Concepts, Definitions, and Inheritance

Interpreting the atoms of lexical decomposition

by

DAVID ZORNEK

Chapter 1: Introduction

Abstract

In theories of lexical decomposition, word meaning is analyzed as composed from some atomic units of semantic content. Most theorists seem to intend for their atoms to be interpreted as concepts, but none have provided a framework to fully develop this interpretation; it is assumed and used, but not justified. All theories of lexical decomposition organize semantic atoms in an inheritance hierarchy. In this thesis, I build on the mathematics of inheritance networks to develop a framework that is used to prove and analyze an isomorphism between semantic atoms and empirical observations of the hierarchical organization of concepts. The framework is presented not only as a means for justifying the conceptual interpretation, but also as a set of formal tools that might be used to facilitate future analysis in the domains of formal linguistics and cognitive science.

A theory of lexical decomposition is a semantic theory that analyzes word meaning as composed from some basic units of meaning, or *atoms*, that are taken as primitive to the theory. Although decomposition is not without its opponents, notably Fodor and Lepore [16] [18] [17] [19], it remains one of the most popular (if not *the* most popular) approaches to lexical semantics. However, theories of lexical decomposition cannot do all of the work we require from a total theory of lexical semantics, or word meaning. David Lewis has observed that sentential decomposition cannot account for semantic content unless we augment the decompositional theory with some story about how atoms hook up with the world [27]. The same can be said for lexical decomposition. Despite this limitation of decompositional theories, there is an overwhelming consensus among cognitive scientists that decompositional models accurately reflect our cognitive representations of meaning, as Fodor and Lepore have themselves observed [19].[1] Given this

[1]Fodor and Lepore express puzzlement at this consensus, on the grounds that "as far as [they] can tell, there is practically no evidence to support it" [19, pg. 1]. However, they imply that they will only accept as evidence an argument that it is impossible to represent meaning in any manner other than decompositionally: "For example, there is no scientific evidence that you *can't* have a word that expresses the concept BREAK_{TR} unless you

consensus, we should not hastily abandon lexical decomposition, but should instead look for some interpretation of the atoms that will fortify theories of lexical decomposition by providing them with a source of semantic content. The aim of this thesis to interpret semantic atoms as concepts.

The present chapter is introductory. I will set some expectations for what will be found later in the thesis, explain some preliminary ideas that will be used throughout the thesis, and define some basic notation. Much of this content will be repeated later.

Chapter 2 will provide suitable background material to understand the relevant aspects of theories of lexical decomposition. In Section 1, Chapter 2, I will survey theories of lexical decomposition authored by Ray Jackendoff [24] and Beth Levin & Malka Rappaport Hovav [25], in order to identify semantic atoms as a formal representation of *semantic content* and *inheritance networks* as a set of formal relations that is used to model relations between semantic atoms. By *semantic content*, I mean the *informational* part of semantics, which includes not only reference, but other information which native speakers of a language will tend to associate with reference. For example, the atom **human** refers to humans, so these are included in the content provided by **human**. Native speakers of English will tend to recognize that humans reproduce sexually, walk on two legs, build skyscrapers, cook dinner, and use language. While "human" does not refer to reproducing sexually, etc., this is all part of the semantic information provided by the atom **human**. *Semantic content* is distinguished from semantic structure, which is the way in which we situate atoms within the total semantics of a word. For example, we might wish to distinguish between essential and non-essential properties of humans.[2] Labelling some properties as essential

have the concept CAUSE. But there ought to be if CAUSE is a constituent of BREAK_{TR}" [19, pg. 1, emphasis added]. But this places too high a standard of justification for the empirical claims that cognitive scientists make; they require only that we *do* represent meaning decompositionally, not that we do so *necessarily*.

[2]Theories that organize semantic content in this way are called *essentialist* theories. Essentialist theories are outside the scope of this paper and are only mentioned here as a way of grasping the difference between semantic content and semantic structure. The structural aspect of essentialism is much easier to intuit by means of a short explanation than the structures that will be seen later.

is an aspect of semantic structure, not semantic content.

An *inheritance network* is an abstract system in which mathematical *types* are organized by the reflexive, transitive inheritance relation. A *type* is a purely abstract mathematical object, studied by *type theory*, that is in some ways analogous to the sets studied by set theory. Where sets have members, types have tokens, but tokens are not *in* the type in the way that members are in sets. A token of a type has constraints placed on it by the type without being *in* the type. In fact, since types are entirely basic, it doesn't make sense to talk about things being *in* a type at all. We will largely be unconcerned with the mathematics of type theory in this thesis. In all of the contexts that types will be used here, they are taken as models of semantic atoms, so for practical purposes, the terms will be somewhat interchangeable in that they have the same extension, but different intensions. "Type" will always refer to semantic atoms, but I will refer to atoms as "types" when taking a mathematical perspective on the atoms. The distinction is subtle and can be ignored by most readers.

Chapter 2 will close by giving a slightly more detailed presentation (in comparison to the presentation of other theories of lexical decomposition) of James Pustejovsky's *Generative Lexicon* [35]. Pustejovsky's system will later serve as a case study in how the conceptual interpretation of semantic atoms can be applied to practical questions.

Having established that inheritance networks appear in all theories of lexical decomposition, in Chapter 3, I will turn toward empirical research on concepts. This chapter will be a somewhat historical story about the downfall of the *classical view of concepts*. I will identify two claims made by the classical view, one pertaining to the intrinsic content of concepts, i.e. that definitions provide necessary and sufficient conditions for category membership, and the other pertaining to the relations that hold between concepts. i.e. that concepts are organized hierarchically. It will be argued that philosophers and cognitive scientists, at some point, lost sight of the relational claim, so that when empirical research conclusively established that the intrinsic claim is false, the classical view came to be almost universally rejected. However, as will be seen, the same research that establishes that

category membership is not determined by necessary and sufficient conditions also establishes that concepts are organized hierarcically, so there remains an aspect of the classical view that has not been overturned. Both of these empirical findings will serve to advance the conceptual interpretation of semantic atoms. Cognitive scientists have found that category membership is determined by overall similarity, or *typicality*, and it will turn out that inheritance is a typicality relation. The inheritance networks of decompositional theories are simply a way of hierarchically organizing semantic atoms according to the inheritance relation.

Chapter 4 comprises the main theoretical material of the thesis. I will give formal definitions of *inheritance*, *inheritance network*, *consistent definition*, and *functional role*. Inheritance and inheritance networks have already been mentioned, but a few words should be said here about the other terms, which are original to this thesis. *Consistent definition* is an abstract set-theoretic structure defined over the atoms included in an inheritance network, which serves as a mathematical model of the genus-differentia definitions in the classical view of concepts. The *functional role* of an atom is the set of all consistent definitions containing the atom. In the climax, a theorem will then be presented[3] stating that atoms in an inheritance network are isomorphic to functional roles. An *isomorphism* is a relation-preserving bijection between two domains. The relations preserved in this isomorphism are the type-theoretic inheritance relation and the subset relation between functional roles. Less formally, some philosophical consequences of the theorem will be explained.

In Chapter 5, I will apply the mathematical framework of Chapter 4 to James Pustejovsky's *Generative Lexicon*, in order to illustrate why an isormophism is useful for understanding concepts. Some theoretical questions about identity between concepts will be raised, and a diagnostic for nonidentity will be introduced and applied to answer these questions. Questions about the practical consequences of the theorem for lexical decomposition will be raised, but an answer will only be given in the broadest strokes; a

[3]The proof of the theorem is not included in Chapter 4, but in an appendix.

detailed answer to these questions is outside the scope of this thesis.

In Chapter 6, I will summarize the results of the thesis and point toward some outstanding questions and directions for future research.

Before continuing, it will be useful to define some notation that will be used throughout the thesis. **Boldface type** will be used for types, semantic atoms, and concepts; since the point of this thesis is that, in the context of a theory of lexical decomposition, types *are* semantic atoms and semantic atoms *are* concepts, I annotate them all in the same way. Words will be mentioned using the standard convention of quotes, e.g. "begin" is a verb. The inheritance relation will be represented by the symbol $\sqsubseteq$. Another relation—*disjointness*—is included in the inheritance networks studied here, but is not presented as a direct model of conceptual hierarchy. Disjointness is necessary for establishing the theorem of Chapter 4, and it does reflect other properties of semantic content. Disjointness will be represented by the symbol #. The notation $\alpha : \beta$ will indicate that α is of type β; when α is a type, this notation is equivalent to $\alpha \sqsubseteq \beta$, with the difference being that α need not be a type in order for the sentence $\alpha : \beta$ to be well-formed. The usual entailment symbol $\vdash$ is used in this thesis to mean specifically entailment within the logic of type-inheritance, which is a very different system from the more-familiar first-order logic. The details of this logic are not important for the purposes of this thesis, but I will at times refer to entailment relations that hold within the logic of type-inheritance. Graphically, inheritance will be represented by an arrow, with $\alpha \rightarrow \beta$ equivalent to $\alpha \sqsubseteq \beta$; disjointness is graphically represented by a dashed line. Graphs of inheritance networks will sometimes be color-coded, and a code will be defined in the text where one is used.

Chapter 2: Theories of Lexical Decomposition

In this chapter, we will identify two properties that are shared by all decompositional theories, one essential and one (seemingly) accidental. It is essential to a theory of lexical decomposition that it includes some sort of semantic *atom* in its ontology; decomposition simply *is* breaking semantic content down into atoms. Although not essential to decomposition per se, we will observe that the atoms of all decompositional theories are related to each other by *inheritance*. Inheritance is a reflexive and transitive relation in which information is said to be *inherited* by one atom from another. If an atom α inherits from an atom β, information obtained about β will generally apply to α. The interpretation of atoms offered in this thesis will explain why inheritance appears in decomposition so consistently, turning this accidental property into an essential one.

To begin, we will look at some examples of lexical decomposition, highlighting those aspects of each theory that are relevant to the present discussion. Each theory is formulated within its own framework, based on whatever set of foundational assumptions are held by each theorist: Jackendoff models internal representations of meaning, as a total theory of semantics, while Levin and Rappaport Hovav model the semantic features of verbs that have consequences for syntax. James Pustejovsky's *Generative Lexicon* (GL) seeks to improve on traditional lexicons by decomposing words in a way that allows new word senses to be generated, rather than requiring all word senses to be stored in the lexicon as independent entries. We will not here be concerned with these foundational differences; regardless of which approach to lexical decomposition we look at, the general *method* of analysis—the method of composing meaning out of atoms, by means of some structural rules of composition—is the same. Our primary concern here is to understand this general method, rather than the nitty-gritty details of each individual theory.

1 Formal Theories of Lexical Decomposition

1.1 Jackendoff's cognitive semantics

The meaning of a lexical item (or a portion of its meaning, at least) is given in a *feature structure* in which types are assigned to different "features" or aspects of the word's meaning. The basic form of one of Jackendoff's feature structures is:

$$\begin{bmatrix} \textbf{event, thing, place,} \ldots \\ \text{token/type} \\ F(\langle \text{Entity}_1, \langle \text{Entity}_2, \langle \text{Entity}_3 \rangle \rangle \rangle) \end{bmatrix}$$

The top feature in the structure specifies an *ontological category* under which the lexical item falls. Ontological categories provide a partition of the ways in which our cognitive architecture allows us to represent word meaning. The middle feature is an immediate *inheritance relation* (explained below) in which the word participates (normally by specifying a type from which the word's meaning inherits), and the third and final feature is the argument[4] structure of the word, i.e. the types of arguments that the word can take, which will change depending on the sense in which the word is being used. For example, take the following typed feature strucutures for "novel" and "begin":

$$\textbf{novel} \begin{bmatrix} \textbf{thing} \\ \textbf{book} \\ F(\langle \textbf{property} \rangle) \end{bmatrix}$$

[4]Jackendoff uses the term "argument" here in a non-linguistic sense, i.e. it does not mean "grammatical argument." He does little to explain what other sense of "argument" he is using, but since everything he calls an argument is an input to the function F, I read him as using "argument" in the sense of arguments of a function. F is a *cognitive mapping* from properties onto meanings, i.e. concepts, which is inherent in biology of human cognition. In the case of verbs, the functional arguments turn out to be the grammatical arguments.

$$\textbf{begin}\left[\begin{array}{l}\textbf{event}\\ \textbf{transition}\\ F(\langle\textbf{animate\_object},\langle\textbf{event}\rangle\rangle)\end{array}\right]$$

"Begin" takes a primary argument of type **animate_object** (since only animate objects can begin actions) and a secondary argument of type **event** (since only events can be begun).[5] "Novel" might take arguments of type **property**, as it is used in the sentence "This novel is long"; but, in the sentence "The novel is on the table," the argument type is **place**. These are differences between senses of "novel," which are largely determined by the syntactic structure of the sentence and a grammatical conceptual structure inherent in human cognition. The details of how this works are interesting, but not relevant to the present project.

The ontological category and immediate inheritance relation, however, *are* relevant to the present project, so it is useful at this point to explain them.

Jackendoff's types are organized according to the *inheritance relation* $\sqsubseteq$.[6] Inheritance is reflexive and transitive, but not symmetric. Given two atoms α and β such that $\alpha \sqsubseteq \beta$, we say that α inherits from β. Or, α is a *subtype* of β. If there is a third atom γ such that $\beta \sqsubseteq \gamma$, then what the inheritance relation tells us is that $\alpha \sqsubseteq \gamma$ as well. If we let $\mathbf{x}$ be a member of any domain to which the types can apply, we write $\mathbf{x}{:}\alpha$ to indicate that $\mathbf{x}$ is of type α. By inheritance, $\mathbf{x}{:}\alpha \vdash \mathbf{x}{:}\beta \vdash \mathbf{x}{:}\gamma$.

The broadest type, of which all other types are subtypes is the type **entity**. Beneath **entity**, there is a set of *ontological categories*, none of which inherit from each other, but all of which inherit from **entity**. These are **thing**, **event**, **state**, **action**, **place**, **path**, **property**, and **amount**.

[5] We might quibble over whether these type assignments are the *right* ones for "begin." It's not important here whether I accurately convey the specific types involved in the semantics of "begin," and these will suffice at least to get a handle on how Jackendoff's feature structures are supposed to work, which is the real goal here.

[6] Jackendoff does not call the relation inheritance; I have modified his terminology in order to make my own terminology consistent throughout this paper. He calls the relation an "IS-A" relation or a "type-token" relation, both of which are fairly common alternatives.

This sort of system might remind the reader of Aristotle's *Categories*, which might be regarded as the earliest example of a type-inheritance theory of lexical semantics. Every lexical item will fall under one of the ontological categories.

At first glance, Ray Jackendoff seems to advocate a version of my own view. In [24], he argues that word meaning is decomposed into concepts, but his usage of the word "concept" is broader than mine. For Jackendoff, a concept is any mental representation, and it includes subjective representations of categories (which is, more or less, what cognitive scientists mean when they talk about concepts), subjective representations of propositions, mental representations of individual objects, and perhaps even the cognitive structural rules which Jackendoff believes to be the source of basic grammatical structure for natural languages.

Jackendoff's notion of concept is too broad to do the work that will be required here. The connection I draw between semantic and conceptual content will rely on empirical results pertaining to the hierarchical organization of concepts (in the cognitive scientist's sense). These results have not been seen for other mental representations.

