

Type Construction and the Logic of Concepts

James Pustejovsky

Computer Science Department
Volen Center for Complex Systems
Brandeis University
Waltham, MA 02254 USA
`jamesp@cs.brandeis.edu`

Abstract

I would like to pose a set of fundamental questions regarding the constraints we can place on the structure of our concepts, particularly as revealed through language. I will outline a methodology for the construction of ontological types based on the dual concerns of capturing linguistic generalizations and satisfying metaphysical considerations. I discuss what “kinds of things” there are, as reflected in the models of semantics we adopt for our linguistic theories. I argue that the flat and relatively homogeneous typing models coming out of classic Montague Grammar are grossly inadequate to the task of modelling and describing language and its meaning. I outline aspects of a semantic theory (Generative Lexicon) employing a ranking of types. I distinguish first between natural (simple) types and functional types, and then motivate the use of complex types (dot objects) to model objects with multiple and interdependent denotations. This approach will be called the *Principle of Type Ordering*. I will explore what the top lattice structures are within this model, and how these constructions relate to more classic issues in syntactic mapping from meaning.

1 Language and Category Formation

Since the early days of artificial intelligence, researchers have struggled to find a satisfactory definition for category or concept, one which both meets formal demands on soundness and completeness, and practical demands on relevance to real-world tasks of classification. One goal is usually sacrificed

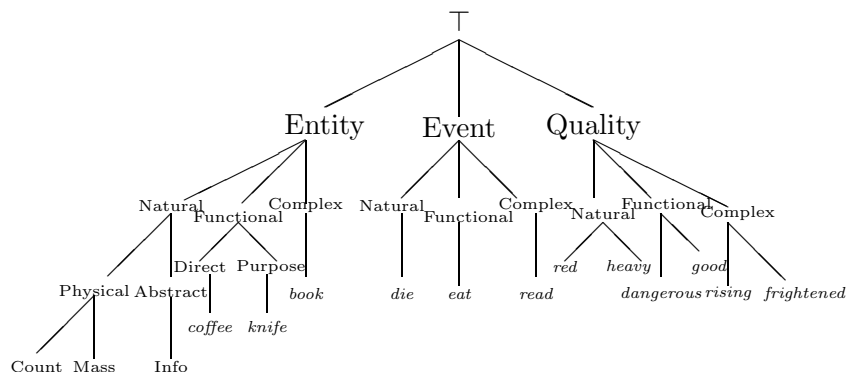
in the hope of achieving the other, where the results are muddled with good intentions but poor methodology.

Such comments are not restricted to classic AI approaches to knowledge representation alone, however, but are appropriately applied to the bulk of research done in theoretical linguistics on lexical representation as well. By rushing to describe a body of data from the perspective of syntactic structure and the resulting logical form, most assumptions regarding the underlying semantic “primitives” or categories used by the theories are opportunistic and do not impact on the question of whether the representations are conceptually appropriate or fitting. Thus, what is typically missing in these accounts of knowledge representation is the luxury of posing the fundamental questions of *sufficient reason*, perhaps the most basic of questions regarding concept construction. These include the following:

1. How can we choose between competing representation languages?
2. What makes one feature or category more appropriate or more suitable than another for constructing an ontology?
3. What justifies the admission of a feature or category into the ontology?
4. How are these categories put together to make larger ones (principle of compositionality)?
5. Can we quantify the size or complexity of our categories?

In this paper, I approach these issues from the combined perspectives of linguistic semantics and computational linguistics. What I hope to illustrate here is a fairly different approach to modelling categories, but one that I believe is both philosophically intuitive and linguistically motivated. I describe a framework within which we can construct increasingly complex types from a set of basic building blocks. The units of composition will correspond to conceptual categories that are primitive, but in no way fixed or static. Rather, these underlying units are highly functional in nature, and are structured according to a theory of semantic underspecification. The result is a type constructional system of concepts based in part on the classic Aristotelian notion of substance, together with the generative mechanisms recently developed in Generative Lexicon Theory. These include *qualia structure*, the *Principle of Orthogonal Inheritance*, and *Dot Object Construction*.

On this view, the resulting architecture of the upper concept lattice is structured into three domains: entities, qualities, and events. Each domain is itself structured by a type ordering relation, from simpler to more complex types. In this paper, I discuss a tripartite system, the simplest model employing the mechanisms of orthogonal inheritance and dot object construction. The resulting type structure is as shown in the figure below.



Tripartite Concept Lattice

The simple types in each domain are *Natural Types*, from which all other grounded types are constructed¹. *Functional Types*, on the other hand, combine qualia-based information from AGENTIVE and TELIC modes of explanation with a ground type, as we see below. *Complex Types* (or dot objects) are even richer in structure and are formed by the application of a type constructor, creating a type which is the reification of a specific *relation* between two types. I argue that this is a desirable characterization of natural categories and is strongly motivated by the linguistic data. Perhaps the most desirable consequence of this architecture of types is that we can provide distinct algebras of subtyping behavior for each type rank. In other words, the criteria for partitioning a type in the domain of Naturals to create subtypes will be very different from those that create subtypes in the Functional types and Complex types.

Before beginning our investigations into category structure, let me present some of the theoretical assumptions I will be making regarding the structure of the lexicon, and how this impacts our model of semantics. I assume that human cognitive abilities are manifested in the languages we speak and that central to this capability is the mental lexicon. Research in Generative Lexicon Theory points to a view of the mental lexicon that is neither that of a classical dictionary nor that of a warehouse of data within an information processing system. Rather, the lexicon is coming to be viewed as a complex, dynamic, and yet coherent system of knowledge, serving as the interface between structural linguistic operations and the compositional rules that create meaning. Such a design reflects some of the basic assumptions that have motivated generative linguistics over the past forty years. The computational perspective of generative lexicon theory entails studying how the combinatorics of semantic expressions is a reflection of the compositionality of thought itself. There are two questions central to this area:

¹Ungrounded types correspond to those concepts which are functional in nature, without reference to specific ground values, such as `tool` and so on. See below for further discussion.

1. What is the nature of our lexical knowledge that allows us to deploy a finite number of words in an unbounded number of contexts, and to successfully use our language to talk about the world in these contexts?
2. Is this lexical knowledge really separable from our ability to categorize and classify concepts in the world?

In this paper, it will be assumed that the first glimpse of our concepts comes from the lexicalization strategies employed in language. Equally important are the compositional processes giving rise to more complex concepts. This does not entail, however, a Whorfian position of conceptual relativism tied to linguistic behavior. Rather, this view is consistent with the belief that it is the logic of concepts itself which is parameter driven, and to which the generative mechanisms of GL are ultimately linked.

Generative Lexicon theory assumes that semantic descriptions, as constructed from lexical expressions, make use of four kinds (levels) of linguistic representations. These are listed below:

1. **ARGUMENT STRUCTURE:** Specification of number and type of logical arguments.
2. **EVENT STRUCTURE:** Definition of the event type of an expression and its subeventual structure.
3. **QUALIA STRUCTURE:** A structural differentiation of the predicative force for a lexical item.
4. **LEXICAL INHERITANCE STRUCTURE:** Identification of how a lexical structure is related to other structures in the type lattice.

The motivation for such apparently rich representations is actually grounded in a fairly conservative strategy of linguistic description matching semantic explanation. Consider, for example, how a conventional view on semantics models a verb such as *build* as a relation between individuals, e.g., $\lambda y \lambda x [build(x, y)]$. Recognizing the state of affairs of “building” itself as an individual gives rise to a neo-Davidsonian single event-place interpretation (cf. Parsons, 1990) of the relational reading;

- (1) labeljsent90111109 **Events as individuals:** $\lambda y \lambda x \lambda e [build(e, x, y)]$;

Furthermore, linguistic and semantic evidence suggests that many causatives are in fact not logically atomic from the perspective of event structure, but have internal events of their own. This gives rise to another enrichment in the logical form associated with the verb, where subevents are associated with subpredicates, corresponding to some logical “portion” of the verb’s meaning:

- (2) a. **Subevents as individuals:** $\lambda e_2 \lambda e_1 [..]$;
 b. **Subpredicates for the subevents:**
 $\lambda y \lambda x \lambda e_2 \lambda e_1 [build_1(e_1, x, ..) \wedge build_2(e_2, y, ..)]$;

Within GL, it has been argued that a further enrichment of this decompositional approach is motivated by both semantic and syntactic data; namely, the introduction of *qualia structure* as a cross-categorial representational tool. Briefly, qualia are Aristotelian “modes of explanation” for an entity or relation. They are defined in Pustejovsky (1995) with the following characteristics:

FORMAL: the basic category which distinguishes it within a larger domain;

CONSTITUTIVE: the relation between an object and its constituent parts;

TELIC: its purpose and function;

AGENTIVE: factors involved in its origin or “bringing it about”.

For relations, the qualia act in a similar capacity to thematic relations, but where the individual qualia are possibly associated with entire event descriptions, and not just individuals. Something like the expression in (3) would correspond to *build*:

- (3) a. **Qualia Identify the Substructures:**
 b. $\lambda y \lambda x \lambda e_1 \lambda e_2 [AGENTIVE = [build_1(e_1, x, ...)]$
 \wedge FORMAL = $[build_2(e_2, y, ...)]$
 \wedge TELIC = ...
 \wedge CONST = ...]

It can be seen that motivation for three of the four levels of representation in GL are tied to fairly familiar methodological strategies.² Viewed more conceptually, the qualia structure can be seen as providing functional tags to words, linking the words to a network of concepts.

$$(4) \quad \left[\begin{array}{l} \alpha \\ \text{ARGSTR} : \left[\begin{array}{l} \text{ARG1} : x \\ \dots \end{array} \right] \\ \text{EVSTR} : \left[\begin{array}{l} \text{EV1} : e_1 \\ \dots \end{array} \right] \\ \text{QUALIA} : \left[\begin{array}{l} \text{CONST} : \mathbf{\text{what } x \text{ is made of}} \\ \text{FORMAL} : \mathbf{\text{what } x \text{ is}} \\ \text{TELIC} : \mathbf{\text{function of } x} \\ \text{AGENTIVE} : \mathbf{\text{how } x \text{ came into being}} \end{array} \right] \end{array} \right]$$

Returning to the questions posed at the beginning of this section, we can now at least approach the major concern of this paper: namely, what kinds of things are there and can there be, conceptually? In the remainder of this paper, I turn to answering this question with the help of qualia structure. I will suggest that the qualia be deployed as the organizing principle of the logic of concepts.

