.4 | Additions and suggestions from Enrico Motta |
| 30-11-01 | 1.0 | Final version |

# Contents

# Executive summary

This deliverable presents a first version of the technical roadmap of the ontology field in Europe and worldwide, which includes the information that has been obtained using the OntoRoadMap application (http://babage.dia.fi.upm.es/ontoweb/wp1/OntoRoadMap/index.html), with the contribution of many research groups, and additional information that has been obtained and organized by the authors. The Roadmap is a living deliverable throughout the Ontoweb project, open to the inclusion of important contributions on the identified topics and also of new relevant topics.

The deliverable is organized as follows:

The first section presents the theoretical foundations of the ontology field, which includes the vocabulary standardisation of the area (ontology, ontological commitments, types of ontologies, components of ontologies, ontology design guidelines, etc.). It also presents the most well-known ontologies, organizing them in top level, linguistic, general and domain ontologies in several areas.

In the second section, we present an exhaustive list of methodologies for building ontologies (either cooperative or not), for reengineering, for merging and integrating, for ontology learning and for ontology evaluation.

The third section presents an overview of the most relevant ontology tools that have been used for the building, merging and integration of ontologies, ontology learning tools and tools for the annotation of web resources in the context of the Semantic Web.

Section 4 deals with ontology specification languages. In this section, we present several widely used languages in the ontology field, grouped in traditional ontology languages, web-based languages and other languages from specific fields of expertise or that have been used for developing specific ontologies.

In section 5, an overview of ontology-based applications is presented, and it is organized in several groups of scenarios, related to Knowledge Management, Natural Language Processing, e-commerce, intelligent integration of information, information retrieval and education. This list of applications is not exhaustive, but tries to show a broad overview of existing ontology-based applications in different areas of knowledge.

Finally, section 6 presents a list of important events that are related to the ontological engineering field.

Inside each section, contents will be organized alphabetically.

# 1 Foundations of ontologies

## *1.1 Introduction*

Already at the middle of 80s, a big knowledge base on common sense began to be built [Lenat et al.; 90]. This knowledge base can be consider an ontology. However, it is not until the beginning of 90s when ontologies are more known. It is in that time when DARPA starts its Knowledge Sharing Effort envisioning a new way in which intelligent systems could be built. They proposed [Neches et al.; 91].: ``Building knowledge-based systems today usually entails constructing new knowledge bases (KB's) from scratch. It could be done by assembling reusable components. Systems developers would then only need to worry about creating the specialized knowledge and reasoners new to the specific task of their system. This new system would interoperate with existing systems, using them to perform some of its reasoning. In this way, declarative knowledge, problem-solving techniques and reasoning services would all be shared among systems. This approach would facilitate building bigger and better systems cheaply..." Since then, considerable progress has been made in developing the conceptual bases needed for building technology that allows knowledge-component reuse and sharing.

In this section, the fundamental concepts about ontologies will be presented. The reader will be able to answer questions as: what is an ontology? What are the main components of an ontology? What are the most well known ontologies? Where can I find them? What design criteria can I follow when I build ontologies?

Most of the following sections are short descriptions so that the reader can obtain a general overview of each important concept. Besides, we provide bibliographic references and URL links to easy the user can find more information if needed.

**References:**


- Lenat, D.B.; Guha, R.V. "Building large knowledge-based systems". *Addison-Wesley Publising Company, Inc*. 1990.
- Neches, R.; Fikes, R.E.; Finin, T.; Gruber, T.R.; Senator, T.; Swartout, W.R. "Enabling technology for knowledge sharing". *AI Magazine*. 12(3)::36-56- 1991.


## *1.2 What is an ontology?*

The word "ontology" came from philosophy. From a philosophical viewpoint, "Ontology" (without the indeterminate article and with the uppercase initial) is the branch of philosophy which deals with the nature and the organisation of reality [Guarino et al.; 95].

At the computer science domain, ontologies aim at capturing domain knowledge in a generic way and provide a commonly agreed understanding of a domain, which may be reused and shared across applications and groups [Chandra-richard-99]. Ontologies provide a common vocabulary of an area and define -with different levels of formality- the meaning of the terms and the relations between them.

One of the first definitions on the computer science domain is due to Neches and colleagues [Neches et al., 91]. They defined an ontology as follows ``An ontology defines the basic terms and relations comprising the vocabulary of a topic area as well as the rules for combining terms and relations to define extensions to the vocabulary". We can say that this definition tells us how to proceed to build an ontology, giving us vague guidelines: identify basic terms and relations between terms, identify rules to combine them, provide definitions of such terms and relations. Note that according to this definition, an ontology includes not only the terms that are explicitly defined in it, but also terms that can be inferred using rules. Later, in 1993, Gruber's definition [Gruber, 93] becomes the most referenced on the literature: ``an ontology is an explicit specification of a conceptualization". In 1997, Borst [Borst, 97] slightly modify Gruber's definition saying that: ``Ontologies are defined as a formal specification of a shared conceptualization". These two definitions have been explained by Studer and Colleagues [Studer et al.; 98] as follows: "*conceptualization refers to an abstract model of some phenomenon in the world by having identified the relevant concepts of that phenomenon. Explicit means that the type of concepts used and the constraints on their use are explicitly defined. Formal refers to the fact that the ontology should be machine-readable. Shared reflects the notion that an ontology captures consensual knowledge, that is, it is not private to some individual, but accepted by a group*".

However, Gruber's definition is not the only accepted one. For example, according to Guarino, an ontology is a formal theory that constrains the possible conceptualisations of the world [Guarino, 98]. According to this author, in a "hard sense", the ontologies that the real applications use are really engineering adaptations of ontologies, not properly ontologies. Thus, for example, following this perspective, saying "the weight of a person is a mass" is a statement that can appear in an ontology, while the statement "the weight of a person is a number" can appear in an adaptation of the ontology because of constraints of time and resources. However, this strict idea of ontology is not ever used even by the most purist people in AI.

Other Definitions emerge as a consequence of how their authors build and use ontologies. We can distinguis here between top-down or botton-up approaches depending on what was first the ontology or the KB. Swartout and colleagues [Swartout et al., 97] define the word ontology as follow: ``an ontology is a hierarchically structured set of terms for describing a domain that can be used as a skeletal foundation for a knowledge base''. This definition is based on the fact that they have built SENSUS (a broad ontology wiht more than 50,000 terms) and they use it as a basis for building domain specific ontologies by identifying those terms in SENSUS that are relevant to a particular domain and then prune the skeletal ontology using heuristic. The result of the prune mechanism is the skeleton upon which the KB is built. So, we can say that this is a top down approach since the ontology was the starting point. The opposite approach was taken at the KACTUS project [Bernaras et al., 96], where the ontology is built after a process of abstraction of the content already represented in a knowledge base.

For a deeper study of ontology definitions we recommend [Guarino et al.; 95]. The authors summarise and comment seven different definitions of "ontology", six of them from the AI perspective. This work shows that, according to the different researchers, an ontology can be a logic theory, a formal semantic account, the vocabulary of a logic theory, a specification of a conceptualisation, etc.

**References:**


- A. Bernaras, I. Laresgoiti, J. Corera. Building and Reusing Ontologies for Electrical Network Applications. Proceedings of the European Conference on Artificial Intelligence (ECAI?96). ECAI 96. (1996). 298-302.
- W.N. Borst, Construction of Engineering Ontologies. University of Tweenty. Enschede, NL- Centre for Telematica and Information Technology. (1997).
- Chandrasekaran, B.; Johnson, T. R.; Benjamins, V. R. "Ontologies: what are they? why do we need them?". *IEEE Intelligent Systems and Their Applications*. 14(1). Special Issue on Ontologies. Pages 20-26. 1999.
- Gruber, T. R. "A translation approach to portable ontology specifications". *Knowledge Acquisition*. Vol. 5. 1993.
- Guarino, N.; Giaretta, P. "Ontologies and knowledge bases. towards a terminological clarification". *Toward Very Large Knowledge Bases*. Ed. IOS Press. 1995. Págs. 25-32.
- Neches, R.; Fikes, R.E.; Finin, T.; Gruber, T.R.; Senator, T.; Swartout, W.R. "Enabling technology for knowledge sharing". *AI Magazine*. 12(3)::36-56- 1991.
- B. Swartout; R. Patil; K. Knight; T. Russ. (1997) Toward distributed use of large-scale ontologies. In AAAI-97 Spring Symposium Series on Ontological Engineering.


## *1.3 What are the main components of an ontology?*

As we said before, ontologies provide a common vocabulary of an area and define –with different levels of formality- the meaning of the terms and the relations between them. Knowledge in ontologies is mainly formalized using five kinds of components: classes, relations, functions, axioms and instances [Gruber, 93]. Classes in the ontology are usually organized in taxonomies. Sometimes, the definition of ontologies have been diluted, in the sense that taxonomies are considered to be full ontologies [Studer et al., 98]. Such ontologies are called *lightweight* ontologies, in contrast with *heavyweight* ontologies, which includes many axioms.

- *Classes* in the ontology are usually organized in taxonomies. Sometimes, the notion of ontology is diluted, in the sense that taxonomies are considered to be full ontologies [Studer et al.; 98]. Classes or Concepts are used in a broad sense. A concept can be anything about which something is said and, therefore, could also be the description of a task, function, action, strategy, reasoning process, etc.

- *Relations* represent a type of interaction between concepts of the domain. They are formally defined as any subset of a product of n sets, that is: R: C1 x C2 x ... x Cn. Examples of binary relations include: subclass-of and connected-to.

- *Functions* are a special case of relations in which the n-the element of the relationship is unique for the n-1 preceding elements. Formally, functions are defined as: F: C1 x C2 x ... x Cn-1 → Cn. Examples of functions are Mother-of and Price-of-a-used-car that calculates the price of a second-hand car depending on the car-model, manufacturing date and number of kilometres.
- *Axioms* are used to model sentences that are always true. They can be included in an ontology for several purposes, such as defining the meaning of ontology components, defining complex constrains on the values of attributes, the arguments of relations, etc., verifying the correctness of the information specified in the ontology or deducing new information.
- *Instances* are used to represent specific elements.

Once the main components of ontologies has been presented, the next question is What does an explicit ontology look like? Uschold and Grüninger [Uschold et al., 96]) have distinguished four kind of ontologies depending on the kind of language used to implement them. They are: Highly informal ontologies if they are written in natural language; Semi-formal ontologies if they are expressed in a restricted and structured form of natural language (i.e., using patterns); Semi-formal ontologies, which are defined in an artificial and formally defined language; and Rigorously formal ontologies if they are defined in a language with formal semantics, theorems and proofs of such properties as soundness and completeness.

**References:**


- Gruber, T. R. “A translation approach to portable ontology specifications”. *Knowledge Acquisition*. Vol. 5. 1993.
- Uschold, M.; Grüninger, M. 1996. "Ontologies: Principles Methods and Applications" Knowledge Engineering Review. Vol. 2.
- Studer, R., Benjamins, R., Fensel, D. *Knowledge Engineering: Principles and Methods*. DKE 25(1-2).pp:161-197. 1998


## *1.4 Most well known ontologies*

This section does not seek to give an exhaustive typology of ontologies as presented in [vanHeijst et al., 97] and [Mizoguchi et al., 95]. However, it presents the most commonly used types of ontologies so the reader can get an idea of the knowledge to be included in each type of ontology. Basically, the following categories are identified: Knowledge representation ontologies, meta-ontologies, domain ontologies, tasks ontologies, domain-task ontologies, application ontologies, index ontologies, tell and ask ontologies, etc.

- Knowledge Representation ontologies (vanHeijst et al., 97) capture the representation primitives used to formalized knowledge in knowledge representation paradigms. The most representative example of this kind of ontologies is the Frame-Ontology (Gruber, 93), which captures the representation primitives (classes, instances, slots, facets, etc.) used in frame-based languages.
- General/Common ontologies (Mizoguchi et al., 95) include vocabulary related to things, events, time, space, causality, behavior, function, etc.
- Meta-ontologies, also called Generic Ontologies or Core Ontologies (vanHeijst et al., 97), which are reusable across domains. The most representative example could be a mereology ontology (Borst, 97) which would include the term part-of.
- Domain ontologies (Mizoguchi et al., 95) (vanHeijst et al., 97) are reusable in a given domain. They provide vocabularies about the concepts within a domain and their relationships, about the activities that take place in that domain, and about the theories and elementary principles governing that domain.
- Task ontologies (Mizoguchi et al., 95) provide a systematized vocabulary of the terms used to solve problems associated with tasks that may or may not be from the same domain. These ontologies provide a set of terms by means of which to generically describe how to solve one type of problems. They include generic names, generic verbs, generic adjectives and others in the scheduling tasks.
- Domain-Task ontologies are task ontologies reusable in a given domain, but not across domains.
- Application ontologies (vanHeijst et al., 97) contains the necessary knowledge for modeling a particular domain.

Along this section, we will briefly describe the most relevant top level ontologies, general ontologies, linguistic ontologies, and domain ontologies

## 1.4.1 Knowledge Representation ontologies

### 1.4.1.1 The Frame Ontology

The frame ontology defines the terms that capture conventions used in object-centered knowledge representation systems. Since these terms are built on the semantics of KIF, one can think of KIF plus the frame-ontology as a specialized representation language. The frame ontology is the conceptual basis for the Ontolingua translators.

One purpose of this ontology is to enable people using different representation systems to share ontologies that are organized along object-centered, term-subsumption lines. Translators of ontologies written in KIF using the frame ontology, such as those provided by Ontolingua, allow one to work from a common source format and yet continue to use existing representation systems.

The definitions in this ontology include and extend the Generic Frame Protocol knowledge model (version 2.0). However, there is no claim that these definitions capture the semantics of existing, implemented systems in full detail. Nuances of the meaning of terms that depend on the algorithms for inheritance, for instance, are not ;addressed in this ontology. See the acknowledgements at the end of ;the file.

This ontology is specified using the definitional forms provided by Ontolingua. All of the embedded sentences are in KIF 3.0, and the entire thing can be translated into pure KIF top level forms without loss of information. Some basic ontological commitments of this ontology are: relations are sets of tuples (named by predicates), functions are a special case of relations, classes are unary relations (no special syntax for types), etc.

**Languages in which the ontology is implemented:** Ontolingua

**Important projects in which the ontology is used:** *Knowledge Sharing Effort*

**URL:** http://www-ksl-svc.stanford.edu:5915

**References:**


- Gruber, T. R. “A translation approach to portable ontology specifications”. *Knowledge Acquisition*. Vol. 5. 1993.


**Size:** number of concepts: Between 10 & 50; number of relations: Between 10 & 50; number of axioms: Between 100 & 500; number of instances: 0

## 1.4.2 Top level ontologies

In a top level or upper level ontology, the following characteristics are desirable (http://www.cyc.com/cyc-2-1/cover.html):

a) It is "universal", that is, every concept one can imagine can be correctly linked into the upper ontology in appropriate places, no matter how general or specific, no matter how arcane or prosaic, no matter what the context (nationality, age, native language, epoch, childhood experiences, current goals, etc.) of the imaginer.

b) It is "articulate" The distinctions which are made in the ontology are both necessary and sufficient for most purposes: By "necessary" they mean that the distinctions are all worth making. There is both theoretical and pragmatic justification for every class, for every predicate and function, for every individual. By "sufficient" they mean that enough distinctions have been made to enable and support knowledge sharing, natural language disambiguation, database cleaning and integration, and other applications.

We will present some of the most known top level ontologies. You can find more information about top level ontologies in http://www-sop.inria.fr/acacia/personnel/phmartin/RDF/phOntology.html.

### 1.4.2.1 Sowa’s Top Level Ontology

Sowa’s top level ontology includes the basic categories and distinctions have been derived from a variety of sources in logic, linguistics, philosophy, and artificial intelligence. The two most important influences have been the philosophers Charles Sanders Peirce and Alfred North Whitehead, who were pioneers in symbolic logic.

This ontology has a lattice structure where the top concept is the *universal type*, and the bottom concept is the *absurd type*. As subtypes of the universal one, there are primitive concepts (*independent*, *relative*, *mediating*, *continuant*, *occurrent*). Combining these primitive concepts, more concepts of the net are obtained, for example, *history = proposition Ç occurrent*.. Continuing the process of combining concepts, the complete lattice is obtained. To close the lattice, the absurd type is considered a subtype of all the concepts without children.

**Languages in which the ontology is implemented:** *Unknown*

**Important projects in which the ontology is used:** *Unknown*

**URL:** http://users.bestweb.net/~sowa/ontology/index.htm

**References:**

- John F. Sowa, Knowledge Representation: Logical, Philosophical, and Computational Foundations, Brooks Cole Publishing Co., Pacific Grove, CA, 1999.

**Size:** number of concepts: Between 10 & 50; number of relations: 0; number of axioms: 0; number of instances: 0

### 1.4.2.2 Top level of particulars of Guarino's group

Guarino's group distinguish between universals and particulars. This distinction has led to build a top level ontology of universals, and a top level ontology of particulars. Particulars are very abstract concepts, for example, object or event, that can be instantiated in particular elements, for example, a computer is an object, and the reading of a hard disk is an event. On the other hand, Universals are also very abstract concepts, for example, type or role, that can be instantiated in particulars. For instance, the property computer is a type. It is possible to prove that every property can be instantiated in at least one of the leaves of the hierarchy. The top level of universal as well as the top level of particulars were implemented using WebODE tool.

**List of the languages in which the ontology is implemented:** XML

**Important projects in which the ontology is used:** Eureka Project IKF (E-2235, Intelligent Knowledge Fusion), National Italian project TICCA (Tecnologie Cognitive per l’Interazione e la Coperazione con Agenti)

**URL:** *Not available*.

**References:**

- Gangemi, A., Guarino, N., Oltramari A. 2001. “Conceptual Analysis of Lexical Taxonomies: The Case of WordNet Top-Level”. Proc. of FOIS2001.
- Guarino, N.; Welty, C. 2000. “A Formal Ontology of Properties”. In R. Dieng and O. Corby (eds.), Knowledge Engineering and Knowledge Management: Methods, Models and Tools. 12th International Conference, EKAW2000. Springer Verlag: 97-112.

**Size:** number of concepts: Between 10 & 50; number of relations: 0; number of axioms: Less than 10; number of instances: 0

### 1.4.2.3 Standard Upper Ontology

It is the Standard Upper Merged Ontology promoted by the IEEE Standard Upper Ontology effort. The purpose of this standard is to specify an upper ontology that will enable computers to utilise it for applications such as data interoperability, information search and retrieval, automated inference, and natural language processing. SUO will provide a structure and a set of general concepts upon which domain ontologies (e.g. medical, financial, engineering, etc.) could be constructed. It is implemented in DAML+OIL (see section of languages).

**Languages in which the ontology is implemented:** DAML+OIL

**Important projects in which the ontology is used:** *Unknown*

**URL:** http://reliant.teknowledge.com/DAML/SUO.daml, http://suo.ieee.org/

**References:** *Not available.*

**Size:** number of concepts: Between 100 & 500; number of relations: Between 100 & 500; number of axioms: 0; number of instances: Between 1000 & 5000

### 1.4.2.4 *Upper Cyc® ontology*

It belongs to Cyc KB [Lenat et al.; 90] (see also the Cyc description in the section of general ontologies). Upper Cyc contains around 3,000 terms capturing the most general concepts of human consensus reality. It is implemented in CycL.

**Languages in which the ontology is implemented:** DAML+OIL, CycL

**Important projects in which the ontology is used:** Cyc project

**URL:** http://www.cyc.com/cyc-2-1/cover.html , www.daml.org/ontologies

**References:**

- Lenat, D.B.; Guha, R.V. "Building large knowledge-based systems". *Addison-Wesley Publising Company, Inc*. 1990.

**Size:** number of concepts: About 3000; number of relations: unknown; number of axioms: unknown; number of instances: unknown

### 1.4.2.5 *WordNet top level ontology*

The WordNet ontology will be described in the section of linguistic ontologies. It has very abstract concepts that can be considered belonging to a top level ontology.

**Languages in which the ontology is implemented:** *Unknown*

**Important projects in which the ontology is used:**

- PANGLOSS
- EuroWordNet
- WebKB

**URL:** http://cogsci.princeton.edu/~wn/

**References:**

- Fellbaum, Christiane, ed., WordNet: "An Electronic Lexical Database", MIT Press, May 1998. Fellbaum, Christiane. ``English verbs as a semantic net." In: International Journal of Lexicography 3 (4), 1990, pp. 278 - 301.
- Gross, Derek and Katherine J. Miller. ``Adjectives in WordNet." In: International Journal of Lexicography 3 (4), 1990, pp. 265 - 277.
- Miller, George A. ``WordNet: a lexical database for English." In: Communications of the ACM 38 (11), November 1995, pp. 39 – 41.
- Miller, George A., Richard Beckwith, Christiane Fellbaum, Derek Gross and Katherine J. Miller. ``Introduction to WordNet: an on-line lexical database." In: International Journal of Lexicography 3 (4), 1990, pp. 235 - 244.
- Miller, George A. ``Nouns in WordNet: a lexical inheritance system." In: International Journal of Lexicography 3 (4), 1990, pp. 245 - 264.

**Size:** number of concepts: Between 10 & 50; number of relations: 0; number of axioms: 0; number of instances: 0

## 1.4.3 Linguistic ontologies

This section will compile information about linguistic ontologies. The main characteristic of this kind of ontologies is that they are bound to the semantics of grammatical units (words, nominal groups, etc.). Regarding other features, they form quite a heterogeneous group of resources, used mostly in natural language processing .Most linguistic ontologies use words as their grammatical units - in fact, of the ontologies reviewed in this report, only the Generalized Upper Model captures information about grammatical units that are bigger than words -but some are guided by word form (e.g. EDR dictionaries), while others are led by word renaming (e.g. WordNet). Moreover, in some of them, there is a one-to-one mapping between concepts and words in a natural language (e.g. WordNet), while in others, many concepts may not map to any word in a language or may map to more than one word in the same language (e. g. Mikrokosmos). There are also differences in respect of their degree of language dependency. Some linguistic ontologies depend totally on a single language (e. g. WordNet); some are multilingual - i.e., are valid for several languages - (e. g. GUM); some contain a language-dependent part and a language-independent one (e. g. EuroWordNet); and others are language independent (e. g. Mikrokosmos). The origin and motivations of these ontologies are varied too: on-line lexical databases (e. g. WordNet), machine translation (e. g. Sensus), natural language generation (e.g. GUM).

### 1.4.3.1 CoreLex

CoreLex is an ontology, lexical semantic database and tagset for nouns, organized around systematic polysemy and underspecification. CoreLex was developed out of a thesis on systematic polysemy and underspecification of nouns, establishing an ontology and semantic database of 126 semantic types, covering around 40,000 nouns and defining a large number of systematic polysemous classes that are derived by a careful analysis of sense distributions in WordNet. The semantic types are underspecified representations based on Generative Lexicon theory and are used in an underspecified approach to semantic tagging, addressing two problems: sense enumeration (the difficulty of deciding the number of discrete senses), due to systematic polysemy; and multiple reference (NP's denoting more than one model-theoretic referent), due to underspecification.

