The State of the Art in Ontology Design

A Survey and Comparative Review

Natalya Fridman Noy and Carole D. Hafner

■ In this article, 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 comparison 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 concept 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.

The major goal of this article is to develop a framework for comparing various projects in ontology design and put a number of prominent ontologies in this framework.

According to Webster's dictionary (Woolf 1981), *ontology* is a particular theory about the nature of being or the kinds of existent. The task of intelligent systems in computer science is to formally represent these existents. A body of formally represented knowledge is based on conceptualization. *Conceptualization* consists of a set of objects, concepts, and other entities about which knowledge is being expressed and of relationships that hold among them. Every knowledge model is committed to some conceptualization, implicitly or explicitly. An explicit specification of this conceptualization is called an ontology (Gruber 1993). Formally, an ontology consists of terms, their definitions, and axioms relating them (Gruber 1993); terms are normally organized in a taxonomy.

Most of the researchers in the area of ontology design agree that the current important goals of ontology research are to (1) make ontologies sharable by developing common formalisms and tools; (2) develop the content of ontologies (ontology design); and (3) compare, gather, translate, and compose different ontologies. Recent work in ontology design has produced a range of different projects, from ontologies that represent general world knowledge to domain-specific ontologies to knowledge representation systems that embody ontological frameworks. There is an agreement in the ontology engineering community that it would be 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 well-developed theory of time, another ontology (say, the one representing biology experiments) could then use this theory without having to reinvent it. There is also an understanding that achieving the interoperability of ontologies is a 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 the differences and similarities in the way they treat some basic knowledge representation aspects. With this understanding, we can see where there is some common base and what the obstacles are 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

Project Name	**Project Description**
CYC	A general ontology for commonsense knowledge to facilitate reasoning
K. Dahlgren's ontology	A linguistically motivated ontology of commonsense knowledge
GENERALIZED UPPER MODEL	A general task and domain-independent ontology that is designed to support sophisticated natural language processing in different languages
GENSIM	A genetic simulation system that represents and models enzymatically catalyzed biomedical reactions
Knowledge interchange format (KIF)	A language for defining ontologies that has declarative semantics and is based on first-order predicate calculus
PLINIUS Project	An ontology for representing mechanical properties of ceramic materials
J. Sowa's ontology	An attempt to synthesize philosophical insights to create a general ontology
Toronto virtual enterprise (TOVE) Project	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)	An ontology of medical concepts
WORDNET	A manually constructed online reference system that is one of the most comprehensive lexical ontologies

Table 1. List of Projects Used in This Study.

is the objective of the study presented here. After giving a brief introduction to a number of ontology projects, we compare them with respect to what they were created for; what the design process was; and how they treat certain fundamental issues in representing knowledge, such as taxonomies, properties, and relations. We identify common themes and consider different approaches to these issues. We try to single out some major approaches in each dimension, group the projects according to this categorization, and point out the ones that do not fit in this categorization.

Another motivation for creating a framework for comparing ontologies is the growing work in producing libraries of ontologies that can be reused. For a researcher to orient himself/herself in an ontology library and determine which ontologies are good candidates for reuse, it would be extremely useful to get a description of ontologies in a library in a more or less standard form. A framework for comparing ontologies, such as the one represented here, is a first step in defining a set of questions and possible answers that would allow an ontology to "introduce" itself to a potential user.

One of the few prior studies that compares ontologies is discussed in Uschold (1996) and Uschold and Gruninger (1996). The ontology comparison dimensions identified in this study 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 the content of ontologies in more detail and assess 10 specific projects based on these dimensions. A brief description of each project (with references) is presented in Project Overviews.

Our criteria for selecting ontologies for the study were to (1) get a representative set of projects, second, (2) use ontologies that are significant in size and relatively well developed, and (3) 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 were available for all the projects in the study, however). The list of projects we used is in table 1 (a more detailed description of each project is in Project Overviews).

We compared these projects according to a number of dimensions, which are summarized in table 2. We begin by comparing general attributes of the projects in General Characteristics: 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. In the comparison, we also

General	The purpose the ontology was created for General or domain specific Domain (if domain specific) Easy integration possible into a more general ontology Size: Number of concepts, rules, links, and so on Formalism used Implementation platform and language, if done Publication, if done
Design process	How was the ontology built? Was there a formal evaluation?
Taxonomy	What is the general taxonomy organization? Are there several taxonomies, or is everything in the same one? What is in the ontology: things, processes, relations, properties? What is the treatment of time? What is the top-level division? How tangled or dense is the taxonomy?
Internal concept structure and relations between concepts	Do concepts have internal structure? Are there properties and roles? Are there other kinds of relation 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)? What are some instances of going beyond first-order logic?
Applications	Retrieval mechanism User interface Application in which the ontology was used
Contributions	Major strengths and contributions Weaknesses

Table 2. Summary of Ontology Comparison Characteristics Used in This Study.

include such technical data as ontology size, formalism that was used, any ontology implementation, and its public accessibility. We then 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: (1) an is-a taxonomy of concepts, (2) the internal concept structure and relations between concepts, and (3) the presence or absence of explicit axioms. Taxonomy 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, and relations. In Taxonomy, we compare top-level categories of the ontologies. The next level of comparison is internal concept structure (see Internal Concept Structure and Relations between Concepts). Internal structure can be realized by properties and roles. Concepts in some ontologies are atomic and might not have any properties or roles or any other internal structure associated with

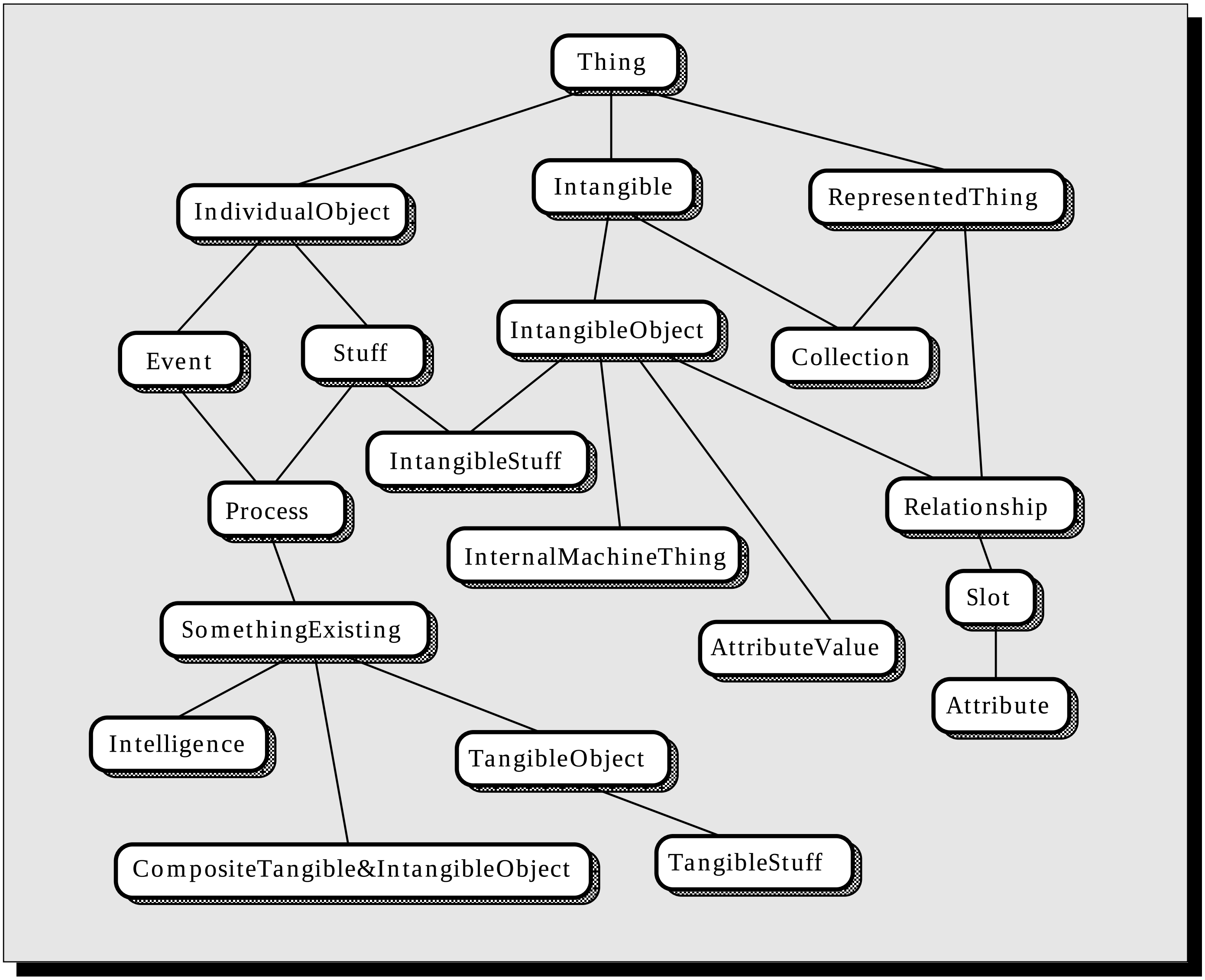

Figure 1. *CYC: Top-Level Categories (adapted from Lenat and Guha [1990]).*

them. We look at how the studied projects treat all these issues. We specifically study the treatment of part-whole relations in these ontologies. The third level in the comparison is the presence or absence of explicit axioms and the associated inference mechanisms (if any). We consider what the ontologies' use of formal axioms is 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, and so on, that use knowledge represented in the ontology. Some applications of the projects are discussed in Applications.

