Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

Research
Report
RR-91-30

Inheritance and Complementation: A Case Study of *Easy* Adjectives and Related Nouns

Daniel Flickinger, John Nerbonne

September 1991

Deutsches Forschungszentrum für Künstliche Intelligenz
GmbH

Postfach 20 80
D-6750 Kaiserslautern, FRG
Tel.: (+49 631) 205-3211/13
Fax: (+49 631) 205-3210

Stuhlsatzenhausweg 3
D-6600 Saarbrücken 11, FRG
Tel.: (+49 681) 302-5252
Fax: (+49 681) 302-5341

Deutsches Forschungszentrum für Künstliche Intelligenz

The German Research Center for Artificial Intelligence (Deutsches Forschungszentrum für Künstliche Intelligenz, DFKI) with sites in Kaiserslautern und Saarbrücken is a non-profit organization which was founded in 1988 by the shareholder companies ADV/Orga, AEG, IBM, Insiders, Fraunhofer Gesellschaft, GMD, Krupp-Atlas, Mannesmann-Kienzle, Philips, Siemens and Siemens-Nixdorf. Research projects conducted at the DFKI are funded by the German Ministry for Research and Technology, by the shareholder companies, or by other industrial contracts.

The DFKI conducts application-oriented basic research in the field of artificial intelligence and other related subfields of computer science. The overall goal is to construct *systems with technical knowledge and common sense* which - by using AI methods - implement a problem solution for a selected application area. Currently, there are the following research areas at the DFKI:

❑ Intelligent Engineering Systems
❑ Intelligent User Interfaces
❑ Intelligent Communication Networks
❑ Intelligent Cooperative Systems.

The DFKI strives at making its research results available to the scientific community. There exist many contacts to domestic and foreign research institutions, both in academy and industry. The DFKI hosts technology transfer workshops for shareholders and other interested groups in order to inform about the current state of research.

From its beginning, the DFKI has provided an attractive working environment for AI researchers from Germany and from all over the world. The goal is to have a staff of about 100 researchers at the end of the building-up phase.

Prof. Dr. Gerhard Barth
Director

Inheritance and Complementation: A Case Study of *Easy* Adjectives and Related Nouns

Daniel Flickinger, John Nerbonne

DFKI-RR-91-30

This work has been supported by a grant from The Federal Ministry for Research and Technology (FKZ ITW-9002 0).

Inheritance and Complementation: A Case Study of *Easy* Adjectives and Related Nouns

Dan Flickinger and John Nerbonne

Hewlett-Packard Laboratories
1501 Page Mill Road
Palo Alto, CA 94304-1126
USA
flickinger@hp.com

DFKI, Project DISCO
Stuhlsatzenhausweg 3
D-66 Saarbrücken 11
Germany
nerbonne@dfki.uni-sb.de

Abstract: Mechanisms for representing lexically the bulk of syntactic and semantic information for a language have been under active development, as is evident in the recent studies contained in this volume. Our study serves to highlight some of the most useful tools available for structured lexical representation, in particular, (multiple) inheritance, default specification, and lexical rules. It then illustrates the value of these mechanisms in illuminating one corner of the lexicon involving an unusual kind of complementation among a group of adjectives exemplified by *easy*. The virtures of the structured lexicon are its succinctness and its tendency to highlight significant clusters of linguistic properties. From its succinctness follow two practical advantages, namely its ease of maintenance and modifiability. In order to suggest how important these may be practically, we extend the analysis of adjectival complementation in several directions. These further illustrate how the use of inheritance in lexical representation permits exact and explicit characterizations of phenomena in the language under study. We demonstrate how the use of the mechanisms employed in the analysis of *easy* enable us to give a unified account of related phenomena featuring nouns like pleasure, and even the adverbs (adjectival specifiers) *too* and *enough*. Along the way we motivate some elaborations of the Head-Driven Phrase Structure Grammar (HPSG) framework in which we couch our analysis, and offer several avenues for further study of this part of the English lexicon.

Keywords: natural language, inheritance networks, lexicon, default reasoning

Table of Contents

Inheritance and Complementation: A Case Study of *Easy* Adjectives and Related Nouns

Dan Flickinger
Hewlett-Packard Laboratories
1501 Page Mill Road
Palo Alto, California 94304-1126
flickinger@hplabs.hp.com

John Nerbonne [*]
Deutsches Forschungszentrum für Künstliche Intelligenz
Stuhlsatzenhausweg 3
D-6600 Saarbrücken 11, Germany
nerbonne@dfki.uni-sb.de

Abstract

Mechanisms for representing lexically the bulk of syntactic and semantic information for a language have been under active development, as is evident in the recent studies contained in this volume. Our study serves to highlight some of the most useful tools available for structured lexical representation, in particular, (multiple) inheritance, default specification, and lexical rules. It then illustrates the value of these mechanisms in illuminating one corner of the lexicon involving an unusual kind of complementation among a group of adjectives exemplified by *easy*. The virtues of the structured lexicon are its succinctness and its tendency to highlight significant clusters of linguistic properties. From its succinctness follow two practical advantages, namely its ease of maintenance and modification. In order to suggest how important these may be practically, we extend the analysis of adjectival complementation in several directions. These further illustrate how the use of inheritance in lexical representation permits exact and explicit characterizations of phenomena in the language under study. We demonstrate how the use of the mechanisms employed in the analysis of *easy* enable us to give a unified account of related phenomena featuring nouns like *pleasure*, and even the adverbs (adjectival specifiers) *too* and *enough*. Along the way we motivate some elaborations of the HPSG (Head-Driven Phrase Structure Grammar) framework in which we couch our analysis, and offer several avenues for further study of this part of the English lexicon.

1 Introduction

The lexicon is a large and complex set of information about the words used in a grammar or natural language processing system. Its importance has become more central in the research of the past decade, which has seen the rise of radically LEXICALIZED theories such as HEAD-DRIVEN PHRASE STRUCTURE GRAMMAR (HPSG), in which phrase structure rules play a vestigial role.

[*] We are indebted to Mark Gawron, Masayo Iida, Bill Ladusaw, Joachim Laubsch, Carl Pollard and Tom Wasow for frequent conversations about this analysis. We are also grateful to Anthony Kroch, the participants at the Tilburg Workshop on Inheritance in Natural Language Processing, and three referees for further comments. This work was partially supported by a research grant, ITW 9002 0, from the German Bundesministerium für Forschung und Technologie to the DFKI DISCO project.

Newer theories place increasingly high demands on lexical representation. A simple calculation may illustrate the quandary of lexical representation: feature systems for contemporary systems normally distinguish at least 30 features (while 40 or 50 is not rare). The number of values a feature takes ranges from 2 to the number of categories (more exactly, to the number of sequences or sets of a small size, where all the members of the sequence, etc. are categories). Under the undoubtedly optimistic assumption that feature value ranges could be reduced to booleans, we still are faced with $2^{30} = 10^9$ feature combinations—whose individual representation is clearly to be avoided, not "solved".[1] The natural tack is certainly to represent just the categories actually used in the vocabulary, but this could incur a good deal of redundancy if it meant that each feature combination were represented separately on each word.

The STRUCTURED OR HIERARCHICAL lexicon solves this difficulty (cf. Flickinger, Pollard and Wasow, 1985 and Flickinger, 1987). In structured lexicons, word classes may stand in a relationship of inheritance to one another, in which case the properties of the bequeathing class accrue automatically to the inheriting class. Once we allow that a single class may be heir to more than one bequeathing class, we allow, in principle, that no word class property EVER need be examined more than once. Thus we eliminate one central source of redundancy in lexical specification. One of the goals of this paper is to motivate the use of inheritance in lexical specification. To do this, we take a narrowly circumscribed phenomenon in English grammar—that of vp-complement-taking adjectives, as in *hard + to deliver*—and spell out the lexical specifications which a thorough treatment demands. The sheer complexity of these specifications cries out for a redundancy-eliminating approach, and we propose a structured lexicon treatment. The grammatical analysis not only serves to motivate the general approach, it also illustrates several key issues in the design of structured lexicons, such as the use of DEFAULT INHERITANCE, the need for lexical rules, and the range of phenomena amenable to this sort of treatment.

The goals of this paper are to introduce the structured lexicon in a fairly simple form, to motivate its basic theoretical device, that of inheritance, with a real example taken from an existing system, and finally to show how the elimination of redundancy achieved with the structured lexicon aids in maintaining the lexicon. We argue for improved maintainability by examining concrete extensions and potential modifications of the grammatical description provided. We turn now to a brief characterization of this phenomenon.

The rich collection of syntactic and semantic phenomena exhibited by a familiar group of adjectives like *tough* and *easy* present a challenge to those who seek to provide explicit formal characterizations of linguistic properties. We offer here a detailed description of the properties of these adjectives, involving optional and obligatory complementation, control, long-distance dependence, optional modification, and specification. The purpose of this description here is not the linguistic analysis itself (which we find interesting, nonetheless), but rather its use in demonstrating the practical utility of inheritance as a tool for linguistic description, and also the predictive analytical power that inheritance affords in the study of the lexicon. In illustration of the latter, we extend our analysis of *easy* adjectives to a similar group of nouns such as *pleasure*, and then to the unusual adverbs *too* and *enough*, which function as specifiers in adjectival gradation.

The fundamental data are illustrated in (1); examples such as these have not attracted attention in computational linguistics, even if they have often appeared in studies within the generative framework. An early discussion of them is found in Miller and Chomsky (1963), with a score and more of additional studies published in the years since. Most of the salient properties of these adjectives have already been brought to light, but in piecemeal fashion and most often as part of a larger debate about the nature of unbounded dependencies, where detailed syntactic and semantic characterizations of these missing object constructions proved less important.[2] We return to the

[1] Cf. Gazdar et al., 1985, Appendix for a small grammar which nonetheless exceeds the size speculated on here.

[2] Related work in theoretical and descriptive linguistics includes Chomsky (1965), Rosenbaum (1967), Ross (1967), Postal (1971), Bresnan (1971), Chomsky (1973), Lasnik and Fiengo (1974), Jackendoff (1975), Chomsky (1977), Fodor (1978), Brame (1979), Nanni (1980), Schachter (1981), Jacobson (1982,pp.221-23), Sag (1982), Maling and Zaenen (1982, pp.253-54), Kaplan and Bresnan (1982, pp.255-63), Culicover and Wilkins (1984), Jacobson (1984), Gazdar, Klein, Pullum, and Sag (1985, pp.150-52) (hence: GKPS), Jacobson (1990), Jones (1990), Bayer (1990), and Hukari and Levine (1991). None of these works have attempted a thorough descriptive analysis of the

characteristic properties of these adjectives in Section 3, where they are catalogued and given formal representation.

(1) a. Bill is easy to talk to.
b. It is easy to talk to Bill.
c. Bill is easy for Mary to talk to.
d. It is easy for Mary to talk to Bill.

We chose this phenomenon as a vehicle to recommend lexical inheritance because it illustrates a wide range of grammatical phenomena, all of which make demands on lexical resources (at least in the lexicalized grammar in which the analysis is framed). In addition to the grammatical demands, the data justify the use of a lexical rule (derivational rule) to relate pairs such as (a) and (b) in (1)—so we shall argue at any rate—thus illustrating a further inheritance-like relationship in the lexicon.

The remainder of the paper is structured as follows: Section 2 summarizes the aspects of HPSG which are important to our proposal, and Section 3 develops the fundamental analysis, which Section 4 illustrates in a series of analytical "snapshots" of a single example. Section 5 suggests extensions of the fundamental analysis, especially to further lexical classes (developing the argument that structured lexicons are easily maintained and extended), and a final section summarizes and suggests directions for future work. Appendix A presents the framework for lexical description developed in Flickinger et al. (1985) and Flickinger (1987). The framework is convenient for feature-based grammars, but it allows the specification of other lexical properties as well. This Appendix presents a notation which is precise while avoiding redundancy, e.g., in characterizing the kinds of complements that these adjectives permit, and in expressing the relationships that hold between pairs like the *easy* of (1a) and that of (1b). Since a fundamental claim of hierarchical lexicons is that they eliminate redundancy and thus improve modifiability, there is a second appendix, Appendix B, which demonstrates the modifiability of the structured lexicon.

2 Grammatical Theory

The phenomena involved in the analysis of the *easy* adjective class illustrate (obligatory and optional) subcategorization, control, long-distance dependence, optional modification, and specification (the last in its interaction with adjectival gradation with *too* and *enough*). As such, it represents an excellent demonstration vehicle for the lexical demands of grammatical analysis. Our analysis is formulated within HEAD-DRIVEN PHRASE STRUCTURE GRAMMAR (HPSG), the grammatical theory developed by Carl Pollard and Ivan Sag during the mid and late 1980's. See Pollard (1984), Pollard (1985), Pollard and Sag (1987), Pollard (1988), Pollard and Sag (1988), Pollard (1989), and Pollard and Sag (1991). As the lengthy list of publications might suggest, this grammatical theory is well enough documented so that we may restrict our remarks here to the distinctive characteristics of the assumptions used here. We assume familiarity with feature-based grammars and basic familiarity with HPSG as well.

In all linguistic theories there is a division of labor between grammatical rules and the lexicon, and this concerns the amount of information contained in each. At the rule-based extreme lie non-feature-based context-free grammars, where the lexicon merely links lexical items to nonterminals; in these grammars it is indeed customary to view the lexicon as a set of unary rules. The grammatical rules thus encode effectively all linguistic information. At the lexical extreme we find feature-based categorial grammars, which allow function-argument application as the only grammatical rule. Here the lexicon bears the burden of encoding linguistic information, and the

range of data we address here, though we are of course indebted to these studies for much of the data and many of the generalizations we seek to express. In particular, our account is consistent with the brief GENERALIZED PHRASE STRUCTURE GRAMMAR (GPSG) analysis of these adjectives given in GKPS (1985, pp.150-2) though we embrace a larger range of data and extend the analysis to related nouns, a topic rarely discussed since its introduction by Lasnik and Fiengo (1974).

contribution of rules is marginal. We emphasize that HPSG is found very close to the lexical extreme, because this highlights the significance of the present work—HPSG is a framework whose lexical demands are very nearly maximal.

SUBCATEGORIZATION information is lexically based in HPSG, much as it is in Categorial Grammar (Bach, 1988). Grammatical heads specify the syntactic and semantic restrictions they impose on their complements and adjuncts. For example, verbs and verb phrases bear a feature SUBCAT whose content is a (perhaps ordered) set of feature structures representing their unsatisfied subcategorization requirements. Thus the feature structures associated with transitive verbs include the information:

$$\left[\text{subcat: } \left\langle \begin{bmatrix} \text{NP} \\ \text{case: acc} \end{bmatrix}, \begin{bmatrix} \text{NP} \\ \text{case: nom} \end{bmatrix} \right\rangle \right]$$

(where NP abbreviates a substantial feature structure.) Applied to adjectival VP complementation, this treatment of subcategorization leads naturally to the postulation of adjectives which subcategorize for VP's, etc. (details below).

The significance of subcategorization information is that it represents a (perhaps ordered) set of grammatical categories with which a subcategorizer combines in forming larger phrases. When a subcategorizer combines with a subcategorized element, the resultant phrase no longer bears the subcategorization specification—it has been discharged. Cf. Pollard and Sag (1987, p.71) for a formulation of the HPSG SUBCATEGORIZATION PRINCIPLE.

We shall in general present subcategorization specifications in a slightly different way from that above, i.e., not as a single feature whose value is a list, but rather as a collection of COMPLEMENT features with category values. Cf. Borsley (1989) for a development of this approach, which we shall not attempt to justify here. We will therefore reorganize the information above in the following way:

$$\begin{bmatrix} \text{subject: } \begin{bmatrix} \text{NP} \\ \text{case: nom} \end{bmatrix} \\ \text{object: } \begin{bmatrix} \text{NP} \\ \text{case: acc} \end{bmatrix} \end{bmatrix}$$

We choose this representation here only because we find the keywording of grammatical functions, *subject*, etc., more perspicuous than an encoding in terms of list positions, but nothing in the analysis hinges on the one or the other representation.

We shall furthermore allow that subcategorized elements be either obligatorily subcategorized or optionally subcategorized. Optionally subcategorized elements need not be discharged from subcategorization specifications. (This necessitates an obvious change to the principle that subcategorization must be satisfied in independent utterances.) In case an element is not discharged, something must be said about its semantics. Here we borrow an idea from Situation Theory, and specify that unsaturated predicate-argument structures (or infons, see Devlin, 1991) may hold when there is some way of filling out the unfilled argument positions so that the result holds. This has the effect of existentially quantifying over unfilled argument positions. Linguistically, there are many other ways in which arguments may be omitted (cf. Fillmore 1985), but this seems to suffice for the adjectives under examination here.

CONTROL AND MODIFICATION, the latter being the relation between an adjunct and a head, are both lexically realized in the case of the *easy* adjectives. We regard there as being a control relation between *for Smith* and *to get* in complex adjectivals such as *easy for Smith to get* (cf. GKPS 1985: 83ff). Modification plays a role when complex adjectivals appear in construction with nominal heads, as in *easy job for Smith to get*. These are common assumptions in the analyses of control and modification.