Moreover, his notion of concept is entirely subjective; that is, Jackendoff's concepts explicitly have no important connection to the external world. This is largely due to his view on the nature of mind, which is little more than a more readable rendition of Kant's *Transcendental Aesthetic* [24, Ch. 2]. Our faculty of sensation is somehow roused into action by external stimuli. Sensations are used as a kind of paint by the mind, which creates a subjective picture or "projected world" (cf. Kant's phenomenal world) by applying the paint in accordance with concepts, which act as a kind of blueprint. It is concepts, not the external world, that determines what we see in the projected world. And the projected world is *all* we see, all our words can possibly refer to. One of our goals here is to explain how words hook up with the external world, but Jackendoff (and Kant) have reinterpreted the word "world" in such a way that it becomes non-sensical to talk about the external world at all.

I have serious reservations about whether Jackendoff should be allowed

to reinterpet "world" in this way, but arguing this point belongs to an entirely different project. For the present purpose, it is taken as an explicit assumption that our words do connect with the external world in some way. Under this assumption, we then ask the question "How do words connect to the external world?" Since Jackendoff has explicitly denied that words (or concepts) have any important relation to the external world, making this assumption places us outside the scope of Jackendoff's framework.

Nevertheless, Jackendoff's theory is decompositional and therefore exhibits the same features of other decompositional theories that will be important here. Where decomposition is concerned, it is type "concepts" (in the Jackendoff-ian sense) that are taken as atomic.

As will be seen below, there will be a fundamental connection between these parts of the feature structure and semantic content. Before spelling out this connection explicitly, however, we will look at a few more examples of lexical decomposition.

1.2 Levin and Rappaport Hovav's verb decomposition

In [25], Levin and Rappaport Hovav offer a decompositional semantics for verbs specifically. In particular, they model those aspects of verb semantics that have consequences for which grammatical arguments must be, might be, or cannot be syntactically realized in a sentence. Like Jackendoff, they hold the view that there is some definite set of ontological categories that can be invoked as part of the semantic theory, i.e. there is some universal set of atoms from which all verb meaning is constructed. In fact, they accept Jackendoff's ontological categories, *plus* an additional set of fixed ontological categories that inherit from Jackendoff's **event** type. Among these additional ontological categories are **cause**, **become**, and **act**; in another paper [26], they give a long list of others that are commonly seen in theories of verb decomposition. The subtypes of Jackendoff's other ontological categories (which are not dubbed as "ontological categories"), are left explicitly open-ended; that is, they can be whatever and however many in number are required to give a semantic analysis of all verbs in a natural language.

Some of the ontological categories in this additional set are called *predicates*, and they will determine the structure of the semantic entry (while arguably simultaneously providing some semantic content), while the others are called *constants*, which only provide semantic content and act as arguments for the predicates.

We cannot give a semantic entry for "novel" in their system, since it deals only with verbs, but we can give an entry for "begin":

$$[[x\ \textbf{act}]\textbf{cause}[x\langle ACTION\rangle]],$$

where *ACTION* is a variable standing for any subtype of the constant **action**. If some agent x *begins* to perform some *action*, then x *acts* to *cause* x to be in a state of *action*. For example, the event structure for the event described by (2) "Maria began reading" is

$$[[\text{Maria}\ \textbf{act}]\textbf{cause}[\text{Maria}\langle\textbf{reading}\rangle]].$$

Again, we have a set of atoms that provide semantic content, i.e. the constants. And again the use of these atoms relies on an inheritance relation: inheritance from **action** is a constraint placed on the content of arguments for "begin."[7]

Levin and Rappaport Hovav's semantic theory is not merely an abstract exercise. Computational lexical resources exist which have real-world technological implementations such as automated translation, corpus annotation, treebanking, automated sentence parsing. These implementations are useful for application in artificial intelligence, speech recognition, language learning software, etc.

VerbNet, a lexical resource created and managed by Karin Kipper-Schuler, Martha Palmer, and others at University of Colorado, is currently the largest on-line verb lexicon for English. Lexical entries are organized into verb

[7]Section 3.1 of this chapter includes some more examples of this kind of constraint. All three examples in this section are in the Levin class of *change of state* verbs, which share the structural template [**become**[$y\langle$*RES-STATE*$\rangle$]]. Any constant that can serve as a value for the variable *RES-STATE* must be a subtype of the constant **resulting_state**.

classes based on the predicates of Levin and Rappaport Havov[8], and it is perhaps in the implementation that we can most-readily observe the type-inheritance system of Levin and Rappaport Hovav's content-bearing constants (or atoms).

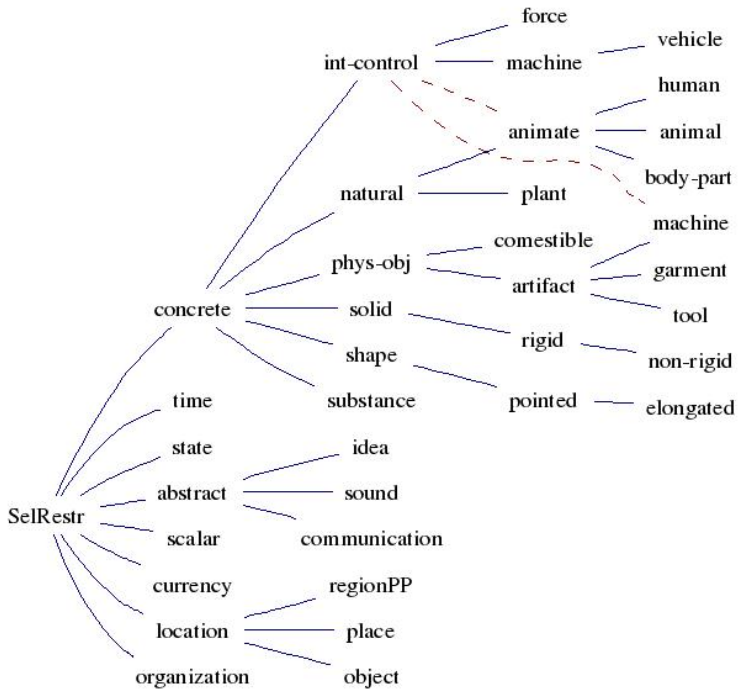

Figure 1: A sample type inheritance hierarchy from VerbNet

[8]In VerbNet, the predicates are called *thematic roles*, following [23], but they come down to the same thing as Levin and Rappaport Havov's predicates. I call them predicates because I have already introduced this term and VerbNet is explicitly based on their theory.

2 Moral of the story: Lexical Semantic Content and Inheritance

2.1 Lexical Semantic Content

In any decompositional theory, there is a structural component and a content-providing component. Levin and Rappaport Havov [26] have made this exact point for their own theory (and others in the same general subclass of theories of lexical decomposition). Although their theory is a theory of verb decomposition, their comments apply to lexical decomposition in general. Many words might have the same general structure for Levin and Rappaport Hovav's lexical entries, only differing in their semantic content, which is represented in the form of semantic atoms. For example, given the structural template [**become**[y⟨*RES-STATE*⟩]], where *RES-STATE* is a variable ranging over the domain of resulting states, we have the following three lexical entries:[9]

1. *dry*: [**become**[y⟨**dry**⟩]]
2. *widen*: [**become**[y⟨**wide**⟩]]
3. *dim*: [**become**[y⟨**dim**⟩]]

It is obvious that "dry," "widen," and "dim" all have very different meanings, yet the structure of these lexical entries is the same, different only in the atom that occurs in the argument for **become**. The part of each lexical entry that is idiosyncratic to each word is the content provided by the atom. The same point applies to Jackendoff, and, as we'll see below, to the Generative Lexicon theory [35].

In lexical decomposition, semantic content is always inherited from some basic atom. But this is only possible if atoms have content, so it makes sense to inquire after the content of atoms.

[9] "Lexical entry" is a general term that applies to any structure or item that is used to convey the meaning of a word in the lexicon of a semantic theory.

David Lewis [27] has given a compelling argument in the domain of sentential decomposition that atoms must be given an interpretation in order for a semantic theory to do any of the work we require from such a theory. A decompositional theory provides a set of atoms or "markers," which are essentially a lexicon for the theory. The compositional rules provide a syntax according to which we combine the atoms into some sort of structure representing the semantics of whatever linguistic unit is in the domain of the theory (in Lewis's case, the linguistic units under consideration are sentences; in ours, they are words). "But," Lewis writes,

> we can know the Markerese translation of an English sentence without knowing the first thing about the meaning of the English sentence: namely, the conditions under which it would be true. Semantics with no treatment of truth conditions is not semantics. Translation into Markerese is at best a substitute for real semantics, relying either on our tacit competence (at some future date) as speakers of Markerese or on our ability to do real semantics at least for the one language Markerese.

Proponents of decompositional theories are able to get by without providing content to their atoms precisely because they and their audience are speakers of the relevant "Markerese." In all theories considered here, the atoms are represented in English. Since we are all able to understand the words represented by these signs, in virtue of the fact that we are all fluent speakers of that natural language, we are able to make sense of decompositional lexical entries even without a "real semantics" for English or Markerese. Theorists of lexical decomposition rely on fluency to supply content to atoms where none is given within the theory. One of the main goals of this project is precisely to indicate one way in which we can do "real semantics" for Markerese, at the same time situating our decompositional theories within the broader context of "real semantics" for natural language.

If we follow Lewis in thinking that uninterpreted atoms do not have content, but want to retain the decompositional approach, then we will want to compose word meaning out of some atom whose content we understand

independently of the decompositional theory. Absent some interpretation that provides content to atoms from without, it is difficult to see how we might obtain such understanding. Any theory of lexical decomposition is necessarily unable to non-circularly provide content to its atoms. Within the theory, the meanings of atoms can only be decomposed into other atoms; theories of lexical decomposition are *only* theories of lexical decomposition. But if the meanings of atoms are given decompositionally, then either: (a) they aren't *actually* atoms of the theory; (b) we will ultimately bottom out with the meanings of some atoms given in terms of themselves; or (c) we have a "decompositional circle," e.g. α is defined in terms of β and γ, β is defined in terms of δ and ϕ, and so on, until we reach some atom defined in terms of α. In any case, we cannot form a coherent view of how a decompositional theory can provide meaning to its own atoms. The current external source of content for atoms is their identification with English words, which retrieve content from fluency of native speakers. But if this is the actual source of content for the atoms, then we have an obvious circularity: linguistic items supply content to atoms, which supply content to linguistic items.

Nevertheless, linguists and philosophers alike (many of them at least) seem to agree that decompositional theories play a crucial role in semantics as a whole.[10] It is not the case that lexical decompositional theories are completely useless until and unless we can find ourselves in possession of a convincing account of the content of its atoms. Any lexical decompositional theory comprises only part of a total science of lexical semantics. Decompositional theories can tell us a great deal about the structural component of meaning.

However, we take ourselves to be talking *about* something when we use words, and except when what we are actually talking about is atoms, to say that atoms are the providers of the content component of meaning doesn't

[10]In the domain of sentential semantics, it is wholly uncontroversial that decomposition is the right approach. There is some controversy over lexical decomposition, and die-hard opponents of lexical decomposition may reject what is said here. Justifying lexical decomposition is outside the scope of this paper; I seek to fortify lexical decomposition, given that we already agree on the appropriateness of the decompositional approach to word meaning.

tell us what our words are *about*. When I say, "The sky is clear today," I've said nothing at all about atoms. I've said something about the sky. Whatever sort of thing meaning is, *aboutness* is a major component of it. We might say that "The sky is clear today" is about the sky because the word "sky" refers to the sky. This is on the right track, but some words have meaning without referring to anything, e.g. unicorn, goblin, etc., so reference cannot be a necessary condition of meaning. As a guiding intuition, we might say: When a word has reference, its reference plays an important role in meaning. In cases where a word has no reference, some story will need to be told about the meanings of words, but I think that what I will argue is compatible with any plausible story that might be offered.

My proposal is that we can identify some cognitive representation—namely, concepts—that is generally agreed to be *about* something, and which will hook up with the world in some way in cases where references exist in the world. If we are able to do this, we will be able to satisfy referentialists such as Lewis, by allowing our cognitive structures to mediate between words and reference while also being palatable to subjectivists such as Jackendoff, who wish to give a theory of internal semantics in which linguistic units have no direct reference beyond what is present in the mind.

We have now identified in passing a number of criteria that will justify concepts as an appropriate interpretation of semantic atoms. It will serve us well to review them explicitly. Two problems with the current situation have been mentioned: there is no coherent, non-circular way for decompositional theories to provide content to their own atoms without relying on prior understanding of the meanings of words; without content for the atoms, we are unable to account for *aboutness*. Also, we should keep in mind that atoms are supposed to be carriers of semantic content; therefore, any viable interpretation must exhibit some behavior that reflects fundamental observations about semantic content. At this point, we are able to identify three criteria that a successful interpretation of the atoms must meet. I will take the current project to have been successful when I have established an interpretation that

(i) accounts for *aboutness*,

(ii) is not itself dependent on linguistic meaning, and

(iii) explains why inheritance networks appear in all theories of lexical decomposition, despite the fact that inheritance is not essential to decomposition *per se*.

3 James Pustejovsky's Generative Lexicon

James Pustejovsky's theory of lexical decomposition, the Generative Lexicon (GL) [35] will provide a case study for developing the conceptual interpretation of semantic atoms, so it will be useful to look at GL in some level of detail.

GL is a theory of lexical semantics based on LKB [41], which is in turn based on Bob Carpenter's logic of typed feature structures [11]. GL is a model of word meaning in which the mechanisms underlying systematic polysemy allow new word senses to be generated "on the fly," without their being given explicitly in the lexicon. Historically, the simplest and most direct means of handling polysemy has been to allow the same word to be listed multiple times in the lexicon, with each listing storing a different semantics for the word. A precise characterization of this sort of *Sense Enumeration Lexicon* SEL is

> A lexicon L is a *Sense Enumeration Lexicon* if and only if for every word w in L, having multiplie senses $s_1, \ldots, s_n$ associated with that word, then the lexical entries expressing these senses are stored as $\{w_{s_1}, \ldots, w_{s_n}\}$. [35, p. 34]

The inadequacies of SELs are spelled out detail by Pustejoveky [35, Ch. 4], and for a fuller discussion of them, the reader is referred to his book. They are not important for our purposes here. What is important is that Pustejovsky capitalizes on the well-known phenomenon of systematic polysemy as a way of reducing the number of entries in the lexicon. To understand what systematic polysemy is, it is helpful to distinguish it from polysemy

(simpliciter) and lexical abiguity (or homophony). Polysemy (simpliciter) is the capacity that some words possess to take on multiple distinct, but related meanings. For example, we use the word "dog" to refer to both actual dogs, drawings of dogs, or even humans whom we call dogs pejoratively to indicate unsavory behavior. However, not all cases of using a common morpheme to refer to different things are cases of polysemy. The very common example of "bank," which can refer to a financial institution or the land beside a river, is not an example of polysemy, but of lexical ambituity (or homophony). These meanings are entirely unrelated, and in fact it is common for linguists to regard these uses of "bank" as entirely different words, which share a phoneme through historical accident. A *systematic polysemy* arises when it is achieved in accordance with some regular, generally established *generative rule* according to which word senses can be produced in a predictable way.

3.1 Typed Feature Structures

In the GL, the semantics of a lexical item α is represented as a quadruple $\langle \mathcal{A}, \mathcal{E}, \mathcal{Q}, \mathcal{I} \rangle$. Each component is discussed below. The inheritance structure $\mathcal{I}$ is covered first, since it is the main one we are concerned with here. $\mathcal{A}$ and $\mathcal{Q}$ will become useful in Chapter 5. $\mathcal{E}$ is included for completenes, but it is not essential to understanding what follows.