**Languages in which the ontology is implemented:** *Unknown*

**Important projects in which the ontology is used:** *Unknown*

**URL:** http://www.cs.brandeis.edu/~paulb/CoreLex/corelex.html

**References:**


- Paul Buitelaar. Semantic Lexicons: Between Ontology and Terminology. In: Proceedings of OntoLex 2000: Ontologies and Lexical Knowledge Bases, OntoText Lab., Sofia, Bulgaria, 2001.
- Paul Buitelaar. CoreLex: Systematic Polysemy and Underspecification PhD Thesis, Computer Science, Brandeis University, February 1998
- Paul Buitelaar. CoreLex: An Ontology of Systematic Polysemous Classes. In: Proceedings of FOIS98, International Conference on Formal Ontology in Information Systems , Trento, Italy, June 6-8, 1998.


**Size:** number of concepts: Between 100 & 500; number of relations: Unknown; number of axioms: Unknown; number of instances: Unknown

### 1.4.3.2 EDR Electronic Dictionary

The EDR Electronic Dictionary was developed for advanced natural language processing. It integrates the relationship between lexical entries and their concepts in the form of concepts hierarchy and semantic relations, together with database of corpus from which lexical and conceptual information were extracted. It consists of five types of large-scale dictionaries: three of them are used to catalogue the lexical knowledge of Japanese and English (the Word Dictionary, the Bilingual Dictionary, and the Co-occurrence Dictionary), one is a unified thesaurus-like concept classifications (the Concept Dictionary) and another has corpus databases (the EDR Corpus).
The EDR Electronic Dictionary is the product of a nine-year project (from 1986 to 1994) and was funded by the Japan Key Technology Center and eight computer manufacturers.
The main approach taken during the project was to avoid a particular linguistic theory and to allow for adaptability to various applications

**Languages in which the ontology is implemented:** *Unknown*

**Important projects in which the ontology is used:** *Unknown*

**URL:** http://www.iijnet.or.jp/edr/

**References:**

- H. Miyoshi, K. Sugiyama, M. Kobayashi and T. Ogino. An overview of the EDR Electronic Dictionary and the Current Status of its utilization. Proceedings of COLING-96, August, 1996.
- T. Ogino, H. Miyoshi, F. Nishino and M. Kobayashi. An experiment on matching EDR concept classification dictionary with Wordnet. IJCAI-97 Workshop WP24 Ontologies and Multilingual NLP, August, 1997.

**Size:** number of concepts: More than 5000; number of relations: Unknown; number of axioms: Unknown; number of instances: Unknown

### 1.4.3.3 EuroWordNet *(see section 1.4.2.8)*

EuroWordNet is a multilingual database with wordnets for several European languages (Dutch, Italian, Spanish, German, French, Czech, Estonian). The wordnets are structured in the same way as the Princeton Wordnet for English in terms of synsets (sets of synonymous words) with basic semantic relations between them. Each wordnet represents a unique language-internal system of lexicalizations. In addition, the wordnets are linked to an Inter-Lingual-Index. Via this index, the languages are interconnected so that it is possible to go from the words in one language to similar words in any other language, and compare synsets and their relations across languages. The index also gives access to a shared top-ontology of 63 semantic distinctions. This top-ontology provides a common semantic framework for all the languages, while language specific properties are maintained in the individual wordnets. The database can be used, among others, for monolingual and cross-lingual information retrieval.

**Languages in which the ontology is implemented:** *Unknown*

**Important projects in which the ontology is used:** ITEM

**URL:** http://www.hum.uva.nl/~ewn/

**References:**

- Vossen Piek (ed.), EuroWordNet General Document. Version 3, Final, July 19, 1999. Vossen, P. (eds) (1998) EuroWordNet: A Multilingual Database with Lexical Semantic Networks, Kluwer Academic Publishers, Dordrecht

**Size:** number of concepts: More than 5000; number of relations: Unknown; number of axioms: Unknown; number of instances: Unknown

### 1.4.3.4 Goi-Taikei's ontology

Goi-Taikei (GT) is a 400,000-word Japanese lexicon developed by NTT for machine translation applications. GT consists of three main components: (i) an ontology, (ii) a semantic word dictionary, and (iii) a semantic structure dictionary.GT?s ontology classifies concepts to use in expressing relationships between words. The meanings of common nouns are given in terms of a semantic hierarchy of 2,710 nodes. Each node represents a semantic class. Edges in the hierarchy represent is-a or has-a relationships. In addition to the 2,710 classes for common nouns, there are 200 classes for proper nouns and 108 classes for predicates. Words can be assigned to semantic classes anywhere in the hierarchy. Not all semantic classes have words assigned to them. The semantic classes are used in the Japanese word semantic dictionary to classify nouns, verbs and adjectives. The semantic classes are also used as selectional restrictions on the arguments of predicates in a separate predicate dictionary.

**Languages in which the ontology is implemented:** *Unknown*

**Important projects in which the ontology is used:** ALT-J/E

**URL:** http://www.kecl.ntt.co.jp/icl/mtg/topics/lexicon-index.html

**References:**

- S. Ikehara, M. Miyazaki, S. Shirai, A. Yokoo, H. Nakaiwa, K. Ogura, Y. Ooyatna and Y. Hayashi (1997). Goi-Taikei ? A Japanese Lexicon. Iwanami Shoten, Tokyo. (5 volumes/CD-ROM).
- Bond F., Yamazaki T., Sulong R. B. and Ogura K (2001) Design and Construction of a machine - tractable Japanese-Malay Lexicon. In 7th Annual Meeting of the Association for Natural Language Processing.

**Size:** number of concepts: Between 1000 & 5000; number of relations: Unknown; number of axioms: Unknown; number of instances: Unknown

### 1.4.3.5 Generalized Upper Model

The Generalized Upper Model (GUM) is a general task and domain independent `linguistically motivated ontology' that supports sophisticated natural language processing while significantly simplifying the interface between domain-specific knowledge and general linguistic resources. The Generalized Upper Model occupies a level of abstraction midway between surface linguistic realizations and `conceptual' or `contextual' representations. It enables abstraction beyond the concrete details of syntactic and lexical representations, while still maintaining close enough contact with linguistic realizations to be solidly founded on objective criteria. That is: if there is no specifiable lexicogrammatical consequences for a `concept', then it does not belong in the GUM. GUM is split into two hierarchies. The first one contains all the concepts and the second one contains all the roles.

**Languages in which the ontology is implemented:** LOOM

**Important projects in which the ontology is used:** AlFresco, GIST, KPML, Komet, Ontogeneration, Penman, TechDoc.

**URL:** http://www.darmstadt.gmd.de/publish/komet/gen-um/newUM.html

**References:**


- J. A. Bateman, R. Henschel and F. Rinaldi (1995), Generalized Upper Model 2.0: documentation, Technical Report, GMD/IPSI, Darmstadt, Germany.
- Bateman, J., Magnini, B. and Fabris, G. (1995), The generalized upper model knowledge base: Organization and use, in `Proceedings of the Conference on Knowledge Representation and Sharing', Twente, the Netherlands.
- Bateman, J. A., Magnini, B. and Rinaldi, F. (1994), The Generalized Italian,German,English Upper Model, in `Proceedings of the ECAI94 Workshop: Comparison of Implemented Ontologies', Amsterdam.
- Henschel, R. and Bateman, J. (1994), The merged upper model: a linguistic ontology for German and English, in `Proceedings of COLING '94', Kyoto, Japan.
- Bateman, J. A., Kasper, R. T., Moore, J. D. and Whitney, R. A. (1990), A general organization of knowledge for natural language processing: the PENMAN upper model, Technical report, USC/Information Sciences Institute, Marina del Rey, California.


**Size:** number of concepts: Between 100 & 500; number of relations: Unknown; number of axioms: Unknown; number of instances: Unknown

### 1.4.3.6 Mikrokosmos' ontology

The Mikrokosmos Ontology is a part of the Mikrokosmos machine translation project. Mikrokosmos is not committed to any particular theory of ontologies, but is built on more practical considerations. The main principle is a careful distinction between language-specific knowledge, represented in the lexicon, and language-neutral knowledge represented in the ontology. As a consequence, the semantics of words is represented partly in the lexical entries and partly in the ontological concepts. A set of detailed guidelines governs what belongs in a concept and what belongs in a lexical entry. The division of semantics also gives us the answer to how concepts are related to lexical items. In Mikrokosmos you are not forced to have one-to-one mapping between words and concepts. Words with related but not equivalent meanings can map to the same concept, while the differences are captured in the lexical entries.

**Languages in which the ontology is implemented:** *Unknown*

**Important projects in which the ontology is used:** Mikrokosmos, OncoTerm.

**URL:** http://crl.nmsu.edu/mikro

**References:**


- Mahesh, K. (1996). Ontology development for machine translation: Ideology and Methodology. Technical Report MCCS-96-292. Computing Research Laboratory, New Mexico State University, Las Cruces, NM.
- Mahesh, K. and S. Nirenburg. (1995). A Situated Ontology for Practical NLP. Proceedings of the Workshop on Basic Ontological Issues in Knowledge Sharing, International Joint Conference on Artificial Intelligence, Montréal, Canada.

**Size:** number of concepts: Between 1000 & 5000; number of relations: Unknown; number of axioms: Unknown; number of instances: Unknown

#### 1.4.3.7 SENSUS

SENSUS is a natural language based ontology developed by the Natural Language group at ISI in order to provide a broad conceptual structure for work in machine translation. SENSUS contains tens of thousands of nodes (more than 70,000) representing commonly encountered objects, entities, qualities and relations. SENSUS provides a hierarchically structured concept base. The upper (more abstract) region of the ontology is called the Ontology Base and consists of approximately 400 items that represent generalizations essential for the linguistic processing during translation. The middle region of the ontology, approximately 50,000 items, provides a framework for a generic world model, containing items representing many English word senses. The lower (more specific) regions of the ontology provide anchor points for different application domains.

**Languages in which the ontology is implemented:** LOOM

**Important projects in which the ontology is used:** GAZELLE, OntoSeek, PANGLOSS.

**URL:** http://www.isi.edu/natural-language/projects/ONTOLOGIES.html

**References:**


- K. Knight. and S. Luk. (1994). Building a Large Knowledge Base for Machine Translation. Proceedings of the American Association of Artificial Intelligence Conference (AAAI-94). Seattle, WA.
- K. Knight, I. Chander, M. Haines, V. Hatzivassiloglou, E. H. Hovy, M. Iida, S.K. Luk, R.A. Whitney, and K. Yamada. (1995). Filling Knowledge Gaps in a Broad-Coverage MT System. Proceedings of the 14th IJCAI Conference. Montreal, Quebec.
- B. Swartout; R. Patil; K. Knight; T. Russ. (1997) Toward distributed use of large-scale ontologies. In AAAI-97 Spring Symposium Series on Ontological Engineering.


**Size:** number of concepts: More than 5000; number of relations: Unknown; number of axioms: Unknown; number of instances: Unknown

#### 1.4.3.8 Wordnet

WordNet is a very large lexical database for English based on psycholinguistic theories. It is organized into 70,000 sets of synonyms ("synsets"), each representing one underlying lexical concept. Synsets are linked with each other via relationships such as hynonymy and antonymy, hypernymy and hyponymy, meronymy and holonymy. Approximately one half of the synsets have short English explanations of their intuitive sense. Wordnet divides the lexicon into five categories: nouns, verbs, adjectives, adverbs and function words. Nouns are organized as topical hierarchies, verbs are organized by a variety of entailment relations and adjectives and adverbs are organized as N-dimensional hyperspaces. Each of these lexical organizations reflects a different way of categorizing experiences and the psychological complexity of lexical knowledge. The last version contains 121,962 words and 99,642 concepts.

**Languages in which the ontology is implemented:** *Unknown*

**Important projects in which the ontology is used:** WebKB-2.

**URL:** http://www.cogsci.princeton.edu/~wn/

**References:** *see section 1.4.1.5.*

**Size:** number of concepts: More than 5000; number of relations: Unknown; number of axioms: Unknown; number of instances: Unknown

### 1.4.4 General ontologies

This section not only present already well-known general and domain independent ontologies, but also a few standards on the e-commerce field. A deeper analysis about e-commerce vocabulary standardization can be found at D3.1.

#### 1.4.4.1 Cyc

Cyc methodology arose from the experience on the development of the Cyc knowledge base (KB), which contains a huge amount of common sense knowledge, and which is being built upon a core of over 1,000,000 hand-entered assertions designed to capture a large portion of what people normally consider consensus knowledge about the world. To codify such KB, the used language is CycL, an augmentation of first-order predicate calculus, with extensions to handle equality, default reasoning, skolemization, and some second-order features (for example, quantification over predicates). The reason why Cyc KB can be considered as an ontology is because it can be useful as substrate for building different intelligent systems and also as a base for their communication and interoperatibility.

**Languages in which the ontology is implemented:** DAML+OIL, CycL

**Important projects in which the ontology is used:** Cyc project

**URL:** http://www.cyc.com

**References:**

- Lenat, D.B.; Guha, R.V. "Building large knowledge-based systems". *Addison-Wesley Publishing Company, Inc*. 1990.

**Size:** number of concepts: More than 5000; number of relations: Unknown; number of axioms: Unknown; number of instances: Unknown

**Applications where it is used:** Knowledge-Enhanced Searching of Captioned Information, Guided Integration of Structured Terminology

#### 1.4.4.2 E-cl@ss

E-cl@ss is a German initiative to create a standard classification of material and services for information exchange between suppliers and their customers. In fact, it is similar to the UNSPSC initiative, and will be used by companies like BASF, Bayer, Volkswagen-Audi, SAP, etc. The e-cl@ss classification consists of four levels of concepts (called material classes), with a numbering code similar to the one used in UNSPSC (each level has two digits that distinguish it from the other concepts). The four levels are: Segment, Main group, Group and Commodity Class. These levels are equivalent to the first four ones provided in UNSPSC; hence, they are not described any further. Finally, inside the same commodity class we can have several products (in this sense, several products can share the same code, and this could lead to a fifth level with all of them).

**URL:** http://www.eclass.de

**References:** *Not available.*

**Size:** number of concepts: More than 5000; number of relations: 0; number of axioms: 0; number of instances: 0

**Applications where it is used:** *Not available.*

#### 1.4.4.3 UNSPSC

UNSPSC is a non-profit organisation composed of partners such as 3M, AOL, Arthur Andersen, BT, Castrol and others. Its coding system is organised as a five-level taxonomy of products, each level containing a two-character numerical value and a textual description. The current version of the UNSPSC classification contains around 12000 products organized in 54 segments.

**Languages in which the ontology is implemented:** It is in HTML format

**Important projects in which the ontology is used:** *Unknown*

**URL:** http://www.eclass.de

**References:** *Not available.*

**Size:** number of concepts: around 12000; number of relations: 0; number of axioms: 0; number of instances: 0

**Applications where it is used:** *Not available.*

## 1.4.5 Domain ontologies

### 1.4.5.1 Chemistry ontologies

#### 1) CHEMICALS

It is an ontology built according to METHONTOLOGY, it contains knowledge within the domain of chemical elements and crystalline structures.

**Languages in which the ontology is implemented:** Ontolingua

**Important projects in which the ontology is used:**

- OntoGeneration
- ChemicalOntoAgent

**URL:** delicias.dia.fi.upm.es/webODE/, www-ksl-svc.stanford.edu:5915/

**References:**

- Fernández-López, M.; Gómez-Pérez, A.; Pazos-Sierra, A.; Pazos-Sierra, J. 1999. "Building a Chemical Ontology Using METHONTOLOGY and the Ontology Design Environment". IEEE Intelligent Systems & their applications. January/February Pages 37-46.

**Size:** number of concepts: Between 10 & 50; number of relations: Between 10 & 50; number of axioms: Between 10 & 50; number of instances: Between 100 & 500.

#### 2) Environmental pollutants

These ontologies represent the methods of detecting the different pollutant components of various media: water, air, soil, etc., and the maximum permitted concentrations of these components, taking into account all the legislation in force.

**Domain:** pollutant chemical elements

**Languages in which the ontology is implemented:** XML

**Important projects in which the ontology is used:** Prototipos de Ontologías para el Medio ambiente (code AM9819)

**URL:** delicias.dia.fi.upm.es/webODE/

**References:**

- Gómez-Pérez, A; Rojas, M. D. 1999. "Ontological Reengineering and Reuse". European Knowledge Acquisition Workshop (EKAW).

**Size:** number of concepts: Between 50 & 100; number of relations: Less than 10; number of axioms: Less than 10; number of instances: 0

**Applications where it is used:** *Not available.*

### 1.4.5.2 Enterprise ontologies

These ontologies are usually created for the definition and organization of relevant knowledge about activities, processes, organizations, strategies, marketing, etc. All this knowledge is meant to be used by enterprises.

#### 1) Enterprise Ontology

The ontology was developed in the Enterprise Project by the Artificial Intelligence Applications Institute at the University of Edinburgh with its partners: IBM, Lloyd's Register, Logica UK Limited, and Unilever. The project was support by the UK's Department of Trade and Industry under the Intelligent Systems Integration Programme(project no IED4/1/8032). This ontology is a collection of terms and definitions relevant to business enterprises and includes knowledge about activities and processes, organizations, strategies, marketing, etc.

**Languages in which the ontology is implemented:** Ontolingua

**Important projects in which the ontology is used:** Intelligent Systems Integration Programme (project no IED4/1/8032)

**URL:** http://www.aiai.ed.ac.uk/~entprise/enterprise/ontology.html

**References:**

- Uschold, M.; King, M.; Moralee, S.; Zorgios, Y. (1998). ?The Enterprise Ontology?. The Knowledge Engineering Review , Vol. 13, Special Issue on Putting Ontologies to Use (eds. Mike Uschold and Austin Tate). Also available from AIAI as AIAI-TR-195.

**Size:** number of concepts: Between 50 & 100; number of relations: Between 50 & 100; number of axioms: 0; number of instances: Between 10 & 50

## 2) TOVE

The goal of the TOVE (TOronto Virtual Enterprise) project is to create a data model that has the following characteristics: 1) provides a shared terminology for the enterprise that each agent can jointly understand and use, 2) defines the meaning of each term (aka semantics) in a precise and as unambiguous manner as possible, 3) implements the semantics in a set of axioms that will enable TOVE to automatically deduce the answer to many "common sense" questions about the enterprise, and 4) defines a symbology for depicting a term or the concept constructed thereof in a graphical context.

**URL:** http://www.eil.utoronto.ca/tove/toveont.html

**References:**

- [Fox 81] Fox, M.S. An Organizational View of Distributed Systems. IEEE Transactions on Systems, Man, and Cybernetics. SMC-11(1):70-80, 1981

**Size:** number of concepts: Unknown; number of relations: Unknown; number of axioms: Unknown; number of instances: Unknown

**Applications where it is used:** *Not available.*

### 1.4.5.3 Engineering Ontologies

In the domain of **engineering ontologies**, special mention deserve the EngMath ontology (Gruber et al., 95) and PhysSys (Borst, 97). *EngMath* is an Ontolingua ontology developed for mathematical modelling in engineering. The ontology includes conceptual foundations for scalar, vector, and tensor quantities, physical dimensions, units of measure, functions of quantities, and dimension quantities. PhysSys is an engineering ontology for modeling, simulating and designing physical systems. It consists of three engineering ontologies formalizing the three viewpoints on physical devices: system layout, physical process behavior and descriptive mathematical relations. Three engineering ontologies formalize each of these viewpoints: a component ontology, a process ontology and the EngMath ontology. The interdependencies between these ontologies are formalized as ontology projections. These ontologies use other meta-ontologies: mereology, topology and system theories.

**Languages in which the ontology is implemented:** Ontolingua

**Important projects in which the ontology is used:** *Unknown*

**URL:** http://

**References:**

- Borst, W. N. *Construction of Engineering Ontologies*. University of Tweenty. Enschede, NL- Centre for Telematica and Information Technology. 1997.
- Gruber, T. R. Toward Principles of the Design of Ontologies Used for Knowledge Sharing. International Journa of Human Computer Studies. 43. 1995. Pp. 907-928.

**Size:** *unknown*

### 1.4.5.4 Medical ontologies

Medical ontologies are introduced for solving problems such as the demand for re-use and sharing of patient data, their transmission and the need of semantic-based criteria for purposive statistical. In this sense, the unambiguous communication of complex and detailed medical concepts is now a crucial feature of medical information systems.

### 1) GALEN

At the heart of GALEN is a semantically valid model of clinical terminology, represented in a formal language, and associated with sophisticated support for different natural languages and conversion between different coding schemes. GALEN is based on a semantically sound model of clinical terminology: the GALEN Coding reference (CORE) model. This model comprises elementary clinical concepts (such as 'fracture', 'bone', 'left', and 'humerus'), relationships (such as 'fractures can occur in bones'), that control how these may be combined, and complex concepts (such as 'fracture of the left humerus') composed from simpler ones.

**Languages in which the ontology is implemented:** GRAIL

**Important projects in which the ontology is used:** *Unknown*

**URL:** http://www.opengalen.org/

**References:**

- A. Rector, W. Solomon, W. Nowlan and T. Rush (1995). A Terminology Server for Medical Language and Medical Information Systems. Methods of Information in Medicine, Vol. 34, 147-157.
- J. Wagner, J. Rogers, R. Baud and J-R. Scherrer (1999). Natural language generation of surgical procedures International Journal of Medical Informatics; Vol. 53, NOS. 2,3 pp. 175-192.

**Size:** number of concepts: Between 1000 & 5000; number of relations: Between 100 & 500; number of axioms: Between 1000 & 5000; number of instances: Unknown

### 2) UMLS

The Unified Medical Language System (UMLS) is large database designed to link biomedical vocabularies together from disparate sources such as clinical terminologies, drug sources, vocabularies in different languages, and clinical terminologies. There are three UMLS Knowledge Sources, which are the following: The Metathesaurus contains semantic information about biomedical concepts, their various names, and the relationships among them. The Semantic Network is a network of the general categories or semantic types to which all concepts in the Metathesaurus have been assigned. The Specialist lexicon contains syntactic information about biomedical terms and will eventually cover the majority of component terms in the concept names present in the Metathesaurus.

**Languages in which the ontology is implemented:** *Unknown*

**Important projects in which the ontology is used:** *Unknown*

**URL:** http://www.nlm.nih.gov/research/umls/

**References:**

- National Library of Medicine, UMLS Knowledge Sources, 1999 edition, available from the NLM, Bethesda, Maryland.
- D. Pisanelli, A. Gangemi and G. Steve. An Ontological Analysis of the UMLS Methatesaurus. Proceedings of AMIA 98 Conference, 1998.

**Size:** number of concepts: More than 5000; number of relations: Between 50 & 100; number of axioms: Unknown; number of instances: Unknown

### 3) MeSH

MeSH is the National Library of Medicine's controlled vocabulary thesaurus. Thesauri are carefully constructed sets of terms often connected by "broader-than," narrower-than," and "related" links. These links show the relationship between related terms and provide a hierarchical structure that permits searching at various levels of specificity from narrower to broader. MeSH consists of a set of terms or subject headings that are arranged in both an alphabetic and a hierarchical structure. At the most general level of the hierarchical structure are very broad headings such as "Anatomy," "Mental Disorders," and "Enzymes, Coenzymes, and Enzyme Inhibitors." At more narrow levels are found more specific headings such as "Ankle", "Conduct Disorder," and "Calcineurin".

