*Technical Report*

UNIVERSIDAD DE LAS AMÉRICAS PUEBLA

Center for Research in Information and Automation Technologies

Interactive and Cooperative Technologies Lab

# An Overview of Ontologies

María Auxilio Medina Nieto

March 2003

*Ex. Hacienda Sta. Catarina Mártir s/n*
*Cholula, Puebla*

# An Overview of Ontologies

María Auxilio Medina Nieto

Center for Research in Information and Automation Technologies
Interactive and Cooperative Technologies Lab
Universidad De Las Américas Puebla – México
sp098988@mail.udlap.mx

**Abstract**

Ontologies emerged in Artificial Intelligence as an alternative to represent knowledge. Nowadays they are a topic of interest in diverse research areas and their usefulness can be perceived in a great number of applications. This paper is aimed to answer the kind of questions a reader could elaborate to acquire a general understanding of ontologies. To achieve this goal, it presents some definitions of ontologies and it distinguishes their main characteristics. Important aspects to build an ontology are mentioned followed by some applications briefly described. Tools and languages to support their management are included as well as some excerpts of relevant research projects.

## Contents

# 1. Introduction

The World Wide Web and digital libraries are enormous repositories which concentrate huge volumes of distributed heterogeneous data. The accessibility, maintenance, organization, preservation among other required tasks to manage these repositories as efficient as possible overcome human capabilities, in such a way that to carry them out, it is necessary the support of automatic processes or tools[1].

To achieve with this goal, an appropriated way to describe these repositories and their data machine understandable is required. Many alternatives have been developed, however, there is not a consensus to use one or a small set of them yet. Artificial Intelligence is a research area which has proposed important approaches as data structures, relational data bases, mathematic logic, procedures, taxonomies among others [Olivares 2002]. This paper analyzes one of these proposals termed ontologies.

Ontologies emerged as an alternative to represent knowledge. However, they have been used to support a great variety of tasks. At present, there are applications of ontologies with commercial, industrial, academical or research focuses.

People, organizations and software programs must communicate, although different needs and backgrounds imply different viewpoints. This divergence is natural and valuable, but it leads to problems in communication, interaction and understanding [Faquhar 1997]. Ontologies are an alternative to address these kind of problems. The main objective of this paper is to provide readers of a general understanding of ontologies with the next specific objectives:

- To identify main characteristics of ontologies, their relevance and some of their applications.

- To review tools and languages to support ontologies.

- To look over relevant research projects.

To provide of a general understanding of ontologies, the content of this report aims to answer questions as the following ones: what are ontologies?, what are their characteristics?, what are they for?, how they address semantical and interoperability problems?, what tools or languages support their management?, how they have been used?.

The paper is organized as follows: definitions of ontologies are presented next. Then, main characteristics and relevant applications are distinguished. Tools and languages to support ontologies management are briefly described and finally excerpts of relevant research projects of ontologies are supplied.

---

[1] An excellent example of a well known research project is the semantic web project which can be briefly described as a web of information that agents can process.

## 2. What are ontologies?

It is possible to find in the literature several definitions of ontologies. One of the most cited is the next one proposed by Gruber, [Gruber 1993]: a*n ontology is a formal, explicit specification of a shared conceptualization.* [Fensel 2001] analyzes this definition identifying four main concepts involved: an abstract model of a phenomenon termed "*conceptualization"*, a precise mathematical description hints the word "*formal"*, the precision of concepts and their relationships clearly defined are expressed by the term "*explicit"*" and the existence of an agreement between ontology users is hinted by the term "*shared"* [Fensel 2001].

The definition proposed by Gruber is general, however ontologies can be defined in specific contexts. For example, taking the paradigm of agents into account, [Russell & Norving 1995] establish that a*n ontology is a formal description of the concepts and relations which can exist in a community of agents*.

The importance of the terms of an ontology can be perceived in the next definition: *an ontology is a hierarchically structured set of terms to describe a domain that can be used as a skeletal foundation for a knowledge base* [Swartout et al. 1996]. More recent definitions of ontologies are the following ones:

An ontology is a common, shared and formal description of important concepts in an specific domain [Fensel 2000].

An ontology is a formal explicit representation of concepts in a domain, properties of each concept describes characteristics and attributes of the concept known as slots and constrains on these slots [Noy & McGuinness 2001]. Sometimes concepts are termed classes, properties are also known as roles while facets are used rather than slots.

An ontology is a theory which uses a specific vocabulary to describe entities, classes, properties and related functions with certain point of view [Fonseca et al. 2002].

An ontology necessarily includes a specification of the terms used, ("terminology") and agreements to determine the meaning of these terms, along with the relationships between them [Starlab 2003].

From these definitions, we can identify some essential aspects of ontologies:

– Ontologies are used to describe a specific domain.

– The terms and relations are clearly defined in that domain.

– There is a mechanism to organize the terms, (commonly a hierarchical structure is used as well as IS–A or HAS–A relationships).

– There is an agreement between users of an ontology in such a way the meaning of the terms is used consistently.

Figure 1 shows a graphical representation of an ontology for documents.

