The State of the Art in Ontology Design: A Comparative Review

Natalya Fridman Noy

Carole D. Hafner

College of Computer Science
Northeastern University
Boston, MA 02115
{natasha, hafner}@ccs.neu.edu

Abstract

In this paper we develop a framework for comparing ontologies, and place a number of the more prominent ontologies into it. We have selected 10 specific projects for this study, including general ontologies, domain specific ones, and one knowledge representation system. The framework includes general characteristics such as the purpose of an ontology, its coverage (general or domain-specific), its size, and the formalism used. It also includes the design process used in creating an ontology and the methods used to evaluate it. Characteristics that describe the content of an ontology include taxonomic organization, types of concepts covered, top-level divisions, internal structure of concepts, representation of part-whole relations, and the presence and nature of additional axioms. Finally we consider what experiments or applications have used the ontologies. Knowledge sharing and reuse will require a common framework to support interoperability of independently created ontologies. Our study shows there is great diversity in the way ontologies are designed and the way they represent the world. By identifying the similarities and differences among existing ontologies, we clarify the range of alternatives in creating a standard framework for ontology design.

1 Introduction

Recent work in ontology design has produced a range of different projects from ontologies that represent general world knowledge, to domain-specific ones, to knowledge representation systems which embody ontological frameworks. There is an agreement in the ontology engineering community that it would be very beneficial to be able to integrate ontologies so that they can share and reuse each other's knowledge. If one ontology, for example, has a very well developed theory of time, another ontology (say, the one representing biology experiments) could then use that without having to re-invent the time microtheory. There is also an understanding that achieving interoperability of ontologies is a very challenging task. For smooth integration to be (at least partially) possible, the first thing to do is to look at the ontology projects that already exist and are fairly well developed and consider what are the differences and similarities in the way they treat some basic knowledge representation aspects. Having understood that, we can see where there is some common base and what are the obstacles to the integration of different ontologies. Identifying a framework for comparing ontologies and placing a number of the more prominent existing projects in this framework is the objective of the study presented below. We compare a number of ontology projects with respect to purposes they were created for, what the design process was, how they treat certain fundamental issues in representing knowledge, such as taxonomies, properties, relations, etc. We identify common themes and consider different approaches to these issues. We try to single out some major approaches in each dimension we consider, group the projects according to this categorization and point out the ones that do not fit in this categorization anyway.

One of the few prior studies that compares ontologies is in (Uschold 1996; Uschold & Gruninger 1996). The ontology comparison dimensions identified in it are *formality* (from highly informal to rigorously formal), *purpose* (what the ontology is used for) and *subject matter* (the nature of the domain that the ontology is characterizing). We add a significant number of other dimensions that consider *content* of ontologies in more detail and assess 10 specific projects based on these dimensions.

Our criteria for selecting ontologies for the study were, first, to get a representative set of projects, second, to use ontologies that are significant in size and relatively well developed, and, third, to use fairly well documented ontologies (at least documented well enough to be able to answer most of the questions we are asking: not all the data was available for all the projects in the study, however). Below is the list of projects we used:

- Cyc (http://www.cyc.com/cyc-2-1/cover.html ; Lenat 1990; Lenat & Guha 1990; Guha & Lenat 1994; Lenat 1995) - a project to create a general ontology for common-sense knowledge to facilitate reasoning.
- K. Dahlgren's ontology (Dahlgren 1988; Dahlgren 1995) - a linguistically-motivated ontology of common-sense knowledge.
- Generalized Upper Model (Generalized-UM) (http://www.darmstadt.gmd.de/publish/komet/gen-um/newUM.html ; Bateman, Magnini & Rinaldi 1994) - a general task and domain independent "linguistically-motivated" ontology that is designed to support sophisticated natural language processing in English, German and Italian.
- Gensim (Karp 1993) - a genetic simulation system that represents and models enzymatically-catalyzed biomedical reactions whose substrates include macromolecules with complex internal structures, such as DNA and RNA.
- Knowledge Interchange Format (KIF) (Genesereth & Fikes 1992) - a language for defining ontologies that has declarative semantics and is based on first-order predicate calculus.
- Plinius project (van der Vet & Mars 1993; van der Vet et al. 1994; van der Vet, Speel & Mars 1994) - an ontology for representing mechanical properties of ceramic materials with the goal of semi-automatic knowledge extraction from natural language texts.
- J. Sowa's ontology (Sowa 1995a; Sowa 1995b) - an attempt to synthesize philosophical insights to create a very general ontology based on the principles of "distinctions, combinations and constraints".
- Toronto Virtual Enterprise (TOVE) project (http://www.ie.utoronto.ca/EIL/tove/ontoTOC.html ; Gruninger & Fox 1995) - a project to create an ontology for enterprise modeling that will be able to deduce answers to queries about the information in the model.
- Unified Medical Language System (UMLS) (http://wwwkss.nlm.nih.gov/Docs/umls.fact.html ; Humphreys & Lindberg 1993) - an ontology of medical concepts designed to facilitate retrieval and integration of information from multiple machine-readable biomedical information resources.
- WordNet (Miller 1990) - a manually constructed on-line reference system which is one the most comprehensive lexical ontologies.

We have compared these projects according to a number of dimensions. These dimensions are summarized in Table 1. We begin by comparing general attributes of the projects in Section 2: what an ontology was created for, whether it is a general or domain-specific one, how can it be integrated in a more general ontology or how a more specific ontology can be linked to it. We also include in the comparison such technical data as ontology size, formalism that was used, whether an ontology was implemented and whether it is publicly accessible in its entirety. In Section 3 we consider the process of designing and evaluating ontologies. There is active discussion in the ontology community about different approaches to these processes.