**Languages in which the ontology is implemented:** *Unknown*

**Important projects in which the ontology is used:** *Unknown*

**URL:** http://www.nlm.nih.gov/mesh/meshhome.html

**References:**

- N. Stuart, A. Aronson, T. Doszkocs, J. Wilbur, O. Bodenreider, F. Chang, J. Mork and A. McCray. Automated Assignment of Medical Subject Headings. Poster presentation at: AMIA 1999 Annual Symp.; 1999 Nov 9; Washington, DC.
- S. Nelson, D. Johnston and B. Humphreys. Relationships in Medical Subject Headings. In: C. Bean and R. Green (eds.). Relationships in the organization of knowledge. New York: Kluwer Academic Publishers.

**Size:** number of concepts: More than 5000; number of relations: Unknown; number of axioms: Unknown; number of instances: Unknown

### 4) ON9

ON9 is a medical ontology developed with the ONION methodology and it include some terminology systems, such as UMLS. ON9 use some representation ontologies provided by default in Ontolingua, such as the "frame-ontology" and the set of "kif-ontologies". ON9 define the following ontologies: “etaontology", "semantic-fieldontology" and "structuring-concepts" in order to link the representation ontologies with the generic ontology library. The sets of "structural ontologies" and of "structuring ontologies" contain generic ontologies. They were designed adopting a minimalistic strategy: only some parts of some theories which are useful for the integration process are "bought".

**Languages in which the ontology is implemented:** Ontolingua, Loom.

**Important projects in which the ontology is used:** *Unknown*

**URL:** http://saussure.irmkant.rm.cnr.it/ON9/index.html

**References:**

- D. Pisanelli, A. Gangemi and G. Steve. WWW-available Conceptual Integration of Medical Terminologies: the ONIONS Experience. Proceedings of AMIA 97 Conference, 1997.
- A. Gangemi, D. Pisanelli and G. Steve (1999). An Overview of the ONIONS Project: Applying Ontologies to the Integration of Medical Terminologies. Data Knowledge Engineering, 31:183-220.

**Size:** number of concepts: Between 1000 & 5000; number of relations: Unknown; number of axioms: Unknown; number of instances: Unknown

## 1.4.5.5 Research ontologies

### 1) AIFB

The AIFB ontology models the domain of research topics and administrative tasks at the Institute AIFB. This ontology will form the basis to annotate documents in order to enable semantic access to these documents. The web presentation of the Institute AIFB (http://www.aifb.uni-karlsruhe.de) is based on this ontology.

**Languages in which the ontology is implemented:** FLogic

**Important projects in which the ontology is used:** *Unknown*

**URL:** http://ontobroker.semanticweb.org/ontos/aifb.html

**References:**

- A. Maedche, S. Staab, N. Stojanovic, R. Studer, Y. Sure. SEAL ? A Framework for Developing SEmantic PortALs. In: ACM K-Cap 2001 - First International Conference on Knowledge Capture, Oct. 21-23, 2001, Victoria, B.C., Canada.

**Size:** number of concepts: Between 50 & 100; number of relations: Between 100 & 500; number of axioms: Less than 10; number of instances: Between 500 & 1000

### 2) (KA)$^2$

KA(2) aims at "intelligent" knowledge retrieval from the Web and automatic derivation of "new" knowledge. In other words, it aims at knowledge-based reasoning on the Web, as opposed to the more usual information retrieval. Another objective of the initiative concerns a distributive ontological engineering process. Community web portals serve as portals for the information needs of particular communities on the web. We here discuss how a comprehensive and flexible strategy for building and maintaining a high-value community web portal has been conceived and implemented. The strategy includes collaborative information provisioning by the community members. We have also implemented a set of ontology-based tools that have facilitated the construction of our show case - the community web portal of the knowledge acquisition community

**Languages in which the ontology is implemented:** Flogic and OIL

**Important projects in which the ontology is used:** $(KA)^2$ project

**URL:** http://ontobroker.semanticweb.org/ontos/ka2.html

**References:**


- Stefan Decker, Michael Erdmann, Dieter Fensel, and Rudi Studer: Ontobroker: Ontology Based Access to Distributed and Semi-Structured Information. In R. Meersman et al. (eds.): Semantic Issues in Multimedia Systems. Proceedings of DS-8. Kluwer Academic Publisher, Boston, 1999, 351-369.
- Staab, S., Angele, J., Decker, S., Hotho, A., Maedche, A., Schnurr, H-P., Studer, S., Sure, Y.: AI for the Web --- Ontology-based Community Web Portals. In: AAAI 2000/IAAI 2000- Proceedings of the 17th National Conference on Artificial Intelligence and 12th Innovative Applications of Artificial Intelligence Conference, Austin/TX, USA, July 30-August 3, 2000, Menlo Park/CA, Cambridge/MA, AAAI Press/MIT Press.
- Staab, S., Angele, J., Decker, S., Hotho, A., Maedche, A., Schnurr, H-P., Studer, S., Sure, Y.: Semantic Community Web Portals.In: Computer Networks (Special Issue: WWW9- Proceedings of the 9th International World Wide Web Conference, Amsterdam, The Netherlands, May, 15-19, 2000), Elsevier.


**Size:** number of concepts: Unknown; number of relations: Unknown; number of axioms: Unknown; number of instances: Unknown

## *1.5 Repositories of ontologies*

Given that ontologies have to be reusable, the existence the libraries and repositories of ontologies is very important. Some of the best known repositories are:

- *DAML repository* (http://www.daml.org/ontologies/). It is a repository with more that 160 ontologies that can be freely uploaded by the users. Ontologies are implemented in DAML+OIL language. They are classified according to different criteria (URI's, submission date, keywords, etc.). There are known ontologies as Cyc, UNSPSC, etc. It is possible to browse using hyper-DAML, a hyper-link system. To reason with that ontologies……

- *Ontolingua Server repository* (www-ksl-svc.stanford.edu:5915/). It stores more than 50 ontologies in Ontolingua. It is possible to view the inclusion relations of ontologies in as a tree. In its turn, it is possible to view the statistics of each ontology (number of classes, relations, axioms, etc.). Each ontology can be directly seen, or can be seen through graphical HTML pages. You can find ontologies as the Frame Ontology, Chemical Elements, etc. To reason with that ontologies

- *Universal repository* (http://www.ist-universal.org/). The purpose of this repository is to enable collaboration among leading educators by providing exchange services for learning resources. This resources are classified by knowledge areas (history, computer science, languages and literature, etc.). There are more than 46 ontologies written in the same/different language.

- *SHOE repository* (http://www.cs.umd.edu/projects/plus/SHOE/onts/). It provides descriptions and links to existing SHOE ontologies. Currently, the ontologies reflect concepts that we have come across in applying SHOE to domains of our interest. Many of these ontologies are in a draft stage, and we welcome your feedback to help us improve them.

There are other public domain libraries that are accessible via Web, for example WebODE (http://babage.dia.fi.upm.es/webode/), WebONTO (http://kmi.open.ac.uk/projects/webonto/) or Ontosaurus (http://www.isi.edu/isd/ontosaurus.html) (see the section of tools). In these cases, an account must be asked for to the developers, so that users can browse and build their ontologies in them.

## *1.6 Overview of design criteria for building domain ontologies*

Here we summarise some design criteria and a set of principles that have been proved useful in the development of ontologies:

- *Clarity and Objectivity* [Gruber, 95], which means that the ontology should provide the meaning of defined terms by providing objective definitions and also natural language documentation.
- *Completeness* [Gruber, 95], which means that a definition expressed in terms of necessary and sufficient conditions is preferred over a partial definition (defined only through necessary or sufficient condition).
- *Coherence* [Gruber, 95], to permit inferences that are consistent with the definitions.
- *Maximum monotonic extendibility* [Gruber, 95]. It means that new general or specialised terms should be included in the ontology in a such way that is does not require the revision of existing definitions.
- *Minimal ontological commitments* [Gruber, 95]. Ontological commitments refer to agreement to use the shared vocabulary in a coherent and consistent manner. They guarantee consistency, but not completeness of an ontology, which means to make as few claims as possible about the world being modeled, giving the parties committed to the ontology freedom to specialize and instantiate the ontology as required.
- *Ontological Distinction Principle* [Borgo et al.; 96] which means that classes in an ontology should be disjoint.
- *Diversification of hierarchies* to increase the power provided by multiple inheritance mechanisms [Arpírez et al.; 98].
- *Modularity* [Bernaras et al.; 96] to minimize coupling between modules.
- *Minimization of the semantic distance between sibling concepts* [Arpírez et al.; 98] which means that similar concepts are grouped and represented using the same primitives.
- *Standardization of names* whenever is possible [Arpírez et al.; 98].

**References:**


- Arpírez, J. C.; Gómez-Pérez, A.; Lozano, A. Pinto, H. S. *(ONTO)$^2$Agent: An ontology-based WWW broker to select ontologies*. Workshop on Applications of Ontologies and Problem-Solving Methods. European Conference on Artificial Intelligence (ECAI'98). Brighton (United Kingdom). 1998. Pp. 16-24.
- Benjamins, V.R.; Fensel, D.; Gómez-Pérez, A.; Decker, S.; Erdmann, M.; Motta, E.; Musen, M. *The Knowledge Annotation Initiative of the Knowledge Acquisition Community (KA)$^2$*. In: Proceedings of the 11th Banff Knowledge Acquisition for Knowledge-Based System Workshop (KAW 98), Banff, Canada, April 1998.
- Borgo, S.; Guarino, N.; Masolo, C. *Stratified Ontologies: the case of physical objects.* In proceedings of the Workshop on Ontological Engineering. Held in conjunction with ECAI96. Pages 5-15. Budapest. 1996.
- Gruber, T. R. *Toward Principles of the Design of Ontologies Used for Knowledge Sharing*. International Journa of Human Computer Studies. 43. 1995. Pp. 907-928.

# 2 Overview of methodologies for building ontologies

This section shows different methodologies for building ontologies classified in the following way:

- Methodologies for building ontologies from the scratch.
- Methodologies for reenginering ontologies.
- Methodologies for cooperative construction of ontologies.
- Ontology learning methodologies.
- Ontology evaluation methods.

For each methodology, we will present: a general description, the URL where you can find information, the bibliographic references, the recommend life cycle, the general steps proposed by the methodology, the tools that provide it technological support, and significant ontologies that have been built following the methodology.

## *2.1 Methodologies for building ontologies from the scratch.*

A bunch of approaches have been reported to build ontologies. Already in 1990, Douglas Lenat and Guha published the general steps [Lenat el at.; 90] and some interesting points related to the Cyc development process. Some years later, in 1995, Uschold and King [Uschold et al.; 95] published the main steps followed on the development of the Enterprise Ontology, and Michael Grüninger and Fox [Grüninger et al.; 95] reported the methodology used for building the TOVE (TOronto Virtual Enterprise) ontology in the domain of enterprise modelling. One year later, Uschold and Grüninger proposed some methodological outlines for building ontologies [Uschold et al.; 96]. At the 12th European Conference for Artificial Intelligence (ECAI'96), Amaya Bernaras and her colleagues [Bernaras et al.; 96] presented a method used to build an ontology in the domain of electrical networks as part of the Esprit KACTUS project. Methontology [Fernández et al.; 96] appeared at the same time and was extended a few years later [Fernández-López et al.; 99], [Fernández-López et al.; 00]. In 1997, a methodology was proposed for building ontologies based on the SENSUS ontology [Swartout et al.; 97].

Although the methodologies shown in this section allow (of course) the reuse of ontologies, if needed, they do not consider the following situations: the use of a core ontology to add new terms; the reengineering of a former ontology; etc.

**References:**


- A. Bernaras, I. Laresgoiti, J. Corera. Building and Reusing Ontologies for Electrical Network Applications. Proceedings of the European Conference on Artificial Intelligence (ECAI'96). ECAI 96. (1996). 298-302.
- M. Fernández-López, A. Gómez-Pérez, A. Pazos-Sierra, J. Pazos-Sierra, Building a Chemical Ontology Using METHONTOLOGY and the Ontology Design Environment. IEEE Intelligent Systems & their applications. (January/February 1999) 37-46.M. Fernández-López, A. Gómez-Pérez, M.D. Rojas-Amaya, Ontologies' crossed life cycles. Workshop on Knowledge Acquisition, Modelling and Management (EKAW). (Springer Verlag, Jean Les Pins, France 2000). 65-79.
- M. Fernández, A. Gómez-Pérez, N. Juristo, METHONTOLOGY: From Ontological Art Towards Ontological Engineering. Symposium on Ontological Engineering of AAAI. (Stanford, California, March, 1997). 33-40.
- D.B. Lenat, R.V. Guha, Building large knowledge-based systems. (Addison-Wesley Publising Company, Inc. 1990).
- B. Swartout, P. Ramesh, K. Knight, T. Russ, Toward Distributed Use of Large-Scale Ontologies. Symposium on Ontological Engineering of AAAI. (Stanford, California, March, 1997). 138-148.
- M. Uschold, M. Grüninger, Ontologies: Principles Methods and Applications. Knowledge Engineering Review. Vol. 2. (1996). 93-155.
- M. Uschold, M. King, Towards a Methodology for Building Ontologies. Workshop on Basic Ontological Issues in Knowledge Sharing. (1995).

### 2.1.1 Cyc methodology

Cyc methodology consists of the following steps: first, you have to extract, by hand, common sense knowledge that is implicit in different sources. Next, once you have enough knowledge in your ontology, new common sense knowledge can be acquired either using natural language or machine learning tools.

**URL:** www.cyc.com

**References:**

- D.B. Lenat, R.V. Guha, Building large knowledge-based systems. (Addison-Wesley Publising Company, Inc. 1990).

**Recommended life cycle:** evolving prototypes

**Recomended steps to build the ontology:**

- Manually codify implicit and explicit knowledge appearing in different sources,
- Tools aided knowledge codification,
- Delegation on tools of the majority of the codification

**Tools that give support to the methodology:** Cyc environment

**Significant developed ontologies using the methodology:**

- Cyc
- cyc.daml

### 2.1.2 Uschold and King

This methodology proposes some general steps to develop ontologies, which are: (1) to identify the purpose; (2) to capture the concepts and the relationships between these concepts, and the terms used to refers to these concepts and relationships; (3) to codify the ontology. The ontology has to be documented and evaluated. Other ontologies can be used to build the new one.

**URL:** *Not available*

**References:**

- Uschold, M. King, M. 1995. "Towards a Methodology for Building Ontologies". Workshop on Basic Ontological Issues in Knowledge Sharing.
- Uschold, M.; Grüninger, M. 1996. "Ontologies: Principles Methods and Applications" Knowledge Engineering Review. Vol. 2.

**Recommended life cycle:** *Not available.*

**Recomended steps to build the ontology:**

- Identify purpose,
- Ontology capture,
- Coding,
- Integrating existing ontologies,
- Evaluation,
- Documentation

**Tools that give support to the methodology:** Ontolingua Server

**Significant developed ontologies using the methodology:**

- Enterprise Ontology

### 2.1.3 Grüninger and Fox

Grüninger and Fox propose a formalized method for building ontologies. First, they propose to identify intuitively the main scenarios (possible applications in which the ontology will be used). Later, a set of natural language questions, called competency questions, are used to determine the scope of the ontology, that is, the questions that could be answered using the ontology. These questions are used to extract the main concepts, their properties, relations and axioms, which are formally defined in Prolog. Therefore, this is a very formal methodology that takes advantage of the robustness of classic logic, and can be used as a guide to transform informal scenarios in computable models.

**URL:** *Not available*

**References:**

- M. Grüninger, M.S. Fox, Methodology for the design and evaluation of ontologies. Workshop on Basic Ontological Issues in Knowledge Sharing. (Montreal, Canada, 1995).
- M. Uschold, M. Grüninger, Ontologies: Principles Methods and Applications. Knowledge Engineering Review. Vol. 2. (1996). 93-155.

**Recommended life cycle:** None

**Recomended steps to build the ontology:**

- Motivating scenarios,
- Informal competency questions,
- Formal terminology,
- Formal competency questions,
- Formal axioms,
- Completeness theorems

**Tools that give support to the methodology:** *None.*

**Significant developed ontologies using the methodology:**

- Tove

### 2.1.4 KACTUS methodology

In the methodology proposed at the KACTUS project the ontology is built on the basis of an application knowledge base (KB), by means of a process of abstraction (that is, following a bottom-up strategy). The more applications are built, the more general the ontology becomes; hence, the further the ontology moves away from a KB. In other words, they propose to start building a KB for a specific application. Later, when a new knowledge base in a similar domain is needed, they propose to generalize the first KB into an ontology and adapt it for both applications. Applying this method recursively, the ontology would represent the consensual knowledge needed in all the applications.

**URL:** *Not available*

**References:**

- A. Bernaras, I. Laresgoiti, J. Corera. Building and Reusing Ontologies for Electrical Network Applications. Proceedings of the European Conference on Artificial Intelligence (ECAI?96). ECAI 96. (1996). 298-302.

**Recommended life cycle:** Evolved prototypes

**Recomended steps to build the ontology:**

- Specification of the application,
- Preliminary design based on relevant top-level ontological categories,
- Ontology refinement and structuring

**Tools that give support to the methodology:** *None.*

**Significant developed ontologies using the methodology:**

- Diagnose faults ontology

## 2.1.5 METHONTOLOGY

Methontology is a methodology for building ontologies either from scratch, reusing other ontologies as they are, or by a process of reengineering them. The Methontology framework enables the construction of ontologies at the knowledge level. It includes: identification of the ontology development process where the main activities are identified (evaluation, configuration management, conceptualization, integration, implementation, etc.); a life cycle based on evolving prototypes; and the methodology itself, which specifies the steps to be taken to perform each activity, the techniques used, the products to be output and how they are to be evaluated. Methontology is partially supported by the ontology development environment WebODE.

**URL:** *Not available*

**References:**


- Fernández-López, M.; Gómez Pérez, A.; Rojas Amaya, M. D. 2000. Ontologies crossed life cycles. Workshop on Knowledge Acquisition, Modelling and Management (EKAW). Editor Springer Verlag. Jean Les Pins (Francia). Pp. 65-79.
- Fernández-López, M.; Gómez-Pérez, A.; Pazos-Sierra, A.; Pazos-Sierra, J. 1999. Building a Chemical Ontology Using METHONTOLOGY and the Ontology Design Environment. IEEE Intelligent Systems & their applications. January/February PP. 37-46.
- Fernández, M.; Gómez-Pérez, A.; Juristo, N. 1997. METHONTOLOGY: From Ontological Art Towards Ontological Engineering. Symposium on Ontological Engineering of AAAI. Stanford (California).
- Gómez-Pérez, A; Rojas, M. D. 1999. "Ontological Reengineering and Reuse". European Knowledge Acquisition Workshop (EKAW). - Gómez-Pérez, A. 1998. ?Knowledge Sharing and Reuse?. In J. Liebowitz (Editor) Handbook of Expert Systems. CRC.


**Recommended life cycle:** evolved protoypes

**Recomended steps to build the ontology:** specification, conceptualisation, formalisation, implementation, maintenance

**Tools that give support to the methodology:** ODE, WebODE

**Significant developed ontologies using the methodology:**

- CHEMICALS
- Environmental pollutants
- Hardware and software ontology
- Monatomic ions ontology
- Reference-Ontology
- Silacate ontology
- Restructured version of the (KA)2 ontology

## 2.1.6 SENSUS methodology

The methodology based on Sensus is a top-down approach for deriving domain specific ontologies from huge ontologies. The authors propose to identify a set of ?seed? terms that are relevant to a particular domain. Such terms are linked manually to a broad-coverage ontology (in that case, the Sensus ontology, which contains more than 50,000 concepts). They select automatically the relevant terms for describing the domain and prune the Sensus ontology. Consequently, the algorithm delivers the hierarchically structured set of terms for describing a domain that can be used as a skeletal foundation for a KB.

**URL:** *Not available*

**References:**

- B. Swartout, P. Ramesh, K. Knight, T. Russ, Toward Distributed Use of Large-Scale Ontologies. Symposium on Ontological Engineering of AAAI. (Stanford, California, March, 1997). 138-148.

**Recommended life cycle:** None

**Recomended steps to build the ontology:**

- A series of terms are taken as seed,
- These seed terms are linked by hand to SENSUS,
- All the concepts in the path from the seed terms to the root of SENSUS are included,
- Terms that could be relevant within the domain and have not yet appeared are added,
- Finally, for those nodes that have a large number of paths through them, the entire subtree under the node is sometimes added

**Tools that give support to the methodology:** Ontosaurus

**Significant developed ontologies using the methodology:**

- SENSUS

### 2.1.7 On-To-Knowledge Methodology

The OTK project applies ontologies to electronically available information to improve the quality of knowledge management in large and distributed organizations. The methodology provides guidelines for introducing knowledge management concepts and tools into enterprises, helping knowledge providers and seekers to present knowledge efficiently and effectively. The methodology includes the identification of goals that should be achieved by knowledge management tools and is based on an analysis of usage scenarios and different roles knowledge workers and other stakeholders play in organizations. More specific, they present the architecture of the OTK tool suite, describe each of its tools, focus on application driven ontology development and finally describe employment and evaluation of the methodology within the OTK case studies.

**URL:** http://www.ontoknowledge.org/

**References:**

- S. Staab, H.-P. Schnurr, R. Studer, and Y. Sure: Knowledge Processes and Ontologies, IEEE Intelligent Systems, 16(1), January/February 2001.

**Recommended life cycle:** *Not available.*

**Recomended steps to build the ontology:**

- Feasibility Study,
- Kick off,
- Refinement,
- Evaluation,
- Maintenance & Evolution

**Tools that give support to the methodology:** OntoEdit

**Significant developed ontologies using the methodology:**

- AIFB
- ProPer

## *2.2 Methodologies for reengineering ontologies*

### 2.2.1 Method for reengineering ontologies of the Ontology Group of Artificial Intelligence Lab. at UPM

Ontological reengineering is the process of retrieving and mapping a conceptual model of an implemented ontology to another, more suitable conceptual model, which is re-implemented. The Ontology Group of Artificial Intelligence Lab. at UPM has presented a method for reengineering ontologies that adapts Chikofsky?s software reengineering schema to the ontology domain. Three main activities were identified: reverse engineering, restructuring and forward engineering.

**URL:** *Not available*

**References:**

- Gómez-Pérez, A; Rojas, M. D. 1999. "Ontological Reengineering and Reuse". European Knowledge Acquisition Workshop (EKAW).

**Recommended life cycle:** *Not available.*

**Recomended steps to build the ontology:**

- Reverse engineering,
- Restructuring,
- Forward engineering

**Tools that give support to the methodology:** *None.*

**Significant developed ontologies using the methodology:** Restructured version of Standard Units

## *2.3 Methodologies for cooperative construction of ontologies*

An ontology is a shared and common understanding of some domain. Righ now, emphasis is put on consensus on the content of ontologies, in the sense that a group of people agrees on the formal specification of the concepts, relations, attributes and axioms that the ontology provides. However, the problems of how to jointly construct an ontology (with a group a people) and how to distributively construct an ontology (with people at different locations) are still unsolved. Euzenat identified the following problems1 concerning collaborative construction of ontologies: management of the interaction and the communication between people; data access control; recognition of a moral right about the knowledge (attribution); detection and management of errors; and concurrent management and modification of the data. The consequence of these problems is that there are a few detailed proposals about how to collaboratively build ontologies. Nevertheless, since several years ago, the main proposal are: CO4, for collaborative construction of KB's at INRIA; and the approach used in ontologies building at the Knowledge Annotation Intiative of the Knowledge Acquisition Community, also know as (KA)2 initiative.

