*In F. Dau, M.-L. Mugnier, and G. Stumme (eds), Conceptual Structures: Common Semantics for Sharing Knowledge. Proceedings of the 13th Annual International Conference on Conceptual Structures (ICCS 2005), pp 91–106. Springer Lecture Notes in AI 3596. 2005.*

# Methodologies for the Reliable Construction of Ontological Knowledge

Eduard Hovy

Information Sciences Institute, University of Southern California
4676 Admiralty Way, Marina del Rey, CA 90292-6695, USA
hovy@isi.edu

**Abstract.** This paper addresses the methodology of ontology construction. It identifies five styles of approach to ontologizing (deriving from philosophy, cognitive science, linguistics, AI/computational linguistics, and domain reasoning) and argues that they do not provide the same results. It then provides a more detailed example of one of the approaches.

## 1 Introduction

After nearly a decade in which statistical techniques made "ontology" a bad word in various computational communities, there are encouraging signs that the pendulum is swinging back. But ontologies will be most readily accepted by their traditional critics only if at least two conditions are met: good methodologies for building and evaluating them are developed, and ontologies prove their utility in real applications. This paper addresses the question of the/a methodology for ontology building, a topic that has received relatively little attention.

Since ontologies have many aspects, Section 2 outlines the aspect of interest for this paper, namely ontology content. Section 3 describes five alternative content construction methodologies that have been adopted in the past. Section 4 provides a generic ontology construction procedure, and Section 5 puts the previous two sections together by illustrating the language-oriented methodology with a more detailed account of how ontology builders might proceed, and what they might produce. Finally, Section 6 expresses the hope that further work on developing this and other ontology construction methodologies can be of use.

## 2 Ontology Content: Shallow Semantics

The construction and use of ontologies for computational purposes has a long and varied history. In natural language processing (NLP), for example, ontologies were initially seen as the ultimate answer to many problems, but later rejected by almost everyone when it became clear that building adequate ones at the time was impossible. Recently, influential NLP figures are beginning to recognize that a certain type of semantics—very *shallow* semantics—is probably necessary to help statistical NLP systems overcome the quality performance ceilings many of them seem to have reached. Since statistical systems learn their operations from suitably prepared training material, the argument is generally that the nature and quality of these systems' performance is limited by the nature of the information in the material. Just as you cannot make gold from stone, you cannot obtain semantically adequate machine translation, text summarization, information retrieval, question answering, dialogue management, etc., without giving the system some access to semantics in its training.

A less extreme variant of this history was experienced in several other computational areas. Eventually, they all face the same core problem: semantics is important, but *which semantics*? Turning to the Knowledge Representation (KR) community, whose origin lies in the traditions of mathematics, logic, and philosophy, does not help. It has not yet been able to build large, widely used, general-purpose semantic theories or semantic resources required for practical use at the scale of NLP and similar applications; such semantic data and theories as do exist are almost always limited to small-scale (or toy) applications. KR work has been excellent in developing formalisms for representation, and for investigating the properties and requirements of various classes of deductive systems. But for practical applications such formalisms need *content*; the deductive systems need to work *on* something. A semantic theory of the kind needed to support NLP and other applications requires at least a collection of (unambiguous) semantic symbols, each carrying a clear denotation; a set of rules for composing these symbols, using a set of relations, into non-atomic representations of more-complex meanings; and some method of validating the results of composition, deduction, and other semantic operations.

Because of the complexity involved, content building of this sophistication has mostly occurred at a smaller scale. So despite the fact that the world is certainly complex enough that it is reasonable to expect more than 20,000 individual 'atoms' of meaning to be used as building blocks, very few term collections of this size are more than flat enumerations (for example, Standard Industrial Classification (SIC) codes, lists of geographical entities and locations, lists of chemicals, or pumps, or plants, all with their properties, etc.). Sets of symbols, taxonomized to enable inheritance of information and to support inference, are often called ontologies. But except for CYC [29], large-scale term sets (over 20,000 nodes) tend to contain little reasoning knowledge, providing mostly lexically anchored networks of words; the best-known examples are WordNet [10,35] and its immediate derivates such as EuroWordNet [46] and other languages' WordNets (http://www.globalwordnet.org/), though a few more-distant derivatives such as SENSUS [26] are also available. For composing atomic meaning symbols into more-complex structures, most theories provide relations, and typical sets of relations range from a dozen or so, such as Fillmore's early case roles [11], to maybe fifty by Sowa [43]. But no accepted standard set of relations exists either.

Probably the most troublesome aspect of conceptual 'content' semantics, however, is the near-complete absence of methodological discussion and emerging 'methodology theory' that would provide to the general enterprise of ontology building and relation creation the necessary rigor, systematicity, and eventually, methods for verification that would turn this work from an art to a science. (A notable exception is the work on DOLCE, which makes a good beginning; see www.loa-cnr.it/DOLCE.html and [12].) Without at least some ideas about how to validate semantic resources, both the semantics builder and the eventual semantics user are in trouble. The builder does not know how to ensure that what is built today is consistent, in a deep sense, with what was built yesterday (and indeed, problems of inconsistency have plagued all larger ontology-building efforts since their inception; for example, CYC at one point discarded more than half its content and started anew). The user does not know how to choose between various alternative semantic theories and resources, and is forced to rely on unverifiable claims, the builders' reputation and/or erudition, or subjective preferences.

