6

How the Grammar Works

6.1 A Factorization of Grammatical Information

Three chapters ago, we began modifying the formalism of context-free grammar to better adapt it to the sorts of generalizations we find in natural languages. We broke grammatical categories down into features, and then we broke the values of features down into features, as well. In the process, we moved more and more syntactic information out of the grammar rules and into the lexicon. In effect, we changed our theory of grammar so that the rules give only very general patterns that cut across grammatical categories. Details about which expressions can go with which are specified in lexical entries in terms of valence features.

With the expanded ability of our new feature structure complexes to express cross-categorial generalizations, our four remaining grammar rules cover a wide range of cases. Two of them – the rules introducing complements and specifiers – were discussed extensively in Chapter 4. The third one – a generalization of our old rules introducing PP modifiers to VP and NOM – was illustrated in the previous chapter.[1] The fourth is the Coordination Rule. The formal statements of these rules were given at the end of the previous chapter, along with informal translations (given in italics below the rules).

In addition to our grammar rules, we must provide (as we did in the case of CFGs) some characterization of the 'initial symbol', corresponding to the type of phrases that can stand alone as sentences of the language. We postpone a careful characterization of this until Chapter 8, when we will have introduced a method for distinguishing finite (that is, tensed) clauses from others. For now, we can treat S (which we characterized in terms of features in Chapter 4) as the initial symbol.

We were able to make our grammar rules so general in part because we formulated four general principles about how information must be distributed in well-formed trees: the Head Feature Principle, the Valence Principle, the Semantic Compositionality Principle, and the Semantic Inheritance Principle. These were also reiterated at the end of Chapter 5.

The richer feature structures we are now using, together with our highly schematized rules, have required us to refine our notion of how a grammar is related to the fully

[1] It should be noted that the Head-Modifier Rule does not cover all kinds of modifiers. In particular, some modifiers – such as adjectives inside NPs – precede the heads that they modify. To accommodate such modifiers, we would need an additional grammar rule. This issue was addressed in Problem 1 of Chapter 5.

determinate phrase structure trees of the language. Intuitively, here is how it works:

First, each lexical entry licenses a family of word structures – each of which is a nonbranching tree. More precisely, a lexical entry $\langle w, \Phi \rangle$ licenses any word structure of the form:

$$\begin{array}{c} F \\ | \\ w \end{array}$$

if and only if F is a resolved feature structure that satisfies Φ. A resolved feature structure F satisfies Φ if and only if it assigns values to all features appropriate for feature structures of its type, and those values are consistent with all of the information specified in Φ.

Such lexical trees form the bottom layer of well-formed phrasal trees. They can be combined[2] into larger trees in the ways permitted by the grammar rules, obeying the constraints imposed by our four principles. This process can apply to its own output, making ever larger phrasal trees. So long as the local tree at the top of each tree structure that we construct is licensed by a grammar rule and conforms to these principles, it is well formed. Typically, each node in a well-formed tree will contain some information that was stipulated by a rule and other information that percolated up (metaphorically speaking) from lower nodes (and ultimately from the lexical entries) via the principles. In summary, the relation between our trees and the grammatical mechanisms that license them is as follows: a tree is well-formed if, and only if, it satisfies all of the conditions imposed by the lexical entries of the words it contains, by the grammar rules, and by the general grammatical principles.

We have formulated our theory so that the number of tree structures consistent with a given terminal string will shrink considerably as constraints from higher levels of structure are brought into the picture. This important effect of contextual constraints can be illustrated with the CASE value of proper nouns. Consider the lexical entry in (1):

(1)
$$\left\langle \text{Leslie}, \begin{bmatrix} \textit{word} & \\ \text{SYN} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \textit{noun} & \\ \text{AGR} & \textit{3sing} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle\ \rangle \\ \text{COMPS} & \langle\ \rangle \\ \text{MOD} & \langle\ \rangle \end{bmatrix} \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \text{MODE} & \text{ref} \\ \text{RESTR} & \left\langle \begin{bmatrix} \text{RELN} & \textbf{name} \\ \text{NAME} & \text{Leslie} \\ \text{NAMED} & i \end{bmatrix} \right\rangle \end{bmatrix} \end{bmatrix} \right\rangle$$

[2] Our informal discussion is worded in terms of a process of building trees up from the bottom. This is a conceptually natural way of thinking about it, but it should not be taken too literally. The formal definition of well-formed tree structure that we give below is deliberately nonprocedural.

This lexical entry gives fully specified values for every feature except CASE and GEND. (It may look underspecified for PER and NUM as well, but recall that the type *3sing* is constrained to have specific values for each of those features.) Since the features CASE and GEND are left underspecified in the lexical entry, the lexical entry licenses six distinct word structures. We have shown two in (2):

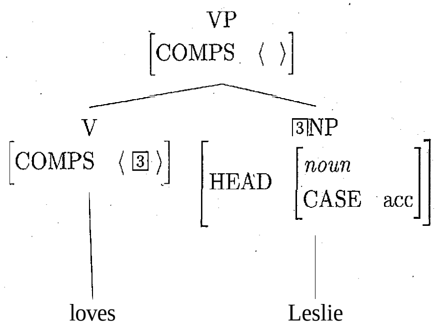
(2) a.
$$\begin{array}{c} \begin{bmatrix} \textit{word} & \\ \text{SYN} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \textit{noun} & \\ \text{CASE} & \text{nom} \\ \text{AGR} & \begin{bmatrix} \textit{3sing} & \\ \text{PER} & \text{3rd} \\ \text{NUM} & \text{sg} \\ \text{GEND} & \text{fem} \end{bmatrix} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle\ \rangle \\ \text{COMPS} & \langle\ \rangle \\ \text{MOD} & \langle\ \rangle \end{bmatrix} \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \text{MODE} & \text{ref} \\ \text{INDEX} & i \\ \text{RESTR} & \left\langle \begin{bmatrix} \text{RELN} & \textbf{name} \\ \text{NAME} & \text{Leslie} \\ \text{NAMED} & i \end{bmatrix} \right\rangle \end{bmatrix} \end{bmatrix} \\ | \\ \text{Leslie} \end{array}$$

b.
$$\begin{array}{c} \begin{bmatrix} \textit{word} & \\ \text{SYN} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \textit{noun} & \\ \text{CASE} & \text{acc} \\ \text{AGR} & \begin{bmatrix} \textit{3sing} & \\ \text{PER} & \text{3rd} \\ \text{NUM} & \text{sg} \\ \text{GEND} & \text{masc} \end{bmatrix} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle\ \rangle \\ \text{COMPS} & \langle\ \rangle \\ \text{MOD} & \langle\ \rangle \end{bmatrix} \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \text{MODE} & \text{ref} \\ \text{INDEX} & i \\ \text{RESTR} & \left\langle \begin{bmatrix} \text{RELN} & \textbf{name} \\ \text{NAME} & \text{Leslie} \\ \text{NAMED} & i \end{bmatrix} \right\rangle \end{bmatrix} \end{bmatrix} \\ | \\ \text{Leslie} \end{array}$$

Notice that we could have abbreviated the mother of these tree structures either as ‘N’ or as ‘NP’, since this is a node of type word whose HEAD value is of type noun with empty SPR and COMPS lists.

Although these two word structures both satisfy the constraints given in the lexical entry equally well, only the tree in (2b) can be embedded within a larger one like (3), licensed by the Head-Complement Rule:

(3)



That is because we have assumed here (following the results of Chapter 4, Problem 6) that the lexical entry for loves specifies that its complement is [CASE acc]. Because the Head-Complement Rule identifies the head daughter's COMPS list with the list of (the feature structures of the) complement daughters, the accusative case specification must be part of the object noun's HEAD value.[3]

The information specified by our rules and lexical entries is thus partial information. Each rule says, in effect, that subtrees of a certain kind are sanctioned, but the rule only specifies some of the constraints that the trees that it licenses must obey. Likewise, a lexical entry says that certain trees dominating the phonological form in that entry are sanctioned, but the entry only specifies some of the information relevant at higher levels of structure. The general principles of our theory constrain the ways in which feature values can be distributed in well-formed phrase structure trees. The job of determining well-formedness can be distributed among the various pieces of our grammatical system because the licensing mechanism requires simultaneous satisfaction of all of the relevant constraints.

In developing our grammar so far, we have arrived at a particular factorization of the information necessary for a precise account of grammatical structure. By far the richest source of information in this factorization is the lexicon. That is, our grammar embodies the claim that both the problem of determining which strings of words constitute well-formed sentences and the problem of specifying the linguistic meaning of sentences depend mostly on the nature of words. Of course, it must also be recognized that there are many regularities about which words go together (and how they go together). The theoretical constructs summarized here capture a number of such regularities; subsequent chapters will provide ways of capturing more.

[3]Nothing in the syntactic context constrains the GEND value, however. The appropriate value there will depend on the non-linguistic context, in particular, on the gender of the person the speaker intends to refer to.

6.2 Examples

6.2.1 A Detailed Example

The best way to understand how the various components of our grammatical theory interact is to work through detailed analyses of linguistic examples. In this subsection, we show in detail how the grammar of English, as we have developed it to this point, handles one simple sentence of English, namely:[4]

(4) They sent us a letter.

We begin our lexical analysis with the entry for the word *letter:*

(5)

$$\left\langle \text{letter}\ , \begin{bmatrix} \textit{word} & \\ \text{SYN} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \textit{noun} & \\ \text{AGR} & \begin{bmatrix} \textit{3sing} & \\ \text{GEND} & \text{neut} \end{bmatrix} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \left\langle \begin{bmatrix} \text{D} & \\ \text{COUNT} & + \\ \text{INDEX} & k \end{bmatrix} \right\rangle \\ \text{COMPS} & \langle\ (\text{PP},)\ \rangle \\ \text{MOD} & \langle\ \rangle \end{bmatrix} \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \text{MODE} & \text{ref} \\ \text{INDEX} & k \\ \text{RESTR} & \left\langle \begin{bmatrix} \text{RELN} & \textbf{letter} \\ \text{INST} & k \\ \text{ADDRESSEE} & m \end{bmatrix} \right\rangle \end{bmatrix} \end{bmatrix} \right\rangle$$

We assume *letter* optionally selects a PP complement, as indicated.