2 Toward a Generative Type System

Conventional approaches to knowledge base design follow a standard strategy of creating an upper model of concepts, and based on subsumption relations, specializes these concepts accordingly to specific types as needed in a domain. Although it is standard to separate things from actions and qualities, beyond this initial partition, there is little agreed upon. For example, within the domain of entities (or things), much is left to interpretation, and multiple inheritance is conventionally a crucial tool to model the relationships between, for example, the concepts **food** and **apple**, or **animal** and **pet**. How are we to model concepts which seem to inherit from multiple basic types,

²The fourth, *Lexical Inheritance Structure*, relates to the very structure of categories themselves, the major topic we address later in this paper.

and what constraints can be imposed on the construction of such concepts, since this seems an unduely powerful mechanism of concept formation?

Consider, for example, the questions that arise when categorizing the relatively simple concepts below.

- (5) a. *stick*: individual physical entity;
- b. *rock*: mass or individual physical entity;
- c. *apple*: individual, possibly given purpose;
- d. *sandwich*: individual, with purpose, prepared artifactual;
- e. *cookie*: individual, with purpose, baked artifactual;
- f. *father*: relational concept between individuals;
- g. *boss*: relational concept between individuals;
- h. *friend*: long-term stage-level relational concept;
- i. *student*: short-term stage-level relational concept;
- j. *audience*: short-term stage-level concept;
- k. *laundry*: mass noun bipolar stage-level;
- l. *groceries*: mass noun monopolar stage-level;
- m. *human*: agentive intentional individual.

The first and most obvious distinction in our conceptual partitioning of the world involves the notion of natural kind. Hence, while sticks and rocks are naturally occurring entities, sandwiches and cookies are not. Apples are certainly natural kinds, but they like other natural objects can be grouped in other types of categories that do not include naturally occurring entities, such as cookies, for example, which are artifacts. Distinguishing between the concepts *father* and *boss* involves understanding the various social and functional roles that are possible between individuals. Such roles are typically identifiable eventualities, but needn't be; similar remarks hold for *student* and *friend*.

Some concepts are bound more directly to a specific activity or event, and because of this, the referring potential of words associated with such concepts is integrally linked to this defining event. For example, understanding the concept of *audience* is possible only by reference to the event which the individuals are engaged in attending. Hence, an audience might remain a *crowd* even after it is actively attending to a particular event, but it is no longer an audience; in other words, audiences are audiences of some event.

Concepts such as *laundry* and *groceries* are similarly defined relative to an activity which must be identified in order to properly understand them. But unlike *audience* however, these nouns do not behave relationally; rather, the activity is incorporated into the way the objects are identified. In fact, *laundry* is interestingly ambiguous because of this fact; it can refer to the dirty clothes that require cleaning, or to the clean clothes that have just been cleaned. The noun *groceries* appears to prefer the interpretation of “recently selected (food) stuff” and not “to be selected (food) stuff”, although this is somewhat unclear.

Conventional approaches to lexicon design and lexicography are relatively liberal with forming taxonomic structures where they seem appropriate, as informed by word senses in the language. A quick glance at the top concepts in WORDNET illustrates that the goal of this lexicographic exercise is to catalogue words as characterized by fairly local clusterings of semantic properties.

{act, activity}	{natural object}
{animal, fauna}	{natural phenomenon}
{artifact}	{person, human being}
{attribute}	{plant, flora}
{body}	{possession}
{cognition, knowledge}	{process}
{communication}	{quantity, amount}
{event, happening}	{relation}
{feeling, emotion}	{shape}
{food}	{state}
{group, grouping}	{substance}
{location}	{time}
{motivation, motive}	{food}

As with many ontologies, it is hard to discern a coherent global structure for the resulting classification beyond a weak descriptive labelling of words into extensionally defined sets. Abstracting away much of the top structure in the figure above, notice that WORDNET handles both contrastive ambiguity and complementary ambiguity (polysemy) in the same way; namely, by allowing multiple inheritance from the word denoting multiple concepts (cf. Pustejovsky, 1995 for discussion). Hence, for a polysemous noun such as

book, the classification strategy associates the different senses with different synsets, the two most important, perhaps, being **artifact** and **mental**. This is shown in the figure below.

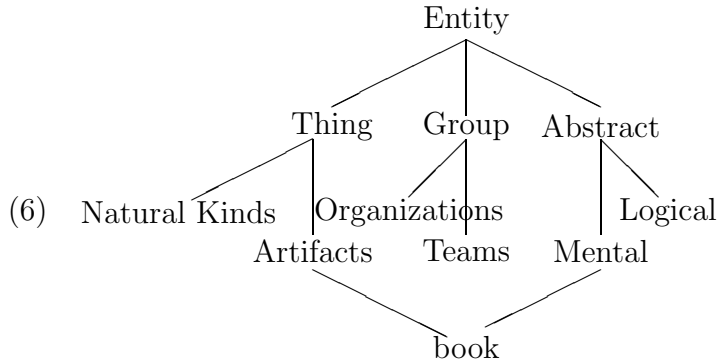


FIG. 1: CLASSIC TAXONOMY OF ENTITIES

It is not the goal of this paper to examine the shortcomings of WORDNET. What is important to point out, however, is the extreme difficulty facing semanticists when attempting to add ontological richness to a formal type system for language. WORDNET, of course, does not pretend to model lexical items in formal terms; many approaches to ontology construction, however, do formalize the categories as types, with no concern for what categories there can and should be. In the next section, I will argue that there are principled and intuitive methods for designing a lexicon, on metaphysical grounds. I will discuss three points relating to the design of an ontology as a type system:

- (a) The qualia are in fact generative coherence relations between types;
- (b) There is a formal *ranking* in the complexity of types;
- (c) The structure of these types is exploited in composition.

In what follows, I will map out how the qualia allow us to differentiate categories of concepts, at first broadly, and then quite specifically.

3 The Natural Types

In this section, I begin with an exploration into the basic distinction between natural kinds and artifacts; this is one of the oldest categorical distinctions made in the literature, and exercises philosophical discourse to the present (see Putnam, 1975, Quine, 1960, others). There are several questions that pertain to this distinction, including the satisfaction conditions on membership in a category, but what will concern us presently is the difference in how natural kinds are *evaluated* relative to artifacts. I will argue that the major discriminant in distinguishing between natural kinds and artifacts is *intentionality*; intentionality is expressed in relation to the AGENTIVE and TELIC qualia in GL. This will form the basis for ranking the types for the kinds of things there are into a hierarchy of types, the *Principle of Type Ordering*.

Consider, for illustration, the use and meaning associated with the two nouns *chair* and *rock* in the contexts below, where “!” will indicate a semantic anomaly.

- (1) a. This is a good chair.
b. !This is a good rock

- (2) a. Mary enjoys that chair.
b. !Mary enjoyed the rock.

On a very basic level, the evaluative predicates present in these examples refer to the function of the object or material; hence the judgments in these sentences are functional ones. For natural kinds such as *rock*, *tiger*, and so on, this is only possible when we have reconceptualized the concept to also carry some intention along with it. For example, if rocks are judged relative to how good they are for climbing on, then the sentences in (1b) and (2b) are well-formed enough. Similarly, although *apples* are natural kind plant products, we can categorize them as a food stuff, a category which is typically grouped into natural kinds as well. Other sorts of food stuff, such as cookies and cakes are, however, artifactual in nature, but what are the discriminant properties that tell us this is so? On this view, classifying them as foods does not tell us what category they belong to, but reference to an event of creation would do just that. Hence, although there are some diagnostics for determining what is natural or not, they do not appear to relate to intentionality.

In order to better understand how intentionality affects the conceptualization strategies in our thought, imagine a language, \mathcal{L}_N , without intentionality. We will say that all words and their compositions are *Natural Types*. Differentiation of the types in this language is accomplished by the FORMAL quale:

- (7) a. *stick, lion, pebble*
 b. *water, sky, rock*

\mathcal{L}_N is closed under the operation, CONSTITUTION:

- (8) a. *wood stick*; $\lambda x \exists y [\textit{stick}(x) \wedge \textit{wood}(y) \wedge \textit{made_of}(x, y)]$
 b. *cloud sky*; $\lambda x \exists y [\textit{sky}(x) \wedge \textit{cloud}(y) \wedge \textit{made_of}(x, y)]$

\mathcal{L}_N should not be closed under AGENTIVE (Coming into Being), however, since it involves intentional action toward the entity.

- (9) a. These are invalid forms: the “passive” predicates entail agency:
 b. *carved stick*; $\lambda x \exists e [\textit{stick}(x) \wedge \textit{carved}(e, x)]$
 c. *chipped stone*; $\lambda x \exists e [\textit{stone}(x) \wedge \textit{chipped}(e, x)]$

But this is only if it is interpreted non-intentionally, and simply as “action.”

Following the notation in Asher and Pustejovsky (1999), I will interpret the conventional feature-based qualia structure of GL in terms of an algebra of types, operating under very restrictive rules of combination. For example, instead of the feature structure in ((10)), where constitution as a quale is a feature value, we can represent this aspect of the meaning as integrally part of the basic type structure itself.

- (10) a.
$$\left[\begin{array}{c} x \\ \text{QUALIA} : \left[\begin{array}{c} \text{FORMAL} : \sigma \\ \text{CONST} : \tau \end{array} \right] \end{array} \right] \implies$$

 b. $x : \sigma \textcircled{C} \tau$

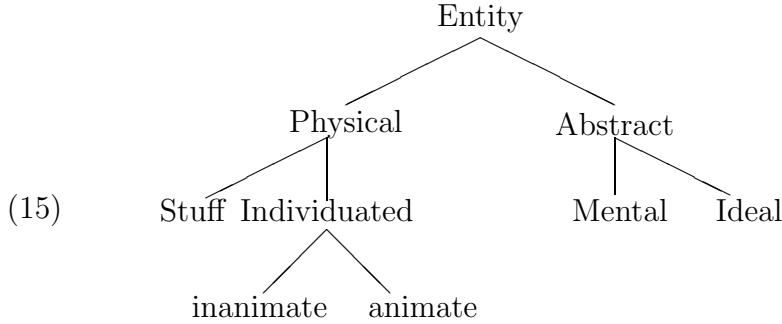
Hence, what is needed to represent both FORMAL and CONST aspects in \mathcal{L}_N

- (11) If σ and τ are types in \mathcal{L}_N , then $\sigma \textcircled{C} \tau$ is also in \mathcal{L}_N , where \textcircled{C} expresses the constitutive relation, as defined in GL.