What to do?

## 3 Five Methodologies for Ontology Construction

The rest of this paper focuses on the problem of creating and organizing into an ontology a set of terms, primitive or not, with which to define meaning in some domain. I use the word "ontology" quite informally here to denote any set of terms organized hierarchically according to the general property inheritance relation following subclass, but without additional requirements that logical entailment or other inferences be defined or that the terms obey such entailment. That is, "ontology" here includes terminology taxonomies such as WordNet. While it would be nice to adopt such requirements, the reality is that most people build ontologies to support their knowledge representation needs in some practical application, and that they are more concerned about the computational effectiveness and correctness of their application than about the formal completeness, correctness, or consistency of the ontology per se. (Should the system work well even if the ontology is somehow formally deficient, the practical projects I know of would be quite satisfied.) Naturally, completeness, consistency, etc., are ways to ensure that the ontology will not lead to unwelcome surprises in system behavior. But unfortunately, the strictures introduced by these requirements are usually so onerous that they make adopting the requirements tantamount to placing a severe limitation on the eventual scope of the ontology and hence of the whole practical enterprise. That is, especially for NLP applications, many people build their ontologies as relatively simple term taxonomies with some inheritance inference, but do not enforce stricter logical requirements.

When constructing an ontology (or domain model; the terms are used here interchangeably) one can either build everything de novo or one can start with the ontologies of others and combine, prune, and massage them together as needed. In recent work, several ontology building projects have created interfaces to assist with the manual merging of ontologies. Typically, these tools extend ontology building interfaces such as those of Ontolingua [97], Intraspect ([22] http://polaris-md.pepperdine.edu/Overview.html) and the Stanford CML Editor [23] by incorporating one or more variants of name matching and other heuristics plus validation routines that check for consistency of edited results [26,20,38,34].

With regard to the creation or extension of new domain models, some work in manual knowledge acquisition is developing interfaces that assist the knowledge entry worker by continually verifying what is entered, actively eliciting information to complete partial specifications, etc., using strategies modeled after human tutoring procedures [24,5].

However, what is still lacking in ontology construction is a systematic and theoretically motivated methodology that guides the builder and facilitates consistency and accuracy, at all levels. The reason for this lack is evident: today, we still do not have an adequate theory on which to base such a methodology. It is not even clear how one would begin to approach the problem of designing theoretically motivated procedures from a suitably general point of view. Consequently, although many people build ontologies today—see for example the OntoSelect website at http://views.dfki.de/Ontologies/ for over 750 ontologies in various domains—not one of the builders would be able to provide a set of operationalizable tests that could be applied to every concept and relation to inform in which cases his or her choices were wrong.

How then would one approach such a methodology? Well, we can consider what ontology builders actually do—the core operation(s)—and study how they justify their actions. As discussed in [1,45,21], three situations can arise when aligning terms from two ontologies: either the two terms are exactly equivalent, or one term is more general than the other, or the terms are incompatible[1]. As soon as inconsistencies are found, one has to make a choice[2]. It helps if one understands *why* the creators of the source ontologies did what they did. Based on lessons learned from practical experience in merging parts of several ontologies [20] and discussions with numerous individuals, the author has identified five types of motivation, which can be identified with five different research approaches: the *philosophers* [16,15,43]; the *cognitive scientists* [35,10,27,2]; the *linguists* [31,39]; the *Artificial Intelligence reasoners* [30,13], which includes the *computational linguists* [3,36]; and the *domain specialists* (too numerous to list). Each of these types of individuals operates in a distinct way, resolving questions with arguments that appeal to different authorities and patterns of reasoning, and (not unexpectedly) lead to very different re-

[1] This is not quite true; concepts can share parts of their meanings; see discussion later, and in [28] and [8]. However, following general practice, in this paper we limit the discussion to 'discrete' ontologies, in which concepts do not overlap.

[2] A reviewer of the paper points out that one does not *have* to make a choice; one can include alternatives, separating them in different namespaces, as long as one is not tied to a specific application. The point here is that when one is tied to an application, then one does have to make this choice, and since one builds ontologies for applications, the problem is prevalent.

sults. (It is important to point out right away that none of these are *correct*, compared to the others; the notion of correctness is itself a point of methodological discussion. But within each mode of thought, it is of course possible to be correct or wrong, to be more or less elegant, and to develop a more or less satisfying solution.)

