New Developments in Ontological Semantics

Antonio Moreno Ortiz*, Victor Raskin†, Sergei Nirenburg††

*University of Málaga & Onyx Inc.
F. Filosofía y Letras, 29071 Málaga, Spain
amo@uma.es
† Purdue University & Onyx Inc.
West Lafayette, IN 47907 USA
vraskin@purdue.edu
†† Computing Research Laboratory, NMSU & Onyx Inc.
Las Cruces, NM 88003 USA
sergei@crl.nmsu.edu

Abstract

In this paper we discuss ongoing activity within the approach to natural language processing known as *ontological semantics*, as defined in Nirenburg and Raskin (forthcoming). After a brief discussion of the principal tenets on which this approach is built, and a revision of extant implementations that have led toward its present form, we concentrate on some specific aspects that are key to the development of this approach, such as the acquisition of the semantics of lexical items and, intimately connected with this, the ontology, the central resource in this approach. Although we review the fundamentals of the approach, the focus is on practical aspects of implementation, such as the automation of static knowledge acquisition and the acquisition of scripts to enrich the ontology further.

1. Fundamentals of ontological semantics

Ontological semantics is defined as "an integrated set of complex theories, methodologies, descriptions and implementations" (Nirenburg and Raskin, forthcoming), where a theory is the set of statements that determine the format of descriptions -obtained by applying certain methodologies- of the phenomena that the theory deals with. From this definition it follows that ontological semantics places strong emphasis on content rather than formalism. This characteristic has a strong impact on all four aspects of the approach, which is highly eclectic with regard to both representation and processing of content, linguistic or otherwise. It also implies, however, that a substantial effort in terms of acquisition of the resources is required prior to the successful implementation of the various processors that the approach requires.

Historically, a number of research projects have contributed to bring ontological semantics into its current state. These projects have been aimed at producing robust large-scale natural language processing systems to be used in machine translation, information retrieval and extraction, text summarization, and other such tasks. Perhaps the most relevant ones are Dionysus (Monarch et al., 1989), Pangloss (Nirenburg, 1994), Mikrokosmos (Nirenburg et al., 1995), CAMBIO (Nirenburg, 2000a), and CREST (Nirenburg, 2000b).

1.1. Theory and methodologies

We define *theory* as a set of statements that determine the format of descriptions of phenomena in the purview of the theory. A theory is effective if it comes with an explicit methodology for acquiring these descriptions. A theory associated with an application is interested in descriptions that support the work of an application. Figure 1 specifies how that schema applies to ontological semantics (the general notions are listed as headers in the four boxes; their interpretation for ontological semantics is given in the rest of the text in the boxes).

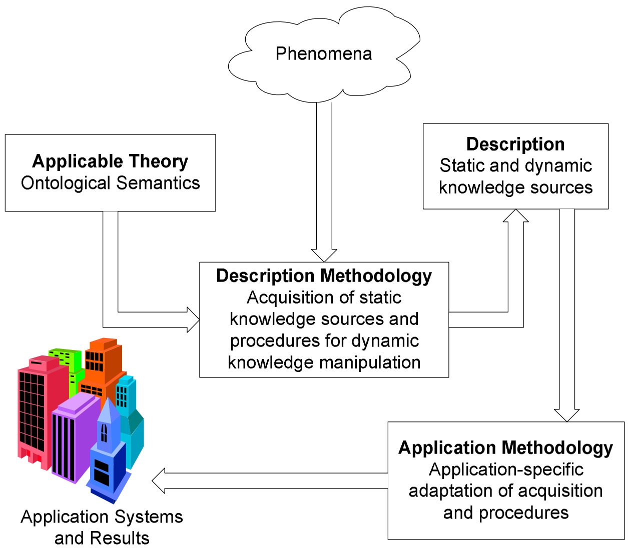

Figure 1: Theory, methodologies, and applications

The theory of ontological semantics includes the format and the semantics of dynamic knowledge sources, i.e., text meaning representations[1] as well as static knowledge sources: the ontology, the fact database, the lexicons and the onomasticons as well as the generic processing architecture for analysis of meaning and its manipulation, including the generation of text off of it. The description part in ontological semantics includes all

[1] The Text Meaning Representation (TMR) is the format used for representing text meaning in ontological semantics. The TMR is constructed compositionally from the meaning of the individual elements, defined either ontologically or procedurally, that constitute texts: words, bound morphemes, syntactic structures, and word, phrase, and clause order in the input text. A proper description of their construction process and format falls beyond the scope of this paper. See Nirenburg and Raskin (forthcoming) for an in-depth discussion.

the knowledge sources, both static and dynamic (generic procedures for extraction, representation and manipulation of meaning), implemented to provide full coverage for a language (or languages) and the world. In practice, an ontological semantic description is always partial, covering only a subset of subject domains and sublanguages, and constantly under development, through the process of acquisition and as a side effect of the operation of any applications based on ontological semantics.

