# Increasing Understanding: Interpreting Events of Change

**Sergei Nirenburg, Marjorie McShane and Stephen Beale**
Department of Computer Science and Electrical Engineering
University of Maryland, Baltimore County
Baltimore, Maryland, 21250 USA
{sergei,marge,sbeale}@umbc.edu

## Abstract

This paper discusses the treatment of events of change within the Ontological Semantics text processing environment using the example of the English lexeme *increase*. It suggests that explanatory, context-specific semantic representations of the meaning of such verbs are both achievable and necessary to support automated reasoners. It describes the content of the lexical and ontological static resources that are leveraged by procedural semantic routines to automatically produce deep-semantic text-meaning representations of input text.

## 1 Introduction

A semantically insightful way to describe events of change is in terms of their preconditions and effects. If a car accelerates, that means that the value of the "speed" attribute as applied to the car's motion is higher in the effect than in the precondition of the acceleration event; and if the importance of a certain political theory grows, that means that the value of the modality "saliency" scoping over that political theory is higher in the effect than in the precondition of the change event. Such explanatory interpretations of change events provide more information for automated reasoning than would a simple pointer to an ontological concept like "accelerate" or "grow".

In this paper we describe the treatment of the lexical and compositional semantics of change events within the Ontological Semantics theory of NLP (Nirenburg and Raskin 2004). Using the example of *increase*, we discuss how lexical and ontological static resources combine with procedural semantic routines to create *text-meaning representations* (TMRs) of sufficient depth to support sophisticated reasoning applications.

The material also provides a good example of the continuous interplay and interdependence between lexical-semantic descriptions, on the one hand, and ontological descriptions, on the other. We demonstrate that explanatory, NLP-oriented ontologies do not need to have a concept for every word sense or even for every synonym set. Many lexical entries can be well described using relatively few ontological concepts – provided the ontological metalanguage includes a sufficient inventory of properties (relations and attributes) to characterize ontological concepts well beyond their position in the subsumption hierarchy.

## 2 A Very Brief Snapshot of OntoSem

OntoSem is a text processing environment that takes as input unrestricted raw text and carries out its tokenization, morphological analysis, syntactic analysis, and semantic analysis to yield formal text-meaning representations (TMRs). Text analysis relies on:

- The OntoSem language-independent **ontology**, which currently contains over 8,000 concepts (amounting to over 140,000 RDF resource-property-value triples). The metavocabulary of the ontology is comprised of about 350 basic properties (relations and attributes). The number of concepts is intentionally restricted, such that a given ontological concept is typically used, with necessary local modifications, in the lexical descriptions of many words and phrases, not only close synonyms.
- An OntoSem **lexicon** for each language processed, whose entries contain, among other information, syntactic and semantic zones (linked through special variables) as well as procedural-semantic attachments that we call *meaning procedures*. The semantic zone most frequently invokes ontological concepts, either directly or with modifications, but can also describe word meaning extra-ontologically, for example, in terms of parameterized values of modality, aspect, time or combinations thereof. The English lexicon currently contains about 35,000 senses, not counting word forms productively analyzed on the fly using lexical rules (taking those into account significantly increases the number of "understood senses" in our lexicon).
- A **fact repository**, which is a persistent knowledge base of what logicians call assertions. It contains real-world facts represented as numbered "remembered instances" of ontological concepts: e.g., SPEECH-ACT-3186 is the 3186th instantiation of the concept SPEECH-ACT in the world model constructed during text processing.
- The OntoSem text **analyzers**, covering everything from tokenization to extended TMR creation (extended TMRs differ from basic TMRs in that they reflect the results of procedural semantic reasoning).
- The TMR language, which is the **metalanguage** for representing text meaning, compatible with the metalanguage of the ontology and the fact repository.

TMRs represent, to our knowledge, the most semantically rich, automatically generated expressions of text meaning produced by any extant system.[1] They can be viewed as a super-semantic alternative to traditional corpus tagging schemes (for details, see McShane et al. 2005).