In generating each new (candidate) ontology item, the ontologizer performs an act of creation. Stated as simply as possible, the ontologizer has to decide *whether* to create a term, and if so, *how to place it* with regard to the other existing terms (which constitutes some portion of the act of defining the term, of course). Then follows *additional specification and definition*. This decision process plays out as follows for the five 'personality types' of ontologizer:

Type 1: Abstract feature recombination (the philosophers). The procedure of concept creation by additive feature specification—systematically adding new differentiae—is the historical method of ontologization; interesting examples can be found all the way back to Aristotle. A modern version is provided in [43], who defines several highly abstract features (*Concrete–Abstract*; *Positive–Negative*; etc.) and then more or less mechanically forms combinations of them as separate concepts, using these features as differentiae. Sowa illustrates this procedure by generating the topmost few dozen concepts and arranging them in their combinatory lattice structure, under which he proposes a more traditionally-derived concept taxonomy be arranged. With the DOLCE ontology (www.loa-cnr.it/DOLCE.html, [12] employ so-called identity criteria to determine whether two concepts are the same or not, and how they may differ; these differences help establish the appropriate differentiae. The rigor adopted in this work, especially the use of identity criteria as a driving methodology, is a model for others. In general, the ontologizers adopting this approach are of course the philosophers; once they believe they have found the essential set of semantic primitives, the rest follows by logic. The approach is elegant, but unfortunately doesn't work beyond the very most abstract levels, and is hence not very useful for practical domain ontologies. Defining a list of the most abstract notions underlying our conceptualizations is a complex enough task; but it is truly scary to consider creating a list of all the differentiae one would have to specify (and arrange in some order, so as to avoid the full combinatory complexity) in order to define such notions as Love, Democracy, and (even) Table.

Type 2: Intuitive ontological distinctions (the cognitive scientists). The oldest and most natural reason for creating a new concept is simply the intuitive feeling that it is not the same as anything else already defined, which means that one has to split it off from its near-siblings and begin a new variation or subspecies. Unfortunately people are not consistent in doing so, and 'split off' new 'concepts' quite actively as the occasion demands, creating ad hoc subgroupings differentiated by whatever feature(s) are relevant to their purposes. This playful freedom is useful for communication and no doubt for thought in general; its results are sometimes recorded (by having words that name the concepts), and sometimes not. The result is a hodge-podge of wordsense families whose meanings partly overlap but differ on arbitrary dimensions, and for which no regular correspondences are found across languages in general (see for example [8] for a very nice paper on plesionyms[3] in various languages). Determining this kind of concept formation is the specialty of the cognitive scientist (especially the one interested in language), whose methodology (and proof) turns to devising clever experiments to measure how people make distinctions between close concepts. But the fluidity of the distinction process, being dependent on the person's interests, knowledge, task, and other circumstances, make this approach to ontology building fraught with inconsistency to the point of hopelessness.

Type 3: Cross-linguistic phenomena (the linguists). For some people, concepts can be motivated simply because words or expressions for them appear in many languages. When many cultures independently name a thought, is that not evidence for the existence of that thought as a separate concept? Whether one believes Vygotsky [47], Sapir [42] and Whorf [49], or Piaget [41] as to which of language and thought is primary (if either), the very close intertwining of them in the mind is generally accepted. As shown for example in EuroWordNet [46] or the plesionym study [8], analysis of cross-language differences uncovers complex and fascinating interrelationships among meanings and meaning facets. Naturally, this approach

[3] Near-synonyms, like *jungle*, *forest,* and *woods*.

fits the linguistic tradition, and for some NLP applications, especially cross-lingual ones, paying attention to many languages for ontology construction and lexicon development can be rewarding (see for example [7], about which more below). But since there are many concepts for which no words exist, and since it is easy to demonstrate shadings of meaning from one concept to another that suggest continua of unimagined dimensionality, no-one will accept the argument that "because it's so in language(s), it has to be exactly so in thought" as a final arbiter. Nonetheless, words, as the richest component in our arsenal of tools to attack meaning, remain central. Therefore language-inspired ontology work usually produces word networks such as WordNet that are strongest in the (typically large) middle region, corresponding approximately to language lexicons; at both the abstract (upper) and domain-specific and particular (lower) region, they tend to lose expressive utility.

<u>Type 4: Inference-based concept generalizations (the computational reasoners)</u>. For computational systems, one creates in effect a data model. The ontology (and domain model) is so arranged that those items in the domain that should be treated similarly are grouped together and typed, so that they can in fact be recognized and treated similarly. Such groupings tend to emphasize domain-specific concepts and produce more abstract concepts (i.e., upper ontologies) only insofar as they are required for grouping. For this methodology the validation method is simple and direct: Do I need separate treatment of some thing(s) by the system? If I create an appropriate new type, does the system work as required? The domain terms thus tend to mirror the metadata and the system variables. They must be defined in enough detail to support a reasonably powerful set of operators or rules, but not be so differentiated as to require too many rules. This methodology is relatively clean, depending on the elegance of the computational solution to the problem. Unfortunately, however, the decision justifications offered by systems builders today are seldom interesting to philosophers, psychologists, and linguists. Given how many ways there are to achieve a computational result, system builders are seldom able to express their analysis of the problem and its solution in terms of necessary information transformation operations, i.e., in compelling general information-theoretic terms that would convince the other disciplines' specialists of the logical necessity of organizing one's knowledge one way or another. This fact lies at the heart of the inevitable communication breakdown and decoupling that occurs when systems builders and any of the other disciplines' researchers set out jointly to build an ontology for some application, a breakdown that requires significant effort to overcome.