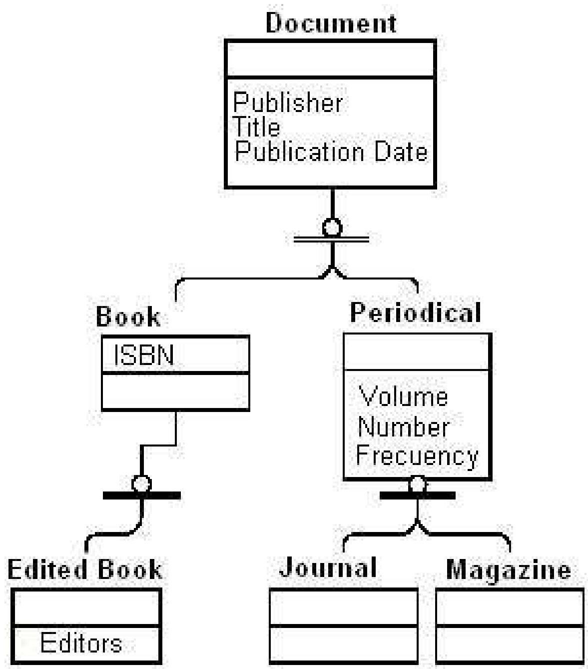


Figure 1. An ontology for documents, published in [Faquhar 1997].

[Farquhar et al. 1997] perceive an ontology as an artifact which must be constructed, structured and manipulated. There are some common artifacts with ontological characteristics such as a glossary, a thesaurus, a dictionary, an encyclopedia, a data dictionary, a class library, a database schema or a knowledge base. Figure 2 shows an excerpt of an ontology. It is written using SHOE language, (Simple HTML Ontology Extensions), which is an XML–compatible knowledge representation language for the web.

| | |
|---|---|
| <ONTOLOGY ID="*Department*" VERSION ="*1.1*"> | **ONTOLOGY NAME** |
| <DEF–CATEGORY NAME="*Person*" ISA = "*base.SHOEEntity*"><br><DEF–CATEGORY NAME="*Worker*" ISA ="*Person*"><br><DEF–CATEGORY NAME="*Teacher*" ISA ="*Worker*"><br><DEF–CATEGORY NAME="*Student*" ISA ="*Person*"> | **CATEGORIES** |
| <DEF–RELATION NAME="*Tutor of*"><br><DEF–ARG POS="*1*" TYPE="*Teacher*"><br><DEF–ARG POS="*2*" TYPE="*Student*"><br></DEF–RELATION> | **RELATIONSHIPS** |

Figure 2. An excerpt of an ontology using SHOE language.

## 3. Main functions of ontologies

Ontologies can be used to support a great variety of tasks in diverse research areas such as knowledge representation, natural language processing, information retrieval, databases, knowledge management, on line database integration, digital libraries, geographic information systems, visual information retrieval or multi agent systems. This section presents some of the most relevant and general functions of ontologies.

An ontology provides meta information which describes data semantics [Fensel 2001]. Ontologies enable shared knowledge and reuse where information resources can be communicated between human or software agents. Semantical relationships in ontologies are machine readable, in such a way they enable making statements and asking queries about a subject domain due to the use of a conceptualization, which describes entities and their relationships. This conceptualization enables that software agents of a vocabulary to represent and to communicate knowledge. The usefulness of ontologies in agent based systems can be briefly summarized as they enable knowledge−level interoperation. In other research areas, ontologies support shared understanding, interoperability between tools, systems engineering, reusability and declarative specification [Farquhar et al. 1997].

On the other hand, ontologies are used to build knowledge bases, (according to [Noy & McGuinness 2001], a knowledge bases is formed by an ontology and a set of individual instances of its classes). Knowledge bases can be queried by agents in order to enrich, reuse and maintain them. Ontologies concentrate state−independent information while the core of knowledge bases is formed by state−dependent information [FIPA 2001].

Ontologies are able to operate as repositories to organize information for specific communities. They can be used as a tool for knowledge acquisition, (teamworks can use ontologies as a common support to classify the knowledge of an organization). Ontologies allow users to reuse knowledge in new systems. They can form a base to construct knowledge representation languages [Tramullas 2003].

Semantic integration of heterogeneous information sources such as digital libraries can benefit with the incorporation of ontologies. Some applications use a domain ontology to integrate information resources and others allow each resource to use its own ontology. Each user can also have his own ontology according to his/her interests, language or role in a determine domain. Ontologies provide a source of precisely defined terms [Sure et al. 2002].

In information retrieval applications, ontologies serve to disambiguate user queries, to elaborate taxonomies of terms or thesaurus in order to enhance the quality of retrieved results [FIPA 2001]. Machine−learning techniques are also used to extend ontologies based on users interactions.

## 4. Outlines to build ontologies

Ontologies are proposed by a team of researches who need to share information in a determine domain. Definitions of basic concepts and relationships between them are machine interpretable. A programmer makes design decisions based on the operational properties of a class, whereas an ontology designer makes these decisions based on the structural properties [Noy & McGuinness 2001].