How many word structures satisfy (5)? The answer to this question may be surprising. There are *infinitely many* word structures that satisfy (5). Moreover, this will be true whenever a lexical entry selects something on its COMPS or SPR list, because lexical entries specify such minimal information about the things they select for. For example, in the absence of further constraints; the member of the SPR list in a word structure licensed by (5) could have a RESTR list of any length. Similarly, if the COMPS list in the word structure contains a PP, that PP could have a RESTR value of any length. And this is as it should be, as there is no upper bound on the length of PP complements of this word:

[4]In this section, we present the details of trees over the course of several pages, depicting various subtrees and how they fit together to make larger trees. In doing this, we use tags to mark identity across distinct diagrams of trees that will eventually be put together into a single tree. We also reuse tags across different trees when the same lexical entry is used in different sentences. Strictly speaking, tags only mark identity within a given description. We are taking this liberty with the tag notation only in this section, because it is a convenient heuristic.

(6) a. the letter to Kim...
b. the letter to Kim and Sandy...
c. the letter to Kim, Lee and Sandy...
d. the letter to the person who signed the document that started the mishap that...

That is, depending on the surrounding context (i.e. depending on which words the PP actually contains), the PP's RESTR list might have one, three, thirty-seven, or two hundred predications on it. The same is true of the specifier, as the examples in (7) indicate:

(7) a. the letter...
b. almost every letter...
c. Sandy's friend's mother's letter...
d. the cricket club's former secretary's letter...

If we assume the analysis of quantificational determiners sketched at the end of Chapter 5, then the word structure for *letter* that is relevant to the sentence in (4), however, has a SPR value whose RESTR is singleton:

(8)
$$
\begin{bmatrix} \textit{word} & \\ \text{SYN} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \textit{noun} & \\ \text{AGR} & \boxed{1}\begin{bmatrix}\textit{3sing} & \\ \text{PER} & \text{3rd}\\ \text{NUM} & \text{sg} \\ \text{GEND} & \text{neut}\end{bmatrix} \\ \text{CASE} & \text{acc}\end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \left\langle \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix}\textit{det} & \\ \text{AGR} & \boxed{1} \\ \text{COUNT} & + \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle\,\rangle \\ \text{COMPS} & \langle\,\rangle \\ \text{MOD} & \langle\,\rangle \end{bmatrix}\end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \text{MODE} & \text{none} \\ \text{INDEX} & k \\ \text{RESTR} & \left\langle \begin{bmatrix} \text{RELN} & \textbf{exist} \\ \text{BV} & k\end{bmatrix}\right\rangle \end{bmatrix}\end{bmatrix}\right\rangle \\ \text{COMPS} & \langle\,\rangle \\ \text{MOD} & \langle\,\rangle \end{bmatrix}\end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \text{MODE} & \text{ref} \\ \text{INDEX} & k \\ \text{RESTR} & \left\langle \begin{bmatrix} \text{RELN} & \textbf{letter} \\ \text{INST} & k \\ \text{ADDRESSEE} & \text{m} \end{bmatrix}\right\rangle \end{bmatrix}\end{bmatrix}
$$

As for the COMPS value, the empty list option has been exercised in this tree, as the sentence whose structure we are building contains no PP complement. Notice that, with no PP, there is no constituent that will realize the ADDRESSEE role. Since we have not imposed any constraint requiring that semantic roles be realized syntactically, this does not present any technical problem. And having an ADDRESSEE role for the noun *letter*, even when no addressee is mentioned, seems quite intuitive. Finally, note that (8) obeys the Specifier-Head Agreement Constraint, which identifies the AGR value of the noun with that of the element on its SPR list.

The word structure for the word a is abbreviated in (9):[5]

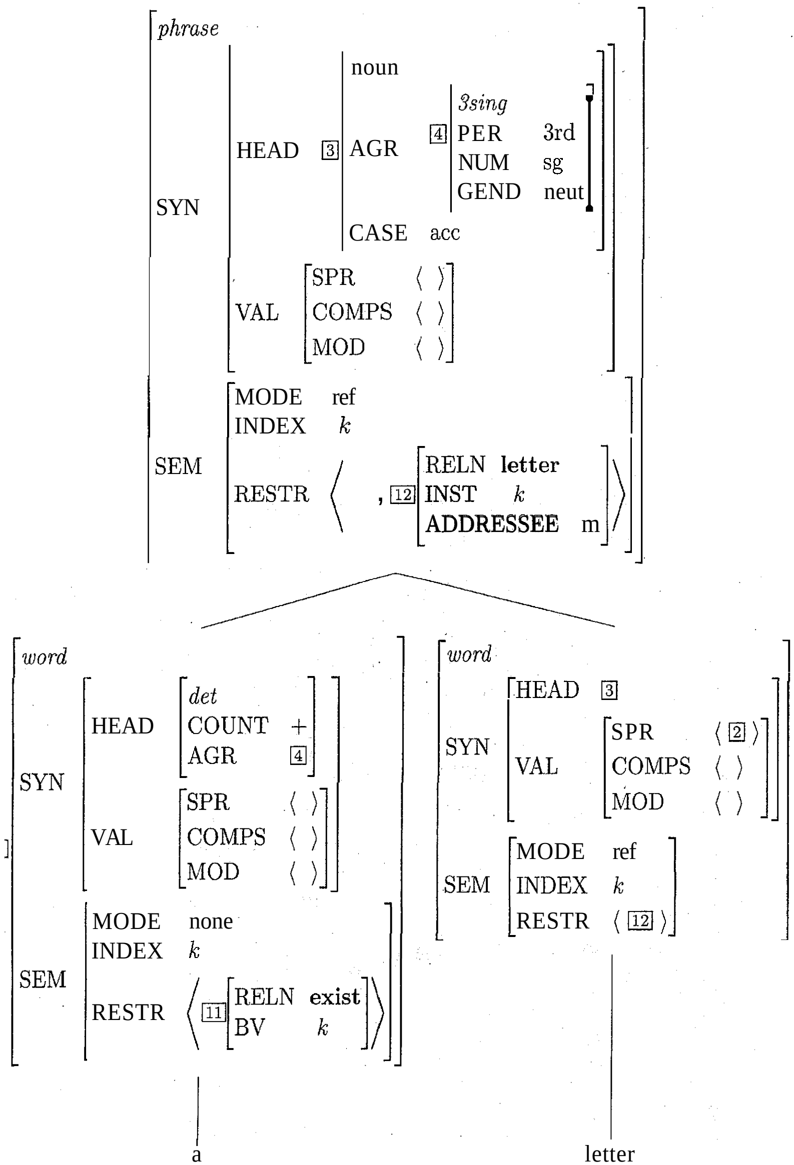
(9)
$$
\begin{bmatrix} \textit{word} & \\ \text{SYN} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \textit{det} & \\ \text{AGR} & \begin{bmatrix}\textit{3sing} & \\ \text{PER} & \text{3rd}\\ \text{NUM} & \text{sg} \\ \text{GEND} & \text{neut}\end{bmatrix} \\ \text{COUNT} & + \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle\,\rangle \\ \text{MODPS} & \langle\,\rangle \\ \text{MOD} & \langle\,\rangle \end{bmatrix}\end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \text{MODE} & \text{none} \\ \text{INDEX} & \text{k} \\ \text{RESTR} & \left\langle \begin{bmatrix} \text{RELN} & \textbf{exist} \\ \text{BV} & k\end{bmatrix}\right\rangle \end{bmatrix}\end{bmatrix}
$$

a

[5]What is not shown in this tree is the complete feature specification for the **exist** predication. See Section 5.8 of Chapter 5 for discussion.

The following tree results from combining (8) and (9) via the Head-Specifier Rule:

(10)



In this tree, the left subtree is exactly the one shown in (9). The identification of the element on the head daughter's SPR list ([2]) and the feature structure of the left daughter is guaranteed by the Head-Specifier Rule, which licenses the combination of this determiner with this noun. When the Head-Specifier Rule enforces this identity, it forms a link in a chain of identities: the lexical entry for letter identifies the INDEX of the element

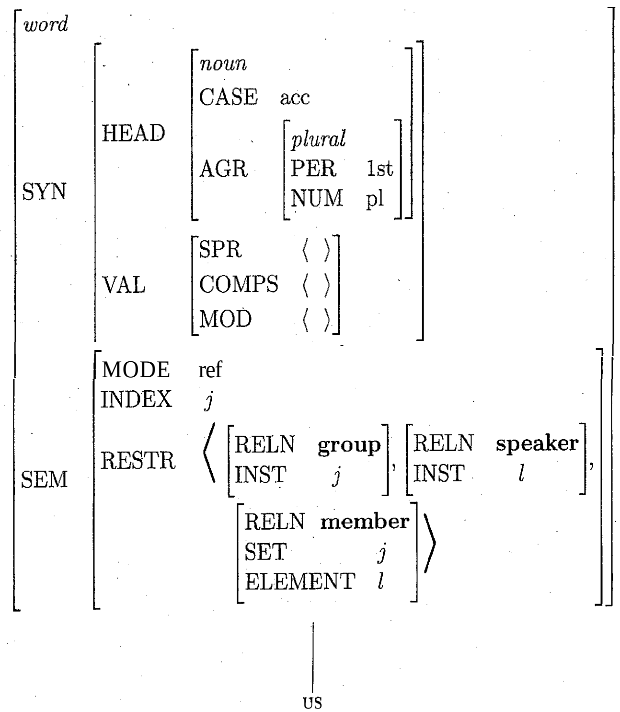
on its SPR list with its own INDEX and INST values. The lexical entry for a identifies its INDEX with its BV value. When these two words combine via the Head-Specifier Rule, the INDEX of the specifier of *letter* and the INDEX of *a* are identified. This chain of identities ensures that the BV of the **exist** predication and the INST of the **letter** predication are one and the same (k).

(10) obeys the HFP: the HEAD value of the head daughter is identified with that of the mother ([3]). And it obeys the Valence Principle: the COMPS value of the phrase is the same as that of the head daughter (the empty list). The mother's SPR value is the empty list, **as** required by the Head-Specifier Rule.

The Semantic Inheritance Principle says that the MODE and INDEX values of the head daughter must be shared by the mother, which is the case in (10). And the Semantic Compositionality Principle requires that the mother's RESTR value be the sum of the two daughters' RESTR lists. This concludes the analysis of the noun phrase a *letter*, as it appears in the sentence in (4).

The lexical entry for the pronoun us is quite straightforward, except for the RESTR list in the semantics. In the following, we have chosen to characterize the meaning of us roughly as reference to a group of which the speaker is a member. We have formalized this as a RESTR list with three elements, but there are many other possible ways of doing this. Our version gives rise to the following lexical tree:

(11)



All this information is lexically specified. Note that because the AGR value is of type *plural*, it contains no GEND specification.

