Ontology

"The specification of a shared conceptualization"

A Review Document

Table of contents:

1. Introduction
2. Definition of Ontology
3. Roles of Ontologies
4. Types of Ontologies
5. Ontology Creation and Design
6. Ontologies and semantic information retrieval
7. Implementation languages
8. Summary
9. References
10. Appendix A – Basic Indices

Introduction

One classical dictionary definition of Ontology may be: *"The branch of metaphysics that deals with the nature of being"*. It begins with Aristotle attempt to classify things in the world.

The term “Ontology” has been introduced to the information sciences and research fields during the 1990’s by several ***Artificial Intelligence (AI)*** research communities. AI researchers adopted the term “Ontology” mainly to describe what they though would be (from the stand point of computational aspects) a proper representation of the world in a program code. It has recently been used in several other information technology fields such as intelligent information integration, information retrieval on the Internet, and knowledge management.

Ontologies are of basic interest in many different fields, largely due to what they promise: a shared and common understanding of some domain that can be the basis for communication ground across the gaps between people and computers. They (Ontology approaches) allow for sharing and reuse of knowledge bodies in computational form. As many traditional activities are changing their manner in the world of today due to the availability of information brought by the World-Wide-Web (WWW), Ontologies are likely to change more when the knowledge is structured in machine readable way, and the abstracts concepts it contains are shared (See Semantic Web for more information in this direction) .

This document attepts to brings a short and only a brief survey of the way experts define Ontology, the different types of Ontologies in use for various areas and how they may be applied to different fields. A special focus is given in the following on the issue of **Information Retrieval** (IR), and how presently used IR methods may interact with Ontology based concepts. Finally, a short list of computer languages, recently aknowladged in the context of creating an Ontology structure, or performing semantic query based on Ontologies, is given.

This document is arranged with its main body offering a general description of each of the issues discussed, along with hyper-links to some specific examples found either on internal documents or the WWW. Some other documents are in their original form and some have undergone some processing in order to focus on the relevant issues.

Definition of Ontology

Of the many definitions have aroused for Ontology the following is recommended by [**(Gruber, 1993; Borst, 1997)** and http://www.cmswiki.com]

> *An Ontology is a formal, explicit specification of a shared conceptualization.*

A "conceptualization" is an abstract model of a phenomenon, created by identification of the relevant concepts of the phenomenon. The concepts, the relations between them and the constraints on their use are explicitly defined. "Formal" means that Ontology is machine-readable and excludes the use of natural languages. For example, in medical domains, the concepts are diseases and symptoms, the relations between them are causal and a constraint is that a disease cannot cause itself.
That an Ontology is a "shared conceptualization" states that Ontologies aim to represent consensual knowledge intended for the use of a group. Ideally the Ontology captures knowledge independently of its use and in a way that can be shared universally, but practically different tasks and uses call for different representations of the knowledge in an Ontology.

Ontology is sometimes confused with *taxonomy*, which is a classification of the data in a domain. The difference between them is in two important contexts:

1. An Ontology has a richer internal structure as it includes relations and constraints between the concepts.
2. An Ontology claims to represent a certain consensus about the knowledge in the domain. This consensus is among the intended users of the knowledge, e.g. doctors using a hospital Ontology regarding a certain disease, artists relating to historical art and so on.

Because Ontologies aim to represent a form of common agreement regarding the knowledge they represent, they are often created in a cooperative process involving different people, sometimes at different places. Ontologies are divided to types in accord with the degree of generality of the principles they contain.

Roles of Ontologies

As stated above, Ontologies play a dominant roles in a growing number of different fields. A few general examples may be reviewed here:

In ***natural-language applications***, Ontologies are used for:

- Natural-language processing (Generalised Upper Model, SENSUS Ontology)
- Automatic extraction of knowledge from scientific texts (tried in the Plinius Ontology)
- Wordnet is one of the largest lexical Ontologies.

In the **database and information retrieval** areas, Ontologies are used for:

- Improving the process of retrieval. More on this issue is found in Ontologies and semantic information retrieval, and also below.
- Solving the problem of heterogeneous information sources that utilize different representations. Mapping each source's data scheme to the Ontology allows the user a unified view of the information regardless of its actual source. The HIDE hydrological tool is an example for such a use, and the InfoMaster system may be viewed as a restricted form of Ontology. SEMEDA is an application to molecular biology data and a good example of the considerations behind choosing an Ontology which best represents the knowledge at hand.

