GENERALIZED PHRASE STRUCTURE GRAMMAR:

A THEORETICAL SYNOPSIS

Gerald Gazclar & Geoffrey K. PuLLum

August 1982

Gerald Gazdar, University of Sussex

Geoffrey K. Pullum, University of California, Santa Cruz

Cognitive Science Research Paper

Serial no: CSRP 007

The University of Sussex
Cognitive Studies Programme
School of Social Sciences
Falmer
Brighton BN1 9QN

Preface

In this paper we provide an introduction to formal aspects of the generalized phrase structure grammar (GPSG) framework. It is based on work that has been done by a number of people since about 1978, and published in books and papers in various places. The published work on generalized phrase structure grammar sometimes differs from what we set out here in respect of specific assumptions, particularly with regard to peripheral matters; but for the most part we believe that the intent of existing work is preserved in what follows below.

Six points about the content are worth noting here.

(i) The present paper is much more explicit about the internal structure of categories and features in GPSG than is anything else either published or readily available.

(ii) Specific proposals are made here about the nature of default assignments to features, and a general definition is given of rules that are properly instantiated with features.

(iii) We introduce the notion "foot feature," and argue that foot features behave in a way that is systematically distinct from the behavior of head features. A percolation principle for foot features is proposed. This principle eliminates the need for, inter alia, the "slash category introduction metarule" proposed in earlier work.

(iv) An explicit account is given of the overall organization of the various components of a GPSG grammar - in particular with regard to the interaction of immediate dominance statements and metarules.

(v) The paper does not, however, contain any substantial analyses of linguistic phenomena. The analyses given generally have a merely illustrative status. The purpose of the paper is to develop and introduce the formal structure of the GPSG framework, not to demonstrate its descriptive and explanatory potential.

(vi) We adopt the position argued for by Klein and Sag (1981), that the semantic translation schemata for rules are predictable from the form of the rules taken together with the semantic types of the lexical items introduced. As a consequence, we say essentially nothing about semantics below.

For the convenience of people teaching courses on GPSG, and as a way of stimulating inquiry among interested readers in general, a number of exercises are provided at the end of the more technical sections. Many of them have a mathematical character, perhaps because we found that kind of exercise easy to think up. Some have straightforward and easily discovered answers. Some others are quite hard, but do have definite solutions. Still others represent open research problems that would merit detailed attention and might furnish suitable topics for term-paper assignments. We have not attempted to partition the exercises explicitly, but we do attempt to list them in ascending order of difficulty.

permeate the present work. Only the geographical accident of their being in Maryland during July 1982 has prevented their being co-authors of this work (and thus absolved from responsibility for our errors). We have benefited immeasurably from being able to consult with them by telephone at every stage.

Many other people have contributed ideas, however. Some of them have contributed by being critical, like the people who pointed out that we had been using syntactic features without having articulated a theory of features and without making it clear what the relation was between te slash notation and the feature system. They were quite right. We make a serious effort below to provide a formalized theory of syntactic features, and of their distribution in syntactic representations. This part of our work should be of interest to any linguists who assume syntactic features in their theory of syntax---both transformationalists and lexical-functionalists, for instance. Our theory of features incorporates the ideas of several people. The Head Feature Convention, generally seen as implicit in the X-bar syntax that goes back to Zellig Harris, has been discussed, though not <u>eo nomine</u>, in works like Hellan (1977), Baker (1979), and Williams (1981). The Control Agreement Principle is due to Sag and Klein (1981), and is itself the progeny of a principle suggested originally by Ed Keenan (see Keenan 1974, 302). The need for a theory of foot features was first pointed out by Bill Ladusaw in 1979, and the specific proposals made here owe a large debt to some important unpublished work by Carl Pollard (see Pollard 1982, which has influenced us a lot). And the idea that "slash categories" are simply part of the feature system is due to John Bear (1981). Our position on features in general owes much to Carl Pollard and Henry Thompson, and the notion of "finite closure" that we adopt here for metarule application is entirely due to oral communications between Martin Kay, Susan Stucky, and Henry Thompson.

To specify more general debts, it is necessary for us to recapitulate the origin of the work presented here. Around 1978, a number of people began to explore ways of describing natural languages with a monostratal (single-level) syntactic theory tied to an explicit formal semantics of the sort proposed by Richard Montague. Gazdar's first paper of this kind was presented in the summer of 1978 at a workshop at the Linguistics Institute in Urbana-Champaign, and subsequently in Texas where Lauri Karttunen and Stanley Peters had been working along similar lines. During the same period, Rick Saenz and Ken Ross had, quite independently, been working out a phrase structure theory of grammar at the University of Massachusetts in Amherst. In January 1979, Gazdar circulated a working paper called "Constituent structures" in which various developments in the mathematical theory of phrase structure were discussed with a view to applying them to natural language description. In April 1979 he distributed a second working paper under the title "English as a context-free language" (henceforth ECFL). ECFL proposed a specific theory for handling unbounded dependencies of the wh-movement type, regarded as a major stumbling-block for nontransformational theories, in terms of context-free phrase structure grammar. This attracted a certain amount of attention. The ideas of ECFL were the basis for two of the papers presented at the conference on the nature of syntactic representation at Brown University in May 1979,.and ECFL has been fairly widely cited in the published literature, despite never having been intended for publication.

The ECFL theory of unbounded dependencies, together with discussion of its interaction with a cross-categorial account of coordination, did not appear in the journal literature until Unbounded dependencies and coordinate structure" (Gazdar 1981) was published. A general introduction to the topics broached in Constituent structures" and ECFL appears in "Phrase structure grammar" (Gazdar 1982 [completed in 1980]). The technical question of whether all natural languages are context-free languages is taken up in detail in Pullum and Gazdar (1982). Between them, these three works incorporate and render obsolete the two earlier papers, which should no longer be regarded as citable references.

The term "generalized phrase structure grammar" post-dates the papers just mentioned. It was adopted in the summer of 1980 during the Round Table Conference on Auxiliaries in Groningen, Holland. Ezmmon 3ach gave a paper at that conference about what he was calling "generalized categorial grammar." Bach's adjective was promptly borrowed by Gazdar, Klein, Pullum and Sag, in order to forestall a growing tendency for people to employ the unpleasantly alliterative (and attributionally inaccurate) name "Gazdar grammar" in connection with their work on phrase structure grammars. (It is a pleasure to acknowledge here that we often steal excellent ideas from Emmon. For instance, the idea of including exercises in a linguistics paper seems to originate with Bach (1981).)

Many people have contributed, directly or indirectly, to the emergence of the framework that we sketch here. Apart from people already mentioned above, they include at least Jan Anward, Mike Barlow, Bob Borsley, Ronnie Cann, Sandra Chung, Robin Cooper, Osten Dahl, David Dowty, Elisabet Engdahl, Dan Flickinger, Janet Fodor, Mark Gawron, Takao Gunji, Polly Jacobson, Aravind Joshi, Jim McCloskey, Joan Maling, Barbara Partee, Tom Wasow, and Annie Zaenen, plus dozens of other people who have discussed and criticized ideas with us, or have put forward ideas of their own that have found their way into current practice. To all of them we express our thanks.

And we also thank a number of financial sponsors. The funding for most of this work came from NSF grant BNS 81-02406 to Ivan Sag and Thomas Wasow at Stanford University. This grant permitted the authors to work together in California during July 1982 to complete this paper. Background support was provided in England by the Social Science Research Council (UK), and in California by the Syntax Research Center at the University of California, Santa Cruz. The Syntax Research Center kindly made Karen Wallace available to us as a full-time research assistant. Without Karen's patience and computer expertise we never would have got it done in time.

0. Introduction

In this paper, we introduce and motivate the main technical devices to be used in Generalized Phrase Structure Grammar (GPSG). A GPSG is a type of generative grammar that exploits several of the resources of transformational grammars but which, crucially, does not employ either transformations or coindexing devices, and which induces only a single level of structural description.

As their name suggests, GPSG's are essentially context-free phrase structure grammars (CF-PSG's). And indeed, GPSG's as we shall define them below are both weakly and strongly equivalent[1] to CF-PSG's. However, a GPSG differs from a simple CF-PSG of the type standardly characterized in computer science and linguistics texts[2] in at least two important respects:

(i) Syntactic categories are not taken to be simple monadic node labels, but rather have considerable internal structure.

(ii) The set of rules is not defined merely by extension (i.e. listing), but rather, in part, by means of a "grammar for the grammar", henceforth a metagrammar, which captures generalizations not expressed by the phrase structure rules themselves.

In the sections that follow, we will introduce these generalizations of CF-PSG and the notational devices that are associated with them. In each case, we will begin with the apparatus of CF-PSG, with which we assume the reader to be familiar, and go on to show how GPSG differs from it in the relevant respects.

1. Syntactic categories and syntactic features

In CF-PSG, as standardly presented, categories like "S", "NP", "VP", "N", etc. are monadic, which is to say that they have no internal structure and are not reducible to anything else.

Likewise, in pre-Jakobsonian phonology, phonemes like /g/, /k/, /p/, /s/, etc. were taken to be monadic and irreducible. In phonology, distinctive feature theory (Jakobson, Fant, and Halle 1952) replaced this view of the phoneme with one in which each phoneme was defined by reference to a set of features that might be specified positively or negatively. Under this conception, /g/, for example, comes to be understood as merely an abbreviation for, say,

(1) [^segment]
[-econsonantal]
[-sonorant]
'[-syllabic]
C +high]
[+back]
[-low]
[-round]
[-anterior]
[-coronal]
[-nasal]
[-continuant]

In syntax, Harris (1951) proposed that the relation between categories such as V and VP, and N and NP, was a systematic one that could be captured by breaking the monadic parts of speech labels into two components, namely a category type and a phrasal level. This insight was subsequently taken up in the "X-bar syntax" suggested by Chomsky (1970) and most fully developed in Jackendoff (1977)•

The potential of complex symbols in a CF-PSG was first recognized by Harman (1963), and Chomsky (1965) developed them further by introducing a notation for them which had obvious parallels in phonology* For reasons very nicely put by Halle (1969) with respect to phonology, there is an exact equivalence between generative systems that use complex symbols (matrices of features) and those that do not. The proof is trivial. Basically, only the way the symbols are interpreted is at issue. A nonterminal symbol [£(1), £(2.)* •••, £(n)], where each £(jj is some feature specification, can be treated as having internal structure to which statements in the grammar can refer to capture generalizations, or it can be regarded as a calligraphically ornate representation of an atomic symbol distinct from all other symbols. Moreover, anything done by a rule referring to, say, [£(2)] could also be done by a rule which referred to the complete list of all complex symbols in which [£(£)] appeared.

Illustration in more concrete terms will make this clearer. Throughout this paper we shall use the traditional categories Noun (N), Verb (V), Adjective (A), and Preposition/Postposition (P), but formally we shall treat them (following Chomsky 1970) as decomposable by means of a feature system that includes a feature +N which only N and A have, and a feature +V which only V and A have.[3] Thus nouns are nominal but not verbal; adjectives are nominal and verbal (capturing a traditional notion of "substantive"); verbs are verbal but not nominal; and prepositions are neither verbal nor nominal. In other words, the features group N, V, A, and P into natural classes as shown in (2).

(2)

	C+N]	C-N]
C+v]	A	7
C-v]	N	P

This enables ... prepositions/postpositions simply by writing [-V].

In phonology, complex symbols for phonological units are commonly taken to be sets of <feature name, feature value> pairs. However, syntacticians have, in general, assumed that complex symbols for syntactic categories have more internal structure. Thus Chomsky (1965: 171) implicitly assumes a hierarchical structure for lexical categories, and Bresnan (1976, 1977) defines categories as ordered pairs of an integer and a feature matrix. More recently, Stucky (1981) has proposed that languages with object agreement be analysed by reference to complex features involving ordered pairs of sets of agreement features. We shall be assuming that categories have a significant amount of internal hierarchical structure.

One particular elaboration of the theory of features seems to us to have great promise, permitting considerable elegance without excessive power. We refer to the notion that features may take other features as their coefficients, an idea that we have adopted from Bear (1981) and Pollard (1982).[4] We assume Pollard's condition requiring that no feature have itself as a descendant under the coefficient-of relation, so ensuring the finiteness of the set of categories.

Formally, we shall take features (and categories), to be directed acyclic graphs of a specific kind: the details are discussed in section 2. Here, however, we will introduce a reasonably transparent notation for these graph-theoretic objects.

(3) A _feature_ consists of a feature name optionally followed by one or more features or feature names. Features begin with a left bracket and end with a right bracket.

This definition does something that may seem a little unexpected in the light of earlier theories of features. It collapses the notions "feature" and "value of a feature". The values that get assigned to our features are themselves features. To put it graph-theoretically, there is no distinction between terminal and nonterminal node labels in the trees that constitute our features.

The initial feature name in a feature is the name of that feature. Let AGR (agreement), PER, 1PER, 2PER, 3PER (person), and NMB, 1NMB, 2NMB (number) be feature names. Then _all_ the following objects are possible features given (3):

(4)
a. [PER]
b. [PER 3PER]
c. [NMB 1NMB]
d. [AGR PER NMB]
e. [AGR [PER 2PER] NMB]
f. [AGR [PER 3PER] [NMB 2NMB]]

Here (4b) corresponds to "third person", (4c) to "singular", (4d) tells us only that agreement consists in person and number features (we assume gender to be semantic in English) but leaves these unspecified, (4e) tells us that we have something second person, but leaves number unspecified, and (4f) completely specifies agreement as "third person plural".[5] Notice that a feature of the form (4d) in effect specifies the syntax of the agreement feature -- we arrive at other coefficients for agreement simply by substituting in features for one or more of the corresponding feature names in (4d), making what is called an extension of (4d) (see below). (The notion "extension of a feature" is important, and will be discussed in detail below.) We can specify the syntax of features by using braces to enclose the range of permissible coefficients. Coefficients are always optional, so this is not marked explicitly. For example:

(5) a. [PER {1PER, 2PER, 3PER}]
b. [NMB {0NMB, 1NMB, 2NMB}]
c. [CASE {NOM, POSS}]

If we introduce features CAT and CAT', then we may define syntactic categories simply as a particular type of feature.

(6) Any feature whose feature name is CAT or CAT' is a syntactic category.

The internal syntax of CAT and CAT' is partially specified as follows for major categories:

(7) a. [CAT' CAT FOOT]
b. [CAT BAR HEAD]
c. [FOOT SLASH WH REFL]

The motivation for the distinction between CAT' and CAT, and discussion of the feature FOOT, is something we shall postpone until section 10, below. Here we focus on CAT.