From the examples above, we would thus have the following types in \mathcal{L}_N , ignoring the actual composition rules giving rise to the compound interpretations:

- (12) a. *wood stick; stick*©*wood*
 b. *cloud sky; sky*©*cloud*

Unless specified explicitly, an explicit constitution for an entity is not entailed. However, for physical entities, *some* constitution is entailed for that type. It is important to point out that this analysis states that the constitutive qualia relation is primary and extensional in nature.³ Viewed in isolation, the set of natural types \mathcal{N} is not very informative or illuminating, but they provide the necessary building blocks on top of which to define our other types. Given these assumptions, the upper lattice structure the natural types will refer only to unintentional objects and will have the following sort of structure:



UPPER NATURAL TYPE LATTICE

³In Pustejovsky (1998), the formal type structural distinction between individuals and mass concepts is handled by an explicit “linkage” between the CONST and FORMAL in the qualia structure. Thus, ((13)) was the abstract feature representation for a mass term such as *water*, while ((14)) modelled count nouns such as *boy* and *stick*.

$$(13) \left[\begin{array}{l} \alpha \\ \text{QUALIA} : \left[\begin{array}{l} \text{FORMAL} : \boxed{a} \\ \text{CONST} : \boxed{1} \end{array} \right] \end{array} \right]$$

Individuals where $a \neq b$:

$$(14) \left[\begin{array}{l} \alpha \\ \text{QUALIA} : \left[\begin{array}{l} \text{FORMAL} : a \\ \text{CONST} : b \end{array} \right] \end{array} \right]$$

With the present type notation, this can be handled in a parallel fashion with the CONST type constructor ©.

Now consider the predicates that select for just these natural types. In other words, once we have defined the natural type entities, we are in a position to define the natural predicates and relations that correspond to these types. First, let us review some notation. I assume a *typing judgment*, $g \vdash \alpha : \tau$, with respect to a grammar to be an assignment, g , an expression, α , and a type, τ , such that under assignment g , the expression α has type τ . In the case of the natural types, I will also assume the following equivalence:

$$(16) \quad g \vdash x : \alpha \in \mathcal{N} =_{df} g \vdash x : e_n$$

Then, for the construction of predicates from natural types, we have the following: for the predicates below, e_n and \underline{t} are in the set of Natural Types, \mathcal{N} , structured as a join semi-lattice, $\langle \mathcal{N}, \sqsubseteq \rangle$;

- (17) a. *die*: $e_n \rightarrow \underline{t}$,
 b. *touch*: $e_n \rightarrow (e_n \rightarrow \underline{t})$,
 c. *be under*: $e_n \rightarrow (e_n \rightarrow \underline{t})$,
 d. *give*: $e_N \rightarrow (e_N \rightarrow (e_N \underline{t}))$.

The predicate and relational types that result from natural type entities are just those predicates and relations that are natural types themselves.

Observe that the propositions formed by the composition of a natural predicate with a natural type entity will be *brute propositions*, making use of brute entities, i.e., natural types. Examples of such propositions are given below:

- (18) a. The rabbit died.
 b. The rock touches the water.
 c. The ants are under the tree.

It is interesting to compare this to Anscombe's (1958) and Searle's (1995) proposals regarding "brute facts", where facts are classified according to the kinds of participant descriptions they contain. This is very much in line with the research strategy here: in fact, as we shall see, the qualia and the principle of type ordering will allow us to enrich this "fact classification" even further.

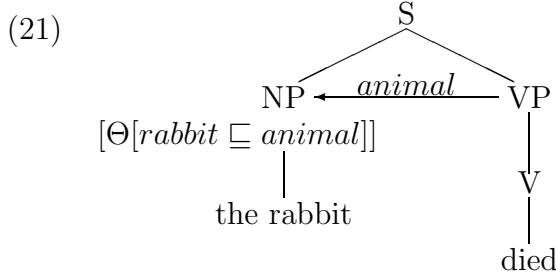
So far, I have outlined a simple class of types for entities and relations, \mathcal{N} and $\mathcal{N} \times \mathcal{N}$. Now let us explore how these types are deployed in composition in syntax. First we turn to the manner in which subtypes in the lattice, \mathcal{N}

are accepted in selective contexts. As in Pustejovsky (1995), we express a *subtyping coercion* relation, Θ , for these judgments as follows:

$$(19) \frac{x : \sigma_1, \quad \Theta[\sigma_1 \sqsubseteq \sigma_2] : \sigma_1 \rightarrow \sigma_2}{\Theta[\sigma_1 \sqsubseteq \sigma_2](x) : \sigma_2}$$

This says that, given a variable x of type σ_1 , which is a subtype of σ_2 , there is a coercion possible between σ_1 and σ_2 , which changes the type of α in this composition, from σ_1 to σ_2 . The typing relation between the subtype **rock** and the type selected by a governing verb *throw*, for example, namely **phys_obj**, is respected by this coercion relation, Θ . Similarly, since the concept **rabbit** is subtyped under **animal**, it falls under the same coercion operation. Both are shown below in ((20)) and a composition is illustrated in ((38)), where the type for the verb *die* is $animal \rightarrow \underline{t}$;

- (20) a. $\Theta[rock \sqsubseteq phys_obj] : rock \rightarrow phys_obj$
 b. $\Theta[rabbit \sqsubseteq animal] : rabbit \rightarrow animal$

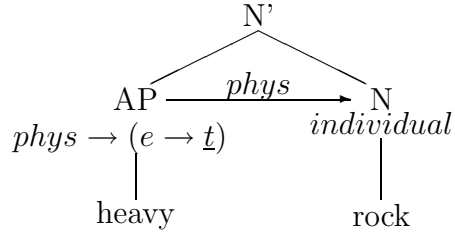


Adjectives will be typed in a similar fashion to predicates, selecting directly for a natural type and thereby becoming *natural qualities*, as shown in ((22)) below.

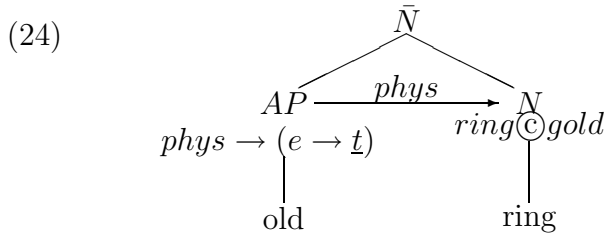
- (22) a. *red*: $e_n \rightarrow \underline{t}$
 b. *heavy*: $e_n \rightarrow \underline{t}$

Ignoring for now how the semantics of attributive modification treats the type differently, we see a straightforward composition in ((40)).

(23)



Given the above representation for FORMAL and CONST qualia as a specialized kind of composite type structure, it is no surprise why adjectives such as *old* in the noun phrase *an old ring* appear to be able to predicate of either quale, FORMAL or CONST.



Briefly, within the type structure for *ring*, either quale is satisfied by the type restrictions of the adjectival, giving rise to the two interpretations below. Although the CONSTITUTIVE interpretation is not as acceptable as the FORMAL, assume composition allows for type selector functions (cf. Pustejovsky, 1995) such as Σ_1 and Σ_2 below, thus solving the classic mereological puzzle discussed in Bach (1986) and Link (1998).

- (25) a. $\Sigma_1[\text{ring} \textcircled{\text{C}} \text{gold}] : \text{ring}$: “the object is old as an artifact”;
 b. $\Sigma_2[\text{ring} \textcircled{\text{C}} \text{gold}] : \text{gold}$: “the constitution of the ring is itself old”.

This states essentially that, when there are distinct values (types) for the CONST and FORMAL, the selective ability of adjectives such as *old* allows for a non-specific interpretation.

An even more interesting ambiguity that arises with the selective properties of *old* is when it assumes the interpretations of *former* and *previous*, as in “an old car” and “an old girlfriend”, vs. “an old friend” which assumes either a FORMAL modification or a TELIC modification, preferably the TELIC; i.e., a friend that one has known for a long time. These were discussed briefly in Pustejovsky (1995) and have recently been discussed more fully in

Bouillon (1997) and Bouillon and Busa (1998). In the present discussion, these interpretations can only be available for functional types, the class of concepts to which we turn presently.

4 Functional Types

Up to this point we have discussed only natural types. In some fairly obvious sense, these types refer to real objects that are identified through classic principles of individuation. But \mathcal{L}_N is not a very expressive language, because any trace of intentionality is absent. This is what the operations of the TELIC or AGENTIVE qualia do; they introduce intentionality, giving rise to the generation of our first *virtual types*, concepts referring to natural types which are colored by intentional descriptive content. For example, the identification of any of the natural types *stick*, *stone*, or *water*, as functioning in the capacity of the activities of *hitting*, *throwing*, or *drinking*, respectively, is a compositional operation that is not supported by \mathcal{L}_N .

- (26) a. Introduction of TELIC:
 b. *hitting stick*; not in \mathcal{L}_N ;
 c. *throwing stone*; not in \mathcal{L}_N ;
 d. *drinking water*; not in \mathcal{L}_N .

The introduction of TELIC above generates a functional description for an entity without of course creating a new entity in the world. Hence, we have a new domain of entities in our type system, the **Functional Types**, \mathcal{F} , which are virtual, in that from a realist perspective, each is still identifiable by the properties that satisfy its being a natural type, which forms the *ground* for the functional type.

The other aspect of intentional description is associated with the AGENTIVE quale. Starting again with natural types, we can identify *artifacts* as those naturals with identified AGENTIVE and TELIC roles (see Pustejovsky, 1995). Natural artifacts are those naturals with no expressed purpose or TELIC associated with them; the Ajd-N cases in ((27)) are a good example of this interpretation.

- (27) a. Introduction of AGENTIVE:
 b. *carved stick*; not in \mathcal{L}_N ;

- c. *flaked stone*; not in \mathcal{L}_N ;
- d. *boiled water*; not in \mathcal{L}_N .

We will call such concepts “semi-intentional”, when reference only to AGENTIVE is made. In these examples, the natural type has been transformed or modified from its original state, but not brought into existence by the referenced activity (event). Clearly, real artifactual concepts such as *table*, *knife*, and *computer*, are intentionally defined by reference to both AGENTIVE and TELIC. In fact, for such objects, it is difficult to imagine creation without purpose.