A possible application of Ontologies is **intra-organization communication** and **knowledge management**. Providing terms, relations and constraints, an Ontology is equipped to allow an accurate and common means of communications between organization members.
In a similar setting, management of "corporate memory" – the essentially important knowledge body that companies and organizations possess and rely on for reaching correct decisions – is a rapidly developing field. These Memories are likely to evolve to large collections of diverse knowledge patterns, making access and distribution of the knowledge difficult. Ontologies may be of great use in structuring and defining the knowledge, and in supporting extraction of relevant elements.
An good example is the **Tool for Clinical Data** which presents a system for clinical-data management designed for hospitals. Other commercially successful implementations are found in different corners of the Knowledge Management field.

The are of **knowledge engineering** is concerned with methodical building of large scale Knowledge Based Systems which have at least two basic components: domain knowledge and problem-solving knowledge. Ontologies are mainly used to analyze, model and implement the domain knowledge, but also affect problem-solving knowledge. How different types of Ontologies take part in the process of domain model construction is addressed in Types of Ontologies below.

Types of Ontologies

There are two main classes of Ontologies: the first would be the one that is employed to explicitly capture "static knowledge" about a domain, in contrast to Ontologies (the second) that provide a reasoning point of view about the domain knowledge (problem solving knowledge).

In the first class a distinction between types is made on the basis of the level of generality, as summarized in the table below:

1. Domain Ontologies	Designed to Represent knowledge relevant to a certain domain type, e.g. medical, mechanical etc.
2. Generic Ontologies	Can be applied to a variety of domain types. Mereology (Part-Whole theory) Ontologies are applicable to many technical domains. Also called "super theory" and "core technology". Core technology article.
3. Representational Ontologies	These formulate general representation entities without defining what should be represented. The Frame Ontology is a well known example.

For the problem solving knowledge class, two types may be found:

1. Task Ontologies	Provide terms specific for particular *Tasks.*
2. Method Ontologies	Provide terms specific to particular *Problem Solving Methods.*

Methods and tasks are two distinct terms in knowledge engineering. A task refers to a type of problem while a method is a means of solving the problem. Thus, a task may be associated with several different methods which are in turn composed of subtasks. For more information in this regard the reader is referred to **Knowledge Engineering: Principles and methods**.

Another type of Ontology is the *Application Ontology*. The Application Ontology is a combination of the Domain and Method Ontologies that includes all the knowledge – static and problem solving – needed for the modeling of a particular domain.

Creation and Design of Ontologies

There are basically two ways to create an Ontology. The first and the most obvious one is to build an Ontology from "scratch", i.e. to define classes, relations instants and so on. Examples may be found in **Tool for Clinical Data**, **Cultural Knowledge Database** and **Creating a Protein Ontology**.

An interesting attempt at semi-automatic Ontology creation is described in **Extracting Ontologies.** The approach presented is based on the fact that the web site services share similar structures and functionality as the underlying software. They apply a grammar-sensitive tool to recognize verb-noun pairs in the software documentation that are suspected to have a similar expression as the related action (e.g. "Get-Username" or "Delete-Entry"), and after applying certain filters, Ontology concepts are attributed to the most significant pairs. Thus, the Ontology engineer has a "head start" in creating the Ontology. Some general terms defined in that paper which relate to retrieval success and Ontology overlap are summarized in Appendix A.

The second way is to combine available Ontologies in several forms. The most frequently used forms are:

- **Inclusion** of one Ontology into another. The result is that the classes, relations and axioms of both Ontologies are found in the unified Ontology. Name conflicts are likely to rise and must be resolved, either manually or using an Ontology engineering tool.
- **Restriction**: An Ontology is applied to a restricted subset of what it was originally intended for. A simple example is the combination of an Ontology of rules for dealing with real numbers with an Ontology for integer number arithmetic. The rule '+' as may be defined in the first Ontology is only applied on the subset of integers.
- **Refinement**: General Ontologies sometimes require refinement in order to be applicable to specific needs. The KACTUS project (**http://hcs.science.uva.nl/projects/NewKACTUS/home.html**) was concerned with constructing large Ontologies for technical devices through incremental refinement of general Ontologies into technical Ontologies.