### 2.3.1 CO4 methodology

CO4 is a protocol to reach consensus between several KBs. Its goal is that people can discus and commit about the knowledge introduced in the KBs of the system. These KBs are built to be shared, and they have consensual knowledge, hence they can be considered ontologies. The experimentation has been done, above all, in the molecular genetic domain. According to Euzenat's proposal, the KBs are organised in a tree. The leaves are called user KBs, and the intermediate nodes, group KBs. On the one hand, the user KBs do not obligatorily have consensual knowledge. On the other hand, each group KB represents the consensual knowledge among its sons (called subscriber KBs). A KB can subscribe to only one group.

**URL:** *Not available*

**References:**

[1] (see http://www.inrialpes.fr/sherpa/papers/euzenat98c.html)

- Euzenat, J. 1996. "Corporative memory through cooperative creation of knowledge bases and hyper-documents". Proceedings 10th KAW, Banff (Canada).
- Euzenat, J. 1995. "Building consensual knowledge bases: context and architecture". In Mars, N. (Ed.). Building and sharing large knowledge bases. IOS Press. Amsterdam (Netherlands). Pages 143-155.

**Recommended life cycle:** *Not available.*

**Recomended steps to build the ontology:**

- Proposal,
- Call for comments,
- Replay,
- Decision

**Tools that give support to the methodology:** CO4 environment.

**Significant developed ontologies using the methodology:** *Unknown.*

## 2.3.2 $(KA)^2$ methodology

The goal of the Knowledge Annotation Initiative of the Knowledge Acquisition community, also acknowledged as the (KA)2 initiative, is to model the knowledge-acquisition community using ontologies developed in a joint effort by a group of people at different locations using the same templates and language. To easy the process of building the Research-Topic ontology, the ontology coordinating agent distributed a template among the Ontopic agents, which used e-mail in their intra-communication and also to send their results to the coordinating agents (experts in different topics). The ontology was generated from the knowledge introduced via the template. Once the ontology coordinating agents got all the portions of the ontologies from the ontopic agents, they integrated them. In this case, it was not difficult the integration process since all the ontopic agents used the same pattern.

**URL:** *Not available*

**References:**

- Stefan Decker, Michael Erdmann, Dieter Fensel, and Rudi Studer: Ontobroker: Ontology Based Access to Distributed and Semi-Structured Information. In R. Meersman et al. (eds.): Semantic Issues in Multimedia Systems. Proceedings of DS-8. Kluwer Academic Publisher, Boston, 1999, 351-369.

**Recommended life cycle:** *Not available.*

**Recomended steps to build the ontology:**

- Template distribution,
- Generation of ontologies,
- Integration

**Tools that give support to the methodology:** *None*

**Significant developed ontologies using the methodology:**

- Organisation ontology
- Project ontology
- Person ontology
- Publication ontology
- Event ontology
- Research-topic ontology
- Research-product ontology

## *2.4 Ontology learning ontologies*

### 2.4.1 Aussenac-Gille's and colleagues methodology

This is a method (also called "ontology design for texts") for knowledge modelling based on knowledge elicitation from technical documents. It benefits of the increasing amount of available electronic texts and of the maturity of natural language processing tools. The approach defines a framework where the knowledge engineer selects the appropriate tools, combines their use and interprets their results to build up a domain model. Its major statements are the following: (1) to start from texts to acquire knowledge; (2) to connect source texts to conceptual models; (3) to explore texts by applying natural language processing tools and techniques based on results in linguistics. This method is promoted by the French working group in Terminology and Artificial Intelligence.

**URL:** http://www.biomath.jussieu.fr/TIA/

**References:**

- Aussenac-Gilles, N., Biébow, B., Szulman, S. Corpus analysis for conceptual modelling. Workshop on Ontologies and Texts. Juan-Les-Pins. France. October 2000.
- N. Aussenac-Gilles, B. Biébow, S. Szulman. Revisiting Ontology Design : a methodology based on corpus analysis. Knowledge engineering and Knowledge Management: methods, models and tools. R. Dieng and O. Corby (Eds). Berlin : Springer, 2000. LNAI lecture notes in Computer Science vol. 1937. pp 172-188.

**Recommended life cycle:** *Not available.*

**Recomended steps to build the ontology:**

- Build up a corpus
- Linguistic Analysis
- Normalisation
- Formalisation

**Tools that give support to the methodology:** LEXTER, LINGUAE, TERMINAE.

**Significant developed ontologies using the methodology:** TH(IC)2.

### 2.4.2 Maedche and colleagues' methodology

The acquisition process starts with the selection of a generic core ontology, which has to be converted into the ontology model. Second, the user must specify which texts should be used in the following steps. The next step is to acquire domain-specific concepts from the available resources as the base ontology is (most likely) generic. Now the ontology contains domain-specific concepts, but still many generic concepts remain. Therefore the given ontology must be focused to the domain. This happens by removing all generic concepts from the ontology. The conceptual structure of the ontology is now established. Based on this structure the next step acquires non-taxonomic conceptual relations from texts. In addition to the relations provided by the base ontology that survived the focusing step (as their domain/range-concepts still exist) new conceptual relations are induced in the next step by applying learning methods to the selected texts.

**URL:** *Not available*

**References:**

- Kietz, J., Maedche, A., Volz, R. "A Method for Semi-Automatic Ontology Acquisition from a Corporate Intranet". Workshop on Ontologies and Texts. Juan-les-Pins. 2000.

**Recommended life cycle:** *Not available.*

**Recomended steps to build the ontology:**

- Select sources,
- Concept learning,

- Domain focusing,
- Relation learning,
- Evaluation

**Tools that give support to the methodology:** *Unknown*

**Significant developed ontologies using the methodology:** *Unknown.*

## *2.5 Ontology merge methodologies*

### 2.5.1 FCA-merge

The method is guided by application-specific instances of the given source ontologies that are to be merged. Techniques from natural language processing and formal concept analysis to derive a lattice of concepts as a structural result of FCA-MERGE are applied. The generated result is then explored and transformed into the merged ontology with human interaction.

**URL:** *Not available*

**References:**

- Stumme, G.; Maedche, A. "FCA-Merge: Bottom-Up Merging of Ontologies". *IJCAI '01 - Proceedings of the 17th International Joint Conference on Artificial IntelligenceK.* Seattle. USA. August, 1-6, 2001. San Francisco. California. Morgan Kaufmann

**Recommended life cycle:** *Not available.*

**Recomended steps to build the ontology:**

- Linguistic Analysis and Context Generation,
- Generating the Pruned Concept Lattice,
- Generating the new Ontology from the Concept Lattice

**Tools that give support to the methodology:** *Unknown*

**Significant developed ontologies using the methodology:** *Unknown*

### 2.5.2 PROMPT

PROMPT, an algorithm that provides a semi-automatic approach to ontology merging and alignment. PROMPTperforms some tasks automatically and guides the user in performing other tasks for which his intervention is required.PROMPT also determines possible inconsistencies in the state of the ontology, which result from the user?s actions, and suggests ways to remedy these inconsistencies. PROMPT is based on an extremely general knowledge model and therefore can be applied across various platforms.

**URL:** *Not available*

**References:**

- Noy, N.F. and Musen, M.A. (2000). "PROMPT: Algorithm and Tool for Automated Ontology Merging and Alignment". In: Proceedings of the Seventeenth National Conference on Artificial Intelligence (AAAI-2000), Austin, TX.

**Recommended life cycle:** *Not available.*

**Recomended steps to build the ontology:**

- Merge clases,
- Merge slots,
- Merge bindings between a slot and a class,
- Perform a deep copy of a class from one ontology to another,
- Perform a shallow copy of a class,

- Solve conflicts

**Tools that give support to the methodology:** Protege-2000

**Significant developed ontologies using the methodology:** *Unknown.*

## *2.6 Ontology evaluation methods*

### 2.6.1 Guarino's group methodology

This methodology for taxonomy evaluation uses philosophical principles based on the concepts of: rigidity, identity, unity and dependence. The user makes annotations in each property of the taxonomy, for example, you can specify if there is a criterion to "identify" every entity of the property, if the property has "dependency" on other property (for example, "director of organisation" depends on "organisation"), etc. Then, the user evaluates the ontology taking into account these annotations.

**URL:** *Not available*

**References:**

- Welty, C., Guarino, N. 2001. Supporting Ontological Analysis of Taxonomic Relationships. To appear in Data and Knowledge Engineering,September 2001.

**Recommended life cycle:** It is not applicable

**Recomended steps to build the ontology:**

- Put tags to every property assigning meta-properties,
- Focus just on the rigid properties,
- Evaluate the taxonomy taking into account the restrictions between meta-properties,
- Consider non-rigid properties,
- Complete the taxonomy with missing properties

**Tools that give support to the methodology:** *None.*

**Significant developed ontologies using the methodology:**

- Top level of particulars of Guarino's group
- Top level of universals of Guarino's group

### 2.6.2 Gómez Pérez's evaluation methodology

This method takes into account previous work done on evaluating ontologies and the criteria (consistency, completeness, conciseness, expandability and sensitiveness) used to evaluate and assess ontologies. It also addresses the possible types of errors made when domain knowledge is structured in taxonomies in an ontology and in knowledge bases: circularity errors, exhaustive and non-exhaustive class partition errors, redundancy errors, grammatical errors, semantic errors and incompleteness errors.

**URL:** *Not available*

**References:**

- Gómez Pérez, A. "Evaluating ontologies: Cases of Study". IEEE Intelligent Systems and their Applications. Special Issue on Verification and Validation of ontologies.
- Gómez Pérez, A. "Evaluation of Taxonomic Knowledge in Ontologies and Knowledge Bases". Twelfth Workshop on Knowledge Acquisition, Modeling and Management (KAW'99). Alberta (Canada). 1999.

**Recommended life cycle:** *Not available.*

**Recomended steps to build the ontology:**

- Evaluation of each individual definition and axiom,

- Evaluation of each collection of definitions and axioms that are stated explicitly in the ontology,
- Evaluation of the definitions that are imported from other ontologies,
- Evaluation of the definitions that can be inferred using other definitions and axioms

**Tools that give support to the methodology:** *Unknown*

**Significant developed ontologies using the methodology:** *Unknown.*

## *2.7 Conclusions*

As a summary, we can say that there are currently several methodologies with different approaches. For instance, if we compare the ontology development strategies that they propose the methodology used at the KACTUS project starts with a given knowledge base and by means of an abstraction process they build the ontology. In contrast, the opposite process was taken on the Sensus methodology, where specific domain ontologies are automatically generated from huge ontologies.

These methodologies can be also compared taking into account the degree of dependency of the ontology developed and its final application. In this sense, we can say that: (a) the methodology used at the KACTUS project is application dependent, because the ontology is built on the basis of a given application; (b) Grüninger and Fox, and Sensus methodologies are semi application-dependent; and (c) Cyc, Uschold and King methodology, and Methontology are application-independent, since the ontology development process is totally independent of the uses of the ontology.

According to the previous analysis and the work reported at [Fernández-López, 99], we can say that:

(a) None of these methodologies are fully mature if we compare them with software engineering and knowledge engineering methodologies. Thus, for example, according to table 1 (see at the end of the document, after URLs), there are key activities that are not proposed by most of the methodologies. However, we can say that Methontology is the maturest one and it has been recommended by FIPA for ontology construction.

(b) Proposals are not unified: currently, each group applies its own methodology. Consequently, great effort is required for creating a consensuated methodology for ontology construction. Collaboration between different groups to unify methodologies seems the most reasonable way to achieve it.

**References:**

- M. Fernández, Overview of Methodologies for Building Ontologies. Workshop on Ontologies and Problem-Solving Methods: Lessons Learned and Future Trends. (IJCAI99) (August, 1999). Pages 4-1 to 4-13.

# 3 Overview of ontology tools

## *3.1 Introduction*

In this section, we will provide an overview of the main ontology engineering tools that have been developed in the last years.

We will first review tools and environments that can be used for building ontologies. These tools usually provide a graphical user interface for building ontologies, which allows the ontologist to create ontologies without using directly a specific ontology specification language. We will focus on their capability of managing ontologies written in different languages and on collaborative working issues.

We will also review tools that have been created for merging and integrating ontologies, such as Chimaera, FCA-Merge and Protégé PROMPT.

In the context of the Semantic Web, some tools have been created during last years for the annotation of web resources in SHOE, RDF(S) or DAML+OIL. Some of these annotation tools are OntoAnnotate, OntoMat, SHOE Knowledge Annotator and UBOT DAML Annotation.

In this section, we will also take into account other supporting tools that are not used specifically for creating content, but for solving queries about ontologies (inference engines), such as Cerebra, OntoBroker, RDF Gateway (which uses RQL -RDF Query Language- as a query language) and SilRI.

Finally, we review some ontology-based text mining tools, which allow extracting ontologies either from structured, semi-structured or free text: ASIUM, CORPORUM-OntoBuilder, LTG Text Processing Workbench, and Text-To-Onto.

## *3.2 Environments for building ontologies*

In this section, we will try to provide a broad overview of some of the available tools and environments that can be used for the building of ontologies, either from scratch or reusing other existing ontologies. We will provide a brief description of each tool, presenting the group that has developed it, its main features and functionalities, its relationship with KR formalisms, etc. We will also provide its URL and bibliographic references (if they are available) for allowing readers to find more information about it. Finally, we have considered interesting to show which languages can be managed by the tool (either exporting or importing ontologies in the tool) and whether it allows or not the collaborative construction of ontologies (as the building of an ontology needs, in many cases, from the collaborative work between several knowledge engineers).

### 3.2.1 APECKS

APECKS (*Adaptive Presentation Environment for Collaborative Knowledge Structuring*) is an ontology server that supports collaboration by allowing individuals to create personal ontologies. These ontologies can be compared with others' to prompt discussion about the sources of their differences and similarities. Support for structured communication in this way should produce richer, more detailed ontologies, as well as design rationales for their structure.

**URL:** *Not available*

**References:**

- Tennison, J. and Shadbolt, N.R. APECKS: a Tool to Support Living Ontologies. KAW98. 1998.

**Exports language:** Not available.

**Imports language:** Not available.

**Collaborative construction:** Yes.

### *3.2.2* Apollo

Apollo is a user-friendly knowledge modelling application. Modelling is based around the basic primitives, such as classes, instances, functions, relations etc.. Internal model is build as a frame system according to the internal model of the OKBC protocol. Apollo does a full consistency check while editing. The application is not bound to any representation language and can be adapted to support different storage formats via I/O plugins. User interface has an open architecture ( view based) and allows for implementation of additional views of the knowledge base. The software is written in the Java programming language and is available for the download.

**URL:** http://apollo.open.ac.uk

**References:**

- Wolff A., Koss M., Zdrahal Z.: Knowledge Modelling of Hypertension Guidelines, *Medinfo 2001 Conference, London.* 2-5 September 2001. http://www.medinfo2001.org

**Imports language**: OCML.

**Exports language**: CLOS, OCML, Ontolingua.

**Collaborative description**: No.

### 3.2.3 CODE4

CODE4 (*Conceptually Oriented Description Environment*) is a general purpose knowledge management system intended for analysing, debugging, and delivering knowledge about some domain. It is designed to be easily adaptable to many applications. It provides different views (hierarchical, graphical, property matrix) to the same knowledge source.

The system features a frame-based representation with a number of inheritance and inferencing modes, a very flexible graphic user interface with various graphing facilities, a hypertext mode of browsing, the ability to specify functional computation like in a spreadsheet, an optional simple restricted English-like syntax, and document scanning and lexicon management facilities. Knowledge representation formalism used in CODE4 is called CODE4-KR. It is based on ideas adopted from frame-based inheritance systems, conceptual graphs, object-orientation and description logic systems.

**URL:** http://www.csi.uottawa.ca/~doug/CODE4.html

**References:**

- Skuce, D. A Unified System for Managing Conceptual Knowledge. International Journal of Human-Computer Studies (1995), 42: 413-451.

**Exports language:** CODE4-KR.

**Imports language:** CODE4-KR.

**Collaborative construction:** No.

### 3.2.4 CO4

CO4 (*Cooperative Construction of Consensual knowledge bases*) is a tool developed at INRIA Rhône-Alpes, which supports the methodology CO4 for the cooperative construction of ontologies. Ontologies are built into a hierarchy of knowledge bases, each one representing the consensual knowledge for the users that are subscribed to that group. At the bottom of the hierarchy, each user can have a number of knowledge bases which they can alter as they want. When a change is proposed for the group base, each of the users subscribed to it is notified of the proposal and may either accept or reject it. If all users accept the proposal, the group knowledge base is changed. Individual users can also choose not to accept any changes into their own individual knowledge bases, even if they are accepted on a group level.

**URL:** http://co4.inrialpes.fr/

**References:**

- C. Alemany, Étude et réalisation d'une interface d'édition de bases de connaissances au travers du World Wide Web, Mémoire CNAM, Grenoble (FR), 1998.
- Jérôme Euzenat, Corporate memory through cooperative creation of knowledge bases and hyper-documents, Proc. 10th KAW, Banff (CA), 1996.

**Exports language:** *Not available.*

**Imports language:** *Not available.*

**Collaborative construction:** Yes.

### 3.2.5 DUET (DAML UML Enhanced Tool)

DUET is a system that is still in a development phase. It provides a UML visualization and authoring environment for DAML. Core DAML concepts are being mapped into UML through a UML profile for DAML. DUET will have the capability to work with multiple ontologies simultaneously, and to interact with the OAB services to interactively build articulations between ontologies. Its intended users are database designers and systems engineers, many of who all ready have a good understanding of UML and object-oriented modeling, which they can leverage to apply DAML to their systems.

**URL:** http://grcinet.grci.com/maria/www/CodipSite/Tools/Tools.html

**References:** *Not available.*

**Exports language:** DAML+OIL, UML.

**Imports language:** DAML+OIL, UML.

**Collaborative construction:** Yes.

### 3.2.6 GKB-Editor

The GKB-Editor (*Generic Knowledge Base Editor*) is a tool for browsing and editing knowledge bases across multiple Frame Representation Systems (FRSs) in a uniform manner, using OKBC. It offers an intuitive, graph-based user interface, in which users can edit a knowledge base through direct manipulation and selectively display the region of a knowledge base that is currently of interest. GKB-Editor has been developed at Artificial Intelligence Center (AIC) in SRI (Stanford Research Institute) International.

GKB-Editor allows the user to view the knowledge base from various viewpoints and display only desired portions of the knowledge base at a time. Incremental browsing allows the user to control what is displayed, and the level of detail.

**URL:** http://www.ai.sri.com/~gkb/

**References:**

- GKB-Editor User Manual (version 2.1). Available at http://www.ai.sri.com/~gkb/user-man.html

**Exports language:** Ontolingua, Ocelot, Loom, Theo, SIPE. All of them through OKBC.

**Imports language:** Ontolingua, Ocelot, Loom, Theo, SIPE. All of them through OKBC.

**Collaborative construction:** Yes.

### 3.2.7 IKARUS

IKARUS (*Intelligent Knowledge Acquisition and Retrieval Universal System*) is a web-based successor for CODE4 knowledge management environment, exploiting the cooperative capabilities of the web. It has been developed by the LAKE group at the University of Ottawa.

IKARUS uses a frame-based hierarchical representation, which allows multiple inheritance. Frames store knowledge about the *subject* in form of one or more *statements*. Statements contain the information about subjects either as *predicates* with well-defined syntax and semantics, or as unstructured fragments of information (such as text, URLs to web documents etc.). Hierarchies of frames are represented graphically, using a flexible coloring scheme to ease viewing and understanding complex internode relations.

**URL:** http://www.csi.uottawa.ca/~kavanagh/Ikarus/IkarusInfo.html

**References:**

- IKARUS: Intelligent knowledge acquisition and retrieval universal system. D. Skuce. October, 1996. Avialable at http://www.csi.uottawa.ca/~kavanagh/Ikarus/Ikarus4.html

**Exports language:** *Not available.*

**Imports language:** *Not available.*

**Collaborative construction:** Yes.

### 3.2.8 JOE (Java Ontology Editor)

JOE (Java Ontology Editor) is an ontology construction and viewing tool developed in Center for Information Technology, University of South Carolina. The basic idea behind JOE is to provide a knowledge management tool that supports multiple simultaneous users and distributed, heterogeneous operating environments. Ontologies are represented using a frame-based approach and can be viewed in three different formats: as an ER diagram, as a hierarchy similar to Microsoft Windows file manager, or as a graphical tree structure. JOE also allows performing queries, which are formulated by using a visual representation of the ontology and a point-and-click approach of adding query conditions.

**URL:** http://www.engr.sc.edu/research/CIT/demos/java/joe/

**References:** *Not available.*

**Exports language:** *Not available.*

**Imports language:** *Not available.*

**Collaborative construction:** Yes.

### 3.2.9 OilEd

OilEd has been developed in the context of the european IST OntoKnowledge project, by the University of Manchester, the Free University of Amsterdam an Interprice GmbH. It is a simple freeware ontology editor which allows the user to build ontologies using OIL, and it is not intended as a full ontology development environment. Consistency checking and automatic classification of the ontologies written with it can be performed using FaCT [Horrocks et al, 99]. Future development will concentrate on the support of the DAML+OIL format.

**URL:** http://img.cs.man.ac.uk/oil/

**References:**

- Horrocks, U. Sattler, S. Tobies. Practical reasoning for expressive description logics. 6th International Conference on Logic for Programming and Automated Reasoning (LPAR'99) (LNAI, Springer-Verlag, 1999). 161-180.

**Exports language:** RDF(S), OIL, DAML+OIL.

**Imports language:** RDF(S), OIL.

**Collaborative construction:** No.

### 3.2.10 OntoEdit

OntoEdit is an ontology engineering environment developed at the Knowledge Management Group (AIFB) of Karlsruhe University. It is a standalone application that provides a graphical ontology editing environment (which enables inspecting, browsing, codifying and modifying ontologies and supports in this way the ontology development and maintenance task) and an extensible architecture for adding new plugins. The conceptual model of an ontology is internally stored using a powerful ontology model, which can be mapped onto different, concrete representation languages. Ontologies are stored in relational databases (in its commercial version) and can be implemented in XML, FLogic, RDF(S) and DAML+OIL.

**URL:** http://ontoserver.aifb.uni-karlsruhe.de/ontoedit/

**References:**

- Maedche, A., Schnurr, H.-P., Staab, S. and Studer, R.: Representation Language-Neutral Modeling of Ontologies. In: Frank (ed.), Proceedings of the German Workshop "Modellierung" 2000. Koblenz, Germany, April, 5-7, 2000.

**Exports language:** XML, FLogic, RDF(S), DAML+OIL.

**Imports language:** XML, FLogic, DAML+OIL.

**Collaborative construction:** No.

### 3.2.11 Ontolingua

Ontolingua Server is a set of tools and services that support the building of shared ontologies between distributed groups, and has been developed by the Knowledge Systems Laboratory (KSL) at Stanford University. The ontology server architecture provides access to a library of ontologies, translators to languages (Prolog, CORBA IDL, CLIPS, Loom, etc.) and an editor to create and browse ontologies. Remote editors can browse and edit ontologies, and remote or local applications can access any of the ontologies in the ontology library using the OKBC (Open Knowledge Based Connectivity) protocol

**URL:** http://www-ksl-svc.stanford.edu:5915/

**References:**


- Farquhar, R. Fikes, J. Rice, The Ontolingua Server: A Tool for Collaborative Ontology Construction, Proceedings of the 10th Knowledge Acquisition for Knowledge-Based Systems Workshop, (Banff, Alberta, Canada 1996) 44.1-44.19.