Other concepts that are semi-intentional in nature are types involving individuals where a relational state is defined in terms of the AGENTIVE quale. For example, the classification of two natural type individual humans as entering into the relations of *brother* or *father* also constitutes a semi-intentional type. Reference to the functional relation of two individuals through the TELIC however, is a purely intentional relation, such as illustrated in the examples in ((28)).

- (28) a. Introduction of TELIC for human:
 b. *boss*;
 c. *friend*;

Combining both TELIC and AGENTIVE for humans gives rise to concepts such as *wife* and *president*, where social function and social modes of creation are folded into one concept and one lexicalized item. Not surprisingly, the relations associated with such types will also be functionally defined, e.g., *elect*, *vote*, *marry* (see below).

A large subclass of such nouns, the *agentive nominals*, has been studied recently in Busa (1996, 1999), where TELIC and AGENTIVE values for nouns such as *violinist* and *pilot* characterize the relation that the individual has to its defining event or event descriptions. For the nouns presented above, institutionally defined roles such as *boss*, *president*, or *wife* make reference to events through the TELIC or AGENTIVE.

For the present discussion we will interpret the feature-based representations of qualia structure as types, adopting and extending the framework introduced in Asher and Pustejovsky (1999).⁴ In earlier treatments of types

⁴In Asher and Pustejovsky’s (1999) Dot Logic, no explicit distinction is made in the type structure between the qualia roles; in other words, they are all introduced as tensor

in GL (Pustejovsky, 1995), unified types were represented as the result of a meet of two types from the type lattice, $\sigma \sqcap \tau$, where

$$(29) \quad \begin{array}{l} \text{a. } \sigma = [\sigma \dots [Q_F = \alpha]] \\ \text{b. } \tau = [\tau \dots [Q_T = \beta]] \end{array}$$

In dot logic, the tensor type constructor \otimes introduces a quale-relation as part of the type directly, $\sigma \otimes \tau_T$.

$$(30) \quad \begin{array}{l} \text{a. } g \vdash x : \alpha =_{df} g \vdash x : e_n \\ \text{b. } g \vdash x : \alpha \otimes \beta_T =_{df} g \vdash x : e_f, \text{ where } x : \alpha \otimes (\dots\beta_i\dots) \wedge Telic(x, y) \rightarrow \\ \quad y : \beta_i \end{array}$$

For example, the feature-based qualia structure for a noun such as *beer*, shown in ((31)),

$$(31) \quad \left[\begin{array}{l} \mathbf{beer} \\ \text{ARGSTR} : [\text{ARG1} : \mathbf{x} : \mathbf{liquid}] \\ \text{QUALIA} : \left[\begin{array}{l} \text{FORMAL} : \mathbf{x} \\ \text{TELIC} : \mathbf{drink}(e^P, \mathbf{y}, \mathbf{x}) \end{array} \right] \end{array} \right]$$

can be viewed directly as a type as follows:

$$(32) \quad \mathbf{beer} : \mathbf{liq} \otimes \mathbf{drink}_T$$

Similarly, the unified type for `phys_artifact_tool`, shown in ((33))

$$(33) \quad \left[\begin{array}{l} \mathbf{phys_artifact_tool} \\ \text{ARGSTR} : \left[\begin{array}{l} \text{ARG1} : \mathbf{x} : \mathbf{physobj} \\ \text{D-ARG1} : \mathbf{y} : \mathbf{human} \end{array} \right] \\ \text{QUALIA} : \left[\begin{array}{l} \text{FORMAL} : \mathbf{x} \\ \text{TELIC} : \mathbf{R}(e^P, \mathbf{y}, \mathbf{x}) \\ \text{AGENTIVE} : \mathbf{make}(e^T, \mathbf{y}, \mathbf{x}) \end{array} \right] \end{array} \right]$$

types to a base type, σ , i.e., $\sigma \otimes \tau$. The basic set of types there is defined as follows:

- (i) If σ and τ are types, then so is $(\sigma \rightarrow \tau)$
- (ii) If σ and τ are types, then so is $(\sigma \bullet \tau)$.
- (iii) If σ and τ_1, \dots, τ_n are types, then so is $(\sigma \otimes (\tau_1 \cdots \tau_n))$.

In the present work, I adopt the basics of the Dot Logic, but also introduce explicit reference to qualia by name, i.e., $\sigma \otimes \tau_T$ and $\sigma \otimes \tau_A$, for reference to TELIC and AGENTIVE respectively. Furthermore, here CONST is introduced as a type operation, $\sigma \odot \tau$, directly on the base type.

can be represented as a type as follows:⁵

$$(34) \quad phys \otimes make_A \otimes \epsilon_T$$

In the previous section, natural predicates were defined in terms of natural entity types. In what follows, we will show how functional predicates are defined in terms of functional entities. As a result of this, it will be shown how the predicate inherits the intentionality of its arguments directly. Assume e_f is in the set of Functional Types, \mathcal{F} , structured as a join semi-lattice, $\langle F, \sqsubseteq, \rangle$. Consider the predicates below;

$$(35) \quad \begin{array}{l} \text{a. } spoil: e_f \rightarrow \underline{t}, \\ \text{b. } eat: e_f \rightarrow (e_n \rightarrow \underline{t}), \\ \text{c. } feed: e_f \rightarrow (e_f \rightarrow (e_f \rightarrow \underline{t})), \end{array}$$

Below are examples of *functional propositions* composed from functional entities and functional predicates, i.e., functional types;

$$(36) \quad \begin{array}{l} \text{a. The beer spoiled.} \\ \text{b. The rabbit ate the carrots.} \\ \text{c. The rabbit fed the bunny the food.} \end{array}$$

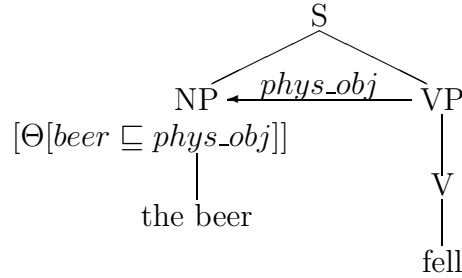
The judgments expressed by the above propositions entail brute (or natural) propositions, but go beyond them as they also express judgments of intentional content, which natural propositions do not. To illustrate how this separation in judgment is actually calculated compositionally, consider the sentence below in ((37)).

$$(37) \quad \text{The beer fell.}$$

If a functional expression such as *the beer* is predicated by a natural event expression, then the natural proposition is denoted by virtue of the base natural type constituting the expression, i.e., *liquid*, as shown in the derivation below:

$$(38)$$

⁵Just as the original Dot Logic was extended to include the constitutive type operator, ©, we could complete the extension to map isomorphically to the complete set of qualia by partitioning the tensor type constructors into an AGENTIVE operator, AT, and a TELIC operator, *. Then, the type in ((34)) would be represented as *phys AT make * ε*.

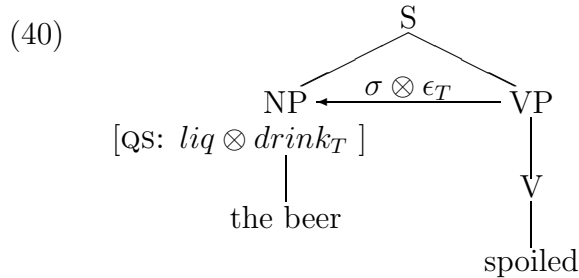


However, in the context of a functional predicate, such as the verb *spoil*, the same expression denotes both a natural proposition and a functional proposition, as explained below. Because the predicate *spoil* selects for not just a natural type, \mathcal{N} , but a *functional type*, \mathcal{F} , as its subject, the general type of the predicate *spoil* is $e_f \rightarrow \underline{t}$, and specifically it is typed as ((39)):

α is from the domain of \mathcal{N} .

$$(39) \text{ phys} \otimes \epsilon_T \rightarrow \underline{t}$$

Therefore, functional type selection is selection of not only the base type but of the functional component as well, as illustrated below.



There are two logically distinct components to what is asserted by this sentence: (a) some physically tangible event is implicated relative to a quantity of liquid (this is the brute proposition); (b) the event is intentionally described as a spoiling, and the liquid is beer (this is the functional proposition).

5 A Complex Type Language \mathcal{L}_C

\mathcal{L}_C introduces types containing a coherence relation between (at least) two natural or functional types. The resulting objects are also *virtual types*:

$$(41) \text{ a. physobj}\cdot\text{info: e.g., book, record;}$$

- b. **event·event**: e.g., *construction*, *examination*;
 - c. **event·question**: e.g., *exam*;
 - d. **animal·rational**: e.g., *person*.
- (42) a. Introduction of DOT:
- b. *book*; not in \mathcal{L}_F ;
 - c. *money*; not in \mathcal{L}_F ;

For the predicates below, e_C is in the set of Complex Types, \mathcal{C} , structured as a join semi-lattice, $\langle C, \sqsubseteq \rangle$;

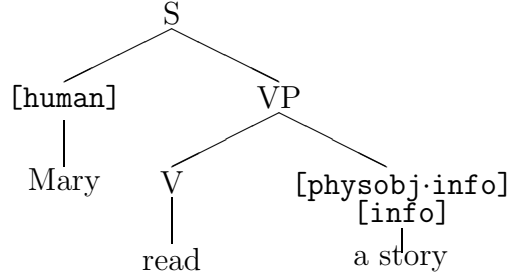
- (43) a. *play*: $e_C \rightarrow \underline{t}$, or $e_C \rightarrow (e_C \rightarrow \underline{t})$,
- b. *read*: $e_C \rightarrow (e_C \rightarrow \underline{t})$,
 - c. *buy*: $e_F \rightarrow (e_C \rightarrow (e_C \rightarrow \underline{t}))$,

Examples of *complex propositions* making use of complex entities, i.e., complex types;

- (44) a. The music played.
 b. The man read the book.
 c. The man bought the bunny from the vendor.
- (45) a. **Type**(*book*) = **physobj.info**
 b. **Type**(*story*) = **info**
- (46) a. Mary read a book.
 b. Mary read a story.
- (47) a. Mary told a story.
 b. *Mary told a book.