## 3 The Examples to be Considered

In this section we describe four types of semantic contexts requiring different interpretations of *increase*: *increase* as applied to **scalar attributes** (3.1), **modalities** (3.2), **count nouns** (3.3) and **non-count nouns** (3.4). For brevity's sake, for each semantic subtype we present just one lexical sense: for example, we discuss the intransitive *The weight of the elephant increased* but not the corresponding transitive *They increased the weight of the elephant* or the nominal *the increase in the weight of the elephant*, which are treated similarly. All in all, we currently have sixteen senses of *increase*. Most of these have some number of synonyms, and most include some number of optional syntactic components.[2] Factoring in all the relevant lexemes and syntactic configurations, over 500 lexico-syntactic input combinations are covered by this microtheory of lexemes indicating increase (and a similar number are covered by the corresponding treatment of decrease).

### 3.1 *Increase* with Scalar Attributes

Among the properties defined in the PROPERTY branch of the OntoSem ontology are SCALAR-ATTRIBUTES, a random sample of which (selected from many dozens) includes COMPLEXITY, COST, INTENSITY, USEFULNESS, RAPIDITY, AGE and ABSTRACTNESS. While SCALAR-ATTRIBUTES can take various types of OBJECTs or EVENTs as their domain, they all take a numerical value, or range of values, as their range. That value can either be a real value or a point on an abstract {0,1} scale. For example, *expensive* is lexically described as .8 on the scale of COST. (Of course, one could quibble

[1] For details of this approach to text processing see Nirenburg and Raskin 2004. The ontology itself, a brief ontology tutorial, an extensive lexicon tutorial and research papers can be viewed at http://ilit.umbc.edu.

[2] Actually, some of the 'synonyms' are not quite synonyms; instead, they imply a slow or fast rate of change. For example, *skyrocket* is increase with a high rate of change and *creep up* is increase with a low rate of change. Such entries are actually recorded separately (not as synonyms of *increase*) and the rate of change is included in their semantic descriptions.

about whether *expensive* should be rendered as .7, .8, between .7 and .9, etc.; but lingering over such questions does not support practical solutions.) Automatic reasoning systems can interpret these abstract values relative to the ontologically listed "normal" range of property values listed for a concept (e.g., an expensive car is a car whose cost is around .8 of the maximum cost listed for cars). During lexicon acquisition we attempt to be consistent in our interpretation of points on the scale: just as *expensive* is .8 on the scale of COST, *tall* is .8 on the scale of HEIGHT and *heavy* is .8 on the scale of WEIGHT. Likewise, *very* consistently shifts the given value by .1 toward the extreme of the scale, and *extremely* shifts it by .2, so *very heavy* will be .9 on the scale of WEIGHT and *extremely heavy* will be 1 (see McShane et al. 2004b for further discussion of modifications to scalars).

In terms of change events, however, it is the relative values of scalar attributes that are most important: if the size of something increases, the value of its range is higher in the effect of the change event than in the precondition. Let us consider how one of the verbal senses for *increase* (increase-v1, below) supports such an interpretation.

For orientation, lexicon entries in OntoSem are written using an extended variety of LFG in LISP-compatible format. Elements of the syntactic structure (*syn-struc*) and semantic structure *(sem-struc)* are linked using variables, and the same variables are referred to in the procedural attachments (*meaning-procedure*) zone. The caret (^) preceding variable names in the sem-struc indicates "the meaning of (the variable)". *Refsem* is a reserved term used primarily for reification. Ontological concepts are written in SMALL CAPS.

```
(increase-v1
 (def "of scalar attributes: the value of the range is
         larger in the effect than in the precondition")
  (ex "The weight of the elephant increased (by 500
         lbs.) (from 1000 lbs.) (to 1500 lbs.)")
  (syn-struc
    ((subject ((root $var1) (cat n)))
      (root $var0) (cat v)
      (pp ((root $var2) (cat prep) (root by) (opt +)
           (obj ((root $var3) (cat n)))))
      (pp ((root $var4) (cat prep) (root from) (opt +)
            (obj ((root $var5) (cat n)))))
      (pp ((root $var6) (cat prep) (root to) (opt +)
           (obj ((root $var7) (cat n))))))
 (sem-struc
  (CHANGE-EVENT
    (PRECONDITION (value refsem1))
    (EFFECT (value refsem2))
    (CHANGE-IN-VALUE ((value ^$var3) add)))
  (refsem1
    (SCALAR-ATTRIBUTE
      (RANGE (value ^$var5))))
  (refsem2
    (SCALAR-ATTRIBUTE
       (RANGE (value ^$var7))))
  (< (value refsem1.RANGE) (value refsem2.range))
  (^$var2 (null-sem +))
  (^$var4 (null-sem +))
  (^$var6 (null-sem +)))
 (meaning-procedure
  (seek-specification (value refsem1) (value ^$var1))
  (seek-specification (value refsem2) (value ^$var1))
  (fill-in-missing-values-for-increase1)))
```

The basic ontological mapping is to CHANGE-EVENT, which is quite high on the tree of inheritance: its parents are PHYSICAL-EVENT and MENTAL-EVENT, which themselves are children of EVENT. As a high-level ontological concept, CHANGE-EVENT does not have very constrained property values. The main work of interpreting CHANGE-EVENTs, therefore, lies in the further specification of their preconditions and effects.