LONG-DISTANCE DEPENDENCE is treated in HPSG in much the same way it was treated in GPSG (cf. GKPS, 1985), and we assume basic familiarity with this type of analysis. We recall that the site of a missing element in a "gappy" constituent bears a feature SLASH, whose value is a specification of the expected material. The SLASH specification is propagated by general principles (which we shall not elucidate) to the higher level constituents, until it is matched by a "filler" or a subcategorizing element. When the gappy constituent is adjoined to a filler or subcategorizing element, the result no longer bears the SLASH value.

Important for our purposes is the possibility of a lexical entry specifying that a dependent may contain a gap. (Cf. GKPS, 1985, pp.150-53 for the first mention of this suggestion.) We shall exploit this in the analysis of several word classes below, viz., the ones which subcategorize for a VP with an NP in SLASH. It is unusual to find a subcategorization specification for SLASH, but not unique: comparatives likewise subcategorize for gappy complements, as in seen in examples such as *taller than it is* Δ *wide*. We shall require lexical specifications that lead to feature structures of the following form:

$$\begin{bmatrix}
\text{stem: } \textit{easy} & \\
\text{sem: } \textbf{easy}(\boxed{1}, \hat{}\,\boxed{2}) & \\
\text{syn.loc.subcat:} & \begin{bmatrix}
\text{subj:} & \begin{bmatrix}\text{syn: NP-nom}\\ \text{sem: }\boxed{3}\end{bmatrix} \\
\text{pp-for:} & \begin{bmatrix}\text{sem: }\boxed{1}\end{bmatrix} \\
\text{xcomp:} & \begin{bmatrix}\text{syn: VP-inf}\\ \text{sem: }\boxed{2}\\ \text{slash: }\begin{bmatrix}\text{sem: }\boxed{3}\end{bmatrix}\end{bmatrix}
\end{bmatrix}
\end{bmatrix}$$

The tag $\boxed{3}$ in the diagram above shows that the semantics of the SLASH value and the adjectival subject semantics have been identified. Thus, once a VP/NP has combined with this adjective, the semantic contribution of the SLASH element is assumed by the subject. Figure 1 shows an analysis tree for an example containing a long-distance dependency.

The variety of linguistic phenomena exemplified in the *easy*-class of adjectives guarantees that it is a demanding testing ground for theories of lexical representation.[3]

3 Adjectival VP Complementation

We assume familiarity with the mechanisms of lexical inheritance and lexical rules in the analysis to follow, but we provide an overview of these mechanisms for lexical representation in Appendix A. The fundamental data we shall be concerned with are repeated in (2):

(2) a. Bill is easy to talk to.
b. It is easy to talk to Bill.
c. Bill is easy for Mary to talk to.
d. It is easy for Mary to talk to Bill.

Other adjectives that show this same distribution include the following:

[3] It is also worth mentioning that HPSG has also been the subject of intensive implementation activity during the past several years; we know of implementations at Hewlett-Packard Laboratories, The German AI Center (DFKI), Stanford University, Carnegie Mellon University, The Ohio State University, Simon Fraser University, University of Edinburgh, ICOT, University of Stuttgart, the IBM LILOG project in Stuttgart, and ATR. We may therefore safely refer the reader to documentations of those implementations, even if these are less generally available than the theoretical literature: Proudian and Pollard (1985), Nerbonne and Proudian (1987), Franz (1990), Emele and Zajac (1990), and Carpenter, Pollard and Franz (1991).

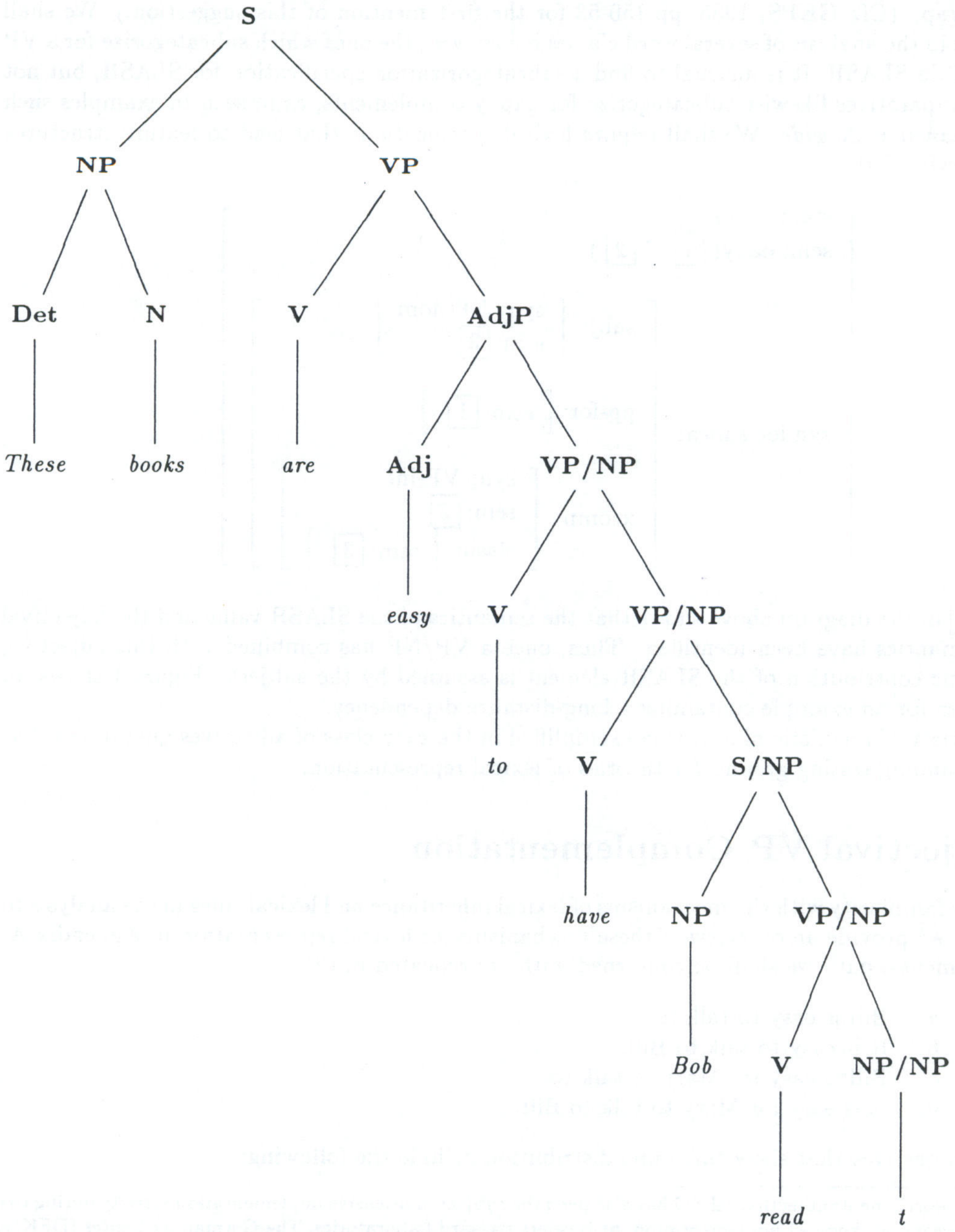

Figure 1: Complex adjectivals such as *easy* subcategorize for a complement VP containing a "slashed" NP, i.e., a VP missing an NP (whose expected position may be arbitrarily deep).

(3)
amusing
annoying
boring
comfortable
confusing
depressing
difficult
exhausting
fun
good
great
hard
important
impossible
impressive
nice
painful
tiresome
terrible
tough

Given pairs like (2a,b) and (2c,d), two clusters of properties begin to suggest themselves as part of the definitions of the relevant lexical entries. The first of these clusters we will associate with the class of words containing lexical entries for the *easy* of (2a,c) and its counterparts in (3), a class we term SLASH-EASY. The other cluster of properties we associate with a second class termed IT-EASY, containing the lexical entries for the variant of *easy* in (2b,d) and its counterparts in (3). We begin by simply identifying the relevant properties in each of these two classes, supported by examples as necessary; then we provide motivation for factoring these properties into several word classes linked by inheritance.

Adjectives in the IT-EASY class have two obligatory complements, an NP subject and a verbal complement; in addition they have one optional complement, a PP headed by the preposition *for*. As seen in (4), the verbal complement can be either infinitival or gerundive, and (5) shows that this complement can be a VP even with a PP-for present, or an infinitval S, again with or without the optional PP-for complement. The subject NP must be the expletive *it*.

(4) a. It was great working for Bill.
b. It was great to work for Bill.

(5) a. It's easiest for the dogs to feed them at noon.
b. For the dogs, it's easiest to feed them at noon.
c. It's easiest for the dogs to be chained up all day.
d. *For the dogs, it's easiest to be chained up all day.
e. It's easiest for me for the dogs to be chained up all day.
f. For me, it's easiest for the dogs to be chained up all day.

(5e,f) demonstrate that not only VP complementation, but also $\overline{S}$ complementation is involved in *easy* subcategorization. Note that $\overline{S}$ complementation never requires a controller, and that the PP phrase in such structures is mobile (5f). In addition to the conclusion that a variety of complementation schemes are used with *easy*, the data above also demonstrate that the exact specification of the controller (the understood subject of the infinitival VP) is nontrivial. (5a) demonstrates that the PP-FOR complement need not control the VP, and (5b) suggests that noncontrolling PP's are more mobile than controllers (5d).

We accommodate these facts semantically by allowing that *easy* and similar adjectives denote two-place relations between individuals and states of affairs. The relation holds between the pair, roughly, when it is easy (or convenient) for the individual when the state of affairs obtains. (5e,f) show that the individual involved in the easy relation need not be involved in the state of affairs, i.e. that there is no necessary semantic control involved in this relation.[4] The control facts are clear enough: when this *easy* is combined with an $\overline{S}$, there is no semantic control; and when it is combined with a VP, there is no grammatically specified controller of the VP—although there may be pragmatic inference about the understood subject.

Adjectives in the SLASH-EASY class also have two obligatory complements, an NP subject and a verbal complement, as well as an optional PP-for complement. In contrast to the first class, this class specifies that the subject is a normal (non-expletive) NP, and that the verbal complement

[4] There is an interesting pragmatic problem lurking in the control specifications involved here. If one specifies the control relationships exactly, then one needs to postulate systematic structural ambiguity in examples such as (5c), where the sequence of PP and VP may or may not be analyzed as an $\overline{S}$ constituent. This seems plausible, but then we would like to have a pragmatic account of why there is *normally* no distinction, i.e., why the control relationship is inferred, or, equivalently for all intents and purposes, why the $\overline{S}$ reading is so strongly preferred.

must contain an NP gap. Moreover, this verbal complement must be infinitival, not gerundive, as seen in (6), and must be a VP, not an S, as shown in (7).[5]

(6) a. Bill was great to work for.
b. *Bill was great working for.

(7) a. For me, Bill was easy to talk to.
b. *Bill was easy for me for Mary to talk to.

In the word class hierarchy we assume, sketched in Appendix A, there is a word class CONTROL which introduces a verbal complement, and which serves as the superclass from which both of the classes IT-EASY and SLASH-EASY inherit. However, neither of these classes is an immediate subclass of CONTROL; we draw on the data provided in (8) and (9) below to motivate two intermediate word classes that will stand between CONTROL and these two in the hierarchy.

The English lexicon contains two more groups of adjectives which have much in common with the two variants of *easy* introduced above, but which must be kept distinct. Lasnik and Fiengo (1974:535) identified a set of adjectives including *pretty* and *melodious*, illustrated in (8).

(8) a. Disneyland is pretty to look at.
b. Sonatas are melodious to listen to.

c. *It is pretty to look at Disneyland.
d. *It is melodious to listen to sonatas.

e. ?Disneyland is pretty for children to look at.
f. ?Sonatas are melodious for serious musicians to listen to.

Members of this class of adjectives share much in common with the SLASH-EASY adjectives, but have two significant differences: first, as shown by (8c,d), they do not have a corresponding entry with an expletive *it* subject, and second, they assign a real thematic role to their subjects. That is, (8a) entails that Disneyland is pretty, while (1a) does not entail that Bill is easy. The two-place relation suggested above for IT-EASY and SLASH-EASY adjectives could not account for the validity of this inference, since the subject of the adjective plays no direct role in the relation whatsoever. A distinct semantic relation is called for here, one in which the subject does play a role (which effectively makes this class a kind of EQUI adjective in contrast to the raising *easy*). It also appears that these adjectives do not permit the optional PP-for complement licensed by *easy* in (1c), though judgments are less clear. In order to express these differences, we introduce a class SLASH-COMP which will include the entries for *pretty* adjectives, and which will also serve as the class from which SLASH-EASY inherits.[6]

Similarly, English has a set of adjectives which have much in common with the IT-EASY adjectives of (1b,d), but with no counterparts of the SLASH-EASY type.

(9) a. It is possible to talk to Bill only at breakfast.
b. It is unnecessary to fire Bill.

c. *Bill is possible to talk to only at breakfast.
d. *Bill is unnecessary to fire.

The second principal difference between adjectives like *possible* and those of the IT-EASY class is that the former do not permit an optional PP-for phrase complement; they do allow the verbal complement to be either a VP or an S (containing a PP-for subject), but (10) shows that if a

[5] Hukari and Levine (1991) note in passing that there is a group of closely related adjectives like *worth* which do take a gerundive complement instead of the usual infinitival complement, as in *That article is not worth looking at.* The extension of our analysis to *worth* is straightforward, but not given here.

[6] Other adjectives of this SLASH-COMP class include *delicious, handsome, attractive,* and *lovely.*

PP-for is present, it must be contained within the S complement.

(10) a. It is unnecessary for Mary to fire Bill. (M firing B)
b. *For Mary, it is unnecessary to fire Bill. (M firing B)
c. *It is unnecessary for Mary for you to fire Bill.

Again, we express the distinction between the set of adjectives like *possible* and the IT-EASY adjectives by introducing a fourth class IT-SUBJ parallel to SLASH-COMP.[7]

These four class definitions, together with one supporting class, are given in (11-16), with the Superclasses attribute showing the relevant inheritance relations.

(11)

IT-SUBJ	
Superclasses	Control
Complements	
Subject-Features	(NForm *it*)
Subject-Role	none
XComp-features	(VForm Infinitival) (Complete + -)

The disjunctive specification (Complete + -) overrides the default (Complete -) specified in the CONTROL class, and means that the verbal complement may be either a VP (Complete -) or an S (Complete +).

(12)

SLASH-COMP	
Superclasses	Control
Complements	
XComp-Subj-Semantics	x
XComp-features	(SLASH (Category Noun) (NForm Normal) (Complete +) (Predicative -) (Case Accusative)) (Semantics Subject-Semantics))

The SLASH feature on the XComp specifies that the VP must contain a gap which is for a normal (non-expletive) noun phrase which is accusative case, and which is not predicative. This nonpredicative specification serves to exclude examples like **Bill is difficult to become.* assuming the complement of *become* is predicative, since the gap for that complement would fail to satisfy the restriction on SLASH given in (12). The SLASH specification furthermore notes that the SLASH semantic value is identical to that of Subject-Semantics. As was explained in Section 3 above, this is the form a lexical specification of semantic coindexing takes.

The controller of the controlled complement is specified through the attribute XComp-Subj-Semantics; for example, in CONTROL, this attribute has the value Subject-Semantics, since subjects are default controllers. But the complements of SLASH-COMP are not grammatically controlled (cf. (8e,f)), a fact which requires an overwriting specification. The semantic variable x is used here because it will not represent the semantics of ANY grammatical complement, which ensures that no grammatical control is effected (see examples (9a,b)). This is an example of a subregularity appearing within an exceptional specification.

The classes for the two variants of *easy* adjectives we have discussed have themselves one cluster of properties in common: they both license the optional PP-for phrase seen in preceding examples. To further reduce redundancy, we define in (13) the class FOR-EXPERIENCER, from which the

[7] Additional members of this IT-SUBJ class include *essential, necessary, sad, silly,* and *illegal.*

two classes in (14-15) also inherit.

(13)

FOR-EXPERIENCER	
Superclasses	
Complements	PP-for
PP-for-Features	(Category Preposition) (Lexical −) (PForm For)
PP-for-Oblig	No
PP-for-Role	For

(14)

IT-EASY	
Superclasses	It-Subj, For-Experiencer
Complements	
XComp-Features	(VForm Infinitival Gerund)

(15)

SLASH-EASY	
Superclasses	Slash-Comp, For-Experiencer
Complements	
Subject-Role	none
XComp-Subj-Semantics	PP-For-Semantics

As expected, the IT-EASY class eases one restriction on the verbal complement; note too that no controller is specified, in keeping with remarks on (5). On the other hand, the SLASH-EASY class blocks inheritance of the subject's thematic role assignment (the default value having been specified in the INCOMPLETE class from which CONTROL inherits), and alters the control relationship (inherited from SLASH-COMP and ultimately from CONTROL) so that the PP-For phrase rather than the subject of *easy* is interpreted as the subject of the VP complement. These are two further examples of the way in which default overwriting is employed; note that the latter represents a subregularity within a subregularity (cf. SLASH-COMP).