Notice how selection by the predicate *read* of a non-complex type such as *story* results in a coercion of the complement to that required by the governing predicate;

- (48)



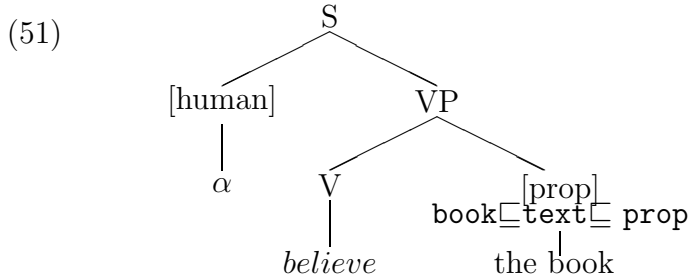
The coercion operations projecting one type from the complex type are a special case of *type pumping* (or projection), defined as Σ_1 and Σ_2 below. These two operations, together with the dot object itself form the definition of the type cluster called a lexical conceptual paradigm (lcp).

$$(49) \text{ lcp} = \{\sigma_1 \cdot \sigma_2, \Sigma_1[\sigma_1 \cdot \sigma_2] : \sigma_1, \Sigma_2[\sigma_1 \cdot \sigma_2] : \sigma_2\}$$

- (50) a. $\Sigma_1[\text{info} \cdot \text{physobj}] : \text{info}$
 b. $\Sigma_2[\text{info} \cdot \text{physobj}] : \text{physobj}$
 c. $\text{info} \cdot \text{physobj} \cdot \text{lcp} = \{\text{info} \cdot \text{physobj}, \text{info}, \text{physobj}\}$

$\text{book} \sqsubseteq \text{text} \sqsubseteq \text{prop}$

In the case of a predicate that actually selects for only one component in the type complex of the complement, such as that shown below with the verb *believe*, type pumping allows the propositional interpretation of the complex type to project and satisfy the selectional constraints of the predicate.



The type derivation below shows both the application of the type pumping operator, Σ , and the subtyping relation, Θ . The lcp interpretation of *book*, and there exists a subtyping relation $\alpha \sqsubseteq S'$.

$$(52) \frac{\Sigma_1(\text{info} \cdot \text{physobj}) : \text{info} \quad , \quad \Theta[\text{info} \sqsubseteq \text{prop}] : \text{info} \rightarrow \text{prop}}{\Theta[\text{info} \sqsubseteq \text{prop}](\Sigma_1(\text{info} \cdot \text{physobj})) : \text{prop}}$$

- (53) a. Mary believes the book.
 b. **believe**($\hat{\Theta}(\Sigma_1(\mathbf{the-book}))$)(**Mary**) \Rightarrow
 c. **believe'**($\hat{\Theta}(\mathbf{the-book:info})$)(**Mary**) \Rightarrow
 d. **believe'**($\hat{\mathbf{the-book:prop}}$)(**Mary**)

6 Coercion Revisited

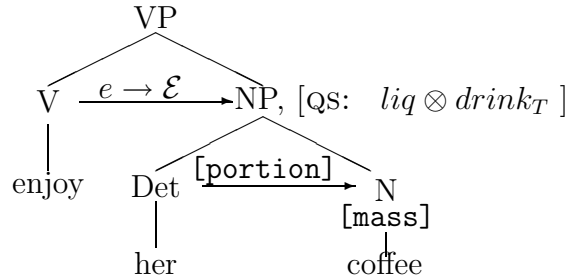
In this section I will discuss briefly what consequences of the present type system there are for operations of type coercion. Following the basic formulation of type coercion in Pustejovsky (1995), as a semantic operation that converts an expression, α , to the type expected by a governing function, β , it was suggested that α has a set of type shifting operators, Σ_α , associated with it which may operate over an expression, changing its type and denotation. These operators are the qualia themselves and the resulting types are the values of the qualia. Following the standard GL analysis of coercion in complement position⁶, I will assume that the verb *enjoy* selects for an eventual function, that is, an unsaturated event description (see Pustejovsky, 1993, 1995 for discussion). Thus, although the NP *her coffee* does not satisfy the typing environment of the governing verb *enjoy*, it is coerced to the appropriate type by the operation of *type coercion*. The compositional processes in the grammar results in the coercion of the NP complement *her coffee* into an event description, whose subject is controlled by the predicate *enjoy*.

(54)

⁶I assume some version of function application with coercion (FAC), as stated below:

FUNCTION APPLICATION WITH COERCION (FAC): If α is of type **c**, and β is of type $\langle \mathbf{a}, \mathbf{b} \rangle$, then,

- (i) if type **c** = **a**, then $\beta(\alpha)$ is of type **b**.
- (ii) if there is a $\sigma \in \Sigma_\alpha$ such that $\sigma(\alpha)$ results in an expression of type **a**, then $\beta(\sigma(\alpha))$ is of type **b**.
- (iii) otherwise a type error is produced.



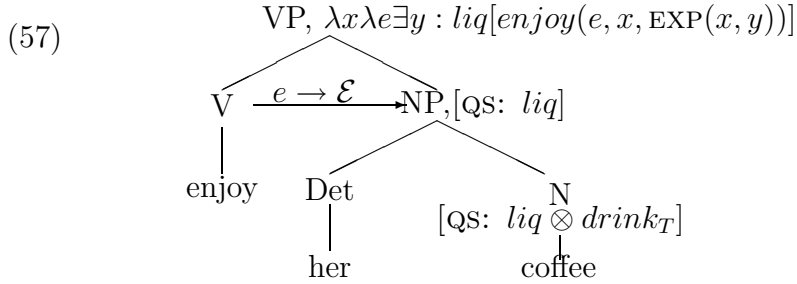
The exact value of the predicate in this event description is mediated by two factors, described in ((55)) below.

- (55) a. SPECIFICATION IN COERCION: For a coercing predicate β and its argument α , the specification is controlled by two factors:
- b. the selectional specificity of the coercing predicate;
 - c. the aliases, Σ_α , available to the argument being coerced. There are two types of aliases for an expression:
 - (i) Globally available methods of type-shifting, such as *grinding* and *packaging*;
 - (ii) Locally available values in the qualia structure of an expression, such as TELIC and AGENTIVE events.

The property exhibited by coercing predicates such as *enjoy* is interesting because the verbs do not seem to ever fail in coercion. That is, although there may be no obvious or “proximate” interpretation for the sentences in ((56)), there are legitimate default readings available, even in the absence of qualia-derived interpretations. The two sentences in ((56)) for example, are cases in point.

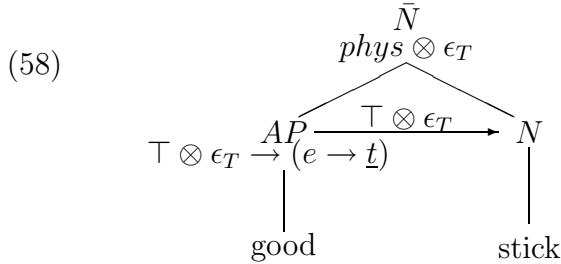
- (56) a. Mary enjoyed the rock.
 b. John enjoyed the flower.

Namely, Mary might enjoy the way a rock feels, and John the way a flower looks or smells, all of which are perceptual experiences. For such top-typed experiencing predicates, it is hard to imagine a semantically ill-formed “enjoying event”. In such cases where the complement is a functional type, the natural component of a functional type can be selected and is itself coerced into the selected type. For this reason, I shall refer to such type changing contexts as *Natural Coercions*. Returning to the example in ((54)), the same structure viewed as undergoing a natural coercion results in the following interpretation:



Hence, natural coercions (such as with *enjoy*) impose an event description interpretation on its complement, even in cases where the complement itself does not directly carry such information.

Evaluative predicates, on the other hand, such as *good* and *effective*, impose functional descriptions on their arguments, and as such shift the order or rank of the argument to a functional type.



With the introduction of type order (or *rank*), it is possible to classify the kinds of coercion operations a bit more precisely than has been accomplished thus far. There are two parameters which can help us distinguish the types of coercion operations that occur in grammar. One refers to the integrity of the type within its rank, while the other refers to the rank itself. For example, coercions preserving the order of the type will be called *Rank-preserving*; this is where an expression $x : \alpha \in \mathcal{N}$, remains in \mathcal{N} after the application of coercion. A coercion not having this property is said to be *Rank-shifting*. A coercion is *Domain-preserving* when the coerced expression α is not shifted from its domain, i.e., entities, events, or properties, during the coercion. If this is not the case, the coercion is said to be *Domain-shifting*. These properties are summarized in the table below.

(59)

COERCION TYPES	Rank-preserving	Rank-shifting
Domain-preserving	<i>Subtyping</i> ((60)a)	<i>Evaluative Predicates</i> ((60)c)
Domain-shifting	<i>Natural Coercion</i> ((60)b)	<i>Imposed Telic</i> ((60)d)

TABLE OF COERCION RELATIONS

The sentences in ((60)) contain instances illustrating the distinct coercion types referred to in the table.

- (60) a. Mary threw the rock.
b. John enjoyed the flower.
c. The water spoiled.
d. John began the rock.

The actual complement in ((60)a) is a subtype of that selected for by the verb *throw*, as seen above in ((20)). The coercion in ((60)b) has already been discussed, while ((60)c) is a clear example of a natural type instance being coerced to a functional interpretation. The aspectual coercion in ((60)d) combines this functional interpretation with a domain-shift to a controlled event description. Because both parameters are being positively deployed in this example, this sentence is clearly the hardest for which to find a natural interpretation.

7 Elements of Type Construction

In this section, I turn to two of the questions posed in the first section of the paper, regarding how discriminant features are chosen for constructing the relationship between types within each type lattice. These questions are repeated below:

- (2) What makes one feature or category more appropriate or more suitable than another for constructing an ontology?
- (3) What justifies the admission of a feature or category into the ontology?

The extensions to GL I have been outlining above are an attempt to answer just these questions; they provide us with two new strategies for type construction:

- (a) The criteria for subtyping are distinct to each type rank; that is, each level of type is structured according to distinct partitioning strategies;
- (b) Underspecification over the different levels of representation in GL allows for parameterizable views on specialization relations in the type system.

To make these points clearer, I will first discuss briefly how the type-subtype relation in the natural type lattice, \mathcal{N} , is structured. Then, I show how the same relation in the domain of functional types, \mathcal{F} , must be characterized in terms of a subtyping relation making reference to event descriptions. The goal of the following sections is not to actually “construct” the types corresponding to the natural lattice, but to rather understand what local constraints exist within the lattice, and how they are structured relative to the global constraints in the type system. Because of space, I will have little to say about the subtyping relation in the rank of complex types, but see Pustejovsky (forthcoming) for discussion.