BAR is a feature that provides for the phrasal level of the category in the sense of X-bar syntax. This feature either takes a coefficient from 1 to 3, or it takes the feature LEXICAL as its coefficient. Thus:

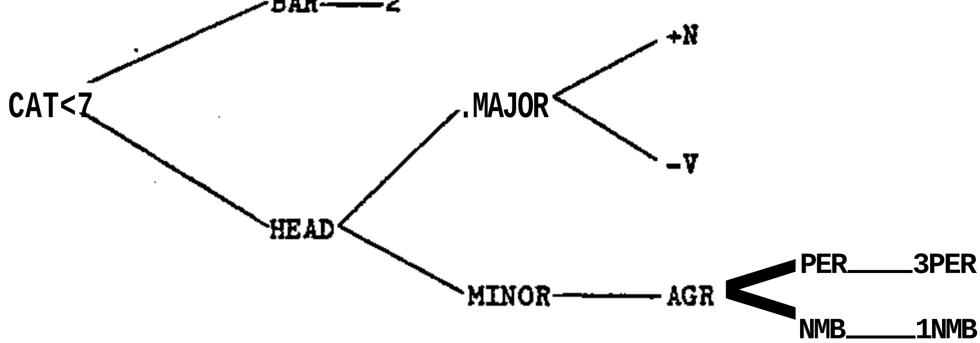
(8) [BAR {LEXICAL, 1, 2, 3}]

We will adopt the convention of writing X[F] or X[+F] if the category X contains the feature name F, and X[-F] if it does not. Thus, we will sometimes use [+LEXICAL] as an abbreviation for categories of the form [CAT [BAR [LEXICAL...]] ...], and [-LEXICAL] for categories of the form [CAT [BAR n]...] ($1 \leq n \leq 3$). The feature LEXICAL, unlike the other possible coefficients of BAR, itself takes coefficients. We will discuss the nature of these, and how they are assigned, in section 5, below.

HEAD is a feature which comprises that syntactic information held in common between phrases and their phrasal or lexical heads. It consists of MAJOR and MINOR features as shown below:[6]

(9) a. [HEAD MAJOR MINOR]
b. [MAJOR {+N, -N} {+V, -V}]
c. [MINOR AGR CASE ...]

We can now illustrate the theory of categories and features as so far elaborated by exhibiting a fairly fully specified category:



This represents the featural structure of the third person accusative[7] singular noun phrase category. Expressions of this category include *her*, *a book*, *the tall android*, etc.

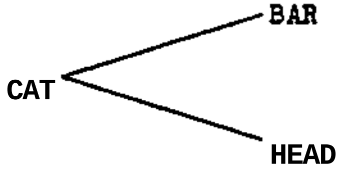
For almost all purposes during our exposition, we shall abbreviate such full category structures by using familiar conventions for referring to categories. For example, (10) could be written as N2[3PER, 1NMB], or even just as N2 or NP when the rest of the feature structure has no bearing on the point at hand. And we will sometimes use feature names to stand for partial functions that, given a category (or feature) as their argument, return the corresponding feature as their value. Thus NMB(NPi) stands for the number feature of NPi, and AGR(YPj) stands for the agreement feature of VPj, as the notation would lead one to expect. Some features can appear more than once in a category. Thus AGR can be a minor feature in VP, but it can also occur as the coefficient of REFL in a VP. In cases where ambiguity may arise, we will write, for example, AGR(MINOR(VPj)) or AGR(REFL(VPj)) to make the distinction clear.

It appears to be generally assumed that grammars employing features must adopt the following stipulation: •

(11) Only a fully specified category may label a node.

However, we shall not make this stipulation. In the present framework, fully specified categories have no privileged status, and something barely specified like (12)

(12)



is just as much a category as (10), above. How can (11) be abandoned without wild overgeneration? The answer lies in the fact that lexical items are fully specified for syntactic features and the feature instantiation principles ensure that, in the general case although not in every case, only fully specified categories appear as node labels. This point will become clearer as we proceed.

In section 2, we give formal definitions of the notions *extension* of a feature, *increment* of a feature, and *unification* of features. Since these notions play an important role in our theory of features, and since not every reader will want to digest the technical discussion below, we will try to give some intuitive informal content to these notions here.

An <u>extension</u> of a feature is like a superset. It contains everything in the original feature and may contain extra material as well. A proper extension, unsurprisingly, then is an extension which definitely does contain extra material. In the examples below, all of (13a)-(13f) are extensions of (13a); (13b), (13d), (13e) and (13f) are extensions of (13b) but (13c) is not; (13c), (13d) and (13f) are extensions of (13c), but (13e) is not; (13d) and (13f) are extensions of (13d) but (13e) is not; (13e) and (13f) are both extensions of (13e); and the only extension of (13f) is (13f) itself.

(13) a. [AGR]
b. [AGR PER]
c. [AGR [NMB 2NMB]]
d. [AGR PER [NMB 2NMB]]
e. [AGR [PER 1PER] NMB]
f. [AGR [PER 1PER] [NMB 2NMB]]
g. [NMB 2NMB]

The <u>unification</u> of a set of features is rather like set-theoretic union, except that it is only defined for sets of nondistinct features (the notion nondistinct is defined in section 2). What results from unification is a feature that brings together all the ingredients of the component features. Thus, in the examples above, (13d) is the unification of (13b) and (13c); (13f) is the unification of (13c) and (13e); and (13f) is also the unification of (13d) and (13e).

The <u>increment</u> F0 of a feature F1 with respect to a feature F2, where F1 is an extension of F2, is the smallest feature which, if unified with F2, gives one F1 as the result. Increments are features which have the same name as the features which they augment. Thus, in the examples above, (13c) is the increment of (13d) with respect to (13b); conversely, (13b) is the increment of (13d) with respect to (13c); and (13c) is the increment of (13f) with respect to (13e). Notice that (13g) is <u>not</u> the increment of (13f) with respect to (13e): it does not share the same feature name as (13e) and (13f) and thus cannot be the increment, despite the fact that, intuitively, (13f) is the result of substituting (13g) for NMB in (13e).

Note finally that since categories are features, we can, of course, talk about the increment of categories, the unification of categories, or of one category being an extension of another.

2. <u>A formal theory of features</u>

This section contains some definitions that are crucially referred to in later sections, but those readers who prefer to skip it on a first reading will find that much of the contents of later sections is intelligible without it.

We take features to be unordered trees with labeled vertices in the sense of Aho and Ullman (1972). We follow their terminology and assume their definitions in the following* Unordered trees are one type of directed acyclic graph. Some theorems that follow straightforwardly from the definitions we give in what follows will be stated, but proofs are left as exercises.

DEFINITION 1.

A feature £ is a ^l-tuple (At *&$ JL₉* JL) where A is a set of vertices, IL is a relation on A> H is a vertex in A, called the root, such that

(i) n has in-degree 0

(ii) all other vertices of £ have in-degree 1, and

(iii) every vertex in A is accessible from £,

and £ is a function from A into the set of feature names such that for every *&f* Ji in A*

(iv) if .a is accessible from *hf* then £Ca) *i* X(Jtl)> and

(v) if *SL* and £ are both direct descendants of some vertex £, then £(ii) *i* £(b)

Conditions (i), (ii) and (iii) guarantee that features are trees, condition (iv) says that no vertex can have a descendant that shares the same label, and condition (v) says that no vertex can have two daughters that share the same label.

THEOREM 1.

If the set of feature names is finite then the set of definable features is also finite.

This follows directly from definition 1, although proving it is not completely trivial.

DEFINITION 2.

The name of a feature £ is £(n).

In words, the name of a feature is the label on its root.

DEFINITION 3.

A feature F1 is an _extension_ of a feature F2 (written Ext(F1, F2)) if and only if

(i) f1(r1) = f2(r2), and

(ii) there is an injection h: A2 --> A1 such that for every a, b in A2,

(a) aR2b if and only if h(a)R1h(b), and

(b) f2(a) = f1(h(a))

This says, in effect, that F1 is an extension of F2 if one can anchor F2 to F1 at the root and then fit it into F1 on a vertex-by-vertex and label-by-label basis. Theorem 2, and definitions 4 and 5, below, follow an exact analogy with the superset relation that holds between sets.

THEOREM 2.

Every feature is an extension of itself.

DEFINITION 4.

Two features F1, F2 are _identical_ (written F1 = F2) if and only if

(i) Ext(F1, F2)

(ii) Ext(F2, F1)

DEFINITION 5.

A feature F1 is a _proper extension_ of a feature F2 if and only if

(i) Ext (F1, F2)

(ii) F1 ≠ F2

DEFINITION 6.

Two features F1, F2 are _nondistinct_ if and only if for some feature F0,

(i) Ext(F0, F1)

(ii) Ext(F0, F2)

This definition is just what one would expect: two features will be nondistinct provided that they have no non-identical coefficients. Thi entails and is entailed by the existence of a common extension. Theore 3 then follows directly from this definition and theorem 2.

THEOREM 3.

If F1 is an extension of F2 then F1 and F2 are nondistinct.

Definition 7 characterizes the smallest feature having some property.

DEFINITION 7.

A feature F1 minimally satisfies a predicate P if and only if

(i) P(F1) is true, and

(ii) there is no feature F0 such that

(a) P(F0) is true, and

(b) F1 is a proper extension of F0

We can now define the notion "unifies":

DEFINITION 8.

A feature F0 unifies features F1, ..., Fn ($1 \leq n$) (written Uni(F0, F1, ..., Fn)) if and only if F0 minimally satisfies the property $\lambda F[\text{Ext}(F, F1) \wedge \ldots \wedge \text{Ext}(F, Fn)]$.

THEOREM 4.

If, for some Fi, $0 \leq i \leq n$, Uni(F0, F1, ..., Fn) then for all i, j $1 \leq i, j \leq n$, Fi and Fj are nondistinct.

THEOREM 5.

If, for some F', F", Fi, $1 \leq i \leq n$, Uni(F', F1, ..., Fn) and Uni(F", F1, ..., Fn), then F' = F".

THEOREM 6

For all F, Uni(F, F).

Definition.8 defines the unification of a set of features as the smallest feature which is an extension of each of them. There may be n< such feature, of course, in which case the finl statement will be false for any FO. However, if there is a unification, then theorem 4 tells us, unsurprisingly, that all the features unified are nondistinct. Theorem 5 tells us that features that can be unified have a unique unification. And theorem 6 just notes that every feature is the unification (i.e. minimal extension) of itself.

DEFINITION 9-

A feature FO increments a feature F1 with respect to a feature F2 (written Inc(F0, F1, F2)) if and only if FO minimally satisfies th< property *F[Uni(F1, F, F2)AExt(F1, F2)].

THEOREM 7.

If, for some F', F", F1, F2, Inc(F*, F1, F2) and Inc(F«, F1# F2), then F^f = F*.

THEOREM 8.

Fcr all features F, there exist3 a feature F^f such that

(i) Inc(F^f, F, F)

(ii) Inc(F, F, F^f)

(iii) F' * < {r*}, $> r', { <r<, f(r)> } >

Definition 9 defines the increment of a pair of features F1, F2, where F1 is an extension of F2, to be the smallest feature which, taken together with F2, yields F1 as the unification. That is, the smallest feature you need to add to F2 in order to get F1. If F1 is not an extension of F2, then there will not be an increment, and the Inc statement will be false. However, if there is an increment, then theorem 7 tells us that it is unique. And theorem 8 notes that the increment of a feature with respect to itself is a feature that consist just of its root. And the increment of a feature with respect to a feature that consists just of its root is the feature itself.

Exercises (section 2)

1. Let A, B, and C be feature names and 1-5 be vertices. Which, if any, of the following objects are features?

a. < {1, 2, 3}, { <1, 2>, <1, 3>, <3, 2> }, 1,

{ <1, A>, <2, B>, <3, C> } >

b. < {1, 2, 3}, { <1, 2>, <1, 3> }, 1,

{ <1, A>, <2, B>, <3, C>, <1, C> } >

c. < {1, 2, 3}, { <1, 2>, <1, 3> }, 2,

{ <1, A>, <2, B>, <3, C> } >

d. < {1, 2, 3}, { <2, 3>, <3, 1> }, 2,

{ < 2, A>, <3, B>, <1, C> } >

e. < {1, 2, 3, 4, 5}, { <1, 2>, <1, 3>, <2, 4>, <3, 5> },

1, { <1, A>, <2, B>, <3, C>, <4, C>, <5, A> } >

f. < {1, 2, 3, 4, 5}, { <1, 2>, <1, 3>, <2, 4>, <3, 5> },

1, { <1, A>, <2, B>, <3, C>, <4, C>, <5, B> } >

2. Which two of the following three features are identical?

a. < {1, 2}, { <2, 1> }, 2, { <1, A>, <2, B> } >

b. < {1, 2}, { <2, 1> }, 2, { <1, B>, <2, A> } >

c. < {1, 2}, { <1, 2> }, 1, { <1, A>, <2, B> } >

3. Prove theorem 8.

4. Prove theorem 1.

5. If there are N feature names, how many definable features are ther
[The answer to this question is of no intrinsic interest, but if you c
answer it correctly, then you clearly understand how features work.]

3. Marking conventions

If some syntactic category is tensed, then that category is not a preposition. Nor is it a prepositional phrase. Likewise, if some consonant is strident, then it is not nasal. In English, typical occurrences of NP's are accusative. And typically, consonants are not nasal. Restrictions of this sort are expressed in generative phonolog by means of what Chomsky and Halle (1968, chapter 9) call marking conventions. Chomsky and Halle employ two kinds of marking convention One kind (1968, 404-407) is illustrated in (14).

[-nasal] —> [-son]
[•high] —> [-low]
[•low] —> [-high]
[•ant] —> [•cor]
[-cor] —> [-lateral]

These rules can be seen to constitute part of the definition of possible phonological segment. Each has the potential of reducing the space of possible segments by up to 25$. We will refer to absolute conditions of the type shown in (14) as "feature cooccurrence restrictions" (FCR^{f}s, hereafter).

The other kind of phonological marking convention (Chomsky and Halle 1968, 405-407) is illustrated in (15).

(15) Di high] —> [•high]
Di nasal] —> [-nasal]
Ui low] —> [-low]
La ant] --> [+ant]
Qi cor] --> [•cor]
Ql cont] —> [•cont]

These rules tell us what the default coefficient is for the feature in question. If nothing is said about it, then we are to assume the coefficient specified on the right of the arrow. We will refer to markedness conventions of this kind as "feature coefficient defaults" (FCD^{f}s$_f$ hereafter).

