In R. Green, C.A. Bean, and S.H. Myaeng (eds) *Semantics of Relationships: An Interdisciplinary Perspective*, pp. 91–110. Kluwer, 2002.

Chapter 6

Comparing Sets of Semantic Relations in Ontologies

Eduard Hovy

Information Sciences Institute
of the University of Southern California
4676 Admiralty Way
Marina del Rey, CA 90292-6695
U.S.A.
tel: +1-310-448-8731
fax: +1-310-823-6714
email: hovy@isi.edu

Abstract

A set of semantic relations is created every time a domain modeler wants to solve some complex problem computationally. These relations are usually organized into ontologies. But there is little standardization of ontologies today, and almost no discussion on ways of comparing relations, of determining a general approach to creating relations, or of modeling in general. This chapter outlines an approach to establishing a general methodology for comparing and justifying sets of relations (and ontologies in general). It first provides several dozen characteristics of ontologies, organized into three taxonomies of increasingly detailed features, by which many essential characteristics of ontologies can be described. These features enable one to compare ontologies at a general level, without studying every concept they contain. But sometimes it is necessary to make detailed comparisons of content. The paper then illustrates one method for determining salient points for comparison, using algorithms that semi-automatically identify similarities and differences between ontologies.

1 INTRODUCTION

Over the past decades, semantic relations have been the focus of much work by a philosophers, computer scientists, linguists, and others. They have developed formalisms, built computational reasoning systems, collected sets of relations, and arranged taxonomies of relations or ontologies. Thus, even for describing the same domain, it is not surprising that many different and mutually incompatible systems of semantic relations exist!

Certainly no set of relations is entirely complete, entirely right, or entirely wrong. To study their collected insight, one might then try to compare them, drawing from each that which is valuable for one's enterprise. But despite some discussion of the relative merits of the various approaches and results, there have been few proposals for empirical ways in which to compare and justify them. At present, knowledge crafters follow their own paths and derive their own results, without much regard to standardized procedures and/or methods of comparing their work to that of others.

Surely however it is not impossible to establish some methodology and/or standards for description and comparison. One could begin by investigating typical methods of creating, justifying, and validating sets of relations and ontologies in general, and then identify the primary commonalities and points of importance. From that, one could derive a list of the kinds of characteristics that are important when one describes an ontology or set of relations, and when one compares various ontologies or sets of relations to one another.

This chapter lays some foundations for doing so, focusing on ontologies as sets of relations and concepts. For generality, we define an ontology rather loosely as a set of terms, associated with definitions in natural language (say, English) and, if possible, using formal relations and constraints, about some domain of interest, used in their work by human, data bases, and computer programs. We view a set of semantic relations, organized into collections and perhaps related in a generalization hierarchy, as a special instance of an ontology. We view a Domain Model as an ontology that specializes on a particular domain of interest.

The steady emergence of new and different ontologies is making it possible to perform ontology characterization and comparison studies. In fact, one can compare at two levels: using the overall general characteristics of the ontologies, and focusing on particular differences in the ontologies' content and structure. The former level requires a framework of characteristics that covers all pertinent aspects, in terms of which one can then characterize each ontology. The latter level requires a method of identifying individual points of difference, for which one can then study differences between specific relations, concept definitions, and interrelatedness with other concepts and/or relations.

To compare ontologies, one can proceed as follows. Given two ontologies or sets of relations, one can first create the general characterizations for each, then identify overall differences, and finally identify particular points of difference between individual pairs of concepts or relations.

This chapter describes the first steps in performing this complex undertaking. Section 2 provides an overall framework of characteristics of ontologies in general, and Section 3 describes a method for identifying individual concept-by-concept differences between ontologies, using semi-automated alignment algorithms.

2 CHARACTERIZING AN ONTOLOGY

2.1 The Need to Characterize Ontologies

The first step outlined above is to characterize an ontology in terms of a general set of descriptive features. To date, only a few systems of features for describing or comparing ontologies have been proposed (Noy and Hafner, 1997; Uschold et al., 1998, Aguado et al., 1998, Jasper and Uschold, 1999; Van Zyl and Corbett, 2000). The framework described in this section was first disseminated at a AAAI Symposium in March 1997, and is offered as a stepping stone on the way toward a fuller and richer set of features. There is no claim that the features outlined here, or their organization, is the only one, or that it is the best possible one. These ideas are meant to suggest a possible avenue of attack; the major claims lie in the approach, and its minor claims in the set of features chosen at each level.

The descriptive features given here are supposed to capture the salient aspects of ontologies in a single standardized way, in order to facilitate cross-ontology comparisons (and possibly, eventually, ontology evaluation). This system recognizes our desire for simple, intuitively clear, approachable features while respecting the complexity and nonconformity of ontologies we encounter today. To allow this, the features are organized into a taxonomy of increasing differentiation and specificity, thereby bounding (on the horizontal axis) the number of (sub)cases to examine at any point—affording simplicity—while still allowing unbounded specialization on the vertical axis, downward—accommodating variability. (Eventually producing an ontology of ontologies would be a most satisfying result.)

The taxonomy of features is described top-down, for clarity of exposition. Unfortunately, this makes term definition difficult, because many terms are most easily explained simply by considering the subtaxonomy they dominate. Striking a compromise, each level includes a fairly informal, intuitive, definition of the terms involved, trusting that the reader will keep dissatisfaction in check until the subsequent level down.

2.2 A System of Features—Upper Levels

At the most fundamental level of description, one has to distinguish the most basic sets of concerns. For Ontology Engineering (as for KR in general), a useful top-level division separates Form, Content, and Usage. In this section we provide a higher-level overview and then, in Section 2.3, provide specifics.

2.2.1 Form

Form denotes the representational framework—the conceptual tools with which the ontology builder defines terms and axioms. Providing the representational substrate, it includes theoretical, notational, and computational concerns. Mainly, this area involves issues in Mathematical Logic and Computational Complexity. Form is usefully separated into two branches:

- ***Theoretical Foundation*** includes four subareas, namely the conceptual roots of the ontology, the principles of terminology design, the denotational theory used, and the treatment of microtheories (contexts).

- ***Computational Foundation*** includes two principal subareas, namely the properties and capabilities of the KR system employed and the notation or formalism used.