With reasonable assumptions about the definitions of other relevant classes in the hierarchy, along with an explicit definition of the class ADJECTIVE, provided here for clarity in (16-17), we can introduce the (sparse) lexical entries for the two variants of *easy* employed in (1a,b), as given in (17,18)

(16)

ADJECTIVE	
Superclasses	Major
Features	(Category Adjective) (Predicative + −)

(17)

easy-1a	
Superclasses	Adjective, Slash-Easy
Semantics	easy
Spelling	"easy"
Phonology	/izi/

(18)

easy-1b	
Superclasses	Adjective, It-Easy
Semantics	easy
Spelling	"easy"
Phonology	/izi/

Pairs of sparse lexical entries like those in (17,18) are related by a lexical rule which we label

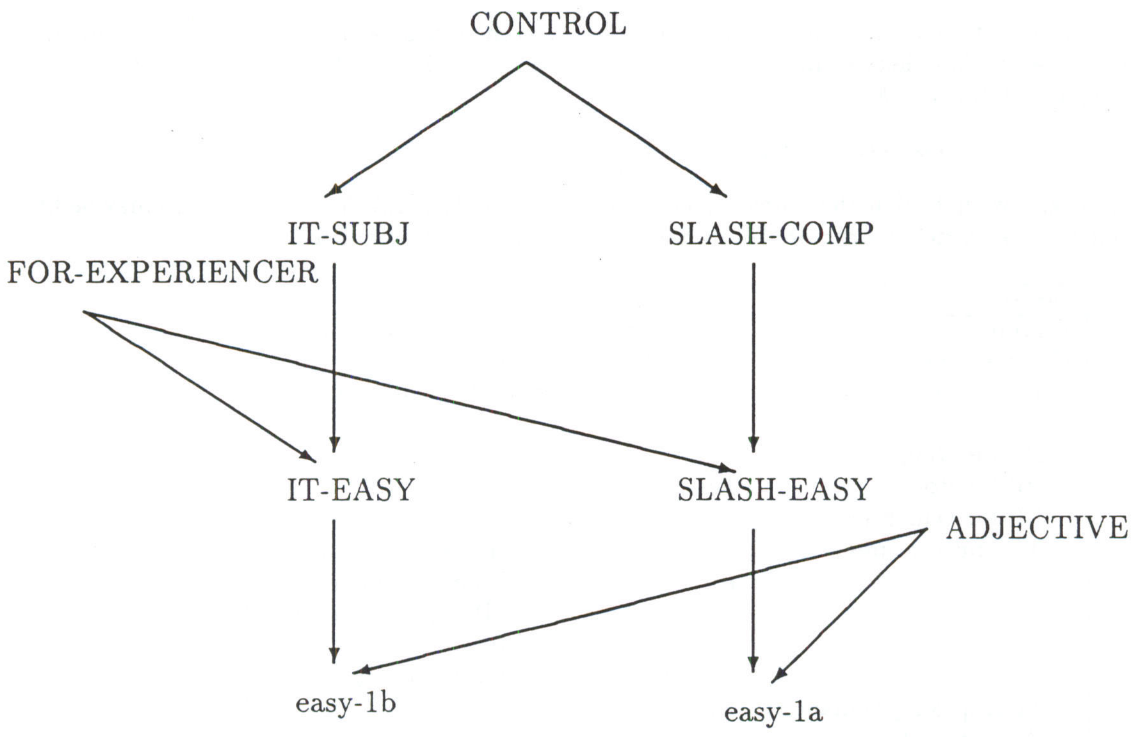

Figure 2: The structure of word classes directly involved in the definition of complex adjectival lexical entries.

LR-EASY, and which simply states that for each member of the class IT-EASY there exists a corresponding lexical entry belonging to the class SLASH-EASY, with everything but the Superclasses property identical in the two (sparse) entries.

LR-EASY lexical rule

LR-EASY		
LE2-Classes - IT-EASY	=	LE1-Classes - SLASH-EASY

Once each of (17) and (18) are fleshed out to include all of their inherited properties, they will of course be quite distinct, as needed to ensure the differences in distribution that we have described. Figure 2 summarizes the inheritance relationships thus far.

4 An Example Analysis

The purpose of this section is primarily illustrative—we would like to demonstrate the effect of the lexical specifications suggested on more familiar elements of grammatical analysis, *viz.* phrases, parse trees, and predicate logic representations.

The semantics of the *easy*-SLASH construction, which treats **easy** as a relation between an individual and a state of affairs is treated as a normal case of lexically inherited semantics, i.e. one in which the relation denoted has an argument place for the denotations of each of the role-playing complements, in this case the PP-FOR phrase and the XCOMP. This class of adjectives also has a SUBJECT among its complements, but it bears no role (as word class SLASH-EASY specifies), because this is a raising construction. For this reason, there is no argument place reserved in the semantics of *easy*-SLASH adjectives for the subject's denotation. To conserve space in the diagrams below, relations will be specified NOT using the keyword-coding shown in word class and lexical entry specifications (above), but rather in the more familiar order-coding.

In order to make not only the semantics but also the syntax somewhat clearer in its intended effect, we include here somewhat elaborate analytical sketches of the complex adjectival phrase *easy to get Mary to hire* in (19):

(19) Tom is easy to get Mary to hire.

To begin, we note that the sparse lexical entry for the SLASH-EASY version of *easy* may be filled out to a much richer structure if inherited properties are noted explicitly:

easy-1a	
Features	(Category Adjective) (Predicative + –)
Complements	PP-for,Subject,XComp
PP-for-Features	(Category Preposition) (Lexical –) (PForm For)
PP-for-Oblig	No
PP-for-Role	For
PP-for-Semantics	PP-For-Semantics
XComp-Features	(Category Verb) (Complete –)(Lexical –) (SLASH (Category Noun) (Complete +) (NForm Normal) (Predicative –) (Case Accusative)) (Semantics Subject-Semantics))
XComp-Subj-Semantics	PP-For-Semantics
XComp-Oblig	Yes
XComp-Semantics	XComp-Semantics
XComp-Role	State-of-Affairs
Subject-Role	none
Semantics	**easy**
Spelling	"easy"
Phonology	/izi/

The features noted above were specified by the lexical entry together with the classes ADJECTIVE, SLASH-EASY, SLASH-COMP, FOR-EXPERIENCER and CONTROL. Further subject properties would be inherited from INCOMPLETE, but for brevity these are not listed. (Of course many other properties, including e.g., gradation properties and the applicability of lexical rules have likewise been suppressed in the interest of clarity in presentation.) This lexical description translates fairly directly (with some further simplifications and abbreviations) into a feature structure of the sort used by HPSG grammars:

$$\left[\begin{array}{ll}
\text{stem: } \textit{easy} & \\
\text{syn.loc.head: Adj} & \\
\text{sem: } \textbf{easy}(\,\boxed{1}\,,{}^{\wedge}\boxed{2}\,) & \\
\text{syn.loc.subcat:} & \left[\begin{array}{ll}
\text{subj:} & \left[\begin{array}{l}\text{syn: NP-nom}\\ \text{sem: }\boxed{3}\end{array}\right]\\
\text{pp-for:} & \left[\begin{array}{l}\text{syn: PP-for}\\ \text{sem: }\boxed{1}\end{array}\right]\\
\text{xcomp:} & \left[\begin{array}{ll}
\text{syn: VP-inf} & \\
\text{sem: }\boxed{2} & \\
\text{subject.sem: }\boxed{1} & \\
\text{slash:} & \left[\begin{array}{l}\text{syn: NP-acc}\\ \text{sem: }\boxed{3}\end{array}\right]
\end{array}\right]
\end{array}\right]
\end{array}\right]$$

We would like to draw attention to two semantic coindexings in the structure, which are both lexically specified and which simplify subsequent (grammatical) processing. The coindexing of the xcomp's subject with the pp-for is effected in the SLASH-EASY word class, and the semantic coindexing seen above is just a consequence of that. The coindexing of the xcomp's slash's semantics value with the subject's semantics, on the other hand, derives ultimately from SLASH-COMP.

In Figure 3 we examine the combination of a token from this class of *easy* adjectives and a VP/NP. The very sparse specification of the mother phrase's features is, in fact, solely for purposes of legibility—all of the information specified on the mother node may be derived from general HPSG principles, so that nothing is specified, e.g., on the rule which licenses head-complement combinations. The fact that the semantics attribute is identified with the subcategorizer's semantics follows from the HPSG SEMANTICS PRINCIPLE, which states that the semantics of a phrasal node is always to be identified with the semantics of a head in a head complement combination. The fact that the slash value of the mother structure is empty follows from the BINDING INHERITANCE PRINCIPLE, which states that slash values are collected going up a tree—unless a head subcategorizes for an element containing a slash value, in which case the slash satisfies the subcategorization requirement. The identification of the feature structure labeled $\boxed{7}$, which is just the representation of the phrasal node dominating *to get Mary to hire*, with one of the adjective's subcategorization specifications, that labeled $\boxed{6}$, is just a condition for the applicability of the head-complement rule, not an additional specification. Of course, the phrasal node is massively underspecified here, but the suppressed information is predictable, not merely hidden.

This is an intriguing aspect of HPSG, but we dwell on it here for self-serving purposes. If the properties of the phrasal combination of this fairly intricate syntactic structure require no further comment, that is largely because the lexicon has provided a wealth of richly structured representation. This would hardly be feasible in the absence of efficient and sophisticated lexical representation mechanisms.

To complete this illustration, we spell out the effects of unification on the structure above in Figure 4. Note in particular that because the slash semantics on the VP phrase is identified with the slash semantics on the subcategorized-for VP, which in turn is identifed with the semantics of the subject for *easy*, the resultant phrase will bind its subject to the deeply embedded object argument position of the verb *hire*. This takes place even though the subject plays no role in the easy relation itself. This is exactly what is wanted semantically of a raising construction.

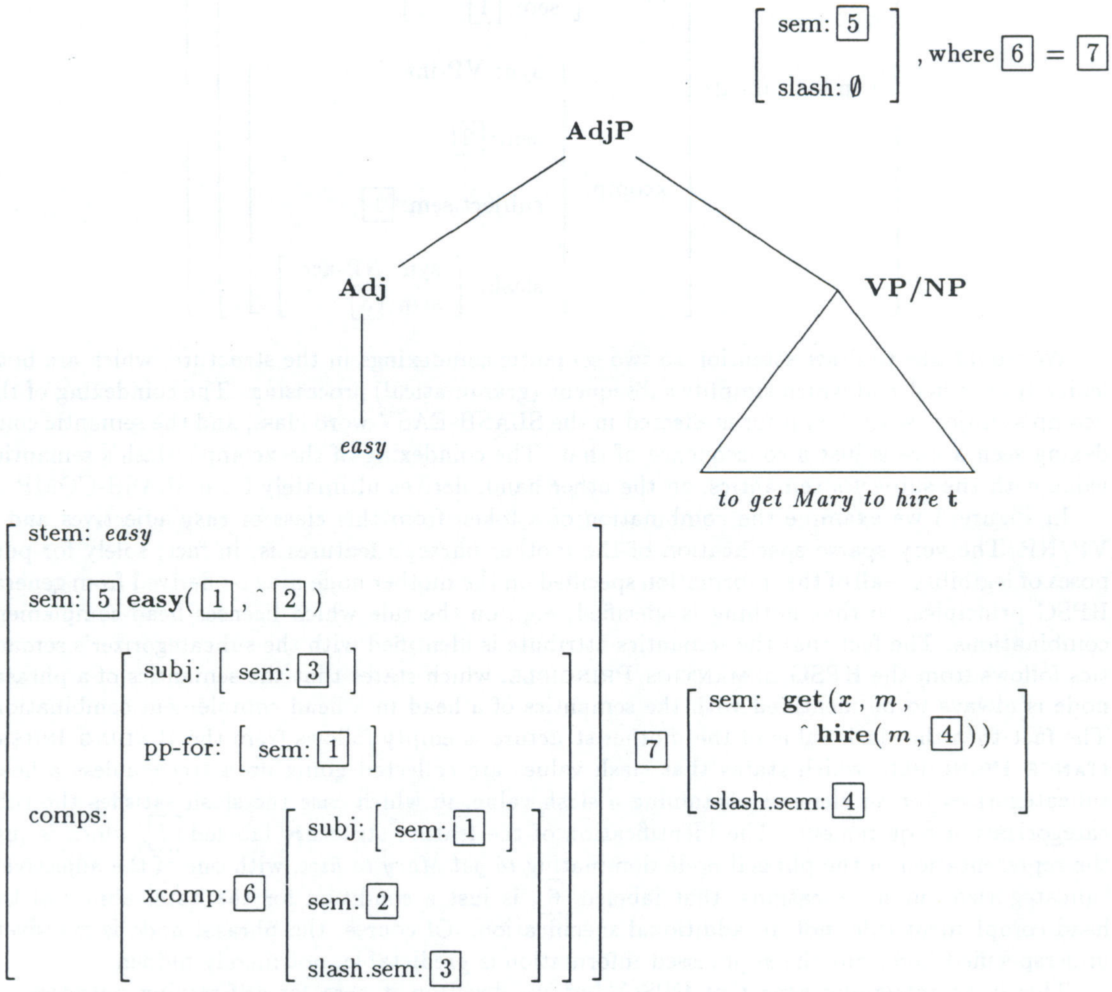

Figure 3: The combination of complex adjective and slashed VP complement.

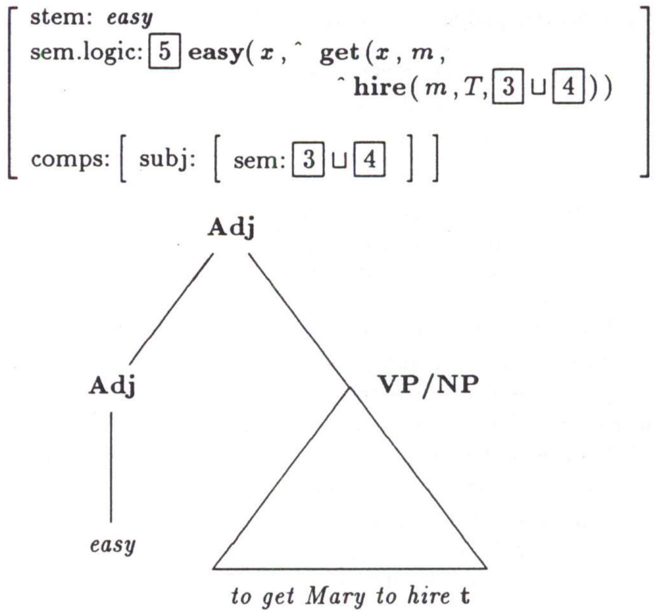

Figure 4: The result of combining complex adjective and slashed VP complement. Note that the subject of *easy* is still semantically coindexed with the missing VP object.

5 Extensions and Lexical Maintenance

The structured lexicon aims ideally at a redundancy-free specification of all lexical properties, and indeed, it achieves this largely through the use of inheritance. While we do see scientific parsimony as an end in itself, we see two further advantages in the employment of the structured lexicon, one scientific and one practical. The scientific advantage of the structured lexicon is that it identifies significant classes in the language. In a feature system with approximately 30 atomic features (including semantics), each of which ranges over approximately 10 values, it is certainly striking that we never see need to distinguish 10^{30} classes of items. In fact we distinguish approximately 300 lexical classes in HP-NL, a large system with very broad grammatical coverage (see Nerbonne and Proudian, 1987).

But the practical advantage of the structured lexicon may ultimately also be of scientific value, and that is because a structured lexicon is more easily maintained and extended than a nonstructured one. This advantage derives immediately from the characteristic that lexical properties are normally specified only once. Modifications tend then to be minimal, and extensions less frightening. The ultimate scientific benefit this may bring derives from the fact that it is then easier in systems with structured lexicons to experiment with grammatical description.

The following section is an attempt to buttress the claim that structured lexicons are easily extended. We examine therefore extensions to the analysis above of adjectives which govern VP complements—to nouns with similar subcategorizations, to the adjectival specifiers *too* and *enough*, and to adjectives which govern S complements rather than VP complements.

5.1 Pleasure nouns

Adjectives like *easy* have been the most widely studied group of lexical types that populate the classes introduced in the analysis above, but they do not have exclusive claim to those classes. Lasnik and Fiengo (1974:568) observed that the English lexicon also contains a group of nouns with similar properties, as illustrated in (20-21),

(20) a. Nureyev is a pleasure to watch.
b. This course is a breeze to pass.
c. Venice is a delight to visit.

(21) a. It is a pleasure to watch Nureyev.
b. It is a breeze to pass this course.
c. It is a delight to visit Venice.