**Exports language:** Loom, Ontolingua, Prolog, CLIPS.

**Imports language:** Ontolingua.

**Collaborative construction:** Yes.

### 3.2.12 Ontological Constraints Manager (OCM)

OCM is a tool that has been mainly developed for performing consistency checking of ontologies with respect to ontological axioms - making it possible to reason about the correctness of an application with respect to ontological constraints. It is composed of two editing tools which provide built-in checks for conflict and subsumption occurrence. OCM has been applied to diverse areas, each of which demonstrates a different use: deploying ontological axioms in business process modelling and enriching the axiomatisation of an Air Campaign Planning(ACP)-based application; identification of ontological constraints in ecological modelling; and evaluation of ontologies on system dynamics theory at the application level.

**URL:** http://www.ecs.soton.ac.uk/~yk1/rp956.ps

**References:**


- Kalfoglou,Y. Deploying Ontologies in Software Design. PhD Thesis. Department of Artificial Intelligence, University of Edinburgh. June, 2000.


**Exports language:** Ontolingua, Prolog.

**Imports language:** PIF.

**Collaborative construction:** Unknown.

### 3.2.13 Ontology Editor by Steffen Schulze-Kremer

This ontology editor uses an object-oriented terminology, instead of a frame-based terminology. It allows defining concepts and their instances, and data can be exported in either Prolog clauses or plain ascii text files. It provides a graphical user interface for building hierarchical trees of concepts, which can either use the is-a relationship or the part-of relationship.

**URL:** http://igd.rz-berlin.mpg.de/~www/prolog/oe.html

**References:** *Not available.*

**Exports language:** Prolog.

**Imports language:** *Not available.*

**Collaborative construction:** No.

### 3.2.14 OntoSaurus

Ontosaurus has been developed by the Information Sciences Institute (ISI) at the University of South California. It consists of two modules: an ontology server, which uses Loom as its knowledge representation system, and an ontology browser server that dynamically crates html pages (including image and textual documentation) that displays the ontology hierarchy. The ontology can be edited by html forms, and translators exist from LOOM to Ontolingua, KIF, KRSS and C++.

**URL:** http://www.isi.edu/isd/ontosaurus.html

**References:**

- B. Swartout, P. Ramesh, K. Knight, T. Russ, Toward Distributed Use of Large-Scale Ontologies. Symposium on Ontological Engineering of AAAI. (Stanford, California, March, 1997).

**Exports language:** LOOM, Ontolingua, KIF, KRSS, C++.

**Imports language:** LOOM.

**Collaborative construction:** No.

### 3.2.15 Protégé-2000

Protégé-2000 is a graphical and interactive ontology-design and knowledge-acquisition environment that is being developed by the Stanford Medical Informatics group (SMI) at Stanford University. It is an open source, standalone application that provides a graphical ontology editing environment and an extensible architecture for the creation of customized knowledge-based tools. Its knowledge model is OKBC-compatible.

Its component-based architecture enables system builders to add new functionality by creating appropriate plug-ins. The Protégé plug-in library contains plug-ins for graphical visualization of knowledge bases, inference-engine for verification of constraints in first-order logic, acquisition of information from remote sources such as UMLS and WordNet, semi-automatic ontology merging, etc. It also provides translators to FLogic, OIL, Ontolingua and RDF(S), and can store ontologies in any JDBC-compatible relational database. Plugins, applications and ontologies, which have been developed both by the Protege group and other Protege users, are available in the Protege Contributions Library.

**URL:** http://protege.stanford.edu

**References:**

- N. F. Noy, M. Sintek, S. Decker, M. Crubezy, R. W. Fergerson, & M. A. Musen. Creating Semantic Web Contents with Protege-2000. IEEE Intelligent Systems 16(2):60-71, 2001.
- N. F. Noy, R. W. Fergerson, & M. A. Musen. The knowledge model of Protege-2000: Combining interoperability and flexibility. 12th International Conference on Knowledge Engineering and Knowledge Management (EKAW\'2000), Juan-les-Pins, France, 2000.
- M. A. Musen, R. W. Fergerson, W. E. Grosso, N. F. Noy, M. Crubezy, & J. H. Gennari. Component-Based Support for Building Knowledge-Acquisition Systems. Conference on Intelligent Information Processing (IIP 2000) of the International Federation for Information Processing World Computer Congress WCC 2000), Beijing, 2000.
- Grosso, W., Gennari, J.H., Fergerson, R. and Musen, M.A. (1998). When Knowledge Models Collide (How it Happens and What to Do). In: Proceedings of the Eleventh Banff Knowledge Acquisition for Knowledge-Bases Systems Workshop, Banff, Canada.

**Exports language:** FLogic, OIL, Ontolingua, RDF(S).

**Imports language:** FLogic, OIL, Ontolingua, RDF(S), XML.

**Collaborative construction:** No.

### 3.2.16 VOID

VOID, the KACTUS toolkit, is an interactive environment for browsing, editing and managing (libraries of) ontologies. It provides an environment in which one can experiment with theoretical issues (e.g. organisation of libraries of ontologies, translating between different ontology formalisms -CML, EXPRESS and Ontolingua-, performing translations between them, etc.) and also perform practical work (e.g. browse, edit and query ontologies in various formalisms). Finally, VOID also provides an API (in

C/C++ and Prolog) for accessing its ontologies, which has been used in the development of applications in the KACTUS ESPRIT Project 8145.

**URL:** http://www.swi.psy.uva.nl/projects/Kactus/toolkit/about.html

**References:**

- Schreiber, A.; Terpstra, P. Sisyphus-VT: A CommonKADS solution. Technical Report, ESPRIT Project 8145 KACTUS, University of Amsterdam, 1995.
- Schreiber, A.; Wielinga, B.; Jansweijer, W. The KACTUS view on the 'O' word. Technical Report, ESPRIT Project 8145 KACTUS, University of Amsterdam, 1995.

**Exports language:** CML, EXPRESS, Ontolingua.

**Imports language:** CML, EXPRESS, Ontolingua.

**Collaborative construction:** Yes.

### 3.2.17 WebODE

WebODE is an ontological engineering workbench that has been developed in the Artificial Intelligence Lab from the Technical University of Madrid (UPM). It provides varied ontology related services, and covers and gives support to most of the activities involved in the ontology development process. It has a very expressive knowledge model. It provides an API for ontology access, which makes it easy the integration with other systems, and translators that import and export ontologies from and to markup (DAML+OIL, OIL, RDF(S), XML) and other traditional ontology languages. It also provides a graphical user interface for creating axioms and an inference engine that uses Prolog and an OKBC-compliant knowledge model of the ontologies.

**URL:** http://delicias.dia.fi.upm.es/webODE/index.html

**References:**

- WebODE: a Scalable Workbench for Ontological Engineering. Arpírez, J.C.; Corcho, O.; Fernández-López, M.; Gómez-Pérez, A. KCAP01. Victoria. Canada. October, 2001.

**Exports language:** CARIN, DAML+OIL, FLogic, OIL, Prolog, RDF(S), XML.

**Imports language:** CARIN, RDF(S), XML.

**Collaborative construction:** Yes.

### 3.2.18 WebOnto

WebOnto is a tool developed by the Knowledge Media Institute (KMI) of the Open University (England). It supports the collaborative browsing, creation and editing of ontologies, which are represented in the knowledge modelling language OCML.

Its main features are: management of ontologies using a graphical interface; support for PSMs and tasks modelling; inspection of elements, taking into account the inheritance of properties and consistency checking; a full tell&ask interface, and support for collaborative work, by means of broadcast/receive and making annotations (using Tadzebao). The WebOnto server is a freely available service provided to the ontology engineering community. A library with over 100 ontologies is accessible through WebOnto and can be browsed with no restrictions on access. Users who wish to use WebOnto for ontology development can get an account and password by emailing Dr John Domingue (j.b.domingue@open.ac.uk).

**URL:** http://webonto.open.ac.uk

**References:**

- J. Domingue, Tadzebao and Webonto: Discussing, Browsing and Editing Ontologies on the Web. In Proceedings of the Eleventh Knowledge Acquisition Workshop (KAW98, Banff, 1998).

**Exports language:** OCML, Ontolingua.

**Imports language:** OCML.

**Collaborative construction:** Yes.

## *3.3 Ontology merging and integration tools*

### 3.3.1 Chimaera

Chimaera is a tool primarily intended for merging knowledge base (KB) fragments, which also supports users in creating and maintaining distributed ontologies on the web. Two major functions it supports are merging multiple ontologies together and diagnosing individual or multiple ontologies. It supports users in reorganizing taxonomies, resolving name conflicts, browsing ontologies, editing terms, etc.

The process of KB merging typically involves activities as resolving name conflicts and aligning the taxonomy. This tool has special support for finding name conflicts and for pointing out interesting places in the merged taxonomy.

**URL:** http://www.ksl.stanford.edu/software/chimaera/

**References:**


- McGuinness, D.; Fikes, R.; Rice, J.; Wilder, S. An Environment for Merging and Testing Large Ontologies. In the Proceedings of the Seventh International Conference on Principles of Knowledge Representation and Reasoning (KR2000), Breckenridge, Colorado, USA. April 12-15, 2000.
- McGuinness, D.; Fikes, R.; Rice, J.; Wilder, S. The Chimaera Ontology Environment. In the Proceedings of the The Seventeenth National Conference on Artificial Intelligence (AAAI 2000), Austin, Texas, July 30 - August 3, 2000.


### 3.3.2 FCA-Merge Tool

The FCA-Merge tool supports the FCA-Merge method for the merging of ontologies, which is a bottom-up technique for merging ontologies based on a set of documents. This technique consists of three steps: the linguistic analysis of the texts which returns two formal contexts, the merging of the two contexts and the computation of the prouned concept lattice, and the semi-atuomatic ontology creation phase which supports the user in modelling the target ontology.

**URL:** *Not available*.

**References:**


- Stumme, G. and Maedche, A. Ontology Merging for Federated Ontologies on the Semantic Web using FCA-Merge.IJCAI'01 Workshop on Ontologies and Information Sharing, Seattle, USA. August, 2001.


### 3.3.3 PROMPT

PROMPT is an interactive ontology-merging tool that guides the user through the merging process making suggestions, determining conflicts, and proposing conflict-resolution strategies. The initial suggestions are based on the linguistic similarity of the frame names. After the user selects an operation to perform, PROMPT determines the conflicts in the merged ontology that the operation have caused and proposes possible solutions to the conflict. It then considers the structure of the ontology around the arguments to the latest operations—relations among the arguments and other concepts in the ontology—and proposes other operations that the user should perform.

**URL:** http://protege.stanford.edu/plugins/prompt/prompt.html

**References:**


- Anchor-PROMPT: Using Non-Local Context for Semantic Matching by N. F. Noy & M. A. Musen. Proceedings of the Workshop on Ontologies and Information Sharing at the Seventeenth International Joint Conference on Artificial Intelligence (IJCAI-2001), Seattle, WA, August 2001.
- N. F. Noy & M. A. Musen. PROMPT: Algorithm and Tool for Automated Ontology Merging and Alignment *Proceedings of the Seventeenth National Conference on Artificial Intelligence (AAAI-2000),* Austin, TX. 2000.
- N. F. Noy & M. A. Musen. SMART: Automated Support for Ontology Merging and Alignment. Twelfth Banff Workshop on Knowledge Acquisition, Modeling, and Management, Banff, Alberta, Canada, 1999.

- N. F. Noy & M. A. Musen. An Algorithm for Merging and Aligning Ontologies: Automation and Tool Support. Sixteenth National Conference on Artificial Intelligence (AAAI-99), Workshop on Ontology Management, Orlando, FL, 1999.

## *3.4 Ontology-based annotation tools*

These annotation tools have arisen recently, in the context of the Semantic Web. Their main objective is the creation and maintenance of ontology-based markups in static web documents. In fact, they are used for managing easily instances, attributes and relationships between web resources, either implemented in SHOE, RDF(S) or DAML+OIL. We will review the following annotation tools:

### 3.4.1 OntoMarkup Annotation Tool

The OntoMarkup annotation tool is an ontology-based tool which allows the annotator to incorporate semantic information in documents. The semantic annotation tool contains an ontology-based markup component which allows the user to browse and mark-up relevant pieces of information, a learning component which learns rules from examples and an information extraction component which extracts the objects and relations between these objects.

**URL:** http://kmi.open.ac.uk/projects/akt/

**References:**

- Knowledge Extraction by using an Ontology-based Annotation Tool, by Maria Vargas-Vera, Enrico Motta, John Domingue, Simon Buckingham Shum and Mattia Lanzoni, In proceedings of the Workshop Knowledge Markup and Semantic Annotation, K-CAP 2001, Victoria Canada, October 2001.

### 3.4.2 OntoMat

Ontomat is a user-friendly interactive webpage DAML+OIL annotation tool. It includes an ontology browser for the exploration of the ontology and instances and a HTML browser that will display the annotated parts of the text. It is Java-based and provides a plugin interface for extensions. The intended user is the individual annotator i.e., people that want to enrich their web pages with DAML-meta data. Instead of manually annotating the page with a text editor, say, emacs, Ont-O-Mat allows the annotator to highlight relevant parts of the web page and create new instances via drag'n'drop interactions. It supports the meta-data creation phase of the lifecycle.

**URL:** http://ontobroker.semanticweb.org/annotation/ontomat/index.html

**References:**

- S. Handschuh, S. Staab, A. Mädche. CREAM — Creating relational metadata with a component-based, ontology-driven annotation framework. K-CAP 2001. Victoria. Canada. October, 2001.

**Annotation language:** DAML+OIL.

### 3.4.3 OntoAnnotate

OntoAnnotate is a semi-automatic annotation tool that enables collecting knowledge from documents and web pages, creating a document base including metadata and enriching web resources or intranets with metadata. It allows annotating not only static HTML documents, but also Word and Excel documents.

**URL:** http://www.ontoprise.de/com/co_produ_tool2.htm

**References:**

- S. Handschuh, S. Staab, A. Mädche. CREAM — Creating relational metadata with a component-based, ontology-driven annotation framework. K-CAP 2001. Victoria. Canada. October, 2001.

**Annotation language:** RDF, FLogic.

### 3.4.4 SHOE Knowledge Annotator

The SHOE Knowledge Annotator is a Java program that allows users to mark-up web pages with SHOE knowledge without having to worry about the HTML-like codes. The Annotator is available as an applet or a stand-alone Java application.

**URL:** http://www.cs.umd.edu/projects/plus/SHOE/KnowledgeAnnotator.html

**References:**

- A Portrait of the Semantic Web in Action, by Jeff Heflin and James Hendler. IEEE Intelligent Systems, 16(2), 2001.
- Ontology-based Web Agents, by Sean Luke, Lee Spector, David Rager, and James Hendler. In Proceedings of First International Conference on Autonomous Agents 1997, AA-97..

**Annotation language:** SHOE.

### 3.4.5 UBOT DAML Annotation

The UBOT DAML Annotation tool aims at the automatic generation of DAML annotation from unconstrained real-world text documents. It uses AeroText, which parses natural language text and extracts interesting items. It has a sophisticated graphical user interface for customizing the default extraction rules.

**URL:** http://ubot.lockheedmartin.com/ubot/

**References:** *Not available.*

**Annotation language:** DAML+OIL.

## *3.5 Ontology learning tools*

These tools are used to learn ontologies from natural language, exploiting the interacting constraints on the various language levels (from morphology to pragmatics and background knowledge) in order to discover new concepts and stipulate relationships between concepts. For each tool, we provide a brief description, its URL and bibliographic references, and, if available, the language that it is able to tackle.

### 3.5.1 ASIUM

Asium is an acronym for "Acquisition of SemantIc knowledge Using Machine learning method". The main aim of Asium is to help the expert in the acquisition of semantic knowledge from texts and to generalize the knowledge of the corpus. Asium provides the expert with a powerful and user-friendly interface which will first help him or her to explore the texts and then to learn knowledge which are not in the texts. During the learning step, Asium helps the expert to acquire semantic knowledge from the texts, like subcategorization frames and an ontology. The ontology represents an acyclic graph of the concepts of the studied domain. The subcategorization frames represent the use of the verbs in these texts.

**URL:** http://www.lri.fr/~faure/Demonstration.UK/Presentation_Demo.html

**References:**

- D. Faure and C. Nedellec. *Knowledge acquisition of predicate argument structures from technical texts using machine learning: The system* ***asium***. In Knowledge Acquisition, Modeling, and Management, Proc. of the 11th European Workshop, EKAW'99, pages 329--334, Dagstuhl Castle, Allemagne, May 1999. Springer-Verlag, LNAI 1621.
- D. Faure and C. N'edellec. ***ASIUM****: Learning subcategorization frames and restrictions of selection*. In Y. Kodratoff, editor, 10th Conference on Machine Learning (ECML 98) -- Workshop on Text Mining, Chemnitz, Germany, Avril 1998.

**Supported languages:** French.

### 3.5.2 CORPORUM-OntoBuilder

CORPORUM-OntoBuilder is composed of two main modules, that can tackle with:

- Structured documents: Ontowrapper, screen-scraping and business rules to extract information from known places on specific sites (e.g. names, email addresses, telephone numbers).

- Unstructured documents: OntoExtract, extracting initial ontologies/taxonomies from natural language on web pages. OntoExtract is able to (through semantic analysis of the content of web pages): provide initial ontologies/taxonomies, refine existing ontologies (incl. more concepts), find relations between key terms in documents, thereby able to relate business areas to each other or allow for new associations,and find instances of concepts within documents. Ontologies are created in RDF(S).

**URL:** http://ontoserver.cognit.no

**References:**

- Information can be obtained from http://www.ontoknowledge.org/ Deliverables 5, 6 and 7.

**Supported languages:** English.

## 3.5.3 LTG Text Processing Workbench

The Language Technology Group Text Processing Workbench is a set of computational tools with (n)SGML pipeline data flow for uncovering internal structure in natural language texts. The main idea behind the workbench is the independence of the text representation and text analysis phase. At the representation phase the text is converted from a sequence of characters to features of interest by means of the annotation tools. At the analysis phases those features are used by statistics gathering and inference tools for finding significant correlations in the texts. The analysis tools are independent from a particular assumption on the nature of the feature-set and work on the abstract level of feature-elements which are represented as SGML items. The workbench is being used both for lexicographic purposes and for statistical language modelling. It supports an incremental process of corpus analysis starting from a rough automatic extraction and organization of lexico-semantic regularities and ending with a computer supported analysis of extracted data and a semi-automatic refinement of obtained hypotheses. For doing this the workbench employs methods from computational linguistics, information retrieval and knowledge engineering.

**URL:** http://www.ltg.ed.ac.uk/%7Emikheev/workbench.html

**References:**

- Mikheev,A and S.Finch . A Workbench for Finding Structure in Texts. *Proceedings of ANLP-97 (Washington D.C.).* ACL March 1997. pp. 8
- A.Mikheev and S.Finch. A Workbench for Acquisition of Ontological Knowledge from Natural Text. *Proceedings of the 7th conference of the European Chapter for Computational Linguistics (EACL'95).* Dublin, Ireland. 1995. pp. 194-201.

**Supported languages:** English.

## 3.5.4 Text-To-Onto

The Text-To-Onto system provides an integrated environment for the task of learning ontologies learning from text. These domain texts may be both natural language texts and HTML formatted texts. For a meaningful text analysis, textual preprocessing has to be performed.

The system is composed by a Text Management module, which enables selecting a relevant corpus and serves as an interface to the Information Extraction Server. If there already exists a domain lexicon, it performs domain specific parsing. The results of the parsing process are stored in XML or feature-value structures. The Management Module offers all existing learning components to the user. Typically these components are parametrizable. Existing knowledge structures (for example a taxonomy of concepts in FLogic) are incorporated as background knowledge. The learning component discovers on the base of the domain texts new knowledge structures, which are grabbed in the Ontology Modeling Module to expand the existing ontology.

**URL:** http://ontoserver.aifb.uni-karlsruhe.de/texttoonto/

**References:**

- Maedche, A. and Staab, S.: Semi-automatic Engineering of Ontologies from Text. In: Proceedings of the Twelfth International Conference on Software Engineering and Knowledge Engineering (SEKE'2000), Chicago, 2000.

- Maedche, A., Staab, S. and Studer, R. Ontology-based Information Extraction and Integration. Workshop on Language Technologies in Information and Knowledge Management. Proceedings of the 7th Conference on Computational Linguistcs of the German Society for Language Technologies. Saarbrücken, Germany, October 7-8, 1999.
- A. Maedche and R. Volz: The Text-To-Onto Ontology Extraction and Maintenance Environment. to appear: Proceedings of the ICDM Workshop on Integrating Data Mining and Knowledge Management, San Jose, California, USA, November 2001.


**Supported languages:** German.

## *3.6 Conclusions*

In this section, we have presented tools that can be used for building ontologies, for ontology merging and integration, for managing ontological annotations and for extracting ontologies from texts, which are some of the most important areas of ontological engineering in which technological support tools are needed.

There are some important parameters that can be used in the comparison and evaluation of existing tools. However, we have not considered all of them in the description of this section because we preferred to provide a broad overview of existing technology that could be used in ontological engineering. They will be included in later versions of this document. Some of these parameters are:

- **Software architecture and tool evolution**. This includes information about hardware and software platforms necessary to use the tool, its architecture (standalone applications, n-tier applications, etc.), extensibility, ontology storage, backup management, etc.

  In this sense, tools are moving towards Java-based applications, most of them in n-tier architectures and accessible in the web. Storage in databases is still a weak point of ontology tools, since just few of them use databases for storing ontologies (OntoEdit, Protégé2000 and WebODE). Backup management, extensibility, etc., are functionalities that are not commonly provided in these tools.

- **Interoperability**. The tools that have been presented are diverse and have been developed by different research groups in the world. As a consequence, there is not a standardization on the tools that are used when performing any of these tasks, and these environments are not usually interoperable: they use different internal representations and interfaces are not clear enough to achieve interoperability. However, some of the newest tools allow exporting and importing ontologies in XML and other markup languages as a means for exchanging ontologies between them.