In comparing the content of ontologies we discuss three different levels: first, an IS-A taxonomy of concepts, second, the internal concept structure and relations between concepts, and, third, presence or absence of explicit axioms. *Taxonomy* (Section 4) is the center part of most ontologies. Taxonomy organization can vary greatly: all concepts can be in one large taxonomy, or there can be a number of smaller hierarchies, or there can be no explicit taxonomy at all. Although all general-purpose ontologies try to categorize the same world, they are very different at their top level. They also differ in their treatment of basic parts of an ontology: *Things, Processes, Relations.* We compare top level categories of the ontologies. The next level of comparison is *internal concept structure* (Section 5). Internal structure can be realized by properties and roles. Concepts in some ontologies are atomic and may not have any properties or roles, or any other internal structure associated with them. We look at how the studied projects treat all these issues. We specifically study the treatment of *part-whole* relation in these ontologies. The third level in the comparison is the presence or absence of explicit *axioms* and the associated inference mechanisms (if any). In Section 6 we consider the ontologies' use of formal axioms, and whether they go beyond first-order logic.

An important test for any ontology is the practical applications it was used for. These can be applications in natural language processing, information retrieval, simulation and modeling, etc. that use knowledge represented in the ontology. Some applications of the projects are discussed in Section 7.

2 General characteristics

2.1 Projects' goals

Many features of an ontology depend on the purpose it was created for. Hence, one of the first comparison criteria was the purpose of a particular project. The major goals of the projects included *natural language understanding; information retrieval; theoretical investigation; knowledge sharing and reuse; simulation and modeling.*

Many ontologies are built for various *natural language applications* ranging from knowledge acquisition from text to semantic information retrieval. Cyc, Generalized-UM, WordNet, Dahlgren's ontology, UMLS and Plinius fall into this category. Although Cyc does not have any particular NL application it is designed for, NLP was one of the major motivations in the project to represent common-sense knowledge. WordNet and UMLS are large reference systems that are built to be used in other NLP

General	The purpose the ontology was created for General or domain-specific Domain (if domain-specific) Can it be easily integrated in a more general ontology Size: number of concepts, rules, links, etc. Formalism used Implemented or not (implementation platform and language) Published or not
Design process	How was the ontology built? Was there a formal evaluation?
Taxonomy	General taxonomy organization Are there several taxonomies or is everything in the same one What is in the ontology: things, processes, relations, properties? Treatment of time Top-level division How tangled or dense the taxonomy is
Internal concept structure and relations between concepts	Do concepts have internal structure? Are there properties and roles? Are there other kinds of relations between concepts? How are *Part-whole* relations represented?
Axioms	Are there explicit axioms? How are the axioms expressed?
Inference mechanism	How is reasoning done (if any) Instances of going beyond first-order logic
Applications	Retrieval mechanism User Interface Application where the ontology was used
Contributions	Major strengths and contributions Weaknesses

Table 1. Summary of ontology comparison characteristics used in this study.

systems, not in any particular application. TOVE has intelligent *information retrieval* as its main purpose. This retrieval however is not text retrieval. TOVE creates enterprise models and then answers queries to those models. Ideally it will not only answer queries with what is explicitly represented, but also be able to deduce answers.

Another class of ontologies are *theoretical investigations* that do not directly pursue the goal of building a working system. Sowa's ontology is an example of such a system. Sowa explores philosophical foundations for building knowledge models, history of ontologies starting from Aristotle and suggests his own model based on these earlier studies. We have also considered KIF which is a knowledge representation system since it also embodies an ontological framework: it provides ways (and limitations) for representing knowledge. A system called Ontolingua (Gruber 1992) is built on top of KIF and provides a common ontology-definition language. This of course is a first step in *knowledge sharing and reuse*: if different ontologies are to be shared they should at least be translatable into the same formalism. Ontologies that facilitate knowledge sharing and reuse are at the center of research in the field right now. Among the projects in this study (besides KIF) Generalized-UM also can claim knowledge sharing and reuse as one of its purposes as it attempts to share knowledge across different languages (the ontology is multi-lingual). Generalized-UM is also the only one of the NLP-supporting ontologies that spans several different languages, not only English.

Simulation and modeling is another purpose for ontology development projects. Gensim, for instance, develops a model of qualitative scientific knowledge about objects and processes in molecular biology and biochemistry, such that this knowledge can be used in a qualitative simulation to predict experimental outcomes.

2.2 General or domain-specific?

One of the very basic characteristics of an ontology is whether it attempts to cover general world or common-sense knowledge or if it is covering a specific domain. In both cases the question should be asked how it can be integrated with other ontologies. For a general one, the issue is whether domain-specific ontologies can be easily attached to it. Conversely, it should be considered whether a domain-specific ontology can be easily integrated into a more general one, and if it can use knowledge that is defined elsewhere, say, in some parts of a more general ontology.

Of the systems studied, Cyc, Dahlgren's ontology, Sowa's ontology, Generalized-UM, WordNet and KIF are not domain-specific. They are targeted at creating a general "world" model. Their view on what this model is, however, is very different from each other, as we will discuss in the later sections. From the other systems, TOVE is in the domain of enterprise modeling, UMLS is an ontology for medical concepts, Plinius deals with material science (specifically, ceramic materials) and Gensim deals with molecular biology and biochemistry.