The methodology of ontological semantics consists of acquisition of the static knowledge sources, discussed in section 3, and of the procedures for producing and manipulating dynamic knowledge structures.

1.2. Components

From the previous description, it follows that a typical architecture of an implementation of ontological semantics comprises the following components:

- A set of *static knowledge sources*: an *ontology*, understood as a model of the world, which underlies any systematic content representation, a *fact database*, i.e., a collection of facts (and opinions), consisting of instances of complex events and objects from the ontology, a (monolingual) *lexicon*, connecting the ontology with a natural language, and an *onomasticon*, a lexicon of names.
- *Knowledge representation languages* for the formal specification of these resources as well as dynamic, i.e., 'runtime', knowledge structures (TMRs).
- A set of *processors*, such as syntactic and semantic analyzers, text generators, etc.

The various methodologies developed for acquisition of the static knowledge sources (the ontology, monolingual lexicons, onomasticons, and fact database) of ontological semantics necessarily involve, at this stage of development, considerable human participation, although the aim is to fully automate all processes involved in the approach, both in terms of acquisition and runtime procedures. It is these methodologies that we will be primarily discussing here, although it is not possible to isolate this aspect from the others, particularly since ontological semantics is the result of constant interaction between them, the main reason for the abovementioned heterogeneity of theories and methodologies.

Given the strong dependence on the availability of knowledge sources, the development of the theory involves the bootstrapping and further massive acquisition of such sources.

2. Static knowledge sources

The set of static knowledge sources in ontological semantics includes the ontology, the fact database and, for each of the languages used in a given application, a lexicon and an onomasticon (a lexicon of names). The interaction between these sources is illustrated in Figure 2.

The ontology obviously plays a fundamental role in an ontological semantics system, as it is the central resource to which all others relate: most (open class) lexical entries obtain their semantics from it by being assigned to one or more ontological concepts, and entries in the Fact DB are instances of such concepts.

All static knowledge sources can currently be managed through a web-based user interface called Knowledge Base Acquisition Editor (KBAE)[2], whose basic functionalities we discuss in this section and illustrate by means of several screen captures showing actual entries. KBAE was designed with collaborative work in mind, allowing users, both local and remote, to search, browse and edit the data contained in the different resources. This does not imply, however, that all editing operations must be performed using this tool. Trained acquirers can opt to use, for example, plain text files for faster acquisition, which will then be imported by the database administrator into the relevant resource, and would subsequently be accessible through the interface. In section 3 we discuss some of the methods we are currently employing to speed up the acquisition process.