- **Expressiveness of the underlying knowledge model**. Differences between the underlying knowledge models of different tools are important, because they will drive the modeling/annotation/learning of ontologies in the tool. Among these differences, we can identify differences in the underlying KR paradigm in which ontologies are internally represented, differences in the components that can be modeled (concepts, binary or n-ary relations, metaclasses, axioms, etc.), etc. Most of these tools are attached to an ontology specification language (for instance, OilEd is attached to OIL, OntoSaurus is attached to LOOM, WebOnto is attached to OCML, etc.), and some other tools, such as Protégé2000, OntoEdit and WebODE, allow modeling ontologies without taking into account the language in which they will be implemented. **Inference services** It is important to know the inference services provided by an ontology tool. Some important aspects are the presence or absence of constraint and consistency checking mechanisms, the type of inheritance of properties that is applied in hierarchies of concepts and relations, the ability to perform automatic classifications, and exception handling. **Methodology support**. It is not usual that a tool gives support to a methodology for building ontologies, except for WebODE, which gives support to Methontology. This is not the case with tools for merging/integration and tools for ontology learning, which are used as a technological support for methods, methodologies or techniques, as presented in the corresponding section. Related to annotation tools, there is not any methodology for annotation of web resources. Hence, these tools do not provide support for any methodology. **Ontology library**. Ontologies are developed with the aim of being reusable pieces of knowledge. It is important to have libraries of ontologies related to a tool, so that ontologists can reuse them easily, either partially or completely. **Cooperative and collaborative construction of ontologies**. We have included this parameter in the description of ontology building tools provided in this section, but many other related parameters could be included in our study: workgroup management, version control of ontologies, tracking of changes, etc.**Usability aspects** The last group of parameters is related to usability aspects of the tools. This is related to the help system, edition and visualization, etc. From our experience, we think that usability must be improved in most of the tools, so that non-experts in ontology engineering are able to use them easily.

## *References*

The main references for this section have been already provided in the template for the definition of each language.

# 4 Overview of Languages for Building Ontologies

## *4.1 Introduction*

A great range of languages have been used for the specification of ontologies during the last decade: Ontolingua, LOOM, OCML, FLogic, CARIN, etc. Many of these languages had been already used for representing knowledge inside knowledge-based applications, other ones were adapted from existing knowledge representation languages, and there is also a group of languages that were specifically created for the representation of ontologies. These languages (which we will call "traditional" languages) are in a stable phase of development, and their syntax consists of plain text where ontologies are specified (many of them have a Lisp-like syntax).

Recently, many other languages have been developed in the context of the World Wide Web (and have had great impact in the development of the Semantic Web): RDF, RDF Schema, SHOE, XOL, OML, OIL and DAML+OIL. Their syntax is based on XML, which has been widely adopted as a 'standard' language for exchanging information on the web, except for SHOE, whose syntax is based on HTML. From all these languages, RDF and RDF Schema cannot be considered as ontology specification languages per se, but as general languages for the description of metadata in the web. Most of these "markup" languages are still in a development phase; hence, they are continuously evolving. Fig. 4-1 presents the main relationships between all these languages as a pyramid of languages, which is considered as the pyramid of languages of the Semantic Web.

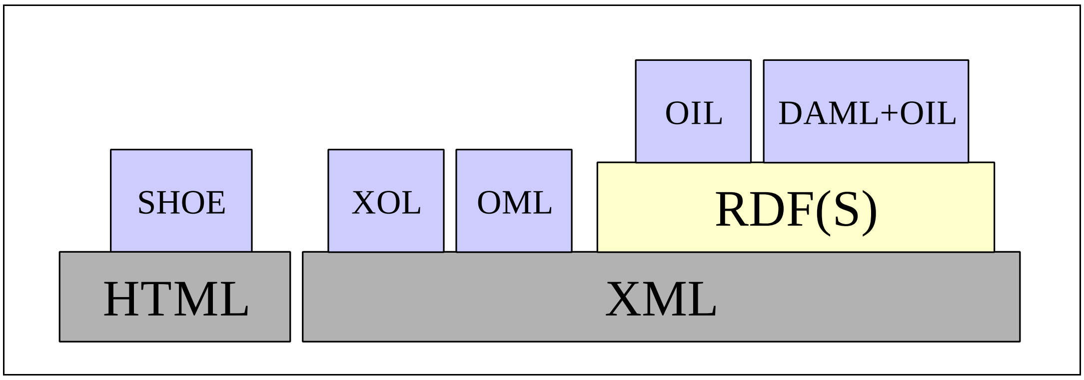


**Fig. 4-1. The pyramid of web-based languages.**

There are many other languages that have been also considered in this survey. For instance, some languages have been created for the specification of specific ontologies, such as CycL (for the Cyc Knowledge Base) or GRAIL (in the medical domain). There are also some other languages that haven't been created specifically for the representation of ontologies, including additional features that are not usual in ontologies, such as NRKL (*Narrative Knowledge Representation Language*). They will be also presented in this section.

For each language that is presented in this section, we will show a brief description, the main sources of information that can be accessed for obtaining extra information about the language (its URL and bibliographic references), and the tools that provide support for the language. We have included tool support in our description of languages because we consider technological support a key aspect for ontological engineering.

## *4.2 Traditional languages*

Languages in this section are presented in alphabetical order.

### 4.2.1 CARIN

The CARIN language is a combination of a Datalog rule-based language (non recursive Horn rules without functions) with the ALN Description Logics. ALN is a simple description logic having the constructors of conjunction between concepts, the type restriction constructor (*forall*), the number restriction constructors (*atleast*, *atmost*), and the negation restricted to atomic concepts. This language has been used in PICSEL.

An ontology in CARIN is made of two components: a terminological component, made of a set of concept definitions, plus possibly a set of inclusion statements (stating inclusion axioms on atomic concepts only), and a set of disjointness statements between pairs of atomic concepts; and a rule component, made of a set of rules such that concepts and roles may appear in rule antecedents but not in rule consequents.

**URL:** *Not available*

**References:**

- Levy, A. Y. and M.-C. Rousset: 1996, `CARIN: A Representation Language Integrating Rules and Description Logics'. In: Proceedings of the European Conference on Artificial Intelligence. Budapest, Hungary.

**Tools that import the language:** WebODE.

**Tools that export the language:** WebODE.

### 4.2.2 FLogic

FLogic (*Frame Logic*) was developed in 1995 in Karlsruhe University. It integrates frames and first order predicate calculus. It accounts in a clean and declarative fashion for most of the structural aspects of object-oriented and frame-based languages, such as object identity, complex objects, inheritance, polymorphic types, query methods, encapsulation, and others. It allows the representation of concepts, taxonomies, binary relations, functions, instances, axioms and deductive rules. Its inference engine can be used for constraint checking and deducting new information.

**URL:** *No URL available*

**References:**

- Kifer, M., Lausen, G., Wu, J. Logical Foundations of Object-Oriented and Frame-Based Languages. Journal of the ACM. 1995

**Tools that import the language:** OntoEdit, Ontobroker, Protege-2000.

**Tools that export the language:** OntoEdit, Ontobroker, Protege-2000, WebODE.

### 4.2.3 LOOM

Loom was developed at ISI (University of South California), in 1991. Initially, it was not created for the specification of ontologies, but for general knowledge bases, expert systems and other intelligent application programs. It is a descendent of the KL-ONE family: it is based on description logics and production rules (achieving a tight integration between rule-based and frame-based paradigms). It allows the representation of concepts, taxonomies, n-ary relations, functions, axioms and production rules. Reasoning is limited to automatic classifications (taxonomies can be created automatically from concept definitions), consistency checking and production rules execution.

**URL:** http://www.isi.edu/isd/LOOM/LOOM-HOME.html

**References:**

- [Loom Guide 91] Loom User's Guide for Loom version 1.4. ISX Corporation, http://www.isi.edu/isd/LOOM/documentation/usersguide1.4.ps, August 1991.
- [MacGregor, 91] MacGregor, R. Inside the LOOM classifier. SIGART bulletin. #2(3):70-76. June, 1991.

**Tools that import the language:** Ontosaurus.

**Tools that export the language:** Ontolingua, Ontosaurus.

### 4.2.4 OCML

The OCML modelling language was developed in 1993 at KMI (Open University) in the context of the VITAL project. It is very similar to Ontolingua, providing additional components, such as deductive and production rules, and operational definitions for functions. It can be considered as a kind of "operational Ontolingua". It allows the specification and operationalisation of functions, relations, classes, instances and rules. It also includes a very powerful constraint checker, which can check both type and cardinality restrictions as well as general, 'free-form' constraints, associated to relations, slots and classes. OCML has been used in dozens of applications, to support knowledge management processes, ontology development, e-commerce and knowledge based system development. Application domains include medicine, electronic publishing, social science, corporate organizational memories, engineering design, web portals and several others. Translators currently exist from OCML to Ontolingua, RDFS, and to an XML-based representation of OCML models. OCML modelling is also supported by a large library of reusable models, providing a useful resource for the knowledge modelling community. This library can be accessed through the WebOnto editor. OCML is freely available for non-commercial use.

**URL:** http://kmi.open.ac.uk/projects/ocml/

**References:**


- Domingue, J., Motta, E. and Corcho, O. (1999). Knowledge Modelling in WebOnto and OCML: A User Guide.
- Motta E. (1999). Reusable Components for Knowledge Modelling. IOS Press, Amsterdam, The Netherlands. ISBN: 1 58603 003 5.


**Tools that import the language:** WebOnto, Apollo.

**Tools that export the language:** WebOnto, Apollo.

### 4.2.5 Ontolingua

Ontolingua was developed in 1992 by KSL (Stanford University), on top of KIF, combining the KR paradigms of frames (Frame Ontology) and first order predicate calculus (KIF). It allows the representation of concepts, taxonomies of concepts, n-ary relations, functions, axioms, instances and procedures. Its high expressiveness led to difficulties in building reasoning mechanisms for it, although a theorem prover has been developed recently for KIF expressions.

**URL:** http://ontolingua.stanford.edu

**References:**


- V. K. Chaudhri, A. Farquhar, R. Fikes, P. D. Karp, & J. P. Rice. Open Knowledge Base Connectivity 2.0. Knowledge Systems Laboratory, January, 1998.
- R. Fikes, A. Farquhar, & J. Rice. Tools for Assembling Modular Ontologies in Ontolingua. Knowledge Systems Laboratory, April, 1997.
- R. Fikes & A. Farquhar. Large-Scale Repositories of Highly Expressive Reusable Knowledge. Knowledge Systems Laboratory, April, 1997.
- A. Farquhar, R. Fikes, & J. Rice. The Ontolingua Server: A Tool for Collaborative Ontology Construction. Knowledge Systems Laboratory, September, 1996.


**Tools that import the language:** Ontolingua, Protege-2000.

**Tools that export the language:** Ontolingua, Ontological Constraints Manager (OCM), Protege-2000.

## *4.3 Web-based ontology specification languages*

In this section, there is not an alphabetical order in the presentation of languages, because we have selected the order that is imposed by the pyramid of languages presented above.

### 4.3.1 SHOE

SHOE is being developed in the University of Maryland. It does not use XML, but an extension of HTML, because its original specification was presented very early, in 1996. It adds the tags that are necessary to embed arbitrary semantic data into web pages, which are divided into two categories: tags for constructing ontologies and tags for annotating web documents.

SHOE allows representing concepts, their taxonomies, n-ary relations, instances and deduction rules, which are used by its inference engine to obtain new knowledge.

**URL:** http://www.cs.umd.edu/projects/plus/SHOE/

**References:**

- A Portrait of the Semantic Web in Action, by Jeff Heflin and James Hendler. IEEE Intelligent Systems, 16(2), 2001.
- Searching the Web with SHOE, by Jeff Heflin and James Hendler. In AAAI-2000 Workshop on AI for Web Search. 2000.
- SHOE: A Knowledge Representation Language for Internet Applications, by Jeff Heflin, James Hendler, and Sean Luke. Technical Report CS-TR-4078 (UMIACS TR-99-71). 1999.
- Ontology-based Web Agents, by Sean Luke, Lee Spector, David Rager, and James Hendler. In Proceedings of First International Conference on Autonomous Agents 1997, AA-97.

**Tools that import the language:** *Not available.*

**Tools that export the language:** *Not available.*

### 4.3.2 XOL

XOL was developed by the AI center of SRI, in 1999, as a XMLization of a small subset of primitives from the OKBC protocol. It is a very restricted language in which just concepts, taxonomies and binary relations can be specified. No inference mechanisms are attached to it, as it was mainly designed for the exchange of ontologies in the biomedical domain.

**URL:** http://www.ai.sri.com/~pkarp/xol/

**References:**

- Karp, R. Chaudhri, V., Thomere, J. XOL: an XML-based Ontology Exchange Language. July, 1999.

**Tools that import the language:** *Not available.*

**Tools that export the language:** *Not available.*

### 4.3.3 OML

OML (*Ontology Markup Language*) is an ontology specification language that is being developed at the Washington University. It is based on description logics and conceptual graphs, and allows representing concepts, organized in taxonomies, relations and axioms in first order logic.

**URL:** http://www.ontologos.org/OML/OML%200.3.htm

**References:**

- Robert E. Kent (1999). Conceptual Knowledge Markup Language: The Central Core. In the Electronic Proceedings of the Twelfth Workshop on Knowledge Acquisition, Modeling and Management (KAW99). Banff, Alberta, Canada, 16-21 October 1999.
- Robert E. Kent (1999). Conceptual Knowledge Markup Language: An Introduction. In Netnomics: Economic research and electronic networking. Special Issue on Information and Communication Middleware.

**Tools that import the language:** *Not available.*

**Tools that export the language:** *Not available.*

### 4.3.4 RDF and RDF Schema

RDF and RDF Schema are being developed inside the W3C consortium for describing web resources. Its combination is known as RDF(S), and is intended for the representation of knowledge using the semantic networks paradigm. This language is much less expressive than other ones presented in this section, just allowing the representation of concepts, taxonomies of concepts and binary relations. It is also intended to provide mechanisms to explicitly represent services, processes and business models, allowing non-explicit information to be recognized. Inference engines have been provided for this language, mainly for constraint checking.

**URL:** http://www.w3.org/TR/rdf-schema/

**References:**

- Lassila, O., Swick, R. Resource Description Framework (RDF) Model and Syntax Specification. W3C Recommendation. January, 99. http://www.w3.org/TR/PR-rdf-syntax.

- Brickley, D., Guha, R.V. Resource Description Framework (RDF) Schema Specification. W3C Proposed Recommendation. March, 1999. http://www.w3.org/TR/PR-rdf-schema

**Tools that import the language:** CORPORUM-OntoBuilder, OILed, OntoEdit, Protege-2000, WebODE.

**Tools that export the language:** CORPORUM-OntoBuilder, OILed, OntoEdit, Protege-2000, WebODE.

### 4.3.5 OIL

OIL is being developed in the context of the European IST Ontoknowledge project. It can be considered as the precursor of DAML+OIL, setting up the foundations for the design of a language for the Semantic Web. It is built on top of RDF(S)., using as much as possible RDF(S) constructs in order to maintain backward compatibility. OIL combines the widely used modelling primitives from frame-based languages with the formal semantics and reasoning services provided by description logics. The FaCT classifier is used to perform automatic classifications of concepts and constraint checking in taxonomies of concepts.

**URL:** http://www.ontoknowledge.org/oil/

**References:**

- D. Fensel et al.: OIL in a nutshell In: Knowledge Acquisition, Modeling, and Management, Proceedings of the European Knowledge Acquisition Conference (EKAW-2000), R. Dieng et al. (eds.), Lecture Notes in Artificial Intelligence, LNAI, Springer-Verlag, October 2000. M.C.A. Klein et al.: The Relation between Ontologies and Schema-Languages: Translating OIL-Specifications to XML-Schema In: Proceedings of the Workshop on Applications of Ontologies and Problem-solving Methods, 14th European Conference on Artificial Intelligence ECAI-00, Berlin, Germany August 20-25, 2000.
- D. Fensel et al.: A Unifying Framework for the Knowledge Web In: Proceedings of the Workshop on Applications of Ontologies and Problem-solving Methods, 14th European Conference on Artificial Intelligence ECAI-00, Berlin, Germany, August 20-25, 2000.

**Tools that import the language:** CORPORUM-OntoBuilder, OILed, Protege-2000.

**Tools that export the language:** OILed, Protege-2000, WebODE.

### 4.3.6 DAML+OIL

DAML+OIL is a semantic markup language for Web resources that has been created as a joint effor of the American and European ontology communities to create a standard language for the Semantic Web. It has been created in the context of the DAML initiative (DARPA Agent Markup Language). It builds on earlier W3C standards such as RDF and RDF Schema, and extends these languages with richer modelling primitives. DAML+OIL provides modelling primitives commonly found in frame-based and description logic languages. DAML+OIL allows representing concepts, taxonomies, binary relations, functions and instances

DAML+OIL (March 2001) extends DAML+OIL (December 2000) with values from XML Schema datatypes. The language has a clean and well defined semantics, and many efforts are being put to provide reasoning mechanisms for DAML+OIL.

**URL:** http://www.daml.org/language/

**References:**

- [Horrocks et al, 01] Horrocks, I., van Harmelen, F. (eds). Reference description of the DAML+OIL ontology markup language. Draft report. 2001. http://www.daml.org/2000/12/reference.html.

**Tools that import the language:** OILed, OntoEdit.

**Tools that export the language:** CORPORUM-OntoBuilder, OILed, OntoEdit, WebODE.

## *4.4 Other ontology specification languages*

### 4.4.1 CycL

CycL is a formal language whose syntax derives from first-order predicate calculus. In order to express real-world expertise and even just plain old common sense knowledge, however, it goes far beyond first order logic. The vocabulary of CycL consists of terms: semantic constants, non-atomic terms (NATs), variables, numbers, strings, etc. Terms are combined into meaningful CycL expressions, ultimately forming meaningful closed CycL sentences (with no free variables.) A set of CycL sentences forms a knowledge base.

**URL**: http://www.cyc.com/cycl.html

**References**:

- Lenat, D., Guha, R. Building large knowledge-based systems. Representation and Inference in the Cyc Project. Addison-Wesley. Reading, Massachussetts. 1990.

**Tools that import the language:** *Not available.*

**Tools that export the language:** *Not available.*

### 4.4.2 GRAIL

GRAIL is the language in which the GALEN ontology (which is an ontology in the medical domain) is specified. It is a description logic language that supports a primitive role hierarchy, transitive roles and concept inclusion axioms. GRAIL has a limited terminological language only conjunction and existential role concepts are supported and an unusual syntax that restricts the way concept expressions can be formed.

**URL:** http://www.opengalen.org/open/CRM/index.html

**References:**

- Rector, S. Bechhofer, C. Goble, I. Horrocks, W. Nowlan, and W. Solomon. The GRAIL Concept Modelling Language for medical terminology. Artificial Intelligence in Medicine, 9:139-171, 1997.

**Tools that import the language:** *Not available.*

**Tools that export the language:** *Not available.*

### 4.4.3 NKRL

NKRL (*Narrative Knowledge Representation Language*) is a knowledge representation language specially developed for describing the semantic content of narrative documents. Narrative documents are multimedia documents that relate the actual or intended state or behaviour of some actors, trying to attain specific results, experience particular situations, manipulate some concrete or abstract materials, communicate with other actors, etc. This language is based on frames, and allows defining concepts, organised in taxonomies, and relationships between them. It also includes a sublanguage (AECS) that allows the construction of complex predicate arguments, and second order tools that allow to encode the connectivity phenomena that, in a narrative situation, can exist between single narrative fragments.

**URL:** *Not available*.

**References:**

- Zarri. G.P. Conceptual Modelling and Knowledge Management for Narrative Multimedia Documents. 2000.
- Zarri, G.P. NKRL, a Knowledge Representation Tool for Encoding the Meaning of Complex Narrative Texts. Natural Language Engineering. Special Issue on Knowledge Representation for Natural Language Processing in Implemented Systems. 3 (1997). pp. 231-253.

**Tools that import the language:** *Not available.*

**Tools that export the language:** *Not available.*

## *4.5 Conclusions*

In this section, we have presented several languages that have been widely used for the specification of ontologies in the last years. These languages have been grouped in three main areas: traditional languages, web-based languages and other non-ontology specific languages (which can be used for coding ontologies).

Differences among languages are, in many cases, relevant. One of the most important difference is their underlying KR paradigm: although most of the languages are based on frames or description logics, some languages also provide primitives and constructors for other paradigms, such as first (and second) order predicate calculus, conceptual graphs, semantic networks, production rules, deductive rules, etc. In many cases, languages are based on combinations of several formalisms.

There is a tight interdependence between expressiveness and reasoning in all the languages, in the sense that the expressive power of a language must be sometimes limited to ensure a good reasoning service. In this sense, some languages have been developed following a layered approach (this is the case of OML and OIL), in which the lower layers (the less expressive ones) have full reasoning support and the higher layers (the most expressive ones) have limited reasoning support, as most of the times they could fall into undecidability.

Finally, the selection of an ontology specification language for the development of an ontology will not only depend on the characteristics of the language, but also on the tools that support it, the applications in which the ontology will be used, and the availability of reusable ontologies in the same domain in a specific language.

## *4.6 References*

The main references for this section have been already provided in the template for the definition of each language.

# 5 Ontology-based applications

In this section, we present several relevant ontology-based applications from diverse areas of research, such as Knowledge Management, e-commerce, Natural Language (processing, generation and translation), intelligent integration of information, information retrieval, semantic portals, education, etc. Sometimes, the task of classifying an application inside an area is hard, because many of them combine and solve problems from different areas. In those cases, we have decided to include the application in one of these areas, and include the other areas as additional topics.

The list of applications that is presented here does not attempt to be complete. On the contrary, it aims at presenting interesting work in different areas from the different groups involved in the ontology research and development community.

## *5.1 Knowledge Management*

In this section, we show approaches to the modelling of corporate knowledge in corporate knowledge management systems. Ontologies play a key role in this framework, as they model the main information used inside the organization, and provide a shared view of it.

### 5.1.1 CoMMA: Corporate Memory Management through Agents

CoMMA is an IST project (1999-12217) aiming at implementing a corporate memory management framework in the context of two scenarios: insertion of new employees in the company and support of technology monitoring. [CoMMA, 2000]. COMMA studies the corporate memory as a "corporate semantic Web" where the ontology O'CoMMA is the keystone. In [Gandon, 2001] are presented both the approach adopted to build O'CoMMA and the resulting ontology. Adaptation and customization of agents to users, rely on machine learning techniques presented in [Kiss and Quinqueton, 2001]. Finally, CoMMA uses multi-agents systems to deployment of a software architecture above the distributed information landscape of the corporate memory [Gandon et al., 2000]. On the one hand, individual agents locally adapt to users and resources they are dedicated to. On the other hand, thanks to cooperating software agents distributed over the network they capitalize an integrated and global view of the corporate memory.

**URL:** http://www.si.fr.atosorigin.com/sophia/comma/Htm/HomePage.htm

**References:**


- CoMMA Corporate Memory Management through Agents, In Proc. E-Work & E-Business, Madrid.
- F. Gandon, Engineering an Ontology for a Multi-Agents Corporate Memory System, In proceedings of ISMICK 2001 Eighth International Symposium on the Management of Industrial and Corporate Knowledge, 22-24 October 2001, Université de Technologie de Compiègne, France, pp 209-228.
- F. Gandon, R. Dieng, O. Corby, A. Giboin, A Multi-Agent System to Support Exploiting an XML-based Corporate Memory, In Proceedings PAKM'00 Basel, Switzerland, October 30-31, 2000, pp 10.1-10.12.
- Kiss A., Quinqueton J. Multiagent Cooperative Learning of User Preferences, Proceedings of European Conference on Machine Learning Principles and Practice of Knowledge Discovery in Databases, 2001.


**Topics:** Agents, Cooperative Knowledge Acquisition, Information Retrieval, Intelligent Information Integration, KA methodologies, Knowledge Elicitation, Knowledge Management, Machine Learning, Ontologies, Semantic Web.