To build an ontology is a similar process than to build an object oriented program, however, classes and objects in a program are about data structures, whereas classes and objects in ontologies are about the domain [Farquhar et al. 1997]. Besides the main functions of ontologies described in the last section, [Noy N.F., McGuinnes D. L. 2001] identify the next reasons someone could have to build an ontology:

– To share common understanding of the structure of information between people or software agents.

– To enable reuse of domain knowledge.

– To make domain assumptions explicit.

– To separate domain knowledge from operational knowledge.

– To analyze domain knowledge.

According to these researchers, the development of an ontology is similar to define a set of data and their structure for other programs to use. They suggest a methodology to develop ontologies for declarative frame–based systems. It includes phases as: the definition of classes, the organization of them in a hierarchical structure, the specification of slots and the assignment of valid values as well as the elaboration of classes.

Another approach to build an ontology is proposed by [Gómez 1998]. It follows the next set of principles:

– *Clarity and objectivity*: An ontology should provide precise and objective definitions in natural language form.

– *Completeness*: Definitions should be expressed by necessary and sufficient conditions.

– *Coherence*: An ontology should allow to make consistent inferences with definitions.

– *Maximal monotonic extendibility*: Inclusion of new general or specific terms should be included in the ontology without revision of existing definitions.

– *Minimal ontological commitment*: Few axioms as possible about the modeled world.

– *Ontological Distinction Principle*: Classes with different identity criteria must be disjoint.

Figure 3 shows some of the most relevant task required tasks to build an ontology. This is an adaptation by [Gómez 1998] of IEEE software development process.

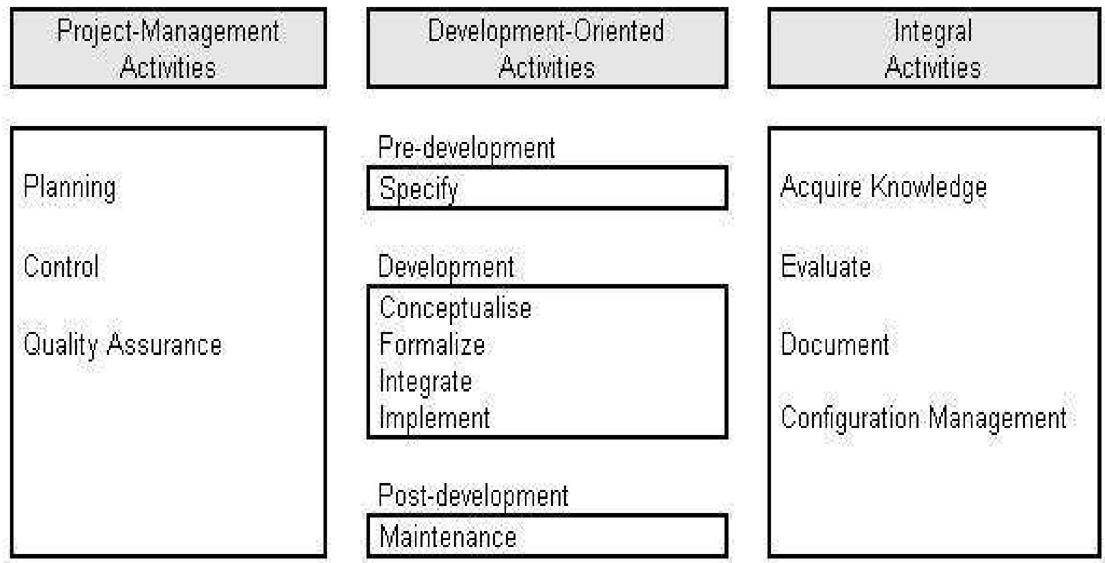


Figure 3. Some required tasks to build an ontology, published in [Gómez 1998].

A good practice to support ontology building process is to produce an ontology specification document written in natural language with information such as: the purpose of the ontology, its end users, use case scenarios, degree of formality used to codify the ontology and its scope. An ontology specification document should be conciseness, (it must include just relevant terms without duplicates), partial completeness and realism, (meanings of the terms and their relationships making sense in the domain), [FIPA 2001].

Some researches propose different kinds of ontologies taking several criteria into account as the formality of the language or the level of dependence on a particular task or point of view. [Guanaro 1998] considers the last one and identifies the next basic kinds of ontologies:

– *Top level ontologies* describe general concepts like space, time, matter, object, event or action, which do not depend of a particular problem or domain. However, the development of a general enough top level ontology has not been accomplished yet.

– *Domain ontologies* and *task ontologies* describe the vocabulary for a generic domain (like biology or medicine), a task or activity (such as selling) by means of specialized terms.

– *Application ontologies* describe concepts which depend on a particular domain and task. The concepts respond to roles played by domain entities while performing certain task.

To identify the kind of an ontology according to a particular classification, it is useful to lead the ontology building process.

