Language Engineering for Lesser-Studied Languages
S. Nirenburg (Ed.)
IOS Press, 2009

doi:10.3233/978-1-58603-954-7-183

Computational Field Semantics: Acquiring an Ontological-Semantic Lexicon for a New Language

Sergei NIRENBURG[1] and Marjorie MCSHANE

Institute for Language and Information Technologies
University of Maryland Baltimore County

Abstract. We present a methodology and tools that facilitate the acquisition of lexical-semantic knowledge about a language L. The lexicon that results from the process described in this paper expresses the meaning of words and phrases in L using a language-independent formal ontology, the OntoSem ontology. The acquisition process benefits from the availability of an ontological-semantic lexicon for English. The methodology also addresses the task of aligning any existing computational grammar of L with the expectations of the syntax-oriented zone of the ontological-semantic lexicon. Illustrative material in this paper is presented by means of the DEKADE knowledge acquisition environment.

Keywords. semantics, computational semantics, lexical acquisition, low-density languages

1. Introduction

1.1. What constitutes a comprehensive set of resources for a particular language?

These days one usually starts the work of developing resources for a particular language with the acquisition of textual corpora, either monolingual or parallel across two or more languages. Such corpora serve as the foundation for the various types of corpus-oriented statistics-based work that have been actively pursued over the past 20 years, machine translation being one of the most prominent end applications. There is, however, a consensus among workers in natural language processing that having at one's disposal formal knowledge about the structure and meaning of elements of a language L is truly beneficial for a broad variety of applications, including even corpus-based ones. This being the case, the questions arise, *What knowledge should be acquired*? and *How should knowledge acquisition be carried out*?

Consider how knowledge acquisition might begin. One can start by describing L's writing system, including punctuation marks, then describe L's conventions concerning word boundaries, the rendering of proper names, the transliteration of foreign words, and the expression of dates, numbers, currencies, abbreviations, etc. All of these

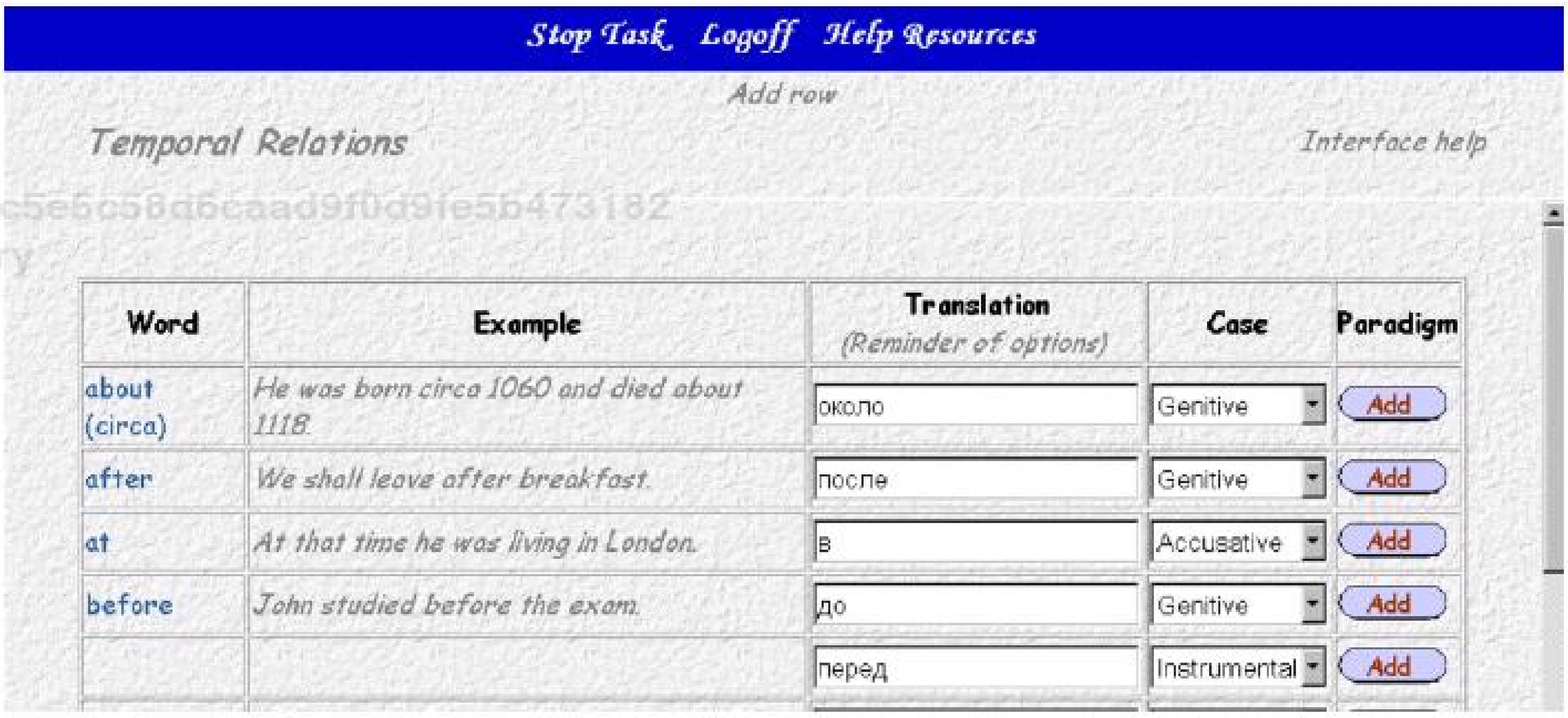
[1] Corresponding Author: Sergei Nirenburg, Department of Computer Science and Electrical Engineering, ITE 325, 1000 Hilltop Circle, Baltimore, Maryland, 21250, USA; E-mail: sergei@umbc.edu.

together comprise what the late Don Walker called language *ecology*. Next comes morphology – information about word structure in L. One should cover paradigmatic inflectional morphology (*run* ~ *runs*), non-paradigmatic inflectional morphology (e.g., agglutinating inflectional morphology, as found in Turkish), and derivational morphology (*happy* ~ *unhappy*).

Next, the structure of the sentence in L should be described. This would include, at a minimum: the structure of noun phrases – i.e., noun phrase (NP) components and their ordering; the realization of subcategorization and grammatical functions, like subject and direct object; the realization of sentence types – declarative, interrogative, etc.; and specialized syntactic structures such as fronting and clefting.

At this point, issues of meaning will come to the fore. First, one will have to deal with "grammatical" meanings in L – meanings that can be realized in various languages as words, phrases, affixes or features. For example, the notion of possession can be expressed by a genitive case marker in Russian, by the preposition *of* in English, and by free-standing pronouns in either language (*my, your*, etc.). Similarly, the fact that a noun phrase is definite can be realized in English by the definite article (*the*), in French by a free-standing word (*le, la, les*) or prefix (*l'-),* and in Bulgarian by a suffix (*-to, -ta, -'t*, etc.). One could expect to have to account for about 200 such grammatical meanings in L. These language-specific realizations will be stored in the so-called *closed-class lexicon* of L, which is the portion of the lexicon that, under normal circumstances, cannot be productively added to by language users – except over very long spans of historical change.

Figure 1 shows a closed-class elicitation screen from the Boas knowledge elicitation system – a system that elicits computer-tractable knowledge about low-density languages from non-linguist speakers of the language.[2]



Temporal Relations

Word	Example	Translation (Reminder of options)	Case	Paradigm
about (circa)	He was born circa 1060 and died about 1118.	около	Genitive	Add
after	We shall leave after breakfast.	после	Genitive	Add
at	At that time he was living in London.	в	Accusative	Add
before	John studied before the exam.	до	Genitive	Add
		перед	Instrumental	Add

Figure 1. Closed-class lexical acquisition in the Boas system.

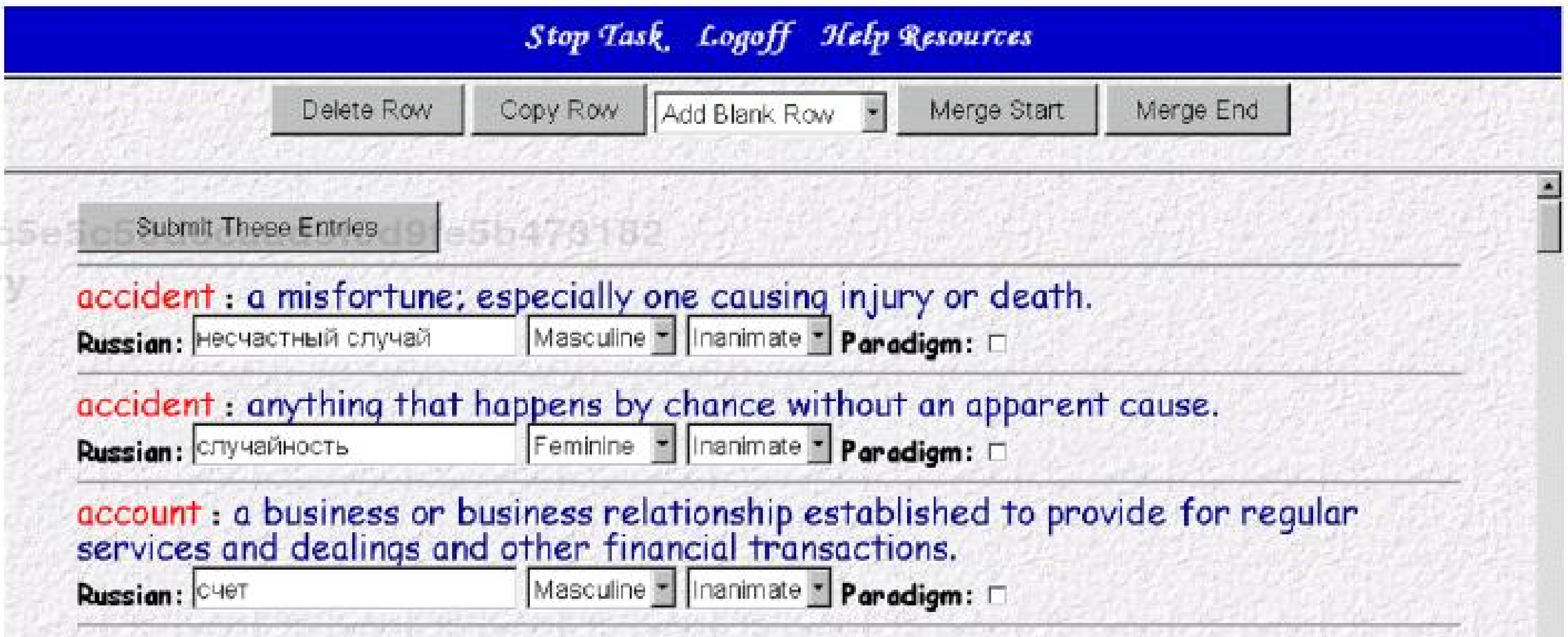
The first column provides an English "prompt" for the sense being elicited (the system assumes that all language informants know English), and the second column

[2] For further description of the Boas system see [6], [7], [8], [9], [14]. For another approach to gathering and processing knowledge for low-density languages, see [20].

provides an illustrative example of how this sense is used. The third column seeks one or more L equivalents for this meaning; note the "Add row" button at the top of the screen, which permits any number of additional rows to be added to the table if more than one realization of a given meaning is possible. The "Reminder of options" button links to a help page that describes all possible means of realizing closed-class meanings cross-linguistically: e.g., as a word, affix, case feature, etc. It also describes how various types of entities should be entered: for example, suffixes are preceded by a hyphen: *-to* is the suffix *to*. The fifth row, Case, is included for those languages that have inflectional case-marking. Since the screen shot was made from an elicitation session for Russian, this column is present and the inventory of cases in the pull-down menu is exactly those that are relevant for Russian. The last column permits the user to enter the inflectional paradigm for the given item, if applicable. Very often, if closed-class meanings have paradigms, they are idiosyncratic; therefore, users are asked to enter the paradigms for closed-class meanings explicitly. The information about a given language that permits the fourth and fifth columns to be catered to that language is elicited prior to the start of work on building the closed-class lexicon. This example shows the types of information that must be elicited in the closed-class lexicon and some practical decisions that were made in building a cross-linguistically robust knowledge elicitation system.