Our theory of syntactic categories will employ both FCR^{f}s and FCD's. However, the two kinds of marking convention have a rather different formal status and we will use distinct notations to reflect this. FCR^{f}s, for us, constitute part of the definition of syntactic category. Some FCR^{f}s will be universal and thus be part of the definition of "possible natural language syntactic category", and some will be language particular, and thus be part of the definition of, for example, "possible syntactic category in English". Consider an analogy with phonotactics. Particular combinations of syntactic features may constitute a possible category for a language, even though the language never happens to employ the category, just as /blik/ is a possible English word, though not an actual one. Likewise, particular combinations of syntactic features may not constitute a possible category for a given language, even though the same combination would be possible for another language. Thus /bnip/ is not a possible English word, despite the fact that nothing in "universal phonotactics" prohibits it from being a word in some other language.

We will state FCR's as material implications holding between Boolean combinations of features. Here, by way of example, is one such FCR.C8]

(16) [+FINITE] Z3 C-M, +V]

This just says that if a category is tensed, then it is a verb (or some phrasal projection of a verb). From this it follows, by modus tollens, that nouns, noun phrases, adjectives, adjective phrases, etc., cannot have tense. Something of the form of, for example, [+N, -V, FINITE] is

Another example of an FCR, and a language particular one in this case, is (!?)•

(17) C+IHV] 3 [+AUX, +FINITE]

This FCR says that a category which carries the feature INV will also carry the features ADX and FINITE. INV is a feature that appears on sentences which include a subject but begin with a verb, and also appears on that sentence initial verb (see Gazdar, Pullum and Sag (1982) for some discussion). This FCH has as a consequence that such a verb will always be a tensed auxiliary verb.

Let us turn our attention now to the second type of marking convention, namely feature coefficient defaults. These form an important part of the link between the highly schematic rules listed in the grammar, and the very fully specified rules necessary to induce structural descriptions. Consider the set of rules which introduce NP^{f}s in English. Most of these NP^{f}s will require accusative case ([-CASE], i.e. the absence of the CASE feature, in our feature system, of. Horrocks (1982)). But we do not want the grammar to stipulate this in instance after instance. Accusative is the unmarked case in English. So what we need to be able to do is stipulate the occurrences of the marked cases (nominative and possessive), and allow general principles of feature instantiation to assign accusative to those NP^{f}s whose case is unspecified. The exact nature of the feature instantiation principles is something we will postpone to section 11• All we need to point out here is that feature instantiation makes reference to FCD^{f}s, and that these define the default feature coefficients for daughter categories mentioned in rules.

It is certainly to be expected that FCD^{f}s, like FCH's, can be usefully divided into those which are universal and those which are parochial. However, that is not a topic that we shall pursue here. We shall refer, however, to a distinction, defined in universal terms, between FCD^{f}s which apply to phrasal categories and FCD^{f}s which apply to lexical categories.

Consider the two rules shown below.

(18) a. VP —> V NP
b. NP ~> Det N

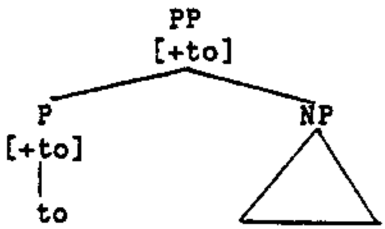
Suppose that FCD^{f}s fail to distinguish between lexical and phrasal categories, and that we have an FCD saying that the default value for case is "accusative".[10] Then the NP introduced in (18a) will be instantiated as an accusative, just as we want. But the N in (18b) will also be instantiated as an accusative. And this we do not want since it will entail that we add a rule to the grammar which is just like (18b), except that it marks the noun as nominative. Introducing such a rule (and, of course, every NP expansion rule will now need a nominative counterpart) loses an obvious generalization about English, namely that subject and non-subject noun phrases have identical internal structure. However, if we distinguish between lexical FCD^{f}s and phrasal FCD^{f}s, then we can say that the phrasal FCD for case is accusative, but that the lexical FCD for case is free (i.e. there is no default coefficient for

(19)	Feature name	Lexical FCD	Phrasal FCD
	CASE	free	-CASE

JU Terminal symbol features

A further elaboration of the feature system that we now introduce involves features named after specific terminal symbols, i.e. English words. For a small proper subset of the lexicon we wish to allow features whose names will simply be the ordinary spellings of the words in question. A very simple example of this device, and of what we might use it for, concerns indirect objects. We need to be able to guarantee that where a verb like hand occurs, a prepositional phrase with £& as its preposition will be present in the VP (because while She handed it ifi *m§L* is grammatical, we want to block *SllS. handed i£, »She handed JLt near me, and so on). The exact way in which it will be done is shown below; but clearly we need to be able to mention in a rule of grammar that a given constituent is a prepositional phrase whose preposition is i&. We introduce a feature [+£&] which can appear in the feature-sets of [-N, -V] categories and enter i& as being a lexical item of category P[+to].[11] Now any rule that has to guarantee the presence of the preposition Jta in a prepositional phrase can simply introduce PP[+to]. The Head Feature Convention (see section 8, below) will then ensure that the P in this PP is Jta in the manner illustrated by the tree in (20)•

(20)



All terminal symbol features can be assumed, by convention, to have negatively specified lexical and phrasal FCD's. They will appear, then, only when they are specifically introduced by rules.

Terminal symbol features seem only to be needed for elements that are present by virtue of the type of construction involved rather than by virtue of their intrinsic lexical properties. We use them, for example, to introduce prepositions whose function is to mark case and whose semantic role is null. Thus, .to. jag. will contribute to the determination of meaning in exactly the same way that jag. would contribute; there is no nontrivial semantic difference between My true love gave £ partridge in .& pear tree to me and Hy. true love gave me £ partridge in *A pe^r* tree. We also use terminal symbol features to introduce certain complementizing particles such as than and gjs., and to introduce the Boolean connective words .and and £jc in coordinate constructions. Let us therefore restrict terminal symbol features to the names of words having interpretations as logical constants. In the present context, a logical constant is a word that always denotes the same thing no matter what facts about the world are assumed.[12] One way a word can have this property is for it to denote an identity function: the "meaningless" case-marking prepositions, for example, denote an identity function on NP meanings. Another way is for the word to denote a Boolean operator such as negation or conjunction. Thus and denotes what it denotes quite independently of what things exist in a given world or what properties they have.

Our restriction leaves us with a rather small class of items that are candidates for having associated terminal symbol features. Outside this class fall all the nouns[13], all the adjectives, most of the verbs (certain auxiliaries, e.g. do, be, and to are candidates), and all the prepositions under their meaningful adverbial interpretations. However, their "meaningless" homonyms that function as case-indicators, verbal particles, complementizers, etc., can appear as terminal symbol features. Our restriction on the availability of terminal symbol features provides a sound basis for the usage of traditional grammarians in talking about "function words" or "grammatical items". Note finally that we are not claiming that every word in a language that corresponds to a logical constant will, *ipso facto*, have an associated terminal symbol feature in the syntax.

5. Lexical subcategorization

Any grammar for English has to provide for the unacceptability of such examples as the following:

(21) *I devoured to him that grass is green.

One approach would be to let the syntax generate strings in which (for example) verbs occur with the wrong number and type of other constituents in the V1, and use the mechanisms that associate syntactic structures with meanings to eliminate them.[14] In other words, treat strings like (21) as grammatical but not semantically interpretable. This approach, which we might call "semantic filtering", is not appropriate to the domain under consideration. The reason is that there is fairly clear evidence that the meaning of a verb does not determine its subcategorization. Consider the following sets of data, in which the first two sentences in each set illustrate the synonymy of two verbs, and the second two examples demonstrate a dissimilarity between them as regards subcategorization.

(22) a. The beast ate the meat (ravenously).
b. The beast devoured the meat.
c. The beast ate (ravenously).
d. *The beast devoured.

(23) a. The ground sometimes shakes under your feet.
b. The ground sometimes quakes under your feet.
c. What is shaking the ground?
d. *What is quaking the ground?

(24) a. He gave this to me.
b. He donated this to me.
c. He gave me this.
d. *He donated me this.[15]

(25) a. AT, xs xiKexy tnat- Axex WZJ.JL ieave#
b. It is probable that Alex will leave.
c. Alex is likely to leave.
d. *Alex is probable to leave.

(26) a. Aren'[f]t you even going to try to solve it?
b. Aren'[f]t you even going to attempt to solve it?
c. Aren'[f]t you even going to try?
d. •Aren't you even going to attempt?

Further examples of the same sort could be given. What they suggest is that there are restrictions on contexts of occurrence for lexical items which the grammar must specify, and which cannot be reduced to facts about meaning.

Consider the following phrase structure rule:

(27) VP —> V (NP) (NP) (PP) (PP) (S)

Our problem is, for example, to ensure that V expands as devour only when V is immediately adjacent to NP, and not, say, PP or S as (27) would permit. Viewed in this light, one obvious strategy for coping with the facts of subcategorization is to enrich the theory of grammar by introducing context-sensitive phrase structure rules and using rules of this type for lexical insertion, along the lines exemplified in (23)

(28) 7 ~> bring/___NP
V ~> persuade/___NP S
V —> decide/___PP
7 —> grow/___AP
7 —> save/___NP PP
7 —> trade/___HP PP PP

However, an approach along these lines, though initially plausible, run into numerous difficulties. One is that these rules include a mass of statements which redundantly repeat things that the phrase structure rules have already said, e.g. that NP PP sequences are found in English 7P'[f]s but PP NP sequences are not.[16] This point is made and illustrate* in detail by Heny (1979)• Another problem is that the rule for expanding 7P will have to introduce all the different sorts of constituent that verbs can demand as their complements in some instances, and since no verbs take .all of these (for example, devour takes an NP but cannot also have an indirect object or a subordinate clause, witness *£ devoured the meat tq him that grass is green) there also have to be numerous negative specifications associated with the rules in (23).[17]

In response to these problems, we adopt an alternative approach to subcategorization, one which only employs context-free phrase structure rules. This approach works as follows. Each phrase structure rule is associated with an identifying integer. Suppose there is a rule of grammar with the identifying integer ±. Rule ± introduces a lexical category £, and only a proper subset of lexical items of category £ can appear under H in the environment created by the syntactic component of rule JL# Let ± be a feature on £, and interpret rule ± as shewn in (29) to be an abbreviation-by-convention of (30):

(29) <i; K ⟶ ... C ... >

(30) <i; K ⟶ ... C[i]... >

Of course, given the theory of categories outlined in the previous section, (30) is not fully explicit. *C[i]* is to be understood here as [CAT [BAR [LEXICAL *i*]]...], in other words, the rule number is assigned, by convention, to be the value of the feature LEXICAL. Note that the use of complex symbols enables the analysis to avoid the charge that totally distinct categories have to be postulated for verbs of different subcategorization types so that one loses generalizations about verbs (e.g. that they all take tense). These generalizations are not lost since all verbs have at least two features in common (namely [+V, -N]) and it is this fact which accounts for the generalizations that can be made.

Whenever a syntactic rule mentions a lexical category (that is, N, V, A, P, or a minor category to which specific words can belong), the rule number appears as one of the features on the lexical category.[18] So, suppose we have a rule like (31) in the grammar:

(31) <4; VP ⟶ V >

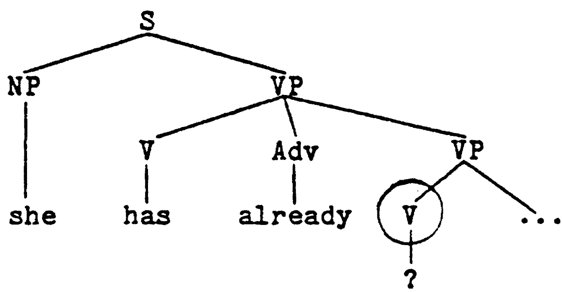
The V is in fact V[4] (which is itself an abbreviation for V[BAR [LEXICAL 4]], of course) by the convention just set up. Now we can simply say that *disappear* belongs to V[4] (i.e. there is a rule in the lexicon which permits V[4] to immediately dominate the terminal symbol *disappear*) while *devour* does not belong to V[4]. Again, suppose that (32) is one of the rules of the grammar.

(32) <5; VP ⟶ V NP >

Now we can ensure that V[5] will immediately dominate *dominate*, *abandon*, *enlighten*, *castigate*, *slap*, and so on, but not *disappear*, *elapse*, *expire*, *faint*, and so on.

The proposal just made has one very specific and obvious consequence: it entails that *only the items introduced by the same rule as a given lexical category C* can be relevant to the question of whether *C's* subcategorization environment is met. It cannot affect the insertability of a given verb that there might be an adverb adjacent to it in the tree but belonging to a higher level of constituent structure, for example. In the tree shown in (33), the adverb *already* cannot be made relevant to the question of which verbs can appear under the ringed V node. Nor can the other verb *has*; and nor can the subject *she*.

(33)



This is not something that is entailed by context-sensitive accounts of to note that Chomsky chooses to stipulate the very constraint that follows as a theorem from the system outlined above (see Chomsky 1965: 96, 99).

There are other advantages of our context-free system for subcategorization besides the desirable restriction just noted. One that we shall comment on briefly here concerns coordination. Consider sentence (34), involving the verb hand, which requires both direct and indirect objects.

(34) We have handed or sent a copy of this letter to every student in the school.

In this example, hand is not in the context defined by the contextual feature specification [+ ___ NP PP]. The problems caused by the fact that coordination of verbs can destroy crucial adjacency relationships in this way are considerable, but have not to our knowledge been addressed by proponents of context-sensitive lexical insertion. In our system, on the other hand, there is no reference to linear adjacency in the conditions on insertion for the terminal symbol hand. Hand belongs to a category, let us say V[i], associated with a rule numbered i which introduces the category V together with an accompanying NP and a PP[+to]. The V category has the feature [i] by convention. By the general schema for coordination discussed in Gazdar (1981, 1982), any category α can dominate one or more α's, a connective word (and or or), and one further α, so a V[i] can dominate two V[i]'s separated by or. Hence a tree for (34) can be admitted. The interaction of coordination and strict subcategorization is successfully predicted by the system assumed here, in fact, whereas it has never been satisfactorily treated within the context of Chomsky's (1965) proposals.