3.1.1. The lexical inheritance structure $\mathcal{I}$. The lexical inheritance structure $\mathcal{I}$ is a lattice which defines what is to be considered as a type for atoms and the relations between types. Following Carpenter's Logic of Type Feature Structures, the fundamental relation between types is inheritance. The inheritance lattice is a partial ordering $\sqsubseteq$ over types and we say that α *inherits from* β just in case $\alpha \sqsubseteq \beta$. If α inherits from β, then α is a *subtype* of β, i.e. for any object $\mathbf{x}{:}\alpha$, it is also the case that $\mathbf{x}{:}\beta$. The entire logic of typed feature structures is a spelling-out of the consequences of the inheritance relation. We do not need to be too concerned with the details of the logic here and can simply refer to Carpenter's work where necessary.

A sample inheritance lattice is given below:

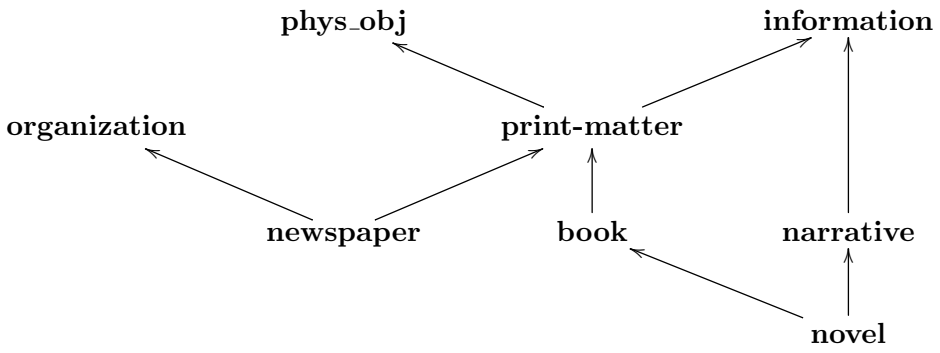

Figure 2

Neither Pustejovsky nor Carpenter commits us to any specific inheritance structure, nor even a specific set of types. Carpenter, in particular, is explicit that, so far as his logic is concerned, any set of types and any inheritance structure will do, provided they possess a certain set of very general characteristics. We need not be too concerned with Carpenter's conditions; it will be difficult to cook up any set of types or inheritance relations which do not meet them, but which are also plausible for use in the GL. A more important issue arises from the fact that Pustejovsky does not consider the question of whether there are some inheritance structures that are better than others for modeling word meaning. Indeed, Pustejovsky doesn't say much about $\mathcal{I}$ at all, other than what it is. In Chapter 4, a proof will be given that there is an important relation between the hierarchical organization of concepts and inheritance structures and that, by giving an appropriate model of concepts which is sufficient to explain hierarchy, we can get greater insight into $\mathcal{I}$. Importantly, it is my hope that we will get some answer to the question of whether any $\mathcal{I}$ at all will do, or whether only some $\mathcal{I}$s are viable candidates as part of a linguistic structure in the GL.

3.1.2. The argument structure $\mathcal{A}$. Argument[11] structure is the best-understood of the components and is regarded as a minimal specification of lexical semantics, although it is far from adequate as a complete characterization of the semantics of any lexical item.

The grammatical arguments of the lexical item "build" are encoded into a list structure ARGSTR in the following manner:

$$\begin{bmatrix} \textbf{build} \\ \\ \text{ARGSTR} = \begin{bmatrix} \text{ARG}_1 = \textbf{animate\_individual} \\ \text{ARG}_2 = \textbf{artifact} \\ \text{ARG}_3 = \textbf{material} \end{bmatrix} \\ \\ \ldots \end{bmatrix}$$

For a verb, the argument structure will give an account of the grammatical arguments expected by the verb, while for a non-verb, the argument structure will correspond, roughly, to the word's behavior when used as the argument for a verb. Arguments corresponding to the subject are indexed with 1, direct objects with 2, indirect objects with 3, 4, 5, etc. This is best understood through example; Chapter 5 will provide a number of examples that illustrate the argument behavior of non-verbs.

3.1.3. The extended event structure $\mathcal{E}$. The extended event structure will not be important for the purposes of this thesis and is included merely for completeness and to prevent the reader from feeling overwhelmed by their presence in the typed feature structures that will be used in Chapter 5. The following is an example of the event structure for the word "build":

[11] Like Jackendoff, Pustejovsky's use of the term "argument" is somehwat misleading for linguists. Pustejovsky's arguments are also arguments of some function. His formal system is explicitly based on Ann Copestake's LKB [14], whose argument structures are explained as inputs to computational functions defined when a grammar is implemented in LKB.

$$\begin{bmatrix} \textbf{build} & \\ \text{EVENTSTR} = & \begin{bmatrix} \text{E}_1 = \textbf{process} \\ \text{E}_2 = \textbf{state} \\ \text{RESTR} = <_\alpha \end{bmatrix} \\ \dots & \end{bmatrix}$$

The E_i are features that correspond to the stages of an event that extends through time. In the above example, the act of building something involves a *process* (of building), and it ends with an object being in a *state* of having been built. The feature RESTR defines the order that is placed over stages of an event, e.g. total ordering, partial ording with simultaneity, partial ordering without simultaneity. $<_\alpha$ is Pustejovsky's symbol for a total order (without simultaneity), which is perfected or completed with the final stage of the event; building ends when the thing we are building is in a state of having been built, and the act of building and the state of having been built cannot occur simultaneously.

3.1.4. The qualia structure $\mathcal{Q}$. Pustejovsky's qualia structure is borrowed from Aristotle's four modes of explanation or four causes. The analysis given here will not depend on a detailed understanding of what the qualia are; we need only understand that they provide semantic information that should be accessible by native speakers of a language when they use a word. In fact, generative lexicons can be created in which the generative mechanisms depend on features other than qualia structure, so there is nothing essential about Pustejovsky's choice of qualia. His choice of the qualia is largely predicated on Moravcsik [31], who argues that Aristotle's four causes serve as a generative mechanism for creating new word senses. The qualia structure QUALIA is constituted by assigning types to the four following features:

1. *Constitutive.* The material properties of an object; the relation between an object and its proper parts. Some examples are:

   (a) Material

   (b) Weight

   (c) Parts and component elements

2. *Formal.* The property according to which an object of a particular species is regarded also as a member of some broader genus. Some examples are:

   (a) Orientation

   (b) Magnitude

   (c) Shape

   (d) Dimensionality

   (e) Color

   (f) Position

3. *Telic.* The purpose or function of an object. Some examples are:

   (a) Purpose for which an object was created by some agent.

   (b) A built-in function or aim toward which the natural activities of an object point.

4. *Agentive.* The origin of an object or factors involved in its "bringing about." Some examples are:

   (a) Creator

   (b) Artifact

   (c) Natural kind[12]

   (d) Causal Chain

As an example, consider the following QUALIA structure for "novel":

[12]Pustejovsky lists this, but I am hesitant to do the same, due to some confusions and/or skepticism I have pertaining to natural kinds, which are off-topic from the present discussion.

$$\begin{bmatrix} \textbf{novel} & \\ \ldots & \\ \text{QUALIA} = & \begin{bmatrix} \text{CONST} = \textbf{narrative} \\ \text{FORMAL} = \textbf{book} \\ \text{TELIC} = \textbf{reading} \\ \text{AGENT} = \textbf{writing} \end{bmatrix} \end{bmatrix}$$

This simple listing of qualia values "tells us nothing about how a particular lexical item denotes, however. For example, although a novel's purpose is the activity of reading and it comes about by someone writing it, we do not want to claim that the common noun *novel* actually denotes such activities" [35, pg. 78]. The qualia provide information *about* what is denoted by the word.

Chapter 3: Typicality and Conceptual Hierarchy

In this chapter, we will become clearer on the relationship between concepts, genus-differentia definitions, and typicality (which will be defined below). In Chapter 2, we saw that inheritance networks appear in all theories of lexical decomposition. In Chapter 4, we will see that inheritance networks provide a model of genus-differentia relations between atoms. The discussion in this chapter will serve as a bridge between concepts and inheritance networks, by arguing that, although the empirical research is conclusive that definitions do not provide a full analysis of concepts, they are an adequate reflection of some aspects of concepts.

Concepts are a kind of cognitive representation; but not all cognitive representations are concepts. Concepts have been called "glue that holds our mental world together" [33] and "units of thought, constituents of belief and theories" [9]. Representations of particular objects, e.g. this particular coffee mug on my desk, my grandmother, etc., are not concepts. Going back to Plato, concepts have been regarded as *general* in character. A concept is a cognitive representation of some general category. In general, I will follow Greg Murphy's convention of using "concept" to refer to mental representations of categories and using "category" to refer to the real-world collections of things.

1 The Classical View of Concepts

Under the *classical view* of concepts, a concept is fully analyzable by a genus-differentia definition, which provides a set of necessary and sufficient conditions for membership in a represented category. This view was implicit in Plato, whose *modus operandi* was to understand a concept by searching for a definition, e.g. *Meno* began as a search for a definition of "virtue," and *Republic* was a search for a definition of "justice." In both cases, it was thought that understanding (and therefore knowledge) would follow from the formulation of an appropriate definition. In pursuit of knowledge, these stories went, definitions were proposed, only to be rejected on the grounds

that the category they picked out was either over- or under-inclusive for what the concept demanded, i.e. when they were found to not provide the right set of necessary and sufficient conditions.

Aristotle followed Plato by making the relation between concepts, definitions and categories explicit, and by giving a more detailed account of what a definition is. For Aristotle, all and only general categories admit of definition, and a definition is a complete account of what a concept is and/or to what category a given object belongs [3] [4]. Classical definitions can be broken down into two components: the *genus* and the *differentia*. The genus is the broader class to which category members belong, while the differentia is a set of properties that distinguishes between category members and other items falling under the genus. For example, take Aristotle's famous definition: A human is a rational animal. "Animal," in this definition, is the genus. All humans belong to the broader class of animals, i.e. the category of humans is a subcategory of the category of animals, i.e. "human" is subsumed under "animal." The differentia here is "rational." All humans possess a rational faculty and no animals other than humans possess a rational faculty. Another way of stating this is in the form of an all-and-only condition for category membership: *all* humans are animals, and humans are the *only* animals that are rational.

From another angle, definitions provide sub- and super-category relations between concepts. The differentia "rational" specifies which subcategory of animals we are thinking about when we use the concept "human." Because "human" is a subcategory of "animal," any knowledge we gain that holds true of all animals can be inferred to hold true of all humans. This kind of knowledge inheritance is the basis for Aristotle's theory of the syllogism, which was the fountainhead of all subsequent work in logic. Concepts, and therefore knowledge, are organized hierarchically.

2 The Downfall of the Classical View of Concepts

We have now identified two aspects of the classical view that will be important for what follows. The first is the *intrinsic* properties of a concept, i.e.

those properties we concern ourselves with when about when talking about a concept in isolation. The second is the *relational* properties of the concept, i.e. those properties we concern ourselves with when situating a concept in relation to others within a conceptual system as a whole. The classical view makes two claims about the relation between definitions and concepts.

(I) Definitions specify what is *intrinsic* to a concept, by virtue of their stipulation of necessary and sufficient conditions for category membership, and

(R) Definitions specify a hierarchy relation that holds between concepts, which *relates* a concept to others within a hierarchy.

What will follow is an argument that the empirical data stand against the viability of (I) as an account of how to go about answering questions about what is intrinsic to a concept (or, what is internal to a category); it is incorrect to equate the question "Is this a human?" with the question "Is this a rational animal?" However, philosophers and cognitive psychologists have been too hasty in rejecting the classical view entirely, since other empirical research does support (R), which is the claim that definitions can answer questions about how concepts relate to one another.

At some point between Aristotle and Kant, philosophers lost sight of (R) to a large degree and began focusing almost exlusively on (I). Kant's distinction between analytic and synthetic truths in *Critique of Pure Reason*, under one (not uncontroversial, but not at all implausible) reading, relies heavily on an appeal to (I), but no appeal is made to (R), except to the degree that he endorses and relies on Aristotle's theory of the syllogism. For Kant, an analytic truth is one in which the predicate is "contained" in—or intrinsic to—the subject concept, and all of his most famous examples of analytic truths (perhaps all of his examples) are cases in which the predicate is a component of or entailed by the definition of the subject concept.

By losing sight of (R), the road was paved for Wittgenstein's influential argument that definitions are not a complete account of semantic content, on the grounds that no set of necessary and sufficient conditions could be found

for membership in the category "game" [44]. As an alternative, he offers us the view that word reference is determined by "family resemblances," in which category membership is determined by general similarity, rather than a static definition. Wittgenstein's argument, it should be observed, addresses only (I). He fails to account at all for the task performed by definitions in (R), which leads him to the view that definitions do not convey word meaning. Wittgenstein's argument against the classical view, although widely regarded as strong and important, had no empirical scientific backing and amounted to little more than "We're having lots of trouble finding necessary and sufficient conditions, so why should we even suppose they exist?"

Beginning with Eleanor Rosch's work in the 1970s, Wittgenstein's family resemblances have amassed a body of empirical support that cannot be ignored. The literature refuting (I) is far too extensive to be surveyed completely here, with literally thousands of experiments spanning a period of over 40 years confirming the ubiquity of the family resemblance model of conceptual representation. Some hangers-on in the fringe notwithstanding, the classical view is now widely accepted by cognitive scientists as—at minimum—unjustifiable on both theoretical and empirical grounds. It is my view, however, that only aspect (I) of the classical view has been overturned. Rosch's work in particular is useful here, since it provided the impetus for all later research on this subject, and since her experiments provide data in favor of (R), while simultaneously providing data against (I), but it has been corroborated at length.

3 Typicality vs. Classical Concepts

3.1 Typicality

In [37], and [40], and [39] Rosch argues that the classical view entails that there are no "better" or "worse" members of a given category. If an item is picked out by the category definition, it is in the category; if it isn't picked out, then it's not in the category. Category membership is all-or-nothing,

and under the classical view it doesn't make any sense to talk about some items being *more* "in the category" than others. This is the aspect of the classical view that Rosch challenged, and it clearly amounts to a challenge against (I).

Rosch's work introduced the notion of *typicality*, which is explicitly presented as a development on Wittgenstein's family resemblances. Under the typicality model, category membership is gradient, rather than binary. Categorization is performed on the basis of some sort of overall similarity between members, i.e. a family resemblance, but there need not be any particular feature that holds true of all members of the category. This allows us to hold the view that birds can fly, but some birds such as penguins and ostriches cannot fly. Typicality is a measure of "goodness" of an item as a category member, and has been generally shown to be predictive of whether an item will be included by test subjects in the category.

Rosch observed a variety of different typicality effects. Each of these effects have been observed in subsequent research, which focuses more closely on specific results that did Rosch's original experiments.[13]

3.1.1. Test subjects *do* regard category members as "better" or "worse." This is perhaps the most fundamental observation of typicality. In [38, Experiment 1], test subjects were presented with members of a category and asked to issue a numerical ranking (from 1 to 7) of the extent to which they regarded the item as representative of the category as a whole. She found that not only were test subjects extremely willing to issue such ranking, but that the rankings were fairly consistent across test subjects. Subjects tended to agree that robins and sparrows are highly typical birds, while turkeys and penguins are not. This result was reproduced by Rips, Shoben, and Smith [36], who agreed with Rosch that category membership is determined by overall similarity, but who argued contra Rosch that typicality effects could be reconciled with essentialism (the view that there are

[13]Often subsequent research was intended as a response to certain problems that were later uncovered in Rosch's conclusions. These problems are not important here. Our purpose is simply to point out the ubiquity of typicality effects in empirical research on concepts.

some properties essential to category membership).