In terms of knowledge sharing and integration with other ontologies, some general ontologies (e.g. Dahlgren's ontology) claim to simplify inclusion of new domains as an integral part of the original ontology or facilitate the interface between a domain ontology and the general ontology (Generalized-UM). Most domain ontologies (Gensim, TOVE) do not touch the subject of integration and it is unclear how this can be done. Plinius can be extended to an ontology of chemical substances, but it is unclear how it can go beyond chemical substances and be integrated with a more general ontology. UMLS is the only one of the domain-specific ontologies that may deal with these issues of integration (although not explicitly): UMLS starts its categorization from general notions of Entity and Event, so one can envision a general ontology where this would fit.

2.3 Implementation details

For the purpose of reference we have summarized some other characteristics[1] in Table 2. These include such characteristics as project's size (both in terms of concepts and axioms); the formalism that was used (frames, first-order logic, conceptual graphs, semantic networks, etc.); whether it was implemented or not (projects started with the purpose of theoretical investigation may not be implemented; for others, some may be implemented completely, some may, for example, just have a proof-of-concept prototype). The last but not least of these general questions is if ontology is published and freely available: some are very easily accessible in their entirety (with appropriate licensing agreements) and can be studied and reused. Some are proprietary information and are not published and one can only study the papers that describe it. In summary, UMLS, Cyc and WordNet are the largest systems in terms of the number of concepts (on the order of 10^5), Cyc also has the largest number of axioms (10^6).

3 Ontology design and evaluation process

There is an ongoing discussion in the ontology community about the best process for building an ontology. Should it be built bottom-up, starting from the most specific concepts and then grouping them in categories; or should it be built top-down by identifying the most general concepts and creating categories at the most general level first? Or should some middle layer of concepts serve as a starting point and then the development goes in both direction from there - middle-out approach (see (Uschold & Gruninger 1996) for the argument for the third alternative)?

More ontologies in the study used a bottom-up approach in constructing a hierarchy than top-down. Although almost no one states explicitly what approach they used, here is what we gathered from what was in the papers: top-down: Sowa's ontology; bottom-up: WordNet, Plinius; middle-out: TOVE.

There is some research toward *acquiring* ontological knowledge from natural language texts automatically, reducing manual effort. However, none of the ontologies we studied used any sort of automatic ontological knowledge acquisition. All were constructed manually with varying amounts of human effort involved depending on the size of the project: from the single-author ontologies of Sowa and Karp (Gensim), to the multi-person 10 year long Cyc effort.

For Generalized-UM that was *extending an already existing* fairly large *ontology* (Pangloss, (Hovy & Knight 1993)) to work for different languages, the design process was as follows: First, there was a Penman Upper Model for English; then German was added and the Merged Upper Model (Merged-UM) was created. The Generalized Upper Model is the extension of the Merged-UM to cover the three languages: English, German, and Italian. For each subhierarchy of the Merged-UM, the set of relevant Italian linguistic behavior was identified. The behavior was then compared to English. If Italian and English/German behavior were compatible, no modification was needed; otherwise the modification was proposed and the English/German model was re-evaluated.

WordNet, Plinius and Dahlgren's ontology used a *text corpus* or *dictionary* as the basis for their development process. The Plinius's approach, for instance, was to use

---

[1] Not all the data on these characteristics is available.

	Size	Formalism	Implemented?	Published or not?
Cyc	10^5 concept types; 10^6 axioms	CycL - Cyc'c representation language	Yes	Partially on-line: 3,000 concept types at the top level (http://www.cyc.com/cyc-2-1/cover.html)
Dahlgren's ontology	1500 nouns; 600 verbs	Prolog Predicates	Yes	Partially in print(Dahlgren 1988; Dahlgren 1995)
Sowa's ontology	90 concepts and concept types; 40 conceptual relations	Conceptual graphs	No	Partially in print(Sowa 1995)
Generalized UM	250 concepts	LOOM	Yes	Published on-line (http://www.darmstadt.gmd.de/publish/komet/gen-um/newUM.html)
WordNet	95,600 word forms in 70,100 synsets	Semantic networks	Yes	Published on-line (ftp://clarity.princeton.edu/pub/wordnet/)
TOVE		Frame knowledge base	Yes	
UMLS	133 semantic types; 49 semantic relations; 252,982 concepts	Semantic networks	Yes	Published on-line (http://wwwkss.nlm.nih.gov/Docs/umls.fact.html)
GENSIM		Frame knowledge base	Yes	
Plinius	about 150 atomic concepts and 6 construction rules	Frame knowledge base	Yes	A detailed report is published, but the ontology is not
KIF	N/A	is itself a formalism	Yes	Yes

Table 2. Technical data on the ontologies in the study.

the corpus as an operational specification of the domain. The ontology was required to cover every relevant concept from the texts and to make every relevant distinction. Similarly, Dahlgren's schema was originally developed to handle predicates, both nouns and verbs, found in 4100 words of text drawn from geography textbooks. Dahlgren also based her schema on cognitive psychology research. Psycholinguistic experiments with people were conducted (in particular, to determine properties and functions of categories). WordNet based its creation on *lexicographers-created files* of word forms and word meanings which were then automatically parsed into a database.

TOVE used the following approach to ontology design: first, create *Motivating Scenarios* - story problems or examples which are not adequately expressed in existing ontologies. Any proposal for a new ontology or extension to an ontology must describe a motivating scenario, and a set of intended solutions to the problem presented in the scenario. Second, formulate *Informal Competency questions* - a set of queries (in an informal form). Ideally, for each new object, relation, etc. there should be a competency question requiring it. These competency questions are used to evaluate expressiveness of the ontology.

One of the more interesting research issues in ontology design is *formal evaluation* of the created ontology (or any evaluation, for that matter) (Gómez-Pérez, Juristo & Pazos 1995). We considered whether there was an evaluation of the conceptual coverage or practical usefulness of the ontology. A proof of concept prototype can serve as such an evaluation (at least, for practical usefulness), or there can be a pre-determined corpus, and an assessment later if all the information in the corpus can be covered with the created ontology.