<u>Type 5: Inherited domain distinctions (the domain specialists).</u> In many ontology building enterprises, the reason for creating and arranging concepts stems neither from abstract theoretical analysis nor experimentation, but from existing domain theory and practice. Biologists, neuroscientists, aircraft builders, pump manufacturers, legal scholars, and anyone else in knowledge-intensive enterprises find it perfectly natural to construct ontologies that reflect the way their fields view their worlds; this class of ontologies is thus one of the most common in practice; see for example the over 700 ontologies listed on the OntoSelect website (http://views.dfki.de/ontologies/), covering space, travel, music, wine, sports, science, etc. [4]. Often the exercise of actually building an ontology is prompted by the desire to work in a computational setting, and frequently the organizational discipline imposed by ontology software causes an experience of some enjoyment and even some reorganization of the builder's own understanding. Connecting a domain ontology to a generic computational system (such as a sentence generator or parser) sometimes requires realignment and/or reconceptualization of the domain terms into the categories interpretable by the computational engine; a typical solution is to embed such domain model(s) under an Upper Model that supports the computation, as illustrated for sentence generation in [3].

<u>Addendum: Type 6: Taxonomic clarity</u>. There is another motivation for introducing concepts, one that almost all ontology builders employ. Sometimes it is simply useful for an ontology builder to insert some mid-level concepts in order to create organizational clarity, without explicitly formulating the criteria that justify their existence (esthetics and/or clarity of display are reasons not generally deemed sufficient to measure up to serious insights derived from psychological experiment, philosophical argument, computational necessity, or cross-linguistic comparison).

It is tempting to consider these approaches as complementary; one could for example ask the philosophers to build the uppermost, most abstract, regions, the cognitive scientists to provide some overall ontology framework that the computationalists and domain specialists can then flesh out and refine, etc. But there is no guarantee that the distinctions natural to ontology builders of one type will in fact correspond to or be useful for others' purposes. In practice, such admixture tends to require that all parties learn a little about every approach, and that one of them becomes the ultimate arbiter, usually on irrelevant grounds such as personality or loudness of argument.

## 4 Ontology Construction Procedure

Mismatches between ontologies are a source of never-ending discussion and wonder. But they are not surprising; when concept creation decisions can be justified on such different grounds as listed above, mismatches are to be expected and are not really very interesting on an individual basis. The ontologies simply differ in content and 'focus'. What is interesting is when one discipline delivers no insight and another must come to its aid. To prevent disaster, a methodology of ontology creation should recognize this fact and assign relative priorities to the various concept creation methods and justification criteria *a priori*, before any actual ontology building is done.

The above considerations apply for all ontology building efforts (although upper ontologies, given their abstraction from domain particulars, are a somewhat special case). To create a domain model, the methodology generally adopted (see for example [14]), which can be called *continual graduated refinement*, is:

1. Determine the general characteristics of the ontology to be built. A list of such characteristics (and additional ones) is provided in [19], and includes the domain of interest, the purpose of the ontology, the target level of granularity, the conceptual and theoretical antecedents, etc. Central to these decisions is selecting the principal criteria of ontologization (concept creation and justification methods) and specifying their order. In this step the task/enterprise is determinate: is this domain model to be used in a computational system? Or is it a conceptual product for domain analysis and description? Who are the intended users/readers of the ontology, and what is their purpose with it? What justification criteria will they find most understandable, and do these criteria match the purpose or task in mind?
2. Gather all additional knowledge resources, including starter ontologies, upper structures or microtheories (of, say, time and space), glossaries of domain terms, supporting descriptive and definitional material, algorithms and tools, existing theoretical descriptions, etc.
3. Delimit the major phenomena for consideration: identify the core concepts, types of features allowed, principal differentiae, etc. To lay out the general area, starting with an existing upper ontology, even one with just some dozen nodes, can be helpful.
4. List all readily apparent terms/concepts important for the task or enterprise. These terms may be derived from a (meta-)data model, from the algorithm of the system (to be) built, from experts' reports on the major components and processes in the domain, etc.
5. For each concept, explicitly record the principle(s) and factors that justify its creation. The definition may still be incomplete and informal, but should contain the principal differentiae and features of interest. Also identify interrelationships between the concept and related concepts (including subclass hierarchicalization, part-wholes, equivalence/synonymy, etc.), and specify/define them.
6. Inspect the nascent domain model for (ir)regularity, (im)balance, etc. Then for each major region (types of entity, types of action, types of state, etc.) repeat steps 3 to 5, refining existing concepts as needed. During this iterative refinement, record all problematic issues; they may require extensions to the upper ontology or even to the basic criteria of ontologization.
7. When done, characterize the ontology or domain model by recording its essential parameters, as spelled out in [19].

Working out the details of this methodology takes time and effort. Not all aspects apply in all cases, and not to all domains or ontologization styles [44]. Careful study of how domain ontologizers actually instantiate this procedure will help flesh out a systematic methodology of ontologizing.