Project Overviews

In this section, we present a brief overview of each of the projects in the study. We describe the general information about the project, such as its goal and scope, and some of the more important aspects of its ontology. We start with more general ontologies, such as CYC and WORDNET, and then go on to domain-specific ones (such as UMLS and TOVE) and the knowledge representation system (KIF). To avoid excessive repetition, we discuss some of the details of the projects later in this article during the comparison.

CYC Project

Twelve years ago, a comprehensive effort was

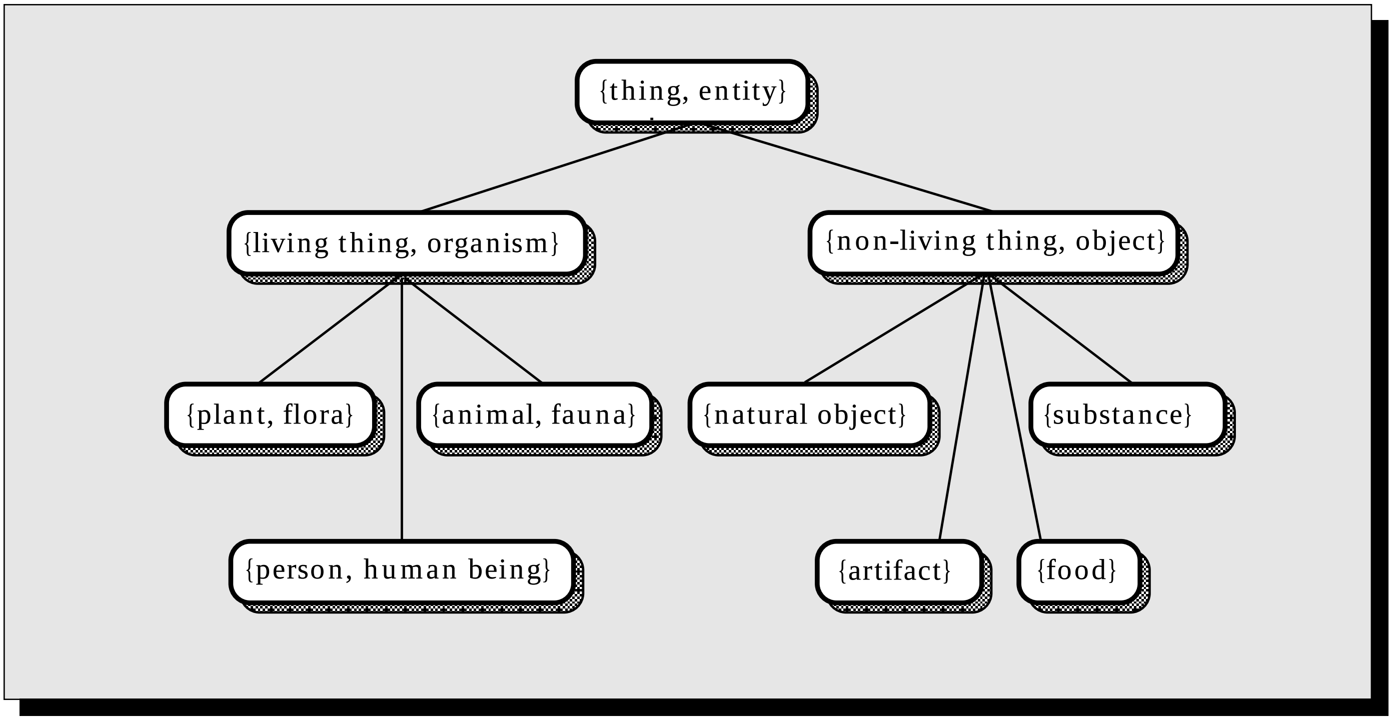

Figure 2. *WORDNET: Representation of Subclass Relation among Synsets Denoting Different Kinds of Tangible Things (Miller 1990).*
Braces enclose concepts in the same synset.

begun to create a general ontology for commonsense knowledge: the CYC Project (Lenat 1995; Guha and Lenat 1994; Lenat 1990; Lenat and Guha 1990; also see www.cyc.com/cyc-2-1/cover.html). CYC contains more than 10,000 concept types used in the rules and facts encoded in the knowledge base. The upper level of the CYC hierarchy is presented in figure 1. At the top of the hierarchy is the Thing concept, which does not have any properties of its own. The hierarchy under Thing is quite tangled. Not all the subcategories are exclusive. In general, Thing is partitioned in three ways:

First is RepresentedThing versus InternalMachineThing. Every CYC category must be an instance of one and only one of these sets. InternalMachineThing is anything that is local to the platform CYC is running on (strings, numbers, and so on). RepresentedThing is everything else. Sowa (1995b) criticizes this proposition saying that CYC should not be excluded from representing things in its own machine.

Second is IndividualObject versus Collection, which is another total partition of Things. Collections include all the categories mentioned in CYC. Hence, Collection doesn't have mass and is imperceptible. Sowa argues that it is unclear where something such as a flock of birds would be (which is a collection that is clearly perceptible).

Third is Intangible versus TangibleObject versus CompositeTangible&IntangibleObject. Every unit in CYC is an instance of exactly one of these three categories. Intangible is anything that has no mass (set of all people, number42, and so on), whereas TangibleObject is anything that does have mass and energy (a rock, a person's body). CompositeTangible&IntangibleObject is something that has both a physical extent and an intangible extent. For example, a particular person has a body (physical extent) and a mind (intangible extent).

It is interesting to note that Event and its subclass, Process, are subclasses of IndividualObject.

WORDNET

One of the most well-developed lexical ontologies is WORDNET (Miller 1990; also see ftp://clarity.princeton.edu/pub/wordnet/). WORDNET is a manually constructed online lexical reference system. Lexical objects in WORDNET are organized semantically (with the basic distinction between nouns, verbs, adjectives, and adverbs). The central object in WORDNET is a *synset,* a set of synonyms. If a word has more than one sense, it will appear in more than one synset. There are 70,000 synsets. Synsets are organized in a hierarchy by superclass-subclass relationship (referred to as hypernymy-hyponymy). Part of the WORDNET hierarchy of

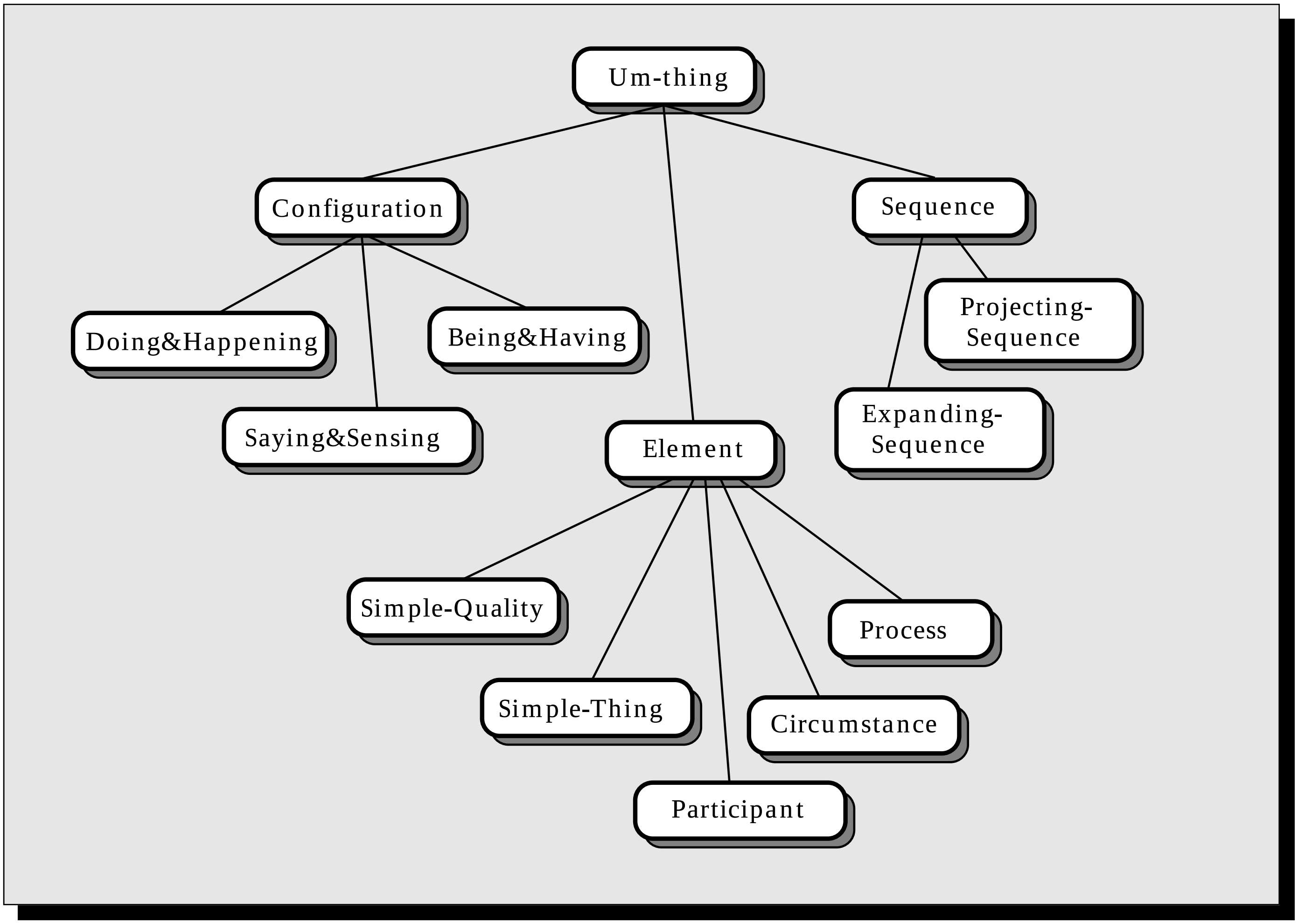

Figure 3. Top-Level Hierarchy from the GENERALIZED UPPER MODEL.

tangible things is presented in figure 2. For each concept (synset), there is a pointer to nouns representing its parts. For example, parts for the bird concept are beak and wings. There are allowances in the WORDNET implementation for other types of pointer (for example, from noun to verb to represent functions or to adjective to represent properties), but these types of pointer have not been implemented yet. WORDNET is a taxonomy; it does not have structured concepts or axioms.