TOVE was the only project that did a formal evaluation of its ontology. This process consisted of representing the competency questions formally and then proving completeness theorems with respect to those queries based on the first and second order logic representation of concepts, attributes and relations. For Gensim, which was designed for simulations, the evaluation consisted of having the program predict outcomes of already known reactions. According to the author the predictions were "flawless". Most projects, however, envision various applications that would use the ontology and, thus, prove its conceptual coverage and practical usefulness (see the discussion on various applications ontologies were used for below).

4 Taxonomy

Formally, an ontology consists of terms, their definitions and axioms relating them (Gruber 1993); terms are normally organized in a taxonomy. This is where some disagreement among ontology researchers arises. Some say that axioms are central to ontology design and a complete or high-level taxonomy does not even have to exist (maybe only for visualization). Others say that, on the contrary, one should first concentrate on defining a taxonomy of fundamental concepts (although they agree that there should be axioms or knowledge in some other form associated with the concepts in the taxonomy). However, most of the ontologies we studied do have some sort of taxonomy (or several taxonomies), and this is our first topic in comparing *content* of the ontologies.

From the previous paragraph, the first question to be asked is if there is an explicit taxonomy of concepts and how are concepts organized: is it just a simple concept hierarchy? a more complex taxonomy with several dimensions at each level? a number of small local taxonomies? or something completely different?

We found three major approaches to concept organization (all of them include having some sort of a taxonomy; we discuss Plinius's taxonomy-less approach later). UMLS, Gensim and WordNet (for noun synsets only) adopt the approach of having everything in a single tree-like concept hierarchy with multiple inheritance. The links in the hierarchy are IS-A links and the division of a concept into subconcepts is disjoint.

Cyc, Generalized-UM, Dahlgren and Sowa use what Sowa calls "*distinctions*" approach. This means that there are several parallel dimensions along which one or more top-level categories are sub-categorized. For example, *Real* vs *Abstract*, *Individual* vs *Collective*, etc. In this case categories are specified by various combinations of values along these dimensions. For instance, `herd` can be categorized as being *Real* and *Collective*, whereas `idea` is *Abstract* and *Individual*. Sowa creates a sub-category for each possible combination of these values (which may lead to combinatorial explosion if more than one distinction is used at more than just a few top levels). This requirement also makes the top-level of the hierarchy very tangled. Dahlgren's ontology and Generalized-UM, however, have more than one distinction at some lower levels of the ontology, but they do not require a category to be created for every possible combination of distinctions.

The third major approach to taxonomy organization is having a large number of small local taxonomies that may be linked together via relations or axioms. TOVE and KIF represent this type of approach. TOVE, for example, divides its domain (enterprise modeling) into a number of different sub-ontologies (e.g. ontologies for: activity, product, time, organization and inside those: part, constraint, requirement, feature, etc.) Even within these smaller ontologies in TOVE no overall taxonomies exist. Its taxonomies seem to be local, each going very few levels deep.

Although WordNet uses a simple hierarchy for noun synsets, it employs different organization of synsets for verbs and adjectives. Descriptive adjectives, for instance, are organized in bipolar clusters (e.g., *dry/wet*). Relational adjectives, such as *fraternal* in *fraternal twins* are organized only in synsets with pointers to the corresponding nouns. Verbs are divided into 15 clusters according to their meaning, with entailment being the primary relationship between the verbs in a clusters.

A completely different way of defining and organizing categories is used in Plinius ontology. Technically, there is no taxonomy per se. The principle used to construct the ontology is called *Conceptual Construction Kit*. In short, an ontology consists of several sets of atomic concepts, such as chemical elements, real numbers, aggregation states (gaseous, liquid, etc.) and others, serving as primitives, and construction rules that define all other concepts. There are rules for groups, chemical substances, phases (built up of chemical substances in relative proportions), etc. Then a taxonomy is defined implicitly by subsumption. Each atomic concept set X has a pseudo-member called *arbitrary (X)* that stands for any member of the set. Now, for a concept that contains this term, any concept where the term is replaced by a particular member of set X, is a sub-concept.

To summarize, Cyc, Dahlgren's ontology, Sowa's ontology, Generalized-UM, UMLS and Gensim have all the concepts in one taxonomy (a simple one or with several dimensions at some levels). TOVE and KIF have a number of small local taxonomies. Plinius does not have any explicit taxonomy at all. And WordNet has a single taxonomy for its nouns, but a very different organization for verbs and adjectives.

4.1 Taxonomy. Treatment of specific categories

Although the projects that we studied were created for different purposes, there are a number of general classes of concepts that are represented in almost all ontologies: things, processes and events, relations and properties. This section discusses which of these classes are represented (or underrepresented) in each ontology.

Things (real or abstract) are represented everywhere. Plinius, Gensim, TOVE and UMLS do not attempt to represent all the Things in the universe, but those relevant to the domain are, of course, present in the ontology.

Processes and *events* are almost as ubiquitous as Things. Cyc, for example, defines *Process* as a subclass of *Event* and *Stuff*, which are both subclasses of *IndividualObject*. An *IndividualObject* that has a temporal extent (starting time, duration, ending time) is called a *Process*. As mentioned above, WordNet treats verbs (which basically correspond to processes and events) separately from nouns and adjectives. Generalized-UM has a separate taxonomy for what they call Configurations, which is the ontology of processes. Gensim has a limited number of experimental processes and reactions (which are also processes) defined.