# 5 Example Language-Based Methodology: Annotator-Driven Concept Granularity using Wordsenses

The core ontologization decisions outlined in the preceding section can be viewed as a question of concept granularity: given some semantic notion circumscribed by one or more near-synonymous words, how many concepts should one define for them, how should one organize the concepts, and how should one validate these choices? This section outlines, as example, a language-based methodology that starts by creating and validating wordsenses and then uses them to suggest concepts. It contains two parts: experiences with wordsense annotation and converting wordsenses into concepts.

## 5.1 Experiences in Wordsense Annotation

The OntoBank project (Weischedel et al., in prep.) is an ongoing attempt by researchers at BBN (Weischedel, Ramshaw, et al.), the University of Pennsylvania (Marcus, Palmer, et al.) and ISI (Hovy et al.) to construct a shallow semantic version of a collection of texts. Should continuation funding be achieved, the goal is to build by hand a set of one million sentences with their associated shallow semantic frames. In this project, shallow semantics includes disambiguated concept/wordsense symbols for each noun, verb, and adjective; basic verb frames with relations to constituents; resolved anaphoric links; canonicalized representations for straightforward dates, numerical expressions, proper named entities, and a few other phenomena. Central to OntoBank is the PropBank wordsense differentiation and annotation procedure [25].

The IAMTC project [7], which ended early for lack of continuation funding after one year, had as goal to uncover the representational machinery required to support an interlingua notation from an analysis of differences across seven languages. Project members annotated some 150 translated texts, each one in both English and its source language (one of Hindi, Arabic, Korean, Japanese, French, and Spanish). Similar to OntoBank, annotation included selection of a disambiguated concept/wordsense for each noun, verb, and adjective; the determination of an appropriate verb frame (in this case, LCS theta role frames [6]) and its connections to sentence constituents; and the design of a series of incrementally deepening representations en route toward the interlingua.

The author participated in both OntoBank and IAMTC, in more or less the same role, as ontology maintainer and developer. In both cases, ISI's Omega ontology at http://omega.isi.edu [40,18] was used as repository of all semantic symbols. In both projects, all nouns, verbs, and adjectives were annotated by multiple people, who selected the appropriate concept(s) to express the words' meaning(s) in context. Both projects paid considerable attention to the annotation interface, annotator training, post-annotation reconciliation discussions, and annotator agreement measures.

Of primary interest for this paper is the ontological considerations that arise when such annotation efforts are conducted. It is relatively straightforward, though not always easy, to build ontologies of specific well-circumscribed domains for computational purposes. But the picture changes somewhat when the focus is annotation of wide-coverage newspaper text in the interest of creating shallow semantic representations. In particular, the ontology maintainer is confronted with a stream of seemingly unrelated decisions about concept granularity and ontology placement, more or less one for every verb, noun, and adjective encountered. The OntoBank methodology is illustrative. Following the PropBank annotation procedure [25], the most frequent *N* words of a given type (say, verbs) are selected for annotation. For each verb, 100 sentences containing it are extracted from the corpus. Two or more annotators each see the same hundred sentences plus a list of candidate concept (sense) choices extracted from the ontology. Their task is to select just those concepts that express the meaning(s) of the verb in a given context.

It is apparent that the nature of the concept alternatives and the quality of their definition are of central importance. Omega, which for a large part is derived from the lexical network WordNet, usually contains too many close alternatives, confusing the annotators (annotator selection agreement when given WordNet senses as options is only around 70% for nouns). For example, the verb "drive" has 22 senses in WordNet, including separate senses for driving a car as chauffeur and driving a car as one's work. In an employment domain, the difference between chauffeur, taxi driver, and other kind of employed driver may be important, but in general texts this distinction is often not made, or it is so implicit that many people don't make it, leading to different annotator choices.

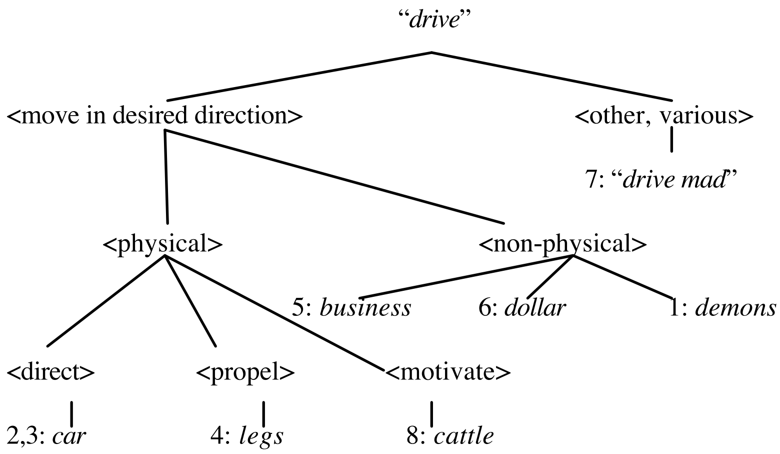
In contrast, the MIKROKOSMOS ontology [33,32], another source of concepts for Omega, almost always offers too little granularity; it has only one symbol for all vehicles, including cars, buses, airplanes, etc. Ideally, one would like something in between: just enough concepts to express the semantic differences that most people easily agree on in the context of the text.