6. Immediate dominance and linear precedence

A phrase structure rule of the familiar sort specifies two distinct relations: (i) immediate dominance relations, and (ii) linear precedence relations among sisters. Consider, for example, the rule shown in (35):

(35) A --> B C D

This rule tells us that a node labelled A may immediately dominate nodes labelled B, C, and D, when the node labelled B linearly precedes the one labelled C, and the latter linearly precedes the one labelled D. Conflating the statement of these two kinds of relation in one rule format could, in principle, lead to the loss of certain kinds of generalization. Consider, for example, the grammar shown in (36):

(36) A --> B C D   C --> A B D

B --> A C D   D --> A B C

Inspection shows that a generalization can be made about the set of trees admitted by this grammar, namely that sister constituents always appear in an order that happens to correspond to the order of the letters A, B, C, D in the English alphabet. This generalization is not expressed by the grammar shown in (36).

Suppose we adopt a mode of characterizing a phrase structure grammar (PSG) that factors out the two relations, and states them separately. For immediate dominance, we can use the format shown in (37), which we shall call an immediate dominance (ID) statement.[19]

(37) A --> B, C, D

This statement allows the induced grammar to contain a set of PS rules which permit an A to immediately and exhaustively dominate a B, a C, and a D. However, it does not in itself define a set of PS rules since it does not say anything about the linear order in which B, C, and D must occur under A.

For linear precedence, we introduce the asymmetric, transitive relation "<", where "A < B" is to be read as "if A and B both appear on the righthand side of a PS rule then A precedes B." We shall call a statement like "A < B" a linear precedence (LP) statement. We will use "A < B < C" as an abbreviation for "A < B and B < C", and so on in an obvious manner.

A grammar is now defined as a pair consisting of (i) a set of ID statements, and (ii) a set of LP statements. The PSG induced by such a grammar definition consists of all those PS rules each of which is consistent with some ID statement and every LP statement. Consider, for example, grammar (38):

(38) i. A --> B, C, D
B --> A, C, D
C --> A, B, D
D --> A, B, C

ii. A < B < C < D

Taken together, (38i) and (38ii) are extensionally equivalent to grammar (36); (38) simply defines the set of rules shown in (36). But (38) does what (36) does not: it expresses the generalization about sister constituent order.

We should make it clear at this point that treating immediate dominance and linear precedence separately in a generative grammar is an idea with a long history. Many linguists have suggested something of this sort.[20] But in fact none of the previous work does exactly what we are proposing here. We do not propose to have any levels of syntactic structure that are nonlinear, or to have phrase structure rules that have sets rather than strings as their right hand sides. Rather, we wish to separate dominance and precedence in the metagrammar only. Our syntactic representations always display both dominance and precedence relations simultaneously, and so do phrase structure rules. Only the form in which the phrase structure grammar is stated factors them out. Accordingly, we do not have rules for linearizing structures (the sort of rules that Chomsky (1965, 124) felt would constitute a redundant component of the grammar). We linearize rules, not structures, capturing generalizations by stating constituent order for whole blocks of rules at a time rather than one at a time as in standard statements of phrase structure grammars.

Let us consider a toy grammar of the kind familiar from elementary introductions to CF-PSG:

(39)

S --> NP VP
S --> AUX NP VP
VP --> AUX VP
VP --> V VP
VP --> V NP
VP --> V NP VP

This grammar fails to express at least two generalizations: (a) AUX and V are constituent-initial, and (b) NP's precede VP's. An extensionally equivalent grammar in the format just introduced, which we shall call Immediate Dominance/Linear Precedence (ID/LP) format, is that shown in (40):

(40)

i.		ii.
S --> NP, VP		AUX < NP
S --> AUX, NP, VP		V < NP
VP --> AUX, VP		NP < VP
VP --> V, VP		
VP --> V, NP		
VP --> V, NP, VP		

This grammar expresses the two generalizations noted above, but at the cost, it seems, of additional statements.[21] This verbosity is only apparent, however, since (40i) contains much less information than (39). The greater economy of the ID/LP format becomes more obvious when we consider a toy grammar for the VP of a language that has much freer word order than the English-like language presupposed by the grammar in (40). Makua (Stucky 1981) is just such a language. We exhibit in (41) almost all the rules necessary to assign correct constituent structure to Makua VP's:[22]

(41)

VP --> V	VP --> V S NP
VP --> V S	VP --> NP V S
VP --> V NP	VP --> NP V PP
VP --> NP V	VP --> PP V NP
VP --> NP V NP	VP --> PP NP V
VP --> NP NP V	VP --> NP PP V
VP --> V NP NP	VP --> V PP NP
VP --> V NP S	VP --> V NP PP

This grammar can be restated very compactly in ID/LP format:

(42)

i.	ii.
VP --> V	V < S
VP --> V, NP	
VP --> V, S	
VP --> V, NP, NP	
VP --> V, NP, PP	
VP --> V, NP, S	

Now observe that this is not just a new format for writing PSG's. There are some PSG's that are not expressible in ID/LP format. Consider, for example, the grammars shown in (43) and (44):

```
(43)    S  -> NP  VP
        S  -> AUX  NP  VP
        VP -> ADX  VP
        VP -> V  VP
        VP -> V  NP
        VP -> V  VP  NP
```

```
(44)    VP -> V               VP -> S  V  NP
        VP --> V  S           VP -> NP  V  S
        VP -> V  NP           VP -> NP  V  PP
        VP -> NP  V           VP -> PP  V  NP
        VP -> NP V  NP        VP -> PP  NP  V
        VP -> NP  NP  V       VP -> NP  PP  V
        VP -> V  NP  NP       VP -> V  PP  NP
        VP -> V  NP  S        VP -> V  NP  PP
```

These two grammars are exactly as complex as (39) and (41), respectively, and yet neither can be expressed in ID/LP format.

There is a formal property that distinguishes grammars (39) and (41) from (43) and (44). It is that in (39) and (41) the set of expansions for given category is closed under a partial ordering that is constant for the expansion of all categories. We will refer to this property as the Exhaustive Constant Partial Ordering (ECPO) property. A CF-PSG can be put into ID/LP format if and only if it has the ECPO property. Grammars (43) and (44) do not have the property. Thus (43) contains rules which exhibit NP VP order as well as VP NP order, but it restricts the former to expansions of S and the latter to expansions of VP« And (44) allows an S V order when both precede NP, but not otherwise.[23]

Exhaustive constant partial ordering is a very abstract property of grammars. As the example above suggest, many plausible-looking sets of CF-PSG rules do not have the ECPO property. Indeed, it is a statistically unexpected property* If we consider the set of possible grammars defined on the same vocabulary as (44), with the same number of rules, and the same upper bound on the size of rules, then only a tiny proportion of this set will have the ECPO property (because there are many more orderings of a set of categories that are specific to certain dominating categories than orderings that are constant across categories). It would therefore be very interesting if ECPO turned out to be a language universal. Clearly, anyone adopting ID/LP as their format for stating grammars, as we shall, is making the rather strong claim that ECPO is a universal.

In a language particular context, also, ID/LP format allows us to capture significant generalizations. Thus, in English, a lexical category always precedes nonlexical (phrasal) sisters. This fact can be expressed with a single LP statement:

(45)   [+LEXICAL] < [-LEXICAL]

From this it follows that verbs will be VP-initial, that auxiliary verbs will be sentence-initial, that nouns will precede their complements, that English has prepositions rather than postpositions, etc.

From now on we shall often refer to ID and LP statements as rules of grammar in order to simplify our exposition of other points. Although strictly ID and LP statements are clauses in the definition of a grammar--i.e. rules of the metagrammar--there should be no resultant confusion.

We conclude this section on the format of the basic rules in our grammar by generalizing the notion "extension of a feature (category)" as defined in section 2, above, to provide us with a definition of the notion "extension of an ID rule".

(46) A rule A, where A is $\alpha 0 \rightarrow \alpha 1, \ldots, \alpha n$ ($1 \leq n$), is an extension of a rule B if and only if B can be written as $\beta 0 \rightarrow \beta 1, \ldots, \beta n$, and, for all i, $0 \leq i \leq n$, αi is an extension of βi.

In words, one rule is an extension of another provided that each category in the former is an extension of its counterpart in the latter. We will have occasion to invoke this notion in section 11, below.

Exercises (section 6)

1. Let grammar a be rule a, grammar b be rules a and b, grammar c be rules a, b, and c, and so on.

a. A --> B C
b. A --> C B
c. B --> B C
d. B --> A C
e. B --> C B
f. C --> C A

Say which, if any, of grammars a-f have the ECPO property.

2. Formulate a comprehensive set of LP statements for the nonlexical categories of English syntax.

3. Prove that all CF-PSG's have subgrammars and supergrammars with the ECPO properties.

4. Two classes of grammars are weakly equivalent if and only if every grammar in each class induces a language (set of strings) identical to a language induced by some grammar in the other class. Prove that the class of ECPO CF-PSG's and the class of CF-PSG's are weakly equivalent.

5. Give a precise formal definition of the ECPO property:

"A CF-PSG G, where G = <N, T, S, P>, has the ECPO property if and only if ..."

6. Give a precise definition of the notion of strong equivalence between classes of grammars that induce ordered trees with labeled nodes (see the definition of equal labeled graph given by Aho and Ullman

(1972: 38) for a possible starting point). Under your definition, is the class of ID/LP grammars strongly equivalent to the class of CF-PSG's? Comment on the following CF-PSG in the light of your result.

S --> A B C
C --> B A
A --> a
B --> b

7. Give some hypothetical examples, as close to real languages as possible, of phenomena that would suggest that ID/LP was *not* the appropriate format for characterizing natural language grammars.

8. How many distinct CF-PSG's (i.e., CF-PSG's that are not alphabetic variants of each other) are definable on a set of nonterminal symbols with cardinality n, a set of terminal symbols with cardinality t, an upper bound u on the number of symbols permitted on the right hand side of a rule, an upper bound r on the number of rules, and a fixed start symbol S? How many have the ECPO property? [Note that with arbitrary upper bounds set on the components of grammars in this way, the set of grammars becomes finite. We are suggesting the imposition of such bounds purely in order to set the mathematical exercise above. It has been suggested that natural conditions on grammars yield, for some theories, the result that the actual set of grammars defined by linguistic theory is finite. We believe that such suggestions are totally groundless; cf. Pullum (1982b) for discussion. It also seems to us that it would be a reductio ad absurdum for a framework if it could be shown that it defined only a finite class of grammars and thus could describe only a finite number of languages.]

7. Metarules

In the previous section, we have outlined one way of indirectly characterizing a set of CF-PSG rules, thereby making it possible to capture generalizations that are not expressed in the set of CF-PSG rules itself. In this section, we examine another.

Consider the set of CF-PSG rules in (47):

(47) a. <5; VP → V NP >
b. <6; VP → V NP PP[to] >

These are responsible for English verb phrases such as those shown in (48a) and (48b), respectively.

(48) a. devoured the carcass
b. handed the book to Sandy

Suppose now that we wish to extend our grammar to permit verb phrases such as those in (49):

(49) a.i. eaten
ii. eaten by Felix

b.i. given to Sandy
ii. given to Sandy by Lee

An obvious way to do this is to add two more rules to our grammar, namely those in (50):

(50) a. <15; VP ⟶ V[PAS] (PP[by]) >
b. <16; VP ⟶ V[PAS] PP[to] (PP[by]) >

But if we do this, then our grammar will fail to express a number of significant generalizations. Notice first that the membership of the category V[15] will be identical to the membership of category V[5], modulo morphological form. Likewise, V[16] and V[6]. Furthermore, the topology of the rules in (50) stands in a systematic relation to that of those in (47). This relation is not expressed if all four rules are merely listed.

To cope with this situation, we introduce the notion of a metarule. A metarule is a function from (sets of) rules to (sets of) rules and is part of the metagrammar.[24] The idea is that by employing metarules we will only need to list rules like those in (47)--we can get those in (50), not by listing, but by the application of a metarule to (47a) and (47b). Such a metarule will look something like this:[25]

(51) VP --> V NP W

⇓

VP --> V[PAS] W (PP[by])

What this says is: for every rule in the grammar which permits VP to dominate V followed by NP possibly followed by other stuff, there is also to be a rule in the grammar which permits a VP to dominate passive V, followed by the other material, if any, from the original rule, followed optionally by a PP carrying the feature [by]. By convention, the rule number of the output rule is set identical to that of the input rule, and any features mentioned on categories in the input rule are retained on those categories in the output rule, unless the metarule itself changes them.

Thus (51) will apply to (47a) and (47b) to give us (52a) and (52b), respectively.

(52) a. <5; VP ⟶ V[PAS] (PP[by]) >
b. <6; VP ⟶ V[PAS] PP[to] (PP[by]) >

Notice how our conventions ensure the identity of rule numbers in (47a) and (52a), and (47b) and (52b). Given our use of rule numbers as lexical subcategorization features, this identity in turn ensures that exactly the same class of lexical items will be permissible in the contexts defined by rules (47a) and (52a). Likewise in the case of rules (47b) and (52b). And the generalizations about passive marking on the verb, absence of the direct object, and optional presence of a by phrase, all follow from the form of the metarule in (51).

However, (51) is not without problems. Consider the following grammatical verb phrase.

(53) given by Lee to Sandy.

Rule (52b) will not permit this to be generated. It seems that the grammar also needs to contain the rule shown in (54).

(54) VP —» V[PAS] PPCby] PP[to]

We could just add the rule to the grammar, of course, but then it becomes merely an accident that it allows a V[PAS] to precede a PP[by], and just a coincidence that the set of lexical items associated with its rule number happen to be identical to those introduced by rule (47b).

Furthermore, (51) stipulates the linear order of categories in the output rule. Thus, it offers no general explanation for why that order is the way it is, rather than, say, being the mirror image. We could just as well have written the metarule as:

(55) VP —» V NP W

⇓

VP —> (PP[by]) W V[PAS]

It is not clear that there is any solution to this problem as long as we maintain that metarules map (sets of) CF-PSG rules into (sets of) CF-PSG rules.

However, suppose that metarules are one degree more abstract, and map, instead, (sets of) ID statements into (sets of) ID statements. Then both problems noted above are immediately solved. Here is the passive metarule reformulated as a function from (sets of) ID statements to (sets of) ID statements:

(56) VP —> V, W, NP

⇓

VP —» V[PAS], W, (PPCby])

Since this applies to ID statements, not CF-PSG rules, we need to reconstruct our example grammar in (47), as follows:

(57) i. ID rules

a. <5; VP —> V, NP >

b. <6; VP —> V, NP, PPCto] >

ii. LP rules

V < NP < PP

Metarule (52) will apply to this grammar to give us the (expanded) grammar in (58).