3.1.2. Typicality, reaction time, and learning speed. Rosch and Mervis [40, Experiment 5] set up a sorting task in which subjects were presented with strings of letters and numbers that experimenters had previously sorted into two categories according to typicality. Subjects were asked to make judgments about which category each string belonged to and were issued negative feedback when their judgment did not conform with the categories predetermined by experimenters. They found that subjects were more quick to make judgments about more typical members of each category and that fewer errors were made before finally learning to categorize typical strings correctly. In [40, Experiment 6], they repeated the same experiment, except that natural categories were used, rather than artifical categories of strings of letters. Results were the same.

Gastgeb, et al. [20], repeated Rosch and Mervis's sorting tasks for populations with different modes of cognitive functioning. They were able to reproduce the results of Rosch and Mervis in a normally functioning population, which was taken as a control group, and showed that high-functioning autistic children not only displayed the same typicality effects, but that typicality effects on reaction time and learning speed were more pronounced in autistics.

3.1.3. Memory and recall. Rosch [38, Experiment 6] primed subjects with a series of 54 objects belonging to the same category. At the end of a priming session, subjects were asked to write down all of the category members they had seen. It was observed that subjects had far greater recall facility for typical items than they did for aytpical ones. Loftus [28] raised some concerns that Rosch failed to control for a tendency of subjects to categorize atypical category members as belonging to a different category; since subjects were cued to perform a task involving a specific category, differences in categorization could introduce a counfounding uncontrolled variable. In later research, Ciss and Heth [12] introduced greater controls and were able to produce Rosch's original result.

3.1.4. The ubiquity of typicality effects. This is only a small sample of the research supporting typicality effects. Rosch endorsed a specific model for explaining typicality, i.e. the *prototype theory* of concepts, which states that typicality is a measure of overall similarity to an abstract list of features, called a *prototype* or "best example" of a category. The competing *exemplar theory* holds that typicality is a measure of overall similarity between a list of particular instances of a category that have been encountered by concept-holders in the past. Although I will endorse the ubiquity of Rosch's typicality effects, I remain mostly agnostic as to whether her prototype theory is the correct model of typicality. The point that is relevant for the present project is that typicality effects have been observed in a variety of cognitive domains and across a number of populations. The abundance of empirical support for typicality, and the relative scarcity of evidence against typicality presents conclusive evidence against aspect (I) of the classical view. However, other research seems to give strong support for aspect (R).

3.2 Hierarchy

As a reminder, let us restate (I) and (R). According to the classical view:

(I) Definitions specify what is *intrinsic* to a concept, by virtue of their stipulation of necessary and sufficient conditions for category membership, and

(R) Definitions specify a hierarchy relation that holds between concepts, which *relates* a concept to others within a hierarchy.

Although genus-differentia definition fails to provide necessary and sufficient conditions for category membership, as (I) states, it seems highly implausible that genus-differentia definition has no relation to concepts whatsoever. (R) seems to be almost trivially true, and despite its inadequacy, many fruitful results were obtained under the classical view, in particular by Piaget's influential work from the 1960s [33, pg. 15]. In fact, the same series of experiments that established the falsity of (I) produces empirical results

that seem to support (R). In failing to remember that (R) is part of the classical view in its original form, researchers have rejected the classical view entirely. In distinguishing clearly between (I) and (R), we will be able to make note of some empirical results regarding the hierarchical organization of concepts, which will help to preserve a portion of the classical view.

Research in cognitive science has established that concepts are organized more-or-less hierarchically. Rosch's original experiments showed that, where typicality effects did not confound response times, subjects were more quick to respond to questions that conformed to a hierarchy of the kind expressed in Aristotle's genus-differentia definitions, and therefore (R), e.g. inferences of the general kind described by Aristotle's theory of the syllogism. She distinguishes between three hierarchy levels—basic, superordinate, and subordinate—similar to Aristotle, and at times, she refers to these levels as species-levels and genus-levels, as Aristotle does in his writing on definitions.[14] Rosch's results have been reproduced and expanded elswhere, as in [22], [13], and [6].

A hierarchy—in the sense intended here—is a network in which members are related by the set-inclusion relation. If one concept C is higher than another concept D in the hierarchy, then the category of which D is a mental representation is a subset of the category represented by C; we say that C is *superordinate* to D and D is *subordinate* to C. It is not too difficult to see how genus-differentia definition naturally falls out of hierarchy. If we take $\phi_D(g, d)$ to be a genus-differentia definition for D, where g is the genus and d is the differentia, then we see that the C-category is a candidate value for g. C is superordinate to D, i.e. C represents a broader category of which D represents a subset, i.e. C is a genus of D. d is some property possessed by all members of the D-category, but no other members of the C-category. Since hierarchy relations are subset relations, they are reflexive and transitive, but not symmetric. But these are properties of the inheritance relation, which

[14]Subjects have a natural tendency to form concepts at the basic level, which is a sort of middle ground between generality of application and specificity of description. Rosch's original metric for measuring basicality was found to be unpredictive of actual categorization behavior in certain domains [32], but later research, such as [34], has offered other metrics which do better.

is enough to *suspect* that conceptual hierarchy and inheritance orders have some fundamental relation to each other.

3.2.1. Objections to the hierarchy In [43], Steven Sloman presents a series of experiments which he claims to support the conclusion that "Categories whose natural organization constitutes an inheritance hierarchy are surprisingly rare." In fact, what he has actually demonstrated is: *Categories whose natural hierarchical organization is not confounded by typicality results in the case of atypical fringe cases are surprisingly rare.*

Sloman's experiments asked subjects to evaluate Aristotelian inferences according to their willingness to accept conclusions on the grounds of the premises. I will describe only one experiment here; the differences between experiments are not important for what I will say here. The kinds of results seen, and the kinds of conclusions drawn by Sloman are the same for each experiment.

Consider the following arguments:

$$\frac{\text{All metals are pentavalent.}}{\text{Iron is pentavalent.}}$$

$$\frac{\text{All metals are pentavalent.}}{\text{Platinum is pentavalent.}}$$

Both are Aristotelian syllogisms that omit the middle premise "X is a metal," where X stands in for either "iron" or "platinum."[15] The ommission of this middle premise is important in order to ensure that hierarchical organization, rather than ability to perform explicit logical inference, is being tested; subjects are, in effect, being asked to make use of their conceptual hierarchy to supply the middle premise. Subjects were first tested to determine a typicality ranking for iron and platinum as different kinds of metals, and it was observed that iron was regarded as a more typical metal than

[15]Readers who are well-versed in Aristotelian logic will recognize the syllogisms as Barbara, from the first figure, with the minor premise ommitted. Aristotle regarded such inferences as syllogisms, but *imperfect* syllogisms, due to the omitted premise.

platinum. Next, subjects were asked to evaluate their willingness to accept each conclusion on the grounds that all metals are pentavalent. It was observed that subjects were more willing to accept the conclusion that iron is pentavalent than they were to accept that platinum is pentavalent.

If concepts are organized hierarchically, and if "platinum" and "iron" are both subordinate to "metal" in the hierarchy, then they are subordinate regardless of typicality. There is no apparent reason why the atypicality of platinum as a metal should inhibit subjects from supplying the middle premise "Platinum is a metal." But this seems to be exactly what is observed, and therefore, Sloman concludes, concepts are not organized heirarchically (except in a few special domains, e.g. biological concepts).

There is an alternate explanation of Sloman's observations, which does not threaten hierarchy to the same degree. Greenberg and Bjorklund [21] performed a series of free recall experiments in which subjects were more willing to recall typical subcategories to a superordinate category than they were to recall atypical subcategories. They have hypothesized that this may be due to a tendency toward poor organization or mis-organization of atpyical categories. This hypothesis is a variant the very complaints raised by Loftus [28] against Rosch's original experiments on free recall; free recall might be inhibited by miscategorization of atypical items. If their hypothesis is correct, then Sloman's observations are consistent with hierarchy and show that people can make mistakes in the way they position concepts within the hierarchy, and that typicality effects have a non-negligible effect on the prevalence of such errors. Under this hypothesis, Sloman's objections against hierarchy don't carry quite the same weight.

4 Summary, or: back to the main point

The real takeaway points of this chapter are as follows:

(i) On the classical view, a concept is fully analyzable by a genus-differentia definition:

   (a) Genus-differentia definitions supply necessary and sufficient con-

ditions for category membership (i.e., what is intrinsic to a concept).

(b) Genus-differentia definitions describe hierarchy relations within a system of concepts (i.e., facts about how concepts are related to one another).

(ii) At some point, researchers began focusing heavily on the intrinsic claim (I) made about genus-differentia definitions in the classical view, while ignoring the relational task definitions play in situating concepts within the hierarchy.

(iii) In showing that genus-differentia definitions do not account for what is intrinsic to a concept, researchers have come to reject the classical view entirely.

(iv) But,the empirical result that concepts are organized hierarchically lends credence to the part of the classical view that claims genus-differentia definitions can describe relations between concepts.

Chapter 4: Inheritance Networks and the GDIT

It should be clear at this point that there is a fairly high degree of analogy between the inheritance relation of decompositional theories and conceptual hierarchy, since both the set-inclusion relation and the inheritance relation are transitive and reflexive, but not symmetric. It might be tempting at this point to think that set-inclusion and inheritance are the same relation, and therefore no additional work is required to establish that conceptual hierarchy and semantic type inheritance are the same thing. However, Carbonell [8] notes several examples of relations between categories that can be modeled in type inheritance networks, but which contradict what would be allowed by set-inclusion:

- Birds fly, penguins are birds, penguins do not fly.
- A flivvit is just like a small car, except it has only three wheels arranged like a tricycle.
- John is a graduate student. John is an heir to the Heinz fortune. Graduate students are not rich. Heirs to fortunes have a lot of money. Graduate students work hard. Heirs to fortunes do not work hard. Is John hard-working or rich or both or neither?

The relations described in both of the above sentences can be implemented by inheritance networks, but they all violate set-inclusion in some way. In the first, the fact that penguins do not fly implies that **penguin** cannot be a subset of **bird** if **bird** is a subset of **flying_things**. Nevertheless, we do expect birds to display flying behavior, and we do classify penguins as birds. In the second case, we define **flivvit** as inheriting from **car**, but because of the arrangement and number of a flivvit's wheels, **flivvit** is not a subset of **car**; **flivvit** is defined precisely by stating the way in which we cannot regard flivvits as cars, while simultaneously having **flivvit**$\sqsubseteq$**car**. In the third case, the type **John** inherits contradictory information from **graduate_student** and **heir_to_fortune**, but it cannot be the case that any set A is a subset of both B and its complement.

In fact, the equivalence between set-inclusion and inheritance fails in exactly the same ways that the classical view of concepts has been seen to fail. And the solution offered by Carbonell is exactly the solution offered by Rosch: typicality rankings. Carbonell writes, "*Inheritance* means that assertions made of a type ought to be transmitted to all [tokens] of that type... Clearly, it is useful to store *typical* information within the type and note the few exceptions on the instances" [8]. To say that α inherits from β is not to say that every α is a β, but rather to say that we expect α to be associated with the majority of the information we would expect to be associated with β.

It is, therefore, a non-trivial matter to show that inheritance networks can serve as a model of semantic content while also serving as a model of a system of set-inclusion relations such as conceptual hierarchy. And there is an open question of *which* set-inclusion relations in the hierarchy are modeled by inheritance networks.

Conceptual hierarchy has already been explained in some level of detail, but semantic content remains far too vague a notion to be much use here. First, I will identify some observations about semantic content that admit of straightforward formalization. I will then define the notion of an *inheritance network*, a partial order over two relations that seem to model these observations. Once the framework has been set up, I will state the Genus-differentia Inheritance Theorem, which establishes an isomorphism between inheritance networks, (aspects of) semantic content, and conceptual hierarchy. Finally, I will identify some philosophical implications of the Theorem, as well as some additional ideas that will be useful in carrying out the case study.

0.1 Two important semantic relations

Before defining the type-inheritance networks that will serve as a model for carriers of semantic content, we should identify which aspects of semantic content we intend to model.

It is a well-known feature of word meaning that lexemes can be related

to one another by a certain class of relations—let's call it the class Δ—in which the semantic content of one word is regarded by native speakers as carrying implications for the semantic content of some other word. D. A. Cruse has offered a number of diagnostic tests for Δ. Two of these tests will provide the basis for semantic relations included in the model offered here.

Certain sentence forms will provoke an intuition of "oddness" when Δ is violated [15]. Two such sentence forms are illustrated in the following four sentences:

(1) ? It's a dog, therefore it must be a cat.

(2) It's a dog, therefore it's an animal.

(3) ? It's a dog, but it can bark.

(4) It's a dog, but it can't bark.

(1) and (3) are odd, while (2) and (4) are not. In (3), it seems odd to conjoin "It's a dog" with "It can bark" using "but," because dogΔbark. Δ can be modeled by the inheritance relation. If we set $\mathbf{dog} \sqsubseteq \mathbf{barks}$, then $\alpha : \mathbf{dog} \vdash \alpha : \mathbf{barks}$, which is a straightforward formalization of the entailment exhibited by the oddness of (3). (4) lacks oddness because the meaning of "but" conforms to the fact that non-barking is unexpected behavior for dogs.

Another important kind of Δ is exhibited in (1) and (2). (1) seems odd because "It's a dog" implies "It's not a cat;" we regard "dog" and "cat" to be semantically disjoint. (2) however, is a perfectly non-odd sentence, because all dogs are animals (i.e. $\mathbf{dog} \sqsubseteq \mathbf{animal}$, i.e. $\alpha : \mathbf{dog} \vdash \alpha : \mathbf{animal}$), which means that "dog" and "animal" cannot be semantically disjoint. No relation has been explicitly introduced thus far which can be used to model semantic disjointness, although the dotted lines in Figure 1, Chapter 2, do represent such a relation, which will be defined explicitly below.

Not all *but*- and *therefore*-sentences tell us about semantic content—most of the ones people actually use don't. In fact, (2) and (4) do not. Nor do such sentences as "He's a Republican, but he's a pretty nice guy,"

or "It's a Chevy, but it gets pretty good mileage." Only sentences like (1) and (3), which do provoke native speakers to have intuitions of oddness are taken by Cruse as diagnostics for semantically important relations between words. A sentence of the form "A, but B" can only serve as a diagnostic if, in some sense, a speaker wouldn't even count as knowing what A (or B) means if she were unsurprised at learning that both conjuncts A and B were true. Similarly, a sentence of the form "A, therefore B" can only serve as a diagnostic if a speaker wouldn't count as knowing what A (or B) means if she is willing to accept that both A and B can be true.

There are other semantic relations we might want to model; if so, we will want to enrich the framework presented below by adding additional relations to our partial order. Adding more relations will add more complications, however, and for the time being I simplify matters by including only those relations that are necessary to establish the isomorphism to concepts.

1 Inheritance networks and genus-differentia definition

I will now set linguistics aside for a moment and develop the notion of an inheritance network. Before giving an abstract formal definition, we look at a concrete example of an inheritance network:

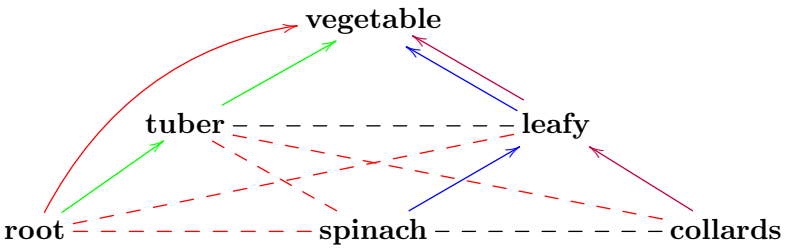

Figure 1

The inheritance relation is represented by directed edges going from subtype to supertype. For example, in the above graph, we see that **tuber** inherits from **vegetable**, which is to say that we expect tubers to share

most of the salient features of vegetables in most contexts. However, as we have noted, this does not necessary entail that the category of tubers is a subset of the category of vegetables, and there may be some contexts in which we would say that a particular tuber is not a vegetable, e.g. when enjoying a rhubarb pie, some gourmands might want to regard rhubarb as a fruit, rather than a vegetable, even though it is a tuber, just as we can regard tomatoes as vegetables in some culinary contexts. The relation between context and inheritance will become important and made clearer in Chapter 5.