As mentioned earlier, the closed-class lexicon of any language is relatively small. The much larger portion of the lexicon is the *open-class lexicon*, which for many languages will contain nouns, verbs, adjectives and adverbs.[3] Unlike the closed-class lexicon, the open-class lexicon can be added to by language users – in fact new nouns and verbs are coined at a great rate, necessitating the constant updating of lexicons.

Figure 2 shows a screen shot of the Boas open-class elicitation environment, again using an example from Russian.



accident : a misfortune; especially one causing injury or death.
Russian: несчастный случай Masculine Inanimate Paradigm: □

accident : anything that happens by chance without an apparent cause.
Russian: случайность Feminine Inanimate Paradigm: □

account : a business or business relationship established to provide for regular services and dealings and other financial transactions.
Russian: счет Masculine Inanimate Paradigm: □

Figure 2. Open-class lexical acquisition in the Boas system.

Like the closed-class interface, the closed-class interface reflects information collected through pre-lexicon knowledge elicitation:

1) The informant posited two inherent features for Russian nouns: one with at least the values masculine and feminine, and the other with at least the value

[3] For different languages, different parts of speech might be utilized for both the closed-class and the open-class lexicon. We will not pursue the complex issue of part-of-speech delineation here.

inanimate (there are actually more feature values but they are not shown in this screen shot).

2) The informant has created inflectional paradigms for Russian, otherwise the "Paradigm" checkbox – which is used to indicate that there is an irregular inflectional paradigm – would not be present.

3) The informant does not think that any of the entries in L has irregular inflectional forms, since no checkboxes are checked. All words that have regular inflectional forms are interpreted based on rules created during the morphological stage of knowledge acquisition.

Since open-class acquisition is a big job, interface functions are provided to speed the process:

- *Delete Row* is used to remove a word from the list and put it into a trash bin. This is for words that cannot be translated or are not important enough in L to be included. The cursor must be in the text field of the given row before clicking on Delete Row. After clicking on it, the screen refreshes with that row missing. (These cursor and refresh comments apply to most functionalities and will not be repeated.)

- *Copy Row* is used when there is more than one translation for a given prompt. For example, there are two Russian words for English *blue* – one meaning *light blue* and the other meaning *dark blue* (there is no umbrella word for *blue*). Multiple translations must be typed in separate rows because they might have different inherent features, or one might be a word whereas another is a phrase, or one or both might have irregular inflectional forms.

- *Add Blank Row* is used to add a completely new entry for which variants in both English and L must be provided. Add Blank Row is actually not a button but a pull-down menu requiring the informant to indicate which part of speech the new item will belong to, since L might require different kinds of information for different parts of speech (e.g., nouns might have inherent features whereas verbs do not); therefore, it is important that a new row of the right profile be added. This function permits the informant to add, on the spot, entities that occur to him during work on the open class—like idioms, phrases, or compounding forms based on a word just translated.

- *Merge Start and Merge End* are a pair of functions that permit the informant to bunch word senses that have the same translation, thus reducing acquisition time, especially if a given entity in L requires additional work, like listing irregular inflectional forms.

Since speed is at the center of the interface design, keyboard-centered methods of working with the interface are encouraged. For example, tabbing takes the user from one action point to the next and if some variety of a Latin keyboard is being used, typing in the first letter of a given word in a drop-down menu will pull up that word.

In this paper, we discuss the acquisition of open-class lexical material. However, the type of lexical information to be focused on is "deeper" than that elicited in Boas. The difference is motivated by the fact that the Boas system was designed to feed into a quick ramp-up machine translation system. Since the focus was on *quick* ramp-up, relatively broad coverage was more important than deep coverage. Other systems, by contrast, benefit from depth of coverage, defined as precise and extensive syntactic and semantic information about each lexical item. It is lexical coverage for the latter types of high-end systems that is the focus here.

1.2. What is needed for processing meaning?

There are many opinions about what constitutes lexical meaning and what level of its specification is sufficient for what types of computational applications (see, e.g., [3]). In this paper we will follow the approach developed in Ontological Semantics, a theory of processing meaning that is implemented in the OntoSem semantic analyzer. In this approach, the goal of text analysis is creating unambiguous, formally interpreted structures that can be immediately used by automatic reasoning programs in high-end applications such as question answering, robotics, etc. A comprehensive description of the theory is beyond the scope of this paper. The most detailed description to-date is [19]. Descriptions of various facets of OntoSem can be found in [1], [2], [10], [12], [13], [15], [16].

OntoSem is essentially language-independent: it can process text in any language provided appropriate static knowledge resources are made available, with only minor modifications required of the processors. In what follows, we suggest a method for creating such knowledge resources for any language L. We concentrate on the knowledge related to the description and manipulation of lexical and compositional meaning. We demonstrate that the availability of a language-neutral ontology and a semantic, OntoSem-compatible, lexicon of English simplifies the task of acquiring the lexical-semantic components of the lexicon for L. Knowledge of non-semantic components of a language – notably, its morphology and syntax – must also be acquired, as it is important as the source of heuristics for semantic processing. The OntoSem resources provide help in formulating the syntactic knowledge of L because the system uses a lexicalized grammar, the majority of the knowledge for which is recorded in the syn-struc of lexicon entries.

There are four main knowledge resources in OntoSem: the lexicon, the ontology, the onomasticon (the lexicon of proper names) and the fact repository (the inventory of remembered instances of concepts: instances of real-world objects events as contrasted with the object and event types found in the ontology). We focus on the first two types of resources in this paper.

2. The OntoSem Ontology

The OntoSem ontology is used to ground meaning in an unambiguous model of the world. It contains specifications of concepts corresponding to classes of objects and events. Formatwise, it is a collection of frames, or named collections of property-value pairs, organized into a directed acyclic graph – i.e., a hierarchy with multiple inheritance.[4] Concepts are written in a metalanguage that resembles English (e.g., DOG,

[4] The use of multiple inheritance is not unwieldy because (a) the inheritance relation is always semantically "is-a", and (b) the ontology contains far fewer concepts than any language would have words/phrases to express those concepts. Contrast this with, for example, with MeSH (http://www.nlm.nih.gov/mesh/) and Metathesaurus (http://www.nlm.nih.gov/pubs/factsheets/umlsmeta.html), which are partially overlapping

WHEELED-VEHICLE, MENTAL-EVENT) but, unlike English words and phrases, concepts are unambiguous: DOG refers only to a domesticated canine, not a contemptible person or the act of tracking someone persistently. Therefore, although the concept DOG looks like the English word 'dog' (which is a convenient approach for the people building and maintaining the knowledge base) they are not equivalent.

The ontology is language-independent, and its links to any natural language are mediated by a lexicon. For example, the English lexicon indicates that one sense of *dog* maps to the concept DOG, another sense maps to HUMAN (further specified to indicate a negative evaluative modality), and yet another sense maps to the event PURSUE. Therefore, the ontology can be used to support language processing and reasoning in any language, given an ontologically linked lexicon for that language. The top levels in the OntoSem ontology are shown in Figure 3.

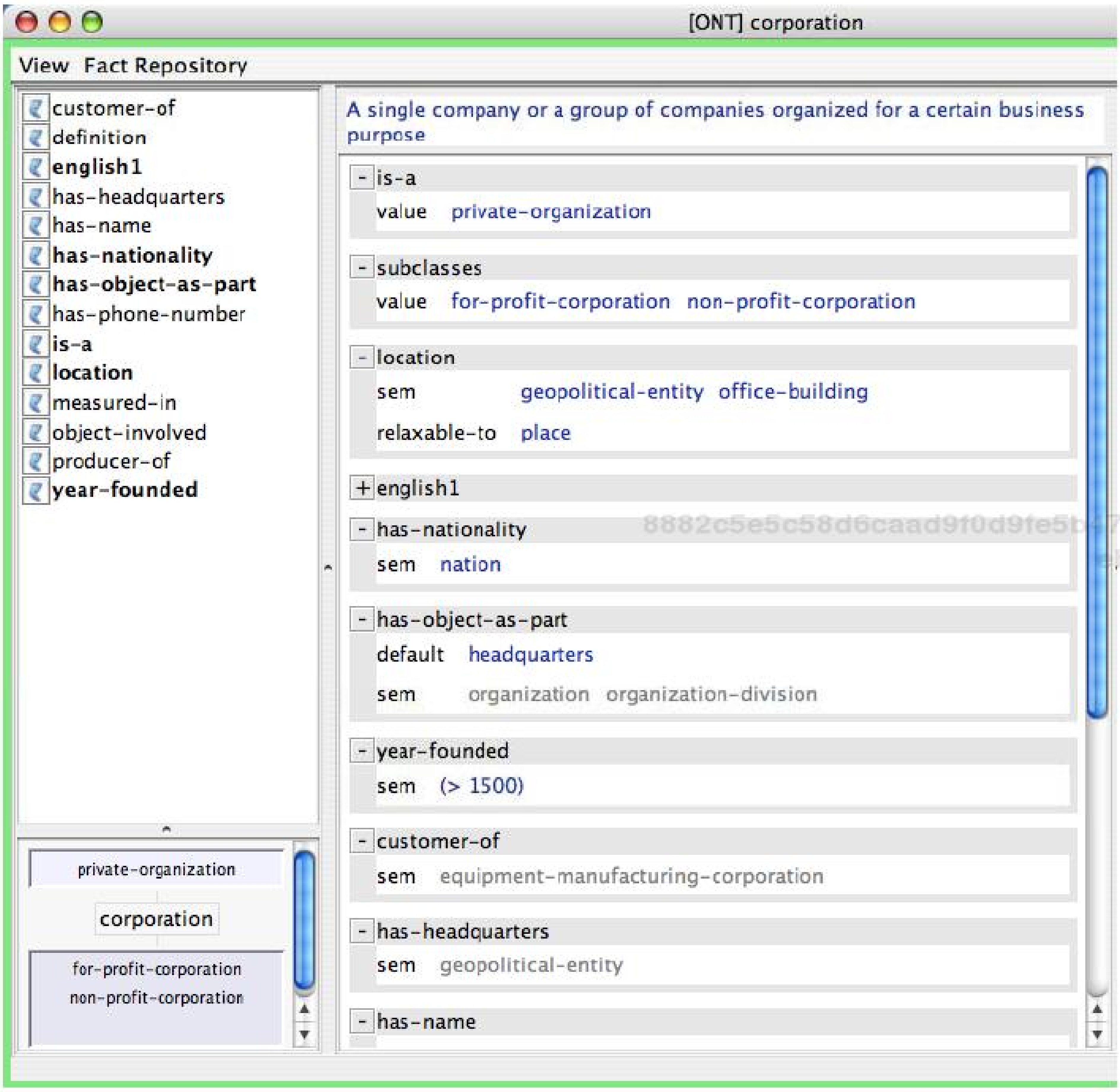
- ALL
  - EVENT
    - MENTAL-EVENT
    - PHYSICAL-EVENT
    - SOCIAL-EVENT
  - OBJECT
    - INTANGIBLE-OBJECT
    - MENTAL-OBJECT
    - PHYSICAL-OBJECT
    - SOCIAL-OBJECT
    - TEMPORAL-OBJECT
  - PROPERTY
    - ATTRIBUTE
    - RELATION

Figure 3. The top levels of the OntoSem ontology.