Now consider the lexical entry for the word *sent*:[6]

(12)
$$\left\langle \text{sent}\ ,\ \begin{bmatrix} \textit{word} & \\ \text{SYN} & \begin{bmatrix} \text{HEAD} & \textit{verb} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \left\langle \begin{array}{c} \text{NP}_i \\ \begin{bmatrix}\text{CASE} & \text{nom}\end{bmatrix} \end{array} \right\rangle \\ \text{COMPS} & \left\langle \begin{array}{c} \text{NP}_j \\ \begin{bmatrix}\text{CASE} & \text{acc}\end{bmatrix} \end{array} , \begin{array}{c} \text{NP}_k \\ \begin{bmatrix}\text{CASE} & \text{acc}\end{bmatrix} \end{array} \right\rangle \\ \text{MOD} & \langle\ \rangle \end{bmatrix} \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \text{MODE} & \text{prop} \\ \text{INDEX} & s_7 \\ \text{RESTR} & \left\langle \begin{bmatrix} \text{RELN} & \textbf{send} \\ & s_7 \\ & i \\ \text{SENDEE} & j \\ \text{SENT} & k \end{bmatrix} \right\rangle \end{bmatrix} \end{bmatrix} \right\rangle$$

Note that; as a past tense form, this lexical entry has an underspecified AGR value. All of the word structures licensed by (12), however, have fully resolved AGR values, and by the SHAC, must share those AGR values with their specifiers. Similarly, although the lexical entry in (12) places no restrictions on the AGR value of the complements, those AGR values are fully specified in the word structures. The word structure for sent that is relevant to the sentence in (4) is shown in (13):[7]

[6] We are ignoring the semantic contribution of the past tense in this discussion.

[7] Although the tree in (13) represents a fully resolved word structure, we have abbreviated somewhat. In particular, we have not shown the SEM values within the elements of the SPR and COMPS lists. Similar remarks apply to many of the trees in the remainder of this chapter.

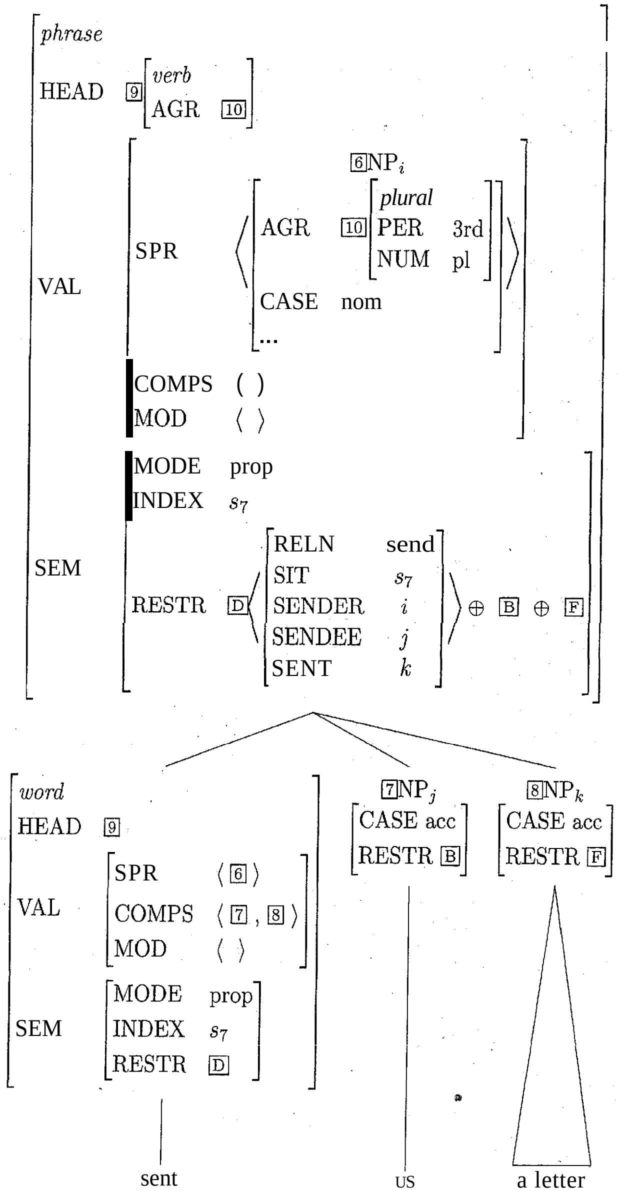
(13)
$$\begin{bmatrix} \textit{word} & \\ \text{SYN} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \textit{verb} \\ \text{AGR} & \boxed{10} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \left\langle \begin{array}{c} \text{NP}_i \\ \begin{bmatrix} \text{CASE} & \text{nom} \\ \text{AGR} & \boxed{10} \begin{bmatrix} \textit{plural} \\ \text{PER} & \text{3rd} \\ \text{NUM} & \text{pl} \end{bmatrix} \\ \ldots \end{bmatrix} \end{array} \right\rangle \\ \text{COMPS} & \left\langle \begin{array}{c} \text{NP}_j \\ \begin{bmatrix} \text{CASE} & \text{acc} \\ \text{AGR} & \begin{bmatrix} \textit{plural} \\ \text{PER} & \text{1st} \\ \text{NUM} & \text{pl} \end{bmatrix} \\ \ldots \end{bmatrix} \end{array} , \begin{array}{c} \text{NP}_k \\ \begin{bmatrix} \text{CASE} & \text{acc} \\ \text{AGR} & \begin{bmatrix} \textit{3sing} \\ \text{PER} & \text{3rd} \\ \text{NUM} & \text{sg} \\ \text{GEND} & \text{neut} \end{bmatrix} \\ \ldots \end{bmatrix} \end{array} \right\rangle \\ \text{MOD} & \langle\ \rangle \end{bmatrix} \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \text{MODE} & \text{prop} \\ \text{INDEX} & s_7 \\ \text{RESTR} & \left\langle \begin{bmatrix} \text{RELN} & \textbf{send} \\ \text{SIT} & s_7 \\ \text{SENDER} & i \\ \text{SENDEE} & j \\ \text{SENT} & k \end{bmatrix} \right\rangle \end{bmatrix} \end{bmatrix}$$

$$|$$

sent

The three trees we have now built up combine via the Head-Complement Rule to give the following tree structure:

(14)



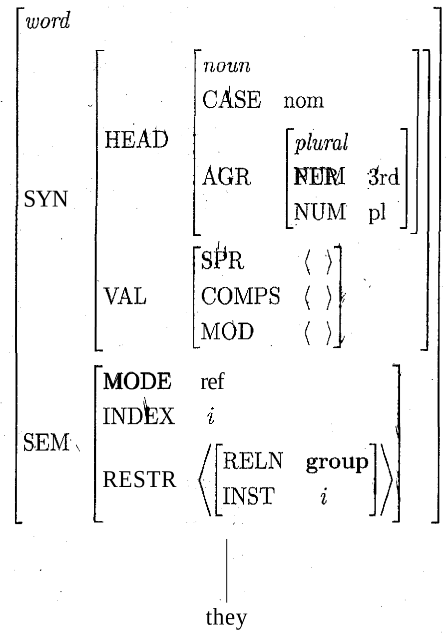
We have done a bit more abbreviating here. The node tagged $\boxed{7}$ is identical to the top node of the word structure in (11). Likewise, the node tagged $\boxed{8}$ is identical to the top node in (10).

The [CASE acc] constraints on both these NPs comes from the COMPS value of the lexical entry for *sent* (see (12)), and hence appears on this node, as required by the Head-Complement Rule. The RESTR values in the semantics for the two NP nodes are the ones shown in (11) and (10). We abbreviated these with the tags $\boxed{B}$ and $\boxed{F}$, respectively.

(14) obeys the conditions on COMPS values specified in the Head-Complement Rule, that is, the head daughter's complements are identified with the non-head daughters and the mother's COMPS value is empty. (14) obeys the Valence Principle, as the SPR value of the head daughter, not mentioned in the rule, is preserved as the mother's SPR value. Likewise, the HEAD value of mother and head daughter are correctly identified here, in accordance with the Head Feature Principle. Finally, the MODE and INDEX values of mother are those of the head daughter, while the RESTR value of the mother is the sum of those of all the daughters, as specified by the semantic principles.

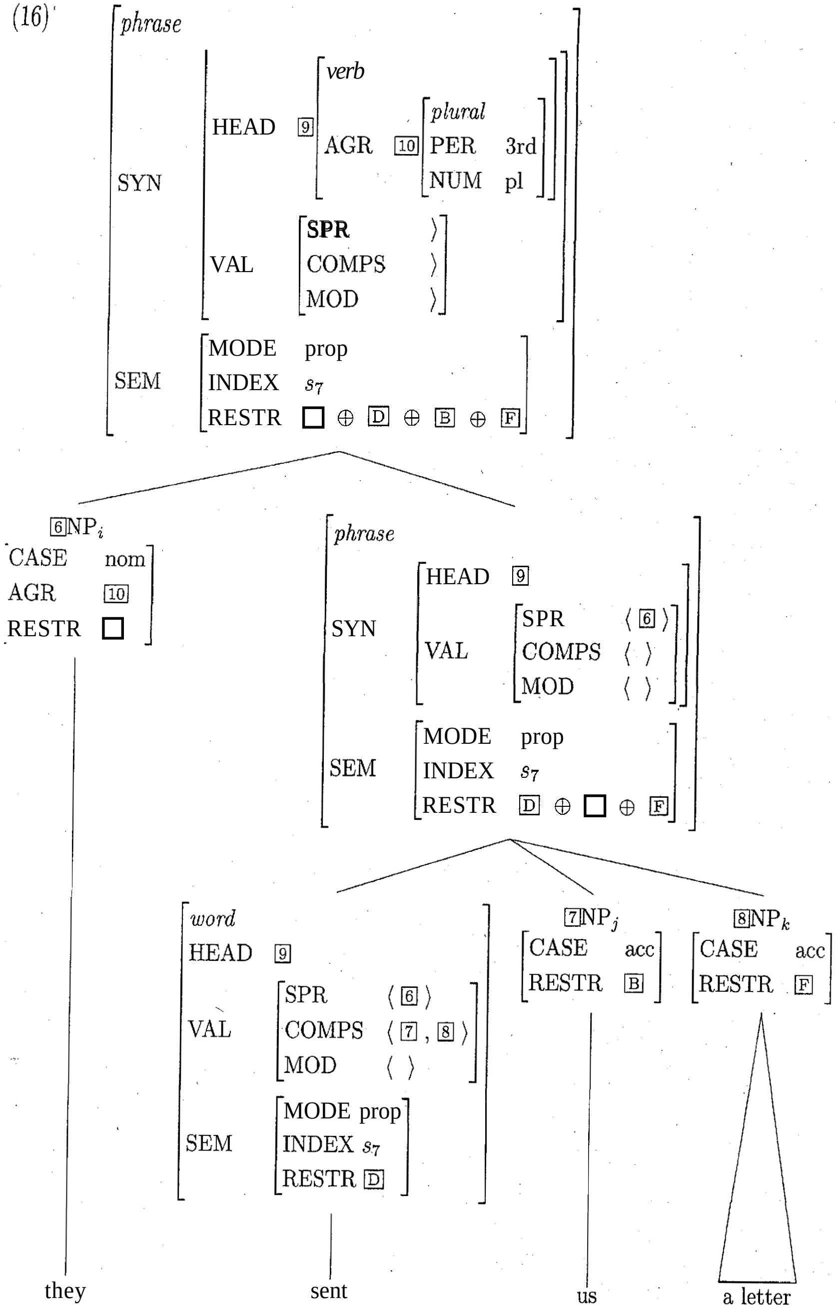
The last step is to combine the VP in (14) with the tree structure for its subject NP. The following is the word structure for the pronoun *they*, as licensed by an appropriate lexical entry:

(15)



The result is the tree in (16):

(16)



Again, we have abbreviated. The node labeled 6 is just the top node in (15). The nodes labeled 7 and 8 are exactly as they were in (14), as is the VP node. We have abbreviated the RESTR values, simply putting in tags or sums of tags. The RESTR value of the top node, fully spelled out (except for the somewhat abbreviated contribution of the word *a*), is the list consisting of the following seven predications (in the indicated order):

(17)
[RELN **group**, INST i] , [RELN **send**, SIT s_7, SENDER i, SENDEE j, SENT k] , [RELN **group**, INST j] ,

[RELN **speaker**, INST l] , [RELN **member**, **SET** j, **ELEMENT** 1] , [RELN **exist**, BV k] ,

[RELN letter, INST k, ADDRESSEE m]

The AGR value in the top node of (16) is identical to that in the subject NP, as required by the interaction of the HFP, the Head-Specifier Rule, and the SHAC. In general, this tree structure obeys the Head Feature Principle, the Valence Principle, and the two semantic principles.

This concludes our analysis of the sentence *They sent* us *a letter*. The various constraints in our grammar interact to ensure that this structure and infinitely many related to it are well-formed, while guaranteeing that infinitely many other structures similar to it are ill-formed.

Exercise 1: The Non-infinity of Us

The lexical entry for *letter* licenses infinitely many word structures, while the lexical entry for us licenses exactly one. What feature specifications in the lexical entries are behind this difference?

6.2.2 Another Example

The detailed analysis we just went through built the sentence from the bottom up. This is one way to use the grammatical machinery we have developed, but it is not the only way. We could equally well have started with at the top of the tree, showing how our rules, principles, and lexical entries interact to license all its parts.

To see this top-down approach in action, consider the following sentence[8]

(18) We send two letters to Lee.

[8]This example sounds a bit odd in isolation, but it would be perfectly natural in the appropriate context, for example, in response to the question, *What do we do if Alex writes to us?*

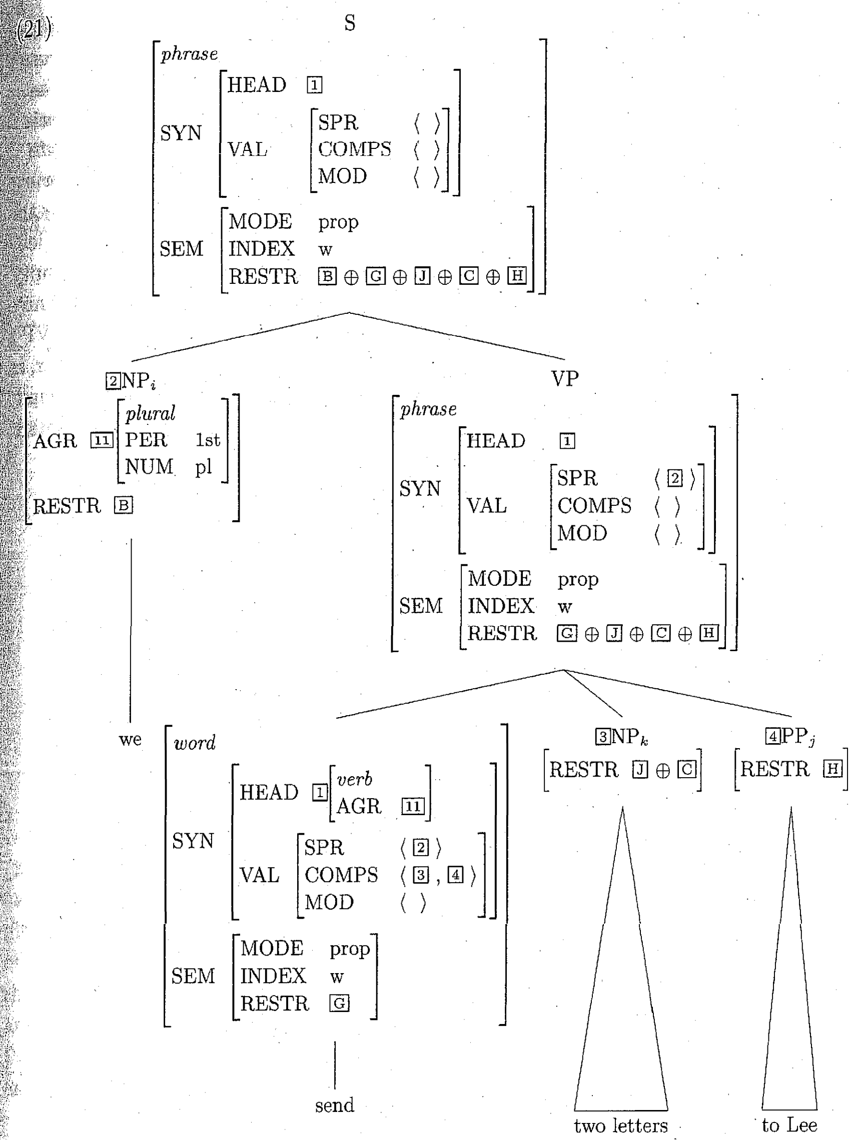
Example (18) is structurally ambiguous in a way analogous to the familiar example, *I saw the astronomer with a telescope.* That is, the PP *to Lee* can *be* attached either to the VP or to the NP headed by letters. In our semantic representation, the two readings correspond to two different RESTR lists, shown in (19) and (20):

(19)
$$\begin{bmatrix}\text{RELN} & \textbf{group}\\ \text{INST} & i\end{bmatrix}, \begin{bmatrix}\text{RELN} & \textbf{speaker}\\ \text{INST} & l\end{bmatrix}, \begin{bmatrix}\text{RELN} & \text{member}\\ \text{SET} & i\\ \text{ELEMENT} & l\end{bmatrix},$$
$$\begin{bmatrix}\text{RELN} & \text{send}\\ \text{SIT} & s_7\\ \text{SENDER} & i\\ \text{SENDEE} & \text{j}\\ \text{SENT} & k\end{bmatrix}, \begin{bmatrix}\text{RELN} & \text{two}\\ \text{BV} & k\end{bmatrix}, \begin{bmatrix}\text{RELN} & \text{letter}\\ \text{INST} & k\\ \text{ADDRESSEE} & \text{m}\end{bmatrix},$$
$$\begin{bmatrix}\text{RELN} & \textbf{name}\\ \text{NAME} & \text{Lee}\\ \text{NAMED} & j\end{bmatrix}$$

(20)
$$\begin{bmatrix}\text{RELN} & \textbf{group}\\ \text{INST} & i\end{bmatrix}, \begin{bmatrix}\text{RELN} & \textbf{speaker}\\ \text{INST} & l\end{bmatrix}, \begin{bmatrix}\text{RELN} & \text{member}\\ \text{SET} & i\\ \text{ELEMENT} & 1\end{bmatrix},$$
$$\begin{bmatrix}\text{RELN} & \textbf{send}\\ \text{SIT} & s_7\\ \text{SENDER} & \text{i}\\ \text{SENDEE} & \textbf{j}\\ \text{SENT} & k\end{bmatrix}, \begin{bmatrix}\text{RELN} & \textbf{two}\\ \text{BV} & k\end{bmatrix}, \begin{bmatrix}\text{RELN} & \text{letter}\\ \text{INST} & k\\ \text{ADDRESSEE} & m\end{bmatrix},$$
$$\begin{bmatrix}\text{RELN} & \text{name}\\ \text{NAME} & \text{Lee}\\ \text{NAMED} & m\end{bmatrix}$$

The only difference between the two semantic representations is which other role the NAMED value of the name predication (i.e. Lee) is identified with: the SENDEE value of the send predication or the ADDRESSEE value of the letter predication.

In this subsection, we will show how our grammar licenses two distinct trees for this sentence, and how it associates each with one of the semantic representations in (19) and (20). For expository convenience, we begin with the rather schematic tree in (21) (similar to (16)), waiting to show the detailed feature structures it contains until we look at its subtrees:

(21)


The top node in this tree is licensed by the Head-Specifier Rule. It differs from its second daughter, the VP, in only two ways: its SPR value is the empty list (as required by the Head-Specifier Rule), and its RESTR value includes the RESTR of the subject NP (as required by the Semantic Compositionality Principle). The HEAD features of the top node and of the VP are identical, as required by the Head Feature Principle. The COMPS list is empty both at the top and in the VP, in accordance with the Valence Principle. And both MODE and INDEX have the same value at the top as in the VP, in keeping with the Semantic Inheritance Principle. The first daughter (the subject NP) is identical to the sole element on the second daughter's SPR list, as required by the Head-Specifier Rule.

The subtree dominating we – that is the subject of the sentence – is labeled 'NP' here, but it could just as well have been labeled 'N'. It is simply a word structure, identical in its feature structure to the one in (11), except that the value of the CASE feature is 'nom', not 'acc'. This structure is the word structure licensed by the lexical entry for we.