The premise of the current work is that it is the lexicalization strategies in a language which give us our first glimpse of the concepts behind our thoughts. Equally important of course are the compositional structures that give rise to new concepts. Constructing a model for how our meanings are built up and modified in context has always been at the very center of Generative Lexicon Theory. The consequences of adopting such a thesis are that ontological considerations are unavoidable when studying categories and have to be seriously addressed when studying the compositional processes involved in language. Although the discussion so far has focused on the structure of the tripartite type system and the effects of this logic on the consequent descriptive and explanatory mechanisms in Generative Lexicon theory, in this section, I would like to touch on some of the assumptions underlying the construction of the types themselves, as seen from underlying conceptual principles. The remarks in this section will be brief, and for a longer treatment of these issues, the reader is referred to Pustejovsky (forthcoming).

As described above, the global algebraic constraints on the type system are uniform but weak in nature. Understanding the partitionings at different levels of the type system involves appreciating what discriminant predicates, D_i , have been employed to generate the subtypes in a particular sublattice. To begin with, following some general assumptions in knowledge representation (see Simons, 198?, Smith, 199?, and Johansson, 198?), I assume that there are metalanguage predicates which act to generate partitions on the upper level of the type system. For the domain of natural types, \mathcal{N} , this will involve reference to the set of *logical discriminants* (L-discriminants), which are essentially a priori categorization schemas, as in Carnap (1928).

Earlier we discussed the structure of the natural lattice \mathcal{N} , but we did not address the issue of what logical coherence existed in the lattice beyond the minimal constraints imposed by the join semi-lattice structure. In other words, many constructional issues remain as to how the type system is built, and according to what principles. For two entities in the set of natural types, $\sigma, \tau \in \mathcal{N}$, where $\sigma \sqsubseteq \tau$, then there must exist a *tangible discriminant predicate*, D , such that D partitions τ into a non-empty set of two or more types, one of which includes σ . Hence, in accord with the intuition that the downward structure in the semilattice of types corresponds to the addition of more information to the concepts modelled, as we proceed down the structure we effectly are conjoining properties expressed by the discriminant D_i at that level. But not all discriminants can be represented as being of the same type, and we must be able to distinguish the way in which properties may predicate of individuals. Clearly the nature of predicate opposition is crucial to better understanding the different modes of predication. For example, the classic distinction between contradictories and contraries suggests at least two sorts of predicate opposition.

- (61) a. Bill is healthy
 a'. Bill is not healthy.
- b. Bill is sick.
 b'. Bill is not sick.

- (62) a. Jan is male.
 a'. Jan is not male.
- b. Jan is female.
 b'. Jan is not female.

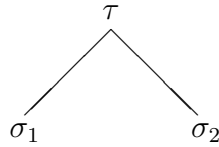
Sentences ((61)a,b), involving *polar opposites* such as *healthy/sick*, are typically viewed as contraries, while ((61)a,a') are contradictories. While contradictories ((61)a,a') and ((62)a,a') usually follow from an interpretation of *not* as “weak negation” (cf. von Wright, 1959, Horn, 1989), the contradictories present in ((62)a,b) and ((62)a',b') cannot be the result of weak negation alone: properties such as *male* and *female* are inherently contradictory when applied to its naturally predicated type, i.e., **animal**.

Rather than suggesting that predicate pairs such as *male/female* make reference to internal negation in their semantics, I will propose that the type structure which models such predicates is simply a Boolean semilattice. Similarly, although *healthy* and *sick* can be partially modelled in terms of negation, there is a stronger constraint that can be imposed on how polar opposites relate to one another through the semantics of a scalar semilattice with poles. It is the interpretation of the semilattice rather than the predicates themselves which makes reference to the operations of negation and ordering. Thus, discriminant properties (both logical and natural) are typed according to kind of opposition involved. Assuming some basic notion of a property sortal array and the type defining it, let us make the following assumptions:

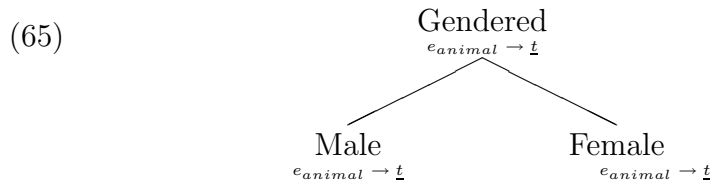
- (63) a. Property semilattice:
 b. $\langle \Sigma, \tau, \sqcup, \sqsubseteq \rangle$ realizes a predicate P , where Σ is a sortal array of types, τ is a local top type for this sortal array, such that $\sigma_i \in \Sigma$ for $\sigma_i \sqsubseteq \tau$.

Let us now define the simplest opposition as a 2-element property semilattice:

- (64) a. Binary Property:
 b. $\langle \sigma_1, \sigma_2, \tau, \sqcup, \sqsubseteq \rangle$ realizes a binary predicate P , where τ is a local top type for this sortal array, such that $\sigma_1, \sigma_2 \sqsubseteq \tau$.
 c.

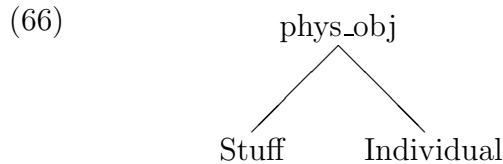


Examples of adjective pairs with this behavior include *male/female* (natural type), *married/unmarried* (functional type), and so forth.



BINARY OPPOSITION PREDICATE

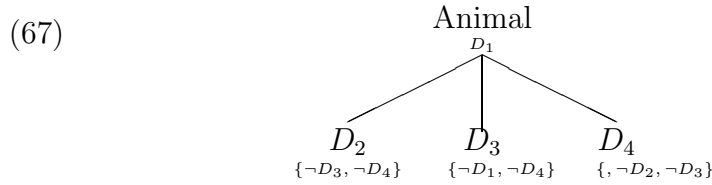
These are examples of natural discriminants, but L-discriminants will be classified by similar means. Foreexample, let us return to the discussion of generating the upper level of the natural type entity lattice. Assume the type `phys_obj` is partitioned by the L-discriminant, *individuated*, generating the structure in ((66))⁷.



In some cases, however, the partitioning of the natural types does not follow from the application of logical discriminants but from naturally identifiable properties associated with the members of a set; these may be natural or functional discriminants. In the construction of the natural types, they must

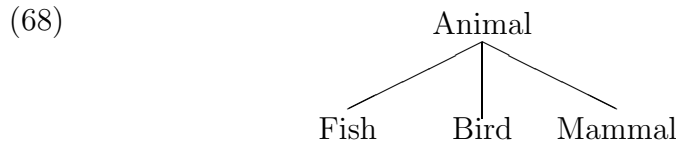
⁷The type `phys_obj` itself is the result of L-discriminants, but we will gloss over this for the current discussion

be naturals, however. For example, the sublattice associated with the type **animal** is arguably constructed according to natural discriminants, such as *flying* (D_3), *walking* (D_4), and *swimming* (D_2), as informally illustrated below in ((67)):



Animal WITH DISCRIMINANTS

I will assume that there are constraints on the possible combinations of discriminants defining a partitioning, but I will not address these issues here. In the above structure, a legitimate subtype is constructed from those expressions with only one positive discriminant value for the three predicates in question. This generates a shallow structure for animal types as shown below.



LATTICE WITH \mathcal{N} -TOPLET Animal

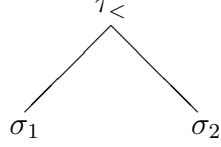
It should be noted that, in general, the discriminants used to create the natural types are both logical and natural predicates.

In addition to the types of discriminant predicates encountered so far, it is possible to justify several other classes of opposition structures. One of the most prevalent in natural language lexicons is the privative/non-privative opposition pairing. Examples of adjectives with this behavior include fully lexicalized pairs such as *dead/alive*, as well as privative anchors with a lexical gap, such as *bald*. The type structure accounting for this ordering is given as follows:

(69) a. Binary Ordered Property:

- b. $\langle \sigma_1, \sigma_2, \tau, \sqsubseteq, <, \sqsubseteq \rangle$ realizes a binary predicate P , where τ is a local top type for this sortal array, such that $\sigma_1, \sigma_2 \sqsubseteq \tau$, and $\sigma_1 < \sigma_2$.

c.

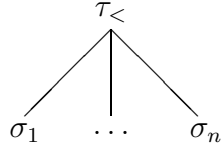


The nature of polar attributes such as *sick/healthy* and *tall/short* is defined in terms of a sortal array with distinguished elements.

(70) a. Polar Property:

- b. $\langle \Sigma, \tau, \sqsubseteq, <, \sqsubseteq \rangle$ realizes a predicate P , where Σ is a sortal array of types, τ is a local top type for this sortal array, such that $\sigma_1, \dots, \sigma_n \in \Sigma$ for $\sigma_i \sqsubseteq \tau$, and $\sigma_i < \sigma_{i+1}$, and there are two poles, σ_1 , and σ_n , that are distinguished sorts.

c.

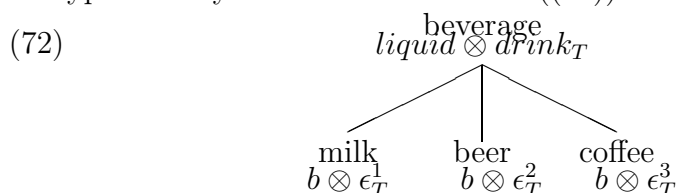


Thus far I have focused on the construction of types within the naturals. Let us now turn briefly to a discussion on subtyping relations in the domain of functional types, \mathcal{F} . Unlike the natural types, where subtypes are defined in terms of natural tangible discriminants, subtyping in \mathcal{F} operates in an entirely different manner, namely, in terms of functional behavior. Thus, the domain of \mathcal{F} will appear to be socially arbitrary and less consistent cross-linguistically, because the lexicalization illustrates socially dependent concepts rather than the more universally grounded discriminants of the naturals.

As a case in point, consider the semilattice associated with the functional concept **beverage**. The conventional qualia structure for **beverage** is shown in ((71)), while the corresponding type structure is $liq \otimes drink_T$.

$$(71) \quad \left[\begin{array}{l} \mathbf{beverage} \\ \text{ARGSTR} : \left[\text{ARG1} : \mathbf{x} : \mathbf{liquid} \right] \\ \text{QUALIA} : \left[\begin{array}{l} \text{FORMAL} : \mathbf{x} \\ \text{TELIC} : \mathbf{drink}(e^P, \mathbf{y}, \mathbf{x}) \end{array} \right] \end{array} \right]$$

The subtypes of beverages, as exemplified in our language lexicalizations, are classified according to functional principles (i.e., the TELIC) rather than natural discriminants, such as with animals or trees. Hence, we have drinks that wake us up, drinks that refresh us, drinks that relax us, and so forth. The operative discriminants are all functionally defined, however. As a lattice structure, the ground type (the natural base) for **beverage** is **liquid**, and the tensor type value for the TELIC is specialized according to the particular subtype activity. This is illustrated in ((72)) below.