Let us start by considering the case when none of the optional PPs is overt in the context, as in the sentence *The weight of the elephant increased.* The meaning is that the value of the range of WEIGHT is greater in the effect than in the precondition of *increase*. This information is captured in the sem-struc statement:

```
(< (value refsem1.range) (value refsem2.range))
```

Refsem1 and refsem2 refer to some SCALAR-ATTRIBUTE whose specification requires knowing the meaning of the subject of the clause. The relevant procedural semantic routine is *seek-specification*, which is called in the *meaning-procedure* zone of the entry. It says 'Seek the meaning of refsem1/refsem2 using the meaning of $var1 as an input parameter.' Assuming that the subject of the sentence is *the weight of the elephant*, the analyzer will select the following sense of the word *weight*:[3]

[3] One bit of lexical complexity is worth mentioning. Some words that map to SCALAR-ATTRIBUTEs have two different interpretations: one in which the range of the attribute is un-

```
(weight-n1
  (def "indicates physical heaviness")
  (ex "The weight of the child is 9 lbs.")
  (syn-struc
    ((root $var0) (cat n)
     (pp ((root $var1) (cat prep) (root of) (opt +)
          (obj ((root $var2) (cat n)))))))
  (sem-struc
    (WEIGHT
      (DOMAIN (value ^$var2)))
    (^$var1 (null-sem +))))
```

This sense says that the word *weight* instantiates the ontological concept WEIGHT whose DOMAIN is the meaning of $var2, which is in this case maps to ELEPHANT. (The descriptor (null-sem +) indicates that no compositional meaning should be attributed to the preposition since its meaning – or, more precisely, function – is taken care of already in the semantic description.) The result of the *seek-specification* meaning procedure, therefore, is to replace, in the TMR, the SCALAR-ATTRIBUTE instantiated by *increase* with WEIGHT (DOMAIN: ELEPHANT).

The TMR resulting from the input sentence *The weight of the elephant increased* will be:

```
CHANGE-EVENT-1
    textpointer          increased
    PRECONDITION         WEIGHT-1
    EFFECT               WEIGHT-2
    TIME                 (< FIND-ANCHOR-TIME)[4]
    (< WEIGHT-1.RANGE  WEIGHT-2.RANGE)
WEIGHT-1
    textpointer          weight
    DOMAIN               ELEPHANT-1
    PRECONDITION-OF      CHANGE-EVENT-1
WEIGHT-2
    textpointer          weight
    DOMAIN               ELEPHANT-1
    EFFECT-OF            CHANGE-EVENT-1
ELEPHANT-1
    textpointer          elephant
    DOMAIN-OF            WEIGHT-1  WEIGHT-2
```

This text-meaning representation was generated from input in which none of the optional PPs for *increase* were overt: that is, there was no indication of the elephant's starting weight, ending weight or change in weight. However, as shown in the "example" field for *increase-v1*, any combinations of PPs can be overt: *The weight of the elephant increased (by 500 lbs.) (from 1000 lbs.) (to 1500 lbs.).* When such optional information is present, it can be used in two ways. First, it directly fills in slots in the *sem-struc* (e.g., the value of a *to*-PP fills in the RANGE of refsem2), which is then rendered as a more information-rich TMR. Second, if at least two of the three values are provided or can be recovered from the context, the full template of values (the value before the increase, after the increase, and the amount of increase) can be filled in. This is the ideal situation for one of the main applications of OntoSem: populating a Fact Repository with real-world facts, both explicit in the text and inferred with a high degree of confidence.