The PROPERTY subtree contains properties that are used to describe OBJECTs and EVENTs. In fact, the meaning of a concept *is* the set of property values used to describe it, such that concepts mean something with respect to other concepts within this model of the world. For people's use, a definition is provided for each concept, which not only provides a quick snapshot of the meaning but also acts as a bridge until all concepts can be described sufficiently to fully differentiate them from other concepts (the latter is, of course, a long-term knowledge acquisition effort).

An excerpt from the ontological frame for CORPORATION is shown in Figure 4. The upper section of the left-hand pane shows a subset of the features defined for this concept; those in boldface have locally specified values. The lower left pane is a snapshot of the parent(s) and child(ren) of this concept. The right-hand shows properties and their values; those in blue are locally defined whereas those in gray are inherited.

ontologies of medical terms developed by the National Library of Medicine. In these resources, many lines of inheritance (even 10 or more) are common, with the semantics of "parenthood" varying significantly. (For a description of our attempts to use these resources for automatic ontology population, see [17].



customer-of
definition
english1
has-headquarters
has-name
has-nationality
has-object-as-part
has-phone-number
is-a
location
measured-in
object-involved
producer-of
year-founded

private-organization
corporation
for-profit-corporation
non-profit-corporation

A single company or a group of companies organized for a certain business purpose

- is-a
  - value private-organization
- subclasses
  - value for-profit-corporation non-profit-corporation
- location
  - sem geopolitical-entity office-building
  - relaxable-to place
- english1
- has-nationality
  - sem nation
- has-object-as-part
  - default headquarters
  - sem organization organization-division
- year-founded
  - sem (> 1500)
- customer-of
  - sem equipment-manufacturing-corporation
- has-headquarters
  - sem geopolitical-entity
- has-name

Figure 4. An excerpt from the OntoSem ontological frame for CORPORATION.

The precision and depth of property-based descriptions of concepts varies from domain to domain. For example, there are currently no property-based differences between the ontological siblings EAGLE and EMU since none of our applications have given priority to describing the animal kingdom; however, such distinctions must ultimately be included to permit artificial agents to reason with the same nimbleness that a human brings to the task. The machine learning of property values to distinguish between OBJECTs has actually been the focus of a recent experiment, as we attempt to bootstrap our hand-crafted resources using machine learning techniques (Nirenburg and Oates 2007).

Selectional restrictions in the ontology are multivalued, with fillers being introduced by one of five facets. The *value* facet is rigid and is used less in the ontology than in the sister knowledge base of real-world assertions, the fact repository. The facets *default* (for strongly preferred constraints) and *sem* (for basic semantic constraints) are abductively overridable. The *relaxable-to* facet indicates possible but atypical restrictions, and *not* blocks the given type of filler. For example, the AGE of

COLLEGE-STUDENT is described as default 18-22, sem 17-26, relaxable-to 13-80, with the latter accounting for kid geniuses and retirees going back to school.

Slot fillers can be concepts, literals or frames, the latter used not only for scripts (i.e., fillers of the property HAS-EVENT-AS-PART) but also for other cases of reification:

concept	property	facet	filler
CAR	HAS-OBJECT-AS-PART	sem	WHEEL (CARDINALITY default 4)

The number of concepts in the OntoSem ontology, currently around 9,000, is far fewer than the number of words or phrases in any language for several reasons:

1. Synonyms (*apartment ~ flat*) and hyponyms (*hat ~ beret*) are mapped to the same ontological concept, with semantic nuances recorded in the corresponding lexical entries. Theoretically speaking, any "synonym" could actually be analyzed as a "near synonym" (cf. [5]) since no two words are *precisely* alike. However, for practical reasons a slightly coarse grain size of description is pursued in OntoSem.
2. Many lexical items are described using a combination of concepts. For example, the event of *asphalting*, as in *The workers asphalted the parking lot*, is lexically described as COVER (INSTRUMENT ASPHALT), understood as "to cover with asphalt."
3. Many lexical items are described using non-ontological representational means like values for aspect or modality. For example, the inceptive phase can be indicated in English by the word *start*, as in *He started running*; and the volitive modality can be indicated by the word *want,* as in *He wanted to win the race.*
4. Meanings that can be captured by scalar attributes are all described using the same scale, with different words being assigned different numerical values. For example, using the scalar attribute INTELLIGENCE, whose values can be any number or range on the abstract scale {0,1}, *smart* is described as (INTELLIGENCE (> .8)) whereas dumb is described as (INTELLIGENCE (< .2)).
5. Concepts are intended to be cross-linguistically and cross-culturally relevant, so we tend not to introduce concepts for notions like *to asphalt* (cf. above) or *to recall* in the sense of a company recalling a purchased good because it is highly unlikely that all languages/cultures use these notions. Instead, we describe the meaning of such words compositionally in the lexicons of those languages that do use it.

3. The OntoSem lexicon

Even though we refer to the OntoSem lexicon as being a semantic lexicon, it contains not only semantic information: it also supports morphological and syntactic analysis and generation. Semantically, it specifies what concept, concepts, property or properties of concepts defined in the ontology must be instantiated in the text-meaning representation to account for the meaning of a given lexical unit of input. Lexical entries are written in an extended Lexical-Functional Grammar formalism using LISPcompatible format. The lexical entry – in OntoSem, it is actually called a *superentry* – can contain descriptions of several lexical senses; we call the latter *entries*. Each entry (that is, the description of a particular word sense) contains a number of fields, called zones. The skeleton for an OntoSem lexicon entry is illustrated below. The purpose of each zone is briefly explained as comments. Underscores show that values for these fields must be filled in. In some cases the values are strings ("__") and in other cases they are structures (__).[5]

```
(word
   (word-pos1                    ; part of speech & sense number
      (cat __)                   ; part of speech
      (def " __ ")               ; definition in English
      (ex " __ ")                ; example(s)
      (comments " __ "))         ; acquirer's comments
      (syn-struc __ )            ; syntactic dependency
      (sem-struc __ )            ; semantic dependency
      (synonyms  "__")           ; string(s) with (almost) the same meaning
      (hyponyms "__")            ; string(s) with a more specific meaning
      (abbrev "__")              ; abbreviation(s)
      (sublang "__")             ; subject domain, e.g., medicine
      (tmr-head __ )             ; semantic head of atypical phrasals6
      (output-syntax __)         ; overall syntactic category of atypical phrasals
      (meaning-procedure __ )) ; call to a procedural semantic routine
   (word-pos2 …)
   …
   (word-posN …))
```

Figure 5. The structure of an OntoSem lexicon entry.

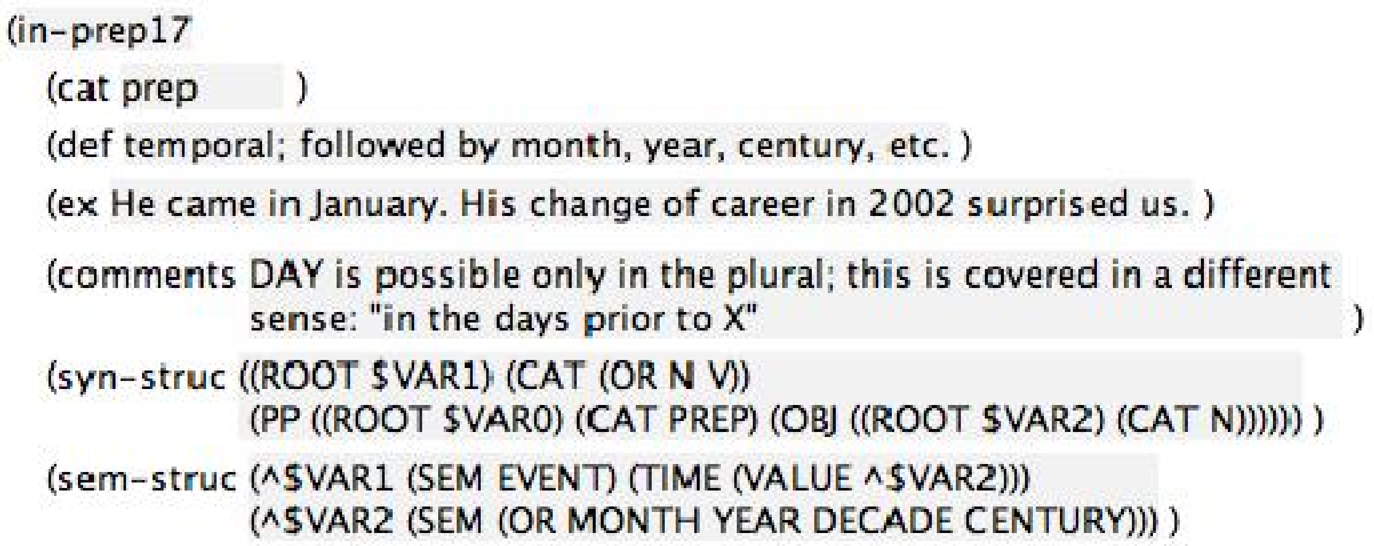
The OntoSem lexicon directly supports the dependency-oriented description of syntax of L, so if a dependency grammar for L exists, it can be adapted to the OntoSem environment. If such a grammar does not exist, the acquisition of the OntoSem-style lexicon for L will aid in developing such a grammar by providing subcategorization information for the lexicon entries of L.

The central zones of a lexicon entry are the *syn-struc*, which describes the syntactic dependency constraints of the word, and the *sem-struc*, which describes the word's meaning. In fact, these two zones, along with *cat*, are the only ones that **must** appear in each lexicon entry (the definition and example zones are for the convenience of acquirers). As an example, consider the seventeenth sense of *in* (Figure 6) in the OntoSem English lexicon, as shown in the DEKADE development environment (see [16] for a description of DEKADE).[7]

[5] Note that in the upcoming screen shots of OntoSem lexical entries the distinction between strings and structures is not overt, but it is understood by the OntoSem analyzer.