Sowa points out in (Sowa 1995) that the distinction between something called an *Object* and something called a *Process* in fact depends on time scale. In his ontology, anything that does not change over time (on a particular time scale) is called an *Object* (*Continuant*) and anything that is "in the state of flux" is called a *Process* (*Occurrent*). To use Sowa's example, consider a glacier and an avalanche. A glacier is a "permanent" object on a scale of minutes and avalanche is a process. On a scale of centuries, however, glacier is also a process.

Much less universal than Things and Processes is the presence of some sort of *taxonomy of relations* and/or *properties* (the presence of which creates a need for higher-order logic; we discuss this issue in Section 6). Generalized-UM has probably the most extensive taxonomy of relations. Many properties in Generalized-UM (e.g. *Color-Property-Ascription*) are defined as concepts in the taxonomy with two (or more) roles for the concepts that are related by it. UMLS has a two-level deep taxonomy for relations and Dahlgren has a list (not a taxonomy) of all relations as part of the ontology.

Other general categories that are present in only one or very few of the studied projects and are worth mentioning are: things internal to the machine (Cyc), classification of spatio-temporal relations in Generalized-UM, axiomatization of sets and lists in KIF, locations in Gensim (active sites on DNA are a separate category in the taxonomy).

We considered the presence and treatment of one specific "microtheory", that is *the ontology of time*. Although for almost any kind of reasoning one needs some representation of time, we noticed that not all ontologies model temporal concepts (and, hence do not support any temporal reasoning). Generalized-UM and TOVE have very simple ones that axiomatize time points and time periods. In Cyc *Time* is a physical quantity possessed by *TemporalObjects* (such as *Events*).*TimeInterval*, which is a first-class object, is a *TemporalObject* that can be characterized fully just by specifying its temporal attributes. *TimeInterval* has dates, years, etc. as its sub-categories. Other ontologies do not touch this issue at all. Gensim justifies this by making an assumption that every experiment happens in a very short period of time. There is a possibility, in case of a smaller ontology being integrated into a larger ontology, of reusing an ontology of time present in a larger model. This, however, requires a smooth integration.

4.2 Taxonomy. Top-level division

One of the most interesting questions pertaining to ontology organization is: what are the major top-level categories in the ontology? How does the ontology divide the world at the top level?

The most ubiquitous top-level division of concepts is Abstract versus Real division. It is present in Dahlgren's and Sowa's ontologies as the top-level distinction (termed *Physical* vs *Informational* in the latter). The *Tangible* vs *Intangible* division in Cyc also reflects this distinction; Cyc, however, has a third category at the same level: *CompositeTangible&IntangibleObject* to denote something that has both a physical extent and intangible extent. *Person* category may be such an example: person's body constitutes the physical extent and person's mind is the intangible extent. In UMLS *Entities* are divided into *Physical Objects* and *Conceptual Entities*: that is, also along the Physical vs Abstract lines.

Another frequently-found top-level categorization is Individual versus Collection. Both of these are top-level distinctions in Dahlgren's and Sowa's ontologies. This distinction is also very pronounced at all levels in Cyc and (Lenat & Guha 1990) devotes a lot of discussion to this issue.

Lack of correspondence between ontologies in their top-level division poses an obstacle to integration of different ontologies. A. Campbell and S. Shapiro in (Campbell & Shapiro 1995) discuss an idea of a "mediation interface" that will translate statements made in one ontology to another ontology. The authors compare top levels of a number of ontologies in order to determine how similar or different they are and, hence how feasible it would be to integrate them. Two of the criteria they use is how tangled and how sparse or dense the top-level hierarchy is. For example, a simple tree-like structure with little or no multiple inheritance would not be considered tangled, whereas hierarchies that employ the *distinction* approach would have a highly tangled structure. Also, the more sub-categories exist at the top level of categorization, the more dense this top level is. It is, of course, easier to integrate ontologies that are more similar in the way they organize their top-level hierarchies (then, there is of course the issue of the top-level categories themselves being alike). Figure 1 illustrates the "tangledness" and "density" scales of the projects that we have studied. They vary from relatively sparse but very tangled hierarchies like Cyc to much more dense but less tangled ones, like WordNet.

5 Internal concept structure and relations between concepts. *Part-whole* relations.

Almost any ontology has something more than just a taxonomy of concepts. The least it can have is a set of properties and components that are meaningful for each category. This is the level of *internal concept structure.* Other relations among concepts (spatial, functional, etc.) may also be represented. Cyc, Dahlgren's ontology, Sowa's ontology, Gensim, Generalized-UM, TOVE, and KIF have properties and roles associated with concepts (often in the form of slots in a frame) and relations that link concepts to each other. Objects in Plinius are structured too, but differently from the frame-based systems above. The structure of its objects is defined by a set of construction rules that specify the internal composition of concepts.