## 5. Languages and tools to support ontologies management

[Farquhar et al. 1997] assert the best candidate for an unambiguous language to represent knowledge is First Order Logic (FOL), which can be presented as an object oriented model with the advantage of anything that does not fit in the object model, can still be said in FOL. However, he expresses that formal does not mean precise neither understandable. Several languages have been developed to support ontologies. Table 1 summarizes their main characteristics.

| ACRONYM | LANGUAGE | CHARACTERISTICS |
|---|---|---|
| **HTML** | Hyper Text Markup Language | Simplicity |
| **XML** | Extensible | Extensions for arbitrary domains and specific tasks. |
| **SHOE** | Simple HTML Ontology Extensions | It is a XML compatible knowledge representation language for the web. It allows page authors to annotate their web documents. It is not actively maintained. |
| **RDF** | Resource Description Framework | Syntactic conventions and simple data models to represent semantics. It supports interoperability aspects with object–attribute–value relationships. |
| **RDFS** | Resource Description Framework Schema | Primitives to model basic ontologies with RDF. |
| **OIL** | Ontology Inference Layer / Ontology Interchange Language | Primitives to model ontologies from frame–based languages, formal semantics and reasoning support based on descriptive logic, a proposal for syntatic interchange of annotations. It is compatible with RDF Schema [Fensel et al. 2000]. |
| **DAML** | DARPA Agent Markup Language | It is formed by DAML–ONT, (a language of ontologies) and DAML–Logic (a language able to express axioms and rules). It inherits many characteristics of OIL, however, it is less compatible with RDF than OIL. |
| **XSL** | Extensible Stylesheet Language | It provides a standard to describe mappings between different terminologies, (a translation mechanism between XML documents). |
| **XOL** | Ontology Exchange Language | Simplicity, a generic approach to define ontologies. It has two syntactical variants based on XML and RDF Schema. |

Table 1. Some languages to support ontologies management

Ontology editors are important tools to support the elaboration of ontologies. They facilitate development and management of ontologies, the definition and modification of concepts, properties, axioms and restrictions, even some of them enable inspection and browsing of ontologies. This section describes some of the most relevant ontology editors.

**Java Ontology Editor**, (abbreviated as **JOE**), supports the construction of entities, attributes and relations for ontologies. In **JOE**, an ontology is considered an entity–attributes relationship or a frame–slot model. Its goal is to provide a graphical interface to represent ontologies in open distributed environments. **JOE**[2] is an applet formed of two major components: an ontology editor and a query editor. It does not support more complex tasks of ontology management as versioning [Mahalingam 1996].

**Ontolingua**[3] is an ontology development environment for browsing, creating, editing modifying and using ontologies. It contains ontology authoring tools by assembling and extending ontologies extracted from a library of reusable ontologies. It combines axioms, definitions and non–logical symbols termed "*words"* from multiple ontologies [Fikes et al. 1997].

**Chimaera**[4] is a merging and diagnostic web–based browser ontology environment. It accepts over 15 designated input format choices such as ANSI Knowledge Interchange Format (KIF), Ontolingua, Protégé, CLASSIC, iXOL as well as any other OKBC–compliant form, (OKBC is an acronym for Open Knowledge Base Connectivity). It consists of an own editing environment, although users are able to employ Ontolingua, too.


Figure 4 shows the look and feel of Chimaera interface.

---

2 JOE is available at http://www.cse.sc.edu/research/cit/demos/java/joe/. You can evaluate JOE functionality at http://www.cse.sc.edu/research/cit/demos/java/joe/Survey.html.

3 Ontolingua is available at: http://www.ksl.stanford.edu/software/ontolingua/

4 Chimaera is available at: http://www.ksl.Stanford.EDU/software/chimaera. A tutorial to start using it is also available.

**Chimaera** has an analysis capability to make incompleteness tests, syntactic or semantic checks as well as taxonomic analysis. Some of the main tasks supported by **Chimaera** are: loading knowledge bases in different formats, reorganizing taxonomies, resolving name conflicts, browsing ontologies and editing terms [McGuinness et al. 2000].

Ontologies management is not a simple task due to their size and complexity. To address these problems, [Sure et al. 2001] propose **OntoEdit**, it is an editor designed to support collaborative development of ontologies, adaptation. It enables to import knowledge models for application systems. It combines methodology based ontology development with capabilities for collaboration and inferencing. It focuses in the next aspects to develop an ontology: specification or requirements, (to collect the requirements for the ontology), refinement, (semi–formal description of the ontology and its formalization using a representation language), and evaluation, (it involves the developers and users to prove ontology usefulness).

The data model of **OntoEdit** is OXML 2.0 which is frame based with meta–classes and predicates or axioms. **OntoEdit** supports F–Logic (Fuzzy Logic), RDF–Schema and OIL. **OntoEdit** interacts with other tools to form a structure called "*Semantic Value Chain"*. It is shown in Figure 5. It has a capable plug–in structure, a lexicon component and an ontology mapping plug–in. It was developed by the Knowledge Management Group of the AIFB Institute at the University of Karlsruhe.