**Ontologies used:** O'CoMMA.

### 5.1.2 Marchmont Observatory Semantic Search Service

In the autumn of 2000 the UK government created the University for Industry (Ufi) whose goal is to provide flexible learning packages in order to improve the quality of life of individuals and to boost business competitiveness. For the Ufi to be successful associated researchers and policy makers would need to discover and disseminate good practice on lifelong learning. It was decided that the main supporting mechanism for this would be a Web portal, termed the National Observatory. Part of this portal is the Marchmont Observatory Semantic Search Service built on ontologies which focus on the attributes associated with lifelong learning. These ontologies index a 'Good Practice' database which hold several hundred hand-coded summaries of articles describing lifelong learning initiatives.

Technically, the observatory interface is a flash™ movie which links to the WebOnto server on which the observatory libraries are stored.

**URL:** http://kmi.open.ac.uk/observatory/

**References:**


- Domingue, J, Motta. E., Buckingham Shum, S, Vargas-Vera, M. , Kalfoglou, Y. and Farnes, N. (2001) Supporting Ontology Driven Document Enrichment within Communities of Practice. K-CAP'01, The First International Conference on Knowledge Capture. Victoria, British Columbia, Canada, October 21-23, 2001 (available at http://kmi.open.ac.uk/projects/akt/kcap01_john_final.pdf).


**Topics:** Knowledge Management, Ontologies.

**Ontologies used:** *Not available.*

### 5.1.3 MGT

The MGT decision-support tool is a web-based system for supporting guideline-centred healthcare. It is available in both a desk-top and a hand-held portable version (which can also be used "off-line"). It is built upon a hierarchy of both medical and supporting non-medical ontologies and can be integrated with a patient database. Having selected a patient from the database, a physician can view the medical history, add new data and based on such data, directly access the most relevant portion of the guidelines, within a web-based discussion-space.

**URL**: http://kronsteen.open.ac.uk/mgt/

**References:** *Not available.*

**Topics**: Medical guidelines, knowledge representation, ontologies

**Ontologies used:** *Not available.*

### 5.1.4 MyPlanet

MyPlanet is an ontology-driven personalised Web-based service. Users submit an email story (e-Story) about a potentially interesting theme in the context of a research lab. We then annotate the e-Story with knowledge structures drawn from the underlying ontology, render the e-Story to produce a high quality Web page to display it, and alert users who might be interested in this e-Story. We also populate our ontology with information extracted from the e-Story, which allow us to provide reasoning services that go beyond traditional keyword-based searches. A designated front-end allow the users to browse the e-Stories archive by specifying their preferences with respect to e-Story's contents.

**URL:** http://eldora.open.ac.uk/my-planet/

**References:**


- Kalfoglou, Y., Domingue, J., Motta, E., Vargas-Vera, M., Buckingham-Shum, S. MyPlanet: an ontology-driven Web-based personalised news service. IJCAI'01 workshop on Ontologies and Information Sharing. Seattle, USA. August, 2001.


**Topics:** Knowledge Management, Ontologies, Semantic Web.

**Ontologies used:** kmi-planet ontology.

### 5.1.5 PatMan

The PATMAN project is concerned with the provision of integrated support for tackling medical and organisational issues during patient management. The PatMan discussion forum integrates the D3E technology for discussion and debate with an ontology-driven query answering engine, which makes it possible to retrieve knowledge about a particular medical guideline. The integration of these tools provides a powerful solution to support the interpretation process of a medical guideline by a community of healthcare providers. D3E supports a web-based set of discussion spaces centred around the various sections of the guideline, while the ontology-driven query answering system makes it possible to retrieve background domain knowledge and to perform semantic search.

**URL:** http://kmi.open.ac.uk/projects/patman/

**References:**

- Motta E., Buckingham-Shum, S. and Domingue, J. (2000). Ontology-Driven Document Enrichment: Principles, Tools and Applications. International Journal of Human-Computer Studies, 52(6), pp. 1071-1109.

**Topics:** Electronic Publishing, Knowledge Management, Ontologies, Semantic Web.

**Ontologies used:** Medical Guidelines ontology, Pressure Ulcer Ontology, Generic MedicalOntology.

### 5.1.6 PlanetOnto

KMi Planet is a web based news server which facilitates communication within an organization, by supporting web-based electronic publishing and the automatic generation of a 'front page' from the news items received by a server. It has been in daily use at the Knowledge Media Institute for a number of years and several versions have been customised for corporate organizations, local schools and academic centres. As an archive of news grows, a number of knowledge management problems arise: how to provide semantic search and knowledge retrieval facilities in an effective and sustainable way; how best to provide individualized presentations and news alerts; how best to emulate the behaviour of a newsroom team. To address these challenges we have developed an integrated suite of tools, which is called PlanetOnto. These tools allow ontology-driven document formalization and augment standard browsing and search facilities with deductive knowledge retrieval. In addition, the PlanetOnto architecture includes specialized agents, which provide personalized news feeds and alerts and can proactively identify potentially interesting news items.

**URL:** http://kmi.open.ac.uk/projects/planetonto/

**References:**

- Domingue, J. B. and Motta, E. (2000). Planet-Onto: From News Publishing to Integrated Knowledge Management Support. IEEE Intelligent Systems, 15(3), May-June 2000, pp. 26-32.

**Topics:** Electronic Publishing, Knowledge Management, Ontologies, Semantic Web.

**Ontologies used:** KMi Planet ontology

## *5.2 E-commerce*

This section presents to ongoing projects in the application of ontologies to e-commerce. Their aim is the development of e-commerce platforms where underlying knowledge is codified by means of ontologies, which are also used (in the case of MKBEEM) for natural language processing (understanding user queries, translating products in catalogues and generating natural language text).

### 5.2.1 MKBEEM

The MKBEEM platform focuses on adding multilinguality to the following stages of the information cycle for multilingual B2C portal services: products or services content and catalogue semi-automated maintenance; automated translation and interpretation of natural language user requests, and natural dialogue interactivity and usability of the service making use of combined navigation and natural language inputs.

The main overall goals of MKBEEM are to: develop intelligent knowledge-based multilingual key components (NLP and KRR) for applications in a multilingual electronic commerce platform; validate and assess the prototypes on a pan-European scale (France and Finland) with three basic languages (Finnish, English and French) and two optional languages.

**URL:** http://www.mkbeem.com/

**References:**

- Leger, A. and others. *Ontology domain modeling support for multilingual services in e-Commerce: MKBEEM*. ECAI'00 Workshop on Applications of Ontologies and PSMs. Berlin. Germany. August, 2000.


**Topics:** e-commerce, ontologies, NL processing, NL generation.

**Ontologies used:** SNCF services, ELLOS metamodel.

### 5.2.2 SMART-EC

The SMART-EC platform is an ontology-based intermediation platform for the provision of complex services on Internet. It is being developed in the context of the IST project SMART EC. The following services are provided by this platform:

- Brokering of services between individual service providers and final users requesting complex services.
- Definition and implementation of the life cycle of a complex service, include the required steps for the service provision and the transactional properties of the service.
- One stop shopping, for multi-items purchases giving access to multiple e-commerce sites.

**URL:** http://www.telecom.ntua.gr/smartec/

**References:** *Not available.*

**Topics:** e-commerce, ontologies.

**Ontologies used:** *Not available*.

## *5.3 Natural Language Processing*

The goal of Natural Language Processing (NLP) is to design and build a computer system that will analyze, understand, and generate languages that humans use naturally.

The origins of NLP are in the mid-fifties in the field of machine translation. This task proved to be much more complex than what it was supposed to be, so research focused in more specific areas such as natural language interfaces. The applications collected in this report can be classified in natural language understanding, natural language generation and machine translation. This classification cannot be strict, because many of them deal with several research areas of natural language.

### 5.3.1 Natural Language Understanding

#### 5.3.1.1 AlFresco

ALFRESCO is an interactive, natural-language centered system for a user interested in Fourteenth Century Italian frescoes and monuments, with the aim not only of providing information, but also of promoting other masterpieces that may attract the user. It runs on a workstation connected to a videodisc unit and a touchscreen. The system, besides understanding and using language, integrates NL and hypermedia both in input and output. In input, the system combines the interpretation of NL deictic references with pointing to images displayed on a touch screen. In output, images videos and generated text with buttons are yielded that offer entry points for further hypertextual exploration. The result is that the user communicates linguistically and by manipulating various entities, images, and text itself .

**URL:** http://ecate.itc.it:1024/projects/alfresco.html

**References:**


- O. Stock. Natural Language and the Exploration of an Information Space: the ALFresco Interactive System. In Proceedings of the International Joint Conference in Artificial Intelligence (IJCAI91), pages 972--978, 1991.
- O. Stock, G. Carenini, F. Cecconi, E. Franconi, A. Lavelli, B. Magnini, F. Pianesi, M. Ponzi, V. Samek-Lodovici and C. Strapparava. ALFRESCO Enjoying the Combination of Natural Language Processing and Hypermedia for Information Exploration. In Mark T. Maybury, editor, Intelligent Multimedia Interfaces, The MIT Press, pp. 197-224, chapter 9. Extended and revised version of a paper published at IJCAI-91. 1993.

- Stock, Oliviero, et al. "Human-Computer Interaction through Natural Language and Hypermedia in AlFresco," SIGCHI Bulletin, 28 (3):102--107 (1996).

**Topics:** NL processing.

**Ontologies used:** Generalized Upper Model.

#### 5.3.1.2 ITEM search engine

The ITEM (recuperación de Información Textual en un Entorno Multilingüe con técnicas de lenguaje natural) project (CYCIT TIC96-1243-C03) had two main goals: 1) integrating a variety of language resources and tools for Natural Language Processing in Spanish, Catalan, Basque and English, and 2) demonstrating the application of such resources and tools in a multilingual information retrieval system. The multilingual search engine is one of the results of the ITEM project. This search engine performs full lexical processing (morphological analysis, tagging and word sense disambiguation) on documents and queries in order to provide language-neutral indexes for querying and retrieval. The indexing terms are the EuroWordNet/ITEM InterLingual Index records that link wordnets in 10 languages of the European Community The user is allowed to select from a number of NL Processing options, and to refine the results of such lexical processing.

**URL:** http://terral.ieec.uned.es/cli, http://sensei.ieec.uned.es/item/principal.htm

**References:**

- J. Gonzalo, A. Peñas, and F. Verdejo. (1999). Lexical ambiguity and Information Retrieval Revisited. Proceedings of the Joint SIGDAT. Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC-99)}, MaryLand.
- Verdejo, F., Gonzalo, J., Peñas, A., López, F. and Fernández, D. (2000).Evaluating wordnets in a Cross-Language Retrieval Environment: the ITEM search engine. Proceedings of the Second Language Resources and Evaluation Conference, Athens, pp 1769-1774.

**Topics:** Information Retrieval, NL processing.

**Ontologies used:** EuroWordNet.

#### 5.3.1.3 OncoTerm

One of the objectives of the project OncoTerm (PB98-1342) was to facilitate the translation of medical texts within the domain of Oncology by elaborating a bilingual terminological database, based on the information extracted from specialized texts as well as medical dictionaries. Medical concepts are organized in categories represented by schemas, which are systematically applied to all category members.

As a consequence, OntoTerm® a full-fledged Terminology Management System (TMS) with multimedia capabilities, was built. Unlike other TMSs, OntoTerm is truly concept-based, as it forces the terminologist to develop a conceptual structuring of the domain. It does this by making it easy to create ontologies. Ontologies in OntoTerm can contain general domain-specific knowledge, or both. Once the conceptual structure (ontolgoy) has been created. It has several other features, such as HTML report generation, and it is based on ISO standards for terminology interchange.

**URL:** http://www.ontoterm.com/

**References:**

- Moreno Ortiz, A. (2000): "OntoTerm: un sistema abierto de representación conceptual". XVI Congreso de SEPLN. Vigo. 2000.
- Moreno Ortiz, A. & Pérez Hernández, C. (2000): "Reusing the Mikrokosmos Ontology for Concept-Based Multilingual Terminology Databases". Proceedings of the 2nd International Conference on Language Resources and Evaluation (LREC 2000). Athens, Greece. 31 May -2 June.

**Topics:** NL processing, Ontologies.

**Ontologies used:** Mikrokosmos' ontology.

## 5.3.2 Natural Language Translation

### 5.3.2.1 ALT-J/E

ALT-J/E (Automatic Language Translator - Japanese to English) is a transfer -based Japanese-to-English machine translation system being developed at the NTT Communication Science laboratories since 1987.

The aim is to produce a high quality system that can be used to translate raw Japanese text, with no human pre-editing. For example, ALT-J/E automatically rewrites complicated Japanese expressions into simpler ones. In addition, it analyses the text to supplement elements that are not expressed in Japanese, but are in English, such as ellided subjects and objects or possessive pronouns.

They are also working on multilingual machine translation. In 1999 they built a prototype Japanese-to-Malay system, ALT-J/M which uses the same techniques but with a different languag e pair.

**URL:** http://www.kecl.ntt.co.jp/icl/mtg/topics/mtg-index.html

**References:**

- Ogura, Kentaro, Francis Bond, & Yoshifumi Ooyama. (1999). ALT-J/M: A prototype Japanese-to-Malay Translation System. In Machine Translation Summit VII, 444-448, Singapore.
- Bond, Francis, Satoru Ikehara, Kentaro Ogura, Satoshi Shirai, & Hiromi Nakaiwa. (1993). ALT-J/E -- the Automatic Language Translator -- Japanese to English. In Forum, 1-7. Key Centre for Asian Languages and Studies, University of Queensland.
- S. Ikehara, S. Shirai, A. Yokoo, and H. Nakaiwa. (1991). Toward an MT System without Pre-Editing - Effects of New Methods in ALT-J/E-. In Third Machine Translation Summit: MT Summit III, 101-106. Washington DC.

**Topics:** NL Translation.

**Ontologies used:** Goi-Taikei's ontology.

### 5.3.2.2 GAZELLE

GAZELLE (formerly Japangloss, a Japanese-English newspaper MT system built at USC/ISI within the PLANGLOSS MT project), is a system that translates Japanese, Arabic, and Spanish texts into English. The system operates over unrestricted newspaper text. The goal is to improve on existing system's accuracy and to make it easier and faster to develop MT systems for new language pairs. The R&D work includes parsing and semantic analysis of the various languages, English sentence planning and generation, Interlingua/ontology construction, and lexicon creation for Japanese, Arabic, Spanish, and English. This project started in 1994 and is funded by the Department of Defense.

**URL:** http://www.isi.edu/natural-language/projects/GAZELLE.html

**References:**

- Germann, U. (1998). Making Semantic Interpretation Parser-Independent. Proceedings of the 4th AMTA Conference.
- Germann, U. (1998). Visualization of Protocols of the Parsing and Semantic Interpretation Steps in a Machine Translation System. COLING-ACL Workshop on Content Visualization and Intermedia Representations. Montreal, Quebéc.
- Knight, K. and K. Yamada. (1999). A Computational Approach to Deciphering Unknown Scripts. Proceedings of the ACL Workshop on Unsupervised Learning in Natural Language Processing.
- Stalls, B. and Knight, K. (1998). Translating Names and Technical Terms in Arabic Text. COLING/ACL Workshop on Computational Approaches to Semitic Languages. Montreal, Quebéc.

**Topics:** NL Translation.

**Ontologies used:** SENSUS.

### 5.3.2.3 Mikrokosmos

The Mikrokosmos Machine Translation System is a knowledge-based machine translation (KBMT) system developed by researchers at the Computing Research Laboratory (CRL) of New Mexico State University.

The Mikrokosmos project was carried out with the ultimate objective of defining a methodology for representing the meaning of natural language texts in a language-neutral interlingual format called a text meaning representation (TMR). The TMR represents the result of analysis of a given input text in any one of the languages supported by the KBMT system, and serves as input to the generation process. The meaning of the input text is represented in the TMR as elements of an independently motivated model of the world (or ontology). The link between the ontology and the TMR is provided by the lexicon, where the meanings of most open class lexical items are defined in terms of their mappings into ontological concepts and their resulting contributions to TMR structure.

**URL:** http://crl.nmsu.edu/mikro

**References:**

- Beale, Stephen, Sergei Nirenburg, and Kavi Mahesh (1995). Semantic Analysis in the Mikrokosmos Machine Translation Project. In Proceedings of the Second Symposium on Natural Language Processing (SNLP-95), August 2-4. Bangkok, Thailand.
- Nirenburg, Beale, Mahesh. (1996).Lexicons in the MikroKosmos Project. In Proceedings of the AISB'96 Workshop on Multilinguality in the Lexicon, Brighton, UK, April 1-2, 1996

**Topics:** NL Translation.

**Ontologies used:** Mikrokosmos' ontology.

### 5.3.2.4 PANGLOSS

The PANGLOSS project is a three-site collaborative effort to build a large-scale knowledge-based machine translation system. The Pangloss team, funded by the Department of Defense, is distributed at three sites, each one providing a key components of PANGLOSS: New Mexico State University's Panglyzer, Carnegie Mellon's translator's workstation, and USC/ISI's PENMAN English generation system. All of these systems combine to form a prototype Spanish-English translation system.

The current system, Pangloss Mark III, differs from all other MT systems because it employs not a single translation engine but a set of several engines (a Knowledge Based system, an Example Based system and a Lexical Transfer system). They are run in parallel to propose translations of various portions of the input, from which the final translation is selected by a statistical language model.

**URL:** http://www.lti.cs.cmu.edu/Research/Pangloss/

**References:**

- Frederking, R., Nirenburg, S., Farwell, D., Helmreich, S., Hovy, E., Knight, K., Beale, S., Domashnev, C., Attardo, D., Grannes, D., Brown, R. Integrating Translations from Multiple Sources within the Pangloss Mark III Machine Translation. In Proceedings of the first conference of the Association for Machine Translation in the Americas, AMTA-94, Columbia, MD, 1994.
- Nirenburg, S. and Frederking, R. Toward Multi-Engine Machine Translation. In Proceedings of the DARPA Human Language Technology Workshop, Princeton, NJ, 1994.
- Nirenburg, Sergei, editor. ``The PANGLOSS Mark III MachineTranslation System''. CMU-CMT-95-145. A Joint Technical Report by NMSU CRL, USC ISI and CMU CMT. April 1995.

**Topics:** NL Translation.

**Ontologies used:** SENSUS.

## 5.3.3 Natural Language Generation

### 5.3.3.1 GIST

The GIST (Generating InStructional Text) project (LRE 062-09) addresses the construction of a multilingual generation system for for the automatic production of texts describing administrative procedures (e.g. the instructions that a citizen has to follow to apply for pension benefits) starting from language independent specifications. Three languages are considered: English, German and Italian. The application domain of the developed prototype is represented by instructions on how to fill out pension forms. The GIST system aims at providing good quality drafts of text; such drafts can then be revised and post-edited by professional writers and/or translators.

**URL:** http://ecate.itc.it:1025/projects/gist.html

**References:**


- Elena Not and Oliviero Stock. Automatic generation of instructions for citizens in a multilingual community. In Proceedings of the First European Language Engineering Convention, Paris, France, July 1994.
- Not, E. and E. Pianta, (1995), "Issues of Multilinguality in the Automatic Generation of Administrative Instructional Texts" in M. Gori and G. Soda (eds.) Topics in Artifical Intelligence, Proceedings of the AI*IA '95 Congress, LNAI, Springer Verlag
  Richard Power and Nico Cavallotto. GIST: Multilingual generation of administrative forms. In INLG'96, pages 17--19, Herstmonceux Castle, Sussex, 1996.
- GIST Team. Generating InStructional Text. Final Report. Date 30/9/1996


**Topics:** NL generation.

**Ontologies used:** Generalized Upper Model.

### 5.3.3.2 Komet

The KOMET text generation system aimed at providing the full flexibility of natural language text production in multiple languages (currently in English, German and Dutch), plus the ability to restrict that flexibility for particular applications in principled, well-specified ways.

In KOMET they pursue a systemic-functionally based approach to modelling the linguistic resources needed for text generation. Linguistic resources are organized along a number of dimensions: strata, metafunctions, axis, rank and delicacy.

**URL:** http://www.darmstadt.gmd.de/publish/komet

**References:**


- John A. Bateman and Elke Teich. Selective information presentation in an integrated publication system: an application of genre-driven text generation. Information Processing and Management: an international journal, 31(5):753--768, September 1995.
- Melina Alexa, John Bateman, Eli Hagen, Klaas Jan Rondhuis, and Elke Teich. Multilingual generation for multiple purposes. In Proceedings of the workshop Language engineering on the information highway', Santorini, September 1994, 1994. Long form available as technical report, GMD/Institut für Integrierte Publikations- und Informationssysteme, Darmstadt, Germany.
- Elke Teich, Liesbeth Degand, and John A. Bateman. Multilingual textuality: Experiences from multilingual text generation. In G. Adorni and M. Zock, editors, Trends in Natural Language Generation: an artificial intelligence perspective, number 1036 in Lecture Notes in Artificial Intelligence, pages 331--349. Springer-Verlag, Berlin, New York, 1996.


**Topics:** NL generation.

**Ontologies used:** Generalized Upper Model.

### 5.3.3.3 KPML

The KPML (Komet-Penman MultiLingual) system is an ongoing development drawing on over a decade of experience in large-scale grammar development work for natural language generation (NLG). The system is a graphically-based development environment for the construction, maintenance, and use of large-scale grammars written with the framework of Systemic-Functional Linguistics (SFL). KPML offers a robust, mature platform for large-scale grammar engineering that is particularly oriented to multilingual grammar development and generation. Grammars have been developed using KPML for a variety of languages including English, German, Dutch, Chinese, Spanish, Russian, Bulgarian, and Czech. Many of these grammars are freely available for further research and development work within the NLG community.

KPML is currently maintained at the University of Bremen, Faculty of Linguistics and Literature.

**URL:** http://www.fb10.uni-bremen.de/anglistik/langpro/kpml/README.html

**References:**


- John A. Bateman.(1997) 'Enabling technology for multilingual natural language generation: the KPML development environment'. Journal of Natural Language Engineering, 3(1):15--55.


**Topics:** NL generation.

**Ontologies used:** Generalized Upper Model.

### 5.3.3.4 Ontogeneration

Ontogeneration is a system for generation of Spanish texts in the domain of chemical substances, funded by UPM (A9706). It uses NL technologies to answer queries about chemical groups, elements and properties, and comparisons between elements.

For this purpose the following steps have been taken: (a) an ontology, called Chemicals, in the chemical domain developed under the METHONTOLOGY framework and the Ontology Design Environment (ODE) has been taken as a knowledge source; (b) the linguistic ontology GUM (Generalized Upper Model) used for other languages has been extended and modified for Spanish; (c) a Spanish grammar has been built following the systemic-functional model by using the KPML (Komet Penman MultiLingual) system. As a result, the final system, called Ontogeneration, permits the user to consult all the information of the ontology in Spanish.

**URL:**
http://delicias.dia.fi.upm.es/../proyectos/terminados/ontogeneration/ontogeneration_proyecto_Esp.html

**References:**

- G. Aguado, A. Bañón, J. Bateman, S. Bernardos, M. Fernández, A. Gómez, E. Nieto, A. Olalla, R. Plaza, A. Sánchez (1998), ?ONTOGENERATION: Reusing domain and linguistic ontologies for Spanish text generation?. Workshop of ontologies and problem-solving methods (ECAI?98), Brighton, United Kingdom.
- S. Bernardos and G. Aguado (2001) ?Adapting the Generalized Upper Model to Spanish?. Proceedings of Recent Advances in Natural Language Processing (RANLP-2001), Tzigov Chark, Bulgaria.