Consider the TMR for the input *The weight of the elephant increased by 500 lbs. to 1500 lbs.*

```
CHANGE-EVENT-2
    textpointer          increased
    PRECONDITION         WEIGHT-3
    EFFECT               WEIGHT-4
    CHANGE-IN-VALUE      + 500 (MEASURED-IN POUND)
    TIME                 (< FIND-ANCHOR-TIME)[5]
    (< WEIGHT-3.RANGE  WEIGHT-4.RANGE)
WEIGHT-3
    textpointer          weight
    DOMAIN               ELEPHANT-2
    PRECONDITION-OF      CHANGE-EVENT-2
WEIGHT-4
    textpointer          weight
    DOMAIN               ELEPHANT-2
    RANGE                1500 (MEASURED-IN POUND)
```

specified, and the other in which the range is understood as having a high value. For example, *height* can mean either "distance from the base of something to the top" or "the condition or attribute of being relatively or sufficiently high or tall". Under the first interpretation, the range of HEIGHT is unspecified, whereas under the second interpretation it is, say, (> .7). If one says *I was surprised by the height of that steeple*, the interpretation is (HEIGHT (> .7)), whereas if one says *The height of the tree increased,* the initial and ending HEIGHTs could be anything. We are working on developing heuristics for such disambiguation in the context of our larger work on semantic disambiguation (see, e.g., Beale et al. 2003 for an overview of disambiguation in OntoSem).

[4] This is a call to a meaning procedure that seeks the anchor time in a dateline or other text source. Since the anchor time cannot be resolved in our short context, the reference to the meaning procedure (which means "prior to the anchor time" and reflects the past tense of the verb) remains in the TMR.

[5] This is a call to another meaning procedure that seeks the anchor time in a dateline or other text source. Since the anchor time cannot be resolved in our short context, the reference to the meaning procedure (which means "prior to the anchor time" and reflects the past tense of the verb) remains in the TMR.

```
    EFFECT-OF          CHANGE-EVENT-2
ELEPHANT-2
    textpointer        elephant
    DOMAIN-OF          WEIGHT-3   WEIGHT-4
```

Since we have two values (end value and amount of change) we can calculate the starting value (the RANGE of WEIGHT-3). The meaning procedure called to do this is descriptively named *fill-in-missing-values-for-increase1*. It is actually a multi-part routine (whose full call is not presented here) that does several things: if two values are present, it calculates the third; if only the amount of change is present, it seeks out the initial or final value from the preceding context (more specifically, the text-meaning representations of the preceding context) and, if successful, uses the now-known two values to calculate the third one. Notice that the *fill-in-missing-values-for-increase1* has a different status than *seek-specification*: whereas *seek-specification* is essential to interpreting the actual textual input, *fill-in-missing-values-for-increase1* attempts to go beyond the input in order to arrive at a fuller representation of all available meaning for the Fact Repository.

Although we have been talking about the example of weight throughout, we must emphasize that this sense of *increase* covers input in which the subject refers to any scalar attribute.

One final note is worth mentioning before we leave the topic of scalar attributes. *Increase* is a particularly complex example since one cannot record beforehand which scalar attribute is in question—that must be compositionally determined based on the meaning of the subject. However, for many other lexemes the relevant scalar attribute can be lexically encoded: e.g., *to accelerate* refers to increasing the value of VELOCITY, *to smooth out* refers to increasing the value of SMOOTHNESS, and *to dry* refers to decreasing the value of WETNESS. The description of such lexemes and their representation in TMR are very similar to the case of *increase* except that the *seek-specification* procedural routine is not required.

## 3.2 *Increase* with Modalities

Modalities express an attitude on the part of the speaker toward the content of a proposition. [6] Within OntoSem, they are treated as extra-ontological aspects of meaning. The modalities currently used in OntoSem are: *epistemic, belief, obligative, permissive, potential, evaluative, intentional, epiteuctic, effort, volitive,* and *saliency*. The scale for the values of each modality goes from 0 to 1, with any decimal value or range in between being valid. Examples of lexical items whose meanings are conveyed by values of modality are:

| | |
|---|---|
| *must* | obligative: 1 |
| *might* | epistemic: .4 < > .6 |
| *important* | saliency: .8 |
| *loathe* | evaluative: 0 |

Modalities are defined for *type, scope, value* and *attributed-to*, with the latter defaulting to the speaker if not overtly specified. For example, one sense of *importance* is described as follows:[7]

```
(importance-n1
 (cat n)
  (def "some unspecified value of the modality
        'saliency'; this sense is used mainly in contexts
        of comparison (since the sense 'highly impor-
        tant' is the default otherwise)")
  (ex "the importance of taking vitamins")
  (syn-struc
    ((root $var0) (cat n)
     (pp ((root $var1) (cat prep) (root of) (opt +)
          (obj ((root $var2) (cat n))))))
  (sem-struc
    (modality
      (type saliency)
      (scope (value ^$var2)))
    (^$var1 (null-sem +))))
```