Although WORDNET uses a simple hierarchy for noun synsets, it employs a different organization of synsets for verbs and adjectives. Descriptive adjectives are organized in bipolar clusters based on antonymy. For example, a bipolar cluster is generated by dry and wet with synonyms of each of the adjectives at the corresponding side of the cluster. Relational adjectives, such as fraternal in fraternal twins, are organized only in synsets with pointers to the corresponding nouns.

Verbs in WORDNET are divided into 15 clusters according to their meaning, with entailment being the primary relationship between the verbs in a cluster. Most of these clusters correspond to semantic domains: verbs of bodily care and functions, change, cognition, communication, competition, and so on. The verbs, such as *suffice, belong,* and *resemble* that do not belong to any of the semantic domains and refer to states, form a separate file.

GENERALIZED UPPER MODEL

The GENERALIZED UPPER MODEL (Bateman, Magnini, and Rinaldi 1994; also see www.darmstadt.gmd.de/publish/komet/gen-um/newUM.html) is a general task and domain-independent linguistically motivated ontology that supports sophisticated natural language processing in English, German, and Italian. Its level of abstraction is in between lexical knowledge and conceptual knowledge. It claims to simplify the interface between domain-specific knowledge and general linguistic resources. The model proposes a

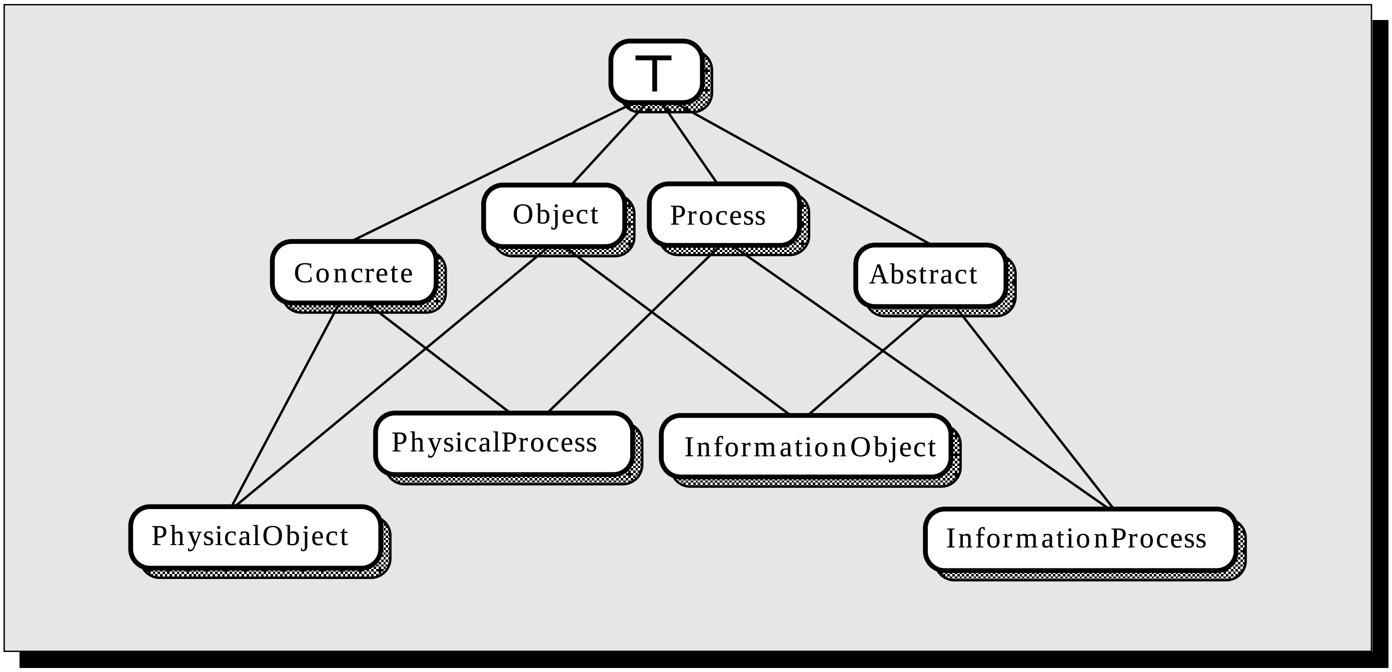

Figure 4. Sowa's Ontology.

Part of the ontology based on the two distinctions: Physical versus Information and Continuant versus Occurrent.

domain organization and consists only of a taxonomy (with the assumption that natural language–processing tools that will use it will encode the axiomatic information in the natural language–processing code itself). There is an extensive hierarchy of concepts (about 250 of them) as well as a separate hierarchy for relationships (um-relation).

At the top of the concept hierarchy (partially represented in figure 3) is the concept *umthing* that represents the most general phenomena or situation. It is subdivided into three major subtypes: (1) *configuration,* "a configuration of elements all participating in some activity or state of affairs"; (2) *element,* a single conceptual term; and (3) *sequence,* "a complex situation where various activities or configurations are connected by some relation to form a sequence" (see www.darmstadt.gmd.de/publish/komet/gen-um/node9.html). Relations are then used to link elements into configurations and sequences. Most of the relations are between a process and its participants, manner, and so on. Thus, the category um-relation is subdivided into process in configuration, circumstance in configuration, and participant in configuration. Causal relation, for example, would then be one of circumstances in configuration relations, and attribute would be one of participants in configuration relations.

Sowa's Ontology

John Sowa (1997, 1995a) states his fundamental principles for ontology design as "distinctions, combinations, and constraints" (1995a, p. 175). He uses philosophical motivation as the basis for his categorization. There are three top-level distinctions:

First is Physical versus Information, or Concrete versus Abstract. This is a disjoint partition of all the categories in the ontology.

Second is Firstness versus Secondness versus Thirdness, or Form versus Role versus Mediation. These categories are not mutually exclusive. For example, Woman is considered to be a *form* (Firstness) because it can be defined without considering anything outside a person. As a mother, a teacher, or an employee, the same individual would be an example of a *role* (Secondness). These roles represent an individual in relation to another type (a chile, a student, an employer). Marriage is a *mediation* (Thirdness) category because it relates several (in this case, two) types together.

Third, Continuant versus Occurrent, or Object versus Process. *Continuants* are objects that retain their identity over some period of time; *occurrents* are processes "whose form is in the state of flux" (Sowa 1995a, p. 179). For example, Avalanche is a process, and Glacier is an object. Note that this distinction depends on the time scale. On a grand time scale of centuries, Glacier is also a process.

These distinctions are combined to generate new categories (figure 4). At a lower level, for example, Script (for example, a computer pro-

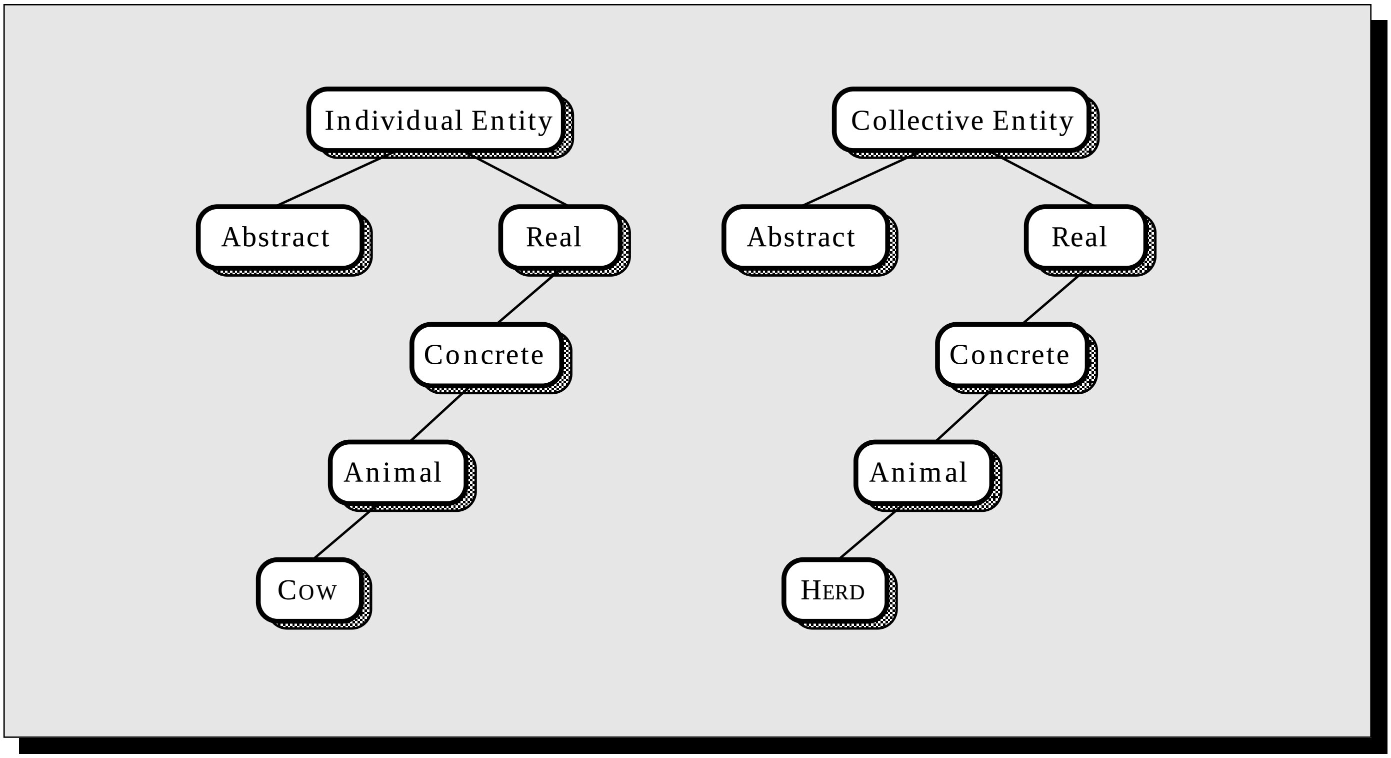

Figure 5. Parallel Portions of the Dahlgren (1988) Ontology.

gram, a baking recipe) is a form that represents sequences and is thus defined as Abstract, Form, Process. Also, History (an execution of a computer program), which is a proposition that describes a sequence of processes, is then Abstract, Form, Object.