(58) i. <5; VP —> V_f NP >
<5; VP —> VCPAS], (PPCby]) >
<6; VP —> V, NP, PP[to] >
<6; VP --> VCPAS], PPCto], (PPCby]) >

ii. V < NP < PP

At first sight, it may appear that (57i.b) fails to meet the structural analysis required by (51) since the latter is of the form V, W_f NP* But appearances here are deceptive: (57i.b) is exactly the j£ffi£ rule .as. (59)t below (See footnote 19_f above).

(59) <6; VP ~> V, PPCto], NP >

The variable W ranges over zero or more categories separated by commas. Where these categories appear in particular typographic representations of the ID rule is quite immaterial.

The expanded ID/LP grammar in (58) in turn induces the set of CF-PSG rules shown in (60).

(60) VP —> VC5] NP
VP ~> VC5, PAS]
VP ~> VC5, PAS] PPCby]
VP —> VC6] NP PPCto]
VP —> VC6, PAS] PPCto]
VP —> VC6, PAS] PPCby] PPCto]
VP —.> VC6, PAS] PPCto] PPCby]

As can be seen, this set of rules contains the rule shown in $(5^0_f$ above, although no special provision was made for it. The possibility simply follows from the absence of any LP rules stipulating precedence relations between subtypes of PP. Furthermore, it is no longer possible for the metarule to stipulate that VCPAS] is phrase initial in VP. However, this ordering restriction is a consequence of the much more general LP rule shown in (58ii).

When, as in our exemplification above, one is only dealing with a single metarule jn, it is easy to see what grammar results from its application: It will consist of all the original rules JR plus the rules resulting from the application of JBL to j£. Matters become slightly more complicated when we consider closing a grammar under the application of a set of metarules having more than one member. Following suggestions due to Martin Kay, Susan Stucky, and Henry Thompson, we adopt a notion of finite closure. Essentially, our rules will be the maximal set that can be arrived at by taking each metarule and applying it to the set of rules that have not themselves arisen as a result of the application of that metarule.[26]

We provide here a recursive definition of finite closure, adapted from Thompson (1982). The finite closure (FC) of a set of rules R under a set of metarules M is defined as follows:

$$FC(R, \emptyset) = R$$

$$FC(R, M) = \bigcup_{m \in M} (m(R \cup FC(R, M-\{m\})))$$

Note that metarules are functions from sets of rules to sets of rules, not functions from rules to rules. Attempting to construe them in the latter way runs into the problem that they may not turn out to be functions, that is, they may not yield a single unique value for every argument.

Exercises (section 7)

!• Formulate a metarule analysis of VP^{f}s like the following:

a. is a woman in the garden
b. be three doctors waiting to see you

2. Formulate a metarule analysis of sentences like the following:

a. Is a woman in the garden?
b. Will Kim like Sandy?
c. Has Sandy seen Kim?
d. Did Kim eat?

3. How is it possible to omit mention of the feature ADX and FINITE ii the metarule of the answer to exercise 2 without allowing for ill-form* sentences to be described?

4. Does your answer to exercise 2 allow for <u>both</u> the following examples:

a. Have you any idea of the time?
b. Has Sandy seen Kim and Lee called Robin?

If not, why not?

5. Assuming, for the sake of simplicity, that no rule ever satisfies the structural analysis of a metarule in more than one way, what is th< maximum number of rules that can result from the finite closure of a s< of *x* rules under a set of £ metarules?

6. What connection, if any, do you see between the answer to exercise 5, above, and your answer to exercise 5 of section 2?

&• <u>The Head Feature Convention</u>

Consider a rule such as that shown in (61):

(61) VP --> V NP

If we assume a two-bar version of X-bar syntax, (61) can be expressed *i* (62):

(62) V1 ~> V N2

And (62) can itself be expressed more perspicuously as (63):

(63)

$$\begin{array}{l}[-N]\\ [+V]\end{array}^{1} \quad \longrightarrow \quad \begin{array}{l}[-N]\\ [+V]\end{array} \; \begin{array}{l}[+N]\\ [-V]\end{array}^{2}$$

Note that (63) does not, of itself, tell us that the verb is the head of the verb phrase. All versions of X-bar syntax are supplied with a definition of head, which runs along the lines shown in (64):

(64) In a rule of the form:

$$\begin{array}{l}[\alpha V]\\ [\beta N]\end{array}^{m} \quad \longrightarrow \quad \ldots \; \begin{array}{l}[\alpha V]\\ [\beta N]\end{array}^{n} \; \ldots$$

where $n \leq m$ (sometimes: $n < m$),

$[\alpha V, \beta N]^{n}$ is the head of $[\alpha V, \beta N]^{m}$.

Having provided such a definition of head, many versions of X-bar syntax then go on to provide for transfer of features other than $[\pm V, \pm N]$, from mother to head by defining a feature-trickling convention. This convention, called the Head Feature Convention (HFC) by Gazdar, Pullum and Sag (1982), says, in effect, that mother and head are to carry the same feature.

This uncontroversial proposal has a very curious property. Consider some feature, say +F(inite), which an appear on verbs and (hence) verb phrases, for example. Suppose we flesh out rule (63) to include this feature:

(65)

$$\underset{a.}{\begin{array}{l}[-N]\\ [+V]\\ [+F]\end{array}^{1}} \quad \longrightarrow \quad \underset{b.}{\begin{array}{l}[-N]\\ [+V]\\ [+F]\end{array}} \quad \underset{c.}{\begin{array}{l}[+N]\\ [-V]\\ \end{array}^{2}}$$

Now, in this rule, category (a) and category (b) instantiate the same values for all their component features. Intuitively, this should simply follow from the feature convention for heads. But it does not, in the formulations given in the work mentioned above. Instead, the correspondence of the [+V, -N] values is used to define the head, and the correspondence of the [+F] value then follows from the head feature convention. So feature correspondence is being used to define the "head of" relation, and then the latter is used to account for feature correspondence (via a head feature convention). The only thing that prevents the enterprise being circular is that different classes of features are involved in the two cases. Notice also that the definition of head, though it may look impressively general, actually only works thanks to a set of rule-stipulated feature correspondences of the kind illustrated in (65).

We wish to propose a slight modification of the theory of grammar that solves this problem. We propose to express the relation "is the head of" in terms of the feature HEAD. We will use H standing on its own as an abbreviation for categories in which the coefficient of this feature is completely unspecified. However, if H appears in a rule annotated with some minor feature, say as H[+F], then this expression abbreviates a category whose HEAD is unspecified except in respect of the feature F. Thus H stands for [CAT [BAB [LEXICAL ...] HEAD]], H[3PEH] stands for [CAT [BAB [LEXICAL ...]] [HEAD MAJOB [MINOB [AGB [PEB 3PEB] NMB]]]], and we can also use H1 and H2 as abbreviations for [CAT [BAB 1] HEAD] and [CAT [BAB 2] HEAD], etc., respectively. Hence, rule (65) will now appear as follows:

```
(66)        1                           2
        C-H]                         [+H]
        [+V]      —>       H          [-V]
        C-fP]
```

Likewise, instead of (67):

(67) V1 —> 7, N2

we shall write:

(68) V1 —> H, N2

And we may define "head[1]* as the minimal category (in the sense of bar level) which is unspecified for major features.

This definition of head entails that in an X2, the head will be an X2, X1, or X that it immediately dominates, whichever has the fewest bars, except that if there is no such category, or if there is not a unique one with fewest bars, then there is no head. Our position on X-bar syntax here is essentially that of Emonds (1976: 12-20), except that we differ from Emonds in taking the category S to be a maximal projection of V, and in taking P to be a lexical category.

Traditionally, a lexical item is often spoken of as being the head of a whole phrase. We too will sometimes speak loosely of, e.g., loves being the head of a given VP, when formally it is not a head at all, under our definition. Pedantically, this informal sense involves the ancestral (reflexive and transitive closure) of the relation we have defined.

Given our definition of head, we can now give a definition of the HFC. The HFC is part of a mapping from ID rules to instantiated extensions of those ID rules, where the latter, informally, are ID rules which have features assigned to their component categories in a manner consistent with feature principles (such as the HFC) and feature defaults. We adopt the convention of writing <X for a category in an instantiated rule A, and & for its counterpart in a rule B which A instantiates.

(69) Head Feature Convention (HFC)

If *fa* is the head of /So then HEAD(Ctp = HEAD(tt$).

As can be seen, this definition requires the coefficients of HEAD in the mother category and the head daughter to be identical, but it does so without trading on a feature identity stipulated in the original rule.

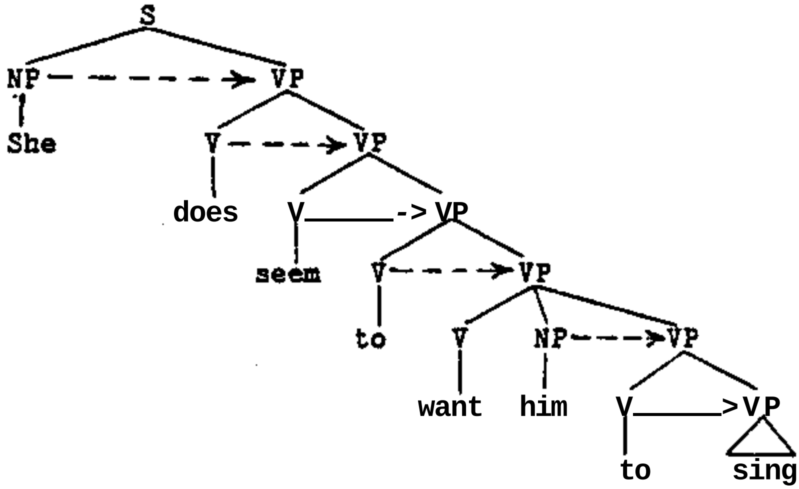
9. The Control Agreement Principle

We now address the question of what is traditionally called agreement. Most current accounts of agreement phenomena provide no basis for thinking that distribution of agreement features is not arbitrary (Lapointe (1981) is an exception). That is, they provide no basis for predicting which constituents will agree in a given language and which will not. An important step toward explaining agreement phenomena is taken by Keenan (1974). Starting from a surfacist conception of semantic structure similar to our own, Keenan (1974: 302) offers the following universal principle governing agreement processes:

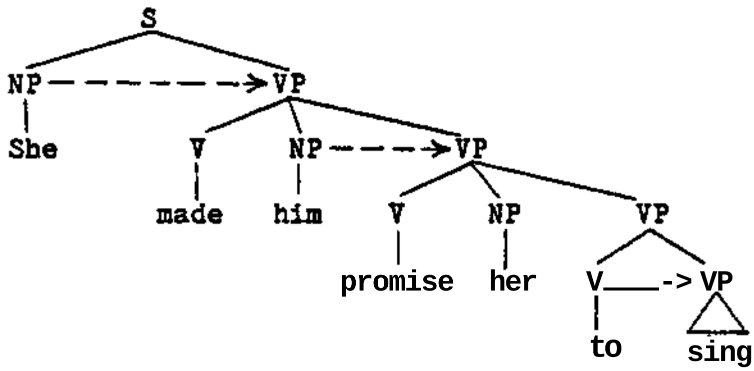
(70) Function symbols may present a morpheme whose form is determined by the noun class of the argument expression.

A more succinct formulation of (70) is: functors may agree with nominal arguments. Keenan's motivation for (70) is that the reference of a nominal argument α can, in general, be determined independently of the interpretation of any functor expression depending on α, while the converse is not true. This dependence, according to Keenan, is reflected syntactically in the fact that the morphological form of a functor may vary with the form of an argument, but not vice versa. We shall return to a consideration of this idea towards the end of the section. For the moment, we note that Keenan's functional agreement principle is significant in attempting to identify concordant constituents on the basis of their semantic relationship. If interpreted as prohibiting any instances of agreement that it does not explicitly allow, (70) places a strong constraint on the class of agreement systems made available within linguistic theory. It provides some basis for explaining why there is agreement between subjects and verb phrases in English, between attributive adjectives and head nouns in many languages, and between verbs and their direct objects (as well as their subjects) in yet others.[27] Which functors exhibit agreement morphologically is, of course, something that varies from language to language. Our theory of agreement is a generalization of Keenan's semantically driven principle. Instead of referring to function-argument application we follow Bach and Partee (1980) and introduce a notion of control that subsumes it. A controllee is a function and a controller is either an argument or an argument-passing function that applies to some controllee (see Klein and Sag (1981) for further details). Here we will restrict ourselves to providing some representative examples in an ad hoc notation. The head of the arrow indicates the controllee, and the tail the controller:

(71)



(72)



As these illustrations suggest, control is a relation holding between sisters. Since sisters are, of necessity, categories introduced by tl same rule, this means that control, as we use the term, is a relation that can be appealed to in defining the feature instantiation princip: for rules. Richer notions of control that make reference to more complex tree configurations than mere sisterhood cannot be reconstruct in a framework such as the present one where features are instantiate< on rules.

The principle we adopt, then, can be stated as follows (we contij to use the conventions introduced in connection with the HFC):

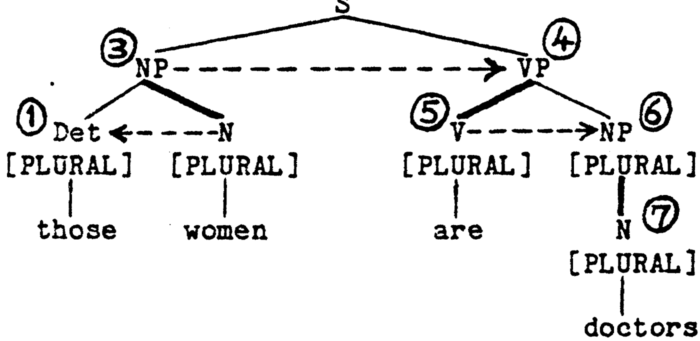
(73) <u>Control Agreement Principle</u> (£A£)

If β_i controls *fij* then AGR(Oi^) = AGR(Otj).

As can be seen, this definition just requires the coefficients of AGR controller and controllee to be identical.