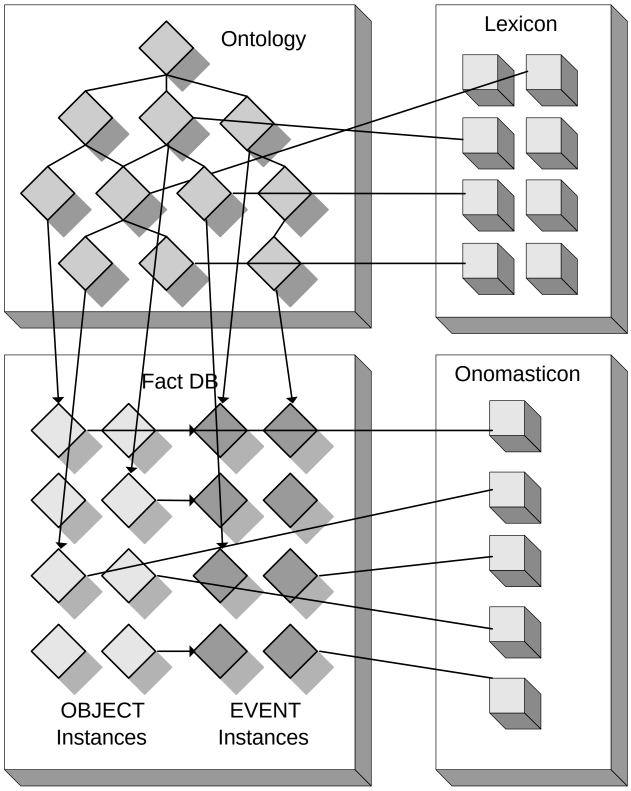

Figure 2: Static knowledge sources schema

2.1. Ontology

The ontology provides a metalanguage for representing and processing the meaning of lexical units of a language as well as for the specification of meaning encoded in runtime knowledge structures. Concepts in the ontology are abstract constructs that must be understood as corresponding to classes of things and events in the world. This world model is constituted by objects and events, which are themselves defined by some properties (relations or attributes). In Figure 3, KBAE is being used to browse the top–level tree.

A detailed, formal description of the format and characteristics of the ontology in its present formulation can be found in Nirenburg and Raskin (forthcoming); here, we briefly enumerate here its outstanding features:

[2] In the near future, we plan to make the resources publicly accessible through a guest, non-editing account using KBAE.

- Concepts are organized in a tangled hierarchy of concept frames
- All concepts are symbols that must be defined in the ontology.
- The top sub-tree consists of the following nodes [ALL[OBJECT, EVENT, PROPERTY [RELATION, ATTRIBUTE]]].
- OBJECTS and EVENTS refer to classes of entities in the world and are defined by their PROPERTIES, which are the real primitives in the ontology.
- PROPERTIES are considered second-order concepts, as their function is to define first-order concepts (objects and events). Properties are defined by their DOMAIN, i.e., the set (sub-tree) of (first-order) concepts they can be used to define and their RANGE of fillers (other concepts in the case of RELATIONS, scalar or literal values in the case of ATTRIBUTES).
- Properties assigned to first-order concepts (*slots* in terms of representation) are inherited down the hierarchy by default unless otherwise specified.

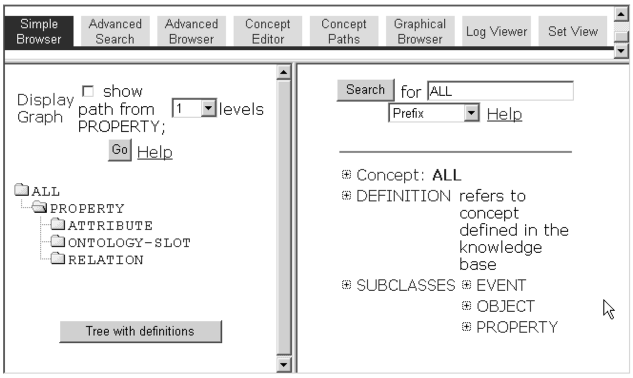

Figure 3: Top-level tree in the ontology

Ontological Semantics is in fact guided by the theory of inheritance (Touretzky, 1984, 1986; Thomason and Touretzky, 1991). The inheritance hierarchy is implemented using IS-A and SUBCLASSES slots. When two concepts, X and Y are linked via an IS-A relation (X IS-A Y), then X inherits all slots (with their corresponding facets and fillers from Y unless this is blocked by means of a special filler NOTHING. While multiple inheritance is fully allowed, no extant implementation of ontological semantics has fully developed sufficiently formal methods for using it.

2.2. Fact database

Language-independent knowledge in ontological semantics is not reduced to the abstract specification of classes of entities (an ontology as sketched above), it also comprises a set of instances of such constructs. In past implementations of ontological semantics (e.g., Mikrokosmos), such instances were integrated in the ontology by means of INSTANCES and INSTANCE-OF slots. However, there are important reasons to keep semantic (i.e., ontological) apart from episodic memory (Tulving, 1985), or contingent from non-contingent knowledge (Bar-Hillel, 1954).

Instances in the Fact DB are indexed by the concept they correspond to, and can be interrelated on temporal, causal and other properties. Figure 4 shows an instance of the concept NATION. A numerical index is used to uniquely identify each instance. Properties are also used to describe instances: BORDERS-ON, HAS-CURRENCY, etc., are in fact slots (attributes) defined in the ontology. Unlike concepts in the ontology, the fillers of slots assigned to instances cannot be concepts; instead, they are either instances of concepts (e.g. “Portugal NATION-160”) or literal or scalar values.