[6] The fields *output-syntax* or *tmr-head* tell the parser how to treat phrasal entries that are composed of a series of immediate constituents (e.g., np, adj) rather than syntactic functions (e.g., subject, direct object).

[7] Here and hereafter, in making screen shots we show only those fields that are relevant, often leaving out the last 7 fields of the entry, starting with *synonyms*.



```
(in-prep17
  (cat prep       )
  (def temporal; followed by month, year, century, etc. )
  (ex He came in January. His change of career in 2002 surprised us. )
  (comments DAY is possible only in the plural; this is covered in a different
              sense: "in the days prior to X"                                      )
  (syn-struc ((ROOT $VAR1) (CAT (OR N V))
              (PP ((ROOT $VAR0) (CAT PREP) (OBJ ((ROOT $VAR2) (CAT N)))))) )
  (sem-struc (^$VAR1 (SEM EVENT) (TIME (VALUE ^$VAR2)))
              (^$VAR2 (SEM (OR MONTH YEAR DECADE CENTURY))) )
```

Figure 6. One lexical sense of the word *in.*

The syntactic structure (syn-struc) indicates that the input covered by this sense of *in* should contain a constituent headed by a noun (n) or verb (v) followed by a prepositional phrase (pp). All syntactic elements in the syn-struc are associated with variables, which permit their linking to semantic elements in the sem-struc. The variable associated with the head word, here *in*, is always $var0; it does not have an explicit sem-struc linking since the whole entry is describing the meaning of $var0 in a particular type of context.

The sem-struc says that the meaning of $var1 ("meaning of" is indicated by a caret (^)) is some ontological EVENT whose time is the same as the time of the meaning of $var2. Moreover, it is specified that the meaning of $var2 must represent a MONTH, YEAR, DECADE or CENTURY. This entry predicts that one cannot say, for example, **in Monday*, since *Monday* is an instance of the ontological concept DAY.

The linking of syntactic and semantic elements is not always straightforward, as can be shown by a few examples:

- More than one entity can have a given case-role: e.g., in the sense of *argue* that covers the input *He argued with me about sports,* both the subject (*he*) and the object of the preposition (*me*) are AGENTS of an ARGUE-CONFLICT event. Similarly, when the sentence *They asphalted the road using huge trucks* is analyzed, a COVER event will be instantiated whose INSTRUMENTS are both ASPHALT and TRUCK ((CARDINALITY > 1) (SIZE > .9)). That is, the word *asphalt* is lexically described as COVER (INSTRUMENT ASPHALT); the instrumental interpretation of *huge trucks* is analyzed on the fly.

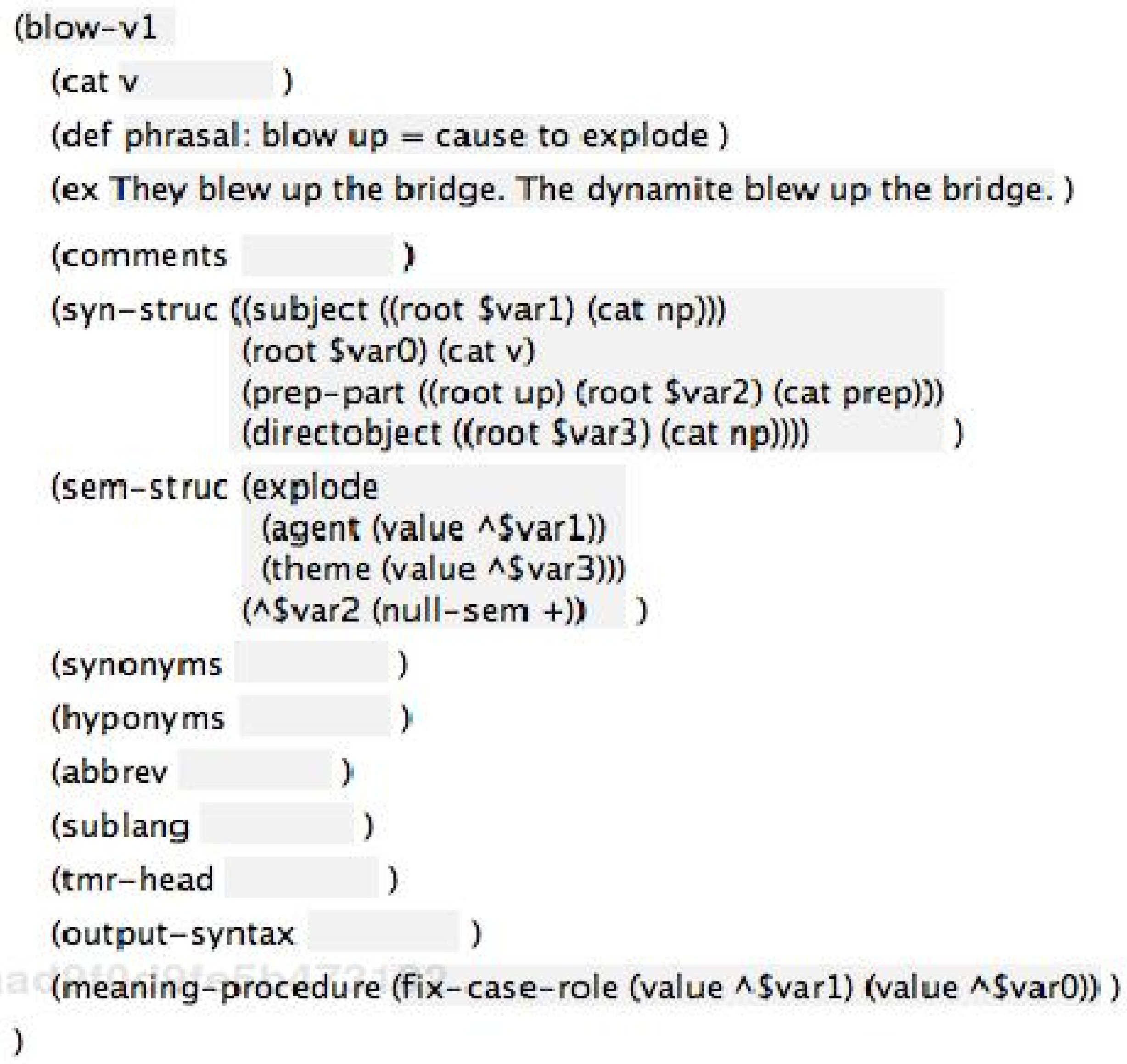
- A given entity can have more than one semantic role: e.g., in the sense of *coil* that covers the input *The snake coiled itself around the tree*, SNAKE is both the AGENT and the INSTRUMENT of COIL (the concept COIL also covers people coiling objects like rope, etc.).
- In some cases, elements of the syn-struc are nullified in the semantic structure, blocking their compositional analysis. This occurs most typically with prepositions within PP arguments or adjuncts of another head word. For example, in the lexical sense for *turn in*, as used in the input *He turned in his homework* (which is mapped to the concept GIVE), the meaning of *in* is nullified because its meaning is folded into the central mapping of *turn in* to the concept GIVE.

In the subsections to follow we describe and provide examples of a number of theoretical and practical advances reflected in the OntoSem lexicon.

3.1. Treatment of Multiword Entities (Phrasals)

Among OntoSem's lexical advances is the robust treatment of multiword elements, what we call phrasals. Phrasals in OntoSem can contain any combination of lexically fixed and ontologically constrained elements. Space does not permit a full description of all types of multi-word elements so rather than attempt a full categorization, we provide just a few examples for illustration.

Example 3.1.1 Two phrasal senses of the verb *blow* are shown in Figures 7 and 8. The first sense is for a transitive sense of *blow up*.



```
(blow-v1
  (cat v          )
  (def phrasal: blow up = cause to explode )
  (ex They blew up the bridge. The dynamite blew up the bridge. )
  (comments          )
  (syn-struc ((subject ((root $var1) (cat np)))
              (root $var0) (cat v)
              (prep-part ((root up) (root $var2) (cat prep)))
              (directobject ((root $var3) (cat np))))          )
  (sem-struc (explode
               (agent (value ^$var1))
               (theme (value ^$var3)))
             (^$var2 (null-sem +))    )
  (synonyms          )
  (hyponyms          )
  (abbrev          )
  (sublang          )
  (tmr-head          )
  (output-syntax          )
  (meaning-procedure (fix-case-role (value ^$var1) (value ^$var0)) )
)
```

Figure 7. An example of the part of speech *prep-part* in a lexicon entry.

The default case-role for the subject is agent, but if the meaning of $var1 cannot be agentive (e.g., *dynamite*), then the procedural routine "fix-case-role" is used to select a more appropriate case-role – here, instrument (see Section 3.2 for further description of procedural semantic routines).

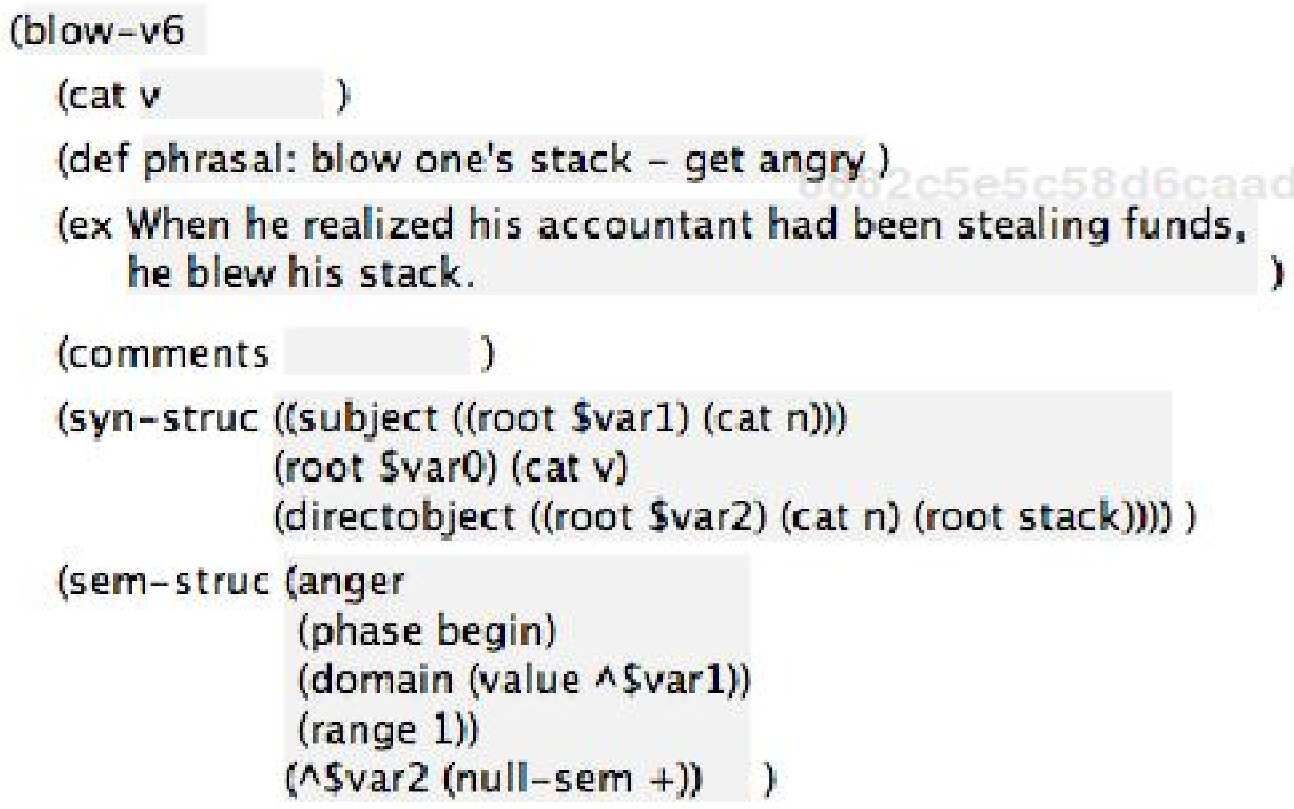
There are three reasons why the phrasal *blow up* is not listed as a multi-word head word (as, e.g., *child care* would be):

(1) The first word can inflect and therefore must be productively analyzed, not "frozen".