Dashed edges represent the *disjointness* relation. For example, **collards** is disjoint from **tuber**, which means that we do not typically expect collards to behave as tubers in most contexts. The disjointness relation between **collards** and **tuber** is a typicality relation; we typically expect that tubers are roots or stems, but typically, when we talk about collards, we are talking about the leafy greens, not the roots or stems of collard plants. In most circumstances, we will not be inclined to bring the information we associate with roots and stems to bear on collards (other than the information inherited from the common ancestor **vegetable**. Since disjointness is a typicality relation, it can be modeled by an inheritance network. However, our gourmand might want to say that collards are used as tubers in some recipes where collard stems are the main ingredient; since these contexts are outside the norm—i.e. outside what is typical—we still model **tuber** and **collards** as disjoint.

The absence of an edge indicates that no relation between types is being modeled by the inheritance network. (Note that this does not mean that there is no relation between the types that can be modeled by an inheritance network; it merely means that no relation is modeled by this particular inheritance network.)

Borrowing some basic definitions from Bob Carpenter [10], we are now prepared to give a formal definition of an inheritance network:

(Inheritance Network) An inheritance network $\mathcal{I}$ is a triple $\langle \mathcal{B}, \sqsubseteq, \# \rangle$ where:

- $\mathcal{B}$ is a finite set of basic elements
- $\sqsubseteq \subseteq \mathcal{B} \times \mathcal{B}$ is the basic *inheritance* relation
- $\# \subseteq \mathcal{B} \times \mathcal{B}$ is the basic *disjointness* relation

(Inheritance/Disjointness) The *inheritance* relation $\sqsubseteq^* \subseteq \mathcal{B} \times \mathcal{B}$ is the smallest such that:

- $P \sqsubseteq^* P$ (Reflexivity)
- if $P \sqsubseteq Q$ and $Q \sqsubseteq^* R$ then $P \sqsubseteq^* R$ (Transitivity)

The *disjointness* relation $\#^* \subseteq \mathcal{B} \times \mathcal{B}$ is the smallest such that:

- if $P\#Q$ or $Q\#P$ then $P\#^*Q$ (Symmetry)
- if $P \sqsubseteq^* Q$ and $Q\#^*R$ then $P\#^*R$ (Chaining)

The * notation above indicates the distinction between *basic* inheritance and disjointness relations vs. total inheritance and disjointness relations. Basic relations are those that are given explicitly in the definition of $\mathcal{I}$. The total inheritance and disjointness relations for any $\mathcal{I}$ include all of the basic relations, plus all relations that can be derived from the basic relations and the axioms of inheritance and disjointness. In most cases, this distinction will be ignored, since it is not generally relevant in practice, and we will not use the * notation further. It is included here out of necessity for giving a coherent formal definition of $\mathcal{I}$. Basic relations are graphed in black, while derived relations are graphed in red (other coloring will be discussed below).

For example, let $\mathcal{I} = \langle \mathcal{B}, \sqsubseteq, \# \rangle$, where $\mathcal{B} = \{a, b, c, d, e, f\}$, $\sqsubseteq^* = \{\langle a, d\rangle, \langle d, f\rangle, \langle c, e\rangle, \langle b, e\rangle, \langle e, f\rangle\}$, and $\#^* = \{\langle b, c\rangle, \langle d, e\rangle\}$. Figure 1 shows all of the basic relations and derived relations of $\mathcal{I}$ (except for relations trivially derived from reflexivity alone).

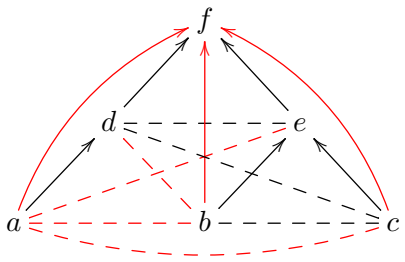

Figure 2

This is just a more abstract representation of the inheritance network shown in Figure 1. It is given in the definition of $\mathcal{I}$ that $\langle a,d\rangle, \langle d,f\rangle \in \sqsubseteq^*$. Now, $\langle a,f\rangle \notin \sqsubseteq^*$. But since $\langle a,d\rangle, \langle d,f\rangle \in \sqsubseteq^* \vdash \langle a,f\rangle \in \sqsubseteq$ by the transitivity of $\sqsubseteq$, Figure 1 shows $a \sqsubseteq f$. Likewise, $\langle a,b\rangle \notin \#^*$, but Figure 1 shows $a\#b$ because $\langle d,e\rangle \in \#^*, \langle b,e\rangle \in \sqsubseteq^* \vdash \langle a,b\rangle \in \#$ by chaining.

Some subsets of $\mathcal{B}$ are of particular import for the present analysis. Assuming potatoes are the only root tuber (a false assumption, but its falsity shouldn't matter for the present purpose), we might take "root tuber" as a genus-differentia definition, where **tuber** serves as a genus and **root** serves as a differentia.[16] Assuming that the meaning of "potato" can be identified with the semantic atom **potato** that belongs in an inheritance network—an assumption made by theorists of lexical decomposition—this locates the meaning of "potato" within the inheritance network, from which we can plainly see that it is part of the meaning of "potato" potatoes are typically regarded as vegetables.

As a model of the notion that the meaning of a word can be located in an inheritance network in this way, I will introduce the concept of *consistent definition*. A full consistent definition for $\mathcal{B}$ of potato is D_{potato} =

[16]It might seem strange to call **root** a differentia, since no other subtype of **tuber** is shown in Figure 4; if it's a differentia, it should be differentiating potatoes from something else. This is a fair point, which could easily be remedied by adding **stem** $\sqsubseteq$ **tuber** to the model. The fact that it is not shown actually demonstrates an important point about this kind of model: not all concepts/atoms need to be included in every model (in fact, I strongly suspect that it is, in principle, impossible to include all of them). But, this means that, when making use of a particular model, we must remain cognizant of the fact that there are things that are *not* being modeled. This is a general feature of using formal models for analysis, however, not just the models I will present here, and discussion of this topic belongs to a different piece of research.

$\{\mathbf{root}, \mathbf{tuber}, \mathbf{vegetable}\}$. However, there is no consistent definition D = $\{\mathbf{spinach}, \mathbf{tuber}, \mathbf{vegetable}\}$ because **tuber** and **spinach** are disjoint. Remember Cruse's diagnostic tests for semantic oddness: "It's spinach, therefore it's a tuber" provokes intuitions of oddness, but "It's spinach, therefore it's a leafy vegetable" does not, and so our notion of consistent definition conforms to existing knowledge about semantic content and captures the semantic relations included in the model at a higher level of abstraction.

The consistent definition of a lexical item is a set of semantic atoms that reflects speaker intuitions about the relations Δ in which the lexical item participates. In a formal theory of lexical decomposition, there will generally be some atom or substructure within the formal semantic structure of a word, which is taken to be the default meaning or "core" meaning of the word. The consistent definition of a lexical item will locate this atom within an inheritance network.

The formal definition of *consistent definition* is:
(Consistent Definition)(cf. [10], *Conjunctive Concept*) A set $D \subseteq \mathcal{B}$ is a *consistent definition* for $\mathcal{B}$ iff:

(1) For all $x, y \in D$, it is not the case that $x \# y$.

(2) For all $x \in D$, $y_1, \ldots, y_n \in \mathcal{B}$, if $x \sqsubseteq y_i$, then $y_i \in D$ iff there is no $y_j \in D$ such that neither $y_i \sqsubseteq y_j$ nor $y_j \sqsubseteq y_i$.

(3) There exists an α_b (called the *base atom* or *base* of D), such that for all $y_1, \ldots, y_n \in \mathcal{B}$, if $y_i \sqsubseteq \alpha_b$, then $y_i \notin D$, i.e. D has a minimal element.[17]

Formally, a consistent definition is a subset of $\mathcal{B}$ containing some atom, called the *base atom* α_b, no disjoint atoms, and a maximal set of atoms $C_{\alpha_b} = \{\beta_1, \ldots, \beta_m\}$ such that $\alpha_b \sqsubseteq \beta_1 \sqsubseteq \ldots \sqsubseteq \beta_m$ for each α_i. Less formally (and perhaps easier to grasp), we can understand a consistent definition as the set of all atoms from which the base atom inherits along exactly one path in the inheritance network. The following is a non-exhaustive list of consistent definitions shown in Figure 1:

[17]Note that being a minimal element of a consistent definition D does not require being a minimal element of the inheritance network $\mathcal{I}$.

(i) $\{a, d, f\}$

(ii) $\{d, f\}$

(iii) $\{f\}$

(iv) $\{b, e, f\}$

Note that $\{a, d\}$ is not a consistent definition, since it does not contain f, and both a and d inherit from f. Neither is $\{b, e, d, f\}$ a consistent definition, since $e\#d$. $\{b, d, f\}$ is not a consistent definition, since there is a derived disjointness relation between b and d.

When it is necessary to call attention to particular consistent definitions, they will be given in colors other than black and red, as in Figure 1.

Multiple inheritance is the feature of some systems in which an element c can inherit from more than one element not identical to c, e.g. $\langle c, a\rangle, \langle c, d\rangle \in \sqsubseteq$. Multiple inheritance is not allowed in all systems, but it will become useful in Chapter 5, so it is allowed here. Figure 3 gives an example of an inheritance network with multiple inheritance:

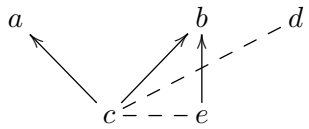

Figure 3

Note that we could not add $a\#b$ to the inheritance network shown in Figure 3, because this would violate chaining. Multiple inheritance is allowed, but restricted. An element c can inherit from multiple parents $a, b, \ldots$, only if none of the parents are disjoint.

Since multiple inheritance is allowed, it is possible that the same atom might serve as the base atom for more than one consistent definition. And this is an exhaustive list of the consistent definitions shown in Figure 3, which has multiple inheritance:

(i) $\{c, a\}$

(ii) $\{c, b\}$

(iii) $\{e, b\}$

(iv) $\{a\}$

(v) $\{b\}$

(vi) $\{d\}$

Although a and b are not disjoint in Figure 3, $\{c, a, b\}$ is not a consistent definition, since $a, b \in D$ would violate condition (2) for consistent definition. This is the "exactly one path" condition in the "less formal" description of consistent definition given above.

(Consistent Definability). We will say that a set S of lexical items is *consistently definable* on $\mathcal{B}$ if and only if there exists a consistent definition D_s for every member of S. A lexical item $s \in S$ is said to be *consistently definable* at a point $b \in \mathcal{B}$ if and only if there exists a consistent definition D_s containing b. An atom $b \in \mathcal{B}$ is *consistently definable* if and only if there exists a consistent definition D such that $b \in D$. A set of atoms is *consistently definable* if and only if all of its members are consistently definable.

Consistent definitions can be combined into:

(Functional Role). The *functional role* of an an atom is a function $i : \mathcal{B} \rightarrow \mathcal{F}$) such that

$$i(x) = \{D \in \mathcal{F} | x \in D\},$$

where $\mathcal{F}$ is the set of all consistent definitions D on $\mathcal{B}$.

We can take a closer look by considering Figure 4:

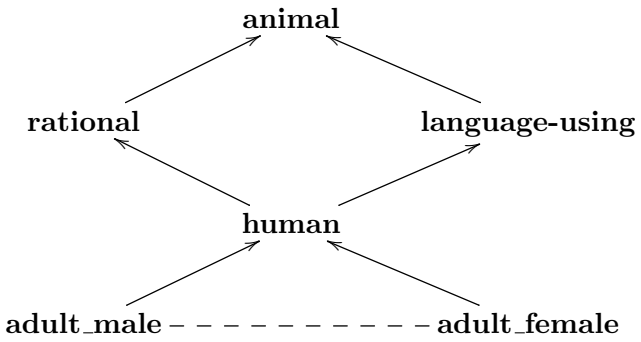

Figure 4

Consistent definitions can be read directly off of the graph:

$$D_{\mathbf{human},1} = \{\mathbf{human}, \mathbf{rational}, \mathbf{animal}\}$$

and

$$D_{\mathbf{human},2} = \{\mathbf{human}, \mathbf{language\text{-}using}, \mathbf{animal}\}$$

are both consistent definitions for $\mathcal{B}$ with $\alpha_b = \mathbf{human}$. Also,

$$D_{\mathbf{man},1} = \{\mathbf{adult\_male}, \mathbf{human}, \mathbf{rational}, \mathbf{animal}\}$$

and

$$D_{\mathbf{man},2} = \{\mathbf{adult\_male}, \mathbf{human}, \mathbf{language\text{-}using}, \mathbf{animal}\}$$

are consistent definitions with $\alpha_= \mathbf{adult\_male}$; similarly for $D_{\mathbf{woman},1}$ and $D_{\mathbf{woman},2}$. We can observe that these are all and only the consistent definitions containing **human** as a member. Then:

$$i(\mathbf{human}) = \{D_{\mathbf{human},1}, D_{\mathbf{human},2}, D_{\mathbf{man},1}, D_{\mathbf{man},2}, D_{\mathbf{woman},1}, D_{\mathbf{woman},2}\}.$$

The following theorem, proven in the Appendix, states that any inheritance network over a set $\mathcal{B}$ of consistently definable atoms is isomorphic to the functional role i with domain $\mathcal{B}$. I will first state it in its abstract mathematical formulation and will then explain more concretely what it means

in relation to our present purpose.

The genus-differentia inheritance theorem (GDIT):[18] Given $\sqsubseteq$ and $\#$, such that $\sqsubseteq$ is inheritance and $\#$ is disjointness over $\mathcal{B}$, a consistently definable set of atoms, there exists a non-empty functional role i such that $a \sqsubseteq b \Leftrightarrow i(a) \subseteq i(b)$ and $a\#b \Leftrightarrow i(a) \cap i(b) = \varnothing$.

2 Philosophical implications of the GDIT

There are a number of ways of understanding what the GDIT does for us. I take it that these are simply different perspectives on the GDIT, not actually distinct facts that the GDIT entails. What the theorem says is that (a) inheritance networks are isomorphic to the function i. When we state things this way, (b) the GDIT can be seen as a representation theorem[19] for inheritance networks. Since, mathematically speaking, inheritance networks are a class of posets, a representation theorem can do a lot of theoretical work for us. One very important aspect of this is that (c) we can also view the GDIT as a soundness and completeness theorem for inheritance networks in the domain of semantic atoms. Given any inheritance network, it is interpretable by some set of consistently definable atoms, and any set of consistently definable atoms can be modeled by an inheritance network. The framework presented here is a formal system—complete with its own well-formedness conditions, axioms, rules of inference, and semantics—and soundness and completeness are important desiderata for any formal systems.

Inheritance networks are a complete formalization of the function i. This is especially appealing for functionalist readers, who will be apt to say that the content of atoms simply is their functional role in the system. For functionalists, then, the GDIT states (d) there is a well-defined class of sets of bearers of semantic content (namely, the class of consistent definitions)

[18] Thanks to Cody Roux for helping me identify the appropriate mathematical tools for stating and proving this theorem.