## 5.2 From Words to Concepts: Hierarchical Graduated Refinement

Palmer and colleagues use sense creation/compression to build PropBank; they have developed a well-tested procedure. A slightly more elaborated and formalized procedure was developed at ISI to mirror appropriate wordsenses in the Omega ontology, which for this work was extended to accommodate separately wordsenses and concepts [18]. This sense creation procedure is interesting to perform, and perfectly illustrates the problem of the ontology builder, and the need for a strict methodology. For example, how many senses (concepts) should one define for the word "drive"? Different members of OntoBank produced quite different results, which differed from the decisions of PropBank's expert sense creator. The continual graduated refinement procedure outlined above is as follows: starting with the word and a number of example sentences representing all/most of its meanings, identify and split out the most semantically different sense (cluster) and create a branch point in the evolving tree; then for each branch, repeat the process downward. At each split write down the criteria, or at least a description of the difference; these give a hint at the thought process for later discussion and may eventually allow one to define more-formal differentiae. For example, consider the following sentences for "drive":

1. *Drive the demons out of her and teach her to stay away from my husband!!*
2. *Shortly before nine I drove my jalopy to the street facing the Lake and parked the car in shadows.*
3. *He drove carefully in the direction of the brief tour they had taken earlier.*
4. *Her scream split up the silence of the car, accompanied by the rattling of the freight, and then Cappy came off the floor, his legs driving him hard.*
5. *With an untrained local labor pool, many experts believe, that policy could drive businesses from the city.*
6. *Treasury Undersecretary David Mulford defended the Treasury's efforts this fall to drive down the value of the dollar.*
7. *Even today range riders will come upon mummified bodies of men who attempted nothing more difficult than a twenty-mile hike and slowly lost direction, were tortured by the heat, driven mad by the constant and unfulfilled promise of the landscape, and who finally died.*
8. *Cows were kept in backyard barns, and boys were hired to drive them to and from the pasture on the edge of town.*

How many concepts/senses should one create? In WordNet, "drive" has 22 senses. Employing our procedure of hierarchical graduated refinement on these (and additional) sentences, the author and a student separately found 7 major senses, in the order of the wordsense hierarchy below (hints for differentiate are indicated in angle brackets, as well as sentence numbers with focal words):




Given more sentences, additional subsenses can be found, including driving a tool ("*drive the hammer*") under <propel>, driving non-cars ("*drive a bulldozer*") under <direct>, employment ("*drive a taxi for work*") or phrasal expressions ("*drive a hard bargain*") under <other>, etc.

Adopting the above hierarchical graduated refinement procedure[4] is useful for several reasons. It supports human analysis and agreement by systematically removing the most glaringly different cases. It helps suggest differentiae. It allows one to vary the granularity at will by simply ending the differentiation process sooner or later, and by grouping together as undifferentiated all senses lower than the chosen level. This capability is extremely useful when one separates word senses from ontology concepts. As argued in for example [37] and discussed in [17], not all wordsenses in the sense hierarchy need (or should) be converted into ontological categories (concepts). The sense hierarchicalization for "drive" above, for example, requires at least three distinct concepts, namely "drive mad" (i.e., something like Cause-Mental-Instability), the nonphysical sense group (rooted in something like Cause-State-Change-toward-Desired-Value), and the physical group (rooted in Cause-Movement-in-Desired-Direction) respectively. Being so different, the first of these will be inserted into the ontology at a point quite remote from the other two. Further, sentence 1, "drive the demons out of her", may be treated at two levels: the surface-metaphorical (in which case "demons" metaphorically stands for "illness" and the driving is non-physical), or the 'true' semantic (in which case the meaning is something like Heal-of-Mental-Disorder, and there is no driving, even metaphorically). In the latter case, sentence 1 would also be classified in a region of the ontology quite remote from the other concepts.

One can continue ontologizing the sense hierarchy to the extent one wishes to (or can) formalize the differentiae and one believes further distinction provides valuable additional explanatory or computational utility. Each step of ontologization requires placing the newly formed concept into the growing ontology where appropriate. This procedure makes apparent that there is no direct homomorphism of the sense hierarchy into the ontology, though as senses are 'closer' lower down in the hierarchy, one would expect their ontological equivalents also to be closer in the ontology.

It is a good reason to stop ontologizing the sense hierarchy when no obviously most-different sense or sense group can be identified, that is, when it is possible to split the remaining group in several equally good ways according to different criteria. At this point, one has most probably reached a level of semantic homogeneity at which several features combine in equal measure to form the concept's unique identity.