These two notions may be partitioned as follows:

- ***Conceptual roots*** pertains to the historical antecedents of the ontology, including all the ontologies and KR systems upon which the current ontology is based and from which design decisions have been drawn.
- ***Principles of terminology design*** include all the considerations that lead to the ontology builders' selection of terms to model aspects of the domain, such as how parsimonious to be, and on what basis to taxonomize the terms.
- ***Denotational theory*** denotes the theory/ies used to establish the formal link between the ontology and its extension in the world of the domain; the approach might be formal, such as possible worlds semantics, or it might be informal, relying on natural language glosses.
- ***Microtheories or contexts*** pertains to the creation and treatment of subareas of the ontology, or collections of axioms, that hold only in particular circumstances; for example, animals can talk only in children's stories. This aspect contains a description of the particular features of microtheories or contexts required in the ontology.
- ***Properties and capabilities of KR system*** includes all aspects of the representation and reasoning support required by the ontology of the underlying knowledge representation system, from simple capabilities such as property inheritance to complex ones such as truth maintenance.
- ***Notation (formalism)*** denotes the design of the notation used, both for defining the ontology's concepts and relations, and for representing instances in the domain by composing terms from the ontology.

The final level of detail for Form is discussed in Section 2.3.1.

2.2.2 Content

Content denotes the terms, axioms, and microtheories used by the ontology builder to model the domain of interest. Mainly, this area involves issues in Philosophy, Epistemology, and, naturally, the domain itself. Content is usefully separated into four branches: Terminology, Axioms, Inferences, and Instances.

- ***Terminology / feature organization*** pertains to the collection of terms (relations, objects, events, etc.) taken as a whole, as included in the ontology by the ontology builder; important aspects include the number of terms, organization, coverage over domain, etc. This topic may even include the terms themselves.
- ***Microtheories / contexts*** describes pertinent aspects of the different microtheories that have been defined in and for the ontology under consideration.

- ***Inferences*** includes all aspects of the inferences defined over the terms in the ontology (to the extent this is separate from the axioms). It may also include the actual inferences themselves.
- ***Individual terms or instances*** pertains to the characteristics of the ontology's representation items, taken individually. This includes the typical quantity and quality of content associated with each term, the expressive adequacy of terms when it comes to representing the domain(s) of interest, the compositionality of terms, and so on.
- ***Axioms*** includes all aspects of the domain axioms and other interrelationships defined among the terms in the ontology. It may also include the actual axioms and relations themselves.
- ***Conceptual roots*** describes the sources of the ontology content, its development history, and methods of extracting and/or creating the content.

The final level of detail for Content is discussed in Section 2.3.2.

2.2.3 Usage

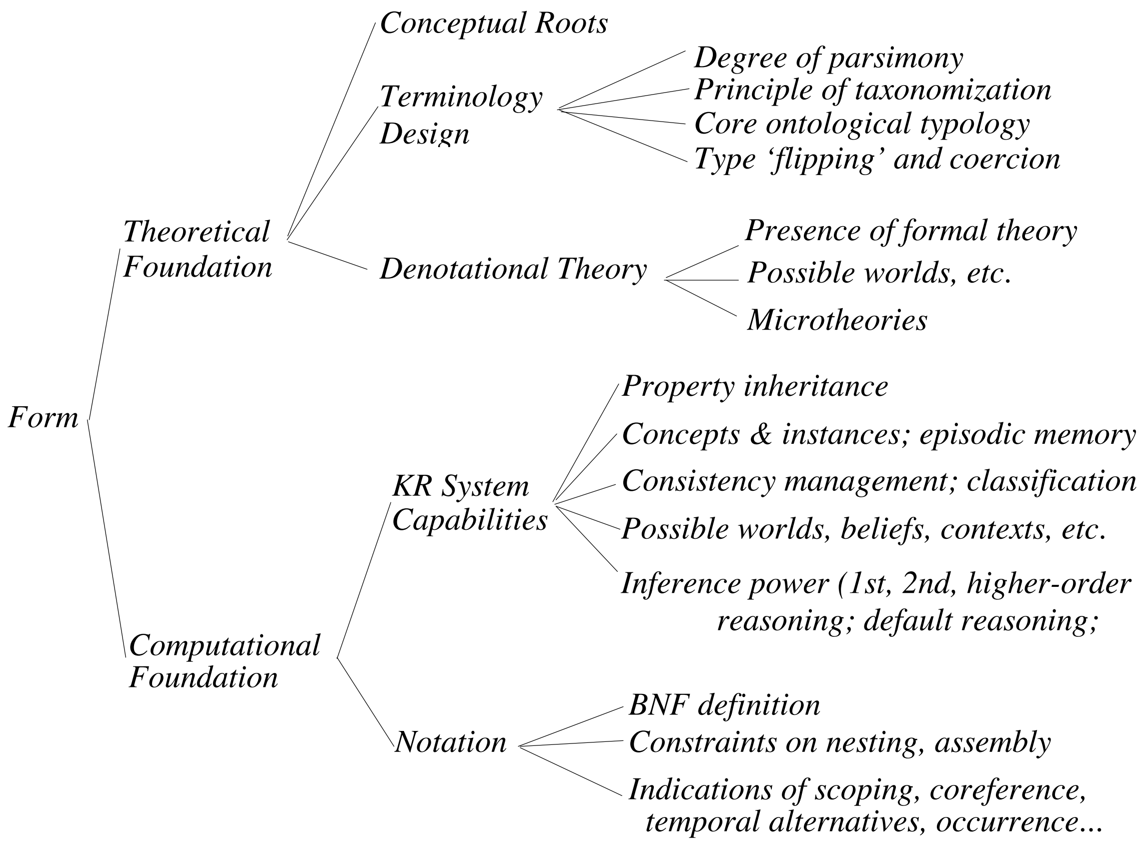
Usage denotes the manner in which the ontology builder goes about building, using, updating, and maintaining the ontology, including the software support and tools for these activities. It involves issues in Software Engineering and Ergonomics. Usage can be subdivided into the following four areas:

- ***Functionality*** pertains to all aspects of the actual and intended use(s) of the ontology, mainly in implemented systems, but also in its historical perspective, exemplifying, for instance, what has been achieved with it. .
- ***Acquisition*** describes the legal and administrative details of acquiring the ontology.
- ***Implementation*** denotes relevant aspects of the computational instantiation(s) of the ontology, such as storage size, inference speed, hardware and software required, etc.
- ***Support*** includes all aspects of ontology builder and user support, such as viewing and editing tools, error handling, documentation, etc.

The final level of detail for Usage is discussed in Section 2.3.3. It is interesting to note that the Knowledge Representation field ignored Usage as a topic of theoretical interest almost from its inception—papers discussing implementations, inference speed, etc., appear in more applied conferences only. Similarly, it factored out, and mostly ignored, issues of Content a little later, certainly by the time of Brachman's (1978) influential KL-ONE system.

2.3 Features—Detail Level

2.3.1 Features of Form



The **conceptual roots** include all the ontologies and KR systems upon which the current ontology is based and from which design decisions and content have been drawn. Knowing its derivation (and the beliefs of its builders) may help one understand the ontology itself.