[19] A representation theorem is a theorem stating that every abstract structure exhibiting certain properties is isomorphic to some concrete structure. In our case, inheritance networks are a concrete structure that is isomorphic to the more-abstract functional role.

that are isomorphic to inheritance networks.

There is a very strong naïve intuition that definitions convey word meaning. When we look into meaning more closely, of course, we find out that this view of meaning is inadequate, but we might search for an aspect of meaning that gives rise to the intuition. By looking toward inheritance networks, our search comes to a close. In theories of lexical decomposition, inheritance networks are an abstract system used to organize the semantic atoms out of which word meanings are composed. Inheritance networks can also serve as models of systems of genus-differentia definitions. In fact, it can be seen from the above that a genus-differentia definition, once thought to be a full account of word meaning, can be understood as a set of coordinates that locate a word's meaning within an inheritance network that models some aspect(s) of semantic content.

There appears to be a deep relationship between semantic content and abstract order theory, in particular inheritance ordering. Looking at semantic content through the lens of abstract orders, we can see why object-oriented programming is so valuable in computational linguistics: object-oriented languages make heavy use of the inheritance relation. Also, it is a perspective with a great enough degree of mathematical precision to facilitate rigorous investigation of linguistic questions, while being abstract and general enough to accommodate a very wide range of specific formal semantic theories. We can understand lexical semantic content *in general*, above and beyond the analysis pursued in specific formal theories of lexical decomposition.[20] In fact, abstract order theory is *so* general that it will afford us a framework for comparing two distinct fields of research: cognitive science and formal lexical semantics.[21] Therefore, this perspective allows us to represent facts about decompositional theories in a way that is especially

[20]Of course, an inheritance network is a specific formal system, and will carry its own limitations. There may be an even more general formal theory of lexical semantic content, which has yet to be discovered, but this does not threaten the observation that inheritance networks provide an advance in generality over current theories.

[21]Cognitive linguistics is, to a very large degree, a response to formal linguistics. Given this connection between cognitive science and formal semantics, one way of reading this essay is as the beginning of an analysis of the relationship between formal and cognitive linguistics. I will not push such a reading here.

disposed toward a cognitive interpretation of atoms.

The GDIT applies to consistently definable concepts as well as atoms. We can just as easily replace the word "atom" with "concept" everywhere it appears in the GDIT and its proof and regard the GDIT as a completeness theorem for inheritance networks in the domain of concepts, instead of in the domain of semantic atoms.

Order isomorphisms are transitive. Since inheritance networks are a kind of partial order, this brings us to the real punchline of this thesis:

> *Since conceptual content and lexical semantic content both have properties that are isomorphic to inheritance networks, they have properties that are isomorphic to each other.*

The GDIT hinges on two things: the notion of consistent definition and on the function i. So we should then ask how they apply to concepts. In the domain of semantic atoms, a consistent definition specified the location of an atom within the inheritance lattice; likewise, in the domain of concepts, a consistent definition locates a concept C within the hierarchy, by specifying a set-inclusion path running from C to its most general superordinate. Note that multiple consistent definitions are possible for the same atom, and the function i maps each atom α to the set of all consistent definitions containing α. In the domain of concepts, i maps each concept C to the set of all set-inclusion paths containing C, i.e. the set of all hierarchy relations in which C participates. Therefore, $i(C)$ is the *functional role* of C in the hierarchy.

The functional role $i(x)$ is a representation of all of the inheritance relations that hold between the semantic content of an atom x and the semantic content of all other atoms in a linguistic theory. Since $\mathbf{adult\_male} \sqsubseteq \mathbf{human}$, by the GDIT we should see that $i(\mathbf{adult\_male}) \subseteq i(\mathbf{human})$. It is left to the reader as an exercise to verify that this is the case. $i(x)$ is also a representation of all of the hierarchy relations that hold between a concept x and all other concepts (included in the model). Since atoms are representations of semantic content and concepts are the providers of semantic content, i is an extremely powerful consolidation of a tremendous amount of information about the semantics of content words.

The analogy between concepts and atoms is quite direct. One *apparent* difference between a conceptual hierarchy and inheritance networks is that, given the way a conceptual hierarchy has been described here, there is no analog of the disjointness relation, which was necessary to model certain semantic entailments that arise out of semantic content. I do think that a disjointness relation can be built into the hierarchy, and it is obvious how theory theories, at least, could accomodate disjointness. Perhaps some of the more extreme versions of the exemplar theory, in which a concept is regarded as a list of *all* objects in the category, could accomodate disjointness as well, but these versions of the exemplar theory are controversial.[22]

If concepts are to do the work required of them here, then they cannot themselves rely on linguistic meaning in any way for their existence or description. We need not be concerned with showing this for *all* concepts. We need only show that this holds for a certain class of concepts, which are viable as an interpretation of atoms. Atoms are supposed to be the most basic components of lexical meaning. It seems right, then, that we might look for some most basic set of concepts. In fact, there is a well-studied set of concepts with exactly this feature.

There is a *basic level of categorization*, which was first studied by Roger Brown [7], later followed by Berlin [6], Malt [29], and others. These researchers have found repeatedly that the most basic level of categorization is neither the most nor the least general level of the hierarchy. Instead, it lies somewhere in the middle. The first precise operational description of the basic level was given by Rosch, et al., in a series of experiments from the 1970s [42] [39]. Their original description was found to be problematic [32], but a number of others have been offered. In particular, Murphy and Brownell have offered a metric in which basicality is determined by *informativeness* and *distinctiveness* [34]. Maximizing informativeness, a measure of the amount of information associated with the concept, predicts a basic level

[22]It may be that *effective* disjointness can be achieved by both prototype and exemplar theories, even where true disjointness cannot. Both theories involve probability distributions over a "feature space" that are everywhere non-zero. However, a probability $P(A)$ can come negligibly close to zero, so that, in practice, we can regard them as effectively disjoint with some other category that assigns a high probability to A.

that is lower in the hierarchy, while maximizing distinctiveness, a measure of the dissimilarity to other concepts with a common superordinate category or genus, predicts a basic level that is higher in the hierarchy. Informativeness and distinctiveness combine into *differentiation*; maximizing differentiation predicts a basic level somewhere near the middle of the hierarchy. The most robust advantages for a basic level that maximizes differentiation have been found when using artificial, purely perceptual categories that are unfamiliar to test subjects and are not associated with any known word, which is very strong evidence in support of the non-linguistic or pre-linguistic nature of basic concepts [33, pg. 220].

Of course, we need atoms at more than just the basic level of hierarchy, and therefore more than just the basic level of concepts needs to be non-linguistc. Unfortunately, the empirical research is not particularly helpful here. Subordinate concepts have not been very well-studied. But subordinate concepts aren't a major concern at any rate. In the inheritance networks of linguistic theories, there is a definite bottom level, which is not the case with concepts. Moreover, the bottom level usually given in examples of lexical decomposition is the kind of thing that is normally seen at the basic level of concepts, e.g. **book**, **dog**. More important are the superordinate concepts, which are obviously an important part of inheritance networks over semantic atoms; we see such general atoms as **event** and **thing**. But research on superordinate concepts is unfortunately linguistic at the outset. Most research into this area is about the linguistic behavior of words that are taken to represent superordinate concepts (such as the fact that superordinate concepts can sometimes be named by mass nouns [30]). This research, it must be pointed out, does not establish that superordinate concepts *are* dependent on some linguistic content, but that the question under consideration is itself linguistic: *What sort of linguistic representation will we tend to give for superordinate concepts?*

Since the empirical research in this area is not helpful for the present purpose, I limit myself to giving a plausible description of how superordinates might not depend on linguistic content. Superordinate concepts are representations of the genera for the basic concepts, united by some feature

possessed by all members of the genus. The basic concepts united under a superordinate concept form a similarity group, in much the same way that the basic level concepts are similarity groups between objects. Surely most of us have recognized similarity between abstract ideas without being able to put in words the way in which they are similar. This experience lends some credence to the idea that superordinate concepts might be obtainable without reliance on linguistic meaning. Ahn and Medin [1] [2] have provided a model of concept formation that does not rely on linguistic content; they do not distinguish between basic and superordinate levels in their research, and it appears that the concepts formed by their subjects are basic level concepts (which makes sense, since the basic level concepts are the ones we would expect them to form most readily), but their model is extendable to the superordinate domain, provided similarity does not necessarily appeal to linguistic meaning.[23]

Here, theory theories seem slightly more problematic than either prototype or exemplar theories. For theory theories, concepts are characterized in part through the linguistic relations in which they participate, even at the basic level, since concepts are defined by their role in a broader body of knowledge and beliefs. Exemplar and prototype theories do not face this difficulty. However, only theory theories can obviously and straightforwardly accomodate disjointness. More work needs to be done in order to resolve this tension, which is impossible to undertake in the present paper. Such tensions between theory theories and prototype/exemplar theories are relatively common in the research on concepts, so it should be unsurprising that a tension arises here. Greg Murphy has observed that these tensions seem to indicate that we should want some sort of hybrid between theory theories and prototype/exemplar theories, but research in this area is not yet advanced enough to give a clear answer as to how such a hybrid will work [33].

[23]In fact, in the same way that an inheritance ordering naturally falls out of genus-differentia definition, an inheritance ordering will naturally fall out out of Ahn's and Medin's model.

2.1 Why inheritance networks are the *right* model of lexical semantic content

The short answer is: (1) because inheritance networks already appear in theories of lexical decomposition and (2) because inheritance networks are a model of concepts.

Linguists who endorse theories of lexical decomposition must already be committed to the use of inheritance networks as a model of semantic content, because their atoms—the bearers of semantic content they already endorse—are modeled within theories of lexical decomposition as related to each other by inheritance. It is clear that (1) should be uncontroversial for theorists of lexical decomposition.

It has already been argued that (2) is true. The inheritance and disjointness relations defined above are typicality relations. Recall from Chapter 3 that both exemplar and prototype theories agree that conceptual content is definable in terms of typicality (the difference between the two is the way in which typicality is used to define conceptual content), and typicality is consistent with the theory theory, especially if we agree with Murphy [33] that the correct view of concepts is likely some marriage between the theory theory and either the prototype or exemplar theory.

The isomorphism given in the GDIT is an isomorphism between inheritance networks and the functional role i. This might be interpreted by some readers as implying that the functional role i is the carrier of semantic content. This is not the view being argued for in this thesis (and in fact, I think it is false). Concepts themselves are carriers of semantic content. Empirical research has established that concepts are organized hierarchically. The functional role is a set of coordinates that locates a concept within the hierarchy. By virtue of this, the functional role is a partial set of identity conditions for concepts; more precisely, the functional role is a necessary (but probably not sufficient) condition for identity of concepts. Concepts themselves are the intended interpretation for atoms of semantic content, and the functional role is an abstract mechanism for identifying a particular concept. (This will be illustrated at length in Chapter 5.)

If theorists of lexical decomposition accept that concepts are carriers of semantic content, then they must accept (2) as justification for the correctness of inheritance networks as a model of conceptual content. In fact, most theorists of lexical decomposition do accept that their atoms are to be interpreted as concepts: Jackendoff, Pustejovsky and Levin & Rappaport Hovav have all referred to their atoms as concepts [24] [26] [35]. My contribution is not to say that concepts are the correct intended interpretation for semantic atoms, but to develop this interpretation on behalf of theorists who already intend it.

Chapter 5: The Generative Lexicon, a Case Study

Thus far, we have approached the linguistic issue, i.e. the problem of interpreting the atoms of lexical decomposition, at a very high theoretical level. An abstract mathematical system has been offered to facilitate the interpretation, complete with a theorem that establishes a way in which semantic atoms and concepts can be regarded as identical. Conversely, we have approached the cognitive science issue from a mostly empirical perspective; the mathematical tools introduced here are offered as a model of empirical results about conceptual hierarchy and typicality. Little has been done to *apply* the mathematics to theories of lexical decomposition, in order to gain understanding of linguistic phenomena, and even less has been done to make use of the formalism as a set of theoretical tools to answer questions for cognitive scientists. In this chapter, we will fill in some remaining gaps and demonstrate, via case study, how to *use* inheritance networks to answer real questions. James Pustejovsky's *Generative Lexicon* [35] is the case we will study.

In Section 3, Chapter 2, I have already explained the basics of the Generative Lexicon. In the present chapter, I will expand on the earlier explanation by looking more closely at two of Pustejovsky's generative rules. Where linguistics is concerned, Pustejovsky provides a formal model of how word senses are generated in cases of systematic polysemy, but the model is just that—purely formal. Equipped with the theorem of Chapter 4, we will be able to look "under the hood," so to speak, in order to gain a better understanding of the cognitive mechanism that enables us to understand novel word senses, at least in the cases where word senses are generated according to Pustejovsky's rules.

Cognitive scientists often use linguistic stimuli as a way to study concepts. In some of the experiments outlined in Chapter 3, subjects where primed with sentences and asked to respond. But words (and therefore sentences) are only an intermediary for what cognitive scientists are really interested in, which is concepts. There is very good reason to believe that

the same word can stand for different concepts in different contexts (more on this below), and therefore cognitive scientists are faced with the problem of identifying cases where two instances of a word do stand for the same concept, as well as cases where they do not.

For example, suppose a cognitive scientist studying the influence of hierarchy on inference primes subjects with the following sentences:[24]

(i) Josh is about to begin reading.

(ii) Josh is holding a novel.

The scientist might then ask test subjects to rate their willingness to infer:

(iii) Josh is about to begin a book.

The idea is that we might test whether subjects are willing to accept this inference on the grounds that **novel** is subordinate to **book** in the hierarchy. But this only tests the influence of hierarchy if the senses of "novel" and "book" are senses that correspond to use of the concepts **novel** and **book**. We shall soon see, by means of a slightly modified example, that there is good reason to think that the sense of "book" in (iii) does not correspond to the concept **book**.

The theorem of Chapter 4 is an isomorphism, i.e. a relation-preserving bijection between two domains. These domains, called isomorphs, are regarded as mathematically indistinguishable, and therefore a main value of an isomorphism is that things learned in one domain can be extended into the other. We can answer the linguistic question by looking at how functional roles of concepts vary under generative rules, and we can answer (part of) the cognitive science question by making use of the formalism offered in decompositional lexical semantics. These questions will be answered simultaneously, not due to careful selection of examples, necessarily, but rather

[24]In reality, it would probably be more likely to see cognitive scientists asking subjects to rate their willingness to infer (iii′) "Josh is reading a book." from (ii′) "Josh is reading a novel." I have constructed this artificial example because it is consistent with the problems that will be raised in this chapter. Although somewhat artificial, it is not entirely implausible, since cognitive scientists may want to compare the effects of hierarchy in inferences of different logical form.

because it is a consequence of the theorem that they are in fact the same question approached from different angles.

1 "Josh began the novel."

Let's begin with an example.[25] Consider the sentence "Josh began the novel." The kinds of things we begin are events; beginning is always beginning *to do* something. But the object of "began," in this sentence, is "the novel," which is not an event. Nevertheless, any native speaker of English will find this to be a perfectly comprehensible sentence. We are, therefore, faced with something of a puzzle: our understanding of the concepts represented by the words in this sentence implies that the sentence should be conceptually incoherent, but the sentence is obviously coherent.