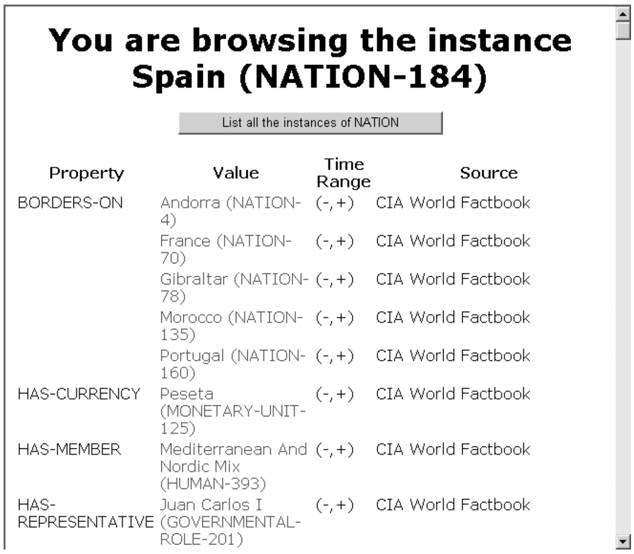

Figure 4: A Fact DB entry

It seems also very practical to keep two separate repositories for these two kinds of knowledge, since the number of instances, unlike the number of abstract concepts, is likely to keep growing exponentially as more information is added to the database.

2.3. Lexicon

Unlike the ontology and Fact DB, the lexicon is a language-dependent knowledge source, which means that each individual ontological semantics lexicon describes one language only. The monolingual lexicons do not contain any sort of connection among their lexical entries, which means that, for tasks involving multilinguality (e.g., translation), the processors are in charge of performing whatever contrasting operations are needed, employing the ontologically-driven semantic descriptions found in the individual lexical entries, as well as the ontology itself.

An ontological semantics lexicon contains, at least, the following sections:

1. *General*: word class, definition, example, comments, variants.
2. *Syntax*: f-structure, phrase structure.
3. *Semantics*: direct or constrained mapping, aspect, modality, style, time.
4. *Linking*: case roles.

Although all these aspects deserve illustration and further discussion, we concentrate here on the semantics section

Figure 5 shows a complete lexical entry in the Spanish lexicon. The entry has two senses, the second one (“compensación-N2”) is semantically described by means of direct mapping: the semantics of the noun fully coincides with an existing concept in the ontology (COMPENSATE), to which the lexical entry is mapped and then modified only in terms of style. The first sense illustrates the use of constrained mapping: since there is no concept that exactly matches the semantics of this sense, we take the closest in meaning and then modify some of its properties to construct a complex knowledge structure that quite accurately reflects the meaning of this noun.

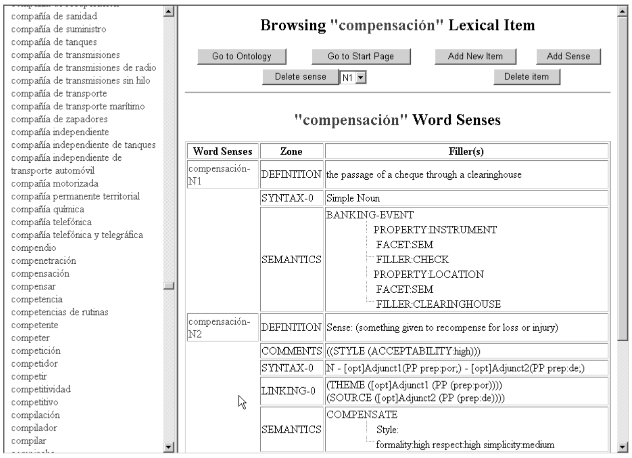

Figure 5: A lexicon entry in KBAE

Constrained mappings are a powerful device for avoiding the proliferation of concepts in the ontology, the drawback being not only increased processing load, but also considerable acquisition work, since we have not yet found a way to automate it, unlike direct mapping, which can partially be automated. In the next section, we discuss some of these automation techniques.

3. Automating knowledge acquisition

Knowledge-based applications involving natural language processing have traditionally carried the stigma of being too expensive to develop, difficult to scale up and to reuse as well as incapable of processing a broad range of inputs[3]. The opinion about the high price of development was due to the perceived necessity to acquire all knowledge manually, using highly-trained and, therefore, expensive human acquirers. The difficulty in scaling up was believed to reflect the deficiencies in description breadth, or coverage of material, in the acquisition task for any realistic application. The all-too-real failure of knowledge-based processors on a broad range of inputs was attributed to the lack of depth (or, using our terminology, coarseness of the grain size) in the specification of world and language knowledge used by the meaning manipulation procedures.