At the top level, there is a category for every possible combination of distinctions. To avoid too many combinations, however, constraints are used at the lower levels to rule out categories that cannot exist. Logical constraints, for example, would rule out triangles with four sides, and empirical constraints would rule out talking birds. Constraints are represented as axioms and are inherited through the hierarchy to lower levels.

Dahlgren's Ontology

Kathleen Dahlgren (1988) presents a linguistically motivated ontology. One of the main points made by Dahlgren is that ontology should be linked closely to language, and her ontology is based on extensive linguistic and psycholinguistic research. The primary construction principle in the ontology is the cross-classification hierarchy, which means that at each node of the taxonomy, branching might occur in several dimensions. For example, at the top level, Entity cross-classifies as either Abstract or Real and as either Individual or Collective. These divisions are assumed to be essentially parallel. Thus, something is Individual and Real (Cow), and something is Collective and Real (Herd). Figure 5 shows parallel portions of the ontology from Dahlgren (1988). Note that not all the links in the ontology are of the same type (which makes it somewhat confusing): Although most of the links are is-a ones, some of the links used in the collective part of the hierarchy are not. For example, Herd is placed under Animal in the hierarchy (clearly, a consists-of relation), but Animal is under Concrete (an is-a relation).

At the bottom of the hierarchy are so-called terminal nodes that "do not have theoretical importance" (Dahlgren 1988, p. 52) and that are not cross-classified. For example, Animal, Role, and Person are all terminal nodes. Then, specific nouns are attached to terminal nodes. Maybe because all the nouns are specified based on their use in sentences, the classification of terminal nodes sometimes contradicts the semantics of a word. For example, a terminal node Culture (with E.coli as one of the nouns attached to it) is classified as Individual and Nonselfmoving (along with Social and Living). E.coli bacteria can move by themselves if treated as individual bacteria. However, Culture is more than a single bacteria and, hence, is not Individual.

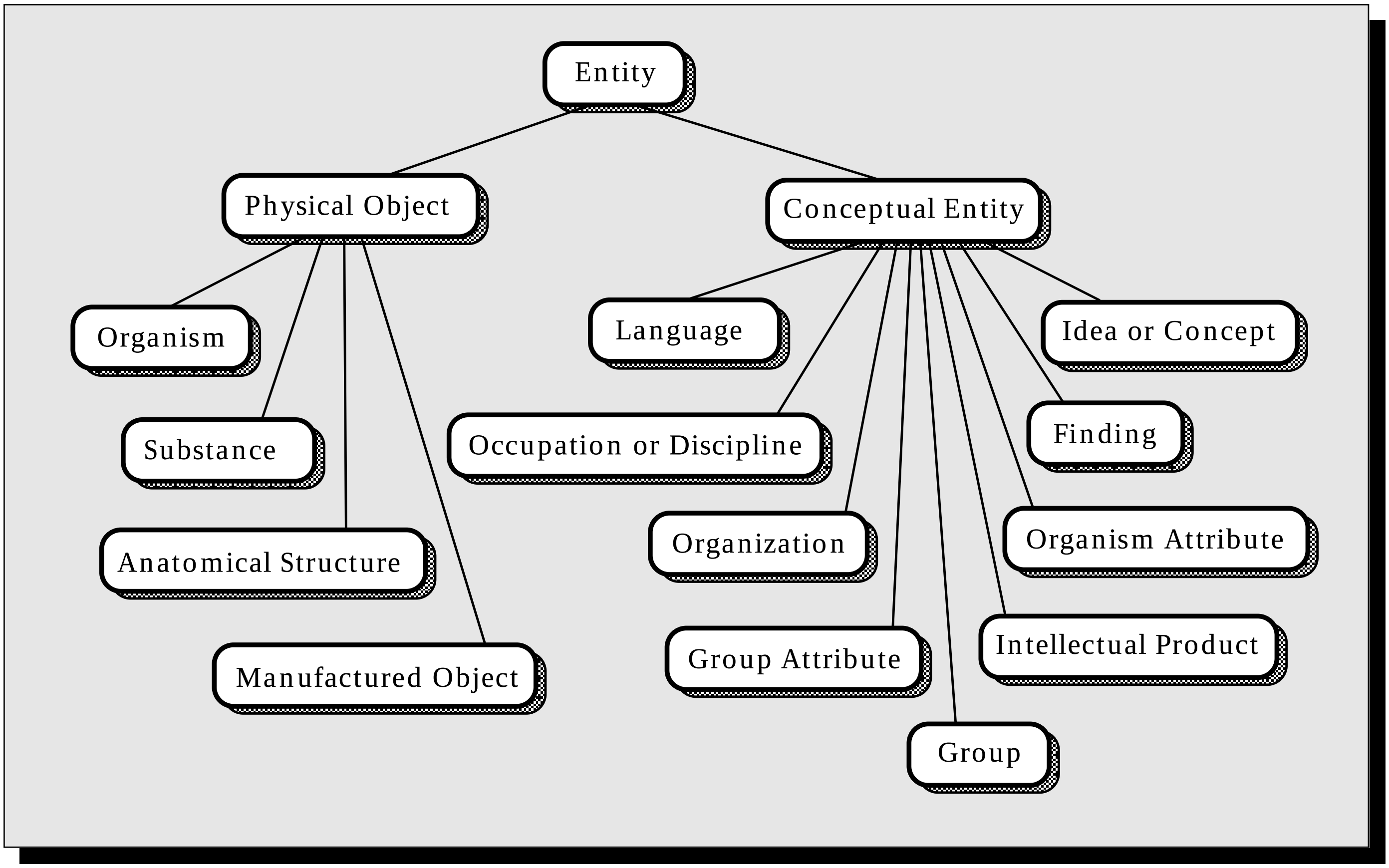

Figure 6. UMLS*: Top-Level Hierarchy of Entities.*

Unified Medical Language System (UMLS)

The unified medical language system (UMLS) of the National Library of Medicine (Humphreys and Lindberg 1993; also see wwwkss.nlm.nih. gov/Docs/umls.fact.html) is an example of a domain-specific system. This system was designed to facilitate retrieval and integration from multiple machine-readable biomedical information resources. The project's goal was to facilitate the link between different sources and users that use different terminology.

UMLS has a concept hierarchy of 135 medical concepts and a semantic network that represents additional (nontaxonomic) relationships between categories. Its concept hierarchy includes both Entities (Physical and Conceptual) and Events, such as Activity, Process, and Injury or Poisoning. Figure 6 shows the top-level hierarchy of Entities.

The relations represented in the system include such semantic relations as physically_ related_to, spatially_related_to, functionally_ related_to, temporally_related_to, and conceptually_related_to. There are 51 different relations in all, and the semantic network of concepts and all the possible relations that hold between them are extremely large. One of the major gaps in this ontology, however, is that it does not consider the dynamics of processes and the dynamic versus static relations between them.

Toronto Virtual Enterprise (TOVE)

The Toronto Virtual Enterprise (TOVE) Project (Gruninger and Fox 1995; TOVE 1995; also see www.ie.utoronto.ca/EIL/tove/ontoTOC.html) is an example of a domain-specific ontology tailored for a specific task: enterprise modeling. The goal of the project is to create enterprise models that can answer questions pertaining to the information explicitly in the model and can also deduce answers to queries. TOVE uses a formal approach to the ontology-engineering process itself and the ontology evaluation. First, through interaction with their industrial partners, the ontology engineers determine the problems that arise in the actual enterprises. In this way, the questions that the ontology should be able to answer are formed; they are called *competency questions*. These questions justify the choice of concepts and relations in the ontology. Then, the competency questions