Like the adjectives discussed above, nouns such as *pleasure* have two variants, one which appears with an ordinary NP subject and an infinitival complement containing an NP gap; and one which selects an expletive *it* subject and an infinitival complement with no gap. Given the word class definitions developed on the strength of the adjectival examples, an obvious analysis of the nominal examples suggests itself: *pleasure*, like *pleasant*, has one lexical entry belonging to the SLASH-EASY class, and a second entry that inherits from the IT-EASY class. The (sparse) descriptions of both entries are given in (22-23), parallel to those for *easy* given in (17-18) above, the salient difference being that the noun entries inherit from the class Common-Noun where the adjective entries inherited from the Adjective class.[8]

(22)

pleasure-1a	
Superclasses	Common-Noun, Slash-Easy
Spelling	"pleasure"
Semantics.Pred	**pleasure**
Phonology	/plEzhr/

(23)

pleasure-1b	
Superclasses	Common-Noun, It-Easy
Spelling	"pleasure"
Semantics.Pred	**pleasure**
Phonology	/plEzhr/

Having declared nouns like *pleasure* to have entries that are members of SLASH-EASY and IT-EASY, nothing more needs to be said in order to capture the syntactic relationship between these two forms of *pleasure*. The lexical rule we proposed earlier to link pairs of adjectives like the two variants of *easy* is defined as a regularity holding between the two classes SLASH-EASY and IT-EASY, making no mention of the class ADJECTIVE in its formulation. Hence it also serves to link the pair of noun entries in (22-23).

Some further explanation needs to be provided about the semantics of this class of nouns, since the nouns do seem semantically anomalous even if we shall maintain that all of the apparent anomaly ultimately stems from their having a subject—and thus being available for control (by *be* and other raising verbs). In general a common noun is interpreted as a relation between a theme argument and the denotation of its complements, if there are any. For example, *friend* is interpreted as a relation between a theme argument and the denotation of the complement PP-OF phrase. We refer to the theme argument of the relation denoted by the common noun as its DENOTATION. An apparent peculiarity of nouns such as *pleasure* is that there appears to be no denotation of the noun in the usual sense, e.g., in (20a). At issue is whether there is any theme argument position for the "pleasure" in the relation denoted by *pleasure*. I.e., does *pleasure* denote the same two-place relation between individuals and states of affairs that *pleasant* does, or is there a third argument position in *pleasure* which is occupied by an (abstract) "pleasure" individual?

The suspicion that no denotation is involved likely stems from our intuition that we do not

[8] Other nouns in this class include *disappointment, ordeal, challenge, joy, inspiration* and *privilege*.

seem to refer to an object which is a *pleasure* in uttering either (20a) or (21a), at least not any more than we would if we had used *pleasant* in the place of *a pleasure*. Now this suggests that the noun (phrase) is used predicatively, much as many noun phrases are after the verb *be*. Compare *Tom is a linguist.*

This does not help a great deal, however. Even though the analysis of predicative NP's is an old topic semantically (cf. the definition of *be* in Montague, 1973, p.261), there has been essentially no successful attempt to treat predicative nouns as if they had no denotation. Any attempt to do so seems to run afoul of the standard (if limited) determination and adjectival modification found in phrases such as *no great pleasure to watch*; at least such examples point out the inevitable duplication a semantic analysis would incur if predicative nominals had no denotation.

We therefore interpret *pleasure* as a three-place relation

$$\mathbf{pleasure}(\text{theme} : e, \text{for} : x, \text{soa} : s)$$

which obtains just in case e is the pleasure x has in case s. It should of course also turn out that this relation for some e holds iff **pleasant**(for : x, soa : y), but we will not be concerned with showing that here. e provides a denotation which is subject to determination (*no*) and (intersective) adjectival modification (*great*). Under this analysis, *a pleasure to watch* and *no pleasure to watch* denote quantifers, i.e., in each case a set of properties of pleasures (e's from above). Of course, a quantifier does not by itself represent a proposition, something which could be true or false—for that it must be paired with a property. In these cases, the relevant property is always the universal (existence) property; i.e. utterances of sentences such as (20a) are true just in case there is a pleasure of the relevant kind (and *mutatis mutandis* for the negative existentials). We therefore postulate that the predicate *be* in these sentences denotes the universal property.[9]

What is striking about this proposal is that it assigns the common noun *pleasure* exactly the semantics which the general scheme predicts—a relation between a theme and the denotations of other complements. For this reason, the word classes for *pleasure* nouns make no special stipulations about semantics.

We therefore derive feature structures such as the following, which are used in the syntax and semantics processing of the word *pleasure*. The first structure represents the member of the SLASH-EASY class, and the second the member of the IT-EASY class. (We have simplified the structures to highlight the semantically relevant parts.)

$$\begin{bmatrix} \text{stem: ``pleasure''} & \\ \text{sem.logic:}\ \mathbf{pleasure}(e, \boxed{1}, {}^{\wedge}\boxed{2}) & \\ \text{subcat:} & \begin{bmatrix} \text{subj:} & \begin{bmatrix} \text{sem:} \boxed{3} \end{bmatrix} \\ \text{pp-for:} & \begin{bmatrix} \text{sem:} \boxed{1} \end{bmatrix} \\ \text{xcomp:} & \begin{bmatrix} \text{subj:} & \begin{bmatrix} \text{sem:} \boxed{1} \end{bmatrix} \\ \text{sem:} & \begin{bmatrix} \text{logic:} \boxed{2} \\ \text{slash:} \boxed{3} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

[9] In fact, we do not stipulate a peculiar semantics for the raising verbs (such as *be*) which are involved here. Instead, we allow *be* to denote the identity relation, which holds of a single argument just in case there is some way of filling in the missing argument—i.e. in case the first exists. This follows from the general treatment of unsaturated relations in Situation Theory (cf. Section 2 under subcategorization). Note, however, the one exceptional aspect, i.e., that the subject of the verb *be* is not linked to any argument position in the relation denoted by the controlled complement (in this case, *pleasure*).

$$\begin{bmatrix} \text{stem: "pleasure"} \\ \text{sem.logic: } \textbf{pleasure}(e, \boxed{1}, \hat{}\boxed{2}) \\ \text{comps: } \begin{bmatrix} \text{subj: } \textbf{NP-}\textit{it} \\ \text{pp-for: } \begin{bmatrix} \text{sem: } \boxed{1} \end{bmatrix} \\ \text{xcomp: } \begin{bmatrix} \text{subject: } \begin{bmatrix} \text{sem: } \boxed{1} \end{bmatrix} \\ \text{sem: } \boxed{2} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

On the other hand, the noun classes ARE exceptional in that the nouns involved have subjects—a property they inherent finally from INCOMPLETE, in the one case through CONTROL, IT-SUBJ and IT-EASY; and in the other from CONTROL, SLASH-COMP and SLASH-EASY. It is this property, shared by the NP's they give rise to, which explains (i) their ability to be controlled e.g., by the verb *be*—only unsaturated phrases are subject to control; (ii) their inability to function in normal NP's, e.g., in the subject position of any intransitive verb; and finally (iii) the fact that they can stand in construction with the main verb *be* without being asserted to be identical to its subject.

We turn now to further points on the syntax of the *pleasure* nouns. The two definitions of entries for "pleasure" also predict the grammaticality judgments seen in (24), analogous to the examples given above for adjectives, and based on the definitions given for the IT-EASY and SLASH-EASY word classes.[10]

(24) a. Nureyev is a pleasure for us to watch.
b. It is a pleasure for us to watch Nureyev.

c. For us, Nureyev is a real pleasure to watch.
d. *For us, Nureyev is a real pleasure for our parents to watch.
e. For us, it is a real pleasure for our parents to watch Nureyev.
f. It is a real pleasure for us for our parents to watch Nureyev.

g. *Nureyev is a pleasure watching.
h. It is a pleasure watching Nureyev.

Recalling further that the adjectives we looked at above fell into not two but four distinct classes, we might expect to find nouns as well that belong to the other two classes, IT-SUBJ and SLASH-COMP. Such instances are found in English, as illustrated for IT-SUBJ nouns by the

[10] Nothing we have said so far captures the fact that some pairs of members of these two classes, like "pleasant" and "pleasure", are morphologically related. We do not offer here a proposal for capturing nonproductive regularities of this kind, though some extension of the lexical rule mechanism might serve, an extension that would depend heavily on the ability to specify negative exceptions to lexical rules, given examples like the following.

(i) It is difficult to hire Bill.
(ii) *It is a difficulty to hire Bill.
(iii) *Bill is a difficulty to hire.

(iv) It is impossible to work with Bill.
(v) *It is an impossibility to work with Bill.
(vi) *Bill is an impossibility to work with.

examples in (25), and for SLASH-COMP nouns by those in (26), drawn from Lasnik and Fiengo.[11]

(25) a. It would be a mistake to fire Bill.
b. It was a shock to find Bill here.

c. *Bill would be a mistake to fire.
d. *Bill was a shock to find here.

(26) a. This room is a pigsty to behold.
b. Nureyev is a marvel to watch.

c. *It is a pigsty to behold this room.
d. *It is a marvel to watch Nureyev.

The noun *mistake* and the adjective *possible* have in common just those properties specified by the IT-SUBJ class (together with its superclasses); and like the differences between *pleasure* and *easy*, their differences result from *mistake* being a member of the COMMON-NOUN class while *possible* inherits from the ADJECTIVE class. Since the lexical rule relating the two variants of *pleasure* (and the two variants of *easy*) is defined to link members of the two classes SLASH-EASY and IT-EASY, the rule correctly does not predict the existence of similar alternate entries for nouns like *mistake* and *pigsty*.

Interaction with Lexical Rules

Given that the domain of lexical rules is always one or more word classes, and that the LR-Intraposition rule is defined on the IT-SUBJ class, we predict the grammaticality of the following examples with *pleasure* nouns, since they also have entries belonging to the IT-SUBJ class, and should be expected to conform to the LR-Intraposition rule. Here again, the combined devices of inheritance and lexical rule produce the desired results for nouns without requiring that anything be added to the analysis motivated from data on adjectives and verbs.

(27) a. (For me) to stay another day would be a real pleasure.
b. It would be a real pleasure (for me) to stay another day.

c. To visit Venice now might be a disappointment for you.
d. It might be a disappointment for you to visit Venice now.

5.2 *Too* and *enough*

To drive home our central point about the expressive and predictive power of inheritance in lexical representation, we turn to a third, small class of lexical entries that show complementation properties like those we have already seen. Jackendoff (1972,p.227) noticed that the two words *too* and *enough* also appear in constructions with an infinitival complement that contains an NP

[11] Additional IT-SUBJ nouns include *battle, disgrace, error, honor, relief, shock* and *surprise*. Other SLASH-COMP nouns include *beauty* and *terror*.

gap, as illustrated in (28) with examples drawn from Lasnik and Fiengo (1974:536).[12]

(28) a. The mattress is thin.
b. *The mattress is thin to sleep on.
c. The mattress is too thin to sleep on.

d. The football is soft.
f. *The football is soft to kick.
g. The football is soft enough to kick.

In particular, the examples in (29) suggest that these adverbs select for complements that are the same as adjectives like *pretty*, entries which are not related via lexical rule to variants that license an expletive *it* subject.

(29) a. *It is too thin to sleep on this mattress.
b. *It is soft enough to kick this football.

Informally, it seems that when *too* or *enough* combines with an ordinary adjective, the resulting phrase (*too thin* and *soft enough*) exhibit complementation properties very much like those of *pretty* adjectives. By defining the lexical entries for these two adverbial specifiers as members of the SLASH-COMP class, we begin to provide an account for examples (28c,g) as well as those in (29). The entry for *too* is given in (30), inheriting both from the ADVERB class and from the SLASH-COMP class; the entry for *enough* is similar, leaving out of the present discussion an account of the linear order difference between the two adverbs with respect to the adjective they modify.

(30)

"too"	
Superclasses	Adverb, Slash-Comp
Spelling	"too"
Phonology	/tu/

With the inclusion of this class of adverbs, our lexical subhierarchy involving complementation of slashed VPs has grown to a point where it surely demonstrates the virtues of the structured lexicon approach. Figure 5 illustrates the more complete structure. It is a curious fact that the number of lexical classes does not grow enormously even while fairly detailed analyses involving very different grammatical areas are undertaken. In several years of development at Hewlett-Packard Laboratories, involving detailed analyses of dozens of constructions, the number of word classes never exceeded 400. This must be due finally, not to the lexical analysis tool, but rather to the tendency of language to reuse significant classes.

This analysis of these two unusual adverbs has left begging an important issue about how the complementation specifications provided by *too* are propagated up to the phrase *too thin.*[13] We have said little here about how lexically supplied subcategorization information is employed in parsing, referring the reader to full accounts given in Pollard and Sag (1987) and related references. Yet it is clear that something more must be said about this construction, given that in HPSG it is the syntactic head of a phrase that imposes constraints on its complements; and we assume that *thin*, not *too*, is the head of the phrase *too thin to sleep on.* To motivate the necessary elaboration of our analysis for these two adverbs, we turn to one more set of data involving gappy infinitival complements, one that has received little study to date.

[12] Baltin (1987) presents a more recent analysis of these "degree complements."

[13] One might be tempted to try a lexical rule approach which would treat *too thin* as a derived lexical item which selects for a VP complement. But slightly more complicated examples quickly render this approach untenable. Cf. *This country is too thinly populated to worry about* (where we take the scope of the specifier *too* to be *thinly populated*). Here, the lexicalized form that selects for a VP complement would have to be *too thinly populated*, a result we regard as unacceptable.

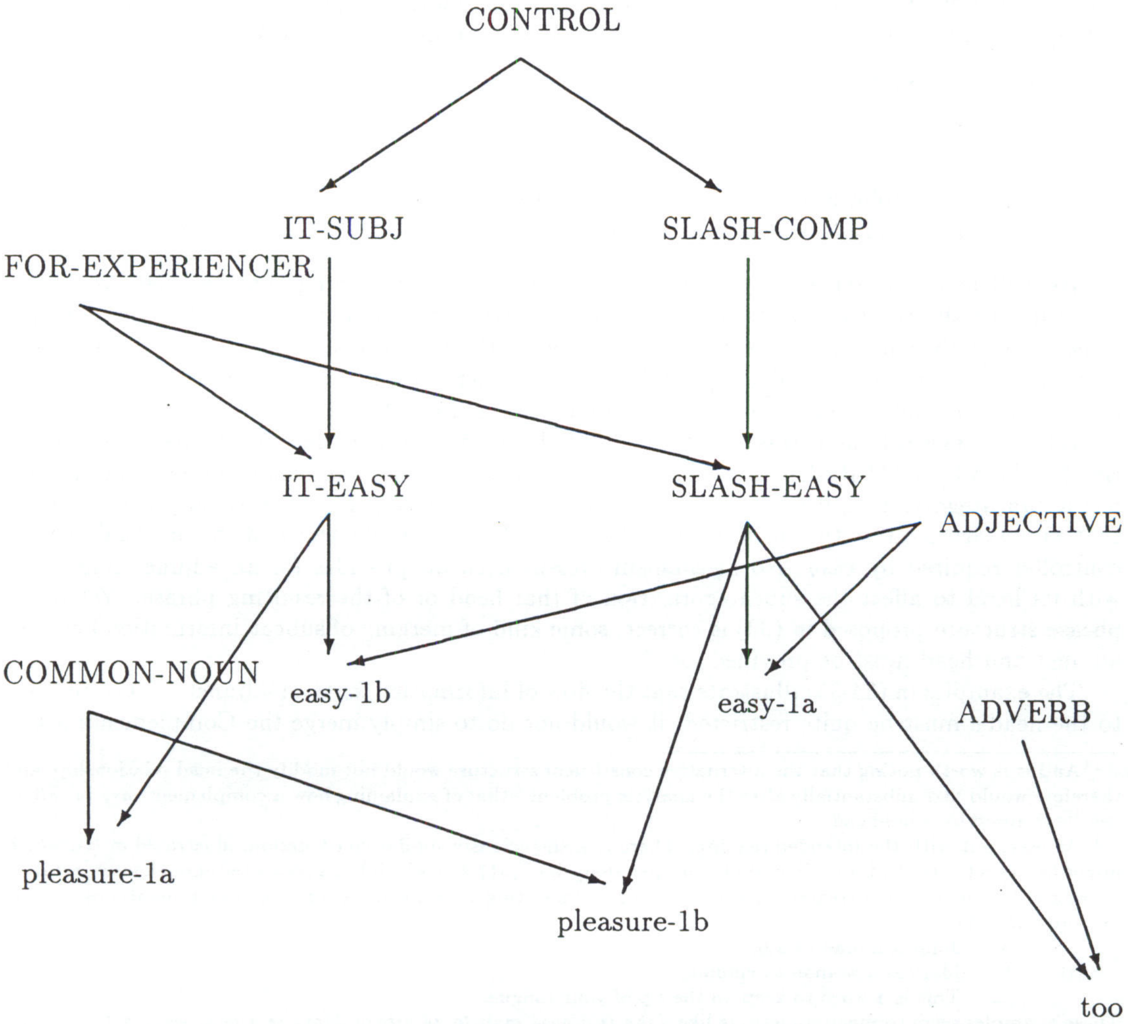

Figure 5: The lexical subhierarchy involving elements which govern "slashed" verb phrases. Note that the original hierarchy needed very little modification, merely addition. We speculate that this is due to the fact that significant classes are being identified in detailed grammatical description. There is also a version of *too* which inherits from ADVERB and IT-EASY which is not shown (since it was not discussed). The asymmetry is only apparent.