In the consecutive implementations of ontological semantics, the above problems have been progressively addressed. While we cannot claim to have completely eliminated the need for controlling the acquisition process by people, we are satisfied that ontological semantics uses about as much automation in the acquisition process as is practical within the state of the art in statistical methods of text processing and human-computer interaction. In addition to that, the acquisition methodology takes advantage of all and any possibilities for minimizing human acquisition effort and maximizing the automatic propagation of semantic information recorded earlier over newly acquired material, as applicable. The use of inheritance in the ontology; of information extraction engines in acquiring facts for the Fact DB; as well as lexical rules and class-oriented syntactic dependency templates in the lexicon, are among the examples of such facilities. We have had numerous opportunities to port the resources of ontological semantics across applications, and found this task feasible and cost-effective, even within small projects. In the rest of this section, we briefly review the methodology of knowledge acquisition that has emerged over the years in ontological semantics.

3.1. Bootstrapping

Before a massive knowledge acquisition effort by teams of acquirers can start, there must be a preparatory step that includes, centrally, the specification of the formats and of the semantics of the knowledge sources, that is, the development of a theory. Once the theory is initially formulated (it is fully expected that the theory will be undergoing further development between implementations), the development of a toolkit for acquisition can start. The toolkit includes acquisition interfaces, statistical corpus processing tools, a set of text corpora, a set of machine-readable dictionaries (MRDs), a suite of pedagogical tools (knowledge source descriptions, an acquisition tutorial, a help facility) and a database management system to maintain the data acquired (see Figure 6).

In many ontology-related projects, the work on the knowledge specification format, on portability and on the acquisition interfaces becomes the focus of an entire enterprise (see, for instance, Ginsberg, 1991; Genesereth and Fikes, 1992; Gruber, 1993; Farquhar et al., 1997 for a view from one particular research tradition). In such format-oriented efforts, it is not unusual to see descriptive coverage sufficient only for bootstrapping purposes. Ontological semantics fully recognizes the importance of fixed and rigorous formalisms as well as good human computer interaction practices. However, in the scheme of priorities, the content always remains the prime directive of an ontological semantic enterprise.

The preparatory step is in practice interleaved with the bootstrapping step of knowledge acquisition. Both steps test the expressive power of the formats and tools and seed the ontology and the lexicon in preparation for the massive acquisition step.

The bootstrapping of the ontology consists of

[3] Today's state-of-the-art rule-based methods for natural language understanding provide good performance in limited applications for specific languages. However, the manual development of an understanding component using specific rules is costly as each application and language requires its own adaptation or, in the worst case, a completely new implementation. In order to address this cost issue, statistical modeling techniques are used in this work to replace the commonly-used hand-generated rules to convert the speech recognizer output into a semantic representation. The statistical models are derived from the automatic analyses of large corpora of utterances with their corresponding semantic representations. To port the semantic analyzer to different applications it is thus sufficient to train the component on the application- and language-specific data sets as compared to translating and adapting the rule-based grammar by hand (Minker et al., 2000, xiv).

- developing the specifications of the concepts at top levels of the ontological hierarchy, that is, the most general concepts;
- acquiring a rather detailed set of properties, the primitives in the representation system (for example, case roles, properties of physical objects, of events, etc.), because these will be used in the specifications of all the other ontological concepts by inheritance;
- acquiring representative examples of ontological concepts that provide models (templates) for specification of additional concepts; and
- acquiring examples of ontological concepts that demonstrate how to use all the expressive means in ontology specification, including the use of different facets, of sets, the ways of specifying complex events, etc., also to be used as a model by acquirers, though not at the level of an entire concept.