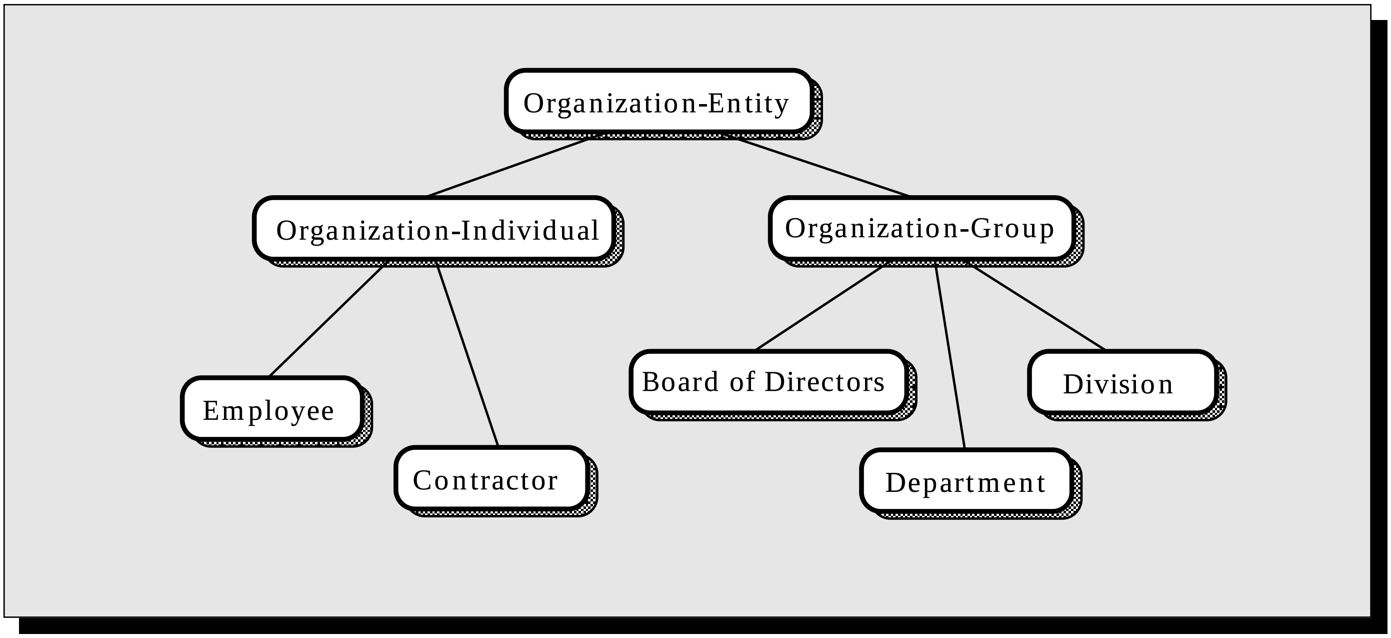

Figure 7. TOVE*: An Organization-Entity Hierarchy in the Organization Ontology.*

An Organization entity can be an Individual or a Group.

and all concepts and axioms are formalized in first-order logic, and the completeness theorems for the competency questions are proved based on this terminology.

As for the ontology itself, there is no single ontology as such but a set of several ontologies for various logical parts in enterprise modeling (that are then interlinked by relations). These ontologies are for activities and states (including a representation of time), products, organization, and activity-based cost management. Within each of these ontologies, a number of small hierarchies (two or three levels deep) represent small clusters of knowledge. Figure 7 gives an example of such a cluster. Axioms and relations are used to link knowledge from various clusters.

GENSIM

GENSIM (Karp 1993) is a genetic simulation system that represents and models enzymatically catalyzed biochemical reactions whose substrates include macromolecules with complex internal structures, such as DNA and RNA. There are two subontologies in GENSIM (called *knowledge bases*): (1) the class knowledge base and (2) the process knowledge base. The top-level hierarchy of the class knowledge base is represented in figure 8. It includes biochemical objects, such as genes, proteins, and biochemical binding sites. General classes of biochemical objects are represented with frames. Frames are also used to represent instances of these objects in simulations. In GENSIM, each object, such as Protein, represents not a single molecule but a homogeneous population of such molecules.

The process knowledge base describes the potential behavior of objects in the system. The author uses a qualitative process theory (Forbus 1984) approach to represent processes, and each process has a detailed frame that includes not only such parameters as input and output of reactions but also preconditions for a reaction to occur and effects of a reaction. This information is then used to simulate reactions based on the specified starting conditions.

PLINIUS

The PLINIUS Project (van der Vet, Speel, and Mars 1994; van der Vet et al. 1994; van der Vet and Mars 1993) is aimed at semiautomatic knowledge extraction from text in the domain of material science, ceramic materials in particular. The conceptualization of the chemical composition of materials is at the center of the PLINIUS ontology. The ontology for processes and properties has not been published yet, so we only discuss the ontology of materials. The authors of PLINIUS use an approach to ontology organization that is different from a traditional hierarchical-axiomatic approach. They call it the *conceptual construction kit:* The ontology starts with sets of atomic concepts, such as chemical elements, real numbers, and aggregation states (gaseous, liquid, and so on), serving as primitives. Elements of these sets are then combined to define all other concepts. This

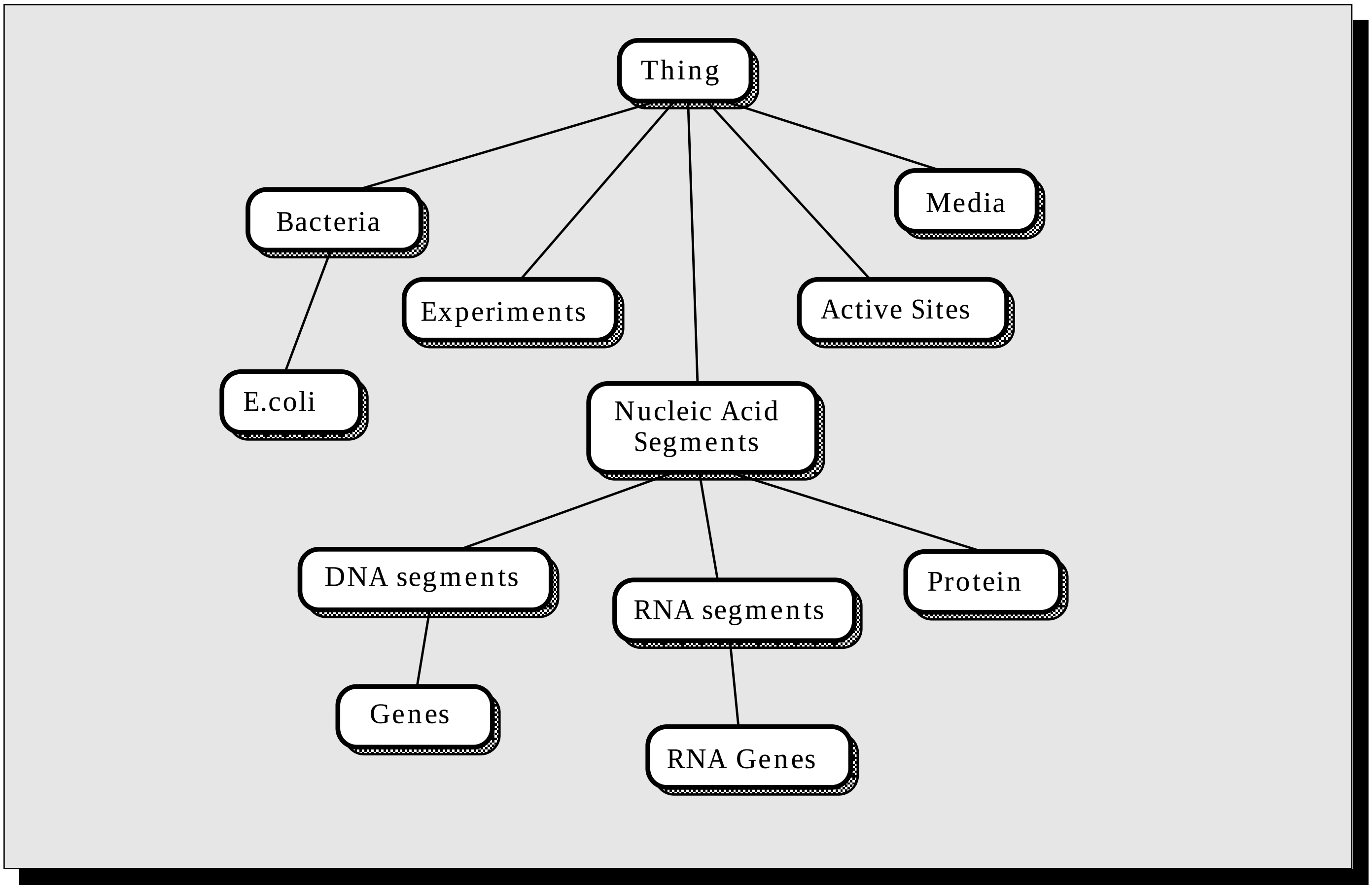

Figure 8. GENSIM: *Top-Level Ontology.*

In fact, the author does not use a general top-level concept Thing, and the five next-level concepts are unrelated. We introduced the top-level concept Thing for representation uniformity with other projects.

combination is defined by construction rules that govern the ways in which the combinations can occur. Construction rules define how complex substances and structures are constructed from the atomic concepts. There are rules for groups, chemical substances, and phases (built of chemical substances in relative proportions).

A taxonomy is then defined implicitly by subsumption. The lattice in figure 9 illustrates this principle. Suppose there are three primitive sets: (1) *A*, (2) *B*, and (3) *C*. Capital letters represent an arbitrary element of the set. Small indexed letters represent specific elements of the corresponding set (we used only one element of each set to simplify the figure): a_1 is an element of the set *A*, b_1 is an element of *B*, and c_1 is an element of *C*. Suppose also that there is a construction rule that combines elements of these three sets to form a complex object. Then, an object where all the components are specific elements of the sets is at the bottom of the hierarchy: It is the most specific one. If we allow one of the three components to be substituted with an arbitrary element of the corresponding set, we get a more general category, of which our original category is a subclass. If we allow two of the three components to be arbitrary elements of the corresponding sets, we get an even more general concept. The most general category is, of course, the one where none of the components is fixed.

It is unclear how this approach can be extended beyond chemistry or how processes and properties would be represented.