LATTICE WITH \mathcal{F} -TOPLET beverage

The first thing to note about this structure is that the functional discriminants can be *uniquely* determined for the subtypes of the functional type. For example, a beverage might be defined in terms of its use in a religious ceremony (e.g., *altar wine*). The complement of this discriminant is not particularly informative for the other sorts in the array. The second point to make regarding functional subtyping in \mathcal{F} is that in many cases there is no justifiable natural subtyping relation between two types. Thus, materially, a table is a table, but the functional specification that determines that a work table is different from a kitchen table is not necessarily part of the discriminant family in \mathcal{N} .

Another fact illustrated by the structure in ((72)) is that functionally defined objects have multiple TELIC roles through inheritance, where some values are more proximate than others. In the case of *coffee*, for example, it would appear to be the following:

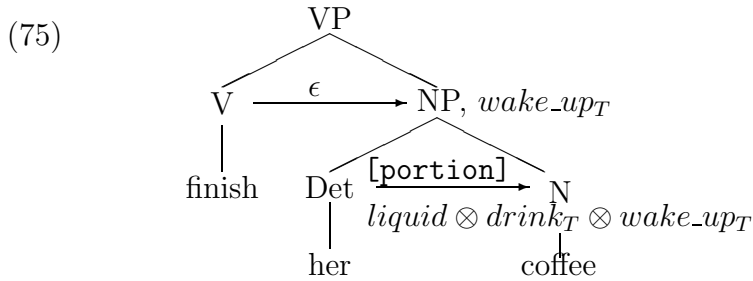
(73) $\text{liquid} \otimes \text{drink}_T \otimes \epsilon_T^3$

where the “discriminating” quale value (relative to **beverage**) is that it is used for keeping one alert. However this cluster of properties is to be described—here it is modelled as controlled event descriptions—this is a more proximate description of the drink than that associated with beverages in general.

Although this functional type description for the types under **beverage** seems motivated, this appears to have some problematic consequences for how coercion exploits the events associated with the objects, as shown in the sentence below.

(74) John finished the coffee.

If coercion is defined as selecting the outermost type of the object by the governing verb, then we have a puzzle, since the proximate TELIC of *coffee* is actually defined relative to how it is distinguished from the other beverages. But, this gives the wrong results, as the derivation below illustrates.



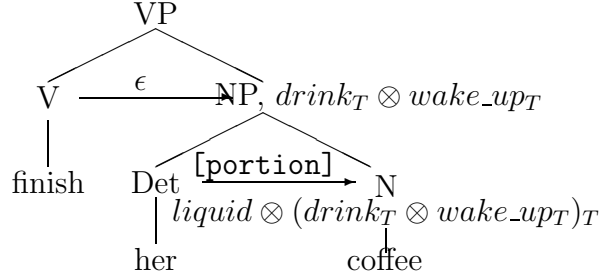
This would predict that the interpretation of the sentence is “finishing (the effects of) the coffee”, which is obviously incorrect. What is clearly going on is an embedding or recursive structure with the type structure, where the TELIC of TELIC is referenced. As a qualia structure, this would correspond to something like ((76)), where the inherited TELIC is embedded as the AGENTIVE of the proximate TELIC, and whose TELIC value is “waking up.”

(76)

$$\left[\begin{array}{l} \mathbf{coffee} \\ \text{ARGSTR} : \left[\text{ARG1} : \mathbf{x}:\mathbf{liquid} \right] \\ \text{QUALIA} : \left[\begin{array}{l} \text{FORMAL} : \mathbf{x} \\ \text{TELIC} : \left[\begin{array}{l} \text{TELIC} : \mathbf{wake_up}(e^T, \mathbf{y}) \\ \text{AGENTIVE} : \mathbf{drink}(e^P, \mathbf{y}, \mathbf{x}) \end{array} \right] \end{array} \right] \end{array} \right] \end{array} \right]$$

Abstracted as a type structure, this would be simply $liquid \otimes (drink_T \otimes wake-up_T)$. Given these assumptions, the coercion will now select the appropriate TELIC value, as illustrated in ((77)) below.

(77)



8 The Conceptual Architecture

In this paper I have presented a new architecture for a type structure corresponding to the concepts of natural language expressions. First, I abstracted away from the conventional type feature structure interpretation of qualia structure to present concepts as ground types and type constructions. The qualia structure of ((78)) becomes a typing specification as shown in ((79)).

$$(78) \quad \left[\begin{array}{c} x \\ \text{QUALIA} = \left[\begin{array}{l} \text{CONST} : \tau \\ \text{FORMAL} : \alpha \\ \text{TELIC} : \epsilon_T \\ \text{AGENTIVE} : \epsilon_A \end{array} \right] \end{array} \right]$$

$$(79) \quad x : \left[\begin{array}{c} \alpha \\ \text{QUALIA} : \left[\begin{array}{l} \text{C} : \tau \\ \otimes : \epsilon_T \\ \otimes : \epsilon_A \end{array} \right] \end{array} \right]$$

Furthermore, I have motivated a three-way distinction of increasingly complex types (or ranks), as summarized below:

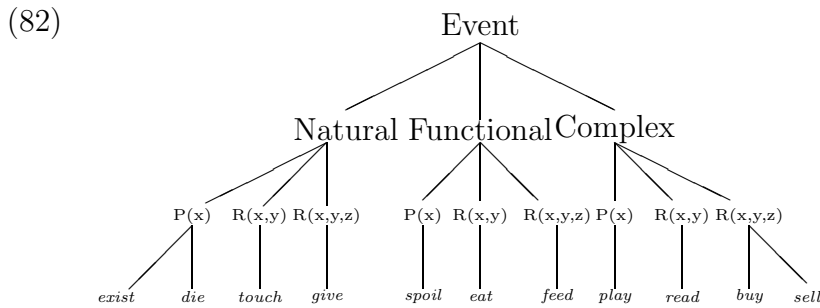
- (a) NATURAL TYPES: Predication from the domain of substance, e.g., the qualia FORMAL or CONST.
- (b) FUNCTIONAL TYPES: Predication includes reference to either AGENTIVE or TELIC qualia.
- (c) COMPLEX TYPES: Cartesian type formed by Dot Object Construction.

Similarly, the domains of relations and properties are also partitioned into three ranks:

- (a) NATURAL EVENTS: Arguments in the predicate or relation are only from the domain of substance, e.g., the qualia FORMAL or CONST.

- (b) FUNCTIONAL EVENTS: At least one argument in the predicate or relation is a functional type, f , e.g., makes reference to either AGENTIVE or TELIC qualia.
- (c) COMPLEX EVENTS: At least one argument in the predicate or relation is a complex type, e.g., a type formed by Dot Object Construction.

The resulting top structure of the types denoting events is shown below:⁸



EVENT/RELATION LATTICE

⁸Following Pustejovsky (1998b), I assume that there is an order of predicates, given as 1-place, 2-place, and 3-place relations. Furthermore, I assume that for 1-place predicates, the constraints on sortal domains discussed above apply. For 2-place predicates, the following *Relational Class Constraints* apply:

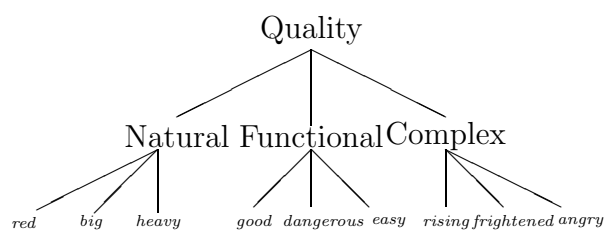
- (80) a. Constraints on 2-place Relation, R_2 :
- b. Reflexivity:
 - c. Symmetry:
 - d. Antireflexivity:
 - e. Antisymmetry:

For 3-place relations, we have the following constraints operative:

- (81) a. Constraints on 3-place Relation R_3 :
- b. Partial Pairwise Reflexivity:
 - c. Partial Pairwise Symmetry:
 - d. Partial Pairwise Antireflexivity:
 - e. Partial Pairwise Antisymmetry:

Although we have yet to discuss the typing of attributive properties, a few remarks are in order. Just as with the distinctions seen in entities and relations, attributes can be usefully distinguished as ordered according to the same principles, as suggested in the type structure below.

(83)

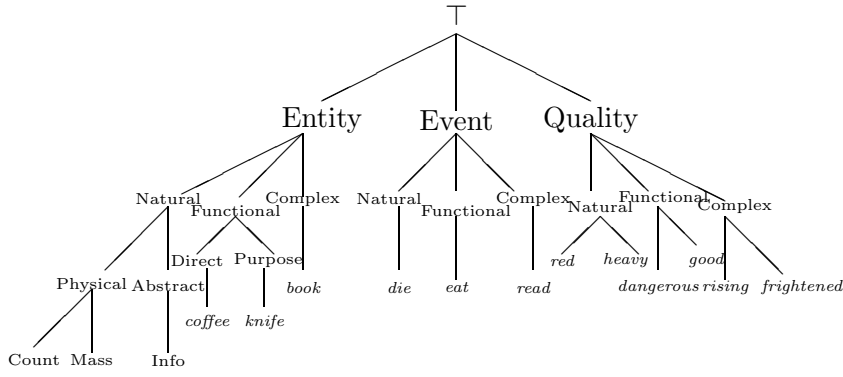


PROPERTY LATTICE

The same patterns of entailment hold for judgments involving the different modes of attribution as with the relations. These phenomena are discussed further in Pustejovsky (forthcoming).

Given what we have presented, the overall architecture of the type system is summarized in the following diagram.

(84)



Tripartite Concept Lattice

In this paper I have outlined a methodology for the construction of ontological types based on the dual concerns of capturing linguistic generalizations and satisfying metaphysical considerations. I have tried to show that the flat and homogeneous formal typing models are too weak to model and describe language and its meaning. To fill this gap, I have presented extensions to Generative Lexicon Theory employing a ranking of types, distinguishing between natural types and functional types, and then complex types. I concluded with a discussion of how the different ranks of types have distinct operations defining the type-subtype relation. This is important for lexicon design, since it shows how conventional models of type structures are too homogeneous in structure for capturing the semantic richness behind natural language expressions.