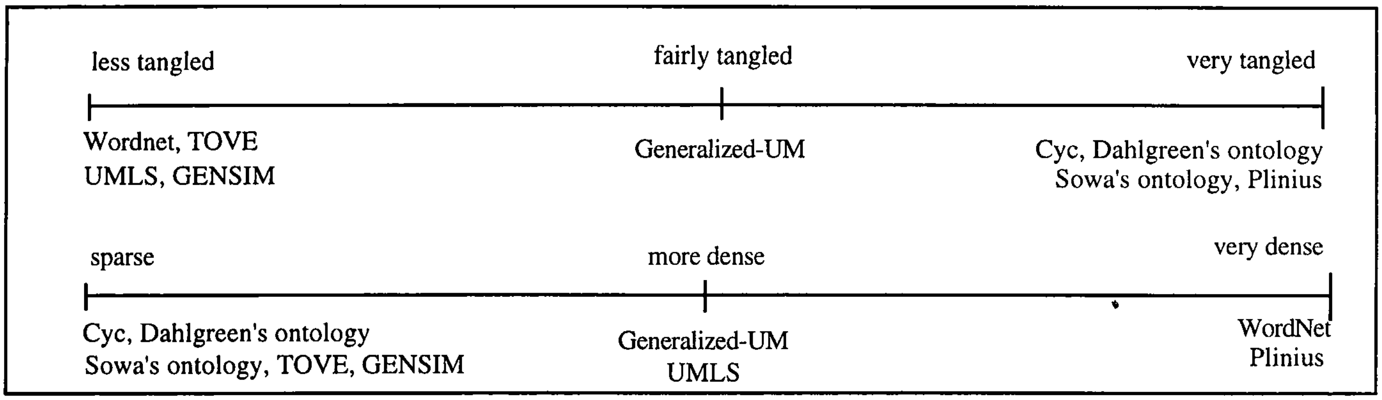

Figure 1. Comparison of how tangled and dense the ontologies in the study are.

WordNet and UMLS do not have any properties or roles associated with objects in their taxonomy: all the concepts are atomic and do not have any internal structure. They can be related to other concepts though, and the pre-defined relations themselves do have a limited structure (unlike concepts). The fact that they are all binary already provides some internal structure to them. In UMLS, relations (which are first-class objects and have their own taxonomy) also have their possible domain categories specified.

5.1 Part-whole relations

When studying relations between categories in the ontologies, we found one relation represented very differently in the ontologies and often not adequately dealt with. This is the issue of *part-whole* relations. There are several types of *part-whole* relation that may require different reasoning. For example, (Winston, Chaffin & Hermann 1987) differentiates the following types of part-whole relation: component-object (branch/tree); member-collection (tree/forest); portion-mass (slice/cake); stuff-object (aluminum/airplane); feature-activity (paying/shopping); place-area (Princeton/NJ); phase-process (adolescence/growing up). We were looking for different categorizations of part-whole relations and treatment for them.

Most of the ontologies do not directly address the issue of part-whole relation and the distinction between subset-of, part-of, member-of, etc. Part-whole relations are handled just like other roles or relations. However, Generalized-UM, Sowa's ontology, and TOVE provide some analysis of part-whole relations. Here is how each of the systems does it.

Generalized-UM: *Part-whole-relation* is a concept in the taxonomy of relations. It is a relation with two roles: *whole* (the domain) and *part* (the range). It is a specialization of a *generalized-possession* relation. There are three possible subtypes that are described for part-whole, although they are not currently distinguished within the grammar:

consists-of — expressed as: <whole> consist of <parts> or <parts> make up <whole>
The filler of the part role of the *consists-of* relation must be *all* of the parts of the whole. For example, protein consists of amino acids. But not: "a car consists of an engine"

constituency — a specialization of the part-whole relation in which the whole is value-restricted to be a *decomposable-object*. For example, an engine is a constituent of a car.

ingrediency — this is the relation between a whole and its parts when the whole is a *mass-object*. For example, Gravel is an ingredient of concrete.

Sowa's ontology: There is a category in the taxonomy (*InternalRole*) for things that play a role with respect to something in which they are contained (as opposed to *External* Role for the things that play a role with respect to something outside themselves). *InternalRole* is subdivided into several categories: when a *Continuant* vs *Occurrent* distinction is applied to *InternalRole* it produces *ObjectPart* and *ProcessPart*. Object parts that can exist independently of the object are called *pieces* (e.g. engine of a car); those that cannot are called *attributes* (e.g. size, weight, color of a car). For *ProcessPart* the same distinction leads to *participant* (e.g. a book and a reader in a reading process) and *manner* (e.g. speed of the wind, style of a dance, etc.).

TOVE: A part is defined as a component of an artifact being designed or a software component. The artifact itself is also considered a part. Parts are classified into: Primitive Part (a part which cannot be further subdivided into components), Composite Conjunct Part (composed of two or more primitive and/or composite parts), Composite Disjunct Part (represent alternatives of parts, i.e. at any point in time, the part has only one of its components as a valid component).

From these three categorizations, Sowa's is the most general one. All the types of *part-whole-relation* in Generalized-UM are what Sowa calls *pieces* (which Generalized-UM goes on to classify). Attributes and process' participants and manner are not considered parts in TOVE and Generalized-UM (Note that although a part-of relation in Gensim is generalized to include processes, part in that case is a sub-process of a process). TOVE's approach to *part-of* relation is different as it reflects the way an artifact is composed from its parts.

WordNet and Plinius also single out the *part-of* relation. In fact, this is the primary relation between concepts (beyond taxonomy) in the two systems: this is the only relation currently implemented between noun synsets in WordNet and it is the only relation between classes in Plinius, along with its counterpart COMPOSES. In UMLS, the following relations are included in the relation taxonomy within the category *physical-relation* : *part-of, contains, consists-of.*

6 Axioms and first-order logic

Besides the taxonomy and structure of concepts, *axioms* are a way of representing more information about categories and their relations to each other, as well as constraints on property and role values for each category. Sometimes, axioms are explicitly specified, and sometimes ontology consists only of categories and corresponding frames and everything else is "hidden" in application code. It is important to note here, that there is a fine line between internal concept structure and axioms. One can represent a category using a frame formalism, having roles and properties represented by slots of a frame. One may also express the same facts using axioms. A taxonomy, too, can be represented using axiomatic notations. For example, the axiom $\forall a(a \in A \Rightarrow a \in B)$ where a is an instance of a category and A and B are categories, states that A is a sub-category of B. Here we will only be looking at explicit axioms that go beyond the hierarchy representation or internal concept structure. We consider how axioms are expressed and if they are, for example, part of a concept definition or can exist by themselves.