Knowledge Interchange Format (KIF)

The knowledge interchange format (KIF) is a language for defining ontologies (Genesereth and Fikes 1992). It provides for the definition of objects, functions, and relations. KIF has declarative semantics, and it is based on first-order predicate calculus. It provides for the representation of metaknowledge and allows

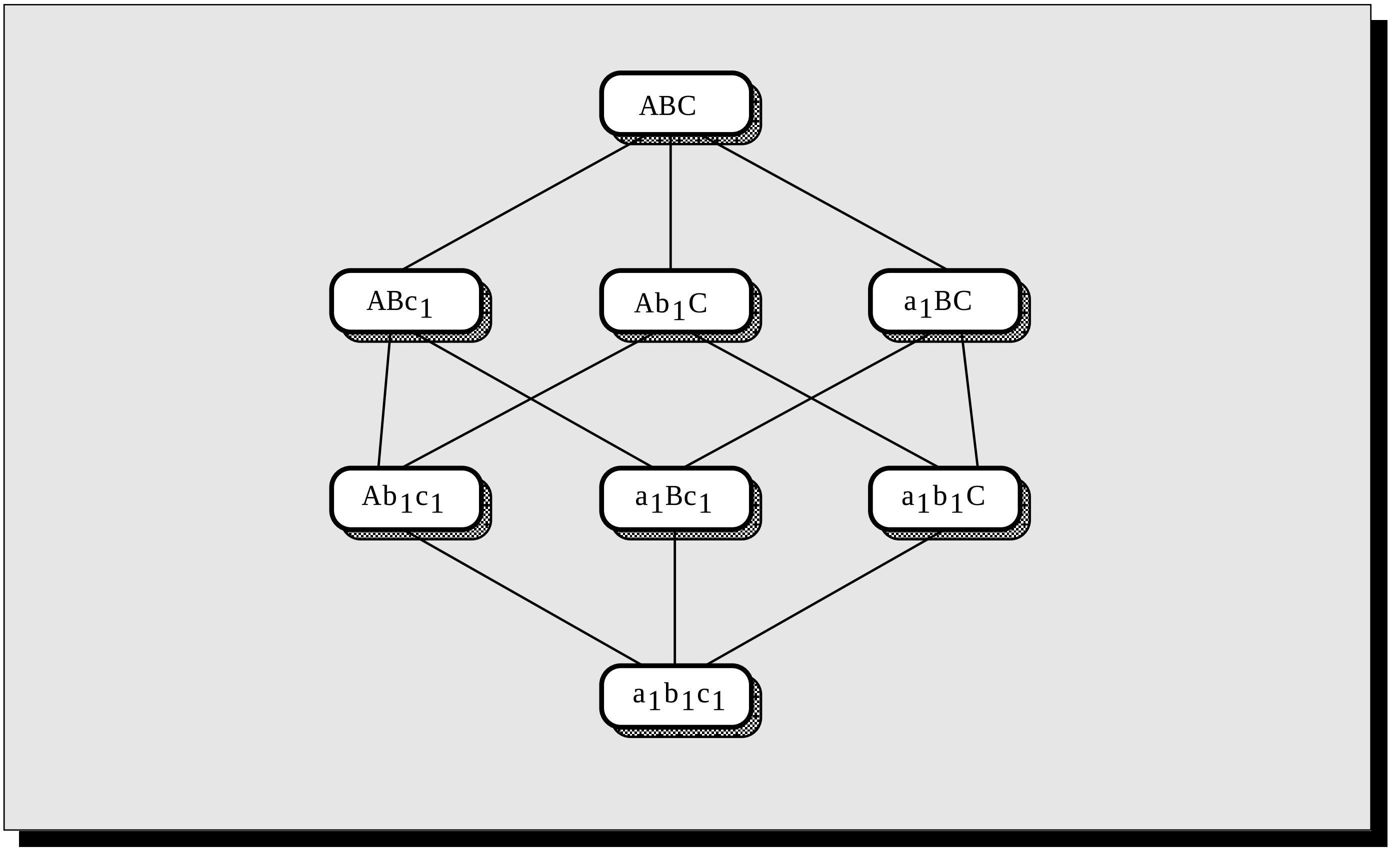

Figure 9. PLINIUS: *An Illustration of Taxonomy Organization by Subsumption.*

Capital letters correspond to arbitrary elements of the sets; small indexed letters correspond to specific elements of the corresponding sets (thus, a_1 is an element of the set A, and so on).

for the representation of nonmonotonic reasoning rules. We consider KIF an ontology because a certain view of the world is incorporated in it. Although its representation is not, by far, at the level of detail of other ontologies described here, microtheories of numbers, sets, and lists, for example, are axiomatized in KIF. Some axioms of the microtheory of sets are presented in figure 10.

As an interchange format, KIF is tedious to use for specifications of ontologies as such, but there are systems built on top of it that allow an ontology to be specified in the familiar terms of classes, relations, and so on. An example of such a system is ONTOLINGUA (Gruber 1992), which is a set of tools for analyzing and translating ontologies. The World Wide Web site for the Knowledge Systems Laboratory at Stanford University (www-ksl.svc.stanford.edu: 5915/), where this system was developed, contains a number of ontologies designed with ONTOLINGUA and a user-friendly ontology editor for creating your own ontologies, which also allows you to incorporate existing ontologies from the database into your own ontology.

General Characteristics

In this section, we discuss the general characteristics of an ontology. These characteristics include what the project goals are and whether the ontology is general or domain specific as well as some implementation details.

Project 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 UPPER MODEL, WORDNET, Dahlgren's ontology, UMLS, and PLINIUS fall into this category. Although CYC does not have any particular natural language application it is designed for, natural language processing was

1. All entities in KIF are either *individuals* or *sets*. This distinction is exhaustive and mutually disjoint:

```
(or (set ?x) (individual ?x))
(or (not (set ?x)) (not (individual ?x)))
```

2. An object can be a member of another object if the latter is a set:

```
(=> (member ?x ?s)
   (set ?x))
```

3. Extensionality property: two sets are identical if and only if they have the same members

```
(=> (and (set ?s1) (set ?s2))
   (<=> (forall (?x)
          (<=> (member ?x ?s1) (member ?x ?s2)))
  (= ?s1 ?s2)))
```

Figure 10. KIF: A Partial Axiomatization of a Microtheory of Sets from Genesereth and Fikes (1992).

one of the major motivations in the project to represent commonsense knowledge. WORDNET and UMLS are large reference systems that are built to be used in other natural language–processing 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 these models. Ideally, it will not only answer queries with what is explicitly represented but also be able to deduce answers.

Another class of ontologies is **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, examines the history of ontologies starting from Aristotle, and suggests his own model based on these early studies. We also considered KIF, which is a knowledge representation system because 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 system, 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 current center of research in the field. Among the projects in this study (besides KIF) GENERALIZED UPPER MODEL also can claim knowledge sharing and reuse as one of its purposes as it attempts to share knowledge across different languages (the ontology is multilingual). GENERALIZED UPPER MODEL is also the only one of the natural language process–supporting ontologies that spans several different languages in addition to English.

Simulation and modeling are another purpose for ontology-development projects. GENSIM, for example, 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.

General or Domain Specific?

One of the basic characteristics of an ontology is whether it attempts to cover general world or commonsense knowledge or a specific domain. In both cases, we should ask how it can be integrated with other ontologies. For a general ontology, the issue is whether domain-specific ontologies can easily be attached to it. Conversely, it should be considered whether a domain-specific ontology can easily be 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 UPPER MODEL, WORDNET, and KIF are not domain specific. They are targeted at creating a general world

	Size	**Formalism**	**Implemented?**	**Published or Not?**
CYC	10^5 concept types; 10^6 axioms	CYCL—CYC's representation language	Yes	Partially online: 3000 concept types at the top level (www.cyc.com/cyc-2-1/cover.html)
Dahlgren's ontology	1500 nouns; 600 verbs	Prolog predicates	Yes	Partially in print (Dahlgren 1995, 1988)
Sowa's ontology	90 concepts and concept types; 40 conceptual relations	Conceptual graphs	No	Partially in print (Sowa 1997)
GENERALIZED UPPER MODEL	250 concepts	LOOM	Yes	Published online (www.darmstadt.gmd.de/publish/komet/genum/newUM.html)
WORDNET	95,600 word forms in 70,100 synsets	Semantic networks	Yes	Published online (ftp://clarity.princeton.edu/pub/wordnet/)
TOVE		Frame knowledge base	Yes	
UMLS	135 semantic types; 51 semantic relations; 252,982 concepts	Semantic networks	Yes	Published online (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	Published report but not ontology
KIF	N/A	Is itself a formalism	Yes	Yes

Table 3. Technical Data on the Ontologies in This Study.

model. Their views on what this model is, however, differ, as we discuss in later sections. Of the remaining 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 (for example, 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 UPPER MODEL). Most domain ontologies (GENSIM, TOVE) do not touch the subject of integration, and it is unclear how integration 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 might 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 ontology would fit in.

Implementation Details

For the purpose of reference, we summarized some other characteristics in table 3.[1] These characteristics include what the project size is (both in terms of concepts and axioms), what formalism was used (frames, first-order logic, conceptual graphs, semantic networks, and so on), and whether it was implemented or not (projects started with the purpose of theoretical investigation might not be implemented; for others, some might be implemented completely, or some might just have a proof-of-concept prototype). The last but not least of these general questions is if the ontology is published and freely available: Some are 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).

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? Should it be built top down by identifying the most general concepts and creating categories at the most general level first? Should some middle layer of concepts serve as a starting point and then the development go in both directions from there—a middle-out approach (see Uschold and 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: The top-down approach is used in Sowa's ontology; the bottom-up approach is used in a WORDNET and PLINIUS; the middle-out approach is used by TOVE.

There is some research toward automatically acquiring ontological knowledge from natural language texts, 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 multiperson 10-year-long CYC effort.

For GENERALIZED UPPER MODEL, which **extended an already existing** fairly large **ontology** (PANGLOSS [Hovy and 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 was created. The GENERALIZED UPPER MODEL is the extension of the MERGED UPPER MODEL to cover the three languages: (1) English, (2) German, and (3) Italian. For each subhierarchy of the MERGED UPPER MODEL, 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 reevaluated.