(2) This phrasal can be used with two different word orders: the particle *up* can come before the object (*He blew up the bridge*) or after the object (*He blew the bridge up*). If this phrasal could only be used with the former word order, then instead of describing *up* as "prep-part" (prepositional particle), we would describe it as a preposition and use a standard prepositional phrase.

(3) Intervening material can come between the components: e.g., one can say *He blew the bridge right up*.

Sense 6 of *blow,* shown in Figure 8, shows *another* sense of the word *blow.*

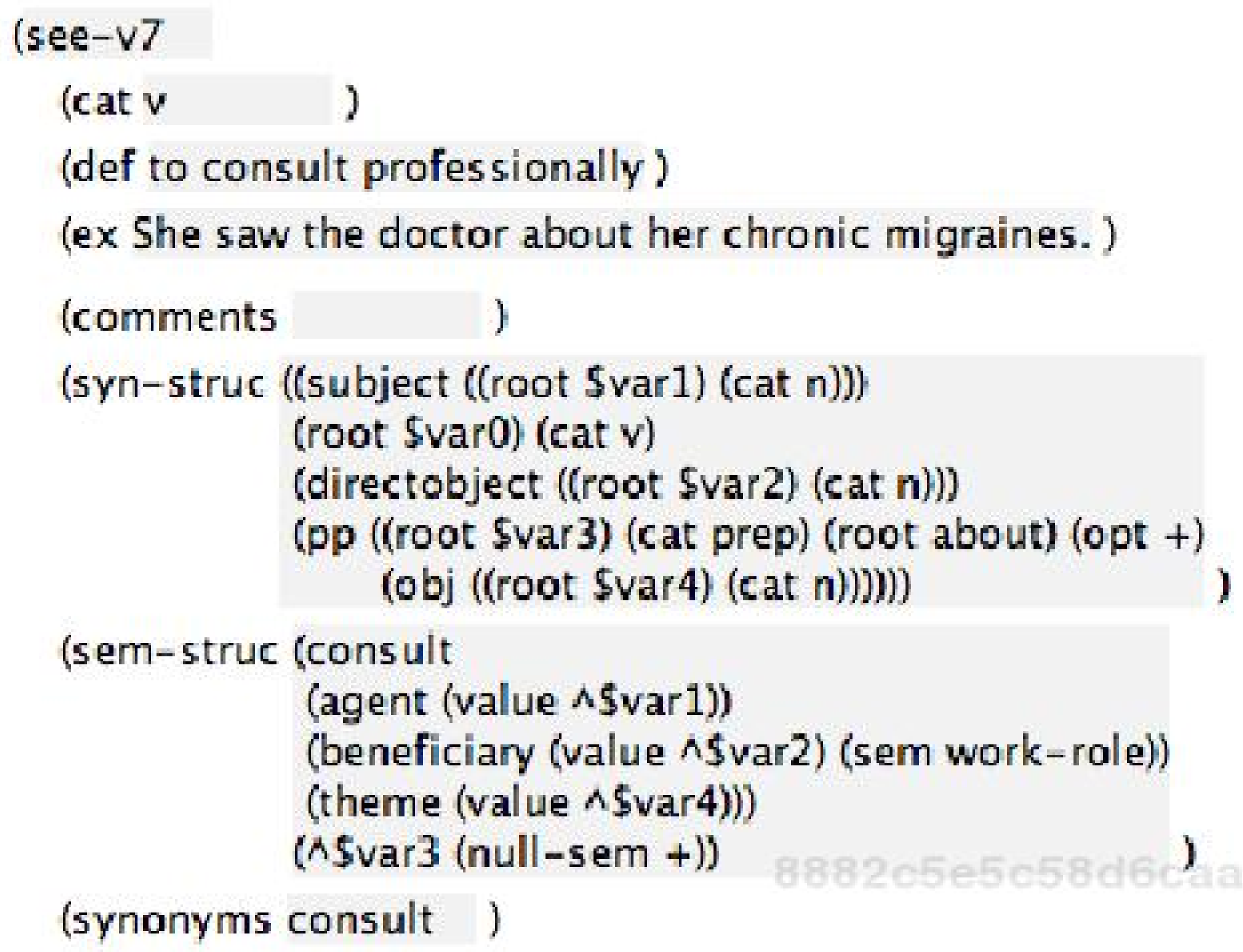


```
(blow-v6
   (cat v            )
   (def phrasal: blow one's stack - get angry )
   (ex When he realized his accountant had been stealing funds,
       he blew his stack.                                          )
   (comments              )
   (syn-struc ((subject ((root $var1) (cat n)))
               (root $var0) (cat v)
               (directobject ((root $var2) (cat n) (root stack)))) )
   (sem-struc (anger
                (phase begin)
                (domain (value ^$var1))
                (range 1))
              (^$var2 (null-sem +))     )
```

Figure 8. An example of a lexically specified direct object.

Syntactically, this is a typical transitive sense except that the head of the direct object must be the word *stack* – or the plural *stacks*, since no number is specified. Semantically, however, the words *blow* and *stack* are not compositional—together they mean *get angry*. This meaning is shown by the scalar attribute *anger* whose domain (the person who is angry) is the meaning of the subject of the sentence, and whose range is the highest possible value on the abstract scale {0,1}. The feature "(phase begin)" shows that this phrasal is typically inceptive in meaning: i.e., the person just begins to be extremely angry. The meaning of $var2 is attributed null semantics since it is not compositional.

Example 3.1.2 The next example, sense 7 of the verb *see* (Figure 9), shows how the meaning of sem-struc elements can be constrained in order to permit automatic disambiguation.



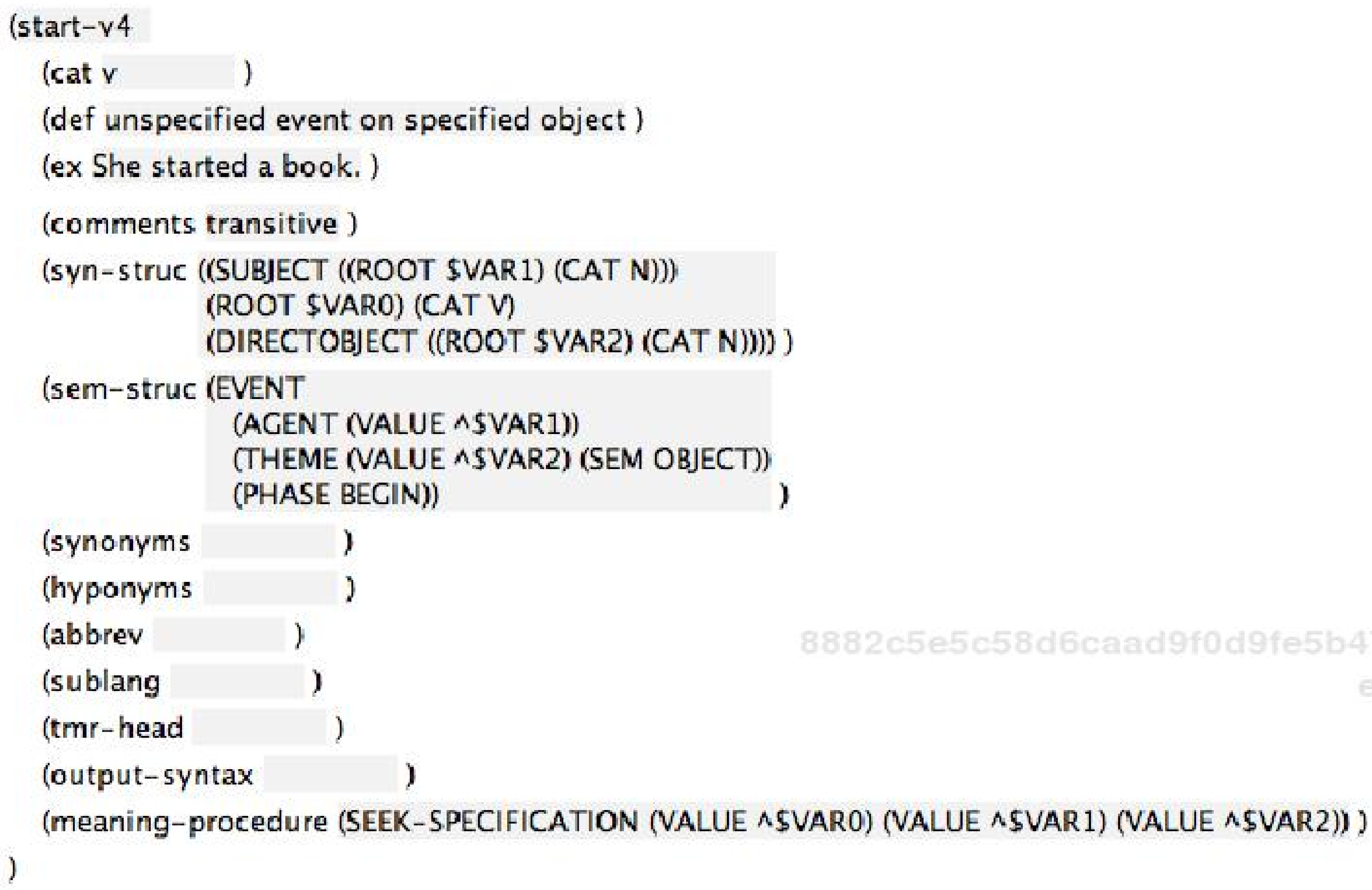
```
(see-v7
  (cat v          )
  (def to consult professionally )
  (ex She saw the doctor about her chronic migraines. )
  (comments          )
  (syn-struc ((subject ((root $var1) (cat n)))
              (root $var0) (cat v)
              (directobject ((root $var2) (cat n)))
              (pp ((root $var3) (cat prep) (root about) (opt +)
                  (obj ((root $var4) (cat n))))))                )
  (sem-struc (consult
               (agent (value ^$var1))
               (beneficiary (value ^$var2) (sem work-role))
               (theme (value ^$var4)))
              (^$var3 (null-sem +))                              )
  (synonyms consult   )
```

Figure 9. Example of a semantic constraint in the sem-struc.