The other daughter of the top node – the VP – is the mother of a tree licensed by the Head-Complement Rule. The VP's feature values are the same as those of its head (leftmost) daughter, except for COMPS and RESTR. The COMPS list of the VP is empty, as specified in the Head-Complement Rule. The RESTR value is the sum of its three daughters' RESTR values, by the Semantic Compositionality Principle. Again, the VP's HEAD, SPR, MODE, and INDEX values are the same as those of the head daughter, in accordance with the HFP, the Valence Principle, and the Semantic Inheritance Principle. The COMPS value of the head daughter is the list consisting of the other two daughters; this is specified by the Head-Complement Rule.

The subtree dominating the verb send is the following:

(22)

$$
\left[\begin{array}{ll}
\multicolumn{2}{l}{\textit{word}}\\
\text{SYN} & \left[\begin{array}{ll}
\text{HEAD} & \left[\begin{array}{ll}\multicolumn{2}{l}{\textit{verb}}\\ \text{AGR} & \boxed{11}\end{array}\right]\\
\text{VAL} & \left[\begin{array}{ll}
\text{SPR} & \left\langle \begin{array}{c}\text{NP}_i\\ \left[\begin{array}{ll}\text{CASE} & \text{nom}\\ \text{AGR} & \boxed{11}\left[\begin{array}{ll}\multicolumn{2}{l}{\textit{plural}}\\ \text{PER} & \text{1st}\\ \text{NUM} & \text{pl}\end{array}\right]\\ \ldots & \end{array}\right]\end{array}\right\rangle\\
\text{COMPS} & \left\langle \begin{array}{c}\text{NP}_k\\ \left[\begin{array}{ll}\text{CASE} & \text{acc}\\ \text{AGR} & \left[\begin{array}{ll}\multicolumn{2}{l}{\textit{plural}}\\ \text{PER} & \text{3rd}\\ \text{NUM} & \text{pl}\end{array}\right]\\ \ldots & \end{array}\right]\end{array}, \text{PP}_j \right\rangle\\
\text{MOD} & \langle\ \rangle
\end{array}\right]
\end{array}\right]\\
\text{SEM} & \left[\begin{array}{ll}
\text{MODE} & \text{prop}\\
\text{INDEX} & s_7\\
\text{RESTR} & \boxed{G}\left\langle\left[\begin{array}{ll}\text{RELN} & \text{send}\\ \text{SIT} & s_7\\ \text{SENDER} & i\\ \text{SENDEE} & j\\ \text{SENT} & k\end{array}\right]\right\rangle
\end{array}\right]
\end{array}\right]
$$

send

This is different from the verb subtree in our previous example (i.e. from (13)) in several ways. The most obvious is that the form is *send*, not *sent*. Although our SEM value does not reflect the clear meaning difference between the present and past tense forms, there are nonetheless several syntactic differences that are represented. Many of these differences follow from differences in the lexical entries that license the word structures. (22) is licensed by the lexical entry in (23):

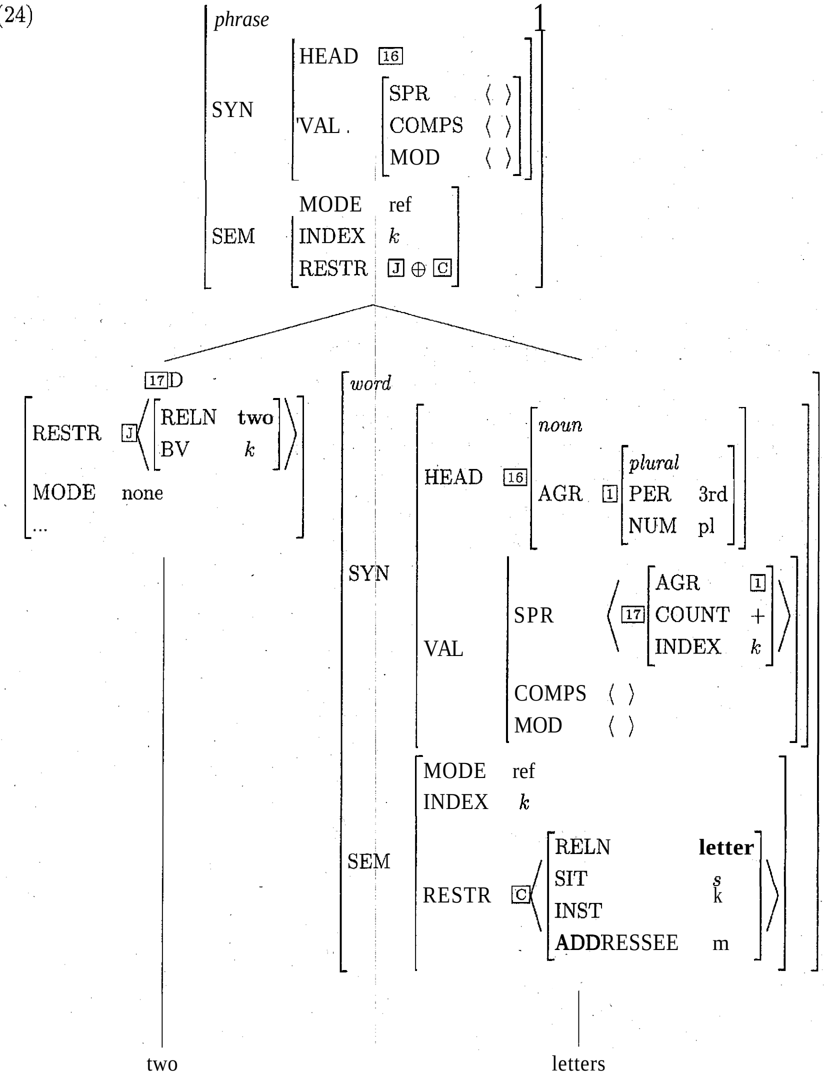
(23)

$$
\left\langle \text{send}\ ,\ \left[\begin{array}{ll}
\multicolumn{2}{l}{\textit{word}}\\
\text{SYN} & \left[\begin{array}{ll}
\text{HEAD} & \textit{verb}\\
\text{VAL} & \left[\begin{array}{ll}
\text{SPR} & \left\langle \begin{array}{c}\text{NP}_i\\ \left[\begin{array}{ll}\text{CASE} & \text{nom}\\ \text{AGR} & \textit{non-3sing}\end{array}\right]\end{array}\right\rangle\\
\text{COMPS} & \left\langle \begin{array}{c}\text{NP}_k\\ \left[\text{CASE}\ \ \text{acc}\right]\end{array} (, \text{PP}_j) \right\rangle\\
\text{MOD} & \langle\ \rangle
\end{array}\right]
\end{array}\right]\\
\text{SEM} & \left[\begin{array}{ll}
\text{MODE} & \text{prop}\\
\text{INDEX} & s_7\\
\text{RESTR} & \left\langle\left[\begin{array}{ll}\text{RELN} & \textbf{send}\\ \text{SIT} & s_7\\ \text{SENDER} & i\\ \text{SENDEE} & j\\ \text{SENT} & k\end{array}\right]\right\rangle
\end{array}\right]
\end{array}\right] \right\rangle
$$

(23)'s specifier is specified as [AGR *non-3sing*]; that is because the verb *send* (unlike *sent*) cannot be combined with a third-person singular subject (like *Terry*). Another difference is that the second element of the COMPS list in (22) is an optional PP, not an obligatory NP. Related to that is the fact that the first complement in (22) refers to the thing sent (indicated by the role 'SENT' in the predication on the verb's RESTR list), and the second complement corresponds to the sendee (also indicated in the RESTR). Problem 3 in Chapter 10 addresses the relation between pairs of lexical entries like (12) and (23).

The subtree for the object NP, two letters, is shown in (24):

(24)

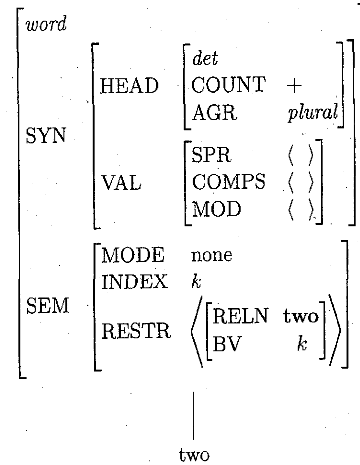


This tree is licensed by the Head-Specifier Rule, which says that the top node must have an empty SPR list and that the second (i.e. head) daughter must have a SPR list whose sole member is identical to the first daughter. The identity of the AGR values of the head noun letters and its determiner two (indicated by $\boxed{1}$) is required by the SHAC. The HEAD value of the top node is identical to that of the second daughter, according to the Head Feature Principle. The COMPS values of these two nodes are identical, as guaranteed by the Valence Principle. The MODE and INDEX values of the second daughter and its mother are likewise shared, courtesy of the Semantic Inheritance

Principle. Finally, the Semantic Compositionality Principle requires that the RESTR value of the determiner combines with the RESTR value for the noun to give the RESTR value of the NP.

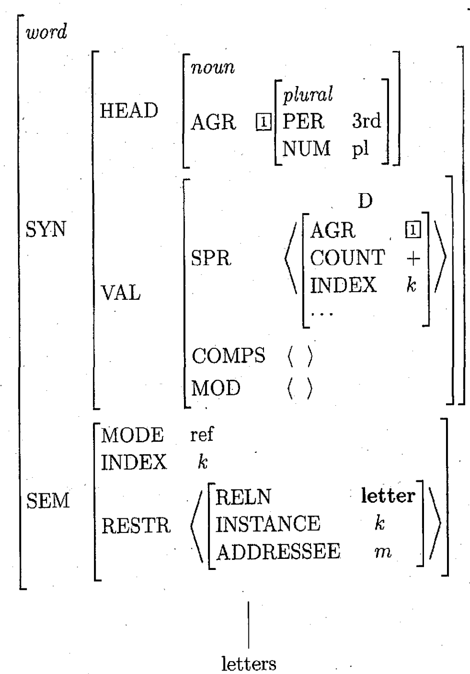
Licensing (24) via the Head-Specifier Rule requires the word structures for each of its words. The following is the word structure for *two*, which is similar to (9) above:

(25)



The relevant word structure for *letters* is sketched in (26):

(26)



This tree is quite similar to (8). The principle difference is that the type of the AGR value is plural, and it therefore lacks the GEND feature. If our treatment of semantics were more detailed, the RESTR value would also be different, since it would have to include some information about the meaning of the plurality of letters; but for present purposes, we will ignore that difference. This word structure is licensed by the entry for letters, shown in (27):

(27)
$$\left\langle \text{letters}, \begin{bmatrix} \textit{word} & \\ \text{SYN} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \textit{noun} & \\ \text{AGR} & \textit{plural} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \left\langle \overset{\text{D}}{\begin{bmatrix} \text{COUNT} & + \\ \text{INDEX} & k \end{bmatrix}} \right\rangle \\ \text{COMPS} & \langle (\text{PP}_m) \rangle \\ \text{MOD} & \langle\ \rangle \end{bmatrix} \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \text{MODE} & \text{ref} \\ \text{INDEX} & k \\ \text{RESTR} & \left\langle \begin{bmatrix} \textbf{RELN} & \textbf{letter} \\ \text{INST} & k \\ \textbf{ADD}\text{RESSEE} & \textbf{m} \end{bmatrix} \right\rangle \end{bmatrix} \end{bmatrix} \right\rangle$$

Notice that this lexical entry, like the one for *letter* in (5), provides for a possible PP complement. The word structure in (26) above uses the empty COMPS list option. We will return to the PP complement possibility below.