The following projects explicitly use axioms that go beyond taxonomic and property/role information: Cyc, TOVE, Sowa's ontology, Gensim, KIF. Generalized-UM has all the axiomatic information incorporated in the NLP code. Dahlgren's ontology and Gensim incorporate axioms in concept definitions. WordNet and UMLS do not have any axioms.

When discussing axioms and formalism, one of the questions that we pay particular attention to is what are the instances of going beyond first-order logic. Some systems use defaults, or ways of expressing modals and uncertain facts. Some do not do this and stay within the boundaries of first-order logic.

One of the most common instances of going beyond first-order logic is having some sort of hierarchy of relations, that is treating relations as first-class objects: Generalized-UM and TOVE are examples of this approach. UMLS also has hierarchy of relations but since it does not have axioms the first-order logic issue is irrelevant for it.

Another common example of going beyond first-order is use of defaults: for example, Cyc and KIF. For KIF this is the only instance of going above first-order as it is based primarily on first-order logic. Gensim employs over-ride inheritance in its process hierarchy (property values in a sub-class can override the corresponding values in a superclass), which leads to non-monotonicity, too. Sowa uses conceptual graphs which themselves use higher-order logic.

Here are some cases when Cyc goes beyond first-order logic (Lenat 1995):

- Certainty: each assertion is assumed true by default but one can make statements like "Assertion A is less likely than assertion B"
- Reification: turning a predicate or function into an object in the language. It allows assertions about categories: "Property P is an opposite of property Q"
- Modals: "John wants assertion A to be true"
- Contexts: an assertion may be true only in a particular context. Contexts are first-class objects in Cyc. Example: "You cannot see someone's heart" (true only as a default, but not true during heart surgery)

7 Applications

An important way of evaluating capabilities and practical usefulness of an ontology is considering what practical problems it was applied to. In this section we summarize some of the applications that the ontologies in this study were used for.

The major classes of applications which ontologies are utilized for are: *natural-language processing, information retrieval* and *simulation and modeling*. Cyc's ontology, for example, is used in a Cyc Natural Language System (CNL) whose purpose is to translate natural-language texts into CycL. Generalized-UM is used in a multilingual text generation system that uses stock phrases for each concept to generate text. Dahlgren's ontology was a basis for a text understanding system that reads newspaper articles to produce a cognitive model of the text content (Dahlgren 1990). Plinius' application also falls into the class of natural-language systems: it is designed for extracting knowledge from titles and abstracts of articles in its domain and creates an interim knowledge base where the knowledge is associated with a particular abstract. Cyc was used in information retrieval for a system called Cyccess, which is a semantic IR system used for consistency checking and information retrieval from structured information such as databases and spreadsheets.

Cyc, Gensim and TOVE were used in simulation and modeling applications. There is a person modeling prototype application that uses Cyc's ontology to put together a model of a person based on pieces of information it might have about person's interests, family, job, etc. This information is then used to sort, for example, advertisements that should or should not be sent to a person based on the model. Gensim was primarily

Strengths and contributions	**Ontologies**
Content creation	Cyc, UMLS, Generalized-UM, WordNet
Well-defined formalism creation	KIF
Approach based on linguistic and psycholinguistic data	Dahlgren's ontology
Thoroughly-motivated top-level	Sowa's ontology
Multilingual	Generalized-UM
Extended hierarchy of relations	Generalized-UM
Novel approach to ontology design	Plinius (Conceptual Construction Kit)
Comparison of different KR formalisms	Plinius
Methodology for formal evaluation	TOVE

Table 3. Summary of major strengths and contributions of the ontologies in the study.

created and used for simulation of metabolic pathways, DNA transcription, etc. TOVE project's primary goal is to model a virtual company and provide a testbed for research into enterprise integration.

From other classes of applications, UMLS was used to implement Internet Grateful Med interface to MEDLAB databases (developed by the NLM itself). KIF served as the basis for Ontolingua which translates definitions written in standard form into specialized representations, including frame-based systems as well as relational languages.

8 Conclusion. Major strengths and contributions of the studied ontologies.

To conclude, we consider major strengths and contributions of each project, such as content creation, well-defined formalism creation, some novel approach to ontology design, and others. We also outline weaknesses of particular projects. Major strengths and contributions of the projects in the study are summarized in Table 3.

Many researchers in the area agree that one of the major challenges in the area of ontology design is creating *content* of ontologies, that is creating large, well-developed, usable ontologies (either general or domain-specific). Cyc, WordNet and UMLS are major steps in this direction. There are very few projects that span more than one language and can be applied to natural-language texts in various languages. Generalized-UM is one of the multilingual projects (it is based on English, German and Italian). One of the very interesting things that was done as part of the Plinius project, was implementing their ontology in several different knowledge representation formalisms (Speel 1995). This was a substantial step in showing experimentally the independence of the ontology itself from the formalism that is used. The TOVE project made a significant step in an under-developed area of ontology research: formal evaluation.

As for integration and of various ontologies, this study shows that at this point there is great diversity in the way ontologies are designed and in the way they represent the world. Before real knowledge sharing and reuse will be practical, some standards should emerge in what an ontology should consist of, what are the basic classes of objects that should be represented (e.g. things, processes, relations), how they are represented (not in terms of formalism but in terms of knowledge that should accompany the concepts).

We believe that a study like the one presented here is a useful step in the process of developing these standards, because, before we try to standardize, we first need to understand the alternatives. The framework presented here examines the range of alternatives in ontology design by doing a comparative study of 10 existing projects.