The key aspect of this structure is that the beneficiary – the person whom one sees – is ontologically a WORK-ROLE. So, if one sees the doctor (PHYSICIAN < MEDICAL-WORK-ROLE < WORK-ROLE) about a headache, sees a mechanic (MECHANIC < TRADE-ROLE < WORK-ROLE) about a clunk in one's car engine, or sees a lawyer (ATTORNEY < LEGAL-ROLE < WORK-ROLE) about divorce proceedings, this sense will be chosen. Of course, one can also *see* any of these people in the sense "visually perceive", which is sense see-v1 in our lexicon. This type of true ambiguity must be resolved contextually by the semantic analyzer.

3.2. Calls to Procedural Semantic Routines

Another advance in the OntoSem lexicon is the inclusion of calls to procedural semantic routines to resolve the meanings of entities that cannot be interpreted outside of context. Although deictic elements, like *you* and *yesterday*, are the most famous of such elements, the need for procedural semantics actually radiates much wider: for example, any time the English aspectual verb *start* (Figure 10) has an OBJECT rather than an EVENT as its complement, as in *She started the book*, the semantically elided event in question must be recovered. This recovery is carried out by the routine called "seek-specification", which attempts to determine the meaning of the head entry (some sort of EVENT) using the meaning of the subject and the meaning of the object as input parameters. The ontology is used as the search space. This routine will return READ and WRITE as equally possible analyses based on the fact that both of these are ontologically defined to have their DEFAULT THEME be DOCUMENT (BOOK < BOOK-DOCUMENT < DOCUMENT).



```
(start-v4
  (cat v          )
  (def unspecified event on specified object )
  (ex She started a book. )
  (comments transitive )
  (syn-struc ((SUBJECT ((ROOT $VAR1) (CAT N)))
              (ROOT $VAR0) (CAT V)
              (DIRECTOBJECT ((ROOT $VAR2) (CAT N)))) )
  (sem-struc (EVENT
                (AGENT (VALUE ^$VAR1))
                (THEME (VALUE ^$VAR2) (SEM OBJECT))
                (PHASE BEGIN))                          )
  (synonyms          )
  (hyponyms          )
  (abbrev          )
  (sublang          )
  (tmr-head          )
  (output-syntax          )
  (meaning-procedure (SEEK-SPECIFICATION (VALUE ^$VAR0) (VALUE ^$VAR1) (VALUE ^$VAR2)) )
)
```

Figure 10. An example of a call to a procedural semantic routine.

As presented earlier, another procedural semantic routine fixes case roles if the listed case role is not compatible with the type of semantic element filling that role. Still other routines are used to resolve the reference of pronouns and other deictic elements.

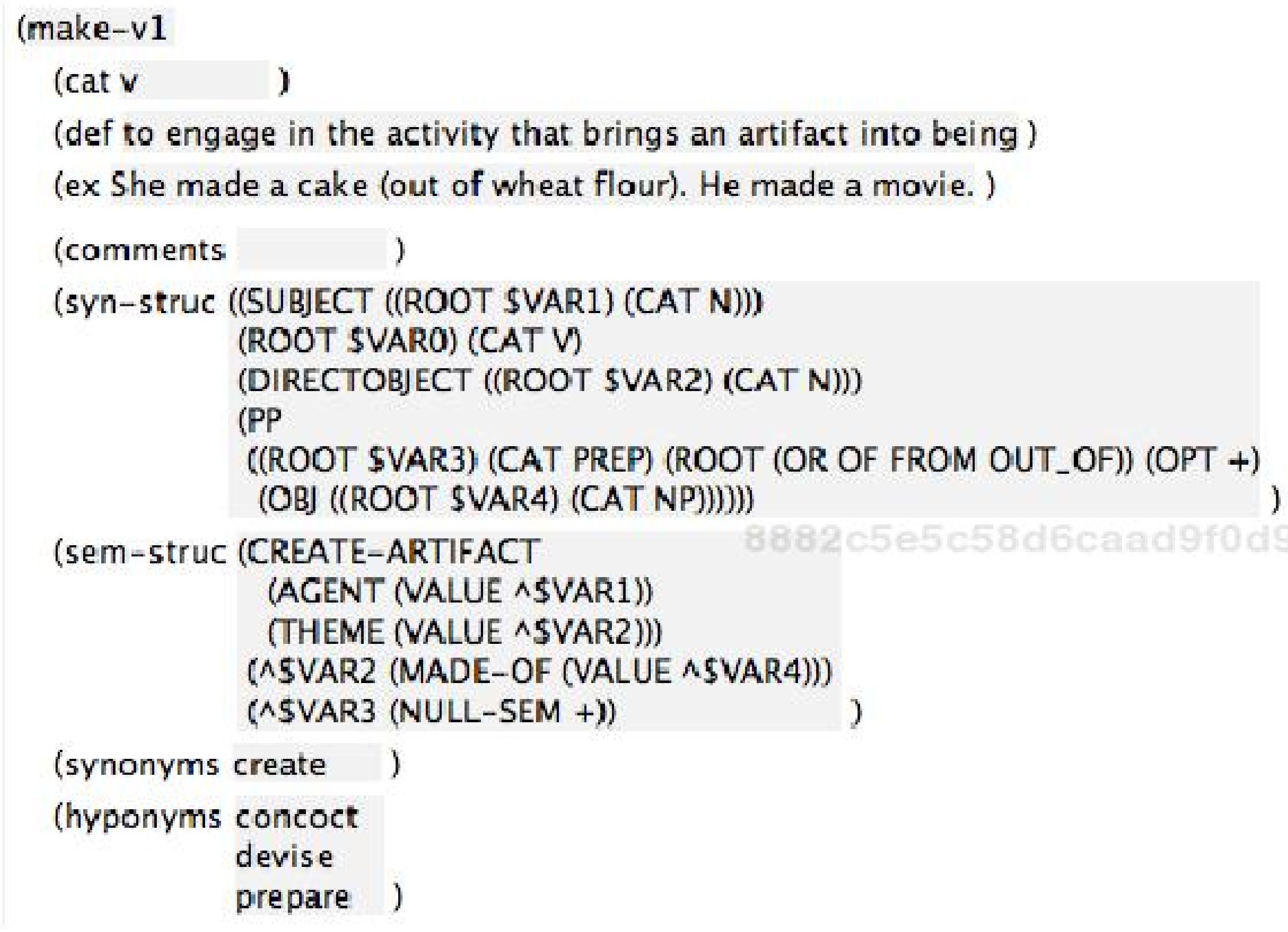
3.3. The Necessity of Constraining Senses

Perhaps the most important aspect of the OntoSem lexicon is that it attempts to constrain each lexical sense sufficiently to permit the analyzer to choose exactly one sense for any given input. Consider again the verb *make*, which currently has 40+ senses and counting. Most of its senses are phrasals, meaning that the syn-struc includes specific words that constrain the use of the sense. The following are just a few examples. The specific words that constrain the sense are in boldface, and the italicized glosses are human-oriented explanations of what each phrasal means. (Of course, in the sem-struc of the respective entries the meanings are encoded using ontological concepts with appropriate restrictions on the meanings of the case roles.)

- X makes **out** Y ~ *X can perceive Y*
- X makes **sure** (that) Y ~ *X confirms Y*
- X makes **away with** Y ~ *X steals Y*
- X makes **an effort/attempt to** Y ~ *X tries to do Y*
- X makes a **noise/sound** ~ *X emits a sound*
- X makes **fun of** Y ~ *X teases Y*

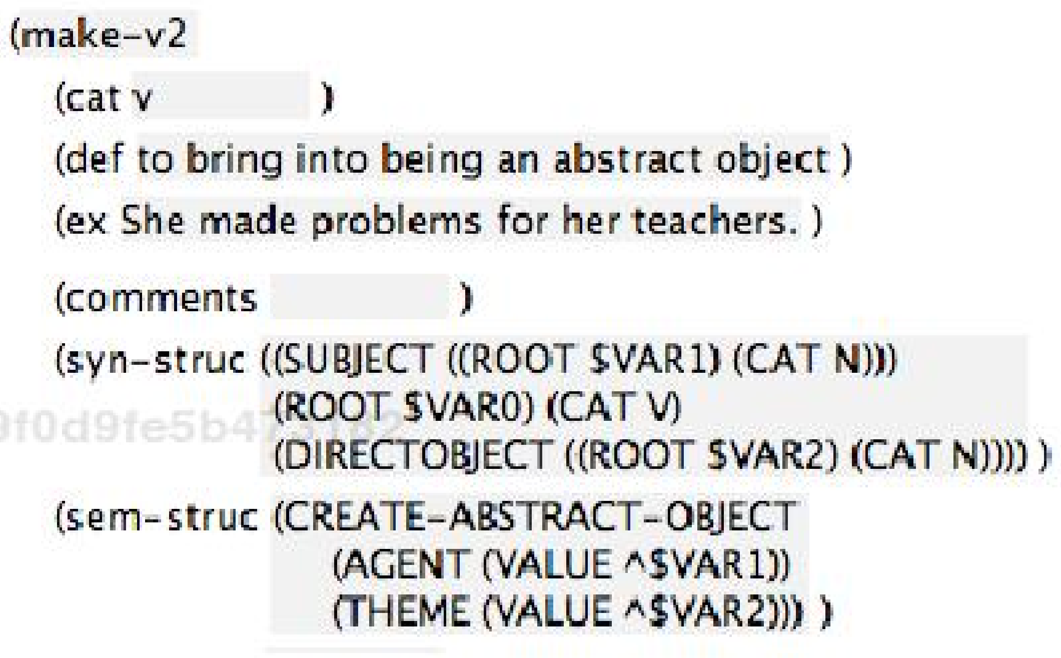
The senses of *make* that are not phrasals are also explicitly constrained to support disambiguation. Compare senses make-v1 and make-v2 shown in Figures 11 and 12.

Both are transitive senses but they take different kinds of direct objects: for make-v1 the direct object is a PHYSICAL-OBJECT, whereas for make-v2 it is an ABSTRACT-OBJECT.



```
(make-v1
  (cat v          )
  (def to engage in the activity that brings an artifact into being )
  (ex She made a cake (out of wheat flour). He made a movie. )
  (comments           )
  (syn-struc ((SUBJECT ((ROOT $VAR1) (CAT N)))
              (ROOT $VAR0) (CAT V)
              (DIRECTOBJECT ((ROOT $VAR2) (CAT N)))
              (PP
               ((ROOT $VAR3) (CAT PREP) (ROOT (OR OF FROM OUT_OF)) (OPT +)
                (OBJ ((ROOT $VAR4) (CAT NP))))))                               )
  (sem-struc (CREATE-ARTIFACT
                (AGENT (VALUE ^$VAR1))
                (THEME (VALUE ^$VAR2)))
              (^$VAR2 (MADE-OF (VALUE ^$VAR4)))
              (^$VAR3 (NULL-SEM +))                )
  (synonyms create     )
  (hyponyms concoct
            devise
            prepare    )
```

Figure 11. The sense of *make* that means creating an artifact.



```
(make-v2
  (cat v          )
  (def to bring into being an abstract object )
  (ex She made problems for her teachers. )
  (comments           )
  (syn-struc ((SUBJECT ((ROOT $VAR1) (CAT N)))
              (ROOT $VAR0) (CAT V)
              (DIRECTOBJECT ((ROOT $VAR2) (CAT N)))) )
  (sem-struc (CREATE-ABSTRACT-OBJECT
                (AGENT (VALUE ^$VAR1))
                (THEME (VALUE ^$VAR2))) )
```

Figure 12. The sense of *make* that means creating an abstract object.