Excursus on Subcategorization Transfer

As the examples in (31) show, adjectives like *easy* appear not only in predicative constructions like those illustrated above, but also as nominal modifiers.

(31) John is an easy man to talk to.

While the example in (31) is good, employing the *easy* which belongs to the EASY-SLASH class, the examples in (32-33) are ungrammatical. The analysis we have provided thus far does not yet explain the grammaticality of (31) and the ungrammaticality of (32,33).

(32) a. *John is an easy to talk to man.
b. *John is an easy man.

(33) a. *John is an easy man to talk to Bill.
b. *John is a man easy to talk to Bill.

We will focus on explaining the grammaticality of (31), assuming that the right syntactic structure for the sentence is the binary-branching structure given in (34), where *easy* forms a constituent with *man*, and where *to talk to* is sister to the phrase *easy man*. We adopt the binary structure largely because it will simplify the exposition here; it might be equally defensible to hold that *easy*, *man*, and *to talk to* are all sisters of a single phrase.[14]

What is awkward about this structure is that the head noun *man* does not by itself subcategorize for the VP/NP.[15] Rather, it seems that when *easy* combines with *man*, the resulting phrase has a subcategorization list which contains not only the optional and obligatory complements that *man* started out with, but also the obligatory VP/NP complement and the optional For-PP controller required by *easy*. No mechanism presented so far provides for an adjunct combining with its head to affect the subcategorization of that head or of the resulting phrase. Yet if the phrase structure proposed in (33) is correct, some kind of merging of subcat information between adjunct and head must be provided for.[16]

The examples in (34-35) illustrate that the flow of information from an adjunct's list of subcats to the head's must be quite restricted; it would not do to simply merge the Complements list of

[14] And it is worth noting that the alternative constituent structure would not modify the head relationship, and therefore would NOT substantially alter the analytic problem—that of explaining how a complement *easy to talk to* can be licensed by a nonhead.

[15] At least not with the intended reading. There is a suspiciously similar construction, illustrated in (i), which might be expected to shed some light on the proper analysis of (31), but which has a restricted enough interpretation to suggest that it should be treated separately, probably inheriting a specification from the more general construction exhibited in (31).

(i) a. John is a man to admire.
b. Mary is a woman to emulate.
c. This is a word to keep on the tip of your tongue.

These examples seem to mean something like *John is a good man to admire* or *Mary is a good woman to emulate*, where the semantic contribution of *good* has been incorporated into the N – VP/NP construction in (i). To test this, consider the examples in (ii), where the *good* reading should lead to an anomalous interpretation, and does (cf. the corresponding examples in (iii).

(ii) a. ?Mary is a person to underestimate.
b. ?Sharks are animals to tame.

(iii) a. Mary is an easy person to underestimate.
b. Sharks are difficult animals to tame.

Given the constrained interpretations of examples like those in (i-ii), it does not seem defensible to treat *easy man to talk to* as simply the modifier *easy* combining with *man to talk to*. Any such attempt would be strained in accounting for (ii); in addition, such an analysis would leave unexplained the ungrammaticality of **John is an easy man*.

[16] It is probably worth noting that extraposition seems unlikely to be generalizable to all cases involving transferable subcats, at least if extraposition is to be bounded uniformly:

(i). An easy man to talk to arrived yesterday
(ii). *An easy man arrived yesterday to talk to.

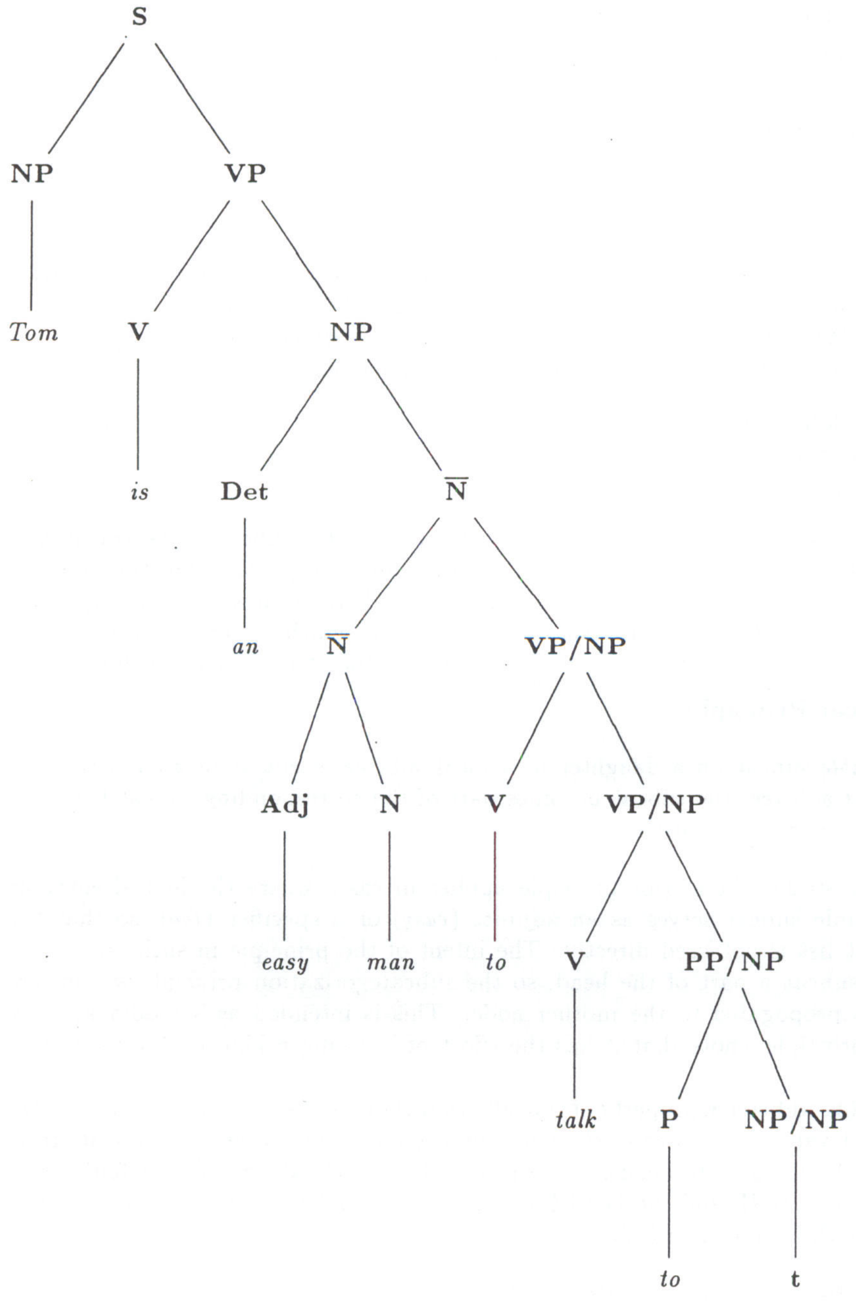

Figure 6: Complex adjectivals "wrapped" around a modified noun. Note that the N head of the $\bar{\mathrm{N}}$ constituent in construction with the complex adjectival has not licensed it. Subcategorization TRANSFER has taken place.

any adjunct with that of the head in every case.

(34) a. *an eager man to please
b. *a fearful man of snakes
c. *a frightened man by snakes
d. *an angry man at John

(35) a. a man eager to please
b. a man fearful of snakes
c. a man frightened by snakes
d. a man angry at John

The above examples might suggest that what distinguishes *easy* from these other adjectives is that the VP/NP complement of *easy* is obligatory, while the PP complements of the above adjectives are optional. While there are not many adjectives against which to test this hypothesis, the one clear case of an adjective that takes an obligatory complement counts against the idea:

(36) a. a man fond of snakes
b. *a fond man
c. *a fond man of snakes

The analysis we propose localizes in lexical entries the ability of a subcat to be transferred from adjunct to head. Just as subcats can be marked for the obligatory/optional distinction in a class definition or in a lexical entry, so can they be marked for a distinction we term *transferable*. While as a default subcats will be Non-Transferable, those subcats which are identified by a class or lexical entry as Transferable will be subject to the following informally stated principle:

Transferable Subcat Principle

> When a *transferable* subcat on a daughter in a local subtree is not associated with some sister in that subtree, the subcat becomes part of the corresponding subcat list of the head daughter in that subtree.

In the constructions studied here, this principle applies in cases where the lexical entry or phrase with a transferable subcat serves as an adjunct (*easy*) or a specifier (*too*), so that the word or phrase's subcat list is not used directly. The intent of the principle in such cases is to make the transferable subcat a part of the head, so the subcategorization principle will ensure that the information is propagated to the mother node. This is intended as a modification of the subcategorization principle—note that it has the effect of licensing a kind of "discontinuous constituent".[17]

Having introduced this additional property of subcats, that they can be specified as *transferable*, we note that the default value for this property must be negative, since in general subcats from adjuncts and specifiers do not pass to heads, as seen in (33) and (34) above. This default value will be overridden for the VP/NP and the For-PP subcats in the SLASH-COMP class, to reflect the grammaticality of both examples in (37).

(37) a. That was a melodious sonata to listen to.
b. John is an easy man to please.

Members of the SLASH-COMP class, including the relevant lexical entry for *easy*, will inherit this non-default *transferable* property for both the XComp and the For-PP, so when *easy* combines as an adjunct with the head noun *man*, these two subcats will become part of the subcategorization

[17] The ability to transfer a subcategorization requirement from a modifier to a mother (or to a head) is perhaps a bit similar in effect to FUNCTION COMPOSITION in categorial grammar (Bach, 1988). But in HPSG the possibility is lexically constrained.

of *man*, by the principle above, and will then become part of the subcategorization of the node *easy man*, accounting for the grammaticality of (31) above.

(32a) will be ruled out because of an independent constraint that restricts pre-head adjuncts to those which are (Head-Final +). (32b) is excluded because *easy* has an obligatory VP/NP complement, which must be included as an obligatory complement of the phrase *easy man*, due to the convention adopted above about merging of subcat information between a head and its sister. Finally, (33a) is excluded because the *easy* which requires an unslashed VP complement will not pass on its XComp subcat to the noun it modifies, since that XComp is, like most subcats, nontransferrable. license the incorporation. So *easy man to please Bill* will be excluded for the same reason that *eager man to please* is excluded: nothing licenses the post-nominal infinitival VP.

(33b) is probably best excluded on semantic grounds, since the subject of *easy to please Bill* is an expletive pronoun, the wrong sort to unify with the head noun being modified. On the assumption that a noun must serve semantically as the subject of adjectival adjuncts, those adjuncts must specify some thematic role for the noun to play. Thus any adjective which requires an expletive subject should give rise to a semantically ill-formed expression when it appears as an adjunct to a noun. What prevents the IT-EASY *easy* from serving as an adjunct to *problem* is the fact that this *easy* requires an expletive *it* subject.

Given this transferable complement mechanism, we may straightforwardly complete the analysis of the earlier *too/enough* examples: the lexical entries for these two adverbs simply specify that their gappy infinitival complement is transferable.[18]

6 Conclusions and Future Directions

The study of inheritance and, more generally, the study of structured lexical representations is an exciting and promising field. We would like to use this section to summarize how we view this work, and to suggest directions in which we feel it should move.

6.1 Conclusions

We have presented a treatment of complementation which uses nonmontonic lexical inheritance. The lexical specifications are quite compact and therefore both readily extendible and easily modified. We pointed out cases where nonmonotonic, default specification seems most natural, and the entire treatment turns on the possibility of there being genuine multiple inheritance of a "complements" attribute.

We adopt a skeptical approach to inheritance conflict. If there are inheritance conflicts in the system presented here, nothing is inherited. Mechanisms which warn users about such conflicts are useful, but we are wary of attempts to decide conflicts "intelligently." They seem likely to us to lead to cases where minor changes may have remote consequences, which would detract from maintainability.

We do not feel that we have overstated the case for structured lexicons by choosing a particularly messy or poorly understood area. To insist on this point somewhat, let us note that we omitted significant aspects of the grammar of the "raising nouns," e.g., their complements, specifiers, and adjuncts.[19] Grammar abounds in poorly understood areas, including comparatives, superlatives, adverbials, internal NP syntax and the "specialized grammars" found in dates, places, and technical vocabulary. All of these areas can benefit from the application of a tool for complex lexical description.

[18] This leaves much to be said about the lexical properties of *too* and *enough*, but more detailed analysis at this point would take us too far afield; it is clear enough that, whatever their other properties, these two adverbs share complementation properties with the adjectives and nouns studied here.

[19] For example, for "pleasure" nouns, some adjectives are okay, but not all (*a real/*competent pleasure to work with*; relative clauses are impossible *Sally is a pleasure *[that is real] to work with*; and some nominal complements are fine. *Sally was a pleasure of the rarest kind to work with*

6.2 Hypotheses or Tools?

The conclusions above may be read as a plea for the employment of an important tool in computational linguistics, and, indeed, we see the primary significance of the use of structured lexicons not in new expressive power which they bring to natural language processing or description (there is perhaps none), but rather in the increased ease and reliability with which they allow old hypotheses to be formulated and put to use.

Brachman (1983, p.35) summarizes the dominant view of inheritance in knowledge representation:

> Even though much has been made in the past of the significance of inheritance in semantic nets, no one has been able to show that it makes any difference in the expressive power of the system that advertises it...It is strictly implementational.

Given this authority on the technical side, it may be surprising to hear more application-oriented users of inheritance mechanisms hedging at all on whether there is any scientific significance to the proposal here. But there is at least a potential candidate: lexical rules may distinguish inherited from specified information.

In expressing the relationships between members of two sets of lexical entries, we make crucial use of the distinction between idiosyncratically specified information (which appears in a sparse (non-redundant) lexical entry) and inherited information. We have adopted here the restrictive hypothesis, proposed in Flickinger (1987:108ff), that lexical rules hold for minimally specified lexical entries, without having access to inherited, predictable information. Adopting this hypothesis imposes a constraint on the form and function of lexical rules which is strong, perhaps too strong, but one which allows a simpler formulation of rules by keeping to a minimum the amount of information to be managed. Only two kinds of information are relevant for a lexical rule: the word classes that each of the two related entries belong to, and any idiosyncratic properties specified by either lexical entry. We note that if lexical rules were insensitive to the distinction between idiosyncratic and predictable properties of lexical entries, the statement of even a simple rule like LR-PAST, given earlier, would be much more difficult. If the lexical rule for past tense verbs had to cope with fully-specified entries that blurred this distinction, it would be difficult to express in the rule just which properties of the one entry had to match in the other, related entry.

For example, the verb *like* idiosyncratically requires a verbal complement which is either infinitival or gerundive, while the verb *enjoy* does not allow the infinitival form, allowing only the gerundive form for its complement. Since all of the inflected forms of *like* allow the same choice of two permissible forms for the verbal complement, while all of the inflected forms of *enjoy* insist on the gerundive complement, the lexical rules like LR-PAST or the similar one for present-third-singular forms must preserve these idiosyncracies. Yet a fully-specified entry for the base form *enjoy* stipulates not just the form of the complement, which would have to be identical in the present third-singular entry *enjoys*; the fully-specified base entry for *enjoy* also specifies that its subject be unmarked for number, an indifference which crucially must not be shared by the entry for *enjoys*. Short of tagging each attribute value in a fully-specified entry as local or inherited, it is not clear how the lexical rule for present-third-singular forms could be constrained to ensure identity of the verbal complement's VFORM value while ignoring differences in the subject's AGREEMENT value for these two entries for *enjoy*. In sharp contrast, this difference in idiosyncratic vs. inherited information can be exploited by lexical rules without stipulation when they are constrained to apply only to minimally-specified entries.

It may not be superfluous to add that, even if the argument above about distinguishing inherited and specified information is ultimately fallacious, so that the use of inheritance were seen purely as a tool, and not at all as a scientific hypothesis, it may nonetheless prove to be of great significance, just as many tools have advanced areas of science that nothing to do with their development. The development of lenses revolutionized astromony, even though glass grinding embodied no astronomical hypotheses.

6.3 Emergent issues in structured lexicons

Perhaps more interesting are the many directions in which this research may be developed. We suggest some of these in the questions below.