Acknowledgments

This research is supported in part by the National Science Foundation under grants No. IRI-9117030 and IRI-9633661

References

Bateman, J. A., Magnini, B., and Rinaldi, F. 1994. The Generalized Italian, German, English Upper Model. In Proceedings of ECAI'94 Workshop: Comparison of Implemented Ontologies. Amsterdam.

Campbell, A., and Shapiro, S. 1995. Ontologic Mediation: An Overview. In Proceedings of Workshop on Basic Ontological Issues In Knowledge Sharing, 16-25. Montreal, Canada.

Dahlgren, K. 1988. *Naive Semantics for Natural Language Understanding*. Boston, MA: Kluwer Academic.

Dahlgren, K. 1990. Naive Semantics and Robust Natural Language Processing. In Proceedings of Symposium on Text-based Intelligent Systems; AAAI Spring Symposium Series, 5-9. Stanford University: AAAI.

Dahlgren, K. 1995. A linguistic ontology. *International Journal on Human-Computer Studies* 43(5/6): 809-818.

ftp://clarity.princeton.edu/pub/wordnet/ WordNet: Princeton University.

Genesereth, M. R., and Fikes, R. E. 1992. Knowledge Interchange Format, Version 0.3, Reference Manual: Knowledge Systems Laboratory, Stanford University.

Gómez-Pérez, A., Juristo, N., and Pazos, J. 1995. Evaluation and Assessment of the Knowledge Sharing Technology. *Towards Very Large Knowledge Bases*. N. J. I. Mars, ed.: IOS Press: 289-296.

Gruber, T. R. 1992. Ontolingua: A Mechanism to Support Portable Ontologies: Knowledge Systems Laboratory, Stanford University.

Gruber, T. R. 1993. Toward Principles for the Design of Ontologies Used for Knowledge Sharing: Knowledge Systems Laboratory, Stanford University.

Gruninger, M., and Fox, M. S. 1995. Methodology for the Design and Evaluation of Ontologies. In Proceedings of Workshop on Basic Ontological Issues in Knowledge Sharing, IJCAI-95. Montreal.

Guha, R. V., and Lenat, D. B. 1994. Enabling Agents to Work Together. *Communications of ACM* 37(7): 127-142.

Hovy, E., and Knight, K. 1993. Motivation Shared Knowledge Resources: An Example from the Pangloss Collaboration. In Proceedings of Workshop on Knowledge Sharing and Information Interchange, IJCAI'93. Chamberry, France.

http://www.cyc.com/cyc-2-1/cover.html Upper Cyc Ontology: Cycorp.

http://www.darmstadt.gmd.de/publish/komet/gen-um/newUM.html The Generalized Upper Model 2.0. Darmstadt, Germany: IPSI/GMD.

http://www.ie.utoronto.ca/EIL/tove/ontoTOC.html TOVE Manual: Department of Industrial Engineering, University of Toronto.

http://wwwkss.nlm.nih.gov/Docs/umls.fact.html Unified Medical Language System. Bethesda, MD: National Library of Medicine.

Humphreys, B. L., and Lindberg, D. A. B. 1993. The UMLS project: making the conceptual connection between users and the information they need. *Bulletin of the Medical Library Association* 81(2): 170.

Karp, P. D. 1993. A Qualitative Biochemistry and Its Application to the Regulation of the Trytophan Operon. *Artificial Intelligence and Molecular Biology*. L. Hunter, ed.: AAAI Press/ The MIT Press: 289-325.

Lenat, D. B. 1990. Cyc: Toward programs with common sense. *Communications of ACM* 33(8): 30-49.

Lenat, D. B. 1995. CYC: A Large-Scale Investment in Knowledge Infrastructure. *Communications of ACM* 38(11): 33-38.

Lenat, D. B., and Guha, R. V. 1990. *Building large knowledge-based systems : representation and inference in the Cyc project*. Reading, Mass: Addison-Wesley Pub. Co.

Miller, G. A. 1990. WordNet: An On-line Lexical Database. *International Journal of Lexicography* 3, 4: 235-312.

Sowa, J. 1995a. Distinctions, Combinations, and Constraints. In Proceedings of Workshop on Basic Ontological Issues In Knowledge Sharing. Montreal, Canada.

Sowa, J. F. 1995b. *Knowledge Representation: Logical, Philosophical, and Computational Foundations*. Boston, MA: PWS Publishing Company.

Speel, P.-H. 1995. Selecting Knowledge representation systems. Ph.D. diss., University of Twente, Enschede, the Netherlands.

Uschold, M., ed. 1996. *EuroKnowledge Glossary*: ESPRIT Project 9806, available from ftp://ftp.aiai.ed.ac.uk/pub/projects/euroknow/delvrbls/glossary.ps.

Uschold, M., and Gruninger, M. 1996. Ontologies: Principles, Methods and Applications. *Knowledge Engineering Review* 11(2).

van der Vet, P. E., Jong, H. d., Mars, N. J. I., Speel, P.-H., and Stal, W. G. t. 1994. Plinius Intermediate report: University of Twente, The Netherlands.

van der Vet, P. E., and Mars, N. J. I. 1993. Structured System of Concepts for Storing, Retrieving and Manipulating Chemical Information. *Journal of Chemical Information and Computer Sciences* 33: 564-568.

van der Vet, P. E., Speel, P.-H., and Mars, N. J. I. 1994. The Plinius ontology of ceramic materials. In Proceedings of ECAI'94 Workshop on Comparison of Implemented Ontologies, 187-205. Amsterdam.

Winston, M. E., Chaffin, R., and Hermann, D. J. 1987. A Taxonomy of Part-Whole Relations. *Cognitive Science* 11: 417-444.