One does not see these constraints overtly in the lexicon entry because they are in the ontological description of CREATE-ARTIFACT and CREATE-ABSTRACT-OBJECT, respectively. That is, CREATE-ARTIFACT is ontologically described as having the THEME ARTIFACT and CREATE-ABSTRACT-OBJECT is ontologically described as having the THEME ABSTRACT-OBJECT. As such, the analyzer "sees" these constraints just as it would see the constraints if they were overtly specified in the sem-strucs of the lexical entries. This points up an important aspect of OntoSem resources: they are designed to be used together, not in isolation. As such, the often difficult decision of whether to create a new concept or use an existing concept with lexical modifications is not really a big problem: either way is fine since the resources are leveraged in tandem.

4. Lexical Acquisition for L Using the OntoSem English Lexicon

The main efficiency enhancing benefit of using an existing OntoSem-style lexicon to acquire a new lexicon is the ability to reuse semantic descriptions of words – i.e., the sem-struc zones. After all, the hardest aspect of creating OntoSem lexicons, or any lexicon that includes semantics, is deciding how to describe the meaning of words and phrases. To create a sem-struc one must, at a minimum:

- be very familiar with the content and structure of the ontology to which words are mapped
- understand which meanings are ontological and which are extra-ontological, like modality and aspect
- understand what grain size of description is appropriate: it would be infeasible to record *everything* one knows about *every* word if one sought to create a lexicon and ontology in finite time
- understand how to combine the meanings of ontological concepts and extra-ontological descriptors to convey complex meanings
- be able to detect the need for procedural semantic routines and write them when needed

We believe that as long as the acquirer understands the meaning of a lexicon entry in the English lexicon, he can express the same meaning in L – be it as a word or a phrase. This belief is predicated on the hypothesis of practical effability, the tenet that every idea can be expressed in every language at a certain realistic level of granularity. Without going into a long discussion of the philosophical underpinnings of this hypothesis, let us just observe that a meaning that can be expressed using a single word in L1 might require a phrase in L2 or vice versa. So it is immaterial that some languages may have forty words for snow while others have one or two – in those other languages, the meaning of the 40 words can certainly be expressed using phrases or even clauses. Indeed, the famous Sapir-Whorf hypothesis that states that our language in a large part shapes our view of the world, is, at least in part, predicated on preferring single-word meaning realizations to phrasal ones. This distinction is less important for the practical automatic understanding of text than it is for philosophical and psychological deliberations.

Let us consider some of the many eventualities an acquirer might face in creating an L lexicon sense from an English one:

- The English sense and the L sense are both single-word entities that have the syn-struc and the same sem-struc. Acquisition of the L sense is trivial: the English head word is simply changed to the L head word.
- The English sense is a single word but the L sense is multiple words. The L acquirer will have to decide if (a) the multiple words are completely fixed (like *child care*), in which case they can be entered as a multi-word head word with an underscore in between (*child_care*) or (b) the words can have inflections, intervening words, etc., in which case they must be acquired as a complex syn-struc.
- The English sense contains multiple words but the L sense is a single word.

- The English sense and the L sense are both argument-taking entities (e.g., verbs) but they require different subcategorization frames, meaning that the inventory of syntactic components needs to be modified. Of course, every time the variables in the syntactic structure are changed, one must check to see if any of the linked variables in the semantic structure require modification.

The above inventory is just a sampling of the more common outcomes, with the full inventory including more fine-grained distinctions. We will now illustrate the process of creating lexicon of L from the lexicon of English, moving from simpler issues to more complex ones and using examples from a variety of languages.

Example 4.1 The first noun entry alphabetically in the English lexicon is, not surprisingly, *aardvark.*

```
(aardvark-n1
    (cat n)
    (syn-struc ((root $var0)(cat n)))
    (sem-struc (AARDVARK))).
```

If L has a word whose meaning corresponds directly to the English word *aardvark*, one can simply substitute it in the header of the entry: in a Russian lexicon, the headword would be аардварк. Of course, AARDVARK in the sem-struc denotes a concept, not a word in any language. In the OntoSem ontology, the ontological concept AARDVARK is at present minimally described as a kind of mammal. However, if or when more information is added to the ontology describing the aarkdvark – its habitat, its preferred food, its enemies, etc. – this information will have to be added only once, in the ontology, and then it will be accessible and usable in applications covering any language for which an ontological-semantic lexicon is available.[8]

Example 4.2 The noun *table* has two entries in the English lexicon, glossed as comments below:

```
(table-n1      ; a piece of furniture
    (cat n)
    (syn-struc ((root $var0)(cat n)))
    (sem-struc (TABLE)))

(table-n2      ; a compilation of information
    (cat n)
    (syn-struc ((root $var0)(cat n)))
    (sem-struc (CHART))).
```

The corresponding entries in a Hebrew lexicon (in transliteration) will be recorded under two different head words:

[8] Compare this “savings” in acquisition to the approach adopted for the SIMPLE project, a comparison that is detailed in [11].

```
(shulhan-n1
   (cat n)
   (syn-struc ((root $var0)(cat n)))
   (sem-struc (TABLE)))

(luah-n1
   (cat n)
   (syn-struc ((root $var0)(cat n)))
   (sem-struc (CHART))
   (synonyms “tavla”)
```

The acquirer will also notice that the Hebrew *tavla* is another way of expressing the meaning (the ontological concept) CHART. As a result, this word may be acquired in one of two ways – using its own entry or as a filler of the synonyms zone of the entry *luah*-n1, as shown above.

Example 4.3 The entry for *desk* is similarly simple:

```
(desk-n1
   (cat n)
   (syn-struc ((root $var0)(cat n)))
   (sem-struc (DESK)))
```

The corresponding entry in a Russian lexicon (given here in transliteration) will have to be headed by the word *stol* ‘table’ and, and the syn-struc will add the necessary modifier that constrains the sense: *pis’mennyj* ‘writing’. The modifier is, of course, attributed null semantics in the sem-struc because its semantics is folded into the ontological concept this sense is mapped to: DESK.

```
(stol-n1
    (cat n)
    (syn-struc ((root $var0) (cat n)
        ((mods (root $var1) (root pis’mennyj))
    (sem-struc
        (DESK)
        (null-sem ^$var1)))
```

Example 4.4 Lexical entries for verbs involve more work, mostly because their subcategorization properties must be described. The entry for *sleep* is as follows:

```
(sleep-v1 (cat v)
  (syn-struc ((subject ((root $var1) (cat n)))
              (root $var0) (cat v)))
  (sem-struc
      (SLEEP (EXPERIENCER (value ^$var1)))),
```

This entry states that *sleep* takes a subject headed by a noun; that its meaning is expressed by the ontological concept SLEEP; and that the EXPERIENCER case role should be filled by the subject of *sleep* when an instance of SLEEP is generated in the text meaning representation of the input sentence. The corresponding entry in French lexicon will be very similar, with *dormir* substituted for *sleep* in the header of the entry. This is because French, just like English, has intransitive verbs, and *dormir* happens to be intransitive, just like *sleep*.

Example 4.5 If the lexical units realizing the same meaning in L and English do not share their subcategorization properties, the acquirer will have to make necessary adjustments. Consider the English entry *live*-v2:

```
(live-v2
   (cat v)
   (syn-struc
       ((subject ((root $var1) (cat n)))
        (root $var0) (cat v)
        (pp ((root in) (root $var2) (cat prep) (obj ((root $var3 (cat n)))))))
   (sem-struc
       (INHABIT
           (AGENT (value ^$var1))
           (LOCATION (value ^$var3)))
       (^$var2 (null-sem +))),
```

This states the following:

- This sense of *live* takes a subject (a noun) and an obligatory adjunct which is a prepositional phrase introduced by *in.*
- The meaning of this sense is expressed by the ontological concept INHABIT whose AGENT and LOCATION case roles are filled by the meanings of the subject and the prepositional object of *live*-v2, respectively.
- The meaning of the preposition itself should be ignored (attributed null semantics) because it is taken care of by the meaning LOCATION in the sem-struc.

In French, this meaning is expressed by the word *habiter*, which is a regular transitive verb. As a result, when acquiring the lexicon for French, the above entry will be changed to:

```
(habiter-v2 (cat v)
   (syn-struc
       ((subject ((root $var1) (cat n)))
        (root $var0) (cat v)
        (directobject  ((root $var2) (cat n))))
   (sem-struc
       (INHABIT
           (AGENT (value ^$var1))
           (LOCATION (value ^$var2))))
```

Even though this slight change to the syn-struc must be entered, this is still much faster than creating the entry from scratch.

Example 4.6 A still more complex case is when the meaning of a word sense does not precisely correspond to any ontological concept. Consider the notion of "marrying" in English and Russian. In English, men can *marry* women and women can *marry* men, using the same verb that maps to the concept MARRY.

```
(marry-v1
   (syn-struc
       ((subject ((root $var1) (cat n)))
        (root $var0) (cat v)
        (directobject ((root $var2) (cat n))))
   (sem-struc
       (MARRY                    ; to take as spouse
           (AGENT (value ^$var1))
           (AGENT (value ^$var2)))))
```

However, MARRY does not fully express the meaning of any single word in Russian. Instead, there is a Russian word for the case of a man marrying a woman (where the man is the AGENT) and another word for the case of a woman marrying a man (where the woman is the AGENT). If the man is the AGENT, the verb is *zhenit'sja*, whereas if the woman is the AGENT a phrasal is used: *vyjit zamuzh za,* literally, "to leave married to". The gender information is in boldface in both entries for orientation.

```
(zhenit'sja-v1
   (syn-struc
      ((subject ((root $var1) (cat n)))
       (root $var0) (cat v)
       (pp ((root na) (root $var3) (cat prep) (obj ((root $var2) (cat n))))))
   (sem-struc
       (MARRY
           (AGENT (value ^$var1) (gender male))
           (AGENT (value ^$var2) (gender female)))
       (^$var3 (null-sem +))))
```

```
(vyjti-v3
   (syn-struc
     ((subject ((root $var1) (cat n)))
      (root $var0) (cat v)
      (directobject ((root $var4) (cat n) (root zamuzh)))
      (pp ((root za) (root $var3) (cat prep) (obj ((root $var2) (cat n))))))
   (sem-struc
       (MARRY
           (AGENT (value ^$var1) (gender female))
           (AGENT (value ^$var2) (gender male)))
       (^$var3 (null-sem +))
       (^$var4 (null-sem +))))
```

Note also that the syntactic structure of these entries is different from that of English *marry*. In the first of these two entries (*zhenit'sja*) the syn-struc describes an intransitive verb with a PP complement introduced by the preposition *na*. In the second entry, the syn-struc describes the phrasal *vyjti zamuzh za,* expressed as the third sense of the verb *vyjti* (whose other senses include "get out" and "be depleted"). This sense includes the direct object *zamuzh* and a prepositional phrase headed by the preposition *za*. To reiterate, in both of the above entries, the ontological concept MARRY is locally modified by constraining the semantics of its agents. Note that this modification is local to the lexicon entry: the concept MARRY, as specified in the ontology, is not affected outside of the above lexicon entries.