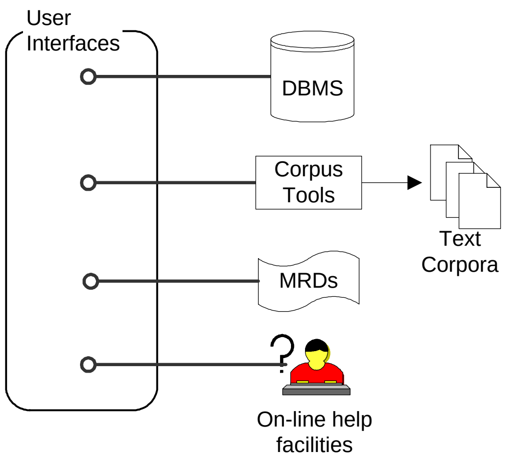

Figure 6: An ontological semantics acquisition toolkit

The bootstrapping of the lexicon for the recent implementations of ontological semantics involved creating entries exemplifying

- all the known types of syntax-to-semantics mapping (linking);
- using every legal kind of ontological filler—from a concept to a literal to a numerical or abstract range;
- using multiple ontological concepts and non-propositional material, such as modalities or aspectual values, in the specification of a lexical entry;
- using such expressive means as sets, *refsems* (internal bound variables) and other special representation devices.

The main purpose of this work is to allow the acquirer during the massive acquisition step to use the example entries as templates instead of deciding on the representation scheme for a meaning from first principles. As usual, practical acquisition leads to the necessity of revising and extending the set of such templates. This means that bootstrapping must be incremental, that is, one cannot expect for it to finish before the massive acquisition step. The preparatory step and bootstrapping are the responsibility of ontological semanticists who are also responsible for training acquirer teams and validating the results of massive knowledge acquisition. The complete set of types of work that ontological semanticists must do to facilitate a move from pure theory to an actual description includes:

- theory specification,
- acquisition tool design,
- resource collection,
- management of acquisition teams
- training,
- work process organization,
- quality control.

3.2. Massive acquisition

At the step of massive knowledge acquisition, the acquirers use the results of the bootstrapping stage to add ontological concepts and lexicon entries to the knowledge base. It is important to understand that, in the acquisition environment of ontological semantics, acquirers do not manually record all the information that ends up in a static knowledge source unit—an ontological concept, a lexical entry or a fact. Following strict regulations, they attempt to minimally modify existing concepts and entries to produce new ones. Very typically, in the acquisition of an ontological concept, only a small subset of properties and property values are changed in a new definition compared to the definition of an ancestor or a sibling of a concept that is used as a starting template. Similarly, when acquiring a lexical entry, the most difficult part of the work is determining what concept(s) to use as the basis for the specification of the meaning of a lexical unit; the moment such a decision is made, the nature of the work becomes essentially the same as in ontological acquisition—determining which of the property values of the ontological concept to modify to fit the meaning. With respect to facts, the prescribed procedure is to use an information extraction system to fill ontologically inspired templates that become candidate entries in the fact database, so that the task of the acquirer is essentially just to check the consistency and validity of the resulting facts. The interface also helps reduce the amount of work by using default values, which can later be edited. All in all, only a fraction of the information in the knowledge unit that is acquired at the massive acquisition step is recorded manually by the acquirer, thus imparting a rather high level of automation to the overall acquisition process.

The lists of candidate ontological concepts and lexicon entries to be acquired are included in the toolkit and are manipulated in prescribed ways. Acquirers take items off these lists for acquisition but as a result of at least some acquisition efforts, new candidates are also added to these lists. For example, when a leaf is added to an ontological hierarchy, it often becomes clear that a number of its conceptual siblings are worth acquiring. When a word of a particular class is given a lexicon entry, it is enticing to immediately add the definitions of all the other members of this class. The above mechanism of augmenting candidate lists can be called deductive, paradigmatic or domain-driven. The alternative mechanism would be inductive, syntagmatic and corpus(data)-driven and will involve adding words and phrases newly attested in a corpus to the list of lexicon acquisition candidates. Because the description of the meaning of some of such new words or phrases will require new concepts, the list of candidates for ontology acquisition can also be augmented inductively.

The results of the acquisition must be validated for breadth and depth of coverage as well as for accuracy. Breadth of coverage relates to the number of lexical entries, depth of coverage relates to the grain size of the description of each individual entry. The appropriate breadth of coverage is judged by the rate at which an ontological semantic application obtains inputs that are not attested in the lexicon. The depth of coverage is determined by the disambiguation needs and capabilities of an application that determine the minimum number of senses that a lexeme should have. In other words, the specification of meaning should not contain elements that cannot be used by application programs. Accuracy of lexical and ontological specification can be checked effectively only by using the acquired static knowledge sources in a practical application and analyzing the failures in such applications. Many of these failures will have to be eliminated by tightening or relaxing constraints on the specification of the static knowledge sources.