The **principles of terminology design** include all the considerations that lead to the ontology builders' selection of terms to model aspects of the domain:

- the ***degree of terminology parsimoniousness***: this feature characterizes the number of the ontology terms, on the scale from parsimonious (very few terms, all 'primitive') to profligate (very many terms, some quite specific). For example, a very parsimonious term set, Conceptual Dependency, contained only 14 action primitives (Schank and Abelson, 1977). In contrast, an argument in favor of profligacy can be found in the paper *Ontological Promiscuity* (Hobbs, 1985). This feature involves what might be called the *Principle of Term Inclusion*; any careful ontology builder should be able to state the general policy being followed regarding the granularity of terms and the resulting effect on the ontology size. Also to be described is the extensibility of the term set—the ease of adding new terms, both defining them adequately to the level of the ontology and finding their proper location(s) in the ontology.
- one or more ***principle(s) of taxonomization***: the principle(s) by which the ontology's taxonomic / network structure is derived, and by which the placement of

new terms is justified and tested. In the Pangloss Ontology Base, for example, the principle is derived from the English grammar encoded in the Penman sentence generator (Bateman et al., 1989); in WordNet, the principle seems to be based on what might be called *naive semantics* with a Cognitive Science bent (Fellbaum, 1998); in the CYC ontology, the principle seems to be based on intuitions guided by Artificial Intelligence inference concerns (Lenat, 1995).

- the ***core ontological typology***, namely the major ontological types, and the theory of the domain that underlies them. Included here are questions about States vs. Events vs. State-Changes vs. Processes; Relations as States or not as States; the relationships between object and event (see 'type flipping' below). A popular approach (taken, for example, by Aristotle), starts with a given list of the basic types that function as the roots of the forest of trees forming the ontology. Another approach is to identify a set of very basic facets of meaning (e.g., concrete-abstract, positive-negative, changing-static) whose various permutations form the starting point for ontological differentiation (Wille, 1997; Sowa, 1999). Despite many interesting theories about the basic ontological types, little formal agreement exists, although a fairly standard general practical approach seems to be emerging.
- the ***standard for 'type flipping' and coercion***: regardless of particular decisions on the core ontological typology, it seems to be a fact about the world and the way people model it that certain concepts receive treatment sometimes as one basic type and sometimes as another. Often the concept is sufficiently different in its alternate guise to merit being called a distinct (but still somehow related) concept. Various classes of 'type flipping' can be differentiated. In metonymy, an invalid type is substituted for another (for example, "In a press statement, the Senate announced that...", or "Last year, London attracted 10 million visitors"—here *Senate* stands for a spokesperson, since the building obviously cannot make announcements, and *London* for the social organization that is the city, not its physical location, Mayoral office, etc.). In type promotion and demotion, some portion of a representation is moved to a different position with respect to the whole, possibly causing changes in terms used (for example, "Mike swims across the river" may initially be represented as a *swim* process but then be changed to a *move* or even a *move-across* process whose manner is *swimmingly*—this kind of change is common in shallow semantic representations in Machine Translation when different languages express different aspects of the same idea (Dorr 1994)). Though such type flipping is readily apparent in linguistic applications, it appears also in AI reasoning. For example, some concepts can be treated as either objects or processes (e.g., are *picnic, parade,* and *explosion* events or objects? How do *explosion* and *to explode* relate? Is a *street corner* a location or an object? And a *city*?). Ontology builders should state their policy on the treatment of type coercion, since this affects their approach to ontology creation.

The **denotational theory** denotes the theory/ies used to establish a formal link between the ontology and its extension in the world of the domain. Several aspects are important, including:

- ***presence of logically rigorous denotational system***: Possible alternatives range from 'informal' (i.e., without such a system) to formal, with the former used more in implemented systems and the latter developed mainly by (and for?) theoreticians. In 'informal' approaches, the world-extension of a representational item is generally simply inferred from its name, which is a carefully chosen natural language word; the dangers of this approach were tellingly pointed out a long time ago by McDermott (1981). Formal approaches include work by Montague (1974), Davidson (1967), Barwise and Perry (1986) on situational semantics, and others.
- the possibilities for representing distinct ***contexts or worlds*** of various kinds, including possible worlds.
- ***microtheories***: the facility of encapsulating a set of terms, relations, and/or axioms into a system distinct from the rest of the ontology, and applied to the world under specific conditions only (for example, in childrens' stories, animals can talk). The CYC ontology supports microtheories for inference (Lenat, 1995); MIKROKOSMOS for Machine Translation (Mahesh, 1996; Nirenburg et al., 1995).

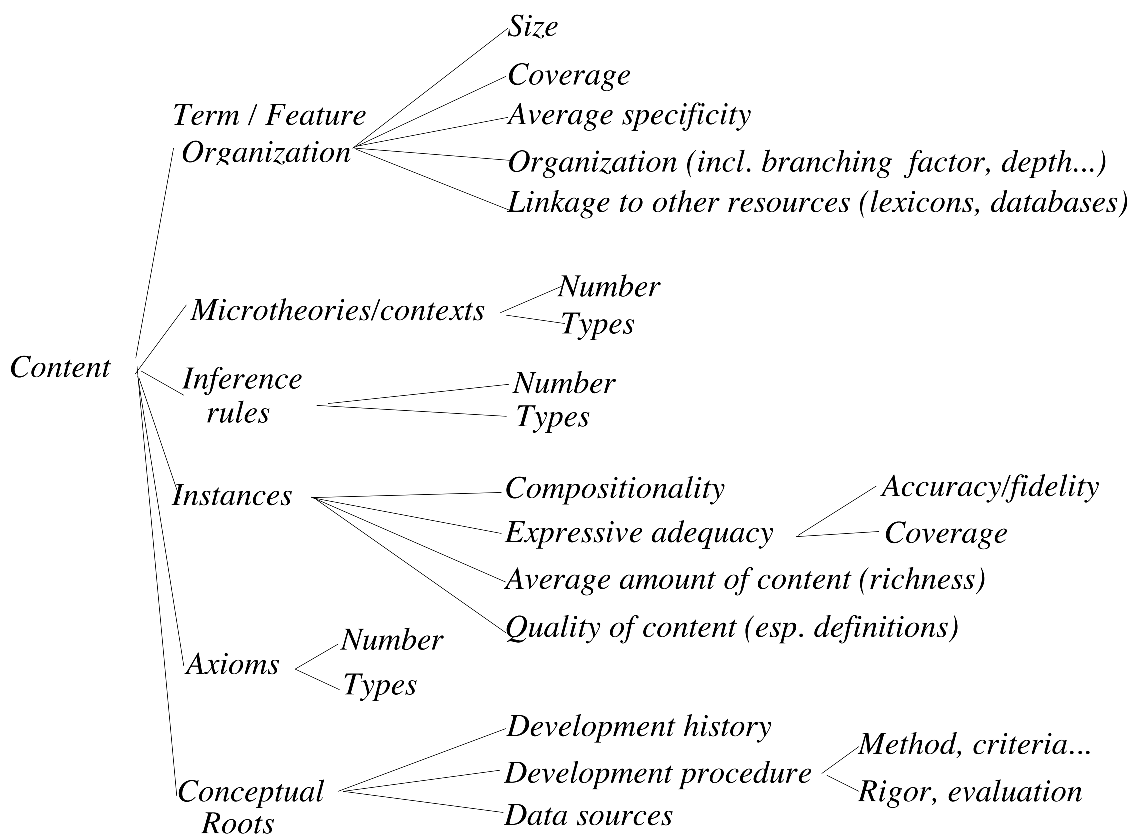
The **properties and capabilities of the KR system** include the representation and reasoning support required by the ontology of the underlying knowledge representation system. It is obviously closely related to the theoretical foundation. Since this area is one of the mainstays of KR research, a large number of such properties and capabilities have been discussed and implemented, including:

- the treatment of ***property inheritance***: Is it supported? Is it strict (i.e., no violations allowed) or not (i.e., overrides allowed)? How are the Nixon diamond and similar problems handled?;
- the ***differentiation of concepts and instances***, which are separated in KL-ONE and successors like Loom (MacGregor, 1991) and CLASSIC (Borgida et al., 1989; Brachman et al., 1999) into the T-Box and A-Box, but not so separated in for example CYC or the name taxonomies of the Yahoo! and Lycos internet search engines;
- the ***support of inference*** involves both the types supported (active/forward inference, or passive/backward inference, or none?), and includes the presence and necessity of additional capabilities such an automated concept classifier (MacGregor, 1991);
- the possibilities for representing ***contexts or worlds*** of various kinds, including possible worlds, microtheories, etc., with details;
- the necessity of or support for a ***truth maintenance*** system, and its details;
- the necessity of or support for a ***time management*** system, and its details;
- the necessity of or support for various forms of ***non-first-order reasoning,*** including second- and higher order and modal reasoning;
- etc.

The **notation (formalism*)*** denotes the design of the notation used to represent instances in the domain by composing terms from the ontology. Many alternatives, often simply notational variants that differ in the placement parentheses and other markings, have been developed; two major families are the frame-style representations and the axiom-style ones. Several characteristics must be described, including:

- a ***BNF description*** of the notation;
- any constraints on the ***nesting and assembly*** of representation instances, including a description of equivalent variant forms, if the notation allows this;
- the notations used for ***indicating various special situations***, including scoping, coreferences (repeated occurrences of the same entity), temporal alternatives (the same entity at different times), actual vs. potential occurrences of events in the domain, and so on;
- etc.

2.3.2 Features of Content



The **terminology and feature organization** includes all aspects of the actual terms included in the ontology, among others:

- ***size*** of ontology (number of terms);
- ***coverage*** over the domain (percentage of concepts identified in the domain that are explicitly represented by terms in the ontology);
- ***richness*** of terminology, measured in various ways (for example, number of predications per term, or total number of interconnections among terms). Terminological Ontologies, such as SENSUS (Knight and Luk 94) contain mostly

terms organized hierarchically, and are less rich (but usually much larger) than Domain Models;

- ***organization*** of terminology, which may be a simple tree taxonomy or an arbitrarily complex network. If it is a simple taxonomy, or a set of parallel taxonomizations over the same terms, then the taxonomizing principle(s) must be described (see under Terminology Design above). Often, the organizing principle is some variant of conceptual generalization, using *isa* or *subclass* and *instance-of;* it might (as in WordNet (Fellbaum, 1998)) also be *part-of, synonym-to, antonym-to,* etc. Various subsidiary measures provide additional information, including the **average branching factor**, the **average depth** of each major portion of the ontology, etc.;
- average level of ***specificity*** of the terms. Terminological Ontologies used for natural language applications tend to be more general (high-level, abstract), especially such language-related ontologies as the Penman Upper Model (Bateman et al., 1989), while Domain Models used for domain-oriented applications are naturally more specific;
- ***linkages***, if any, from terms into other resources, such as lexicons of various languages (as typically used in Machine Translation (Knight and Luk, 1994)), termlists acquired from outside agencies (often present in ontologies modeling database contents), etc.;
- the set of ***terms*** themselves.

The **microtheories** that included the ontology—their ***number*** and ***types***—are listed and described.

Domain **inference rules** includes all aspects of the inferences defined over the terms in the ontology (to the extent this is separate from the axioms), including:

- the ***number*** of inference rules defined;
- the ***inferential capabilities*** required or permitted of the inference rules: whether, for example, inferences may create new instances, delete them, add properties and other relations, change their status from (for example) actual to potential, etc.;
- the actual ***inferences*** themselves.

Instances comprises the aspects of representational instances created for actual domain entities existing at specific times with states of being. This includes the following:

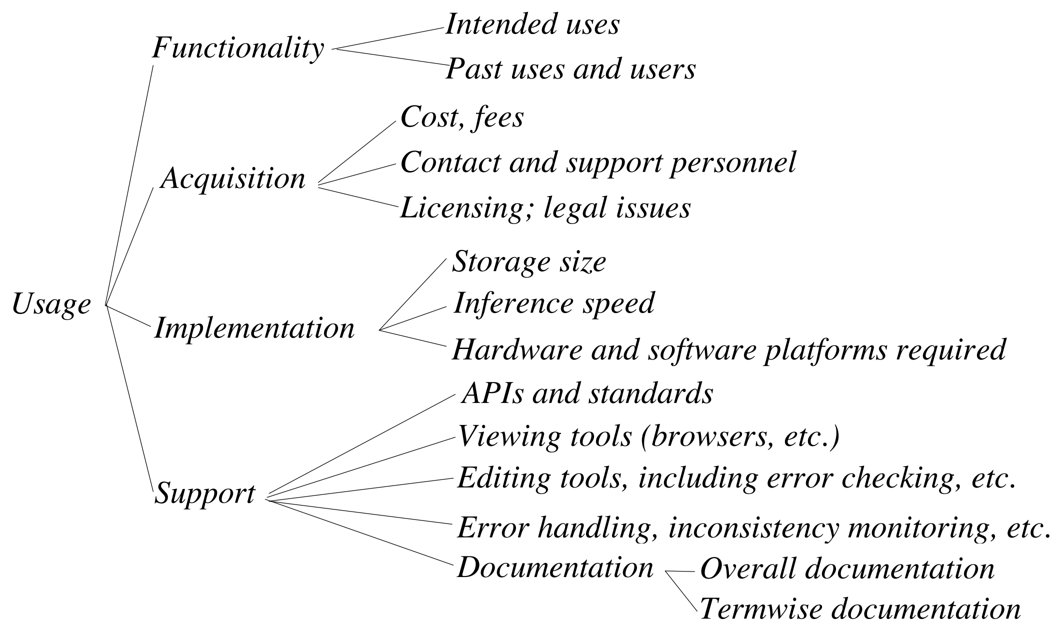
- ***expressive adequacy***, which involves both the ***fidelity*** or correctness of the instance (that is, the degree to which it represents what the ontology builder or system user believes to be true about the actual instance in the domain—*the truth, and nothing but the truth*), and the ***coverage*** or completeness of the instance (that is, the degree to which it represents all the facts about the instance in the domain—*the whole truth*);
- ***compositionality***: the ease of composing instances, either anew or out of other instances, the readability of instances, ease of access to their parts, etc.;
- etc.