Values for modality are compared in inputs like the following, extracted from a corpus: *The importance of this issue <The need for help, The emphasis on health care, Efforts by insurers> increased.* The lexical sense of *increase* that covers such inputs is *increase-v2*:

```
(increase-v2
  (def "used with modalities: the value increases")
  (ex "The importance of diplomacy has increased")
  (syn-struc
    ((subject ((root $var1) (cat n)))
```

[6] There is often semantic ellipsis of part of the proposition: e.g., if one says that 'honor is important', the proposition is a person's having the attribute of being honorable. For further discussion of semantic ellipsis, see McShane et al. 2004a.

[7] Another sense of *importance* is 'high level of importance', in which the value for saliency would be (> .7). Cf. footnote 3.

```
    (root $var0) (cat v)))
  (sem-struc
   (CHANGE-EVENT
     (PRECONDITION (value refsem1))
     (EFFECT (value refsem2)))
   (refsem1 (modality))
   (refsem2 (modality))
   (< (value refsem1.value) (value refsem2.value)))
  (meaning-procedure
   (seek-specification (value refsem1) (value ^$var1))
   (seek-specification (value refsem2) (value ^$var1))))
```

This sense assumes, as appears to be justified from a small corpus study, that PP adjuncts indicating exact values will not typically be used with modals (modifiers like *a lot* and *significantly* will be treated compositionally and need not be referred to in the entry for *increase*). Apart from its relative simplicity due to a lack of optional PPs, *increase-v2* is actually quite similar to *increase-v1*, the differences reducing to those listed in Table 1.

| | *increase-v1* | *increase-v2* |
|---|---|---|
| Sem-struc element requiring procedural-semantic specification | SCALAR-ATTRIBUTE | type of modality |
| What changes from the precondition to the effect of the CHANGE-EVENT | the range of the SCALAR-ATTRIBUTE | the value of the modality |

Table 1. Comparing *increase-v1* and *increase-v5*

The analysis of an input like *The importance of diplomacy increased* will produce the following TMR:

```
CHANGE-EVENT- 3
   textpointer         increased
   PRECONDITION        MODALITY-1
   EFFECT              MODALITY-2
   TIME                (< FIND-ANCHOR-TIME)
   (< MODALITY-1.VALUE  MODALITY-2.VALUE)
MODALITY-1
   textpointer         importance
   SCOPE               DIPLOMATIC-EVENT-1
   TYPE                SALIENCY
   PRECONDITION-OF     CHANGE-EVENT-3
MODALITY-2
   textpointer         importance
   SCOPE               DIPLOMATIC-EVENT-1
   TYPE                SALIENCY
   EFFECT-OF           CHANGE-EVENT-3
DIPLOMATIC-EVENT-1
   textpointer         diplomacy
   SCOPE-OF            MODALITY-1  MODALITY-2
```

The semantic analyzer can disambiguate between *increase-v1* and *increase-v2* because each sense imposes semantic constraints on ^$var1: in *increase-v1* it must be a SCALAR-ATTRIBUTE and in *increase-v2* it must be a type of modality.

### 3.3 *Increase* with Count Nouns

Ellipsis in language is widespread, which poses well-known problems for NLP. However, some cases of ellipsis are predictable and can be planned for using a combination of static resources and programs that use them. Consider the following corpus excerpts:

- After that **the mosquitoes increased** and there was a considerable amount of fever in October and November.
- Following cessation of wolf control in 1960 **wolves increased** and attained densities of approximately 16 wolves/1000 km$^2$ by 1970.
- As Figure 2 shows, **total accidents increased** modestly from 1993 through 1997.

The implications of the above three sentences are, respectively, that the number of mosquitoes, the number of wolves and the number of accidents increased, even though there is no explicit reference to *number* in any of the contexts. The pivotal clue that underpins the automated interpretation of such contexts is the recognition that the subject NP is a count noun – regardless of whether it refers to an ontological OBJECT (MOSQUITO, WOLF) or EVENT (ACCIDENT).