4. Scripts

A spate of advanced new applications have called for a massive effort in script acquisition. Conceptualized as complex-events, they have been provided for in the ontology since its inception (see Carlson and Nirenburg, 1990) and their format has always been reasonably well-defined as well as constantly adjusted to the consecutive releases (see Nirenburg and Raskin, forthcoming, Section 7.1.5). Throughout the early and mid-1990s, however, lower-end NLP applications, such as knowledge- and meaning-based MT, did not necessitate a heavy use of scripts. The new generation of higher-end Q&A and similar IE applications make it necessary to recognize individual events and their effects as part of scripts, both because humans do and because such recognition is necessary for establishing (co)reference relations. Thus, in the following text, only the availability of the APPROACH-BANKRUPTCY script can relate (i) and (ii), which may be immediately adjacent in a text:

(i) ACME, Inc., was actually doomed the moment Jorge Jimenez and 52 other employees were laid off without a warning.

(ii) That bankruptcy was not, however, the last blow.

As an example, we will sketch out the creation process of the APPROACH-BANKRUPTCY script. Three stages of the process are discussed: acquisition of background knowledge, propositional representation, and ontological representation. It is assumed that the last stage is entered directly into the ontology, along with the appropriate senses of pertinent lexical entries, if not already contained in the lexicon. The process is premised, for now, on taking maximum advantage of the existing ontological concepts and minimizing the acquisition of new ones. This principle of parsimony, however, may not turn out to be the best guide in the process (see the discussion below).

Background knowledge for the acquisition of scripts can be sought in two different places: related literature and text corpora on the subject field. The former source is a good place to start, but it typically contains too much implicit knowledge, while the latter displays the actual input that our application will have to deal with, thus becoming an important clue to what needs to be made more explicit.

An analysis of bankruptcy-related texts would lead us to finding cues such as the following (extracted from Netscape News):

> "Kmart Corp., (…) declare for Chapter 11 bankruptcy protection. (…) has seen the major credit rating agencies cut their ratings for its debt in recent weeks. Its key food supplier, Fleming Cos., cut off shipments to Kmart, saying it is owed $78 million by Kmart."
>
> "(…) said Kmart has to restructure its debt, close unproductive stores and streamline at the corporate level."
>
> "(…) the company suffered heavy losses, closed stores and laid off employees."

The actual specific events leading to bankruptcy are still not made explicit enough. This is what we try to make up for at the next stage of the process, the propositional representation (see Figure 7), which also introduces the simple and transparent Boolean target notation.

```
HAS-PARTS           Value
               IF
                        OR   o Company has cash problems
                             o Company can't meet payroll
                             o Company lays off employees
                             o Company misses loan payment
                             o Company seeks loan
                             o Company's ratings are lowered

             THEN            ➢ Company may near bankruptcy
```

Figure 7: Propositional representation of script

```
IF
     AND
          Owe
                 Agent        Corporation-1
                 Beneficiary  Human
                                 employed-by  Corporation-1
                              Lending-institution
                              Corporation
                 Theme        Money
          Pay
                 Agent        Corporation-1
                 Beneficiary  Human
                                 employed-by
                              Lending-institution Corporation-1
                              Corporation
                 Theme        Money
THEN
          Bankrupt
                 Agent        Corporation-1
Modality
     Type   Potential
     Scope  Pay
     Value  0
Modality
     Type   Epistemic
     Scope  Bankrupt
     Value  0.6
```

Figure 8: Ontological representation of script

The script above is presented in its simplest and probably coarsest form. The gain is parsimony. Are there losses? The text above talked, for instance, about a supplier's refusal to ship stuff to the bankrupt corporation.

It does that because the corporation cannot pay them for the supplies. Can we consider it covered in the script? What if a text mentions the inability to meet the payroll? Meeting the payroll may deserve a script of its own. It may be seen to be covered sufficiently in the script above, but laying off employees may not. To owe a loan is actually to owe an installment payment on a certain date, and to be unable to pay the loan means, actually, the inability to pay an installment payment of the loan on a certain date. The script above also omits the entire ratings game.

The rationale for having the scripts is not, surprisingly, to do what Schank declared his group would a quarter of a century ago (Schank, 1975; Schank and Abelson, 1977) and, unlike them, to deliver a workable non-toy product, in which the whole script is evoked when any element of it at any level of the script hierarchy occurs lexically in the text.. The simplistic representation above obligates our analyzer to reduce any such pertinent lexical material to the level of owing and paying. Is it possible? The alternative is to develop much more elaborate scripts, involving a great deal more of ontological acquisition and change.