The **axioms** of the ontology, including any other interrelationships defined among the terms, involve the following aspects:

- the ***number*** of axioms and/or relationships defined;
- the ***types*** of axioms and relations allowed in the ontology;
- the actual ***axioms and relations*** themselves.

Conceptual roots lists the ***sources*** of all the content terms themselves, and describes their ***history of development*** as well as the ***procedures*** by which they were co-opted, altered, merged, etc.

2.3.3 Features of Usage



Functionality pertains to all aspects of the actual and intended use(s) of the ontology, mainly in implemented systems, but also in its historical perspective, exemplifying, for instance, what has been achieved with it. Listed here are both the ***intended uses***, which can play a major part in one's attitude toward the utility of an ontology; see (Van Zyl and Corbett, 2000; Ushold et al., 1998), for example, and ***past and present users***, namely those to whom one can turn for further information, experience, etc.

Acquisition describes the legal and administrative details of acquiring the ontology. This includes ***licensing***, ***cost***, etc.

Implementation denotes relevant aspects of the computational instantiatation(s) of the ontology. This includes:

- ***storage size***, both of ontology when stored on disk and when loaded in core;
- ***inference speed***, and speed of other computations on the ontology;
- hardware and software ***platforms required*** by the ontology's KR system(s);
- etc.

Support includes all aspects of computational support for the ontology builder and user, including:

- ***APIs and standards***: a description of the degree and rigor of standardization of the ontology as data;
- ***viewing tools***: the presence and quality of tools such as browsers, ontology prettyprinters, Web-based interfaces, etc.;
- ***editing tools***, with or without version control, automated error checking, etc.;
- ***error handling***, including inconsistency monitoring;
- ***documentation***: its presence and quality;
- etc.

3 COMPARING CONCEPTS OR RELATIONS SEMI-AUTOMATICALLY

Comparing two sets of relations or two ontologies using the framework just described, one can create a list of some global differences between them. Often, however, it is most useful to frame one's questions in terms of specific point-by-point differences between individual concepts or relations. When one has identified alternative (equivalent, or somewhat equivalent) concepts or relations, one can use especially the features of Content to understand the details of variation.

However, when presented with two ontologies or sets of relations, each containing several thousand items, how does one find the (near-)equivalent ones? One cannot manually compare each concept in one ontology to all other concepts in the other; this approach may require $M \times N$ comparisons (for M and N items respectively in the two ontologies); if each comparison takes just one minute, two 5,000-item ontologies would require over 200 person-years. An automated search is clearly required, if only to pinpoint likely candidate equivalent items. But even if each concept or relation is fully specified by logical propositions, what search mechanism can be conceived to 'understand' the definitions well enough?

In recent work, several ontology building projects have build interfaces to assist with the manual alignment and merging of ontologies. Typically, these tools extend ontology building interfaces such as those of Ontolingua (Farquhar et al., 1997) and the Stanford CML Editor (Iwasaki et al., 1997) by incorporating variants of matching heuristics plus several validation routines that check for consistency of edited results (McGuinness et al., 2000; Noy and Musen, 1999a; 1999b). Given the large numbers involved, however, we are interested in automating the alignment process.

This section outlines one experiment in automatically identifying potentially (near-) equivalent concepts or relations, using several techniques to each provide contributory evidence, which can then narrow down the amount of comparative work a human analyst has to perform. By applying this method and discarding invalid suggestions, the analyst can more quickly and easily produce a list of purportedly equivalent concepts or relations, whose differences can then be studied one by one. Section 3.1 outlines the context of cross-ontology comparison work; Section 3.2 describes the alignment technique, and Section 3.3 provides some results.

3.1 Background: Searching for a Standard Reference Ontology

In a world of increasing computerization, it is inevitable that different computer programs represent the same basic entity in different ways. The concept *Person*, for example, connotes something different to a medical specialist (and hence to his or her computer program) than it does to a Census Bureau worker (and his or her program) or a video game player. But in many of these applications, much of the information associated with the entity is the same. To the extent that establishing a single, large, standard ontology of terms is feasible, it would hold at least the following advantages: standardization of terms, easier knowledge transfer to new domains, and better interoperability between computer programs.

Recognizing these advantages, members of several ontology projects and others interested in the issues formed an Ad Hoc Group on Ontology standards in 1996. Meeting roughly twice a year, the group became a subcommittee of the ANSI Committee X3T2, later called NCITS. It included representatives from various universities (including Stanford University and the Natural Language Group at USC/ISI), research laboratories (including LADSEB in Italy and Lawrence Berkeley Laboratories in California), companies (including EDR in Tokyo, CYC in Texas, IBM in California, and TextWise in New York), U.S. Government officials, and private individuals.

The group sought to create a short document to serve as a model ontology standard, and to create a small illustrative example of such a standard, called the Reference Ontology. As described in Section 3.2, the group commissioned the author to create early versions of the topmost portions of this Reference Ontology out of the top regions of the SENSUS, MIKROKOSMOS, and CYC ontologies. Although no Reference Ontology exists today, the general technique and the heuristics are of interest for this chapter.

SENSUS (Knight and Luk, 1994) was built at the Information Sciences Institute of the University of Southern California (USC/ISI). SENSUS currently contains approx. 90,000 terms, linked together into a subsumption (*is-a*) network, with additional links for part-of, pertains-to, and so on. SENSUS is a rearrangement and extension of WordNet (Fellbaum, 1998) (built at Princeton University on general cognitive principles), retaxonomized under the Penman Upper Model (Bateman et al., 1989) (built at USC/ISI to support natural language processing). For most of its content, SENSUS is identical to WordNet 1.5; its primary structuring is hence based on cognitive grounds. It can be characterized as a 'shallow', lexically oriented, term taxonomy.

SENSUS can be accessed publicly via the Web, using the Ontosaurus browser (Swartout et al., 1996), at *http://mozart.isi.edu:8003/sensus/sensus_frame.html*. SENSUS has been used to serve as the internal mapping structure (the Interlingua termbank) between lexicons of Japanese, Arabic, Spanish, and English, in several projects, including the GAZELLE: machine translation engine (Knight et al., 1995) and the SUMMARIST: multilingual text summarizer (Hovy and Lin, 1999). The lexicons contain over 120,000 root words (Japanese), 60,000 (Arabic), 40,000 (Spanish), and 90,000 (English), and 90,000 (Bahasa Indonesia), of which various amounts have been linked to SENSUS. SENSUS terms serve as connection points between equivalent language-based words.