Taken together, the HFC and CAP provide the basis for a highly effective theory of agreement. Consider the following example:

(74)

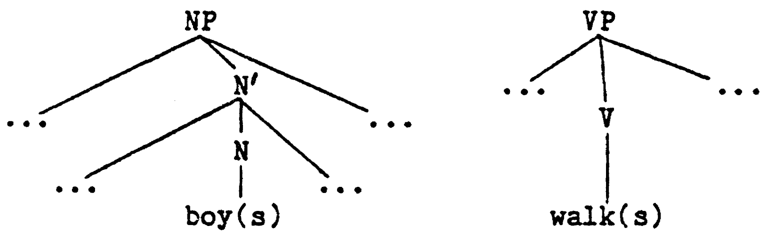


In (74), 1 agrees with 2 by CAP, 2 with 3 by HFC, 3 with 4 by CAP, 4 with 5 by HFC, 5 with 6 by CAP[28], and 6 with 7 by HFC.

Curiously, agreement has often been used in textbooks of generative linguistics to illustrate the inadequacy of context-free phrase structure grammars. So, for example, Grinder and Elgin (1973: 57-59) summarize the facts of English verb phrase agreement and assert that "The grammatical phenomenon of Subject-Predicate agreement is sufficient to guarantee [that] English is not a CF-PSG language." The phenomenon in question does not in fact determine non-CFL status for English; indeed, Lyons (1968: 239-246) provides a detailed discussion of government and concord in the course of which he gives (242-243) a CF-PSG for English subject-verb agreement. And our own analysis, outlined above, is entirely based on context-free rules, although our statement of the analysis is several levels of abstraction removed from the actual CF-PSG rules which admit structures like (74), above.

In addition to getting the basic facts right, our account correctly entails that agreement features are on whole phrases, not merely on lexical items, which means it could be manifested on several or all members of the phrase, as it is in many languages (e.g. in French _la jeune fille intelligente_, _les jeunes filles intelligentes_). And the context-free character of the account is an advantage when one considers that any of the "..." spaces in (75) might in a specific instance be filled by extraneous sentence material.

(75)



Lyons' own context-sensitive grammar fragment (see rule (4) on p. 245 of Lyons 1968) does not allow for, e.g., an adverb before the verb, but our treatment would be unaffected by this possibility, since it would not affect the head-of relation, nor the control relations.

Note finally that we have not had to make any additions to the syntax of English in order to handle subject predicate agreement of the sort we have been discussing. The facts simply follow from interaction ux two universal principles une nr^ ana UAr; wii;n line iorm or the syntactic and semantic rules which are motivated quite independently of the facts of agreement. In a real sense, as far as the central phenomena considered here are concerned, there are no rules of agreement in English, only consequences of universal grammar.[29]

10, The Foot Feature Principle

Not all features are head features. Consider the example in (76).

(76) These reports, the wording on the covers of which has caused so much controversy, are to be destroyed.

The word which is not the head of the NP the wording of the covers of which in anybody's theory of grammar. And yet which is indubitably responsible for the wh-ness of that NP, and its consequent ability to appear in the position which it occupies. If, as in many analyses, we postulate a WH feature on this NP, then we are faced with the problem of relating it to the non-head lexical item that is, in some obvious intuitive sense, responsible for it.

The same isue arises in feature-driven theories of reflexives, such as that of Gazdar and Sag (1981).[30] Such theories need to be able to say that a PP containing a reflexive NP is itself reflexive, even though the reflexive NP is not the head of the PP, and cannot be.

Finally, we note that the issue also arises if, following Bear (1981), one construes the "slash categories" of Gazdar (1981) as being categories that carry a feature encoding all relevant syntactic information about the category that is missing from the slash category. We will refer to such features, assuming them to exist, as slash features. These features plainly do not obey the HFC. Consider the example in (77).

(77) Who did you want to seem to like?

Under a slash category analysis employing slash features, we must claim, among other things, that £& like, carries the slash feature in seem to like and that i£ seem to like carries the slash feature in want to seem to like. And yet in neither case is the infinitive VP the head of the constituent that introduces it.

At first sight, it appears that we have three very different phenomena here, phenomena whose only common property is a negative one, namely the property of failing to obey the provisions of the Head Feature Convention. However, we shall be claiming that exactly the same feature instantiation principle governs all three cases, and that the differences between the phenomena all follow from independently motivated differences between the various ID rules responsible for the constructions.

As already noted, though not discussed, in section 1, above, we take the syntax of CAT^1 to be as shown in (78).

(78) [CAT^1 CAT FOOT] (= (7a))

This introduces a feature FOOT whose syntax is shown in (79).

(79) [FOOT SLASH WH REFL] (= (7c))

Since FOOT does not appear in HEAD, the foot features will not be governed by the Head Feature Convention. But that observation merely leaves open the question of what principle, if any, does govern their instantiation.

Before we turn to that question, however, it is worth pausing to point out that the features introduced by FOOT themselves have internal structure:

(80) [SLASH CAT]

(81) [WH AGR WHMOR]

(82) [REFL AGR]

Thus SLASH takes as its coefficient a category (in the sense of CAT); WH takes two coefficients, the agreement feature AGR, and a feature WHMOR encoding the morphological type of the wh word involved[31]; and REFL takes AGR as its coefficient.[32]

We will call the principle responsible for the distribution of foot features "Foot Feature Principle" (FFP). The intuitive idea underlying the FFP is a rather straightforward one, although its formal expression may turn out a little opaque to readers who skipped section 2. The idea is this. There are foot features that are explicitly specified in listed ID rules, or which have arisen through the operation of metarules. Such foot features are inviolate and cannot be copied or otherwise tampered with in the feature instantiation mapping. However, other foot features arise on daughters in virtue of the free instantiation permitted in that mapping. And these features must appear on the mother also. So consider a hypothetical case of an ID rule introducing 3 daughters, where neither mother nor daughters have any foot features specified. Then feature instantiation might lead to one daughter getting SLASH, another WH, and the third REFL. The FFP simply requires the mother to carry all three.

Consider the case where feature instantiation "tries" (to speak anthropomorphically) to put the category PP as the value for SLASH on one daughter, and the category NP as the value for SLASH on another. FFP will require the mother to have both as the coefficient of SLASH in its FOOT feature. But this is impossible since NP and PP are distinct categories. Thus feature instantiation will not take place.[33]

Now consider the case where feature instantiation puts, say, [REFL [AGR [PER 1PER] [NMB 1NMB]]] into the FOOT of one daughter PP of a VP rule, and exactly the same reflexive feature into the FOOT of another PP daughter. Here FFP will require the mother VP to have both these features as one coefficient of its FOOT. And, since they are one and the same feature, this will be possible. Thus we arrive at the VP rule responsible for such examples as (83a), but feature instantiation will not legitimate the rule that would be necessary to generate (83b).

(83) a. To talk to myself about myself is boring.
b. *To talk to myself about yourself is fascinating.

Having, hopefully, now made the mode of operation of the FFP reasonably clear, we can now express it somewhat more precisely, using the terminology, though not, as yet, the formalism, of section 2.

(84) Foot Feature Principle (FFP)

> The increment of the mother category's FOOT feature is the unification of the increments of the daughter categories' FOOT features.

We will exhibit a proper formal definition of FFP in the next section. (84) says that the foot feature appearing on the mother in virtue of feature instantiation brings together all the additions to foot features in daughters that have arisen in virtue of feature instantiation.

We will illustrate this by reference to SLASH. We introduce the notation α/β as an abbreviation for
[CAT' α [FOOT [SLASH β]]].
Thus S/NP is a sentence with an NP CAT as the coefficient of its SLASH feature. Consider the three ID rules shown in (85).[34]

(85) a. AP --> A, VP/NP
b. VP/NP --> V, VP
c. VP --> V, PP, PP

Here, (85a) is the rule responsible for expressions like easy to solve, (85b) for expressions like force to solve the problem as in who did you force to solve the problem, and (85c) for expressions like talk to Kim about Sandy.

From these, the FFP will allow feature instantiation to provide us with, for example:

(86) VP/NP --> V, PP, PP/NP

But it will not license either (87a) or (87b) in virtue of (85a) and (85b).

(87) a. AP/NP --> A, VP/NP
b. VP/NP --> V, VP/NP

Thus the grammar will generate examples like those in (88a) and (88b) but not those in (89a) and (89b):

(88) a. Who did you talk to Kim about?
b. Who did you talk about Sandy to?

(89) a. *What was the problem easy to solve?
b. *Who did you force to solve?

Exercises (section 10)

1. Assume just the following metarule for slash elimination:

$$\alpha \text{ --> } W \quad XP[-CASE]$$
$$\Downarrow$$
$$\alpha/XP[-CASE] \text{ --> } W$$

together with a relevant set of ID and LP rules. Which of the follow examples will the grammar induce?

a. *Which slave did you give a picture to Felix?
b. Which slave did you give ___ to Felix?
c. Which slave did you give a picture to ___?
d. Which slave did you buy a picture of ___?
e. ?Which slave did you give a picture of ___ to Felix?
f. *Which slave did you give ___ ___?
g. *Which slave did you give ___ to ___?
h. ??Which slave did you talk to ___ about ___?
i. Which slave did you give a picture of ___ to ___?
j. Which citizen do you think ___ gave a slave to Felix?
k. *Which citizen do you think that ___ gave a slave to Felix?
l. ??Which slave do you think that a picture of ___ would please F
m. ?Of which slave do you think that a picture ___ would please F
n. Which citizen's slave did you give ___ to Felix?
o. *Which citizen's did you give ___ slave to Felix?
p. *Which citizen's did you give the clothes to ___ slave?
q. *Which citizen's did you give the clothes of ___ to a slave?

2. Discuss the implications of the discrepancies, if any, between th predictions of the grammar and the acceptability of the examples list in exercise 1.

3. Assume the following rule for topicalization:

S --> XP S/XP

Does the grammar induce either or both of the following examples?

a. ?Nixon, pictures of ___, nobody collects ___.
b. Himself, Nixon admires ___.

In each case, if it does <not>, then explain why it does <not>.

4. Assume the topicalization rule given in exercise 3, and the following rule for free relatives:

XP --> XP[WH EVER] S/XP

These rules do <u>not</u> interact in such a way as to generate the followin example.

a. *Whoever, Liddy admires [___ [Nixon admires ___]].

Modify the theory of categories so that this ungrammatical example <u>wi</u> be generated.

5. Do you see any connection between your answers to exercises 3 and Elaborate.

11. Feature Instantiation

In the two preceding sections we have outlined the two fundamental feature instantiation principles responsible for ensuring the proper distribution of features in rules. But we have not said how these principles interact with each other, nor have we explained how they interact with the system of feature coefficient defaults discussed in section 3, above. It is to this question of interaction that we turn our attention now.

We define the notion instantiated extension of a rule. Intuitively, an instantiated extension of a rule is an extension in which features not specified in the original rule have either been freely assigned in a manner consistent with their default coefficients, if any, or have been set equal to some other set of features in virtue of the Head Feature Convention, the Control Agreement Principle, or Foot Feature Principle.

Before we define instantiated extensions, however, we need to provide a subsidiary definition of the notion privileged. Intuitively, a coefficient is privileged if it is exactly as some rule, convention, or universal principle says it should be, and thus is not intended to be within the scope of the default or "elsewhere" statements for filling in neutral or "unmarked" feature values.

Let A and B be rules that can be written $\alpha_0 \rightarrow \alpha_1, \ldots, \alpha_n$ and $\beta_0 \rightarrow \beta_1, \ldots, \beta_n$ respectively ($1 \leq n$), and for each i ($0 \leq i \leq n$), α_i is an extension of β_i. Then we can define the notion privileged, as follows:

(90) An occurrence C of a feature coefficient is privileged in α_i ($1 \leq i \leq n$) if and only if at least one of the following conditions is met.

1. C is in β_i.
2. C is in HEAD(α_i) and β_i is the nonlexical head of β_0.
3. C is in AGR(α_i) and, for some j, $1 \leq j \leq n$, β_i controls or is controlled by β_j.
4. C is in FOOT(α_i).

Clause 1 says that features specified in rules take precedence over defaults. Clause 2 exempts phrasal heads from being required to take default coefficients. Without this clause the HFC would not work properly: intermediate phrasal heads would standardly be assigned default coefficients and this would entail that marked coefficients could never be dripped or percolated through such heads. Clause 3 exempts controllers and controllees from being required to take default coefficients. Clause 4 exempts FOOT features from being required to take default coefficients.

Given this definition of privileged, we are now in a position to define the notion instantiated extension of a rule. A and B remain as characterized above,

(91) A is an instantiated extension of B if and only if the following conditions are satisfied for all i, j (1 ≤ i, j ≤ n):

1. *CCi* is a well-formed syntactic category,

2. Head Feature Convention (HFC)
If &i is the head of A, then HEAD(CCp r HEAD(α0).

3* Control Agreement Principle (CAP)
If &i controls Qu, then AGR(&£) = AGR(C6j).

4. Foot Feature Principle (FFP)
There exist features Fk, 0 ≤ k ≤ n, such that

(i) Uni(F0, ..., Fn), and
(ii) Inc(Fk, FOOT(#k), FOOTjSk))

5. Default Assignment Convention (DAC)
All non-privileged coefficients in CC^ match their defaults, if any.

This is more or less self-explanatory. Clause 1 ensures that putative categories in the instantiated extension really are categories, that is, it requires them to be consistent both with the specified syntax for categories, and with the FCR's, Clauses 2, 3, and 4 are the HFC, CAP, and FFP, as discussed in previous sections, except that this version of FFP is formally precise. And clause 5 ensures that defaults are assigned to the non-privileged occurrences of coefficients.

Exercises (section 11)

1. Existing GPSG analyses of unbounded dependency constructions miss a generalization about rules having the following form:

A —> B C/D

Such rules (topicalization, relative clauses, *Mh* questions, etc.) all have to stipulate that B and D are identical except, perhaps, in respect of certain features specified in the rule itself. Formulate a feature instantiation principle, or generalize an existing one, to capture this generalization, and show how the relevant rules can be simplified.

2, Does your solution to exercise 1 extend to the rules necessary for free relatives, it clefts and easy type adjectives? If it does, then demonstrate this. If it does not, then revise your solution so that it does. Discuss any issues that arise.