Cross-language studies are useful in helping to identify a division between 'sense space' and 'ontology space'. In the IAMTC project, when the languages in question (Hindi, Arabic, Korean, French, Spanish,

[4] Sense differentiation in PropBank is not performed hierarchically as shown here; the expert produces a flat list. Nonetheless, trained staff can easily estimate the level of granularity that will ensure an inter-annotator agreement of over 90% (the target value in OntoBank; this figure, on average, required a 50% reduction of WordNet senses, for example).

and Japanese) provided different words (and hence usually also senses) after translation (by various professional-quality translators) into English, then the granularity of the concept in question had to be such as to represent the senses common across the various translations, while their individual, language-idiosyncratic, facets of difference remained in the sense hierarchy. One can thus think of 'ontology space' as the *interlingual* representation symbols (symbols capturing common, or common enough, meaning aspects); of 'sense space' as the *multi-lingual* representation symbols (symbols for senses that may or may not co-occur across languages, but that are mapped to meanings no more specific than they denote themselves), and of 'lexical space' as the *monolingual* representation symbols (namely, the words of each language). There is a complex many-to-many mapping across both gaps.

## 6 Conclusion

The two-stage methodology—creating wordsenses and annotating them to determine granularity, followed by conversion of (part of) the sense hierarchy into ontology concepts has several desirable properties. It is 'empirical', in the sense that the semantic distinctness of a set of wordsense (concept) candidates can be validated through annotator agreement. This gives an upper bound on the granularity of concepts, and interestingly blends linguistic and cognitive motivations in a computational setting. The hierarchical graduated refinement procedure is easily extensible to finer detail, and provides suggested differentiae for formalization. The overall methodology accommodates comparison across and incorporation of different languages by separating language-dependent wordsenses from language-independent concepts.

It is not claimed that the methodology described here is a proven, or even adequately explored, methodology for ontology creation. But given the increasingly pressing need for more attention to methodology in current-day ontology creation, it is to be hoped that these thoughts will inspire further exploration. In addition, hopefully the new interest in ontologies will usher in more work on methodology as well. It is a fascinating and rewarding direction of research.

## Acknowledgements

I am indebted to Graeme Hirst and Patrick Pantel for their comments, both in print and in person.

## References

1. 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.
2. Alberdi, E., D.H. Sleeman, and M. Korpi. 2000. Accommodating Surprise in Taxonomic tasks: The Role of Expertise. *Cognitive Science* 24(1), 53–92.
3. 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.
4. Buitelaar, P., T. Eigner, and T. Declerck. 2004. OntoSelect: A Dynamic Ontology Library with Support for Ontology Selection. *Proceedings of the Demo Session at the International Semantic Web Conference.* Hiroshima, Japan.
5. Chklovski, T., V. Ratnakar, and Y. Gil. 2005. User Interfaces with Semi-Formal Representations: A Study in Designing Augmentation Systems. *Proceedings of the Conference on Intelligent User Interfaces (IUI05).* San Diego, CA.

6. Dorr, B.J., M. Olsen, N. Habash, and S. Thomas. 2001. The LCS Verb Database. Technical Report Online Software Database, University of Maryland, College Park. http://www.umiacs.umd.edu/bonnie/LCS_Database_Docmentation.html.
7. Dorr, B., D. Farwell, R. Green, N. Habash, S. Helmreich, E.H. Hovy, L. Levin, K. Miller, T. Mitamura, O. Rambow, F. Reeder, A. Siddharthan. (submitted). Interlingual Annotation of Parallel Corpora. Submitted to *Journal of Natural Language Engineering*.
8. Edmonds, P. and G. Hirst. 2002. Near-synonymy and Lexical Choice. *Computational Linguistics* 28(2), 105–144.
9. 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.
10. Fellbaum, C. 1998. (ed.) *WordNet: An On-Line Lexical Database and Some of its Applications*. Cambridge: MIT Press.
11. Fillmore, C. 1976. The Case for Case. In E. Bach and R. Harms (eds) *Universals in Linguistic Theory*, 1–88. Holt, Rinehart, and Winston.
12. Gangemi, A., N. Guarino, C. Masolo, A. Oltramari, and L. Schneider. 2002. Sweetening Ontologies with DOLCE. In A. Gomez-Perez and V.R. Benjamins (eds), *Knowledge Engineering and Knowledge Management*. Proceedings of Ontologies and the Semantic Web at the 13th EKAW Conference. Siguenza, Spain. Springer Verlag, 166–181.
13. Gil, Y., R. MacGregor, K. Myers, S. Smith, and W.R. Swartout. 1999. CommonP: A Common Plan Representation for Air Campaign Plans. USC/ISI Technical Report available at http://www.isi.edu/isd/HPKB/planet/alignment.
14. Gruber, T.R. 1993. Toward principles for the design of ontologies used for knowledge sharing. In N. Guarino and R. Poli (eds.), *International Workshop on Formal Ontology,* Padova, Italy. Revised August 1993. Published in *International Journal of Human-Computer Studies*, special issue on Formal Ontology in Conceptual Analysis and Knowledge Representation (guest editors: N. Guarino and R. Poli) (to appear). Available as technical report KSL-93-04, Knowledge Systems Laboratory, Stanford University.
15. 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.
16. Guarino, N. and C. Welty. 2004. An Overview of OntoClean. In S. Staab and R. Studer (eds), *Handbook on Ontologies*. Springer Verlag, 151–159.
17. Hirst, G. 2004. Ontology and the Lexicon. In S. Staab and R. Studer (eds), *Handbook on Ontologies*. Springer Verlag, 209–229.
18. Hovy, E.H., A. Philpot, P. Pantel, M.B. Fleischman. (in prep.) The Omega Ontology.
19. Hovy, E.H. 2002. Comparing Sets of Semantic Relations in Ontologies. In R. Green, C.A. Bean, and S.H. Myaeng (eds), *The Semantics of Relationships: An Interdisciplinary Perspective*, 91–110. Dordrecht: Kluwer.
20. 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.
21. Hovy, E.H. and S. Nirenburg. 1992. Approximating an Interlingua in a Principled Way. *Proceedings of the DARPA Speech and Natural Language Workshop*. Arden House, NY.
22. Intraspect Corp. 1999. *http://www.intraspect.com/product_info_solution.htm*.
23. Iwasaki, Y., A. Fraquhar, R. Fikes, and J. Rice. 1997. *A Web-Based Compositional Modeling System for Sharing of Physical Knowledge*. Nagoya: Morgan Kaufmann.
24. Kim, J., Y. Gil, and M. Spraragen. 2004. A Knowlesdge-Based Approach to Interactive Workflow Composition. *Proceedings of the Workshop on Planning and Scheduling for Web and Grid Services* at the International Conference on Automatic Planning and Scheduling (ICAPS 04). Whistler, Canada.
25. Kingsbury P. and M. Palmer. 2002. From Treebank to PropBank. *Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC-2002)*. Las Palmas, Spain.
26. Knight, K. and S.K. Luk. 1994. Building a Large-Scale Knowledge Base for Machine Translation. *Proceedings of the AAAI Conference*, 773–778.
27. Korpi, M. 1988. *Making Conceptual Connections: An Investigation of Cognitive Strategies and Heuristics for Inductive Categorization with Natural Concepts*. Ph.D. dissertation, Stanford University.
28. Lakoff, G. 1987. *Women, Fire, and Dangerous Things: What Categories Reveal about the Mind*. Chicago: University of Chicago Press.