The subtree for the PP, to Lee, is highly schematized in (21). A more detailed version of the tree is given in (28):[9]

[9]As with the proper noun Leslie discussed in Section 6.1 above, the lexical entry for Lee is underspecified for GEND. All of the word structures that satisfy that lexical entry are fully specified, and therefore contain a value for GEND. Here we have arbitrarily chosen a word structure that is [GEND fem].

(28)
$$\begin{bmatrix} \textit{phrase} & \\ \text{SYN} & \begin{bmatrix} \text{HEAD} & \boxed{6} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle\ \rangle \\ \text{COMPS} & \langle\ \rangle \\ \text{MOD} & \langle\ \rangle \end{bmatrix} \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \text{INDEX} & j \\ \text{RESTR} & \boxed{H} \end{bmatrix} \end{bmatrix}$$

$$\begin{bmatrix} \textit{word} & \\ \text{SYN} & \begin{bmatrix} \text{HEAD} & \boxed{6} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle\ \rangle \\ \text{COMPS} & \langle \boxed{7} \rangle \\ \text{MOD} & \langle\ \rangle \end{bmatrix} \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \text{INDEX} & j \\ \text{RESTR} & \langle\ \rangle \end{bmatrix} \end{bmatrix} \qquad \boxed{7}\text{NP}\begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{CASE} & \text{acc} \\ \text{AGR} & \begin{bmatrix} \textit{3sing} & \\ \text{PER} & \text{3rd} \\ \text{NUM} & \text{sg} \\ \text{GEND} & \text{fem} \end{bmatrix} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle\ \rangle \\ \text{COMPS} & \langle\ \rangle \\ \text{MOD} & \langle\ \rangle \end{bmatrix} \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \text{MODE} & \text{ref} \\ \text{INDEX} & j \\ \text{RESTR} & \boxed{H}\left\langle \begin{bmatrix} \text{RELN} & \textbf{name} \\ \text{NAME} & \text{Lee} \\ \text{NAMED} & j \end{bmatrix} \right\rangle \end{bmatrix} \end{bmatrix}$$

to   Lee

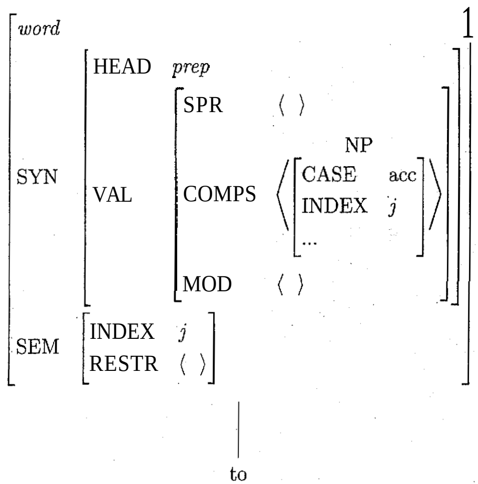
The most interesting thing about this subtree is how we have analyzed the semantics. The preposition *to* in this sentence is functioning to mark the role of its object NP with respect to the verb. That is, it does what many languages would do by means of case inflections on the noun. Since English has only a vestigial system of case marking, it relies on prepositions and word order to mark the roles of various NPs in the sentence. Note that the preposition can be omitted if the verb's arguments are presented in another order: *We sent Lee two letters.* Consequently, we have given the preposition no semantics of its own. Its RESTR value is the empty list, and its index is simply identified as the index of the object NP. We have said nothing about the MODE value, but in the next chapter, we will argue that it, too, should be identified with the MODE of the object NP.

The PP assumes the same INDEX value as the preposition (and hence as the NP) by the Semantic Inheritance Principle. Other identities in (28) should by now be familiar:

the one element of the preposition's COMPS list must be the object NP, by the Head-Complement Rule; the same rule specifies that the PP has an empty COMPS list; the Valence Principle is responsible for the fact that the PP and P have the same (empty) SPR list; the PP and the P share the same HEAD features in virtue of the Head Feature Principle; and the PP's RESTR value is the same as the NP's, in accordance with the Semantic Compositionality Principle (together with the fact that the preposition's RESTR is the empty list).

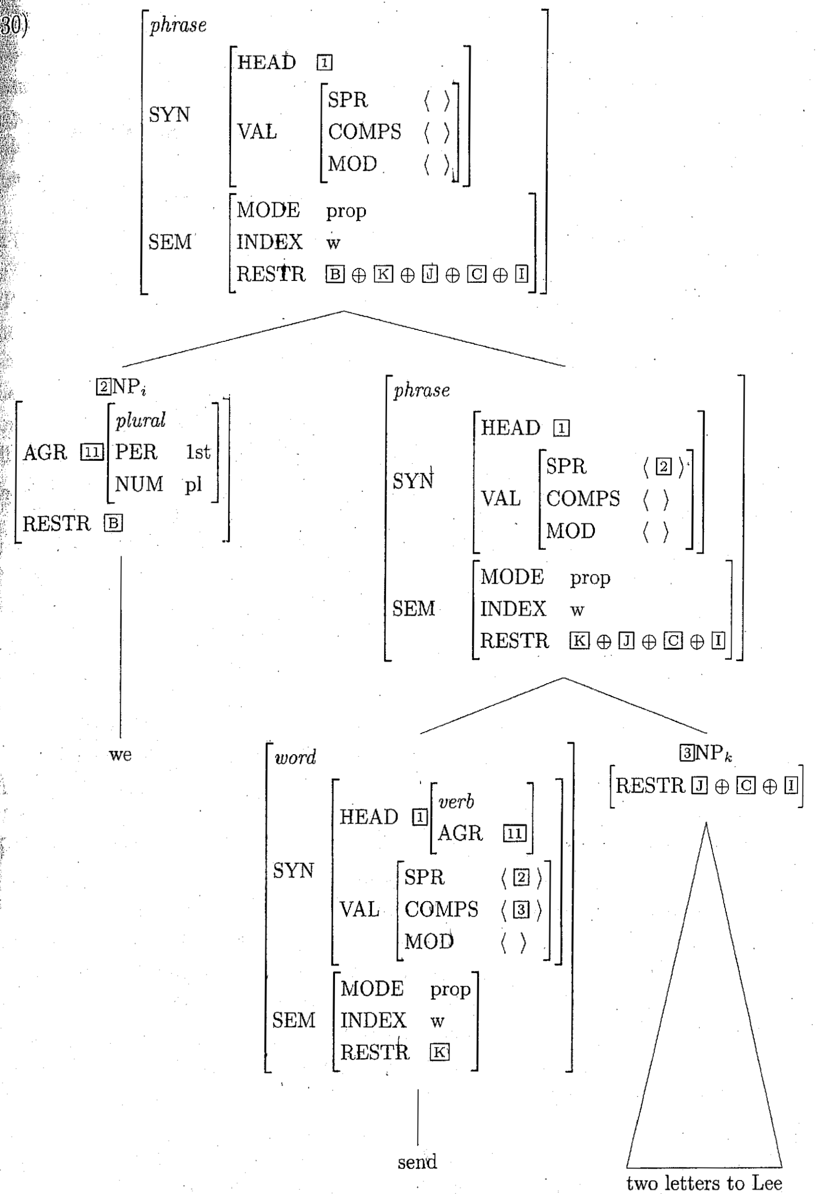
The NP in (28) is [CASE acc] because objects of prepositions in English are always accusative (although there is no morphological marking of it in this sentence). This requirement is encoded in the lexical entry for the preposition, as we will see when we look at the word structure for *to*, which is shown in (29):

(29)



This completes the analysis of one parse of We send **two** *letters to Lee*. A schematic tree for the other parse is given as (30):

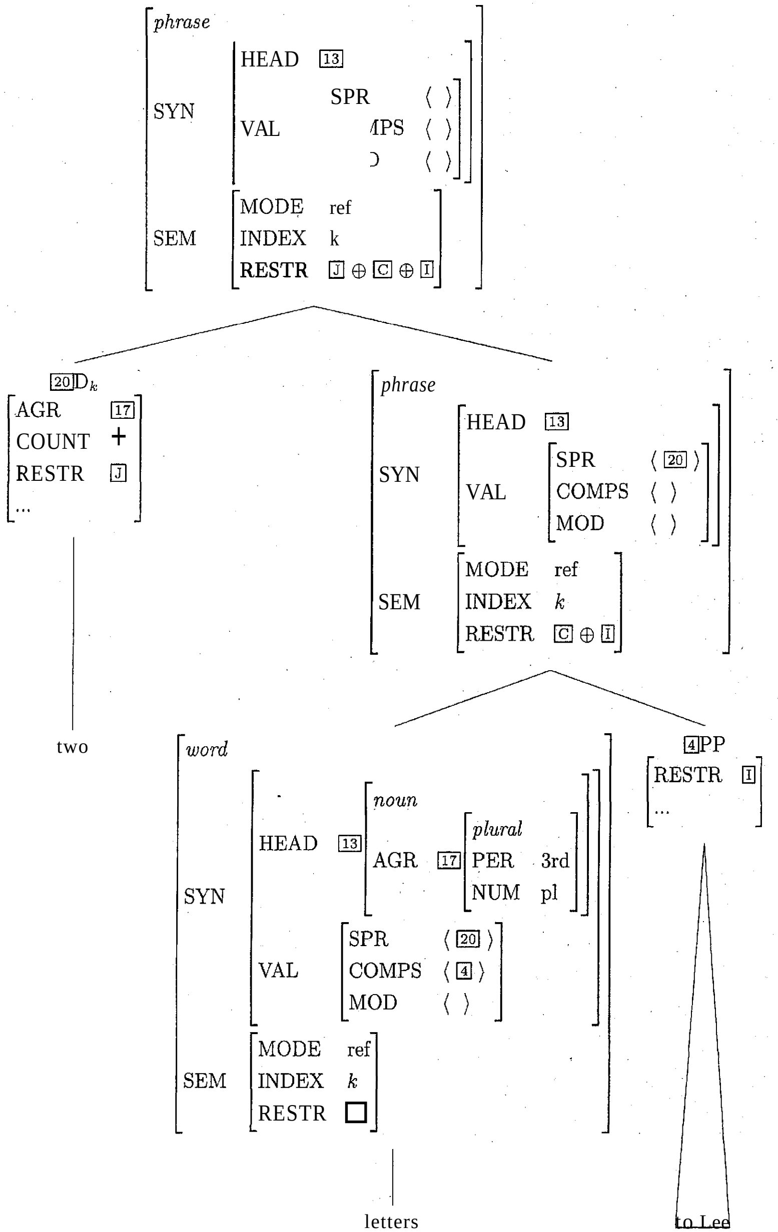
(30)