Whatever way an Ontology is created, some design principles are mentioned in related texts in order to optimize its use:

- **Modularity** – Small units make understanding the structure and reuse easier.
- **Internal coherence** – self consistency of the structural conceptuality system
- **Extensibility** – Ontologies are often enhanced by adding single concepts or classes, as a necessary evolution. Easing this process should be a design objective.
- Centre definitions on **natural categories** – Makes it simple and user friendly.
- Minimal Ontological commitment.

The principle of minimal commitment brings again the generality issue. The less there is Ontological commitment, the more general the Ontology it is, and therefore easier to reuse. However, the usability – reusability trade-off exists, and generalizing the Ontology beyond a certain point may reduce its effectiveness in representing the knowledge it was intended to represent. Thus, perhaps a better choice of words is "Optimal Ontological Commitment", when all parties understand the merits of generalization.

A detailed guide to the basics of Ontology designing may be found found in: http://protege.stanford.edu/publications/Ontology_development/Ontology101-noy-mcguinness.html

Ontologies and semantic information retrieval

The accelerated processes of digitalization and globally connected databases sprouting that are occurring in recent years have changed the focus of the information problems. It is no longer difficult to find information and gain knowledge about a certain topic, but rather to select from the huge heap of information the most relevant elements only. Search engine traditionally utilize a **syntactic** approach, searching for keywords, and performing operations on their abundance in order to rank the information elements. These methods suffer from problems such as vocabulary inconsistency – a situation in which a certain information object contains relevant information but is not retrieved because it uses different words to describe it – and its "opposite" in which irrelevant information is retrieved due to similarity of words.
Lately however, a new approach is emerging – the **semantic** approach. This approach aims to use meta-data – data about data – in order to answer the users' requirement in a more satisfactory way for Data Retrieval and navigation
.
Ontologies can be very useful in improving the process in two ways:

1. It allows to abstract the information and represent it explicitly- highlighting the concepts and relations and not the words used to describe them.
2. Ontologies can possess *inference functions*, allowing more intelligent retrieval. For example a "basketball player" is also a "professional athlete", and an Ontology that defines the relations between these concepts can retrieve one when the other is queried.

If Ontologies are to satisfy the demands of information retrieval (IR) needs, then, a large number of detailed Ontologies must be created, and methods for **semi-automatic and automatic Ontology creation** are heavily researched.

IR & AI (see hyperlink) gives a summary on how Ontologies and Ontology-based methods may interact with traditional IR methods. Specifically, **Co-Occurrence Theory** utilized in keyword-based searches may be applied to the semi-automatic creation of Ontologies. The basic idea behind the theory is that words that co-occur often, have a strong relation between them and the relation between the two concepts embodied in these words may be weighted by their co-occurrence statistics. In the specific example the **Salton Index**, an important measure of co-occurrence which is not biased by naturally high occurrence of certain keywords, was used as the weights for concept relations.

Extracting Ontologies, as mentioned before, describes an attempt to use the already available text documentation in software programming interface (API) codes in order to automatically create domain Ontologies. Here, the authors utilize a grammatically sensitive system which can locate verb-noun pairs in the documentation that hypothetically resemble an action performed by the software. For example "Add-Data" and "Get-Username" are two such verb-noun pairs. After filtration of significant pairs, a domain Ontology may be built based on these significant pairs as concepts, manually at first but semi-automatically ideally. Some **basic indices** for assessing the success of such a retrieval process, and very generally any retrieval process, are brought in the article and summarized in Appendix A.

The other side of the coin is, of course, actually retrieving data, information and knowledge using Ontologies. Two examples of such projects are the **SHOE** (click here

for website and here for descriptive PDF article) and the Ontobroker projects. The SHOE method adds Ontological annotation to existent web pages and thus when a user fills a query he/she is presented with available contexts making the search much more specific. Ontobroker also introduced an inference engine that is able to perform useful functions on the formal semantics, after several translation steps. Such inference functions allow retrieving subclasses of a queried concept and making implicit information more explicit and accesable.