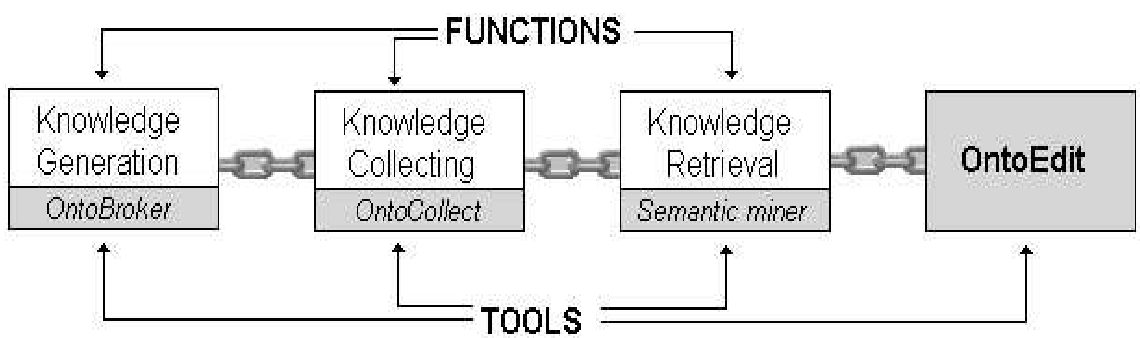


Figure 5. A semantic value chain in the context of OntoEdit[5].

**OilEd**[6] is a freely available editor for OIL and DAML–OIL developed by the University of Manchester. It is an ontology editor with a frame interface. It uses reasoning to support ontology design and maintenance. Its main advantage is the extension of a frame editor paradigm to deal with an expressive language and the incorporation of a descriptive logic reasoning engine which is used to check class consistency and inference relationships. It enables class definition, slots and axioms. In contrast to other frame systems, it facilitates developers the use of arbitrary boolean combinations of frames or classes connected by *and*, *or* and *not*. However, OilEed does not provide and support for versioning or with multiple ontologies [Bechhofer et al. 2001].

[5] Figure X is an adaptation of an image presented in OntoEdit home page, (http://www.ontoprise.de/products/ontoedit).

[6] OilED is available at: http://oiled.man.ac.uk/

**Protégé** is an open source Java tool. It allow users to construct a domain ontology, to customize knowledge−acquisition forms and to enter domain knowledge. It is able to operate as a platform to extend access to other knowledge based systems, embedded applications, or as a library which can be used by other applications to access and visualize knowledge bases. It has a a powerful tailorability and it enables knowledge acquisition. **Protégé**[7] offers a graphical user interface which allow ontology developers to focus on conceptual modeling without requiring to know syntax of an output language, such as RDFS or OIL [Noy et al. 2001].

## 6 Relevant Research Projects

Ontologies have been a research topic for more than a decade. This section presents some excerpts of relevant research projects and well known applications.

**Loom** is a research project in the Artificial Intelligence group at the University of Southern California's Information Sciences Institute. Its goal is to develop tools for knowledge representation and reasoning. A very interesting project is **Ontosaurus**[8], which is a web browser for Loom knowledge bases [Swartout 1996]. It is formed by an ontology server and ontology browser clients. **Ontosaurus** dynamically creates HTML pages which display the ontology hierarchy and it uses HTML forms to allow user to edit the ontology.

Due to **Ontosaurus** incorporates **Loom**, it has reasoning capabilities and it provides users of a richer collection of semantic services, tests for consistency and compatibility of new concepts. It offers full access to ontologies, on−line editing and tools for translating and managing ontologies. The ontology development approach of **Ontosaurus** is the development of a comprehensive linguistically ontology to provide a common organizational framework to develop a knowledge base. This structure can be enriched with domain terms and facts to generate rich ontologies. One of the most recent versions of **Ontosaurus** is shown in figure 6.

**Ontolingua**[9] is an ontology server to support collaborative construction of ontologies. It uses Ontolingua language proposed previously by Gruber [Gruber 1993]. The server extends the original language providing explicit support to build ontological modules which can be assembled, extended and refined. It also establishes a explicit separation between ontology's presentation (by KIF axioms) and ontology's representation. **Ontolingua** server allows users to publish, browse, create and edit ontologies. A network protocol and an application program interface enable remote applications to use **Ontolingua**. Users can build new ontologies from a library of models as well as to extend or to restrict definitions. It enables reuse on ontologies from a modular structured library by inclusion, polymorphic refinement and restriction [Farquhar et al. 1997].

---

7 Protégé is available at: http://protege.stanford.edu/
8 Ontosaurus is available at: http://mozart.isi.edu:8003/sensus2/.
9 Ontolingua is available at: http://ontolingua.standord.edu