29. Lenat, D.B. and R.V. Guha. 1990. Building Large Knowledge-Based Systems. Reading: Addison-Wesley.
30. Lenat, D.B. 1995. CYC: A Large-Scale Investment in Knowledge Infrastructure. *Communications of the ACM* 38(11), 32–38.
31. Levin, B. 1993. *English Verb Classes and Alternations*. Chicago: University of Chicago Press.
32. Mahesh, K. and S. Nirenburg. 1995. A Situated Ontology for Practical NLP. *Proceedings of the Workshop on Basic Ontological Issues in Knowledge Sharing, International Joint Conference on Artificial Intelligence (IJCAI-95)*. Montreal, Canada.
33. Mahesh, K. 1996. Ontology Development for Machine Translation: Ideology and Methodology (CRL report MCCS-96-292). Las Cruces: New Mexico State University.
34. 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*).
35. Miller, G.A. 1990. WordNet: An Online Lexical database. International Journal of Lexicography 3(4) (special issue).
36. Nirenburg, S., V. Raskin and B. Onyshkevych. 1995. Apologiae Ontologia. *Proceedings of the International Conference on Theoretical and Methodological Issues* (TMI). Leuven, Belgium.
37. Nirenburg, S. and Y. Wilks. 2001. What's in a Symbol: Ontology, Representation and Language. *Journal of Experimental and Theoretical Artificial Intelligence* 13(1), 923.
38. Noy, N.F. and M.A. Musen. 1999. 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>.
39. Palmer, M., J. Rosenzweig, and W. Schuler. 1998. Capturing Motion Verb Generalizations with Synchronous TAGs. In P. St. Dizier (ed.) *Predicative Forms in NLP*. Boston, MA: Kluwer Academic Press.
40. Philpot, A., M. Fleischman, E.H. Hovy. 2003. Semi-Automatic Construction of a General Purpose Ontology. *Proceedings of the International Lisp Conference*. New York, NY. October 2003. Invited.
41. Piaget, J. 1959 (English translation). *Language and Thought of the Child*. London: Routledge Classics.
42. Sapir, E. 1929/1964. The Status of Linguistics as a Science. *Language* 5, 207–214. Reprinted in D.G. Mandelbaum (ed) *Culture, Language, and Personality: Selected Essays of Edward Sapir*. Berkeley, CA: University of California Press.
43. Sowa, J. 1999. *Knowledge Representation*.
44. 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.
45. 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.
46. Vossen, P. (ed) 1998. *EuroWordNet: A Multilingual Database with Lexical Semantic Networks*. Dordrecht: Kluwer Academic Publishers.
47. Vygotsky, L.S. 1978 (English edition). *Mind in Society: The Development of Higher Psychological Processes*. Cambridge, MA: Harvard University Press.
48. Weischedel, R., E.H. Hovy, M. Marcus, M. Palmer, L. Ramshaw. (in prep.) The OntoBank Project.
49. Whorf, B.L. 1940/1972. Science and Linguistics. *Technology Review* 42(6), 227–231, 247–248. Reprinted in J.B. Carroll (ed) *Language, Thought, and Reality: Selected Writings of Benjamin Lee Whorf*. Cambridge, MA: MIT Press.