MIKROKOSMOS (Mahesh, 1996), built at the Computing Research Laboratory of the New Mexico State University (NMSU), contains 4790 concepts. MIKROKOSMOS was designed to support interlingual machine translation between a variety of languages. The Interlingua definition symbols are housed in MIKROKOSMOS, together with internal interrelationships and property constraints. MIKROKOSMOS is thus somewhat 'richer' (in the sense of Section 2.3.2) than SENSUS, and its primary structuring is oriented toward supporting the linguistic generalizations that support language processing. Its concepts, also taxonomized under the *is-a* relation, partially resemble the upper portions of SENSUS, although they contain many additional abstractions. Many of the MIKROKOSMOS concepts are not lexicalizable in English by a single word.

The CYC Ontology (Lenat, 1995) has been under development by CYCorp Inc., Texas, since the mid-1980s, in an ambitious attempt to create a single large ontology in terms of which one can model the whole world. At present, the full CYC ontology contains about 40,000 concepts and over 300,000 axioms (inter-concept relations and constraints), and is used primarily to support metadata schemas for large pharmaceutical databases. CYC's concepts are organized according to two relations, *isa* and *genls*, that express two different types of subsumption (essentially *element-of* and *subset-of*); this approach avoids many problems experienced by less sophisticated ontologies. The CYC ontology is intended to support general conceptual inference for Artificial Intelligence systems; it contains semantic models of time, space, matter, and other experiential phenomena. It is not oriented toward supporting language-based applications; the majority of its concepts are not lexicalizable by single English words. There is no stated theory of semantics or systematic methodology of semantic modeling. Still, the CYC ontology is the 'richest', most semantic, of the three ontologies discussed here. Currently available to the public are only the topmost 2,500 concepts of the CYC ontology (see http://www.cyc.com), but no inter-concept relations other than the two for subsumption.

3.2 Aligning Ontology Terms

To fuse together the topmost regions of CYC (2500 concepts) and SENSUS (350 concepts) in 1996, several cross-ontology alignment and validation heuristics, described in this section, were developed and used (Hovy, 1996; Lehmann, 1997; Chalupsky, 1996). In 1997, the author extended the heuristics to also align MIKROKOSMOS (4800 concepts) to the topmost region of SENSUS (6768 concepts); see (Hovy, 1998).

The technique used combines the heuristics in a repeated alignment/integration cycle, which employs:

1. three heuristics (NAME match, DEFINITION match, and TAXONOMY match) that make initial cross-ontology alignment suggestions[1],
2. a function for combining their suggestions,
3. several alignment validation criteria and heuristics,
4. an evaluation metric.

Despite the apparent difficulty of having machines 'understand' concepts well enough to suggest alignments, the heuristics described here perform well enough to identify a workable number of concept pairs.

In aligning the MIKROKOSMOS ontology with the top region of SENSUS, the number of possible alignment arrangements is enormous: 6768! / 4790! Without any limitation of search, the number of concept pairs that each alignment heuristic has to consider is over 32.4 million. The alignment suggestion cycle was repeated 5 times. Alignment suggestions were computed once only with the NAME and DEFINITION match heuristics, at the beginning of the sequence, since these suggestions never change (the names and definitions remain the same). However, since every new alignment creates a new bridge across the ontologies, the TAXONOMY match heuristic was re-run 5 times. Its new suggestions, subsequently combined with the other heuristics' suggestions, produced the new suggestions in every run. After 5 runs, the combined heuristics extracted 883 suggestions for manual validation (= 2.72% of the total number of pairs, or 13% of the portion of SENSUS under consideration). Of these, 244 (= 27.6%) were found to be correct, 383 (= 43.4%) were incorrect, and 256 (= 30.0%) were nearly correct (no better alignment existed). Coverage is more difficult to measure; it would involve manually searching the entire MIKROKOSMOS to see how many concepts the heuristics should have picked out but did not. Informal estimates, based on browsing the MIKROKOSMOS files and seeing if plausible-looking concepts were picked up for suggested alignment, place the coverage performance fairly high, at over 85%.

Despite the preliminary nature of this work (ongoing research focuses on new alignment suggestion heuristics, more sophisticated validation techniques, and useful tools, such as browsers and editors), we found it very useful during the alignment to be able to focus on only about 13% of the SENSUS terms, and of these, to find just over half to be correct or nearly correct.

3.3 Findings of Alignment

Correct alignments occurred when all three heuristics (name, definition, and taxonomy match) combined well[2]:

---

[1] Other alignment heuristics have been investigated between ontologies (Agirre et al., 1994; Ageno et al., 1994; Rigau and Agirre, 1995; Agangemi et al., 1998) and between dictionaries and ontologies (Knight and Luk, 1994; Okumura and Hovy, 1994).

[2] SENSUS concepts are prefixed with 'S@' and named within upright bars; MIKROKOSMOS concepts are prefixed with 'M@'. The three heuristics' scores ranged as follows: 0≤NAME<150; 0≤DEF<15; 0≤TAX<1.

|S@foodstuff<food| = a substance that can be used or prepared for use as food
superconcepts: (|S@food|)

|M@FOODSTUFF| = a substance that can be used or prepared for use as food
superconcepts: (|M@FOOD| |M@MATERIAL|)

(COMBINATION SCORE = 13.35, NAME = 91, DEF = 10.00, TAX = 0.14)

|S@change of location,move| = the act of changing your location from one place to another
superconcepts: (|S@MOTION-PROCESS|)

|M@MOTION-EVENT| = a physical-event in which an agent changing location moves from one place to another
superconcepts: (|M@CHANGE-LOCATION|)

(COMBINATION SCORE = 4.59, NAME = 26, DEF = 4.50, TAX = 0.20)

Sometimes, correct alignments caused errors to be uncovered: is an archipelago land or sea?

|S@archipelago| = many scattered islands in a large body of water
superconcepts: (|S@dry land|)

|M@ARCHIPELAGO| = a sea with many islands
superconcepts: (|M@SEA|)

(COMBINATION SCORE = 1.522, NAME = 131, DEF = 1.33, TAX = 0.00)

However, when studying differences in reasoning between domain modelers and ontologizers, completely equivalent (i.e., correctly aligned) concepts and completely unrelated concepts are not really as interesting as the alignments that are nearly correct. Why do differences exist? How would ontology builders defend (or change) their decisions? How can the reasoning they went through be systematized and used later to guide ontology creation?