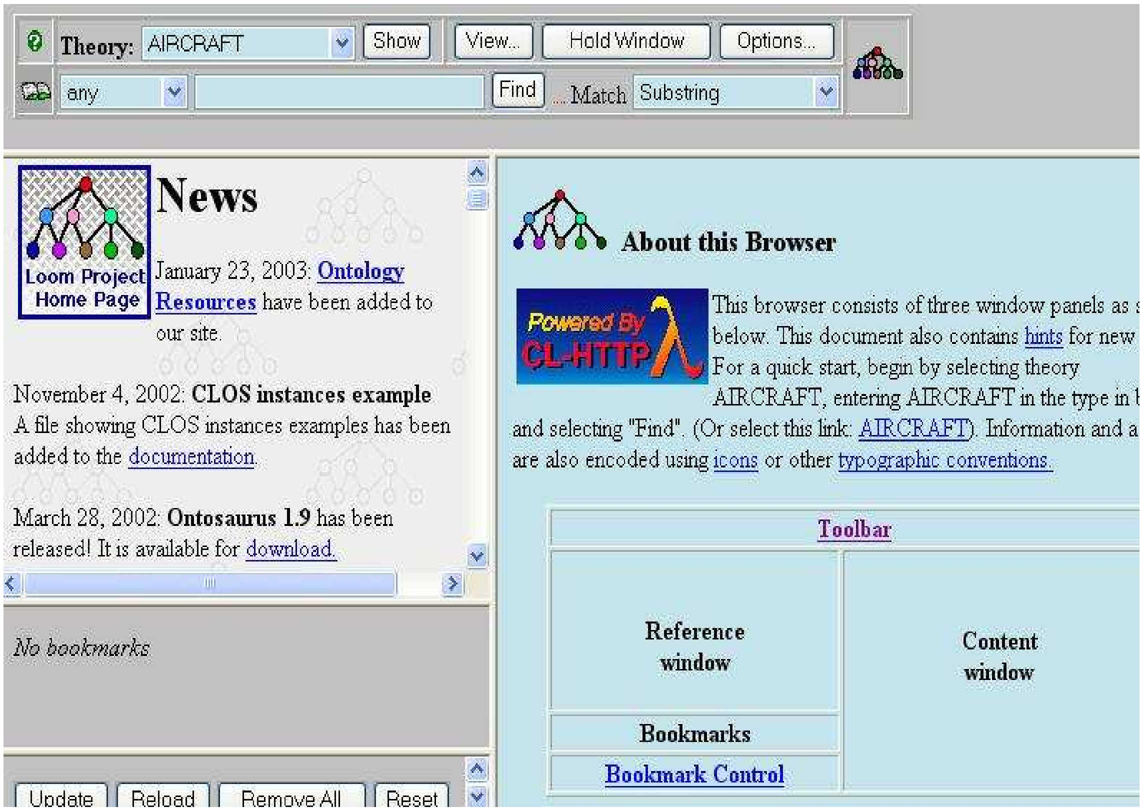


Figure 6. A snapshot of Ontosaurus system.

**Ontology Service Specification** is a research project which defines interaction protocols, communicative acts and a vocabulary which agents must adopt to use this service. Ontologies give users information of the vocabulary available to interact with the system and the meanings of the terms of the vocabulary in that domain [Farquhar et al. 1997].

**(ONTO)²Agent** is an ontology based web broker about ontologies. It uses a domain ontology about ontologies termed "*Reference Ontology",* it includes descriptions and links to existing ontologies. Reference Ontology deals with the problem of the dispersion of ontologies over several servers and the absence of common formats to represent relevant information about ontologies with the same logical organization [Arpírez et al. 1998]. **(ONTO)²Agent** emerges as an alternative to support developers when searching ontologies due to problems such as ontology content formalization depends on the server at which it is stored or ontologies on the same server are described with different detail levels. The architecture of **(ONTO)²Agent** uses a SQL database to formalize an ontology, a web form and an ontology generator to store knowledge and trees to retrieve information. It is web accessible [10].

On the other hand, the goal of **IBROW** project is to develop a broker to access

[10] ONTO²Agent is available at: http://delicias.dia.fi.upm.es/OntoAgent/

dynamic reasoning services in web. Intelligent brokers are required to configure reusable components in knowledge systems. This projects moves work on inference services in the direction of multi−agent systems. An important subject in this project is a matchmaking between user requests on the side and competence descriptions of available agents at the other, as well as delegation of tasks to heterogeneous agent societies [Fensel & Motta 1998].

**Ontobroker** relies on the use of ontologies to annotate web pages, formulate queries and derive answers. It provides of a broker architecture with three core elements: a query interface for formulating queries, an inference engine used to derive answers, and a webcrawler used to collect the required knowledge from the web [Decker 1999]. A more recent version of **Ontobroker** is known as **On2broker**.

[Buckingham et al. 2000] describe the design of an ontology based digital library server designed to support scholarly interpretation and discourse. It is called **ScholOnto** and pretends to add a semantic layer to conventional meta data. It will allow researchers to describe or debate contributions of documents via a semantic network. **ScholOnto** will provide users with a set of argumentation links for concepts or claims, they are also called semantic links. It is planed that **ScholOnto** will use the OCML language, (Operational Conceptual Modeling Language), which supports the construction of knowledge models, OCML enables the specification of functions, relations, classes, instances, rules. It has mechanisms to define ontologies and problem solving methods. **ScholOnto** focuses in the separation of content and structure in order to express competing claims and discourses.

FIPA, (Foundation for Intelligent Physical Agents) fosters agent−based applications and provides specifications to support interoperability between agents. In particular, it has a specification to manage ontologies provided by an **Ontology Agent** (abbreviated as **OA**). Some of the services of an **OA** are: searching and accessing public ontologies, maintaining a set of ontologies and a translation mechanism between ontologies. It is able to answer queries about relationships between terms or ontologies. At FIPA, it is not mandated that every **OA** will be able to offer these services, instead, every **OA** must establish a communication about them. The implementation of these services is left to developers [FIPA 2001].