A more complex and more accurate level of representation, with all the intermediate subsidiary scripts embedded in other scripts as well as component simple events enriched with precondition and effect (and, we, increasingly believe, goal values), will be much costlier, so the question is whether the gain in analysis makes it worthwhile. We expect this to be dictated by the needs of the current and future applications as manifested in their goals and the nature of the texts in the pertinent corpora.

5. Current issues and goals

Currently, important efforts are being made which not only involve the implementation of new applications, but also seek to expand and improve many of the existing resources and tools, including the ontology, the monolingual lexicons for English and Spanish, the acquisition and data management tools and interfaces, and the analyzers. In particular, the following methodological goals are pursued:

The improvement and eventual automation of the entire acquisition process by establishing a coherent, unambiguous acquisition protocol and interactions between the different sources. Since content acquisition is central to the development of ontological semantics, it is crucial that these methodologies be optimized.

- The expansion of existing resources, leading toward closer integration between them and with the tools that manage them.
- The specification of better description and acquisition frameworks and interfaces for lexical information that accommodate to the idiosyncrasies of the languages involved, while maintaining the homogeneity across languages that the processors require.
- The fine-tuning of the processing modules and of the intermediate knowledge structures that the former generate from the input and the static knowledge sources.

It is important to realize that ontological semantics is not a finished product; in fact, it is constantly evolving as new implementations occur.

6. References

Bar-Hillel, Y., 1954. Indexical Expressions. *Mind*, 63: 559-379.

Carlson, L., and S. Nirenburg 1990. World Modeling for NLP. Technical report CMU-CMT-90-121. Center for Machine Translation, Carnegie Mellon University, Pittsburgh, PA. A short version appeared in *Proceedings of the 3rd Conference on Applied Natural Language Processing*, Trento, Italy: IRST-ITC, April 1990.

Genesereth, M.R., and R.E. Fikes, 1992. Knowledge Interchange Format, Version 3.0 Reference Manual. Technical Report Logic-92-1, Computer Science Department, Stanford University, Stanford, CA.

Ginsberg, M., 1991. Knowledge Interchange Format: the Kif of Death. *AI Magazine*, 163:5-63.

Gruber, T. R., 1993. A translation approach to portable ontology specifications. *Knowledge Acquisition* 5:199-220.

Minker, W., A. Waibel, and J. Mariani, 1999. *Sto-chastically-Based Semantic Analysis*. Boston-Dor-drecht-London: Kluwer.

Monarch, I., S. Nirenburg, and T. Mitamura, 1989. Ontology-Based Lexicon Acquisition for a Machine Translation System. *Proceedings of the Fourth Workshop on Knowledge Acquisition for Knowledge-Based Systems*, Banff, Canada. August.

Nirenburg, S. (ed.), 1994. The Pangloss Mark III Machine Translation System. A Joint Technical Report by NMSU CRL, USC ISI, and CMU CMT.

Nirenburg, S., V. Raskin, and B. Onyshkevych, 1995. *Apologiae ontologiae*. Memoranda in Computer and Cognitive Science MCCS-95-281. NMSU CRL.

Nirenburg, S., 2000a. CAMBIO: Progress Report. Working Paper, NMSU CRL.

Nirenburg, S., 2000b. CREST: Progress Report. Working Paper, NMSU CRL. Presented at the DARPA TIDES PI Meeting, Chicago, October.

Nirenburg, S. and V. Raskin, forthcoming. *Ontological Semantics*.

Schank, R.C., 1975. *Conceptual Information Processing*. Amsterdam: North Holland.

Schank, R.C. and R.P. Abelson, 1977. *Scripts, Plans, Goals and Understanding*. New York: Lawrence Erlbaum Associates.

Thomason, R.H. and D.S. Touretzky, 1991. Inheritance Theory and Networks with Roles. In J.F. Sowa (ed.) *Principles of Semantic Networks*. San Mateo, CA: Morgan Kaufmann.

Touretzky, D.S., 1984. Implicit Ordering of Defaults in Inheritance Systems. *Proceedings of the Fourth National Conference on Artificial Intelligence*. AAAI Press: 322-325.

Touretzky. D.S., 1986. *The Mathematics of Inheritance Systems*. Los Altos, CA: Morgan Kaufmann.

Tulving, E., 1985. How Many Memory Systems Are There? *American Psychologist*, 40: 385-398.