Appendix: Interpreting the Qualia

The remarks below will hopefully clarify some of the formal aspects of the qualia structure as presented in the discussion above. These explanations are adapted from Pustejovsky (1998).

Assuming that the FORMAL as defined above has a natural interpretation, we begin with a suggestive translation of the constitutive mode, CONST:

$$(85) \text{ Const}(\lambda x[\alpha(x)]) = \lambda y[Q(y)] \leftrightarrow \forall x[\alpha(x) \rightarrow \exists y[Q(y) \wedge \text{made_of}(x, y)]]$$

$$(86) \text{ a. } \left[\begin{array}{l} \mathbf{snowball} \\ \text{ARGSTR} : \left[\begin{array}{l} \text{ARG1} : x \\ \text{D-ARG1} : y \end{array} \right] \\ \text{QUALIA} : \left[\begin{array}{l} \text{FORMAL} : \text{ball}(x) \\ \text{CONST} : \text{snow}(y) \end{array} \right] \end{array} \right] \Rightarrow$$

$$\text{b. } \lambda x[\text{ball}(x) \wedge \text{const}(x) = \lambda y[\text{snow}(y)]]$$

$$(87) \text{ Formal}(\lambda x[\alpha(x)]) = \lambda x \exists e^S [Q(x) \wedge E(e^S, x)] \leftrightarrow \alpha \subseteq Q$$

The AGENTIVE quale makes reference to the mode of coming into being:

$$(88) \quad \forall x, e[\alpha(e, x) \rightarrow \exists e' \exists y[\psi(e') \wedge e' \prec e \wedge \text{make}(e', y, x)]]$$

$$(89) \quad \text{a.} \quad \left[\begin{array}{l} \mathbf{wine} \\ \text{ARGSTR} : \left[\begin{array}{l} \text{ARG1} : x \\ \text{D-ARG1} : y \end{array} \right] \\ \text{EVENTSTR} : \left[\begin{array}{l} \text{D-E1} : e \end{array} \right] \\ \text{QUALIA} : \left[\begin{array}{l} \text{FORMAL} : \text{liquid}(x) \\ \text{AGENTIVE} : \text{make}(e, y, x) \end{array} \right] \end{array} \right] \Longrightarrow$$

$$\text{b.} \quad \lambda x[\text{wine}(x) : \text{formal}(x) = \lambda x[\text{liquid}](x) \wedge \text{agentive}(x) = \lambda e[\psi(e)]]$$

$$(90) \quad \text{Telic}(\lambda x[\alpha(x)]) = \lambda y \lambda e \exists x[\phi(e, y, x)] \leftrightarrow \\ \lambda y \forall x \forall e \forall y[\psi_\alpha(e, y, x) > \exists e'[(\phi(e', y, x) \wedge e < e')]]$$

In this representation of TELIC, ψ_α denotes the appropriate circumstances of doing something to an x of type α , and $>$ is again the default conditional of Asher and Morreau (1991).

Applying this interpretation to the artifactual liquid *wine* gives the following expanded interpretation:

$$(91) \quad \lambda x \exists e[\text{wine}(e, x) : \text{formal}(x) = \lambda x[\text{liquid}(x)] \\ \wedge \text{agentive}(x) = \lambda e[\psi(e)] \\ \wedge \text{telic}(x) = \lambda e \lambda y[\text{drink}(e, y, x)]]$$

Then, with the full definitions of TELIC and AGENTIVE substituted, this expression becomes:

$$(92) \quad \lambda x \exists e', e, y[\text{wine}(e, x) : \text{formal}(x) = \text{liquid}(x) \\ \wedge \text{agentive}(x) = [\psi(e') \wedge e' \prec e \wedge \text{make}(e', y, x)] \\ \wedge \text{telic}(x) = \lambda y \forall e[\psi_{\text{wine}}(e, y, x) > \exists e'[(\text{drink}(e', y, x)]]$$

This expression now captures our intuitions about a qualia-based interpretation of word meanings; in this case, that wine is a particular liquid made for the purpose of drinking.

Acknowledgments

This paper is a revision of “Specification of a Top Concept Lattice: Version 1.0”, a manuscript from April, 1998, where the basic proposal for a tripartite type structure was proposed. Subsequent work on the SIMPLE model

(Busa et al, 1999) implemented and greatly expanded on this proposal and contributed significantly to pointing out several gaps in the original model. The present paper has profited greatly from the work done in the SIMPLE group, as well as from other colleagues. In particular, I would like to thank Nicholas Asher, Federica Busa, Jose Castano, and Bob Ingria for useful comments and suggestions to this paper. All remaining inconsistencies are my own. This grant was partially supported by NIH Grant DC03660.

References

- Anscombe, G.E.M. (1958) “On Brute Facts,” *Analysis* 18, no. 3.
- Asher, N. and Lascarides, A. (1993). Temporal Interpretation, Discourse Relations and Commonsense Entailment. *Linguistics and Philosophy* 16, 437–494.
- Asher, N. and Pustejovsky, J. (1999). The Metaphysics of Words. ms. Brandeis University and University of Texas.
- Bach, E. (1986) “The Algebra of Events,” *Linguistics and Philosophy* 9:5-16.
- Bierwisch, M. (1983). Semantische und konzeptuelle Repräsentationen lexikalischer Einheiten. In R. Ruzicka and W. Motsch (eds.), *Untersuchungen zur Semantik*, Berlin: Akademische-Verlag.
- Bouillon, P. (1997). *Polymorphie et sémantique lexicale : le cas des adjectifs*, Lille: Presses Universitaires du Spentrion.
- Bouillon, P. and Busa, F. (forthcoming). Where’s the polysemy? A study of adjective-noun constructions. *Proceedings of the Second Workshop on Lexical Semantics Systems*, Pisa, Italy.
- Busa, F. (1996). *Compositionality and the Semantics of Nominals*, Ph.D. Dissertation, Brandeis University.
- Busa, F., Calzolari, N., Lenci, A. and Pustejovsky, J. (1999). Building a Semantic Lexicon: Structuring and Generating Concepts. *Proceedings of IWCS-III*, Tilberg, The Netherlands.

- Calzolari, N. (1991). Acquiring and Representing Semantic Information in a Lexical Knowledge Base. In J. Pustejovsky (ed.), *Proceedings of the Workshop on Lexical Semantics and Knowledge Representation*, Berkeley, CA.
- Carnap, Rudolf. (1928) *The Logical Structure of the World*, University of Chicago Press.
- Chomsky, N. (1995). Language and Nature. *Mind* 104.
- Copestake, A. (1993). Defaults in the LKB. In T. Briscoe and A. Copestake (Eds.), *Default Inheritance in the Lexicon*, Cambridge: Cambridge University Press.
- Copestake, A. and Briscoe, E. (1992). Lexical Operations in a Unification-Based Framework. In J. Pustejovsky and S. Bergler (Eds.), *Lexical Semantics and Knowledge Representation*, New York: Springer Verlag.
- Dowty, D. R. (1979). *Word Meaning and Montague Grammar*, Dordrecht: Kluwer Academic Publishers
- Fellbaum, C., (editor) (1998) *Wordnet : An Electronic Lexical Database*, MIT Press.
- Fillmore, C.J. (1985) Syntactic Intrusions and the Notion of Grammatical Construction. *BLS* 11, 73–86.
- Fodor, J. and Lepore, E. (1996) “The Red Herring and the Pet fish: why concepts still can’t be Prototypes,” *Cognition* 58.
- Fodor, J. and Lepore, E. (1998). The Emptiness of the Lexicon: Critical Reflections on J. Pustejovsky’s *The Generative Lexicon*. *Linguistic Inquiry* (This Issue).
- Guarino, N. (1998). “Some Ontological Principles for Designing Upper Level Lexical Resources,” *Proceedings of the First International Conference on Language Resources and Evaluation*, Granada, Spain.
- Jackendoff, R. (1992). *Semantic Structures*, Cambridge, MA: MIT Press.

- Horn, L. R. 1989. *A Natural History of Negation*, University of Chicago Press, Chicago.
- Jackendoff, R. (1985). Multiple Subcategorization and the Theta-Criterion: The Case of *Climb*". *Natural Language and Linguistic Theory* 3, 271–295.
- Johansson, I. (1989) *Ontological Investigations*, Routledge.
- Kayser, D. (1988). What Kind of Thing is a Concept?. *Computational Intelligence* 4, 158–165.
- Levin, B., (1993) *Towards a Lexical Organization of English Verbs*, Chicago: University of Chicago Press.
- Levin, B. and Rappaport, M. (1995). *Unaccusatives: At the Syntax-Lexical Semantics Interface*, Cambridge: MIT Press.
- Link, G. (1998) *Algebraic Semantics in Language and Philosophy*, CSLI, Cambridge University Press.
- Moravcsik, J. M. (1975). Aitia as Generative Factor in Aristotle’s Philosophy. *Dialogue* 14, 622–36.
- Moravcsik, J. (1998). *Meaning, Creativity, and the Partial Inscrutability of the Human Mind*, Stanford: CSLI.
- Nirenburg, S. and Raskin, V. (1998) “Ten Criteria for Lexical Semantics”, ms. New Mexico State University.
- Nunberg, G. (1979). The Non-uniqueness of Semantic Solutions: Polysemy. *Linguistics and Philosophy* 3, 143–184.
- Partee, B. and Rooth, M. (1983). Generalized Conjunction and Type Ambiguity. In Bäuerle, Schwarze, and von Stechow (eds), *Meaning, Use, and Interpretation of Language*, Walter de Gruyter.
- Pustejovsky, J. (1995). *The Generative Lexicon*, Cambridge, MA: MIT Press.

- Pustejovsky, J. (1998a). Generativity and Explanation in Semantics. *Linguistic Inquiry*.
- Pustejovsky, J. (1998b). Specification of the Top Lattice. ms Brandeis University.
- Pustejovsky, J. (1999). The Semantics of Lexical Underspecification. *Folia Linguistica* XXXII.
- Pustejovsky, J. (forthcoming). *Language Meaning and The Logic of Concepts*, MIT Press.
- Searle, J. (1995) *The Construction of Social Reality*, Free Press.
- Simons, Peter. (1987) *Parts: A Study in Ontology*, Oxford, Clarendon Press.
- Smith, Barry (ed.) (1982) *Parts and Moments: Studies in Logic and Formal Ontology*, München, Philosophia.