What are lexical classes and lexical entries? The careful reader noted in Section 5 above that our lexical specifications are TRANSLATED into feature structures. Theoretically, we could dispense with the translation for nearly all of the information involved, and have the lexicon describe feature structures directly. But this does not correspond to our implementation, nor are we clear on how e.g., information on lexical rules and their application ought to be rendered in features. Perhaps lexical entries must be structured so that one component of a lexical entries is a feature structure, while others are not.[20]

Can inheritance be exploited in the specification of inflectional variation? This appears to be a promising area of application, since in general, one can view inflected elements as further specifications of abstract lexemes (cf. Evans and Gazdar, 1989 for an intriguing proposal).

Can derivational lexical rules be treated more satisfactorily? For example, it is clear that at least some lexical rules relate not merely a pair of word classes, but rather entire lexical substructures (involving several classes) to one another. Can the techniques of inheritance be applied here, so that exceptional elements may be easily accommodated?

References

[1] Bach, Emmon. 1988. Categorial grammars as theories of language. In Richard T. Oehrle, Emmon Bach, and Deirdre Wheeler (eds.) *Categorial Grammars and Natural Language Structures.* Dordrecht: Reidel, 17-34.

[2] Baltin, Mark. 1987. Degree Complements. In Geoffrey J. Huck and Almerindo E. Ojeda (eds.) *Syntax and Semantics 20: Discontinuous Constituency.* New York: Academic Press, 11-26.

[3] Bayer, Samuel 1990. Tough Movement as Function Composition. In Aaron Halpern (ed.) *Proceedings of the 9th West Coast Conference on Formal Linguistics.* Stanford: CSLI, 29-42.

[4] Borsley, Robert. 1983. A Welsh agreement process and the status of VP and S. In Gerald Gazdar, Ewan Klein, and Geoffrey Pullum (eds.) *Order, Concord, and Constituency.* Dordrecht: Foris Publications, 57-74.

[5] Borsley, Robert. 1989. Subjects, complements and specifiers in HPSG. to appear.

[6] Brachman, Ronald J. and James G. Schmolze. 1985. An overview of the KL-ONE knowledge representation system. *Cognitive Science*, 9:171-216.

[7] Brachman, Ronald J. 1983. What IS-A is and isn't: An analysis of taxonomic links in semantic networks. *IEEE Computer*, 30-36.

[8] Brame, Michael. 1979. *Essays Toward Realistic Syntax.* Seattle: Noit Amrofer.

[9] Bresnan, Joan. 1971. Sentence stress and syntactic transformations. *Language*, 47:257-281.

[10] Bresnan, Joan (ed.) 1982. *The Mental Representation of Grammatical Relations.* Cambridge: MIT Press.

[11] Carpenter, Bob, Carl J. Pollard, and Alex Franz. 1991. The specification and implementation of constraint-based unification grammar. *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics.*

[20] Krieger and Nerbonne (1991) attempt to characterize ALL lexical information (including inflectional and derivational relationships) in typed feature structures—with no further information.

[12] Chomsky, Noam. 1965. *Aspects of the Theory of Syntax.* Cambridge: MIT Press.

[13] Chomsky, Noam. 1973. Conditions on Transformations. In Stephen Anderson and Paul Kiparsky (eds.) *A Festschrift for Morris Halle.* New York: Holt, Rhinehart and Winston, 232-286.

[14] Chomsky, Noam. 1977. On *wh* Movement. In Peter Culicover, Tom Wasow, and Adrian Akmajian (ed.) *Formal Syntax.* New York: Academic Press, 71-132.

[15] Culicover, Peter and Wendy Wilkens. 1984. *Locality in Linguistic Theory.* New York: Academic Press.

[16] Devlin, Keith. 1991. *Logic and Information.* Oxford: Oxford University Press.

[17] Emele, Martin and Remi Zajac. 1990. Typed unification grammars. In *Proceedings of the 13th International Conference on Computational Linguistics,* Helsinki.

[18] Evans, Roger and Gerald Gazdar. 1989. Inference in **DATR**. In *Proceedings of the 4th Conference of the European Chapter of the Association for Computational Linguistics,* 66-71.

[19] Evans, Roger and Gerald Gazdar. 1990. The Semantics of **DATR**. In *Proceedings of 7th Conference of the Society for the Study of Artificial Intelligence and Simulation of Behaviour.* London: Pittman/Morgan Kaufmann, 79-87.

[20] Fillmore, Charles. 1985 . Pragmatically controlled zero anaphora. In *Proceedings of the 7th Meeting of the Berkeley Linguistic Society.*

[21] Flickinger, Daniel. 1987. *Lexical Rules in the Hierarchical Lexicon.* Stanford doctoral dissertation.

[22] Flickinger, Daniel, Carl Pollard, and Tom Wasow. 1985. Structure-sharing in lexical representation. In *Proceedings of the 23rd Annual Meeting of the Association for Computational Linguistics,* 262-267.

[23] Fodor, Janet D. 1978 Parsing strategies and constraints on transformations. *Linguistic Inquiry,* 9:427-473.

[24] Franz, Alex. 1990. A parser for HPSG. Technical Report LCL-90-3, CMU Laboratory for Computational Linguistics.

[25] Gazdar, Gerald et al. 1985. *Generalized Phrase Structure Grammar.* Harvard University Press.

[26] Goldstein, Ira and B.Roberts. 1977. The FRL Manual, MIT-AI Memo 409, MIT, Cambridge.

[27] Gunji, Takao. 1987. *Japanese Phrase Structure Grammar.* Dordrecht: Reidel.

[28] Hukari, Tom and Robert Levine. 1991. On the Disunity of Unbounded Dependency Constructions. *Natural Language and Linguistic Theory* 9:97-144.

[29] Jackendoff, Ray. 1972. *Semantic Interpretation in Generative Grammar.* Cambridge: MIT Press.

[30] Jackendoff, Ray. 1975. *Tough* and the trace theory of movement rules. *Linguistic Inquiry,* 6:437-446.

[31] Jacobson, Pauline. 1982. Evidence for gaps. In Pauline Jacobson and Geoffrey K. Pullum (eds.) *The Nature of Syntactic Representation.* Dordrecht: Reidel, 187-228.

[32] Jacobson, Pauline. 1984. Connectivity in Phrase Structure Grammar. *Natural Language and Linguistic Theory*, 1:535-581.

[33] Jacobson, Pauline. 1987. Phrase Structure, Grammatical Relations, and Discontinuous Constituents. Geoffrey J. Huck and Almerindo E. Ojeda (eds.) *Syntax and Semantics 20: Discontinuous Constituency*. New York: Academic Press, 27-70.

[34] Jacobson, Pauline. 1990. Raising as Function Composition. *Linguistics and Philosophy*, 13(4):423-476.

[35] Johnson, Mark. 1988. *Attribute Value Logic and the Theory of Grammar*. Stanford: CSLI.

[36] Jones, Charles. 1990. Decapitation (of some so-called "Null-Operator Constructions"). In Aaron Halpern (ed.) *Proceedings of the 9th West Coast Conference on Formal Linguistics*. Stanford: CSLI, 317-30.

[37] Kaplan, Ronald and Joan Bresnan. 1982. Lexical-Functional Grammar: A formal system for grammatical representation. In Joan Bresnan (ed.) *The Mental Representation of Grammatical Relations*. Cambridge: MIT Press, 173-281.

[38] Krieger, Hans-Ulrich, and John Nerbonne. 1991. Feature-Based Inheritance Networks for Computational Lexicons. DFKI Research Report, DFKI, Saarbrücken. To appear in: Ted Briscoe and Anne Copestake (eds.) *Proceedings of the ACQUILEX Workshop on Lexical Inheritance*. Cambridge, UK.

[39] Lasnik, Howard and Robert Fiengo. 1974. Complement object deletion, *Linguistic Inquiry*, 5:535-571.

[40] Maling, Joan and Annie Zaenen. 1982. A phrase structure account of Scandinavian extraction phenomena. In Pauline Jacobson and Geoffrey K. Pullum (eds.) *The Nature of Syntactic Representation*. Dordrecht: Reidel, 229-282.

[41] Miller, George and Noam Chomsky. 1963. Finitary Models of Language Users. In R.D.Luce, R.Bush and E.Galanter (eds.) *Handbook of Mathematical Psychology* Vol.II. New York: Wiley, 419-492.

[42] Montague, Richard. 1973. The proper treatment of quantification in ordinary English, in Jaakko Hintikka, Julius Moravcsik, and Patrick Suppes (eds.) *Approaches to Natural Language*. Dordrecht: Reidel.

[43] Nanni, Deborah L. 1980. On the surface syntax of constructions with easy-type adjectives. *Language*, 56:568-81.

[44] Nerbonne, John and Derek Proudian. 1987. The HP-NL system. Technical report, Hewlett-Packard Labs.

[45] Pollard, Carl. 1984. *Head Grammars, Generalized Phrase Structure Grammars, and Natural Language*. Stanford doctoral dissertation.

[46] Pollard, Carl. 1985. Phrase structure grammar without metarules. In *Proceedings of the 4th Annual Meeting of the West Coast Confernece on Formal Linguistics*.

[47] Pollard, Carl. 1988. Categorial grammar and phrase structure grammar: An excursion on the syntax-semantics frontier. In Richard T. Oehrle, Emmon Bach, and Deidre Wheeler (eds.) *Categorial Grammars and Natural Language Structures*. Dordrecht: Reidel, 391-416.

[48] Pollard, Carl. 1989. The syntax-semantics interface in a unification-based phrase structure grammar. In Stephan Busemann, Christa Hauenschild, and Carla Umbach (eds.) *Views of the Syntax-Semantics Interface: Proceedings of the Workshop "GPSG and Semantics", Technische Universität Berlin, 22-24.Feb 1989*, 167-184. KIT FAST, Technische Universität Berlin.

[49] Pollard, Carl and Ivan Sag. 1987. *An Information-Based Theory of Syntax and Semantics, Vol.I.* Stanford: CSLI.

[50] Pollard, Carl and Ivan Sag. 1988. An information-based theory of agreement. In Diana Brentari, Gary Larson, and Lynn McCleod, editors, *Proceedings from the Parasession on Agreement*, Chicago: Chicago Linguistics Society, 236-257.

[51] Pollard, Carl and Ivan Sag. 1991. An information-based theory of syntax and semantics, Vol.II. unpublished manuscript in preparation.

[52] Postal, Paul. 1971. *Cross-over Phenomena.* New York: Holt, Rinehart, and Winston.

[53] Proudian, Derek and Carl Pollard. 1985. Parsing head-driven phrase structure grammar. In *Proceedings of the 25th Annual Meeting of the Association for Computational Linguistics.*

[54] Purdy, William C. 1988. A Lexical Extension of Montague Semantics, Report CIS-88-1, School of Computer and Information Science, Syracuse University.

[55] Roberts, Diana. 1991. Linking Rules in an HPSG Lexicon, unpublished Master's Thesis, Cornell University.

[56] Rosenbaum, Peter S. 1967. *The Grammar of English Complement Constructions.* Cambridge: MIT Press.

[57] Ross, John R. 1967. *Constraints on Variables in Syntax.* MIT doctoral dissertation.

[58] Sag, Ivan A. 1982. A semantic theory of 'NP movement' dependencies. In Pauline Jacobson and Geoffrey K. Pullum (eds.) *The Nature of Syntactic Representation.* Dordrecht: Reidel, 427-466.

[59] Schachter, Paul. 1981. Lovely to look at. *Linguistic Analysis*, 8:431-48.

[60] Shieber, Stuart, 1986. *An Introduction to Unification-Based Approaches to Grammar.* Stanford: CSLI.

Appendix A: Lexical Fundamentals

In the analysis of adjectives governing VP complements that we provide here, we adopt the approach to lexical representation developed in Flickinger (1987). We provide here a brief sketch of the framework, limiting our discussion to those aspects which are relevant to the analysis in the main body of the paper.

Lexical Framework

Much of the information in a fully specified entry within the lexicon is not unique to that particular entry. Viewing this information as a set of discrete properties which make up the lexical entry, related lexical items will have in common some of the properties of that entry. Lexical items can be grouped into classes defined by those properties that are common to all members of the class. By giving a precise characterization of these word classes, one can eliminate a good deal of the redundancy found in a lexicon that consisted of fully specified entries. Put differently, one can make use of these word classes to capture generalizations about the elements of the lexicon, and to make predictions about the behavior and distribution of a lexical item on the basis of the classes it belongs to.

To avoid redundancy entirely, each property relevant to representing the elements of a lexicon should only be mentioned once in some single class, with all elements of the lexicon that have this property being members of that class. If so, a given lexical item may have to belong to many classes in order to obtain all of its properties. These word classes form a hierarchy over which

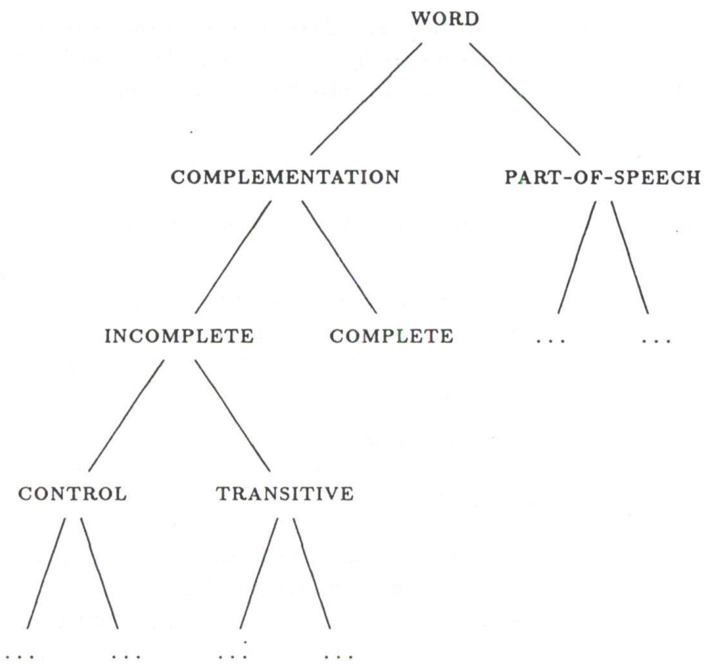

Figure 7: A sample of lexical classes near the top of the lexical hierarchy.

rules are defined that govern the inheritance of information from class to class, and ultimately to individual lexical entries. One of the nicest aspects of this idea is that it is readily visualized (cf. Figure 7).

In order to present the class definitions that formally express the properties of the adjectives governing VP complements as collections of attributes with assigned values, we shall make use of the hierarchy of word classes above, where classes inherit their properties from other more general classes, or indeed from several such classes. This hierarchy of classes is intended as the repository of both the syntactic and semantic properties that comprise the fully fleshed-out lexical entries of a language; we will of course only present a very few of the classes that would populate a complete hierarchy, sufficient we hope to prove the promise of such a representation. Part of this hierarchy is devoted to specifying the number and types of complements that predicates allow or require, and among this group of word classes is one defining the properties shared by lexical entries that take a subject and a complement which is semantically controlled (possibly by the subject); we label this class simply CONTROL. Above, we illustrate the barest outlines of the top of the word class hierarchy for English, to suggest where the CONTROL class fits in. We provide the content of CONTROL later.

It is important to note that word classes do not merely form a simple hierarchy, but rather that they form a set of interconnected hierarchies; as Chomsky (1965) argued in his discussion of lexical representation in *Aspects* (pp. 79-83), the need to cross-classify a given lexical item according to several distinct properties renders impossible the use of a simple branching hierarchy to represent the lexicon.

We take as adequate motivation for the existence of a word class the demonstration that some particular syntactic or morphological property (or cluster of properties) is shared by a number of lexical items. The forcing function in the identification of word classes is our assumption that the best representation for lexical information is one in which each new piece of information, each distinct property exhibited by one or more elements of the lexicon, is introduced exactly once in the lexicon. A property shared by more than one lexical item should be introduced in a word class common to those items, or those lexical items should be related by lexical rule. In the simplest case where a word class or lexical entry belongs to just one superclass, the inheritance rule for how values of an attribute are assigned in the word class hierarchy is quite straightforward. The two alternatives to be considered depend on whether a given attribute permits only one value, or multiple values.

Inheritance of Values

> The value assigned to a particular word class (or member) W for a given attribute is determined as follows:
>
> a. For a single-valued attribute, the assigned value is either introduced directly in W, or is the one introduced in the most specific class to which W belongs. If there is no value introduced anywhere in the linked classes between W and the root WORD-CLASS, inclusively, no value is assigned to W for that attribute.
>
> b. For a multiple-valued attribute, the assigned values are the members of the set consisting of all distinct values introduced for that attribute in W and in any of the classes linking W with the root WORD-CLASS, inclusively.[21]

In cases where a class or member belongs to more than one superclass, the picture might be more complicated, since each of two immediate superclasses might specify a different value for the same single-valued attribute. One way to address the potential conflict would be to define another rule of inheritance to take account of multiple parents, a rule which for each attribute assigns priority to some one of the parent classes (see the paper on the ISSCO lexicon in this collection). We instead assume that the hierarchy is specified in such a way that each single-valued attribute of some given class or member was assigned a value by at most one of the immediate superclasses (or its parents), so conflicting values do not occur.