In OntoSem, “count” and “non-count” nouns are not defined lexically, they are defined ontologically. Count nouns are mapped to concepts which are in the domain of CARDINALITY, whereas non-count nouns are mapped to concepts that are in the domain of AMOUNT. Roughly speaking, MATERIAL and INTANGIBLE-OBJECT are defined for AMOUNT, whereas all other OBJECTs and EVENTs are defined for cardinality. The semantic analyzer will select *increase-v3* (below) only in those contexts in which the subject maps to an entity whose ontological mapping is in the domain of CARDINALITY.

```
(increase-v3
 (def "used with count nouns")
 (ex "The mosquitoes increased")
 (syn-struc
   ((subject ((root $var1) (cat n)))
    (root $var0) (cat v)))
 (sem-struc
   (CHANGE-EVENT
     (PRECONDITION (value refsem1))
     (EFFECT (value refsem2)))
   (refsem1
     (set (element-type (value ^$var1))))
   (refsem2
     (set (element-type (value ^$var1))))
   (<  (value refsem1.CARDINALITY)
       (value refsem2.CARDINALITY))))
```

Note that no meaning procedure is required to seek the specification of the property in question: the property CARDINALITY is asserted to be the one in question for all inputs whose ^$var1 refers to a "count" OBJECT or EVENT.

The TMR produced for the input *The mosquitoes increased* is:

```
CHANGE-EVENT- 4
   textpointer        increased
   PRECONDITION       SET-1
   EFFECT             SET-2
   TIME               (< FIND-ANCHOR-TIME)
   (< SET-1.CARDINALITY SET-2.CARDINALITY)
SET-1
   ELEMENT-TYPE       MOSQUITO-1
   PRECONDITION-OF    CHANGE-EVENT-4
SET-2
   ELEMENT-TYPE       MOSQUITO-1
   EFFECT-OF          CHANGE-EVENT-4
MOSQUITO-1
   textpointer        mosquitoes
   CARDINALITY        (> 1) ; because plural
   ELEMENT-OF         SET-1 SET-2
```

### 3.4 *Increase* with Non-Count Nouns in N-N Compounds

A similar type of semantic ellipsis can occur with non-count nouns: an *amount* of something can be referred to without the word 'amount', as in *Potable water increased.* For the sake of variety, let us consider a lexical sense of *increase* that treats implied amounts but in a slightly more complex syntactic structure – one in which the subject is a noun-noun compound. The configuration in question is illustrated by examples like the following: *Wine consumption <Calcium intake, Cocaine use> increased.* Here, the first noun in each N-N compound has two notable properties: it is the THEME of the EVENT referred to by the second noun of the compound, and its AMOUNT is understood to increase. The lexical sense that covers such contexts is *increase-v6*.

```
(increase-v6
 (def "intransitive; the subject is a N-N compound
       in which the first N is a non-count noun")
 (ex "Wine consumption increased")
 (syn-struc
   ((subject
      (n ((root $var1) (cat n)))
      (n ((root $var2) (cat n))))
    (root $var0) (cat v)))
 (sem-struc
   (CHANGE-EVENT
     (PRECONDITION (value refsem1))
     (EFFECT (value refsem2)))
   (^$var2 (sem EVENT)
     (THEME (value ^$VAR1)))
   (refsem1
     (AMOUNT
        (DOMAIN (value ^$var1))))
   (refsem2
     (AMOUNT
        (DOMAIN (value ^$var1))))
   (<  (value refsem1.RANGE)
       (value  refsem2.RANGE))))
```

The *syn-struc* of this entry explicitly requires a N-N compound its subject. The *sem-struc* says that there is a CHANGE-EVENT by which the AMOUNT of ^$var1 (WINE) in the PRECONDITION is less than in the EFFECT. The *sem-struc* also asserts that that ^$var1 (WINE) is the THEME of ^$var2 (DRINK), and that ^$var2 itself must be an EVENT, which it is in our example. This latter semantic constraint supports disambiguation between *Wine consumption increased* (increase-v6) and *Wine vinegar increased.* The latter would be covered by increase-v4, a sense—not shown here—that expects the subject to indicate a non-count entity.