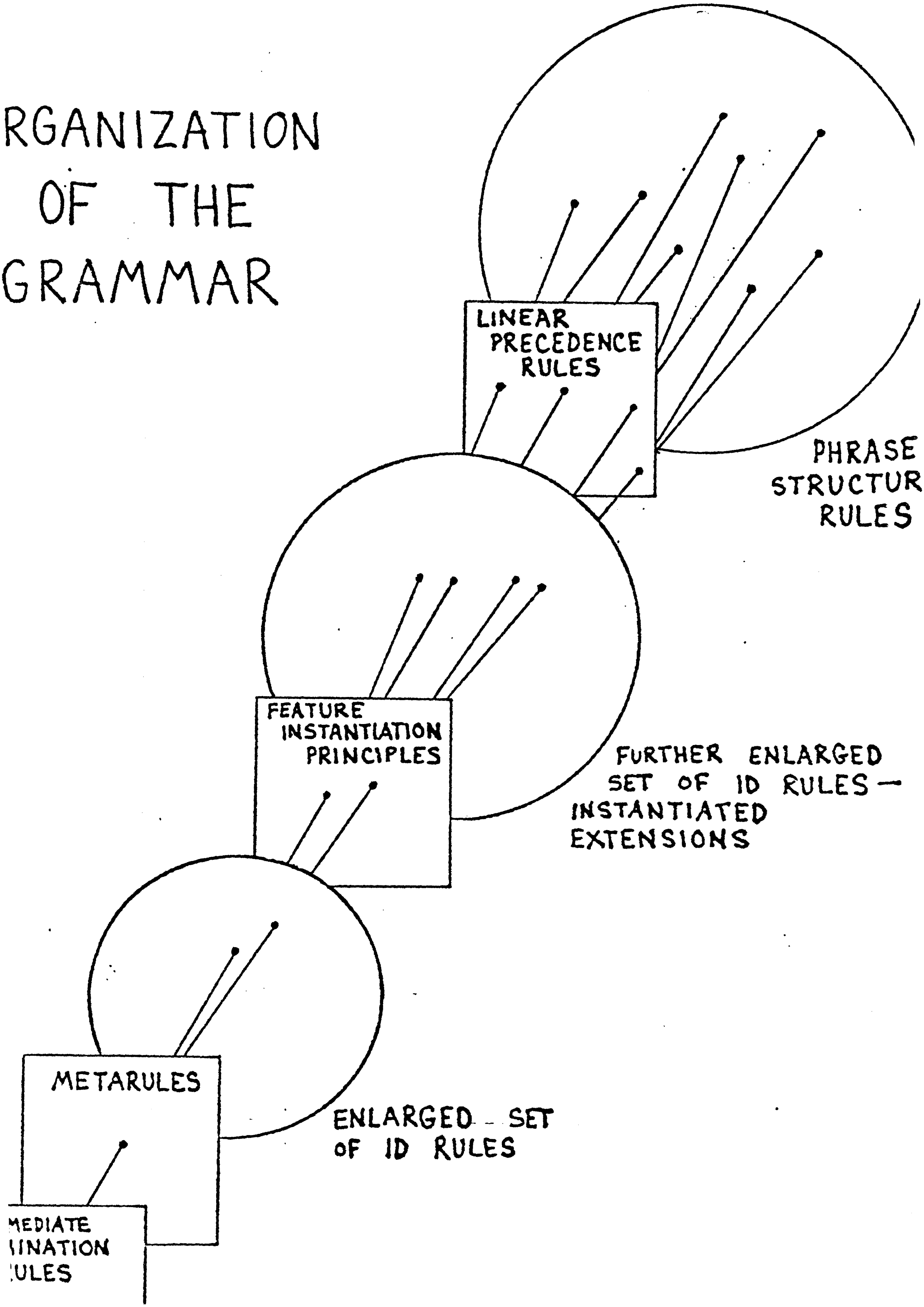
12. The organization of the grammar

The theory of grammar outlined in this paper contains four components as summarized in (92).

(92)
- i. Immediate dominance rules
- ii. Metarules
- iii. Feature instantiation principles
- iv. Linear precedence rules

This metagrammar defines a phrase structure grammar in the following way: the phrase structure grammar consists of all those phrase structure rules which are consistent with every linear precedence rule and some instantiated extension of an immediate dominance rule in the finite closure of the immediate dominance rules under the metarules. This may be more clearly expressed in the form of a diagram.

ORGANIZATION OF THE GRAMMAR



Organizing the grammar in this way makes a number of non-trivial empirical claims. For example, (i) Metarules cannot make reference to features assigned by default, or in virtue of the HFC, CAP, or FFP. (ii) Metarules and feature instantiation principles cannot make reference to ordering relations holding between sisters (or any other ordering relation, for that matter). This predicts that no language will instantiate features differently for different constituent orders of sisters. So, for example, we would not expect to find a language with a special case for the first NP of a sequence of sisters, or a special case on an NP that immediately follows a verb regardless of the grammatical status of that NP. (iii) By determining linear precedence after feature instantiation, the grammar leads us to expect that there will be languages which, for example, order lexical heads differently according to category[35], or which have linearization principles that make reference to features assigned by convention or default.[36] And (iv) by assigning linear precedence to rules, rather than by having linearization principles applying to structures, we predict that ordering constraints in natural languages are limited to sisters, without having to stipulate this as an extra condition on linearization

[1] Two grammars are weakly equivalent if and only if they generate th same language (i.e. set of strings). Two grammars are strongly equivalent if and only if they induce isomorphic sets of structural descriptions. This latter definition presupposes the availability of definition of isomorphism over the types of structural description involved.

[2] See, e.g., Aho and Ullman (1972), Hopcroft and Oilman (1979), Kimball (1973), or Wall (1972).

[33 See Chomsky (1970, 1974) for the origin of this proposal, and Bresnan (1976, 1977) for further development. And see Jackendoff (197 for discussion and defense of an alternative.

[4] There are some indications that this idea is being tacitly assumed even in phonology. For example, consider the statement of Nasal Assimilation in Odawa Ojibwa provided by Iverson and Sanders (1980, 180):

(i) [C, +nasal] —> [Ctplace]/_____[^obstruent, OCplace]

Iverson and Sanders say nothing about what sort of feature system they assume. It is possible they assume a system in which [place] is a multi-valued feature with values like "interdental", "palato-alveolar" "uvular", etc. But there is a way to interpret their notation under t more widespread assumption that place of articulation is specified by giving values for a set of binary features like the [anterior] and [coronal] of Chomsky and Halle (1968). Suppose [place] takes as its coefficient a sequence of feature specifications: one for [anterior] and one for [coronal]. Writing features in terms of the notation adopted below, we would have instantiations for *[OL*place] looking like for example, [place [anterior 1] [coronal 0]] (picking out the labials Iverson and Sanders' variable 06 could be taken as ranging over whole coefficients for [place], i.e. over specifications for [anterior] and [coronal], which themselves demand coefficients from the set {0, 1} (t phonologist's - and +).

[5] Note that [F1 F2] and [F1 [F2]] are the same feature, but [F1 F2 F and [F1 [F2 F3]] are distinct, and [[F1] [F2]] and [[F1] F2] are not features at all. The leftmost feature name is the name of the feature and thus has a quite different status from the features or feature nan that follow it.

[6] For reasons of familiarity and convenience, we use a rather peculi notation here. The coefficients of the feature MAJOR, "+N", "-V", etc., should be regarded as monadic feature coefficients here, not notational abbreviations for the presence of an "N" feature and the absence of a "V" feature. Everywhere else in the feature system, "+" and "-" are used according to the convention described in the text above.

[7] Note that "accusative" is represented by the <u>absence</u> of the featur CASE.

[8] Here, and elsewhere in the paper, we employ some obvious abbreviatory devices in the presentation of features. Thus, for

example, [-N +V] can only be coefficients of MAJOR and there is thus no need to mention the latter in (16).

[9] There is an interesting issue about whether (16) represents a universal. There are languages--take Turkish, for example--in which superficially verbless clauses have a tense marker attached to a predicative adjective. And in Navajo, the tense of a clause can be morphologically manifested on its complementizer. It remains to be determined whether the right analyses in these cases involve α[+FIN] for $\alpha \neq$[-N,+V].

[10] "Accusative" is the traditional name for the case of a word like me; but nothing hangs on the name. What we really intend here is a reference to the neutral, unmarked case for NP's in a language. In an ergative language this case is usually called the "absolutive".

[11] The homonymic directional preposition to as in Sandy walked to the station will be listed in the lexicon separately, and not treated as an instance of P[+to]. This separate listing is independently necessitated by the semantic difference between the two to 's.

[12] This definition is intendedly imprecise. One could only make it precise by adopting a position on proper names and the status of rigid designators (see McCawley 1981: 483-484), and this is not an appropriate topic to raise here. It is sufficient to note that proper names are not to be taken to be logical constants for our purposes.

[13] Thus setting up a terminal symbol feature associated with a word like hippopotamus, for example, is ruled out in principle.

[14] Brame (1978) and Bresnan (1978) might seem to take this view, but in fact they postulate a level of "functional structure" which apparently has to be regarded as syntactic, in the sense that linguistic rules can refer to properties of representations at that level. We are not sure that anyone currently espouses the "semantic filtering" approach we are discussing.

[15] Under some transformational approaches this example and the next two could be taken to illustrate governed application of transformational rules rather than subcategorization; but in all lexicalist theories and most recent versions of transformational grammar, subcategorization is involved.

[16] Sentences like We extend to you our sincere condolences might look as if they refute the claim in the text, but when such structures are treated as they are in Gazdar (1981), they are not incompatible with it.

[17] Chomsky (1965: 110-111, 166-167) discusses inconclusively some conventions that might supply these negative specifications.

[18] As a matter of historical interest, we note that Chomsky and Halle propose a very similar mechanism to the one we propose in the course of dealing with lexical exceptions to phonological rules (1968, 173):

> Each rule of the phonology has a certain identifying number. We associate with each number n a new "distinctive feature" [$\pm$n]. Suppose that the rule numbered n is A --> B / C—D. Then we stipulate that A must be marked [+n] if the rule numbered n is

to apply to it.

[19] This ID rule can be written typographically in 6 different ways (A ~> B, C, D;' A ~> C, B, D; A ~> D, C, B; A —> B, D, C; A —> C, D, B; A —> D, B, C). However, these six versions of the rule all make exactly the same claim, and are, in fact, the same rule.

[20] Early suggestions by Curry (1961) and Šaumjan and Soboleva (1963) were criticized by Chomsky (1965, 124ff), though in retrospect his criticisms seem to have little substance. Dowty (1982) has since taken up some of the ideas of Curry's paper. Several other linguists worked during the late 1960's and early 1970's on types of transformational grammar that had unlinearized trees as syntactic representations at early stages of derivations (see e.g. Anderson 1967 [published 1976], Staal 1967, Sanders 1970, Peterson 1971, Hudson 1972), and the idea has been an important consideration in relational grammar (Perlmutter and Postal 1977) and has surfaced again recently in transformational work (Zubizaretta and Vergnaud 1981) and elsewhere (Flynn 1982).

[21] Note that neither (39) nor (40) captures a relevant generalization about AUX and V, namely that they share the same ordering constraint. Cf. Gazdar, Pullum and Sag (1982) on this point. (Note, though, that this paper was written before ID/LP format was developed.)

[22] Two points should be made here. First, the list in (41) is <u>not</u> Stucky's analysis. Stucky deals with the freedom of order in Makua by defining a small set of basic rules and a set of metarules that operate on them. We have closed part of her grammar under the operation of her metarules to obtain the set of PS rules that <u>would</u> have to be listed if metarules were not employed. Second, we have omitted some VP rules that introduce infinitives and unbounded dependency-containing constituents, because these rules raise issues irrelevant to our concerns here. The rules we omit do not, however, raise any problems for the type of analysis we are presenting.

[23] In each case, the addition or subtraction of rules suffices to achieve a partial ordering closure, as ECPO requires. These examples show that an ECPO grammar can have a proper subgrammar that is not ECPO, and it can have a proper supergrammar that is not ECPO. Furthermore, all grammars have ECPO subgrammars and supergrammars. This means that if we are interested in the ECPO property, we must be careful not to limit our attention to some proper subset of constructions in the language that, uncharacteristically for the overall grammar, either have the property, or lack it.

[24] The idea of using a grammar to generate one's grammar originates, as far as we know, with van Wijngaarten (1969)> who used the technique to give a perspicuous syntax for ALGOL68. A good introduction to his work can be found in Cleaveland and Ozgalis (1975). Janssen (1979) employs a van Wijngaarten-style two-level grammar to define a generalization of Montague's PTQ syntax. See also Langendoen (1976) for some mathematical results on the generative capacity of systems employing grammar generating grammars.

[25] The notation used makes metarules look suspiciously like transformations, but appearances here are deceptive: a transformation

formalizations, trees into trees), whereas a metarule maps rules into rules. If one adds transformations to the theory of CF-PSG, then (i) one is employing two quite distinct rule types; (ii) one completely changes the expressive power of the theory; and (iii) one ends up with a grammar that assigns at least a pair of structural descriptions to each string generated. By contrast, if one adds metarules in the way that we do here, where they amount in fact to nothing more than a novel type of rule-collapsing convention, then one merely enlarges, in a rule-governed way, the set of CF-PSG rules one is employing. The grammar itself remains a CF-PSG, and remains CFL-inducing.

[26] If a the members of a set of metarules are permitted to interact in such a way as to recursively define an infinite set of rules, it is possible for the language generated by the grammar to be non-context-free, even if the rules of the infinite grammar are all of the context-free type (Langendoen (1976)). Gazdar (1982, 180, n. 28) mentions a conjecture of Aravind Joshi's that if no more than one essential variable is permitted in a metarule, only context-free languages can result. Interestingly, this is now known to be false. Hans Uszkoreit has proved that a metagrammar using only one-variable metarules and context-free basic rules can define a non-context-free language. A brief and elegant demonstration of this has since been constructed by Chris Culy. Culy offers the following metagrammar:

```
Basic rule:     S  -->  abc

Metarules:  (i) S --> aX        (ii) S --> bX        (iii) S --> cX

                     ⇓                   ⇓                     ⇓

                S --> Xaa            S --> Xbb             S --> Xcc
```

The language generated cannot be context-free, because its intersection with the regular language a*b*c* is the following non-context-free language:

$$a^{2^n}b^{2^n}c^{2^n}$$

The theory of metarule application adopted here, i.e. finite closure, does not have consequences of this sort.

[27] Bach (1981, 145-146) contains the first published application of Keenan's principle to the issue of achieving concord in a phrase structure grammar, so far as we are aware.

[28] We assume here, uncontroversially, that predicate nominals are functions, not arguments.

[29] Of course, not all properties of agreement in all languages follow from universal grammar. There are some messy codicils on even a rudimentary system like the English one (cf. Morgan 1972), and elsewhere considerable complexity is found, as is well known (cf. Corbett, in press). How much of this complexity can be related to interesting principles and how much must be stipulated in rules is a topic for

future research.