WORDNET, PLINIUS, and Dahlgren's ontology used a **text corpus,** or **dictionary,** as the basis for their development process. The PLINIUS approach, for example, was to use the corpus as an operational specification of the domain. The ontology was required to cover every relevant concept from the texts and 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 lexicographer-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, *motivating scenarios,* story problems or examples that are not adequately expressed in existing ontologies, were created. 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, *informal competency questions*, a set of queries (in an informal form), were formulated. Ideally, for each new object, relation, and so on, there should be a competency question requiring it. These competency questions are used to evaluate the expressiveness of the ontology.

One of the more interesting research issues in ontology design is the formal evaluation of the created ontology (or any evaluation, for that matter) (Góméz Perez, Juristo, and 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 predetermined 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 of the various applications that ontologies were used for).

Taxonomy

Formally, an ontology consists of terms, their definitions, and axioms relating them (Gruber 1993); terms are typically 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 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), which is our first topic in comparing the content of the ontologies.

Based on the previous paragraph, the first question to ask is whether there is an explicit taxonomy of concepts? Then, how are concepts organized? Is it just a simple concept hierarchy or a more complex taxonomy with several dimensions at each level? Is it a number of small local taxonomies or something completely different?

We found three major approaches to concept organization (all of them have some sort of taxonomy; we discuss the PLINIUS 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 UPPER MODEL, and Dahlgren's and Sowa's ontologies use what Sowa calls the *distinctions approach*. Thus, there are several parallel dimensions along which one or more top-level categories are subcategorized, for example, Real versus Abstract and Individual versus Collective. In this case, categories are specified by various combinations of values along these dimensions. For example, Herd can be categorized as Real and Collective, whereas Idea is Abstract and Individual. Sowa creates a subcategory for each possible combination of these values (which can 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 tangled. Dahlgren's ontology and GENERALIZED UPPER MODEL, 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 might be linked together by relations or axioms. TOVE and KIF represent this type of approach. TOVE, for example, divides its domain (enterprise modeling) into a number of different subontologies (for example, ontologies for activity, product, time, and organization and inside those for part, constraint, requirement, feature). 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 a different organization of synsets for verbs and adjectives. Descriptive adjectives, for example, are organized in bipolar clusters (for example, dry-wet). Verbs are divided into 15 clusters according to their meaning, with entailment being the primary relationship between the verbs in a cluster.

A completely different way of defining and organizing categories is used in the PLINIUS ontology. Technically, there is no taxonomy as such. The principle used to construct the ontology is the conceptual construction kit. In short, an ontology consists of several sets of atomic concepts, such as chemical elements, real numbers, and aggregation states (gaseous, liquid, and so on), serving as primitives and construction rules that define all other concepts. There are rules for groups, chemical substances, phases (built of chemical substances in relative proportions), and so on. Then, a taxonomy is defined implicitly by subsumption. Each atomic concept set *X* has a pseudomember 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 *subconcept*.

To summarize, CYC, Dahlgren's ontology, Sowa's ontology, GENERALIZED UPPER MODEL, 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. WORDNET has a single taxonomy for its nouns but a different organization for verbs and adjectives.

Treatment of Specific Categories

Although the projects that we studied were created for different purposes, there are a number of general classes of concept that are represented in almost all ontologies: things, processes and events, relations, and properties. This subsection 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 UPPER MODEL 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 (1977) points out 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 a state of flux is called a Process (Occurrent). Consider a tree, for example. A tree is a permanent Object on a scale of minutes. On a scale of years, however, a tree is a Process.

Much less universal than Things and Processes is the presence of some sort of taxonomy of relations or properties (the presence of which creates a need for higher-order logic; we discuss this issue in Internal Concept Structure and Relations between Concepts). GENERALIZED UPPER MODEL has probably the most extensive taxonomy of relations. Many properties in GENERALIZED UPPER MODEL (for example, 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's ontology has a list (not a taxonomy) of all relations as part of the ontology.

Other general categories that are present in only one or a few of the studied projects and worthy of mention are things internal to the machine (CYC), classification of spatiotemporal relations in GENERALIZED UPPER MODEL, axiomatization of sets and lists (KIF), and locations (GENSIM) (active sites on DNA are a separate category in the taxonomy).

We considered the presence and treatment of one specific microtheory—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 UPPER MODEL and TOVE have 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, and so on, as its subcategories. Other ontologies do not touch this issue at all. GENSIM, for example, justifies not using any ontology of time by making an assumption that every experiment happens in a short period of time. In the case of a smaller ontology being integrated into a larger ontology, there is a possibility of reusing an ontology of time present in a larger model. This approach, however, requires a smooth integration.

There are a number of general classes of concept that are represented in almost all ontologies: things, processes and events, relations, and properties.

Top-Level Division

Among the most interesting questions pertaining to ontology organization are, 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 versus Informational in the latter). The Tangible versus 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 an intangible extent. Person category can be such an example: A person's body constitutes the physical extent, and a person's mind is the intangible extent. In UMLS, Entities are divided into Physical Objects and Conceptual Entities, that is, also along the Physical versus 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

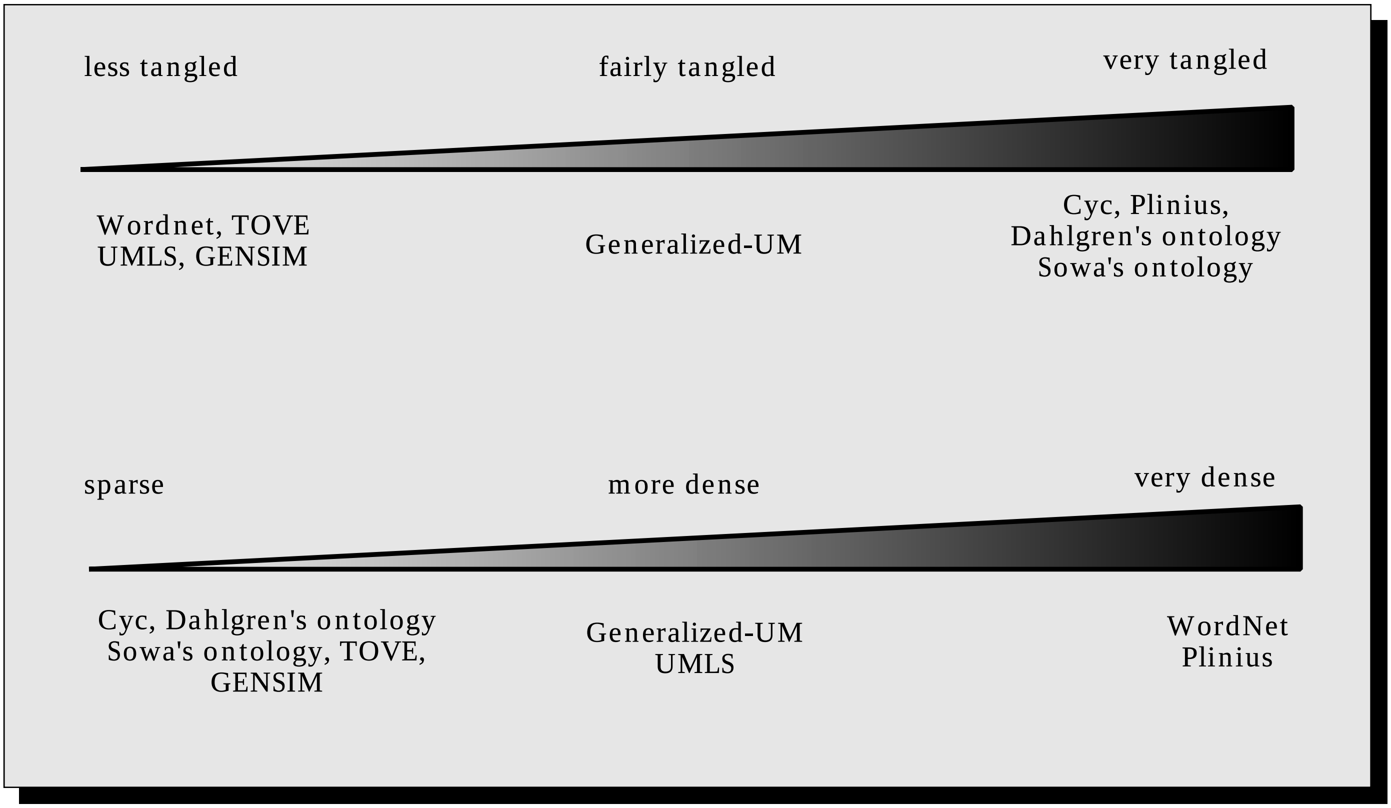

Figure 11. Comparison of How Tangled and Dense the Ontologies in the Study Are.

pronounced at all levels in CYC, and Lenat and Guha (1990) devote a lot of discussion to this issue.

The lack of correspondence between ontologies in their top-level division poses an obstacle to the integration of different ontologies. Campbell and Shapiro (1995) discuss the idea of a mediation interface, which will translate statements made in one ontology to another ontology. The authors compare top levels of a number of ontologies 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 treelike 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 subcategories that 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 11 illustrates the tangledness and density scales of the projects that we studied. They vary from relatively sparse but very tangled hierarchies such as CYC to much more dense but less tangled hierarchies such as WORDNET.

Internal Concept Structure and Relations between Concepts

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 level is the internal concept structure. Other relations among concepts (spatial, functional, and so on) can also be represented. CYC, Dahlgren's ontology, Sowa's ontology, GENSIM, GENERALIZED UPPER MODEL, 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. The structure of its objects is defined by a set of construction rules that specify the internal composition of concepts.

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 predefined 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.

When studying relations between categories in the ontologies, we found one relation represented very differently in the ontologies and often was not adequately dealt with. This is the issue of part-whole relations. There are several types of part-whole relation that can require different reasoning. For example, Winston, Chaffin, and Hermann (1987) differentiate 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), and 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, and so on. Part-whole relations are handled just like other roles or relations. However, GENERALIZED UPPER MODEL, Sowa's ontology, and TOVE provide some analysis of part-whole relations. Here is how each of the systems does it.