Very often, and interestingly, it was quite difficult to decide whether an alignment suggestion was in fact correct, nearly correct, or wrong:

|S@library>bibliotheca| = a collection of literary documents or records kept for reference
superconcepts: (|S@aggregation|)

|M@LIBRARY| = a place in which literary and artistic materials such as books periodicals newspapers pamphlets and prints are kept for reading or reference an institution or foundation maintaining such a collection
superconcepts: (|M@ACADEMIC-BUILDING|)

(COMBINATION SCORE = 2.74, NAME = 59, DEF = 3.57, TAX = 0.00)

|S@geisha| = a Japanese woman trained to entertain men with conversation and singing and dancing
superconcepts: (|S@adult female| |S@Japanese<Asian|)

|M@GEISHA| = a Japanese girl trained as an entertainer to serve as a hired entertainer to men
superconcepts: (|M@ENTERTAINMENT-ROLE|)

(COMBINATION SCORE = 1.54, NAME = 46, DEF = 2.27, TAX = 0.00)

|S@man<soul| = the generic use of the word to refer to any human being: "it was every man for himself"
superconcepts: (|S@PERSON|)

|M@HUMAN| = homo sapien
superconcepts: (|M@PRIMATE|)

(COMBINATION SCORE = 1.43, NAME = 19, DEF = 0.00, TAX = 0.33)

All three |S@library>bibliotheca|, |S@geisha|, and |S@man<soul| seem to align well to their partners, until one compares their respective superconcepts in SENSUS and MIKROKOSMOS, which focus on different aspects of their nature. A library is *both* a collection of books and a building that houses such a collection; the word can be used to mean both quite naturally, even in the same sentence. A geisha is an adult female with the role of entertainer, a relationship the MIKROKOSMOS builders considered adequately modeled by the use of *is-a*. A human is certainly both a person and a primate, but |S@PERSON| had no equivalent within MIKROKOSMOS (other than, perhaps, |M@HUMAN|, again).

The result of such subsumptive ambiguities has spurred some recent work on what can be called ‘concept sense differentiation’. Both in the area of philosophy of knowledge representation (Guarino, 1997; 1998) and of computational lexicography (Pustejovsky, 1995) attempts have been made to distinguish the major kinds of interpretations one can bring to bear on objects. Guarino identifies seven perspectives and Pustejovsky four so-called qualia—for example, one can view an object as a lump of material, as a spatial structure, by its function (if it has one), as a living entity (if it is such), as a social organism (if it is such), and so on. To Guarino, a set of so-called identity criteria can be used to distinguish which sense(s) are relevant at any time: when you smash a glass, is the result still the same glass, for current purposes? Materially, it is; but structurally and functionally, it is obviously not. When after 200 years an organization consists of an entirely new membership, is it still the same organization? Socially, it is; but according to its ‘material’, its parts, it is not. Taking this approach, |S@geisha| is physically/structurally an |S@adult female| *and is also* functionally an |M@ENTERTAINMENT-ROLE|, for example. (One should not, of course, take this approach too far; although as Heraclitus observed you never step into the same stream twice, one needs to determine carefully which criteria to adopt!)

By carefully comparing the relationships of each aligned or nearly aligned concept with its neighbors in both ontologies, differences of ontological interpretation can be found. Sometimes, as shown above, they are mere errors. Other times, they point to interesting questions of semantic modeling (Visser et al., 1998). Then it becomes appropriate to ask the two ontology builders why such a difference might exist, and how they might motivate or justify their modeling decisions. In particular, such questions may prompt the development of new 'concept senses' and identity criteria.

4 CONCLUSION

As more, and more elaborate, ontologies of all kinds—large-scale cross-domain concept ontologies and detailed and rich Domain Models—become available, and as we start using, sharing, and comparing ontologies, the need for characterizing ontologies and for establishing some methodology, to the extent possible, becomes increasingly important. Only when we have done so will Ontology Engineering evolve from an art into a science.

5 ACKNOWLEDGMENTS

Many thanks to Bruce Jakeway for a lot of work in Section 3, and to comments from Michael Gregory, Graeme Hirst, Fritz Lehmann, Dagobert Soergel, John Sowa, and Mike Uschold for comments on Section 2, and to Steffen Staab and the editors for general comments.

6 REFERENCES

Agangemi, A., D. Pisanelli, and G. Steve. (1998). Ontology Alignment: Experiences with Medical Terminologies. In *Proceedings of the International Workshop on Formal Ontology in Information Systems*, held in association with the KR-98 Conference. Trento, Italy.

Ageno, A., I. Castellon, F. Ribas, G. Rigau, H. Rodriguez, A. Samiotou. (1994). TGE: Tlink Generation Environment. *Proceedings of the 15th COLING Conference*. Kyoto, Japan.

Agirre, E., X. Arregi, X. Artola, A. Diaz de Ilarazza, K. Sarasola. (1994). Conceptual Distance and Automatic Spelling Correction. *Proceedings of the Workshop on Computational Linguistics for Speech and Handwriting Recognition*. Leeds, England.

Aguado, G., A. Ba1on, J. Bateman, S. Bernardos, M. Fernández, A. Gómez-Pérez, E. Nieto, A. Olalla, R. Plaza, and A. Sánchez. (1998). ONTOGENERATION: Reusing Domain and Linguistic Ontologies for Spanish Text Generation. *Proceedings of the ECAI Workshop on Applications of Ontologies and Problem Solving Methods*, 1–10. ECAI Conference. Brighton, England.

Barwise, J. and J. Perry. (1986). *Situations and Attitudes*. Cambridge: MIT Press.

Bateman, J.A., Kasper, R.T., Moore, J.D., and Whitney, R.A. 1989. A General Organization of Knowledge for Natural Language Processing: The Penman Upper Model. Unpublished research report, USC/Information Sciences Institute, Marina del Rey, CA. A version of this paper appears in 1990 as: Upper Modeling: A Level of Semantics for Natural Language Processing. *Proceedings of the 5th International Workshop on Language Generation*. Pittsburgh, PA.

Borgida, A., R.J. Brachman, D.L. McGuinness, and L. Alperin Resnick. (1989). CLASSIC: A Structural Data Model for Objects. *Proceedings of the ACM SIGMOD Conference*. Portland, OR, 59–67.

Brachman, R.J. (1978). A Structural Paradigm for Representing Knowledge. BBN Report no 3605, Bolt, Beranek, and Newman, Cambridge, MA.

Brachman, R.J., A. Borgida, D.L. McGuinness, and P.F. Patel-Schneider. (1999). “Reducing” CLASSIC to Practice: Knowledge Representation Theory Meets Reality. *Artificial Intelligence*.

Chalupsky, H. (1996). Report on ontology alignment validity test. Ms, USC Information Sciences Institute.

Davidson, D. (1967). Truth and Meaning. *Synthese* 17, 304–323.

Dorr, B.J. (1994). Machine Translation Divergences: A Formal Description and Proposed Solution. *Computational Linguistics* 20(4), 597–634.