We take the sentence to mean is that Josh is beginning to do something to/with/etc. the novel. There is an implicit event that is being begun, which is not syntactically realized, and "the novel" is something on which or with which the implicit event is being performed. For example, two very natural ways of understanding the sentence are "Josh began reading the novel" and "Josh began writing the novel." The conceptual question is whether the concept **novel** is, in fact, an event concept (i.e., a representation of a category of actions, rather than a category of objects)—contrary to intuitions—or whether "the novel" stands for a more complex concept of **reading the novel** or **writing the novel** in this sentence. The linguistic question is how the semantics of "the novel" can allow it to be coherently used as an argument for "began." James Pustejovsky has answered the linguistic question for us, by defining a generative rule that makes use of inheritance networks to resolve the type error. By trading on the mathematical indistinguishability of the aspects of semantic content that are represented in inheritance networks and conceptual hierarchy, we can make use of his formal answer

[25] I have modified things slightly in this section from what was including in our toy example above. This was done because the semantics dealth with here will be more intuitive and natural to deal with than what would be required to deal directly with the sentences used above. The basic ideas, however, are the same, and it should not be too difficult for the reader to see how what is said here relates to what has been said above.

to the linguistic question in answering the conceptual question.

1.1 True Complement Coercion

Recall the following (partial) typed feature structures for "novel" and "begin" from Chapter 2:

$$\begin{bmatrix} \textbf{novel} & \\ \mathcal{A} = & \begin{bmatrix} \text{ARG}_1 = \textbf{x:book} \end{bmatrix} \\ \mathcal{Q} = & \begin{bmatrix} \text{CONST} = \textbf{narrative(x)} \\ \text{FORMAL} = \textbf{x} \\ \text{TELIC} = \textbf{read(y,x)} \\ \text{AGENT} = \textbf{write(y,x)} \end{bmatrix} \end{bmatrix}$$

$$\begin{bmatrix} \textbf{begin} & \\ \mathcal{A} = & \begin{bmatrix} \text{ARG}_1 = \textbf{y:animate\_obj} \\ \text{ARG}_2 = \textbf{z:event}_1 \end{bmatrix} \\ \mathcal{E} = & \begin{bmatrix} \text{E}_1 = \textbf{transition} \\ \ldots \end{bmatrix} \\ \mathcal{Q} = & \begin{bmatrix} \text{FORMAL} = \textbf{P(event}_1\textbf{,y)} \\ \text{AGENT} = \textbf{begin\_act(event}_1\textbf{y,z)} \end{bmatrix} \end{bmatrix}$$

The types given in the typed feature structures are supplied by the following inheritance network (ignore the colors for the time being; we will make use of them shortly):

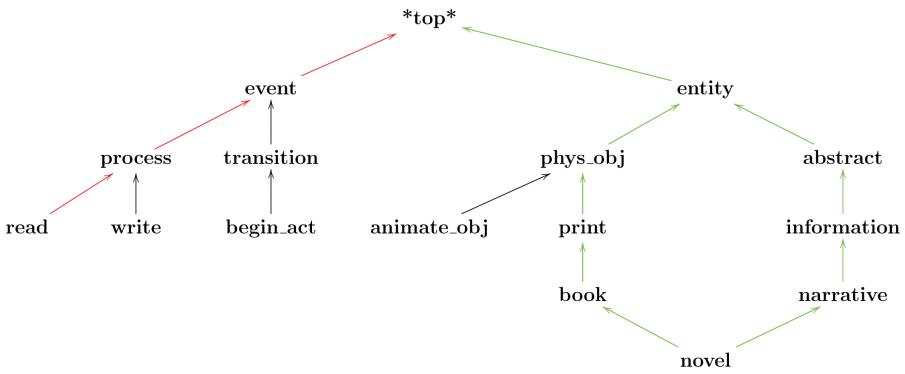

Figure 1

We can read the argument expectations for "begin" directly off of the typed feature structure. In particular, we see that "begin" expects a direct object of type **event**; this type assignment is a representation of our intuition that events are the kinds of things that can be begun. Since "novel" is not a verb, as has been explained, its argument structure specifies the default behavior of "novel" when it is used as an argument; it behaves as an argument of type **book**. But, as can be read off of the inheritance network, **book** does not inherit from **event**, so the sentence "Josh began the novel" entails a violation of the type expectations for arguments of "begin."

According to Pustejovsky, we resolve the type error by applying a generative rule that searches the qualia structure of "begin" for some type that inherits from **event**.[26] In fact, we find that the qualia structure of "novel" includes two qualia, i.e. TELIC and AGENT, whose assigned types inherit from **event**; we read off of the graph that both **read** and **write** inherit from **event**. Coercing "novel" to behave as an argument of type **read** or **write** leads us to understand "Josh began the novel" as "Josh began *to read* the novel" or "Josh began *to write* the novel," respectively.[27]

[26]Pustejovsky does offer a formal definition of this rule. His formalization is quite opaque, however, and I think unpacking it will take us needlessly afield from the main point, since the general idea behind the rule can be understood to the degree required here without the formalization.

[27]In fact, Pustejovsky faces something of a problem here: there are other possible

In fact, type coercion isn't *merely* shifting from one type to another. Pustejovsky states, informally, that "true type coercion *involves* the strict shifting of one type to another specified type ... but embeds the existing type into the resulting type by the proper coercion operation" [35, p. 115]. Formally and computationally, how this is done can be read off of the typed feature structure for "novel". In the argument structure of "novel," we see that the default behavior of "novel" is specified as **x:book**. But this reads "**x** of type **book**," where **x** is a variable that takes the value "novel" in the sentence "Josh began the novel."[28] If we observe that each of the qualia is a function of **x**, we see that the coercion is not to **read**, but to **read(x)**, which is understood as **read(a novel)** (ignoring the definite article).

Pustejovsky's type coercion is closely related to Barsalou's *ad hoc* categories [5]. Barsalou makes the point that some categories are well-established in memory, and therefore we have familiar, well-established concepts to represent those categories. However, in practice we make use of novel categories all the time. For example, Barsalou points out that we might be faced with a situation in which we need to make use of the category "things to take from one's home during a fire." Surely, we have no well-worn concept to represent this category. (Much pity on anyone who does!) Because we lack such a concept, there seem to be no identifiable features among items in the category (such as children, pets, laptop computers, etc.) that serve to bind the category together by family resemblance. Nevertheless, we are able to form and make use of the necessary concepts "on-the-fly," when the need arises.[29]

Likewise, we have a word "novel" with a well-worn sense that corre-

readings of "Josh began the novel." In some unusual cases, we might think of readings such as "Josh began to eat the novel" or "Josh began to skim the novel," and it isn't clear how Pustejovsky's qualia can accommodate the seemingly innumerable readings we might conceive of.

[28]We actually have multiple variables at play in "Josh began the novel." "Josh" is a value taken by the variable **y** in the typed feature structure for "begin," which is designated to be of type **animate_obj**. Since Josh is an animate object, no type error, and therefore no coercion, occurs.

[29]Baralou's theory of ad hoc categories is, incidentally, predicated largely off of Rosch's work. I have no particular opinion on how and whether this observation relates to the project of this thesis; I bring it up merely as a point of interest.

sponds to the concept **novel**. Pustejovsky's type coercion is a model of the mechanism by which we create an *ad hoc* word sense "on-the-fly" when the need arises. The default argument behavior of "novel" is given by the type assignment of the feature ARG_1. Since novel behaves as an argument of type **x:book**, we take this to mean that the default (or "well-worn") sense of "novel" corresponds to some concept that inherits from **book**, i.e. the concept **novel**. When the argument behavior of "novel" is coerced to a new type, we are then faced with the question of whether the new type (read: concept) is identical with the original one.

We might recall from Section 2, Chapter 4, that identity betweeen functional roles is a necessary condition for identity between concepts. The functional role $i(\mathbf{novel})$ is highlighted in green in the above graph. A consistent definition $D_{\mathbf{read}}$ is highlight in red. We can read directly off of the graph that $D_{\mathbf{read}} \notin i(\mathbf{novel})$. But, by the definition of $i(\mathbf{read})$, we know that $D_{\mathbf{read}} \in i(\mathbf{read})$. Therefore, $i(\mathbf{read}) \neq i(\mathbf{novel})$, and therefore the concept being represented by "novel" in "Josh began the novel" cannot be the concept **novel**. We have answered the question of whether the concepts corresponding to each sense of "novel" are identical.

2 Another example: "Mary drives a Honda."

Consider the sentences:

(i) Mary drives a Honda.

(ii) Mary bought stock in Honda.

We only have one word "Honda," which refers to both a category of cars and a car-manufacturing company. We might wonder, however, whether these are both different parts or aspects of the same concept **Honda**. After all, the two domains of reference are clearly related to one another, and in fact the very reason we use the word "Honda" to refer to the category of cars is because they are made by Honda, the car manufacturer. Does "Honda" correspond to a single concept, which is a representation of a category that

includes both the cars and the manufacturer? Or do these two uses of "Honda" correspond to distinct concepts?

Some typed feature structures:

$$\begin{bmatrix} \textbf{drive} & \\ \mathcal{A} = & \begin{bmatrix} \mathrm{ARG}_1 = \mathbf{x} : \textbf{human} \\ \mathrm{ARG}_2 = \mathbf{y} : \textbf{vehicle} \end{bmatrix} \\ \mathcal{E} = & \begin{bmatrix} \mathrm{E}_1 = \mathbf{e}_1 : \textbf{process} \\ \mathrm{E}_2 = \mathbf{e}_2 : \textbf{process} \\ \mathrm{RESTR} = <_{\circ_\propto} \end{bmatrix} \\ \mathcal{Q} = & \begin{bmatrix} \mathrm{FORMAL} = \textbf{move}(\mathbf{e}_2, \mathbf{y}) \\ \mathrm{AGENT} = \textbf{drive\_act}(\mathbf{e}_1, \mathbf{x}, \mathbf{y}) \end{bmatrix} \end{bmatrix}$$

$$\begin{bmatrix} \textbf{buy stock} & \\ \mathcal{A} = & \begin{bmatrix} \mathrm{ARG}_1 = \mathbf{x} : \textbf{human} \mathrm{ARG}_2 = \mathbf{y} : \textbf{coporation} \end{bmatrix} \\ \mathcal{E} = & \begin{bmatrix} \mathrm{E}_1 = \mathbf{e}_1 : \textbf{transition} \end{bmatrix} \\ \mathcal{Q} = & \begin{bmatrix} \mathrm{FORMAL} = \textbf{transfer\_ownership}(\mathbf{e}, \mathbf{x}, \textbf{stock\_share}, \mathbf{y}) \\ \mathrm{TELIC} = \textbf{gain}(\mathbf{e}, \mathbf{x}, \textbf{profit}) \\ \mathrm{AGENT} = \textbf{trade}(\mathbf{e}, \mathbf{x}, \mathbf{y}) \end{bmatrix} \end{bmatrix}$$

I have omitted feature structures for the senses of "Honda" since it isn't clear or important to understand whether and/or how one sense of "Honda" is generated from the other by type coercion. Creating a feature structure for "Honda" would require ironing out these issues and identifying an appropriate generative rule, which would be messy (to say the least) and would take us too far afield from the present task. Everything we need to know can be seen from the following inhertance hierarchy, which provides the types we need:

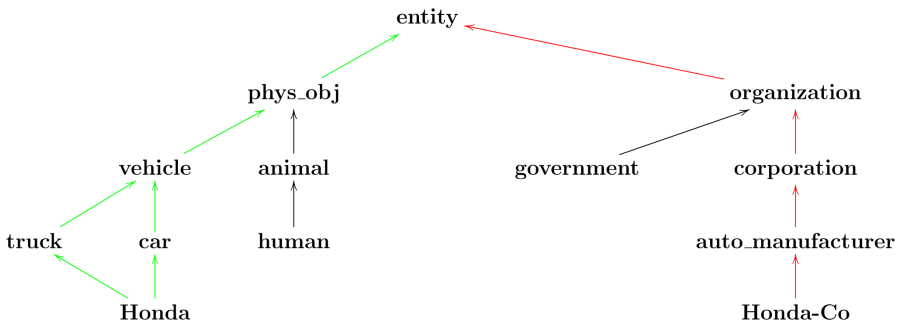

Figure 2

We know from the argument structures of "drive" and "buy stock" that the sense of "Honda" in (i) must display argument behavior of type **vehicle**, while the sense of "Honda" in (ii) must display argument behavior of type **corporation**. Again, we can read directly off of the graph that $D_{\textbf{Honda-Co}} \notin i(\textbf{Honda})$, and therefore **Honda-Co** and **Honda** cannot be the same concept.

3 Some final remarks

It is important to realize that the diagnostic demonstrated here is only a diagnostic for non-identity between concepts. Since the framework of inheritance networks and consistent definitions does not provide a sufficient condition for identity of concepts, we cannot establish that two concepts α and β *are* identical unless we are able to check that *every possible* $D_\alpha \in i(\beta)$. This would only be possible, in principle, if the number of possible D_α's is finite, and there is no reason to suppose that this is true. (I suspect that it is not.)

Also, the Theorem and the conceptual interpretation of semantic atoms have no consequence for the formalism in GL, i.e. the generative rules, the features, and the structure of lexical entries are unaffected by the interpretation. (Or for the formalism in any theory of lexical decomposition.) They do have consequences for the meta-theories of lexical decomposition,

however, in that they give a deeper account of *why* such theories have the features they do, by interpreting the atoms as concepts. In the case of GL, we are now equipped to understand type coercion as remapping the relation between concepts and words (in cases where the diagnostic justifies this understanding). Also, since cognitive scientists have already developed a robust set of empirical tools for studying conceptual hierarchy, we can now make use of those tools to determine which inhertance networks are admissable in theories of lexical decomposition.

For theories of lexical decomposition that dictate specific inheritance relations, we can turn toward cognitive science to determine whether these relations are empirically justified. For theories of lexical decomposition like GL, which leave wide open the question of what our inheritance networks should look like, we now have an empirical means for discovering the right inheritance networks to use. The details of how this can be done are outside the scope of this thesis. In fact, obtaining the same results I did when applying the diagnostic above relies crucially on having set up the inheritance hierarchies in the way that I have. Getting the right inheritance network is a matter of getting the right hierarchy relations, and so the diagnostic always "faces the tribunal of sense experience," to borrow some of Quine's words, in that empirical discoveries always have the final say about how to set up the inheritance network. Note: Since the point of these questions is that distinct types can represent the same concept, provided they have the same functional roles, including **Honda** and **Honda-Co** as separate atoms does not "stack the deck" in favor of regarding them as distinct concepts. This does not mean that, in practice, we must engage in an empirical study of hierarchy at the outset for every inheritance network we want to define. For the most part, our intuitions about what is a plausible hierarchy should suffice—it would seem very strange indeed to place **animal** subordinate to **human**—but empirical data can serve as an arbiter, when challenges against a particular network are raised.

4 Summary

We can make use of the isomorphism to study semantic atoms by looking at concepts and vice versa. Taking GL as a case study allows us to see how the functional role i can provide us with diagnostics to determine whether two distinct uses of a word represent the same concept. In making use of diagnostics to establish that the functional roles of the concepts being represented by words are not identical, we can infer that the concepts are not identical. However, the framework of inheritance networks affords us with no known diagnostic to establish that two concepts *are* identical. Such a diagnostic would require possession of a sufficient condition for identity of concepts, but the tools defined here can only provide us with a necessary condition for identity of concepts.

Chapter 6: Conclusion

The argument presented here can be summarized as follows:[30]

(i) Theories of lexical decomposition break word meaning down into primitive semantic atoms organized in a type-inheritance network.

(ii) If advocates of lexical decomposition wish to have a complete, coherent account of lexical semantics, the atoms must be interpreted in some way that explains how they hook up with the world to acquire their content.

(iii) Any viable interpretation will explain why type-inheritance networks appear in all theories of lexical decomposition.

(iv) Interpreting atoms as concepts explains why type-inheritance networks appear in all theories of lexical decomposition.