Example 4.7 Perhaps the greatest motivation for "reusing" an existing OntoSem lexicon is avoiding the necessity of inventing the semantic representation of complex words from scratch. Above we have seen rather straightforward entries for which available ontological concepts can be utilized. However, when describing entries like conjunctions and adverbs, the actual analysis required to create a sem-struc, and the procedural semantic routines needed to support it, can be non-trivial.

Let us consider the case of adverbs more closely. Not surprisingly, they tend not to be included in ontologies or semantic webs (or, for that matter, in corpus annotation). However, they are as important as any other lexemes to a full semantic interpretation and, as such, receive full treatment in OntoSem lexicons. Take the example of *overboard*, whose sem-struc says that the event that it modifies must be a MOTION-EVENT whose SOURCE is SURFACE-WATER-VEHICLE and whose DESTINATION is BODY-OF-WATER.

```
(overboard-adv1
  (cat adv)
  (anno
    (def "indicates that the source of the motion is a boat and the
           destination is a body of water")
    (ex "They threw the rotten food overboard. He jumped overboard."))
  (syn-struc
   ((root $var1) (cat v)
    (mods ((root $var0) (cat adv) (type post-verb-clause)))))
  (sem-struc
    (^$var1   (sem MOTION-EVENT)
     (SOURCE  SURFACE-WATER-VEHICLE)
     (DESTINATION   BODY-OF-WATER))))
```

While this description is quite transparent, it requires that the acquirer find three key concepts in the ontology, which takes more time than simply replacing the head word by an L equivalent (e.g., Russian *za bort*). More conceptually difficult is an adjective like *mitigating*:

```
(mitigating-adj1
  (cat adj)
  (anno
   (def "having the effect of moderating the intensity of some property")
   (ex  "mitigating circumstances (i.e., circumstances that lessen the intensity
                of some property of some object or event that is recoverable from
                the context)"))
        (syn-struc
          ((mods ((root $var0) (cat adj))
           (root $var1) (cat n))
        (sem-struc
          (^$var1
            (effect (> (value refsem1.intensity))))
          (refsem1 (property)))
        (meaning-procedure (seek-specification (value refsem1) reference-procedures)))
```

This semantic description says: the noun modified by *mitigating* has the effect of lessening the intensity of some property value of some object or event; *which* property of *which* object or event needs to be determined using procedural semantic reasoning, using the function called in the meaning-procedures zone. There are three important points here: first, coming up with a semantic interpretation for this word is not easy; second, once we do come up with one, it would be nice to use it for more than one language; and, third, despite the fact that the recorded semantic analysis of this entity does not take care of all aspects of its interpretation, like those that must be contextually determined by procedural semantics, it does as much as a lexical description can be expected to do.

It is not only adjectives and adverbs that can present a choice space that takes time to sort through. Here are a few examples of select senses of words from other parts of speech, written in what we hope is an obvious shorthand:

fee (n.)
    MONEY (THEME-OF: CHARGE)
violist (n.)
    MUSICIAN (AGENT-OF (PLAY-MUSICAL-INSTRUMENT (THEME: VIOLA)))
file (n.)
    SET (MEMBER-TYPE: DOCUMENT)
aflame (adj.)
    the modified is the THEME of BURN
exempt (from sth.) (adj.)
    the modified is the BENEFICIARY of an EXEMPT event whose THEME is the object of the *from*-PP
managing (adj.)
    the modified is the AGENT of a MANAGEMENT-ACTIVITY (so 'managing editor' is an EDITOR (AGENT-OF MANAGEMENT-ACTIVITY))

In sum, any time that a semantic description requires more work than direct mapping to an ontological concept, there are gains to be had by interpreting that description as a language-neutral representation of meaning that can then be associated with the corresponding head words in different languages.

Example 4.8 What happens if the English lexicon does not contain a word or phrase that must be acquired for the lexicon of L? This case is identical to the task of acquiring the English lexicon in the first place. Consider, for example, the English verb *taxi*. It is applicable to aircraft and denotes the action of its moving on a surface. The ontology contains the concepts AIRCRAFT and MOVE-ON-SURFACE. When faced with the task of acquiring the entry for *taxi*-v1 for the English lexicon, the acquirer faces the choice of either putting the construct (MOVE-ON-SURFACE (theme AIRCRAFT)) in the sem-struc zone of the lexicon entry or opting for creating a new ontological concept, say, TAXI-EVENT, in which the same information will be listed. In the latter case, the sem-struc zone of the entry for *taxi*-v1 will be a simple reference to the new ontological concept TAXI-EVENT.

The choice of strategy in such cases may be beyond the purview of this paper, as it will depend on a particular application. The general rule of thumb is to try to keep the ontology as small as possible and at the same time make sure that it can help to describe the meaning of as many words and phrases in L as possible. This is a well-known desideratum in formal descriptions, cf. [4] for a succinct early explanation.

If, by contrast, available ontological knowledge is not sufficient for rendering the meaning of the new word, then the ontology itself must be augmented before a lexicon entry can be created. This, of course, makes the task of writing lexicon entries much more complex.

5. Final Thoughts

Acquiring resources for low- and mid-density languages is difficult since there tends to be little manpower available to compile them. For that reason, reusing resources that already exist should always be considered an option worth exploring. Of course, the temptation in working on low- and mid-density might be to avoid depth of analysis, instead relying only on large corpora and stochastic methods for text processing. For this reason, one must answer the question, What is all this semantic information good for? It is good for any application that can benefit from disambiguation, since the single most compelling reason to engage in knowledge-rich natural language processing is to permit applications to work on disambiguated knowledge, rather than highly ambiguous text strings. To our thinking, this includes *all* NLP applications, though we acknowledge this opinion as not universally held. Two other obvious beneficiaries of semantically analyzed text are automated reasoners and machine learners, both of which can benefit from more semantic features in the feature space. Apart from these practical uses of OntoSem resources, we believe that there are significant theoretical reasons for pursuing rigorous broad-scale and deep lexical semantics for NLP. Indeed, the stated goal of linguistics is to explain the connection of texts with their meanings. The broad goal of computational linguistics should then be developing computational means of establishing correspondences between texts and their meaning. If we are serious about reaching this goal, the development of semantic lexicons for the various languages and of the semantic metalanguage of description should be viewed as the core tasks of the field.

References

[1] Beale, Stephen, Sergei Nirenburg and Marjorie McShane. 2003. Just-in-time grammar. *Proceedings 2003 International Multiconference in Computer Science and Computer Engineering*, Las Vegas, Nevada.

[2] Beale, Stephen, Benoit Lavoie, Marjorie McShane, Sergei Nirenburg and Tanya Korelsky. 2004. Question answering using Ontological Semantics. *Proceedings of ACL-2004 Workshop on Text Meaning and Interpretation*, Barcelona.

[3] Cruse, D.A. 1986. *Lexical Semantics*. Cambridge University Press.

[4] Hayes, P.J., 1979. The naive physics manifesto. In: D. Michie (ed.), *Expert Systems in the Microelectronic Age*. Edinburgh, Scotland. Edinburgh University Press.

[5] Inkpen, Diana and Graeme Hirst. 2006. Building and using a lexical knowledge-base of near-synonym differences. 2006. *Computational Linguistics* 32(2): 223-262.

[6] McShane, M., S. Nirenburg, J. Cowie and R. Zacharski. 2002. Embedding knowledge elicitation and MT systems within a single architecture. *Machine Translation* 17(4): 271-305.

[7] McShane, Marjorie. 2003. Applying tools and techniques of natural language processing to the creation of resources for less commonly taught languages. *IALLT Journal of Language Learning Technologies* 35 (1): 25-46.

[8] McShane, Marjorie and Sergei Nirenburg. 2003. Blasting open a choice space: learning inflectional morphology for NLP. *Computational Intelligence* 19(2): 111-135.

[9] McShane, Marjorie and Sergei Nirenburg. 2003. Parameterizing and eliciting text elements across languages. *Machine Translation* 18(2): 129-165.

[10] McShane, Marjorie, Stephen Beale and Sergei Nirenburg. 2004. Some meaning procedures of Ontological Semantics. *Proceedings of LREC*-2004.

[11] McShane, Marjorie, Sergei Nirenburg and Stephen Beale. 2004. OntoSem and SIMPLE: Two multi-lingual world views. *Proceedings of ACL-2004 Workshop on Text Meaning and Interpretation*, Barcelona.

[12] McShane, Marjorie, Sergei Nirenburg and Stephen Beale. 2005. An NLP lexicon as a largely language independent resource. *Machine Translation* 19(2): 139-173.

[13] McShane, Marjorie, Sergei Nirenburg and Stephen Beale. 2005. Semantics-based resolution of fragments and underspecified structures. *Traitement Automatique des Langues* 46(1): 163-184.

[14] McShane, Marjorie and Ron Zacharski. 2005c. User-extensible on-line lexicons for language learning. Working Paper #05-05, Institute for Language and Information Technologies, University of Maryland Baltimore County.

[15] McShane, Marjorie, Sergei Nirenburg and Stephen Beale. 2005. The description and processing of multiword expressions in OntoSem. Working Paper #07-05, Institute for Language and Information Technologies, University of Maryland Baltimore County.

[16] McShane, Marjorie, Sergei Nirenburg, Stephen Beale and Thomas O Hara. 2005. Semantically rich human-aided machine annotation. *Proceedings the Workshop on Frontiers in Corpus Annotation II: Pie in the Sky*, ACL-05, Ann Arbor, June 2005, pp. 68-75.

[17] Nirenburg, Sergei, Marjorie McShane, Margalit Zabludowski, Stephen Beale, Craig Pfeifer. 2005. Ontological Semantic text processing in the biomedical domain. Working Paper #03-05, Institute for Language and Information Technologies, University of Maryland Baltimore County.

[18] Nirenburg, Sergei and Tim Oates. 2007. Learning by reading by learning to read. *Proceedings of ICSC-07*. Irvine, CA. September.

[19] Nirenburg, Sergei and Victor Raskin. 2004. *Ontological Semantics*. MIT Press.

[20] Probst, Katharina, Lori Levin, Erik Peterson, Alon Lavie, Jaime Carbonell. 2002. MT for resource-poor languages using elicitation-based learning of syntactic transfer rules. *Machine Translation* 17/4: 245-270.