Farquhar, A., R. Fikes, and J. Rice. (1997). The Ontolingua Server: A Tool for Collaborative Ontology Construction. *International Journal of Human-Computer Studies* 46, 707–727.

Fellbaum, C. (1998). (ed.) *WordNet: An On-Line Lexical Database and Some of its Applications*. Cambridge: MIT Press.

Guarino, N. (1997). Some Organizing Principles for a Unified Top-Level Ontology. New version of paper presented at AAAI Spring Symposium on Ontological Engineering, Stanford University, March 1997.

Guarino, N. (1998). Some Ontological Principles for Designing Upper Level Lexical Resources. *Proceedings of the First International Conference on Lexical Resources and Evaluation* (LREC), 527–534. Granada, Spain.

Hobbs, J.R. (1985). Ontological Promiscuity. *Proceedings of the Conference of the Association for Computational Linguistics*, 61–69.

Hovy, E.H. (1996). Semi-Automated Alignment of Top Regions of SENSUS and CYC. Presented to ANSI Ad Hoc Committee on Ontology Standardization. Stanford University, Palo Alto, September 1996.

Hovy, E.H. (1998). Combining and Standardizing Large-Scale, Practical Ontologies for Machine Translation and Other Uses. *Proceedings of the 1st International Conference on Language Resources and Evaluation* (LREC), 535–542. Granada, Spain.

Hovy, E.H. and C.Y. Lin. (1999). Automated Text Summarization in SUMMARIST. In M. Maybury and I. Mani (eds), *Advances in Automatic Text Summarization,* pp 81–98. Cambridge, MA: MIT Press.

Iwasaki, Y., A. Fraquhar, R. Fikes, and J. Rice. (1997). *A Web-Based Compositional Modeling System for Sharing of Physical Knowledge*. Nagoya: Morgan Kaufmann.

Jasper, R. and M. Uschold. (1999). A Framework for Understanding and Classifying Ontology Applications. *Proceedings of the 12th Workshop on Knowledge Acquisition, Modeling, and Management*. Banff, Canada.

Knight, K. and S.K. Luk. (1994). Building a Large-Scale Knowledge Base for Machine Translation. *Proceedings of the AAAI Conference*, 773–778.

Knight, K., I. Chander, M. Haines, V. Hatzivassiloglou, E.H. Hovy, M. Iida, S.K. Luk, R.A. Whitney, and K. Yamada. (1995). Filling Knowledge Gaps in a Broad-Coverage MT System. *Proceedings of the 14th IJCAI Conference*, 1390–1396. Montreal, Canada.

Lehmann, F. (1997). Reworked Alignment of Top Regions of CYC and SENSUS. Presented to ANSI Ad Hoc Committee on Ontology Standardization. Stanford University, Palo Alto, March 1997.

Lenat, D.B. (1995). CYC: A Large-Scale Investment in Knowledge Infrastructure. *Communications of the ACM* 38(11), 32–38.

MacGregor, R. (1991). Inside the LOOM Descriptive Classifier. *ACM SIGART Bulletin* 2(3), 70–76.

Mahesh, K. (1996). Ontology Development for Machine Translation: Ideology and Methodology (CRL report MCCS-96-292). Las Cruces: New Mexico State University.

McDermott, D.V. (1981). Artificial Intelligence Meets Natural Stupidity. In J. Haugland (ed), *Mind Design*, 143–160. Cambridge: MIT Press.

McGuinness, D.L., R. Fikes, J. Rice, and S. Wilder. (2000). An Environment for Merging and Testing Large Ontologies. *Proceedings of the Seventh International Conference on Principles of Knowledge Representation and Reasoning (KR2000*).

Montague, R. (1974). *Formal Philosophy*. New Haven: Yale University Press.

Nirenburg, S., V. Raskin and B. Onyshkevych. (1995). Apologiae Ontologia. *Proceedings of the International Conference on Theoretical and Methodological Issues* (TMI). Leuven, Belgium.

Noy, N.F. and C.D. Hafner. (1997). The State of the Art in Ontology Design. A Survey and Comparative Review. *AI Magazine 18(3), 53–74.*

Noy, N.F. and M.A. Musen. (1999a). SMART: Automated Support for Ontology Merging and Alignment. *Proceedings of the Twelfth Banff Workshop on Knowledge Acquisition, Modeling, and Management.* Banff, Canada. Also available at <http://smi-web.stanford.edu/pubs/SMI_Abstracts/SMI-1999-0813.html>.

Noy, N.F. and M.A. Musen. (1999b). An Algorithm for Merging and Aligning Ontologies: Automation and Tool Support. *Proceedings of the Workshop on Ontology Management at the Sixteenth National Conference on Artificial Intelligence (AAAI-99)*. Orlando, FL. Also available a <http://www-smi.stanford.edu/pubs/SMI_Reports/SMI-1999-0799.pdf>.

Okumura, A. and E.H. Hovy. (1994). Ontology Concept Association using a Bilingual Dictionary. *Proceedings of the 1st AMTA Conference*, 177–184. Columbia, MD.

Pustejovsky, J. (1995). *The Generative Lexicon: A Theory of Computational Semantics*. Cambridge: MIT Press.

Rigau, G. and E. Agirre. (1995). Disambiguating Bilingual Nominal Entries against WordNet. *Proceedings of the 7th ESSLLI Symposium*. Barcelona, Spain.

Schank, R.C. and R.P. Abelson. (1977). *Scripts, Plans, Goals, and Understanding*. Hillsdale: Lawrence Erlbaum Associates.

Sowa, J. (1999). *Knowledge Representation*.

Swartout, W.R., R. Patil, K. Knight, and T. Russ. 1996. Toward Distributed Use of Large-Scale Ontologies. *Proceedings of the 10th Knowledge Acquisition for Knowledge-Based Systems Workshop*. Banff, Canada.

Uschold, M., P. Clark, M. Healy, K. Williamson, and S. Woods. (1998). Ontology Reuse and Application. *Proceedings of the International Workshop on Formal Ontology in Information Systems*, 179–192. Held in association with the KR-98 Conference. Trento, Italy.

Visser, P., D. Jones, T. Bench-Capon, and M. Shave. (1998). Assessing Heterogeneity by Classifying Ontology Mismatches. *Proceedings of the International Workshop on Formal Ontology in Information Systems*, 148–162. Held in association with the KR-98 Conference. Trento, Italy.

Van Zyl, J. and D. Corbett. (2000). A Framework for Comparing the Use of a Linguistic Ontology in an Application. *Proceedings of the ECAI Workshop on Applications of Ontologies and Problem Solving Methods*. ECAI Conference. Berlin, Germany.

Wille, R. (1997). Conceptual Graphs and Formal Concept Analysis. *ICCS*, 290–303