*Increase-v6* as applied to the input *Wine consumption increased* will yield the following TMR:

```
CHANGE-EVENT-5
   textpointer        increased
   PRECONDITION       AMOUNT-1
   EFFECT             AMOUNT-2
```

```
    TIME               (< FIND-ANCHOR-TIME)
    (< AMOUT-1.RANGE AMOUNT-2.RANGE)
AMOUNT-1
    DOMAIN             WINE-1
    PRECONDITION-OF    CHANGE-EVENT-5
AMOUNT-2
    DOMAIN             WINE-1
    EFFECT-OF          CHANGE-EVENT-5
WINE-1
    textpointer        wine
    DOMAIN-OF          AMOUNT-1 AMOUNT-2
    THEME-OF           DRINK-1
DRINK-1
    textpointer        consumption
    THEME              WINE-1
```

## 4 The Cross-Lingual Connection

As is clear from the examples above, OntoSem provides significant expressive power for meaning representation, which includes mapping to the ontology (which itself is rich in property-value descriptors), mapping to the ontology with lexical supplementation of properties, or referring to extra-ontological microtheories like those that treat modality, aspect, comparison, ellipsis resolution, time, etc. What must be emphasized, however, is how language neutral – and therefore portable across languages – the semantic descriptions are.

Whereas it is typical to assume that lexicons are language-specific whereas ontologies are language-independent, most aspects of the *sem-struc* zones of the OntoSem lexicons are language-independent, apart from the linking of specific variables to their counterparts in the *syn-struc*. Stated differently, if we consider *sem-strucs* – no matter for what language they originate – to be building blocks of the representation of *lexical meaning* (as opposed to conceptual meaning, as is done in the ontology), then the job of writing a lexicon for L2 based on the lexicon for L1 is in large part limited to a) providing an L2 translation for the head word(s), b) making any necessary *syn-struc* adjustments and c) checking/modifying the linking among variables in the *syn-* and *sem-strucs*. This conception of cross-linguistic lexicon development derives in large part from the Principle of Practical Effability (Nirenburg and Raskin 2004), which states that what can be expressed in one language can *somehow* be expressed in all other languages, be it by a word, a phrase, etc.

Apart from this theoretical justification for conceptualizing the *sem-strucs* as building blocks for lexical representation, there are two practical rationales: supporting consistency of meaning representation across languages and using acquirer time most efficiently in large-scale lexical acquisition.

As regards consistency, the potential for paraphrase must be considered when building multilingual resources. For instance, 'weapons of mass destruction' can be described as the union of CHEMICAL-WEAPON and BIOLOGICAL-WEAPON, or it can be described as WEAPON with the ability to KILL > 10,000 HUMANs (the actual number recorded will be treated by the analyzer in a fuzzy fashion; however, it would be less than ideal for a lexicon for L2 to record 10,000 while a lexicon for L3 recorded 50,000). While both representations are valid, it is desirable to use the same one in all languages covered. In addition, the decision of how to describe a notion – whether by ontologizing it, describing it using extra-ontological means, describing it using an existing concept with additional properties and values defined – is often a judgment call. It would not be desirable for the acquirer of German to map the word *Schimmel* 'white horse' to the concept HORSE with the lexical restriction COLOR: WHITE, while the acquirer of some other language that also has a word for 'white horse' introduced an ontological concept specifically for this entity. Again, while both representations are valid and, in this case, semantically equivalent, the general tendency should be to strive toward uniformity where possible.

As concerns acquisition time, composing *sem-strucs* is, by far, the most time- and effort-intensive aspect of writing OntoSem lexicon entries. This is a result of the richness of expressive means available to the acquirer; the fact that microtheories of time, reference, etc., are naturally built during lexicon development (recall that our environment is fully integrated with processors); and the fact that ontology development occurs hand-in-hand with lexicon development. Therefore, work on the first lexicon entry that describes a word sense takes much more time than editing a word sense for a new language. Moreover, although in the worst case some editing of entries is necessary for L2, L3, etc., in most cases no such editing is needed. Although one might hypothesize this state of affairs based on cross-linguistic principles, we have also tested it in the lexicon-porting experiment described in McShane, Zabludowski et al. 2004.

## 5 Discussion

This paper has presented our view of one of the main stated discussion issues for the workshop – the relation between ontological knowledge and knowledge of language. We have also touched on the issue of how ontologies facilitate multilingual processing by making it cheaper to acquire lexicons for L2, L3, etc. given the existence of an ontology/lexicon tandem of the kind used in OntoSem.