The subject NP *we* and the PP *to Lee* are exactly the same in this structure as in (21). The verb *send*, however, has two complements in (21) and only one in (30). That is because the lexical entry in (23) above, which licenses both verbal word structures, specifies that its second (PP) complement is optional. The noun *letters* in the two examples is licensed by the same lexical entry (27), which takes an optional PP complement.

In (21), there was no node spanning the string *two letters to* Lee. In (30), however, there is such a node. A more detailed subtree for that NP is the following:

(31)



The top node in this subtree is licensed by the Head-Specifier Rule which requires the identity of the determiner with the one element on the head's SPR list. The second daughter, dominating letters to Lee is licensed by the Head-Complement Rule, which also requires that the element on the COMPS list of the head noun is identical to the PP complement. The other identities are enforced by various principles in ways that should now be familiar.

Notice that the tag on the RESTR of to Lee in (30) and (31) is different from the tag in (21). That is because the role played by Lee is subtly different in the two sentences. In (30), the SENDEE role does not correspond to any syntactic constituent; in (21), the PP to Lee (and the noun Lee, with which it is coindexed) plays the SENDEE role. On the other hand, in (30), the PP plays the ADDRESSEE role with respect to the noun *letters* – a role that is syntactically unrealized in (21). While most letters are sent to their addressees, it is possible for the sendee and the addressee to be different, as in *I sometimes inadvertently send letters to my sister to my brother.* We have annotated this difference by giving *Lee* the two minimally different RESTR values in (32):[10]

(32) a. $\boxed{H}\left\langle\begin{bmatrix}\text{RELN} & \textbf{name}\\ \text{NAME} & \text{Lee}\\ \text{NAMED} & j\end{bmatrix}\right\rangle$

b. $\boxed{I}\left\langle\begin{bmatrix}\text{RELN} & \textbf{name}\\ \text{NAME} & \text{Lee}\\ \text{NAMED} & m\end{bmatrix}\right\rangle$

Since j is the index for the SENDEE role in all of our trees in this section, □ is used when *Lee* is the SENDEE argument of the verb send. We use m as the index for the ADDRESSEE role, so we use $\boxed{I}$ when Lee plays the ADDRESSEE role with respect to the noun *letters.* [11]

[10]For readers who are still skeptical of the existence of this second structure (and interpretation), we provide an alternative appropriate embedding context.

The Corrupt Postal Worker Ransom Context:

[Postal workers **A**, B and C have stolen some important letters. C, who is negotiating ransom money for release of the letters addressed to Lee, is going over the plan with A and B:]

C: So if the phone rings twice, what do you send us?

B: We send two letters to Lee.

[11]This difference could have been annotated in another way. We could have used the same RESTR value for *to Lee* in both cases and assigned alphabetically different values to the SENDEE and ADDRESSEE roles in the two sentences. These two alternatives are not substantively different. They only appear to be distinct because of the way we use tag identity across different sentences in this section.

6.3 Appendix: Well-Formed Structures

In this appendix, we lay out more precisely the constructs of the theory whose effects we have been illustrating in this chapter. This presentation (like the elaborations of it given in Chapter 9 and Appendix A) is intended for readers concerned with the formal foundations of our theory. For most purposes and for most readers, the relatively informal presentation in the body of text, taken together with the definitions in section *6.3.6* below, should be sufficient.

6.3.1 Preliminaries

According to our approach, a grammar G is defined by the following components:

- a finite set of features: $\mathcal{F}$ = {SYN, SEM, HEAD, AGR, ...),
- a finite set of primitive items:
  $\mathcal{A}_{atom} = \mathcal{A}_{pol} \cup \mathcal{A}_{gr.atom} \cup \mathcal{A}_{mode} \cup \mathcal{A}_{reln}$, where:
  1. $\mathcal{A}_{pol} = \{+, -\}$,
  2. (a set of ground atoms) $\mathcal{A}_{gr.atom}$ = {1st, 2nd, 3rd, sg, pl, ..., run, dog, ...),
  3. $\mathcal{A}_{mode}$ = {prop, ques, dir, ref, none), and
  4. $\mathcal{A}_{reln}$ = {walk, love, person, ...),
- a denumerably infinite set of primitive items: $\mathcal{A}_{index} = \mathcal{A}_{ind} \cup \mathcal{A}_{sit}$, where:
  1. $\mathcal{A}_{ind}$ = {i, j, ...) and
  2. $\mathcal{A}_{sit} = \{s_1, s_2, \ldots\}$,
- the distinguished element elist (empty-lzst), discussed below,
- a finite set of types: $\mathcal{T}$ = {noun, agr-pos, plural, expression, ...},
- a type hierarchy with a tree structure associated with constraint inheritance (for instance, the type hierarchy represented by the tree and table in Section 5.10.1 and 5.10.2),
- a set $\mathcal{LT} \subset \mathcal{T}$ called the leaf types (a type τ is a leaf type if it is associated with a leaf in the type hierarchy tree, i.e. if τ is one of the most specific types),
- a set of list types (if τ is a type, then *list*(τ) is a type),
- a set of grammar rules (like the ones we have already encountered, see Section 5.10.4),
- a set of principles (like those in Section 5.10.5), and
- a lexicon (which is a finite set of lexical entries like those in Section 5.10.6).

Thus a grammar G comes with various primitives grouped into two sets: $\mathcal{A}_{atom}$ ($\mathcal{A}_{pol}$, $\mathcal{A}_{gr.atom}$, $\mathcal{A}_{mode}$, $\mathcal{A}_{reln}$) and $\mathcal{A}_{index}$ ($\mathcal{A}_{ind}$, and $\mathcal{A}_{sit}$). G assigns the type *atom* to all elements of $\mathcal{A}_{atom}$. The elements of $\mathcal{A}_{index}$ are used by the grammar for describing individual objects and situations; they are associated with the leaf type index. We assume that no items in these sets of primitives can be further analyzed via grammatical features.

Our grammar appeals to several ancillary notions which we now explicate: feature structure description, *feature* structure, satisfaction of a descrzption, and tree structure.

6.3.2 Feature Structure Descriptions

For expressing the constraints associated with the grammar rules, principles, types, and lexical entries, we introduce the notion of a feature structure description. The feature structure descriptions are given as attribute-value matrices, augmented with the connective '|', set descriptors ({...}), list descriptions (⟨...⟩, attribute-value matrices with FIRST/REST, or two list descriptions connected by $\oplus$), and a set Tags of tags (labels represented by boxed integers or letters).

6.3.3 Feature Structures

The set of feature structures $\mathcal{FS}$ is given by the following recursive definition:

(33) $\phi \in \mathcal{FS}$ (i.e. ϕ is a feature structure) iff

a. $\phi \in \mathcal{A}_{atom} \cup \mathcal{A}_{index}$, or

b. ϕ is a function from features to feature structures; $\phi : \mathcal{F} \longrightarrow \mathcal{FS}$ satisfying the following conditions

1. ϕ is of a leaf type τ;
2. $DOM(\phi)$ = {F | G declares F appropriate for τ} $\cup$

   {F' | $\exists\tau'$ such that τ' is a supertype of τ and

   G declares F' appropriate for τ'},

   i.e. ϕ is defined for any feature that is declared appropriate for τ or for any of τ's supertypes;
3. for each F $\in DOM(\phi)$, G defines the type of the value $\phi(F)$ (we call the value $\phi(F)$ of the function ϕ on F the value of the feature F);and
4. \$ obeys all further constraints ('type constraints') that G associates with type τ (including those inherited from the supertypes τ' of τ), or

c. ϕ is of type *list*(τ), for some type τ, in which case either:

1. ϕ is the distinguished element elist, or else:
2. A. $DOM(\phi)$ is {FIRST, REST),

   B. the type of \$(FIRST) is τ, and

   C. the type of \$(REST) is *list*(τ).

6.3.4 Satisfaction

We explain how feature structures satisfy descriptions indirectly – in terms of denotation, which we define as follows:

Denotation of Feature Structure Descriptions

The denotation of a feature structure description is specified in terms of a structure M:

(34) $\mathcal{M}$ = (A, $\mathcal{F}$, $\mathcal{T}$, **Type, I**), where:

1. $\mathcal{A} = \mathcal{A}_{atom} \cup \mathcal{A}_{index} \cup \{elist\}$,
2. $\mathcal{F}$ is a finite set of features,
3. $\mathcal{T}$ is a finite set of types,
4. Type is a function mapping feature structures to types –
   Type : $\mathcal{FS} \longrightarrow \mathcal{LT}$, where $\mathcal{LT}$ is the set of the leaf types, and

5. I is a function mapping feature names and atomic descriptors to features and atoms of the appropriate sort:

$$I \in I_{\widetilde{\mathcal{F}}} \cup I_{\widetilde{\mathcal{A}}_{atom}} \cup I_{\widetilde{\mathcal{A}}_{ind}} \cup I_{\widetilde{\mathcal{A}}_{sit}} \cup \{\langle elist, elist\rangle\},$$

where

$$I_{\widetilde{\mathcal{F}}} \in \mathcal{F}^{\widetilde{\mathcal{F}}},\ I_{\widetilde{\mathcal{A}}_{atom}} \in \mathcal{A}_{atom}^{\widetilde{\mathcal{A}}_{atom}},\ I_{\widetilde{\mathcal{A}}_{ind}} \in \mathcal{A}_{ind}^{\widetilde{\mathcal{A}}_{ind}},\ I_{\widetilde{\mathcal{A}}_{sit}} \in \mathcal{A}_{sit}^{\widetilde{\mathcal{A}}_{sit}},$$

and $\widetilde{X}$ denotes the set of expressions that have denotations in the set X.[12]

The function I is called an *interpretation function*. An assignment function is a function

$$g : Tags \longrightarrow \mathcal{FS}.$$

We say that a feature structure ϕ is of type $\tau \in \mathcal{T}$ iff there is a (unique) leaf type $\tau' \in \mathcal{LT}$ such that:

(35)
1. τ' is a subtype of τ, and
2. $Type(\phi) = \tau'$.