  (a) The classical view of concepts holds that genus-differentia definitions (I) provide necessary and sufficient conditions for category membership, while also (R) positioning concepts within a hierarchy.

  (b) Empirical research has established that category membership is determined by typicality, rather than necessary and sufficient conditions.

  (c) Empirical research has also established that concepts are organized hierarchically.

  (d) Inheritance networks are a model of conceptual hierarchy that accounts for category membership determined by typicality.

(v) It is generally agreed that concepts hook up with the world in some way.

[30]This is a *logical* reconstruction of the argument offered in this thesis, and it does not map exactly onto the order in which premises were presented.

(vi) Therefore, concepts are a viable interpretation of semantic atoms that will allow advocates of lexical decomposition to move toward a complete, coherent account of lexical semantics.

The framework presented here only models two semantic relations, namely disjointness and the relation Δ. But many other semantic ordering relations exist, e.g. meronymy and holonymy (relations between parts and a whole). If we wish to use inheritance networks as a total model of lexical semantics, we will have to add models of these other relations to the network.

Nevertheless, there are some important applications of the theory as it stands. In particular, the theory has been used to answer some questions about whether two instances of a word correspond to the same or distinct concepts. This was done by means of a diagnostic that trades on the observation that functional roles provide a necessary condition for identity between concepts. This framework, then, provides a set of mathematical tools that can be applied to analysis of concepts, and future research can perhaps identify additional diagnostics or other applications of the framework to different kinds of conceptual questions.

Many theories of lexical decomposition leave wide open the question of which atoms should be included in an inheritance network and which inheritance relations hold between them. Very little guidance is given within the theories about how to go about identifying an appropriate inheritance network to use. This vagueness carries over into technology: computational resources such as VerbNet are based explicitly on theories of lexical decomposition. There is significant debate among researchers who work on these implementations regarding when to *lump* two classes together into the same type and when to *split* them into distinct types. By interpreting types as concepts, we have a well-justified criterion for determining, at least, when to split. When a diagnostic shows that types correspond to distinct concepts, we should regard them as separate atoms in the inheritance network, and therefore it stands to reason that we should split them in the computational lexicon (issues pertaining to the computational efficiency of the implementation notwithstanding).

This thesis was facilitated by proving that inhertance networks are isomorphic to a new mathematical structure I have called the *functional role* of an atom. Functional roles provide a necessary condition for identity between concepts, but they do not provide a sufficient condition. In order to complete the conceptual interpretation of atoms, more research must be done to discover a sufficient condition. I am not prepared to offer anything resembling a sufficient condition, and there is no reason to suppose that one will so neatly fall out of defining a new mathematical concept as we have seen here. Concepts appear to be inherently fuzzy at the edges. Until we are able to better understand this fuzziness, it would seem, a clear sufficient condition for identity of concepts will have to lie in wait. It may be that no such condition is possible. After all, this question is closely allied to the question, "What is a member of the category of concepts?" But **concept** is itself a concept, and it is an important part of the theory presented here that the kinds of necessary and sufficient conditions we are asking for do not, in general, exist.

Appendix: Formal Definitions and Proofs

(Inheritance Network) An inheritance network $\mathcal{I}$ is a triple $\langle \mathcal{B}, \sqsubseteq, \# \rangle$ where:

- $\mathcal{B}$ is a finite set of basic elements
- $\sqsubseteq \subseteq \mathcal{B} \times \mathcal{B}$ is the basic *inheritance* relation
- $\# \subseteq \mathcal{B} \times \mathcal{B}$ is the basic *disjointness* relation

(Inheritance/Disjointness) The *inheritance* relation $\sqsubseteq^* \subseteq \mathcal{B} \times \mathcal{B}$ is the smallest such that:

- $P \sqsubseteq^* P$ (Reflexivity)
- if $P \sqsubseteq Q$ and $Q \sqsubseteq^* R$ then $P \sqsubseteq^* R$ (Transitivity)

The *disjointness* relation $\#^* \subseteq \mathcal{B} \times \mathcal{B}$ is the smallest such that:

- if $P\#Q$ or $Q\#P$ then $P\#^*Q$ (Symmetry)
- if $P \sqsubseteq^* Q$ and $Q\#^*R$ then $P\#^*R$ (Chaining)

(Consistent Definition)(cf. [10], *Conjunctive Concept*) A set $D \subseteq \mathcal{B}$ is a *consistent definition* on $\mathcal{B}$ iff:

1. For all $x, y \in D$, it is not the case that $x\#y$.
2. For all $x \in D$, $y \in \mathcal{B}$, if $x \sqsubseteq y$, then $y \in D$.

Lemma 3.1. For $a, b \in \mathcal{B}$ such that it is not the case that $a\#b$, let $D_a = \{c | a \sqsubseteq c\}$ and $D_b = \{c | b \sqsubseteq c\}$. $D^* = D_a \cup D_b$ is a consistent definition on $\mathcal{B}$.

Proof. Because of the reflexivity of $\sqsubseteq$, it is obvious that D^* meets condition (2) above. Suppose (1) does not hold of D^*, i.e. there exists $x, y \in D^*$ such that $x\#y$. Then, by chaining we know that $a\#b$, since for all $x \in D^*$, either $a \sqsubseteq x$ or $b \sqsubseteq x$. But we have already said that it is not the case that

$a\#b$, so (1) must hold of D^*. Therefore D^* is a consistent definition on $\mathcal{B}$. □

Lemma 3.2 Let $a, b \in \mathcal{B}$ be such that it is not the case that $a \sqsubseteq b$, and let $\mathcal{F} = \{D | D \text{ is a consistent definition}\}$. If a is consistently definable, then there exists some $D^a_{\neg b} \in \mathcal{F}$ such that $a \in D^a_{\neg b}$ and $b \notin D^a_{\neg b}$.

Proof. Assume a is consistently definable. Then there exists some $D^a \in \mathcal{F}$ such that $a \in D^a$. Either $a\#b$ or not. Suppose $a\#b$. Then $b \notin D^a$, by condition (1) for consistent definition. Suppose it is not the case that $a\#b$. Then there is no relation between a and b, which means that if D^a is a consistent definition, then $D^a_{\neg b} = D^a \setminus \{b\}$ is also a consistent definition. By the definition of $D^a_{\neg b}$, $b \notin D^a_{\neg b}$ and $a \in D^a_{\neg b}$. □

The genus-differentia inheritance theorem (GDIT): Given $\sqsubseteq$ and $\#$, such that $\sqsubseteq$ is inclusion and $\#$ is disjointness over $\mathcal{B}$, a consistently definable set of atoms, there exists a set $\mathcal{F}$, a set $\text{Rep}_{\mathcal{B}} \subseteq \mathcal{P}(\mathcal{F})$ (where $\mathcal{P}(\mathcal{F})$ is the power set of $\mathcal{F}$), and a function $i : \mathcal{B} \to \text{Rep}_{\mathcal{B}}$ such that $a \sqsubseteq b \Leftrightarrow i(a) \subseteq i(b)$ and $a\#b \Leftrightarrow i(a) \cap i(b) = \varnothing$.

Proof. Let $\mathcal{F} = \{D | D \text{ is a consistent definition on } \mathcal{B}\}$, and let $i : \mathcal{B} \to \mathcal{P}(\mathcal{F})$ be a function such that $i(x) = \{D \in \mathcal{F} | x \in D\}$.

Assume $a \sqsubseteq b$. Suppose $D \in i(a)$. Then $a \in D$, by the definition of i, which means that $b \in D$ by (2). So $D \in i(b)$, by the definition of i. We therefore have $a \sqsubseteq b \Rightarrow i(a) \subseteq i(b)$.

Assume that it is not the case that $a \sqsubseteq b$. Then, by (3.2) there exists some $D^a_{\neg b} \in \mathcal{F}$ such that $a \in D^a_{\neg b}$ and $b \notin D^a_{\neg b}$. Then $D^a_{\neg b} \in i(a)$, but $D^a_{\neg b} \notin i(b)$, which means that $i(a) \nsubseteq i(b)$. So we have: If it is not the case that $a \sqsubseteq b$, then it is not the case that $i(a) \subseteq i(b)$, which contrapositis to $i(a) \subseteq i(b) \Rightarrow a \sqsubseteq b$.

We have therefore shown that $a \sqsubseteq b \Leftrightarrow i(a) \subseteq i(b)$, and we now turn to proving that $a\#b \Leftrightarrow i(a) \cap i(b) = \varnothing$.

Assume $a\#b$. Now suppose $i(a) \cap i(b) \neq \varnothing$. Then there exists some $D \in \mathcal{F}$ such that $a \in D$ and $b \in D$. But, by (1), this cannot be the case. So, $a\#b \Rightarrow i(a) \cap i(b) = \varnothing$.

Assume $i(a) \cap i(b) = \varnothing$. Then there exists no $D \in \mathcal{F}$ such that $a \in D$ and $b \in D$. Now suppose it is not the case that $a\#b$. Let D_a, D_b, and D^*

be defined as in (3.1). Then D^* is a consistent definition on $\mathcal{B}$. But, by the reflexivity of $\sqsubseteq$, $a \in D_a$ and $b \in D_b$, which means that $a, b \in D^*$. So there is some $D \in \mathcal{F}$ such that $a \in D$ and $b \in D$, a contradiction. Therefore, $i(a) \cap i(b) = \varnothing \Rightarrow a \# b$.

We have therefore shown that $i(a) \cap i(b) = \varnothing \Leftrightarrow a \# b$. $\square$

References

[1] Woo-Kyoung Ahn. *A two-stage model of category construction*. Ph.D. dissertation, University of Illinois, Urbana-Champagne, 1990.

[2] Woo-Kyoung Ahn and Douglas L. Medin. A two-stage model of category construction. *Cognitive Science*, 16(1):81—121, 1992.

[3] Aristotle. *Categories and De Interpretatione*. Clarendon Press, 1975.

[4] Aristotle. *Metaphysics*. Green Lion Press, 1999.

[5] Lawrence W. Barsalou. Ad hoc categories. *Memory & Cognition*, 11(3):211—227, 1983.

[6] B. Berlin. *Ethnobiological Classification: Principles of Categorization of Plants and Animals in Traditional Societies*. Princeton University Press, Princeton, NJ, 1992.

[7] R. Brown. How shall a thing be called? *Pychological Review*, 65:14–21, 1958.

[8] J. Carbonell. Default reasoning and inheritance mechanisms on type hierarchies. In *Computer Science Department*. 1980.

[9] Susan Carey. *The Origin of Concepts*. Oxford University Press, Oxford, 2009.

[10] Bob Carpenter. Inclusion, disjointness and choice: the logic of linguistic classification. In *ACL '91 proceedings of the 29th annual meeting on Assocation for Computational Linguistics*. Association for Computational Linguistics, 1991.

[11] Bob Carpenter. *The Logic of Typed Feature Structures*. Number 32 in Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, Cambridge, 1992.

[12] D. Cisse and D.C. Heth. An evaluation of differential encoding and feature overlap accounts of typicality effects in free recall. *Canadian Journal of Psychology*, 43:359–368, 1989.

[13] A.M. Collins and M.R. Quillian. Retrieval time from semantic memory. *Journal of Verbal Learning and Verbal Behavior*, 8:432–438, 1969.

[14] Ann Copestake. *Implementing Typed Feature Structure Grammars*. CSLI Publications, Stanford, 2002.

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

[16] Jerry Fodor and Ernst Lepore. The emptiness of the lexicon: reflections on james pustejovsky's the generative lexicon. *Linguistic Inquiry*, 29(2):269—288, 1998.

[17] Jerry Fodor and Ernst Lepore. Impossible words? *Linguistic Inquiry*, 30(3):445—453, 1999.

[18] Jerry Fodor and Ernst Lepore. Morphemes matter; the continuing case against lexical decomposition. *RuCSS Tech Report*, 34, 2000.

[19] Jerry Fodor and Ernst Lepore. Impossible words: A reply to Kent Johnson. *Mind & Language*, 20(3):353—356, 2005.

[20] H. Gastgeb, M. Strauss, and N. Minshew. Do individuals with autism process categories differently? the effect of typicality and development. *Child Development*, 77:1717–1729, 2006.

[21] M. Greenberg and D. Bjorklund. Category typicality in free recall: Effects of feature overlap or differential category encoding? *Journal of Experimental Psychology*, 7:145–147, 1981.

[22] B. Inhelder and J. Piaget. *The early grown of logic in the child: classification and seriation*. Routledge, London, 1964.

[23] R. Jackendoff. *Semantic interpretation in generative grammar*. MIT Press, Cambridge, 1972.

[24] Ray Jackendoff. *Semantics and Cognition*. The MIT Press, Cambridge, MA, 1983.

[25] B. Levin and M. Rappaport Havov. Building verb meanings. In M. Butt and W. Geuder, editors, *The Projection of Arguments: Lexical and Compositional Factors*. CSLI Publications, Stanford, CA, 1998.

[26] B. Levin and M. Rappaport Havov. Lexical conceptual structure. In K. von Heusinger, C. Maienborn, and P. Portner, editors, *Semantics: An International Handbook of Natural Language Meaning*. Mouton de Gruyter, Berlin, 2008.

[27] David Lewis. General semantics. *Synthese*, 22(1):18—67, 1970.

[28] E. Loftus. Spreading activiation within semantic categories: Comments on rosch's "cognitive representation of semantic categories". *Journal of Experimental Psychology: General*, 104:234–230, 1975.

[29] B. C. Malt. Category coherence in cross-cultural perpective. *Cognitive Psychology*, 29:85–148, 1995.

[30] E. M. Markman. Why superordinate category terms can be mass nouns. *Cognition*, 19, 1985.

[31] J.M. Moravcsik. Aitia as generative factor in Aristotle's philosophy. *Journal of Philosophy*, 14:622–636, 1975.

[32] G. L. Murphy. Cue vailidity and levels of categorization. *Psychological Bulletin*, 91, 1982.

[33] G. L. Murphy. *The Big Book of Concepts*. The MIT Press, Cambridge, MA, 2002.

[34] G. L. Murphy and Brownell H. H. Category differentiation in object recognition: Typicality constraints on the basic category advantage. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 11, 1985.

[35] James Pustejovsky. *The Generative Lexicon*. The MIT Press, 1998.

[36] L. Rips, E. Shoben, and Smith E. Semantic distance and the verification of semantic relations. *Journal of Verbal Learning and Verbal Behavior*, 12:1–20, 1973.

[37] E. Rosch. Natural categories. *Cognitive Psychology*, 4:328–350, 1973.

[38] E. Rosch. Cognitive representations of semantic categories. *Journal of Experimental Psychology: General*, 104:192–233, 1975.

[39] E. Rosch. Principles of categorization. In E. Rosch and B. B. Lloyd, editors, *Cognition and Categorization*. Erlbaum, Hillsdale, NJ, 1978.

[40] E. Rosch and C. Mervis. Family resemblances: Studies in the internal structure of categories. *Cognitive Psychology*, 7:573–605, 1975.

[41] Antonio Sanfillipo. LKB encoding of lexical knowledge. In Ted Briscoe, Valeria De Paiva, and Ann Copestake, editors, *Inheritance, Defaults, and the Lexicon*, Studies in Natural Language Processing, pages 190—222. Cambridge University Press, Cambridge, 1993.

[42] C. Simpson, E. Rosch, and Miller R. S. Structural bases of typicality effects. *Journal of Experimental Psychology: Human Perception and Performance*, 2:491–502, 1976.

[43] Steve Sloman. Categorical inference is not a tree: The myth of inheritance hierachies. *Cognitive Psychology*, 35:1—33, 1998.

[44] Wittgenstein. *Philosophical Investigations*. Blackwell, Oxford, 1953.