We have not, here, compared our work to other work in the field for the following two reasons. First, we have recently published comparisons between OntoSem and numerous available resources: WordNet, FrameNet and XTAG (Nirenburg et al. 2004b); SIMPLE (McShane et al. 2004c); and a number of corpus annotation schemes (McShane et al. 2005). Second, the purview of OntoSem is the automatic creation of structured knowledge from text, with that knowledge then serving as input to automatic reasoning systems ("intelligent agents") in a variety of planning and general problem-solving applications. This purview is broader than that of other approaches and puts a set of strong concurrent requirements on the various tasks of text processing as well as the nature of its results.

The core of semantic analysis, as we view it, is creating an unambiguous, language-independent representation of the meaning of text. Such a representation should, ideally, abstract away from the specific form of lexical input. This means that all of the following inputs – and many more – should give rise to the same Fact Repository information about our elephant (disregarding some details of speaker attitude and the like, which will also be recorded), which is extracted directly from the TMR of each sentence:

- That elephant's weight increased from 1000 to 1500 pounds.
- The elephant used to weigh 1000 pounds, now it weighs 1500.
- That elephant's weight went from 1000 lbs. to 1500 lbs.
- That same elephant gained a whopping 500 pounds: now it's 1500 pounds.
- What an elephant! He went from 1000 to 1500 pounds in no time!
- The elephant put on 500 pounds, recently weighing in at 1500 lbs.

Within our environment, all of these inputs do, in fact, result in the same basic Fact Repository information. In this way, when applications like question-answering, knowledge extraction and reasoning use OntoSem's Fact Repository as their search space, they use knowledge that is decoupled from the form in which it is expressed in text. Preparing the system to arrive at the same basic knowledge representation for such syntactically and lexically differing inputs requires the type of detailed acquisition efforts described in this paper.

A final word about implementation and evaluation. The research we are reporting is fresh off the presses, with the algorithmic work complete and implementation underway. We expect all aspects of processing described here to be implemented and tested prior to the conference. In the bigger picture, our research group is currently experiencing an exciting burst of progress, having recently implemented a development environment, called DEKADE, that supports the creation and evaluation of text-meaning representations, as well as the acquisition of the requisite knowledge resources, in a far more sophisticated manner than ever before (see McShane et al. 2005 for a description). Within the year we expect to collect a sufficient amount of data to permit a meaningful evaluation of the cost and efficiency of lexical and ontological acquisition. We have also started an evaluation regimen for the quality and efficiency of ontological-semantic analysis (see Nirenburg et al. 2004a). This regimen is by necessity, quite different from the ones typical of shallow-semantic systems, since the glass-box analysis of failures will be key to the continued improvement of the text-processing environment. Our longer-term goal with respect to evaluation is to conduct a full-scale evaluation of OntoSem within an end application.

Zooming out further still, versions of the OntoSem analyzer have already been used in a variety of applications, including question answering, machine translation and information extraction. While work must continue on both the OntoSem processors and its static knowledge resources (centrally including the ontology and the lexicons), the above applications have demonstrated the utility of this approach for higher-end NLP applications. OntoSem has been used with languages including English, Spanish, Chinese, Arabic and Persian, to varying degrees of lexical coverage (e.g., earlier, less fine-grained English and Spanish lexicons contained 40K entries and were used for MT in the Mikrokosmos project). What makes OntoSem amenable to efficient cross-linguistic usage is that many of the resources are either fully language independent (the ontology, the fact repository, the TMR metalanguage) or parameterizable in well understood ways (the lexicon).

## References

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

Marjorie McShane, Stephen Beale, Sergei Nirenburg and Tom O'Hara. 2005. Semantically rich human-aided machine annotation. *ACL Pie-in-the-Sky Workshop*, Ann Arbor, June 2005.

Marjorie McShane, Stephen Beale and Sergei Nirenburg. 2004a. OntoSem methods for processing semantic ellipsis. *Proceedings of HLT/NAACL 2004 Workshop on Computational Lexical Semantics.*

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

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

Sergei Nirenburg, Stephen Beale and Marjorie McShane 2004a. Evaluating the performance of the OntoSem semantic analyzer, *Proceedings of the ACL-04 Workshop on Text Meaning*, Barcelona, Spain, July.

Sergei Nirenburg, Marjorie McShane and Stephen Beale. 2004b. The rationale for building resources expressly for NLP, *Proceedings of LREC-2004.*

Sergei Nirenburg and Victor Raskin. 2004. *Ontological Semantics*. MIT Press.