GENERALIZED UPPER MODEL: A 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:

First, *consists-of* is 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 the parts of the whole. For example, a protein consists of amino acids, but a car does not consist of an engine.

Second, *constituency* is 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.

Third, *ingrediency* 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 ExternalRole for the things that play a role with respect to something outside themselves). InternalRole is subdivided into several categories: When a Continuant versus Occurrent distinction is applied to InternalRole, it produces ObjectPart and ProcessPart. Object parts that can exist independently of the object are called *pieces* (for example, engine of a car); those that cannot are called *attributes* (for example, size, weight, color of a car). For ProcessPart, the same distinction leads to *participant* (for example, a book and a reader in a reading process) and *manner* (for example, speed of the wind, style of a dance).

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 that cannot be further subdivided into components), *Composite Conjunct Part* (composed of two or more primitive or composite parts), and *Composite Disjunct Part* (represent alternatives of parts; that is, at any point in time, the part has only one of its components as a valid component).

Of these three categorizations, Sowa's is the most general. All the types of part-whole-relation in GENERALIZED UPPER MODEL are what Sowa calls *pieces* (which GENERALIZED UPPER MODEL goes on to classify). Attribute and process participants and manner are not considered parts in TOVE and GENERALIZED UPPER MODEL (note that although a part-of relation in GENSIM is generalized to include processes, part in this case is a subprocess of a process). TOVE's approach to part-of relation is different because it reflects the way an artifact is composed from its parts.

WORDNET and PLINIUS also single out the part-of relation. In fact, this relation between concepts (beyond taxonomy) is the primary one in the two systems: This relation is the only one 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, and consists-of.

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 formal-

… 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.

Strengths and Contributions	**Ontologies**
Content creation	CYC, UMLS, GENERALIZED UPPER MODEL, 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 UPPER MODEL
Extended hierarchy of relations	GENERALIZED UPPER MODEL
Novel approach to ontology design	PLINIUS (conceptual construction kit)
Comparison of different knowledge representation formalisms	PLINIUS
Methodology for formal evaluation	TOVE
Use of ontology for simulation	GENSIM

Table 4. Summary of Major Strengths and Contributions of the Ontologies in the Study.

ism, having roles and properties represented by slots of a frame. One can also express the same facts using axioms. A taxonomy, too, can be represented using axiomatic notations. For example, the axiom where *a* is an instance of a category, and *A* and *B* are categories states that *A* is a subcategory of *B*. Here, we are only 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, and KIF. GENERLAIZED UPPER MODEL has all the axiomatic information incorporated in the natural language–processing 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 the instances are of going beyond first-order logic. Some systems use *defaults*, or ways of expressing modals and uncertain facts. Some do not do 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 UPPER MODEL and TOVE are examples of this approach. UMLS also has hierarchy-of relations, but because it does not have axioms, the first-order–logic issue is irrelevant for it. Another common example of going beyond first-order logic is the use of defaults, for example, CYC and KIF. For KIF, this is the only instance of going beyond first order because it is based primarily on first-order logic. GENSIM uses override inheritance in its process hierarchy (property values in a subclass can override the corresponding values in a superclass), which leads to nonmonotonicity too. Sowa uses conceptual graphs, which themselves use higher-order logic.

Here are some cases where CYC goes beyond first-order logic (Lenat 1995):

Certainty: Each assertion is assumed true by default, but one can make statements such as "assertion *A* is less likely than assertion *B*."

Reification: A predicate or function is turned 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 can be true only in a particular context. Contexts are first-class objects in CYC; for example, "You cannot see someone's heart" (true only as a default but not true during heart surgery).

Applications

An important way of evaluating the 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 that ontologies are used 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 UPPER MODEL 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 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 information-retrieval 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 a person's interests, family, job, and so on. 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 created and used for simulation of metabolic pathways, DNA transcription, and so on. The primary goal of TOVE is to model a virtual company and provide a test bed for research into enterprise integration.

From other classes of applications, UMLS was used to implement the internet grateful med interface to MEDLAB databases (developed by the National Library of Medicine). 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.

Conclusion

To conclude, we considered major strengths and contributions of each project, such as content creation, well-defined formalism creation, and some novel approach to ontology design. We also outlined weaknesses of particular projects. Major strengths and contributions of the projects in the study are summarized in table 4.

Many researchers in the area agree that one of the major challenges in the area of ontology design is creating the 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. Few projects span more than one language and can be applied to natural language texts in various languages. GENERALIZED UPPER MODEL is one of the multilingual projects (it is based on English, German, and Italian). One of the interesting things that was done as part of the PLINIUS Project was implementing the 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 underdeveloped area of ontology research: formal evaluation.

As for the integration of various ontologies, this study showed 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 the basic classes of object are that should be represented (for example, things, processes, relations), and how they are represented (not in terms of formalism but in terms of knowledge that should accompany the concepts).

We believe that a study such as 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.

Before real knowledge sharing and reuse will be practical, some standards should emerge in what an ontology should consist of, what the basic classes of object are that should be represented …, and how they are represented

Acknowledgments

This research is supported in part by the National Science Foundation under grants IRI-9117030 and IRI-9633661.

Note

1. Not all the data on these characteristics are available.

References

Bateman, J. A.; Magnini, B.; and Rinaldi, F. 1994. The Generalized Italian, German, English Upper Model. Paper presented at the Eleventh European Conference on Artificial Intelligence (ECAI'94) Workshop on Comparison of Implemented Ontologies, 8–12 August, Amsterdam, The Netherlands.

Campbell, A., and Shapiro, S. 1995. Ontologic Mediation: An Overview. In Proceedings of the Workshop on Basic Ontological Issues in Knowledge Sharing, 16–25. Ottawa: University of Ottawa.

Dahlgren, K. 1995. A Linguistic Ontology. *International Journal on Human-Computer Studies* 43(5–6): 809–818.

Dahlgren, K. 1990. Naive Semantics and Robust Natural Language Processing. In Proceedings of the Symposium on Text-Based Intelligent Systems. Presented at the AAAI Spring Symposium Series, 27–29 March, Stanford University.

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

Forbus, K. D. 1984. Qualitative Process Theory. *Artificial Intelligence* 24:85–168.

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. In *Toward Very Large Knowledge Bases,* ed. N. J. I. Mars, 289–296. Amsterdam, The Netherlands.

Gruber, T. R. 1993. Toward Principles for the Design of Ontologies Used for Knowledge Sharing, KSL-93-04, Knowledge Systems Laboratory, Stanford University.

Gruber, T. R. 1992. ONTOLINGUA: A Mechanism to Support Portable Ontologies, KSL-91-66, Knowledge Systems Laboratory, Stanford University.

Gruninger, M., and Fox, M. S. 1995. Methodology for the Design and Evaluation of Ontologies. Paper presented at the Workshop on Basic Ontological Issues in Knowledge Sharing, 19–20 August, Montreal, Quebec, Canada.

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

Hovy, E., and Knight, K. 1993. Motivation-Shared Knowledge Resources: An Example from the PANGLOSS Collaboration. Paper presented at the Workshop on Knowledge Sharing and Information Interchange, 28 August–3 September, Chamberry, France.

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. In *Artificial Intelligence and Molecular Biology,* ed. L. Hunter, 289–325. Menlo Park, Calif.: AAAI Press.

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

Lenat, D. B. 1990. CYC: Toward Programs with Common Sense. *Communications of the ACM* 33(8): 30–49.

Lenat, D. B., and Guha, R. V. 1990. Building Large Knowledge-Based Systems: Representation and Inference in the CYC Project. Reading, Mass.: Addison-Wesley.

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

Sowa, J. F. 1997. Knowledge Representation: Logical, Philosophical, and Computational Foundations. Boston, Mass.: PWS Publishing. Forthcoming.

Sowa, J. F. 1995a. Distinctions, Combinations, and Constraints. In Proceedings of the Workshop on Basic Ontological Issues in Knowledge Sharing, 19–20 August, Montreal, Canada.

Sowa, J. F. 1995b. Top-Level Ontological Categories. *International Journal on Human-Computer Studies* 43(5–6): 669–685.

Speel, P.-H. 1995. Selecting Knowledge Representation Systems. Ph.D. diss., University of Twente, Enschede, The Netherlands.

TOVE. 1995. TOVE Manual: Department of Industrial Engineering, University of Toronto. Available at www.ie.utoronto.ca/EIL/tove/ontoTOC.html.

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): 93–155.

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. Paper presented at the Eleventh European Conference on Artificial Intelligence (ECAI'94) Workshop on Comparison of Implemented Ontologies, 8–12 August, Amsterdam, The Netherlands.

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.

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

Woolf, H. B., ed. 1981. *Webster's New Collegiate Dictionary.* Springfield, Mass.: G. & C. Merriam.

Natalya Fridman Noy is a Ph.D. student in the College of Computer Science at Northeastern University. She received her B.S. in applied mathematics from Moscow State University, Russia. She received her M.A. in computer science from Boston University. Her doctoral dissertation is on developing an ontology for intelligent information retrieval in experimental sciences. Her research interests include ontology design, semantic information retrieval, and natural language processing. Her e-mail address is natasha@ccs.neu.edu.

Carole D. Hafner is an associate professor of computer science at Northeastern University. She received her Ph.D. in computer and communication sciences from the University of Michigan and was a senior research scientist at the General Motors Research Laboratories before joining the Northeastern faculty. Her research interests include knowledge representation, information retrieval, and natural language processing. She is coeditor of the journal *Artificial Intelligence and Law and* secretary-treasurer of the International Association for Artificial Intelligence and Law. Her e-mail address is hafner@ccs.neu.edu.