In adopting this discipline for inheritance of lexical information, we follow Flickinger (1987) rather than accepting the more rigorously defined but more restrictive rules of inheritance defined for DATR by Evans and Gazdar (1989, 1990). As will be seen in the discussion to follow, we believe that the ability to inherit a lexical property from more than one potential source can be important in capturing relevant linguistic generalizations. There would be no point to our use of multiple-valued attributes if we did not allow genuine multiple inheritance. We employ the multiple-valued attribute "complements" in order to collect subcategorization information about several subcategorized-for elements simultaneously. In its most natural form, the single property "complements" inherits from several ancestors simultaneously.[22]

We also follow Flickinger, this time accompanied by Evans and Gazdar, in assuming that lexical attributes may be assigned default values as part of a word class definition, with those defaults possibly being overridden in the definition of a subclass or lexical entry inheriting from that word class. In this we part ways with Pollard and Sag (1987)'s strong assumption of monotonicity in the inheritance of lexical properties, again for reasons that we identify in the discussion to follow.[23]

While word classes and the associated mechanism of nonmonotonic inheritance provide powerful tools for representing one kind of shared information, that which links a category to its sub-categories, a distinct formal device is required to link two morphologically related classes of different categories. We employ the familiar notion of a lexical rule to represent this second kind of systematic (but not exceptionless) relationship. Given a word belonging to the first set, a lexical rule predicts the existence of a corresponding word belonging to the second set, with the differences and similarities between the two words captured both in the formulation of the rule, and in the definitions of the classes each word is a member of.

[21] We employ multiple-valued attributes only within the lexicon in order to gather subcategorization speicifications. A translation step converts these to sets (or—with more information—lists) for use in the feature system.

[22] The DATR position carefully disallows multiple inheritance of a single property from two or more classes, even while allowing inheritance from various classes into different properties (in a single word class). We find genuine multiple inheritance seems useful, even if it may be dispensable; cf. the treatment of the lexical attribute "complements" below.

[23] But see Pollard and Sag, 1987, p.194 note.

We illustrate our notation for lexical rules with an example of a relatively simple one, the inflectional rule relating the base forms of verbs with their past tense forms. We can express the relationship between, say, *walk* and *walked* as given in the LR-PAST rule below, leaving out specifics of phonology and only hinting at semantics.

LR-PAST lexical rule

LR-PAST		
LE2-Classes - PAST	=	LE1-Classes - BASE
LE2-Spelling	=	(AFFIX-ED LE1-Spelling)
LE2-Phonology	=	...
LE2-Semantics	=	(PAST LE1-Semantics)

This lexical rule, like any other, expresses a relation holding between two sets of lexical entries, the first set represented by a canonical lexical entry LE1, and the second set by LE2. The rule's applicability is governed by the relevant classes that LE1 and LE2 each belong to, with these classes named in the statement within the rule that relates the one entry's list of parent classes with the other entry's class list. Having specified the range of applicability, each rule then states the particular dependencies holding between properties of LE1 and corresponding properties of LE2.[24]

We might view inheritance within a hierarchy of word classes as a tool that eliminates redundancy along one dimension within the lexicon, while lexical rules provide the same service along another dimension. A given lexical item, by virtue of being a member of one or more word classes, shares inherited properties with other lexical items that belong to those same classes, but does not necessarily share a common morphological or semantic base (or indeed any of its idiosyncratic properties) with any of those other items. That same lexical item, by participating in one or more lexical rules, has properties in common with a second set of lexical items, where the shared properties crucially include some or all of the idiosyncratic information which distinguishes that lexical item from others in its class. The members of this second set, related by lexical rules, all do share a single common semantic and morphological base (except for suppletions).

Of course, if a lexical rule relates two entries that both belong to a given word class (as happens with the verbal inflection rules), those two entries will share some inherited properties as well as the idiosyncracies. However, the lexical rule only establishes joint membership in that given class and the relationship of the idiosyncratic information in the two entries; all other properties shared by the two are established by inheritance within the word class hierarchy.

Both of these formal devices, inheritance and lexical rules, serve to express that which is common among (often overlapping) sets of fully-specified lexical entries, including properties that are morphological, syntactic, and semantic. In their capacity as redundancy mechanisms, the two devices permit a parsimonious representation of the existing lexicon.

Lexical Content

For convenience of exposition, we view the syntactic properties of lexical items as being of two kinds: one a set of features, separated into those with atomic values and those with category values, and the other a set of subcategorization specifications, divided into complements (obligatory and optional), and adjuncts. Even though HPSG represents both types of information (features and subcategorization specifications) uniformly as attribute-value pairs (as noted above), we shall represent them lexically as distinct. We have two motivations for this: first, in representing the information differently from its normal form in HPSG we demonstrate the independence of the lexical ideas presented here. We employ HPSG in the grammatical analysis presented here because it is a useful grammatical framework, and because it makes strenuous lexical demands; but the lexicon framework does not presuppose HPSG (for example, PATR-II systems can make

[24] We include this rule primarily in order to introduce the notation we shall later employ. We would, however, be quite sympathetic to an alternative treatment of the relation between PAST tense forms and untensed lexemes which employed lexical inheritance rather than inflectional rules to account for the relationship. We deny that this sort of treatment can be extended from inflectional to derivational rules, however.

use of structured lexicons, and nearly do, in the form of templates. Cf. Shieber, 1986, pp.54-55.). Second, a uniform feature notation needed for subcategorization and nonsubcategorization information threatens to obscure points addressed below, so the two kinds of information are separated in the representations of lexical entries used here.

The atomic-valued features that we employ in specifying lexical entries (and in specifying categories) are drawn from a (small) finite set where each feature has a limited set of possible atomic values,[25] e.g., the binary feature INVERTED, indicating whether or not a verb can appear as the head of an inverted sentence. The feature VFORM, on the other hand, draws its values from a set containing among others BASE, PAST, and PAST-PARTICIPLE, to represent the morphological form of a verb. Category-valued features take as their value a feature-structure, a specification for some syntactic category. Since any non-empty set of feature-value pairs is (by definition) enough to specify a category, any such set constitutes a possible value for one of these category-valued features.

Each complement or adjunct entry, referred to here as a *subcat*, consists of a category specification and its semantic properties. Since reference to subcategorization properties of subcats is excluded in specifying complements or adjuncts within a lexical entry (cf. Pollard and Sag, 1987: 143-4 for a similar—but not identical—"locality" restriction), we make use of a feature COMPLETE, quite similar to the SUBJ feature proposed by Borsley (1983), and employed in GKPS (1985:61f) to distinguish INCOMPLETE from COMPLETE categories. Incomplete constituents lack one or more of their obligatory complements, including at least their final complement (usually the subject), and are marked [COMPLETE –], while complete categories are marked [COMPLETE +] to represent the property that no obligatory arguments are missing. (Complete categories correspond roughly to maximal projections in an X-bar framework.) To distinguish lexical categories from phrasal ones, we use the binary feature LEXICAL. With these two features COMPLETE and LEXICAL, we can follow Pollard (1984) in dispensing with the widely used (and abused) X-bar machinery, while maintaining the full range of necessary distinctions among lexical and phrasal categories.

The content of the word-class CONTROL, promised above, is presented here:

(38)

CONTROL	
Superclasses	Incomplete
Complements	XComp
XComp-Features	(Category Verb) (Complete –) (Lexical –)
XComp-Subj-Semantics	Subject-Semantics
XComp-Oblig	Yes
XComp-Semantics	XComp-Semantics
XComp-Role	State-of-Affairs

As the hierarchy of word classes sketched above indicates, this class inherits from the INCOMPLETE class (which specifies an obligatory subject complement), and introduces a second obligatory complement which is a verb phrase (not complete, which would be its maximal projection, a sentence; and not lexical, which would be just a verb without any complements). It will play the role State-of-Affairs (abbreviated 'soa') in relations denoted by words inheriting from CONTROL, and it will be semantically interpreted by the variable XComp-Semantics. The specification of the semantics will occasionally be omitted below, since the convention should be clear. It is this CONTROL class which will serve as the superclass from which both of the adjectival VP complement classes (cf. IT-EASY and SLASH-EASY below) inherit.

Before concluding the sketch of the lexicon, we turn to the lexical representation of semantics, which likewise plays a role in the final analysis.

[25] For linguistic defense of many of the actual features used here, see GKPS (1985)

Lexical Semantics

The use of hierarchies of classes of information, advocated here as a means of representation for grammatical information, is also common in the representation of semantic HYPONYMY relations, e.g., the relation between *boy* and *child* (cf. Brachman and Schmolze, 1985). Thus *boy* is a hyponym (subconcept) of both *child* and *male*, which are in turn hyponyms of more abstract words and concepts. This may be modeled in the same multihierarchical fashion we employ for grammatical information. Such semantic hierarchies may be of utility in constructing more efficient NL inference engines (Purdy, 1988).

We exploit a very different use of semantic inheritance in the present treatment, however, beginning with the observation familiar from categorial grammar (Bach, 1988) that semantics and subcategorization are interdependent: subcategorizers denote relations among the denotations of their complements. Montague (1973) effectively exploited this by interpreting multiplace verbs and verb phrases as functions into the denotations of lesser-place verbs. We exploit the interdependence by allowing some semantic inheritance to follow syntactic subcategorization lines. To be more precise, we allow subcategorizers to specify not only the syntax of their complements, but also the semantic ROLE the complement is assigned in the relation denoted by the subcategorizer.

INCOMPLETE	
Superclasses	Complementation
Complements	Subject
Subject-Features	(Complete +)(Category Noun)
Subject-Role	Source

TRANSITIVE	
Superclasses	Incomplete
Complements	Object
Object-Features	(Complete +) (Case Accusative)(Category Noun)
Object-Role	Theme

Thus INCOMPLETE assigns its subject the role *source*; TRANSITIVE inherits this role assignment and extends it by assigning *theme* to objects.

Roles may be understood by their function in atomic formulas: in standard predicate logic the binding of arguments to argument positions is mediated by the order in which argments appear. $Rxy \not\equiv Ryx$. The use of explicit roles in semantic relations accomplishes this task and obviates the order of arguments: $R(\text{source:}x, \text{theme:}y) \equiv R(\text{theme:}y, \text{source:}x)$. Role-coded formulas are easily readable when there are many roles, and the use of roles seems essential in semantic theories of topics such as variable-place relations or variably-binding arguments such as the possessive. There is furthermore a substantial body of work on the so-called "linking" of semantic roles to syntactic information, including especially Roberts (1991) who applies this theory to HPSG. (Even though we use role- or keyword-coded arguments in lexical specifications, we will occasionally revert to order-coded representations for the purposes of illustration. They are more concise.)

What is important about roles for the present application is that we may exploit the inheritance mechanism to derive (specifications for) semantics for multiplace subcategorizers. Instances of the TRANSITIVE class are assigned the following semantics, using the multi-valued inheritance scheme discussed above:

$$\begin{bmatrix} \text{pred:} & \\ \text{source:} & \textbf{Subject-Semantics} \\ \text{theme:} & \textbf{Object-Semantics} \end{bmatrix}$$

Lexical specifications for the arguments to the roles have not been shown, but the general scheme should be clear. The predicate must of course be assigned by each individual lexical entry. "Subject-Semantics" is used because it is useful to be able to refer to the semantics of a given complement (the controller) in cases of grammatical control, as the semantics specifications for the lexical class CONTROL demonstrate.

Exceptions to Lexical Rules

Since few if any lexical rules prove to be completely exceptionless, we assume that individual lexical entries can and do stipulate among their idiosyncratic properties exceptional behavior with respect to particular lexical rules. One such exceptional property is that a given entry belongs to a class to which a lexical rule applies, but that rule is not applicable to this entry.[26] Thus in the present case, adjectives like *necessary* and *possible* will include as part of their sparse lexical entries the stipulation that the lexical rule which usually relates IT-SUBJ and S-SUBJ members does not apply to these entries. Calling that lexical rule LR-Intraposition, given in (39), the entry for *necessary* can then be represented as in (40).

LR-Intraposition lexical rule

(39)

LR-Intraposition		
LE2-Classes - IT-SUBJ	=	LE1-Classes - S-SUBJ

(40)

necessary-1	
Superclasses	Adjective, It-Subj
Spelling	"necessary"
Phonology	/nEsIseri/
Lexical-Rules	(LR-Intraposition Not-Applicable)

It may be worth noting that it seems unlikely that properties such as the applicability of lexical rules can be incorporated into the feature system of HPSG (i.e., in any explanatory way). They seem inexpressible because they are a kind of second-order property. This is, in fact, exactly the sort of information which suggests to us that a lexicon may have to be MORE than a particular kind of feature system. See Pollard and Sag (1987:209, note) for a concurring view.[27]

Appendix B: Refinements and Lexical Modifiability

In addition to allowing extensions painlessly, we expect a lexical system to be easily modified. This is of practical value given the relatively inexact state of present linguistic knowledge. Linguistic descriptions are under frequent revision, and lexical systems must accommodate this. In the present section we examine several refinements of the analyses above as a means of demonstrating the modifiability of structured lexicons. We wish to underscore the richness of detail that demands accommodation even in this one corner of the lexicon, and we hope to probe the limits of the formalism we have adopted for this lexical representation.

We began our analysis by presenting two variants of adjectives like *easy* , one with an expletive *it* subject, and one with a normal NP subject and a verbal complement containing a gap. There is, of course, a third variant for most adjectives of this kind, one with an infinitival VP or S as its

[26] For a more complete discussion of the types of exceptional behavior exhibited by lexical entries with respect to lexical rules in this framework, see Flickinger 1987,pp.122ff.

[27] The initial implementation of the lexicon was reported in Flickinger et al. (1985), and was done in HP-RL, a language derived from MIT's frame representation language, FRL (Goldstein and Roberts, 1977). It has since undergone reimplementations in Common Lisp, Common Objects, and CLOS. The work reported on here was implemented and saw daily (experimental) use for over two years. The basic analyses in Section 4 were all implemented and thoroughly tested through a good variety of surrounding grammars and application efforts. We also suggest analyses in the main body of the paper that were not implemented fully, in particular in the section on adjectival specification (*too* and *enough*) and nondenoting nominals *S. is a pleasure to see.*

subject, and no verbal complement, as illustrated in (41).

(41) a. To talk to Bill would be great.
b. For me to talk to Bill would be great.
c. For Bill to lose this race would be great for Mary.

This selection for infinitival subjects is a property shared with other classes of well-studied predicates, including verbs like *bother* and *require*, as illustrated in (42).

(42) a. For me to talk to Bill would bother Mary.
b. To win this race will require your fullest commitment.

To see that these infinitival subjects must be lexically licensed, consider the examples in (43), where at least some members of the IT-SUBJ class cannot appear with such subjects.

(43) a. *In the final analysis, to win this race will not be necessary.
(cf. Winning this race will not be necessary.)
b. *To talk to Bill is possible only in the mornings.

We thus define a subclass of INCOMPLETE here named S-SUBJ, similar to IT-SUBJ and SLASH-COMP, to identify the relevant properties exhibited by lexical entries for the adjectives and verbs in (41-42).

(44)

S-SUBJ	
Superclasses	Incomplete
Complements	
Subject-features	(Category Verb) (VForm Infinitival) (Complete + −)

To this class belong verbs like *require* and *bother*, but we must also define a subclass which we call S-EASY for adjectives like *great* and *difficult*, since these, unlike the verbs, also permit an optional PP-for phrase, provided as before by the FOR-EXPERIENCER class.

(45)

S-EASY	
Superclasses	S-Subj, For-Experiencer

It appears that in general adjectives of the IT-EASY class alternate with entries like those in (41), while adjectives of the IT-SUBJ class do not. Hence we are tempted to define the lexical rule relating adjectives having *it* subjects with those having infinitival subjects so that the rule holds between the two classes S-EASY and IT-EASY. However, verbs like *bother* and *require* with entries that are members of the IT-SUBJ class should also be covered by this same lexical rule, suggesting that it must hold between the two classes IT-SUBJ and S-SUBJ. This leaves us the task of excluding those IT-SUBJ adjectives like *necessary* and *possible* that do not have S-SUBJ counterparts.

Distinctions among Unbounded Dependencies

A second refinement of our analysis of *easy* adjectives is motivated by examples like those in (46), which show that some further constraints need to be placed on the gappy verbal complement which such adjectives subcategorize for. Informally, the generalization seems to be that extraction is not possible out of finite clauses embedded within the complement to *easy* adjectives, but is otherwise licensed.[28]

(46) a. Bill was easy to get Mary to hire.
b. Palm trees are hard to learn to climb.
c. Arias are fun to try to sing.

d. *Bill was easy to see that Mary admired.
(cf. Bill, it's easy to see that Mary admires.)
e. *Palm trees are hard to learn that one can climb.
f. *Arias are fun to insist that people sing.