**Topics:** NL generation, Ontologies.

**Ontologies used:** Generalized Upper Model.

### 5.3.3.5 Penman

Penman is a natural language sentence generation program developed at USC/ISI. It provides computational technology for generating English sentences, starting with non-linguistic input specifications.

By accepting various input notations, ranging from linguistically very sophisticated to application domain-oriented, Penman is designed to be used effectively by people with various degrees of linguistic and computational sophistication. It has been further developed into the KPML system.

**URL:** http://www.isi.edu/natural-language/penman/penman.html

**References:**

- Hovy, E., Lavid, J., Maier, E., Mittal, V., Paris, C. Employing knowledge resources in a new text planner architecture. In R. Dale, E. Hovy, D. Rösner, and O.Stock, editors, Aspects of automated natural language generation, pages 57--72, Berlin, 1992. Springer-Verlag.
- Robert T. Kasper. A flexible interface for linking applications to PENMAN's sentence generator. In Proceedings of the DARPA Workshop on Speech and Natural Language, 1989. Available from USC/Information Sciences Institute, Marina del Rey, CA.
- Christian M. I. M. Matthiessen and John A. Bateman. Text generation and systemic-functional linguistics: experiences from English and Japanese. Frances Pinter Publishers and St. Martin's Press, London and New York, 1991.

**Topics:** NL generation, Ontologies.

**Ontologies used:** Generalized Upper Model.

### 5.3.3.6 TechDoc

In the TechDoc Project a prototypical software system has been developed, which demonstrates the automatic generation of multilingual technical documents (currently in English, German and French) from a language independent representation of form and content. To perform this task, the system has to include linguistic as well as general technical domain and product specific knowledge, which has been gained by the comparative analysis of parts of manuals in different languages.

Starting from an instance of the concept "plan", a document structure is generated. The elements of the document structure are complex RST relations. In the next step the discourse structures are broken down into clause sequences. As a result of this transformation, a sequence of SPL terms is created. SPL terms can be used as input to the sentence generator PENMAN. In a final step the document structure is exploited for an automatic formatting of the output text.

**URL:** *No URL available*

**References:**


- Dietmar Rösner and Manfred Stede. TECHDOC: a system for the automatic production of multilingual technical documents. In G. Görz, editor, Proceedings of the First German Conference on Natural Language Processing (KONVENS '92), Informatik aktuell. Springer-Verlag, Heidelberg, 1992.
- Dietmar Rösner and Manfred Stede. Generating multilingual documents from a knowledge base: the TECHDOC project. In Proceedings of the 15th. International Conference on Computational Linguistics (COLING 94), volume I, pages 339--346, Kyoto, Japan, 1994.


**Topics:** NL generation.

**Ontologies used:** Generalized Upper Model.

## *5.4 Intelligent Integration of Information*

The growth of the Internet and other on-line information repositories has greatly simplified the access to numerous sources of information. But this growth has vastly complicated tasks involving finding, extracting, merging or synthesizing such information. In this section, we present some approaches that use ontologies for the tasks of performing an intelligent integration of information from several heterogeneous sources.

### 5.4.1 Ariadne

The Ariadne project aims at the development of technology and tools for rapidly constructing intelligent agents to extract, query, and integrate data from web sources. It is being developed in the Information Sciences Institute (ISI) at the University of Southern California, with the support of DARPA. This project tries to improve SIMS, allowing the management of semi-structured sources such as Web pages.

The application can access to the different sources via an Ariadne information mediator, which uses ontologies codified in LOOM. This mediator has mappings between the ontologies and the information sources.

**URL:** http://www.isi.edu/info-agents/ariadne/index.html

**References:**


- Arens, Y.; Knoblock, C.A.; Shen, W-M. "Query reformulation for dynamic information integration". Journal of Intelligent Information Systems, Special Issue on Intelligent Information Integration, 6(2/3). 1996. Pp. 99-130.
- Barish, G.; Knoblock, C.A.; Chen, Y-S; Minton, S.; Philpot, A.; Shahabi, C. "The theaterloc virtual application". Twelfth Annual Conference on Innovative Applications of Artificial Intelligence (IAAI-2000). Austin, Texas, 2000.


**Topics:** Intelligent Integration of Information, Agents.

**Ontologies used:** *Not available.*

### 5.4.2 OBSERVER

OBSERVER is being developed by Universidad del País Vasco, MCC, and the University of Georgia. It uses different ontologies (implemented in CLASSIC) to represent the different information sources, and a set of mappings between the different ontologies. When a user poses a query, the first answer is given using just an ontology. If the user is not satisfied, he can choose other one to extend the result. Given that the correspondence between the concepts of an ontology with the concepts of the other one is not exact, the system provides the estimated loss of information. That is, it estimates how many items are not proper answers, and how many items should appear, and they do not appear.

**URL:** http://siul02.si.ehu.es/~jirgbdat/OBSERVER/

**References:**


- Mena, E.; Illarramendi, A.; Kashyap, V.; Sheth, A.P. "OBSERVER: An Approach for Query Processing in Global Information Systems based on Interoperation across Pre-existing Ontologies", In the international journal Distributed and Parallel Databases (DAPD), Volume 8, Number 2, ISSN 0926-8782, April 2000. Pp. 223-271.


**Topics:** Intelligent Integration of Information.

**Ontologies used:** *Not available.*

### 5.4.3 PICSEL

PICSEL is an information integration system over sources that are distributed and possibly heterogeneous. It has been developed by LRI (France). The approach which has been chosen is to define an information server as a knowledge-based mediator between users and several existing information sources relative to a same application domain. The mediator gives its users the illusion of a centralized and homogeneous information system. It allows them to ask domain-level queries and takes in charge in their place the access to the relevant sources in order to obtain the answers to the queries.

**URL:** http://www.lri.fr/~picsel/

**References:**


- François Goasdoué, Véronique Lattes and Marie-Christine Rousset, The Use of CARIN Language and Algorithms for Information Integration: The PICSEL Project, International Journal ofCooperative Information Systems (IJCIS), 1999.
- François Goasdoué, A Knowledge Based Approach for Information Integration: The PICSEL System, In Nicolas Spyratos, K. Vidyasankar and Gottfried Vossen Eds, Declarative Data Access on the Web, Dagstuhl-Seminar-Report 251, p. 7, Dagstuhl Castle, Germany, September 12-17 , 1999.


**Topics:** Intelligent Integration of Information.

**Ontologies used:** *Not available.*

## *5.5 Information Retrieval*

In this section, we review some applications of ontology-driven information retrieval, in which users can pose queries to the systems and it retrieves relevant information for them.

### 5.5.1 OntoSeek

Ontoseek is a system of cooperating intelligent agents. A first prototype of the system has been developed in a co-operative project between the COnsorzio di RIcerca Nazionale Tecnologia Oggetti (CORINTO), a partnership of IBM Semea, Apple Italia and Selfin SpA, and LADSEB-CNR, as part of a project on retrieval and reuse of object-oriented software components.

OntoSeek is designed for content-based information retrieval from online yellow pages and product catalogs. It combines an ontology-driven content-matching mechanism with a moderately expressive representation formalism. Differently from most of current systems, the user is not assumed to have familiarity with the vocabulary used for component encoding, but the system relies on a large linguistic ontology called Sensus to perform the match between queries and data. It assumes that the information encoding and retrieval processes will involve a degree of interactivity with a human user.

**URL:** *No URL available*

**References:**


- Borgo, S., Guarino, N., Masolo, C., and Vetere, G. (1997). Using a Large Linguistic Ontology for Internet-Based Retrieval of Object-Oriented Components. In Proceedings of 1997.
- Conference on Software Engineering and Knowledge Engineering. Madrid, Knowledge Systems Institute, Snokie, IL, USA: 528-534.

- Guarino, N., Masolo, C., Vetere, G. (1999). OntoSeek: content-based access to the Web. IEEE Intell. Syst. (USA), 70-80, vol.14, no.3, May-June 1999


**Topics:** Agents, Information Retrieval.

**Ontologies used:** SENSUS.

### 5.5.2 WebKB-2

WebKB-2 is a knowledge server that permits Web users to retrieve and add knowledge in a shared knowledge base. The following features distinguish WebKB-2 from other ontology servers or KBMSs: (i) the ontology is large (at present, 69,000 categories and 87,800 links mostly coming from WordNet) and extendible at any time by any user, (ii) asynchronous cooperation between users is supported and encouraged while the knowledge base is kept unique to maximize knowledge interconnection, retrieval and inconsistency detection, (iii) the proposed knowledge representation languages are designed to be both expressive and readable to permit and encourage the users to enter all the knowledge they want (though that still requires motivation).

WebKB-2 is ultimately intended to permit cooperatively-built Yellow-Page like catalogs, that is, permit Web users to publish their information in a way that is automatically retrievable and comparable with other users' knowledge.

**URL:** http://www.webkb.org/

**References:**


- Martin Ph. & Eklund P. (2001). Large-scale cooperatively-built heterogeneous KBs. ICCS'01, 9th International Conference on Conceptual Structures (Springer Verlag, LNAI 2120, pp. 231-244), Stanford University, California, USA. July-August, 2001.
- Martin Ph. & Eklund P. (2000). Knowledge Indexation and Retrieval and the Word Wide Web. IEEE Intelligent Systems, special issue "Knowledge Management and Knowledge Distribution over the Internet", May/June 2000.
- Martin Ph. (1997). The WebKB set of tools: a common scheme for shared WWW Annotations, shared knowledge bases and information retrieval. Proceedings of ICCS'97, 5th International Conference on Conceptual Structures (Springer Verlag, LNAI 1257, pp. 585-588), Seattle, USA, August 4-8, 1997.


**Topics:** Information Retrieval, Knowledge Management, NL processing, Ontologies.

**Ontologies used:** Wordnet.

## *5.6 Semantic Portals and Web Communities*

These applications aim not only at presenting a web site to users, but also at adding semantics to web sites, so that they provide an integrated access to the various information resources of a community or organization.

### 5.6.1 C-WEB

C-WEB (Community-Web) intends to support specific communities on the Web. These communities have in common some identified and formalized knowledge that they use and make evolve in their joint activities, along with a variety of information resources. In this context, the main challenge is to provide a single point of useful, ubiquitous comprehensive, and integrated access to the various information resources of a community.

**URL:** http://cweb.inria.fr/

**References:**


- Managing RDF Metadata for Community Webs, S. Alexaki, V. Christophides, G. Karvounarakis, D. Plexousakis, K. Tolle, Bernd Amann, Irini Fundulaki, Michel Scholl, Anne-Marie Vercoustre, 2nd International Workshop on the World Wide Web and Conceptual Modeling (2000), pp. 140-151.


**Topics:** Semantic Web, Semantic Portals, Knowledge Management.

**Ontologies used:** physics (mechanics) and biology (genome). *Under development*.

### 5.6.2 SEAL

SEAL (SEmantic portAL) is a generic ontology-based approach for developing semantic portals. It exploits the semantics for providing and accessing information at a portal as well as constructing and maintaining it. This approach has already been used inside AIFB at Karlsruhe University to generate their own web pages on a semantic basis.

**URL:** http://ontobroker.semanticweb.org/ontos/aifb.html

**References:**

- Alexander Maedche, Steffen Staab, Nenad Stojanovic, Rudi Studer, and York Sure. SEmantic portAL - The SEAL approach. In: Creating the Semantic Web. D. Fensel, J. Hendler, H. Lieberman, W. Wahlster (eds.) MIT Press, Cambridge, MA., 2001.

**Topics:** Semantic Web, Semantic Portals, Knowledge Management.

**Ontologies used:** AIFB.

## *5.7 Education*

The role of ontologies in educational systems is discussed in this section. We present some intelligent computer based educational systems that have been used for educational purposes.

### 5.7.1 RichODL

RichODL is a learning Web-based environment developed at the Open University. It is intended for training students and practitioners in modelling and simulation of dynamic systems. RichODL makes use of KMi knowledge technologies such as OCML and WebOnto to provide a means of indexing and searching solved examples of simulation and modelling of dynamic systems and accompanying text, and D3E to provide a space for discussing results. Distributed tutorial material is accessed accessed, modified and executed over the Web. Ontologies are used to describe the physical domain of the modelled systems and modelling mappings and tricks.

**URL:** http://rich-odl.open.ac.uk/

**References:**

- Zdrahal Z., Mulholland P., Domingue J. Hatala M. (2000). Sharing engineering design knowledge in a distributed environment. Behaviour and Information Technology. Vol. 19, No. 3. pp. 189-200.

**Topics:** Individual and Group Learning, Ontologies

**Ontologies used:** *Not available*.

### 5.7.2 SmartTrainer

SmartTrainer is a Computer Based Training System including a set of simulators in the area of Electric Power System. The target task of SmartTrainer is mainly to recover the accidents of substations in the electric power system. When an accident happens, the electric power transmission will be interrupted, and the operators should recover it as quickly as possible. The operators should find the spot of the accident, continue to supply the electric power to some special places such as hospital, police station at once by borrowing some power from the other substations, find the causes of the accident and recover it within the limited time. The goal of the training oriented by SmartTrainer is to improve capability of not only skill-based or rule-based reasoning but also knowledge-based reasoning.

**URL:** http://www.ei.sanken.osaka-u.ac.jp/

**References:**

- L. Jin, M. Ikeda, R. Mizoguchi, Y. Takaoka. Role Explication of Simulation in Intelligent Training Systems by Training Task Ontology. Proc. of AI-ED-97 Workshop on Architectures for intelligent simulation-based learning environments, Kobe, Japan (1997).
- L. Jin, M. Ikeda, R. Mizoguchi, Y. Takaoka, K. Hirobe. Ontological Issues on Computer-based Training, Proc. of PRICAI-95 Workshop. 1995.

**Topics:** Education, Ontologies.

**Ontologies used:** Not available.

### 5.7.3 ScholOnto (Scholarly Ontologies Project)

Research fields and other dynamic domains that cannot easily be modelled require 'sensemaking' tools to support principled disagreement, conflicting perspectives, and the resulting ambiguities and inconsistencies. The ScholOnto project is developing an ontology-based digital library server to support scholarly interpretation and discourse. It enables researchers to describe and debate via a semantic network their view of the contributions a document makes, and its relationship to the literature. By providing an interpretational layer above a digital resource, ScholOnto contrasts with most semantic web applications that require consensus on the structure of a domain.

**URL:** http://kmi.open.ac.uk/projects/scholonto/

**References:**

- S. Buckingham Shum, E. Motta, J. Domingue ScholOnto: An Ontology-Based Digital Library Server for Research Documents and Discourse. International Journal on Digital Libraries (2000), 3 (3), 237-248.

**Topics:** Scholarly/Scientific Publishing, Argumentation, Knowledge Management, Ontologies, Semantic Web.

**Ontologies used:** ScholOnto ontology.

## *5.8 References*

The main references for this section have been already provided in each subsection.

# 6 Events

This section presents a list of relevant past and upcoming events for the communities of ontologies, knowledge management, semantic web and e-commerce. They are ordered by year.

## *6.1 IJCAI 99 Workshop BUS-1: Knowledge Management and Organizational Memory*

This event is concerned about every issue related to knowledge management. It belongs to ECAI 2000

**Location:** Sweden, Stockholm

**Date:** from:1-July-1999; to:31-January-1999

**URL:** http://www.dsv.su.se/ijcai-99/

**Topics:** Knowledge Management.

## *6.2 16th International Joint Conference on Artificial Intelligence IJCAI 1999*

*Not available*

**Location:** Sweden, Stockholm

**Date:** from:31-July-1999; to:6-August-1999

**URL:** http://www.dsv.su.se/ijcai-99/

**Topics:** *Not available.*

## *6.3 IJCAI-99 Workshop on Ontologies and Problem-Solving Methods: Lessons Learned and Future Trends*

The goal of this workshop is to make a significant step towards a framework that captures a thorough understanding of the overall field of ontologies and problem-solving methods. That is, what the key issues are, what we have learned, what the synergies and overlaps are between different aspects in the field, as well as to focus attention on what research issues are likely to be fruitful. We expect to discuss these issues for ontologies, problem-solving methods and for their integration.

**Location:** Sweden, Stockholm

**Date:** from:2-August-1999; to:2-August-1999

**URL:** http://www.dsv.su.se/ijcai-99/

**Topics:** Ontologies, PSMs.

## *6.4 Twelfth Workshop on Knowledge Acquisition, Modelling and Management KAW'99*

The knowledge-acquisition workshops provide a forum for those developing theories, methods, systems, and empirical studies that relate to the process of building intelligent systems of all kinds. Increasing emphasis has been placed on approaches for modelling expertise and for management of knowledge within a variety of computational frameworks.

**Location:** Banff (Alberta), Canada

**Date:** from:16-October-1999; to:21-October-1999

**URL:** http://http://sern.ucalgary.ca/KSI/KAW/KAW99/

**Topics:** Cooperative Knowledge Acquisition, Evaluation of KA, KA methodologies, Knowledge Elicitation, Ontologies, Ontology Learning.

## *6.5 ECAI00 workshop on Applications of Ontologies and PSMs*

The aim of this workshop is to present, discuss and evaluate applications of ontologies and PSMs, from a knowledge representation point of view, in areas such as: e-commerce, knowledge management and enterprise modelling, intelligent integration information, communication between people and organisations, knowledge discovery in databases and data-mining, interoperability between systems (data bases, digital libraries), knowledge elicitation from text and the web, natural language, etc.

**Location:** Germany, Berlin

**Date:** from:2-August-2000; to:6-August-2000

**URL:** http://delicias.dia.fi.upm.es/WORKSHOP/ECAI00/index.html

**Topics:** Ontologies, Software Reuse.

## *6.6 14th European Conference on Artificial Intelligence ECAI 2000*

*Not available*

**Location:** Germany, Berlin

**Date:** from:20-August-2000; to:25-August-2000

**URL:** http://www.ecai2000.hu-berlin.de/

**Topics:** *Not available.*

## *6.7 Workshop on Knowledge Management and Organizational Memories*

This event in concerned about every aspect of knowledge management. This workshop belongs to ECAI 2000

**Location:** Berlin, Germany

**Date:** from:21-August-2000; to:22-August-2000

**URL:** http://www-sop.inria.fr/acacia/WORKSHOPS/ECAI2000-OM/call.html

**Topics:** Knowledge Management.

## *6.8 First International Workshop on the Semantic Web (SemWeb) in conjunction with the ECDL 2000*

This workshop is concerned about every issue on semantic Web.

**Location:** Portugal, Lisbon

**Date:** from:21-September-2000; to:21-September-2000

**URL:** http://www.ics.forth.gr/proj/isst/SemWeb/program.html

**Topics:** Semantic Web.

## *6.9 12th International Conference on Knowledge Engineering and Knowledge Management EKAW 2000*

EKAW\'2000 aims at gathering researchers working on any area concerning methods, techniques and tools for the construction and the exploitation of knowledge-intensive systems and for knowledge management.

**Location:** France, Juan Les Pins

**Date:** from:2-October-2000; to:6-October-2000

**URL:** http://www-sop.inria.fr/acacia/ekaw2000/

**Topics:** Cooperative Knowledge Acquisition, Evaluation of KA, KA methodologies, Knowledge Management, Ontologies, Ontology Learning, Semantic Web.

## *6.10 SemWeb'2001 - Second International Workshop on the Semantic Web*

The goal of the workshop is to share experiences about Semantic Web systems, exchange ideas about improvements of existing tools and creation of new systems, principles and applications. Also an important goal is to develop a cooperation model among Semantic Web developers, and to develop a common vision about the future developments.

**Location:** China, Hongkong

**Date:** from:1-May-2001; to:1-May-2001

**URL:** http://semanticweb2001.aifb.uni-karlsruhe.de/

**Topics:** Semantic Web.

## *6.11 IJCAI'01 Workshop on Ontologies and Information Sharing*

The goals of this workshop are twofold:

1) to go into detail in one particular application area: information sharing,

2) to continue the further understanding of the field in general from both theoretical and practical standpoints.

**Location:** USA, Seattle

**Date:** from:6-August-2001; to:6-August-2001

**URL:** http://www.semantic-translation.com/IJCAIwp/

**Topics:** Ontologies.

## *6.12 IJCAI 2001 Workshop on the IEEE Standard Upper Ontology*

This workshop will support the IEEE Standard Upper Ontology working group effort. This workshop aims to make progress toward the goal of producing a Standard Upper Ontology. The following describes the scope and purpose, as determined by the working group.

**Location:** Seattle, USA

**Date:** from:6-August-2001; to:6-August-2001

**URL:** http://workshop.ijcai-01.org/

**Topics:** Ontologies.

## *6.13 FOIS.ORG - Formal Ontology in Information Systems*

We envision FOIS to be a unique gathering whose essential character is strongly interdisciplinary, and truly unlike any other conference. The program committee includes representatives of three broad disciplines: computer and information science, philosophy, and linguistics, and includes researchers from sub-areas as diverse as formal ontology, knowledge engineering, logic, database design, natural language processing, library science, knowledge representation, descriptive metaphysics, and geographic information systems.

**Location:** USA, Ogunquit, Maine

**Date:** from:17-October-2001; to:19-October-2001

**URL:** http://www.fois.org/fois-2001/

**Topics:** Ontologies.

## *6.14 First International Conference on Knowledge Capture K-CAP 2001*

The aim of K-CAP 2001 is to provide a forum in which to bring together disparate research communities whose members are interested in efficiently capturing knowledge from a variety of sources and in creating representations that can be (or eventually can be) useful for reasoning. This conference will promote multidisciplinary research that could result in a new generation of tools and methodologies for knowledge capture.

**Location:** Canada, Victoria, B.C.

**Date:** from:21-October-2001; to:21-October-2001

**URL:** http://http://sern.ucalgary.ca/ksi/K-CAP/K-CAP2001/

**Topics:** Cooperative Knowledge Acquisition, Evaluation of KA, KA methodologies, Knowledge Elicitation, Knowledge Management, Ontologies, Ontology Learning, Semantic Web.

## *6.15 KCAP 01 Markup and Annotation Workshop*

The workshop intends to bring together researchers and practitioners from such research areas as the Semantic Web, knowledge acquisition, computational linguistics, document processing, terminology, information science, and multimedia content, among others, to discuss various aspects of knowledge markup and semantic annotation in an interdis ciplinary way.

**Location:** Canada, Victoria, B.C.

**Date:** from:21-October-2001; to:21-October-2001

**URL:** http://semannot2001.aifb.uni-karlsruhe.de/#objectives

**Topics:** *Not available.*

## *6.16 9th Spanish Association for Artificial Intelligence Conference*

The ninth edition of the Spanish Association for Artificial Intelligence Conference (CAEPIA), together with the fourth Technological Transfer of Artificial Intelligence Congress (TTIA), will take location at Gijón, November 14-16, 2001. This biennial meeting of these conferences is promoted by AEPIA to foment the diffusion of the Artificial Intelligence, and is a reference in the new paradigms and technologies of the Artificial Intelligence. The last editions of the CAEPIA were held at Alicante (1995), Málaga (1997) and Murcia (1999).

**Location:** Spain, Gijón

**Date:** from:14-November-2001; to:16-November-2001

**URL:** http://http://www.aic.uniovi.es/caepia.2001/

**Topics:** *Not available.*