One of the biggest research project involving ontologies is the **Semantic Web**. It can be briefly described as an XML application to provide intelligent access to heterogeneous and distributed information. In this project, agents operate as mediators between user needs and available information resources [Fensel 2000]. Tim Berners−Lee vision of **Semantic Web** project is a web where resources are also available to automated processes (agents) which could perform useful tasks such as an improved search (in terms of precision), resource discovery, information brokering and filtering [Berners−Lee 1999]. To reach this vision, it is necessary to use standard annotation languages. Proposals of some languages have emerged as XML, XML−Schema, RDF (Resource Description Framework) or RDF Schema. [Bechhofer et al. 2001].

**Ontoknowledge** is a running project under the 5th European Framework program. It will provide information access in digital networks. It is designed to support efficient and effective knowledge management. It focuses on acquiring, maintaining, and accessing weakly−structured on−line information sources. **Ontoknowledge** develops a three layered architecture for information access: at the lowest level, (*information level*), weakly−structured information sources are processed to extract machine−processable meta−information. The intermediate level, (*representation level*) uses this meta−information to provide automatic access, creation and maintenance of these information sources, and the highest level, (*access level*), uses agent−based techniques as well as state−of the art querying and visualization techniques that fully employ formal annotations to guide users to access to the information [Fensel et al 2002].

At the **STARLab Systems Technology and Applications Research Laboratory**[11], [Starlab 2003], relevant projects related with ontologies are being implemented. Table 2 summarizes some of them.

| PROJECT | NAME | DESCRIPTION |
|---|---|---|
| OGMA | Developing Ontology−Guided Mediation of Applications | Linking ontologies, database semantics, information systems, corporate knowledge management, agent technology, Internet databases, the semantic web. |
| SCOP | Semantic Connection of Ontologies to Patient record data | Heterogeneity of data sources make propagation of medical information difficult. An alternative to face this problem is on the meaning level of the data bases. |
| FF POIROT | Financial Fraud Prevention−Oriented Information Resources using Ontology Technology | This project aims at compiling for several languages (Dutch, Italian, French and English) a computationally tractable and sharable knowledge repository for a financial domain. |
| OntoWeb | Ontology−based Information Exchange for Knowledge Management and Electronic Commerce | Its goal is to offer researcher and industrials ontological services to improve information exchange in areas such as information retrieval, knowledge management, bioinformatics or electronic commerce. |
| BonaTema | Development of an ontology−based knowledge system containing biological information on yeast | It aims the development of an ontology−based knowledge system with biological information on yeast to efficiently retrieve its biological information. |
| IrisWeb | Ontology−supported ways to perform browsing in a (digital) library | This project illustrates some possibilities to emulate a richer experience based on research ontology−supported ways in a digital library. Vlaamse Gemmenschapscommissie and Hoofdstedelijke Openbare Bibliotheek Brussel cooperate in this project, too. |

Table 2. Ontology projects at STARLab.

[11] STARLab home page is available at: http://www.starlab.vub.ac.be/research/index.htm

## 7 Conclusions

"It is perhaps enlightening to see how ontologies in a sense may achieve a form of *semantics independence* for information and knowledge based systems: just as database schema achieved data independence by making the specification and management of stored data elements external to application programs, ontologies now will allow to specify and manage domain semantics external to those programs as well [Starlab 2003]".

Ontologies provide a set of well defined termed avoiding disambiguation in the communication between software or human agents. They are an alternative to represent knowledge. Although ontologies emerged in Artificial Intelligence, they have been case of study in diverse research areas. The applications presented here are not very specific in order to readers can acquire a general understanding of ontologies.

Some languages and tools have been developed to support ontologies management. This paper has just presented relevant aspects of them to focus attention in the main ideas and relevant concepts related with ontologies. However, footnotes and references are a valuable source of information to get more details.

Although the research on ontologies has been carried out for more than a decade, there is still a lot of work to do. In particular, the next step of this research work will be focus on the design and implementation of autonomous process termed ontological agents which will be used in a federation of digital libraries to address information retrieval and semantic interoperability problems.

## Acknowledgments

This research is partly supported by the Mexican National Council of Science and Technology (Conacyt Projects No. 35804−A and G33009−A).

## References

Arpírez, J.; Gómez−Pérez, A.; Lozano, A.; Pinto, S.H. 1998. $(ONTO)^2$Agent: an Ontology−based WWW broker to Select Ontologies Workshop on Application of Ontologies and Problems Solving Methods. ECAI'98. Brighton, ECAI'98. $(ONTO)^2$Agent is available at:
http://delicias.dia.fi.upm.es/REFERENCE_ONTOLOGY/

Berners−Lee T. 1999. *Weaving the web*. Orion Business Books.

Bechhofer S., Ian Horrocks, Carole Goble, Robert Stevents. 2001. OilED: A Reaon−able Ontology Editor for the Semantic Web. Lecture Notes in Computer Science.

Buckingham S., Motta E. Domingue J. 2000. ScholOnto: an ontology−based digital library server for research documents and discurse. International Journal on Digital Libraries, 3(3) (August/September, 2000), Springer−Verlag.