General constraints imposed by the framework we have adopted here prevent us from attempting to describe these facts by making *easy* adjectives select for a verbal complement whose head requires ITS complement to be non-finite.[29] Instead, we follow Hukari and Levine (1991), who propose that two types of unbounded dependencies might be distinguished, with one dependency path, marked by the new binding feature SLASH′, treating finite S's as islands, while the ordinary SLASH feature marks the usual dependency path which is insensitive to finite S nodes. Then *easy* adjectives of the SLASH-EASY class would more precisely subcategorize for an infinitival complement that has an NP gap of the marked SLASH′ variety rather than the usual SLASH. To illustrate, consider example (46d): the SLASH′ feature that would (by hypothesis) be introduced at the extraction site within the embedded $\overline{S}$ *that Mary admired* will be passed up from that site by a general principle, but will stop its ascent when it reaches that $\overline{S}$. Assuming that the verb *see* does not select for this unusual kind of SLASH′ complement, the sentence will not be admitted as grammatical. In contrast, the example in (46a) will still be admitted since the SLASH′ introduced at the extraction site in *to hire* will be faithfully passed up by the same binding inheritance principle until it reaches the node dominating *to get Mary to hire*. Now this VP[SLASH′ NP] is, according to our proposed refinement, precisely the kind of complement that *easy* requires, so the sentence is grammatical.

Of course, to properly defend this addition of SLASH′ to the collection of binding features for English, we would like to find independent evidence of the claim that finite S's can serve as islands for SLASH. We leave the matter here as one meriting further study, and refer the interested reader to Jacobson (1987) for a similar suggestion to distinguish various slash attributes.

Pied Piping

A third refinement of the analysis given above is motivated by examples like those in (47), where the *easy* adjective appears in a noun phrase with an infinitival complement containing a pied-piping construction, not accounted for in what we have said thus far.[30]

(47) a. Mary is an easy boss for whom to work.
b. New York would be an awkward city from which to flee.
c. Bill might be a hard person in whom to confide.

The most straightforward characterization of phrases like *for whom to work* is to describe them as infinitival VPs containing a relative pronoun. Since English independently prohibits relative pronouns from appearing in situ within a verb phrase, the only way such a phrase can be produced is to have a pied-piped prepositional phrase extracted from the VP and sister to it.[31] Assuming that such a phrase must be admitted by the grammar, we can formally represent its syntactic category as shown in the following definition for the new class REL-EASY. What we make

[28] Jones (1990) attributes the observation to Chomsky (1977).

[29] For discussion, see Flickinger, 1987, pp.67ff; Pollard and Sag, 1987, p.143

[30] Chomsky (1977) cites examples like these; we appreciate Anthony Kroch's bringing them to our attention.

[31] Among other details, the phrase structure linking rule which would be necessary to admit this [VP[REL NP] → PP[REL NP], VP/NP] construction will also have to be made explicit in a fuller analysis than we provide here, but that should not be problematic. We have in mind a simple generalization of the sentential linking rule, so that no novel rule would be required.

explicit here is the idea that adjectives of the SLASH-EASY class have corresponding members (linked via a lexical rule similar to ones seen above) that take an unusual kind of VP complement, differing further in that these REL-EASY adjectives do not seem to license an optional PP-For complement. The final property identified in this class is that its members are marked as not predicative, effectively restricting its members to attributive adjectives. This property accounts for the ungrammaticality of the examples given in (49).

(48)

REL-EASY	
Superclasses	Control,Adjective
Features	(Predicative –)
Complements	
XComp-features	(VForm Infinitival)
	(REL (Category Noun) (Complete +)
	(NForm Normal) (Predicative –)
	(Case Accusative Dative))

(49) a. *Bill is easy for whom to work.
b. *Bill is a pleasure for whom to work.

In both examples in (49), the complement of the copula *is* must be predicative, but the phrases headed by *easy* and *pleasure* would have to be non-predicative in order for those heads to license THEIR VP[REL NP] complements. Indeed, (49b) is also ruled out by another constraint which we finessed in our brief introduction of *pleasure* nouns, for clarity of exposition: we described such nouns as belonging to the ordinary COMMON-NOUN class, but that assignment also needs refining in a more detailed account, since *pleasure* nouns exhibit only a few of the properties of regular common nouns. In particular, it seems clear that these nouns must be predicative, and hence unfit for membership in the REL-EASY class, since attempting to assign them to this class would introduce a conflict of inherited values for the attribute PREDICATIVE, and such conflicts are prohibited, as we noted above in our introduction to the general framework.

Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

DFKI
-Bibliothek-
PF 2080
6750 Kaiserslautern
FRG

DFKI Publikationen

Die folgenden DFKI Veröffentlichungen oder die aktuelle Liste von erhältlichen Publikationen können bezogen werden von der oben angegebenen Adresse.
Die Berichte werden, wenn nicht anders gekennzeichnet, kostenlos abgegeben.

DFKI Publications

The following DFKI publications or the list of currently available publications can be ordered from the above address.
The reports are distributed free of charge except if otherwise indicated.

DFKI Research Reports

RR-90-01
Franz Baader: Terminological Cycles in KL-ONE-based Knowledge Representation Languages
33 pages

RR-90-02
Hans-Jürgen Bürckert: A Resolution Principle for Clauses with Constraints
25 pages

RR-90-03
Andreas Dengel, Nelson M. Mattos: Integration of Document Representation, Processing and Management
18 pages

RR-90-04
Bernhard Hollunder, Werner Nutt: Subsumption Algorithms for Concept Languages
34 pages

RR-90-05
Franz Baader: A Formal Definition for the Expressive Power of Knowledge Representation Languages
22 pages

RR-90-06
Bernhard Hollunder: Hybrid Inferences in KL-ONE-based Knowledge Representation Systems
21 pages

RR-90-07
Elisabeth André, Thomas Rist: Wissensbasierte Informationspräsentation:
Zwei Beiträge zum Fachgespräch Graphik und KI:
1. Ein planbasierter Ansatz zur Synthese illustrierter Dokumente
2. Wissensbasierte Perspektivenwahl für die automatische Erzeugung von 3D-Objektdarstellungen

24 Seiten

RR-90-08
Andreas Dengel: A Step Towards Understanding Paper Documents
25 pages

RR-90-09
Susanne Biundo: Plan Generation Using a Method of Deductive Program Synthesis
17 pages

RR-90-10
Franz Baader, Hans-Jürgen Bürckert, Bernhard Hollunder, Werner Nutt, Jörg H. Siekmann: Concept Logics
26 pages

RR-90-11
Elisabeth André, Thomas Rist: Towards a Plan-Based Synthesis of Illustrated Documents
14 pages

RR-90-12
Harold Boley: Declarative Operations on Nets
43 pages

RR-90-13
Franz Baader: Augmenting Concept Languages by Transitive Closure of Roles: An Alternative to Terminological Cycles
40 pages

RR-90-14
Franz Schmalhofer, Otto Kühn, Gabriele Schmidt: Integrated Knowledge Acquisition from Text, Previously Solved Cases, and Expert Memories
20 pages

RR-90-15
Harald Trost: The Application of Two-level Morphology to Non-concatenative German Morphology
13 pages

RR-90-16
Franz Baader, Werner Nutt: Adding Homomorphisms to Commutative/Monoidal Theories, or: How Algebra Can Help in Equational Unification
25 pages

RR-90-17
Stephan Busemann: Generalisierte Phasenstrukturgrammatiken und ihre Verwendung zur maschinellen Sprachverarbeitung
114 Seiten

RR-91-01
Franz Baader, Hans-Jürgen Bürckert, Bernhard Nebel, Werner Nutt, Gert Smolka: On the Expressivity of Feature Logics with Negation, Functional Uncertainty, and Sort Equations
20 pages

RR-91-02
Francesco Donini, Bernhard Hollunder, Maurizio Lenzerini, Alberto Marchetti Spaccamela, Daniele Nardi, Werner Nutt: The Complexity of Existential Quantification in Concept Languages
22 pages

RR-91-03
B.Hollunder, Franz Baader: Qualifying Number Restrictions in Concept Languages
34 pages

RR-91-04
Harald Trost: X2MORF: A Morphological Component Based on Augmented Two-Level Morphology
19 pages

RR-91-05
Wolfgang Wahlster, Elisabeth André, Winfried Graf, Thomas Rist: Designing Illustrated Texts: How Language Production is Influenced by Graphics Generation.
17 pages

RR-91-06
Elisabeth André, Thomas Rist: Synthesizing Illustrated Documents A Plan-Based Approach
11 pages

RR-91-07
Günter Neumann, Wolfgang Finkler: A Head-Driven Approach to Incremental and Parallel Generation of Syntactic Structures
13 pages

RR-91-08
Wolfgang Wahlster, Elisabeth André, Som Bandyopadhyay, Winfried Graf, Thomas Rist:
WIP: The Coordinated Generation of Multimodal Presentations from a Common Representation
23 pages

RR-91-09
Hans-Jürgen Bürckert, Jürgen Müller, Achim Schupeta: RATMAN and its Relation to Other Multi-Agent Testbeds
31 pages

RR-91-10
Franz Baader, Philipp Hanschke: A Scheme for Integrating Concrete Domains into Concept Languages
31 pages

RR-91-11
Bernhard Nebel: Belief Revision and Default Reasoning: Syntax-Based Approaches
37 pages

RR-91-12
J.Mark Gawron, John Nerbonne, Stanley Peters:
The Absorption Principle and E-Type Anaphora
33 pages

RR-91-13
Gert Smolka: Residuation and Guarded Rules for Constraint Logic Programming
17 pages

RR-91-14
Peter Breuer, Jürgen Müller: A Two Level Representation for Spatial Relations, Part I
27 pages

RR-91-15
Bernhard Nebel, Gert Smolka: Attributive Description Formalisms ... and the Rest of the World
20 pages

RR-91-16
Stephan Busemann: Using Pattern-Action Rules for the Generation of GPSG Structures from Separate Semantic Representations
18 pages

RR-91-17
Andreas Dengel, Nelson M. Mattos:
The Use of Abstraction Concepts for Representing and Structuring Documents
17 pages

RR-91-18
John Nerbonne, Klaus Netter, Abdel Kader Diagne, Ludwig Dickmann, Judith Klein:
A Diagnostic Tool for German Syntax
20 pages

RR-91-19
Munindar P. Singh: On the Commitments and Precommitments of Limited Agents
15 pages

RR-91-20
Christoph Klauck, Ansgar Bernardi, Ralf Legleitner
FEAT-Rep: Representing Features in CAD/CAM
48 pages

RR-91-21
Klaus Netter: Clause Union and Verb Raising Phenomena in German
38 pages

RR-91-22
Andreas Dengel: Self-Adapting Structuring and Representation of Space
27 pages

RR-91-23
Michael Richter, Ansgar Bernardi, Christoph Klauck, Ralf Legleitner: Akquisition und Repräsentation von technischem Wissen für Planungsaufgaben im Bereich der Fertigungstechnik
24 Seiten

RR-91-24
Jochen Heinsohn: A Hybrid Approach for Modeling Uncertainty in Terminological Logics
22 pages

RR-91-25
Karin Harbusch, Wolfgang Finkler, Anne Schauder: Incremental Syntax Generation with Tree Adjoining Grammars
16 pages

RR-91-26
M. Bauer, S. Biundo, D. Dengler, M. Hecking, J. Koehler, G. Merziger:
Integrated Plan Generation and Recognition
- A Logic-Based Approach -
17 pages

RR-91-27
A. Bernardi, H. Boley, Ph. Hanschke, K. Hinkelmann, Ch. Klauck, O. Kühn, R. Legleitner, M. Meyer, M. M. Richter, F. Schmalhofer, G. Schmidt, W. Sommer:
ARC-TEC: Acquisition, Representation and Compilation of Technical Knowledge
18 pages

RR-91-28
Rolf Backofen, Harald Trost, Hans Uszkoreit:
Linking Typed Feature Formalisms and Terminological Knowledge Representation Languages in Natural Language Front-Ends
11 pages

RR-91-30
Dan Flickinger, John Nerbonne:
Inheritance and Complementation: A Case Study of *Easy* Adjectives and Related Nouns
39pages

RR-91-31
H.-U. Krieger , J. Nerbonne:
Feature-Based Inheritance Networks for Computational Lexicons
11 pages

RR-91-32
Rolf Backofen, Lutz Euler, Günther Görz:
Towards the Integration of Functions, Relations and Types in an AI Programming Language
14 pages

DFKI Technical Memos

TM-90-02
Jay C. Weber: The Myth of Domain-Independent Persistence
18 pages

TM-90-03
Franz Baader, Bernhard Hollunder: KRIS: Knowledge Representation and Inference System -System Description-
15 pages

TM-90-04
Franz Baader, Hans-Jürgen Bürckert, Jochen Heinsohn, Bernhard Hollunder, Jürgen Müller, Bernhard Nebel, Werner Nutt, Hans-Jürgen Profitlich: Terminological Knowledge Representation: A Proposal for a Terminological Logic
7 pages

TM-91-01
Jana Köhler: Approaches to the Reuse of Plan Schemata in Planning Formalisms
52 pages

TM-91-02
Knut Hinkelmann: Bidirectional Reasoning of Horn Clause Programs: Transformation and Compilation
20 pages

TM-91-03
Otto Kühn, Marc Linster, Gabriele Schmidt:
Clamping, COKAM, KADS, and OMOS:
The Construction and Operationalization of a KADS Conceptual Model
20 pages

TM-91-04
Harold Boley:
A sampler of Relational/Functional Definitions
12 pages

TM-91-05
Jay C. Weber, Andreas Dengel , Rainer Bleisinger:
Theoretical Consideration of Goal Recognition Aspects for Understanding Information in Business Letters
10 pages

TM-91-06
Johannes Stein: Aspects of Cooperating Agents
22 pages

TM-91-08
Munindar P. Singh: Social and Psychological Commitments in Multiagent Systems
11 pages

TM-91-09
Munindar P. Singh: On the Semantics of Protocols Among Distributed Intelligent Agents
18 pages

TM-91-10
Béla Buschauer, Peter Poller, Anne Schauder, Karin Harbusch: Tree Adjoining Grammars mit Unifikation
149 pages

TM-91-11
Peter Wazinski: Generating Spatial Descriptions for Cross-modal References
21 pages

DFKI Documents

D-90-03
Ansgar Bernardi, Christoph Klauck, Ralf Legleitner: Abschlußbericht des Arbeitspaketes PROD
36 Seiten

D-90-04
Ansgar Bernardi, Christoph Klauck, Ralf Legleitner: STEP: Überblick über eine zukünftige Schnittstelle zum Produktdatenaustausch
69 Seiten

D-90-05
Ansgar Bernardi, Christoph Klauck, Ralf Legleitner: Formalismus zur Repräsentation von Geo-metrie- und Technologieinformationen als Teil eines Wissensbasierten Produktmodells
66 Seiten

D-90-06
Andreas Becker: The Window Tool Kit
66 Seiten

D-91-01
Werner Stein , Michael Sintek: Relfun/X - An Experimental Prolog Implementation of Relfun
48 pages

D-91-03
Harold Boley, Klaus Elsbernd, Hans-Günther Hein, Thomas Krause: RFM Manual: Compiling RELFUN into the Relational/Functional Machine
43 pages

D-91-04
DFKI Wissenschaftlich-Technischer Jahresbericht 1990
93 Seiten

D-91-06
Gerd Kamp: Entwurf, vergleichende Beschreibung und Integration eines Arbeitsplanerstellungssystems für Drehteile
130 Seiten

D-91-07
Ansgar Bernardi, Christoph Klauck, Ralf Legleitner
TEC-REP: Repräsentation von Geometrie- und Technologieinformationen
70 Seiten

D-91-08
Thomas Krause: Globale Datenflußanalyse und horizontale Compilation der relational-funktionalen Sprache RELFUN
137 pages

D-91-09
David Powers and Lary Reeker (Eds):
Proceedings MLNLO'91 - Machine Learning of Natural Language and Ontology
211 pages
Note: This document is available only for a nominal charge of 25 DM (or 15 US-$).

D-91-10
Donald R. Steiner, Jürgen Müller (Eds.):
MAAMAW'91: Pre-Proceedings of the 3rd European Workshop on „Modeling Autonomous Agents and Multi-Agent Worlds"
246 pages
Note: This document is available only for a nominal charge of 25 DM (or 15 US-$).

D-91-11
*Thilo C. Horstmann:*Distributed Truth Maintenance
61 pages

D-91-12
Bernd Bachmann:
$\text{H}^{\text{iera}}\text{C}_{\text{on}}$ - a Knowledge Representation System with Typed Hierarchies and Constraints
75 pages

D-91-13
International Workshop on Terminological Logics
Organizers: Bernhard Nebel, Christof Peltason, Kai von Luck
131 pages

D-91-14
Erich Achilles, Bernhard Hollunder, Armin Laux, Jörg-Peter Mohren: KRIS : Knowledge Representation and Inference System
- Benutzerhandbuch -
28 Seiten

Inheritance and Complementation: A Case Study of *Easy* Adjectives and Related Nouns

Dan Flickinger and John Nerbonne

RR-91-30

Research Report