Retrieving precise information is naturally requires a certain degree or means for ranking the relevance of information available at each source. Ontology-Based Information Retrieval Model deals with this issue together with the issue of incomplete knowledge bases (KBs). The approach adopted there assumes that any semantic retrieval algorithm must be robust enough to deal with the incompleteness of KBs and Ontologies as they are not yet sufficiently developed to rely upon absolutely, and so they suggest incorporating a keyword based "Plan B" retrieval method in the overall system. Their semantic ranking method, brought here as an example for such methods is usually based on two steps:

1. Annotating documents with a *weighted* annotation, descriptive of the importance of the annotated instance is to the text. So, each instance is thus weighted:

$$d_i = \frac{freq_{d,i}}{\max_k \{freq_{d,k}\}} \times \log \frac{N}{n_i}$$

d_i - The weight of instance *i* in document *d*.
$freq_{d,i}$ - number of occurrences of *i* in *d*.
$max_k\{freq_{d,k}\}$ - Maximum number of occurrences of any instance in *d*.
N - Overall number of documents in the search space.
n_i - number of documents annotated with *i*.

The idea is that some annotations may bring us closer to the goal of evaluating the concepts in the document.

2. When evaluating the relevance of documents a similarity index is calculated based on the annotations:

$$sim(d,q) = \frac{d \bullet q}{|d||q|}$$

d - The vector of annotation weights related to the document.
q - A vector resembling the query. q_i is the number of variables in the query which are related to instance *i*.

As is stated in the article (see reference), this method of ranking is an adaptation of the classic vector-space model used for similarity evaluation. The final similarity result for the document is:

$$sim_T(d,q) = \alpha \cdot sim_O(d,q) + (1-\alpha) \cdot sim_K(d,q),$$

where sim_O is the Ontology based similarity index and sim_K is a keyword based similarity index.

If the KB is incomplete then Ontology indices will suffer greater errors, but the keyword based indices will be available to soften the errors' implications.

Implementation languages

Remembering that Ontologies were first introduced by AI researchers, it is not surprising that the languages used to define Ontologies are mostly derived from the knowledge representation (KR) subfield of AI. The KR languages were part of earlier efforts to represent the aforementioned taxonomies, and inherently support definitions of classes, relations such as inheritance ("is-a" e.g. enzyme is-a protein…), properties and instances. The group of KR languages which are relevant to Ontology representation is called **description logics**. One of the first Ontology-dedicated languages was Stanford's Ontolingua.

Nowadays, the area of Ontology dedicated languages is very active. Prominent examples are OIL − Ontology Interface Layer – which was created by enhancing the capabilities of the pre-existent WWW frame language RDF with formal semantics and reasoning services, and XOL which similarly enhanced pre-existent XML. Another new development is OWL (a resource in the format of a PowerPoint presentation about its development).

At the same time that efforts are concentrated towards standardization and optimization of Ontology creation languages, some other researchers believe that the presently used query languages are inadequate for the task of meta-data querying and have designed specific tools for that purose. An example is the Xcerpt language (Here for PDF) and its close relative Xchange.

It must be noted that a very large information body can be found regarding these languages or others, and evidently interest and development is rapidly growing. Quite possibly new solutions will soon emerge, using the existent platforms or perhaps creating different ones, and will change the picture described above.

Summary

As shown above, Ontologies – first introduced more or less two decades ago – are now a focal point for interest and research. Perhaps the greatest expectations from Ontology arise from the role they are to play in the **Semantic Web**, the next generation of the well known WWW. As much as the current web has changed the lives of billions, so is the Semantic Web likely to do it again and Ontologies are to play a key role in that change. The ability to deal with abstract concepts through the Ontologies, rather than the "flat" texts and keywords associated, allow capabilities for inference, for context-related search, and ultimately, for the reuse and sharing of knowledge that is machine processable. The possibilities of semantic information retrieval were generally described above, with its tight relation to Ontologies and Ontology creation (semi-automatic and automatic methods).

However, Ontologies are not restricted to global network knowledge representation, and are applicable for any knowledge base that is intended for shared use. An example brought here is an application of Ontology for properly representing the critical clinical database of a hospital and the growing understanding of the importance of "corporate memory" assures many organizations will be interested in similar systems for knowledge management.