Decker, S. Erdmann. M. Fensel, D. Studer R. 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

Farquhar, A. Fikes R. Rice J. 1997. The Ontolingua Server: A tool for collaborative ontology construction. International Journal of Human−Computer Studies, 46,(6):707−727.

Fensel D., 2000. The semantic web and its languages. *IEEE Computer Society 15,* 6 (November /December), 67−73.

Fensel D., Horrocks I., Harmelen F., Decker S., Erdmann M., Klein M. 2000. OIL in a nutshell. In *Proceedings of the European Knowledge Acquisition Conference,* (EKAW−2000), R. Dieng et al. (eds), Lecture Notes in Artificial Intelligenc, LNAI, Springer−Verlag, October.

Fensel D., 2001. *Ontologies: A Silver Bullet for Knowledge Management and Electronic Commerce.* Springer.

Fensel, D. and Motta, E. 1998. Structured Development of Problem Solving Methods. In *Proceedings of the 11th Workshop on Knowledge Acquisition, Modeling and Management* (KAW´98). Banff, Canada, April 18th – 23rd.

Fikes R., Farquhar A, Rice J. 1997. Tools for Assembling Modular Ontologies in Ontolingua. Knowledge Systems Laboratory. Available at: http://www.ksl.stanford.edu/software/ontolingua/project−papers.html

FIPA 2001. Foundation for Intelligent Physical Agents.FIPA Ontology Service Specification. March. Available on: http://www.fipa.org/specs/fipa00086/XC00086D.html

Fonseca, F. Egenhofer M., Agouris, P., Camara G. 2002. Using Ontologies for Integrated Geographic Information Systems. Transactions in GIS, −(6):3 in print.

Horrocks I., Stevens R. 2001. OilEd: a Reason−able Ontology Editor for the Semantic Web. Proceedings of Joint German/Austrian conference on Artificial Intelligence, September 19−21, Vienna. Springer−Verlag LNAI Vol. 2174, pp 396−408.

Noy N.F., McGuinnes D. L. 2001. ‘‘Ontology Development 101: A Guide to Creating Your First Ontology’’. Stanford Knowledge Systems Laboratory Technical Report KSL−01−05 and Stanford Medical Informatics Technical Report SMI−2001−0880, March.

Gómez−Pérez, A.. 1998. Knowledge Sharing and Reuse In The handbook on Applied Expert Systems. By Liebowitz. ED CRC Press. 1998.

Guarino, N. 1998. Formal Ontology in Information Systems. In N. Guarino (ed.) *Formal Ontology in Information Systems. Proceedings of FOIS’98, Trento, Italy, 6−8 June 1998.* IOS Press, Amsterdam: 3−15.

Gruber T.R. 1993. A Translation Approach to Portable Ontology Specifications, *Knowledge Acquisition*, 5:199−220.

Mahalingam K. 1996. "The Java Ontology Editor (JOE)", Technical Report. Center for Information Technology, Electrical anc Computer Engineering Department, University of South Carolina.

McGuinness, D.L., Fikes R., Rice J. Wilder S. 2000. "The Chimaera Ontology Environment". In *Proceeding of the Seventeenth National Conference on Artificial Intelligence* (AAAI 2000, Austin Texas, July 30–August 3).

Motta, E. 1998. An Overview of the OCML Modelling Language. In *Proceedings of the Eighth Workshopon Knowledge Engineering Methods & Languages* (KEML '98, Karlsruhe, Germany, Jan. 21–22), 1998.

Noy F. N., McGuinness D. L. 2001. Ontology Development 101: A Guide to Creating Your First Ontology. Stanford Knowledge Systems Laboratory Technical Report KSL–01–05 and Stanford Medical Informatics Technical Report SMI–2001–0880, March.

Noy N., Sintek M., Decker S., Crubezy M., Fergerson W., Musen M. 2001. Creating Semantic Web contents with Protégé–2000. IEEE Intelligent Systems 16(2):60–71.

Olivares C. 2002. Un modelo de Interacción entre agentes con propósito, ontologías mixtas y eventos inesperados. Tesis de Doctorado. Centro de Investigación en Computación. Instituto Politécnico Nacional.

Starlab 2003. Systems Technology and Applications Research Laboratory home page. Faculty of Sciences, Department of Computer Science, Vrije Universiteit Brussel. Available at: http://www.starlab.vub.ac.be/default.htm

Russell S., Norvig P. 1995. *Artificial Intelligence: A Modern Approach.* Prentice Hall, Englewood Cliffs, NJ.

Sure Y. , Erdmann M, Angele J., Staab S., Studer R., Wenke D. 2002. OntoEdit: Collaborative Ontology Development for the Semantic Web. In *Proceedings of the 1st International Semantic Web Conference – ISWC2002*, Springer, LNCS.

Swartout B., Patil R., Knight K., Russ T. 1996. Toward distributed use of large–scale ontologies. In *Proceedings of the Tenth Knowledge Acquisition for Knowledge– Based Systems Workshop.* (KAW '96 November 9–14, Banff, Alberta, Canada).

Tramullas, J. 2003. Material from Profesor Jesús Tramullas Saz, Department of Documentation, Universidad de Zaragoza, Spain. Available at: http://www/tramullas.com