[30] Gazdar and Sag say glibly that they "assume that independently needed feature conventions allow" the relevant things to happen. But they do not provide them.

[31] As has often been observed, there are slight, but nonetheless absolute, distinctions in the sets of *Mh* items that can appear in the various jail constructions. Thus items ending in -*ever* (*whoever*, *whomsoever*, *whatever*, etc.) cannot appear as relative pronouns, and nor can *what* in standard varieties of British and American English.

[32] Since CAT will, in general, contain an occurrence of AGR, it follows that all three foot features contain it as a potential ingredient. There may be a generalization waiting to be expressed here

[33] These informal remarks are misleadingly "process-like". No procea of any kind is involved. Feature instantiation is defined in the next section as a simultaneous well-formedness definition. Rules which satisfy it can be invoked to induce trees, those that cannot cannot.

[34] We would claim that rule (85b) arises as a result of a metarule. But the distinction between listed rules and those produced by metarule is not relevant to feature instantiation. See section 12 for discussion.

C35] E.g. Latin, Persian, German, andNahuatl, to name but four.

[36] E.g. tense in German, case in Dyirbal, animacy in Navajo, person i Spanish, pronominality in Haida, reflexiveness in French, and many othe cases.

REFERENCES

Aho, Alfred V., and Jeffrey D. Ullman. 1972. The Theory of Parsing, Translation, and Compiling. Vo^pmfl £• Parsing, Prentice-Hall, Englewood Cliffs, NJ.

Anderson, Stephen R. 1967C1976]. Concerning the notion "base component of a transformational grammar". In James D. McCawley, ed* 1976, Syntax ai^d Semantics £: Notes from the Linguistic? Underground, 113-128. Academic Press, New York, NY,

Bach, Emmon. 1981a. Nontransformational syntax. Linguistic Research 17, 131-147.

Bach, Emmon. 1981b. Discontinuous constituents in generalized categorial grammar. In 7. A. Burke and J. Pustejovsky, eds., Proceedings of the 11tfr Annual Meeting off the North Easterq Linguistic Society, 1-12. Department of Linguistics, University of Massachusetts, Amherst.

Bach, Emmon, and Barbara Partee. 1980. Anaphora and semantic structure. In J. Kreiman and A. E. Ojeda, eds., Papers from the Parasession on Pronouns an^ Anaphora, 1-28. Chicago Linguistic Society, Chicago, IL*

Baker, Carl Leroy. 1978. Introduction to transformational-generative syntax. Englewood Cliffs: Prentice-Hall.

Bear, John. 1981. Gaps as syntactic features, MA dissertation, University of Texas, Austin, TX.

Bresnan, Joan W. 1976. On the form and functioning of transformations. Linguistic Inquiry 7, 3-40.

Bresnan, Joan W. 1977« Transformations and categories in syntax. In Ronald Butts and Jaako Hintikka, eds., Baaifi Problems JUa Methodology and Linguistics. D. Reidel, Dordrecht, Holland.

Bresnan, Joan W. 1978. A realistic transformational grammar. In Linguistic theory and psychological reality, ed. by Morris Halle, Joan W. Bresnan, and George A. Miller. 1-59* MIT Press, Cambridge, MA.

Chomsky, Noam. 1965. Aspects of the theory of syntax. MIT Press, Cambridge, MA.

Chomsky, Noam. 1970. Remarks on nominalization. In Readings in English transformational grants.f, ed. by Roderick A. Jacobs and Peter S. Rosenbaum, 184-221. Ginn, Waltham, MA.

Chomsky, Noam, and Morris Halle. 1968. *Ih&* M n d Pattern of English. Harper and Row, New York.

Cleaveland, J. Craig, and Robert C. Uzgalis. 1975. fTr^r"Par;8t for programmini? languages i what every progrpniT^r should fenow about grammar. American Elsevier, New York, NT.

Corbett, Greville. 1982* Resolution rules: agreement in person, number, and gender. In Gerald Gazdar, Ewan Klein, and Geoffrey K. Pullum, eds., Orderf Concordf and Constituency. Foris, Dordrecht, Holland*

Curry, H. B. 1961 • Some logical aspects of grammatical structure. In R. Jakobson, ed., Structure of Language and its Mathematical Aspects (Proceedings of the Twelfth Symposium in Applied Mathematics), 56-68. American Mathematical Society, Providence, HI.

Dowty, David. 1982. Grammatical relations and Montague grammar. In Pauline Jacobson and Geoffrey K. Pullum, eds., JLHSL Nature of Syntactic Representation 79-130. D. Reidel, Dordrecht, Holland.

Emonds, Joseph E. 1976. A transformational approach to English syntax;: rootf structure-preservingf and local transformations. Academic Press, New York, NY.

Flynn, Michael. 1982. A categorial theory of structure building. In Gerald Gazdar, Ewan Klein, and Geoffrey K. Pullum, eds., Orderf Concordf 2LDA Constituency. Foris, Dordrecht, Holland.

Gazdar, Gerald. 1981. Unbounded dependencies and coordinate structure. Linguistic Inquiry 12, 155-184.

Gazdar, Gerald. 1982. Phrase structure grammar. In Pauline Jacobson and Geoffrey K. Pullum, eds., The Nature of Syntactic Representation 131-186. D. Reidel, Dordrecht, Holland.

Gazdar, Gerald, Geoffrey Pullum, and Ivan A. Sag. 1981. Auxiliaries and related phenomena in a restrictive theory of grammar. Language 58.

Gazdar, Gerald, and Ivan A. Sag. 1981. Passive and reflexives in phrase structure grammar. In J. A. G. Groenendijk, T. M. V. Janssen, and M. B. J. Stokhof, eds., Formal Methods jja the Study JQ£ Language (Mathematical Center Tract 135), 131-152. Mathematical Center, Amsterdam, Holland.

Grinder, John T., and Suzette Haden Elgin. 1973* A fiuidfi. i£ Transformational gpflffffmr. Holt Rinehart and Winston, New York, NY*

Halle, Morris. 1969• How not to measure length of lexical representations and other matters. Journal of Linguistics 5f SOS-SOS.

Harman, Gilbert. 1963. Generative grammars without transformation rules: a defense of phrase structure. Language 39, 597-616.

Harris, Zellig. 1951. Methods in structural linguistics. Chicago University Press, Chicago, IL.

Hellan, Lars. 1977. $\bar{X}$-syntax, categorial syntax and logical form. In Thorstein Fretheim and Lars Hellan, eds., Papers from the Trondheim Syntax Symposium, 83-135. University of Trondheim, Trondheim, Norway.

Heny, Frank. 1979. Review of Noam Chomsky, The Logical Structure of Linguistic Theory. Synthese 40, 317-352.

Hopcroft, John E., and Jeffrey D. Ullman. 1979. Introduction to Automata Theory, Languages, and Computation. Addison-Wesley, Reading, MA.

Horrocks, Geoffrey. 1982. The order of constituents in modern Greek. In Gerald Gazdar, Ewan Klein, and Geoffrey K. Pullum, eds., Order, Concord, and Constituency. Foris, Dordrecht, Holland.

Hudson, Grover. 1972. Is deep structure linear? In George Bedell, ed., Explorations in Syntactic Theory (UCLA Papers in Syntax 2), 51-77. University of California, Los Angeles.

Iverson, Gregory K., and Gerald A. Sanders. 1980. The functional determination of phonological rule interactions. Studia Linguistica 34, 168-196.

Jackendoff, Ray S. 1977. $\bar{X}$ syntax. MIT Press, Cambridge, MA.

Jakobson, Roman; Gunnar Fant; and Morris Halle. 1952. Preliminaries to Speech Analysis. MIT Press, Cambridge, MA.

Janssen, T. M. V. 1979. On problems concerning the quantififcation rules in Montague grammar. Preprint, Mathematisch Centrum, Amsterdam, Holland.

Keenan, Edward L. 1974. The functional principle: generalizing the notion of 'subject of'. Papers from the Tenth Regional Meeting, 298-309. Chicago Linguistic Society, Chicago, IL.

Kimball, John. 1973. The Formal Theory of Grammar. Prentice-Hall, Englewood Cliffs, NJ.

Klein, Ewan, and Ivan A. Sag. 1981. Semantic types and control. Presented to the Annual Meeting of the Linguistic Society of America, New York, NY.

Langendoen, D. Terence. 1976. On the weak generative capacity of infinite grammars. CUNYforum 1, 13-24.

Lapointe, Steven G. 1981. A lexical analysis of the English auxiliary verb system. In Teun Hoekstra, Harry van der Hulst, and Michael Moortgat, eds., Lexical Grammar, 215-254. Foris, Dordrecht, Holland.

Lyons, John. 1968. Introduction to Theoretical Linguistics. Cambridge University Press, Cambridge, England.

McCawley, James D. 1981. Everything That Linguists Always Wanted To Know About Logic* (*but were afraid to ask). Chicago University Press, Chicago, IL.

Morgan, Jerry L. 1972. Verb agreement as a rule of English. Papers from the Eighth Regional Meeting, 278-286. Chicago Linguistic Society, Chicago, IL.

Perlmutter, David M., and Paul M. Postal. 1977. Toward a universal theory of passivization. Proceedings of the Third Annual Meeting of the Berkeley Linguistic Society, 394-417. Berkeley Linguistic Society, Berkeley, California.

Peterson, Thomas. H. 1971. Multi-ordered base structures in generative grammar. In Papers from the Seventh Regional Meeting, Chicago Linguistic Society, 181-192.

Pollard, Carl. 1982. Generalized grammar. Unpublished paper, Stanford University.

Pullum, Geoffrey K. 1982. How many possible human languages are there? Paper presented at the Linguistic Society of America Summer Meeting, Uniiversity of Maryland.

Pullum, Geoffrey K., and Gerald Gazdar. 1982. Natural languages and context-free languages. Linguistics and Philosophy 4, in press.

Sanders, Gerald A. 1970. Constraints on constituent ordering. Papers in Linguistics 2, 460-502.

Šaumjan, S. K., and P. A. Soboleva. 1963. Applikativnaja poroždajuščaja model' i isčislenie transformacij v russkom jazyke. Izdatel'stvo Akademii Nauk SSSR, Moscow.

Staal, J. F. 1967. Word Order in Sanskrit and Universal Grammar. (Foundations of Language Supplementary Series 5.) D. Reidel, Dordrecht, Holland.

Stucky, Susan. 1981a. Word order variation in Makua: a phrase structure grammar analysis. PhD dissertation, University of Illinois, Urbana-Champaign.

Thompson, Henry. 1982. Handling metarules in a parser for GPSG. (Department of Artificial Intelligence Research Paper 175, University of Edinburgh.) *Proceedings of the Ninth International Conference on Computational Linguistics*. North-Holland, Amsterda Holland.

Wall, Robert. 1972. *Introduction to Mathematical Linguistics*. Prentice-Hall, Englewood Cliffs, NJ.

Wijngarten, A. van. 1969. Report on the algorithmic language ALGOL 6 *Numerische Mathematik* 14, 79-218.

Williams, Edwin. 1981. On the notions 'lexically related' and 'head of a word'. *Linguistic Inquiry* 12, 245-274.

Zubizaretta, Maria-Louisa, and Jean-Roger Vergnaud. 1981. On virtual categories. Unpublished paper, MIT.

Cognitive Science Research Papers

1. C.S. Mellish, (1982) 'Incremental Evaluation: An Approach to the Semantic Interpretation of Noun Phrases'. [Price £1.00] Serial no: CSRP 001

2. G. Gazdar, (1982) NL's CFL's and CF-PSG's. [Price £0.50] Serial no: CSRP 002

3. S. Hardy, (1982)The Poplog Programming System. [Price £1.00] Serial no: CSRP 003

4. A. Sloman & Monica Croucher, (1981) 'You don't need a soft skin to have a warm heart: Towards a computational analysis of motives and emotions'. [Price £1.25] Serial no: CSRP 004

5. J. C. Rutkowska, 'Explaining Infant Perception: Insights from Artificial Intelligence?' [Price £0.75] Serial no: CSRP 005

6. G. Gazdar, E. Klein, G.K. Pullum & I.A. Sag, (1982) 'Coordinate Structure and Unbounded Dependencies'. [Price £1.00] Serial no: CSRP 006

7. G. Gazdar & G.K. Pullum, (1981) 'Generalized Phrase Structure Grammar: A Theoretical Synopsis'. [Price £1.25] Serial no: CSRP 007

8. A. Sloman, (1982) 'Image Interpretation: The Way Ahead'. [Price £0.75] Serial no: CSRP 008

9. A. Sloman, (1982) 'Towards a Computational Theory of Mind'. [Price £0.50] Serial no: CSRP 009

10. C. Mellish & S. Hardy (1982) 'Integrating Prolog in the Poplog Environment' [Price £0.50] Serial No: CSRP 010

11. A. Sloman (1982) 'Computational Epistemology'. [Price £0.75] Serial no: CSRP 011

12. A. Sloman (1982) 'Why beginners need powerful systems' [Price £0.50] Serial no: CSRP 012

13. A. Sloman (1981) 'Skills learning and parallelism' [Price £0.50] Serial no: CSRP 013

14. A. Sloman (1981) 'Artificial Intelligence and Popper's Three Worlds' [Price £0.50] Serial no: CSRP 014

15. M.A. Boden, 'Failure is not the Spur' [Price £0.50] Serial no: CSRP 015

16. M.A. Boden, 'Artificial Intelligence and Animal Psychology' [Price £0.50] Serial no: CSRP 016

17. M.A. Boden, 'The Educational Implications of Artificial

18. M.A. Boden 'Artificial Intelligence and Social Forecasting' [Price £0.75] Serial no: CSRP 018

19. M.A. Boden 'Fashions of Mind'. [Price £0.50] Serial no: CSRP 019

20. M.A. Boden 'Methodological Links Between AI and Other Disciplines', [Price £0.50] Serial no: CSRP 020

21. M.A. Boden 'Artificial Intelligence and Biological Reductionism', [Price £0.50] Serial no: CSRP 021

22. M.A. Boden 'The Meeting of Man and Machine', [Price £0.75] Serial no: CSRP 022

Also available:

S. Hardy, & A. Sloman, (1982) 'POPLOG A Multi-Purpose, Multi-Language Program Development Environment'. [NO CHARGE]

All orders and enquiries should be made to Judith Dennison, ARTS E, University of Sussex, Falmer, Brighton, BN1 9QN. Please make cheques payable to "The University of Sussex".