To conclude, it is expected that a large variety of tools and methods that either utilize Ontologies for improved knowledge management or retrieval, or assist in creating Ontologies more easily and generically, will be emerging in the near future. These will take the world toward another step in the direction of digitization of knowledge, as machines will be able to perform basic reasoning with it. The advances in this area will probably have a large impact on modern life.

References

1. Amandeep S. Sidhu, Tharam S. Dillon,Fellow IEEE, Elizabeth Chang,Member IEEE, **Creating a Protein Ontology Resource**

2. David Vallet, Miriam Fernández, and Pablo Castells, **An Ontology-Based Information Retrieval Model**

3. Fran¸cois Bry, Tim Furche, Paula-Lavinia Patranjan, and Sebastian Schaffert, **Data Retrieval and Evolution on the (Semantic) Web: A Deductive Approach**

4. Guoqian Jiang, Katsuhiko Ogasawara, Naoki Nishimoto, Akira Endoh, Tsunetaro Sakurai, **FCAView Tab: A Concept-oriented View Generation Tool for Clinical Data Using Formal Concept Analysis**

5. G. Marcos, H. Eskudero, C. Lamsfus , M.T. Linaza, **Data Retrieval From a Cultural Knowledge Database**

6. Jacob Köhler and Steffen Schulze-Kremer, **The Semantic Metadatabase (SEMEDA): Ontology based integration of federated molecular biological data sources**

7. Jeff Heflin and James Hendler, **Searching the Web with SHOE**

8. Marta Sabou, **Extracting Ontologies from Software Documentation: a Semi-Automatic Method and its Evaluation**

9. Rudi Studer, V. Richard Benjamins, Dieter Fensel , **Knowledge Engineering: Principles and methods**, Data & Knowledge Engineering 25 (1998) 161-197

10. Setfan Decker, Micheal Erdmann, Dieter Fensel and Rudi Studer, **Ontobroker: Ontology based Access to distributed and Semi-Structured Information**

11. Ying Ding, **IR and AI: The role of Ontology**

Interesting Links:

- **http://www.lsi.upc.es/~luigi/Ontologies.htm**
- **http://www.cs.utexas.edu/users/mfkb/related.html**
- **Ontobroker Links**
  - **Website homepage**
  - **Nutshell PDF file**
  - **Full article**

Appendix A – Some basic IR definitions

Two basic metrics used to quantify success in an IR task are Recall and precision. They are given here as they are used in several references without explanation, and are defined:

$$\mathrm{Re}\,call = \frac{\#\,\text{Relevant Documents Retrieved}}{\text{Total}\,\#\,\text{of Relevant Documents}}$$

$$Percision = \frac{\#\,\text{Relevant Documents Retrieved}}{\text{Total}\,\#\,\text{of Documents Retrieved}}$$

Indices for evaluating Ontology coverage are presented in **Extracting Ontologies from Software Documentation: a Semi-Automatic Method and its Evaluation**, for evaluation of the success of a semi-automatically created Ontology in contrast to a manually created one (the gold standard).

The lexical overlap (LO) equals to the ratio of the number of concepts shared by both Ontologies and the number of concepts we wish to extract:

$$LO(O_1, O_2) = \frac{\left|L_{O_1} \cap L_{O_2}\right|}{\left|L_{O_2}\right|}$$

Here LO1 is the set of all the concepts extracted by the tested method and LO2 the set of concepts of the Gold Standard.

The Ontology improvement (OI) equals the ratio of new concepts extracted by the tested nethod (expressed as the set difference between extracted and desired pairs) and all pairs of the gold standard Ontology.

$$OI(O_1, O_2) = \frac{\left|L_{O_1} \setminus L_{O_2}\right|}{\left|L_{O_2}\right|}$$

The Salton Index is an important measure of co-occurrence which is not biased by naturally high occurrence of certain keywords. It is defined as:

$$SI(X, Y) = \frac{C_{xy}}{\sqrt{C_x C_y}}$$

Where:
Cxy - The number of co-occurrences of x and y.
Cx - is the number of occurrences of x.
Cy - is the number of occurrences of y.