Given $\mathcal{M}$, the interpretation $[\![d]\!]^{\mathcal{M},g}$ of a feature structure description d with respect to an assignment function g is defined recursively as follows:

(36)
1. if $v \in \widetilde{\mathcal{F}} \cup \widetilde{\mathcal{A}}_{atom} \cup \widetilde{\mathcal{A}}_{index}$, then $[\![v]\!]^{\mathcal{M},g} = \{I(v)\}$;
2. if τ is a type, i.e. $\tau \in \mathcal{T}$, then $[\![\tau]\!]^{\mathcal{M},g} = \{\phi \in \mathcal{FS} : \phi \text{ is of type } \tau\}$;
3. if $F \in \widetilde{\mathcal{F}}$, and d is a feature structure description, then $[\![[F\ d]]\!]^{\mathcal{M},g} =$ $\{\phi \in \mathcal{FS} : \text{there is some } \phi' \text{ such that } \phi' \in [\![d]\!]^{\mathcal{M},g} \text{ and } \langle I(F), \phi'\rangle \in \phi\}$;[13]
4. if $d = \begin{bmatrix} d_1 \\ \dots \\ d_n \end{bmatrix}$ where $n \geq 1$, and $d_1, \dots, d_n$ are feature structure descriptions, then
$$[\![d]\!]^{\mathcal{M},g} = \bigcap_{i=1}^{n} [\![d_i]\!]^{\mathcal{M},g};$$
5. if d is a set descriptor $\{d_1, \dots, d_n\}$, then
$$[\![d]\!]^{\mathcal{M},g} = \bigcup_{i=1}^{n} [\![d_i]\!]^{\mathcal{M},g}$$
$([\![\{\ \}]\!]^{\mathcal{M},g} = \emptyset)$;
6. $[\![d_1 \mid d_2]\!]^{\mathcal{M},g} = [\![d_1]\!]^{\mathcal{M},g} \cup [\![d_2]\!]^{\mathcal{M},g}$;
7. if $d \in Tags$, then $[\![d]\!]^{\mathcal{M},g} = g(d)$;
8. if $d \in Tags$ and d' is a feature structure description, then $[\![d\ d']\!]^{\mathcal{M},g} = \{\phi \in \mathcal{FS} : g(d) = \phi \text{ and } \phi \in [\![d']\!]^{\mathcal{M},g}\}$;
(Note that tagging narrows the interpretation down to a singleton set.)

[12] Y^X is the standard notation for the set of all functions $f : X \rightarrow Y$.

[13] Note that the definition of a feature structure in (33), taken together with this clause, ensures that each element ϕ of the set $[\![[F\ d]]\!]^{\mathcal{M},g}$ is a proper feature structure.

9. List Addition:[14]
a. $[\![elist \oplus d]\!]^{\mathcal{M},g} = [\![d]\!]^{\mathcal{M},g}$,
b. if $d = \begin{bmatrix} \text{FIRST} & d_1 \\ \text{REST} & d_2 \end{bmatrix}$
then $[\![d]\!]^{\mathcal{M},g} =$
$\{\phi \in \mathcal{FS} : \phi(\text{FIRST}) \in [\![d_1]\!]^{\mathcal{M},g} \text{ and } \phi(\text{REST}) \in [\![d_2 \oplus d_3]\!]^{\mathcal{M},g}\}$.

Satisfaction of Feature Structure Descriptions

A feature structure $\phi \in \mathcal{FS}$ satisfies a feature structure description d iff there is some assignment function g such that $\phi \in [\![d]\!]^{\mathcal{M},g}$.

Examples:

(37)
a. ϕ satisfies [NUM sg] iff $\langle$NUM, sg$\rangle \in \phi$.
b. ϕ satisfies [AGR [NUM sg]] iff there is a feature structure ϕ' (which is unique) such that $\langle$AGR, $\phi'\rangle \in \phi$ and $\langle$NUM, sg$\rangle \in \phi'$.
c. ϕ satisfies [AGR *3sing*] iff there is a feature structure ϕ' (which is unique) such that $\langle$AGR, $\phi'\rangle \in \phi$ and ϕ' is of type *3sing*.
d. ϕ satisfies [PER {1st, 2nd, 3rd}] iff $\langle$PER, 1st$\rangle \in \phi$, $\langle$PER, 2nd$\rangle \in \phi$, or $\langle$PER, 3rd$\rangle \in \phi$.
e. ϕ satisfies [ARGS $\langle s_1, s_2, s_3\rangle$] iff:
$\langle$ARGS, {$\langle$FIRST, $s_1\rangle$, $\langle$REST, {$\langle$FIRST, $s_2\rangle$, {REST, {$\langle$FIRST, $s_3\rangle$, $\langle$REST, *elist*$\rangle$}}$\rangle$}$\rangle$}$\rangle \in \phi$.
f. ϕ satisfies:
$$\begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{AGR} & \boxed{1} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle [\text{SYN} [\text{HEAD} [\text{AGR} \boxed{1}]]] \rangle \end{bmatrix} \end{bmatrix} \end{bmatrix}$$
iff
1. ϕ(SYN)(HEAD)(AGR) = ϕ(SYN)(VAL)(SPR)(FIRST)(SYN)(HEAD)(AGR),[15] and
2. ϕ(SYN)(VAL)(SPR)(REST) = *elist*

6.3.5 Tree Structures

Finally, we assume a notion of tree structure described informally as follows:

(38) A tree structure is a directed graph that satisfies a number of conditions:[16]

[14] Where no confusion should arise, we use 'FIRST', 'SYN', etc. to refer either to the appropriate feature (an element of $\mathcal{F}$) or to its name (an element of $\widetilde{\mathcal{F}}$).

[15] Note that parentheses here are 'left associative': '$\phi(X)(Y)$' is equivalent to '$(\phi(X))(Y)$'. That is, both expressions denote the result of applying the function ϕ to (the feature) X and then applying the result to (the feature) Y.

[16] Here, we assume familiarity with notions such as root, mother, *terminal* node, *nonterminal* node, and branches. These and related notions can be defined more precisely in set-theoretic terms, as is done in various texts. See, for example, Hopcroft et al. 2001 and Partee et al. 1990

1. it has a unique root node,
2. each non-root node has exactly one mother,
3. sister nodes are ordered with respect to each other,
4. it has no crossing branches,
5. each nonterminal node is labelled by a feature structure, and
6. each terminal node is labeled by a phonological form (an atom).

6.3.6 Structures Defined by the Grammar

We may now proceed to define well-formedness of tree structures in terms of the licensing of their component trees (recall from Chapters 2 and 3 that a local subtree consists of a mother and all its daughters):

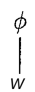
(39) Well-Formed Tree Structure:

Φ is a Well-Formed Tree Structure according to G if and only if:

1. Φ is a tree structure,
2. the label of Φ's root node satisfies S,[17] and
3. each local subtree within Φ is either phrasally licensed or lexically licensed.

(40) Lexical Licensing:

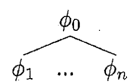
A word structure of the form:



is licensed if and only if G contains a lexical entry $\langle d_1, d_2 \rangle$, where w satisfies d_1 and ϕ satisfies d_2.

(41) Phrasal Licensing:

A grammar rule $\rho = d_0 \rightarrow d_1 \ldots d_n$ licenses a local subtree:

$\Phi =$ 

if and only if:

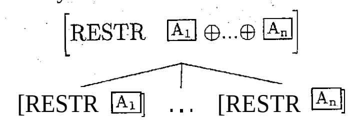
1. for each $i, 0 \leq i \leq n$, ϕ_i is of[18] the type expression,
2. there is some assignment function g under which the sequence $\langle \phi_0, \phi_1, \ldots, \phi_n \rangle$ satisfies the description sequence $\langle d_0, d_1, \ldots, d_n \rangle$,[19]
3. Φ satisfies the Semantic Compositionality Principle, and
4. if ρ is a headed rule, then Φ satisfies the Head Feature Principle, the Valence Principle and the Semantic Inheritance Principle, with respect to ρ.

[17] Recall once again that S abbreviates a certain feature structure constraint, as discussed in Chapter 4.

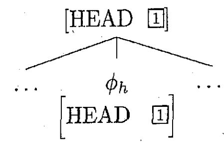
[18] That is, assigned to some leaf type that is a subtype of the type ***expression***.

[19] Note that this clause must speak of a sequence of feature structures satisfying a sequence description. This is because of identities that must hold across members of the sequence, e.g. those required by particular grammar rules.

(42) Φ satisfies the Semantic Compositionality Principle with respect to a grammar rule ρ if and only if Φ satisfies:

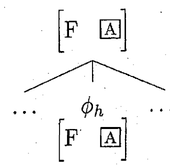


(43) Φ satisfies the Head Feature Principle with respect to a headed rule ρ if and only if Φ satisfies:



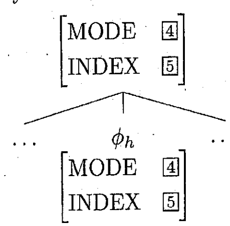
where ϕ_h is the head daughter of Φ.

(44) Φ satisfies the Valence Principle with respect to a headed rule p if and only if, for any VAL feature F, Φ satisfies:



where ϕ_h is the head daughter of Φ and ρ does not specify incompatible F values for ϕ_h and ϕ_0.

(45) Φ satisfies the Semantic Inheritance Principle with respect to a headed rule ρ if and only if Φ satisfies:



where ϕ_h is the head daughter of Φ.

6.4 Problems

Problem 1: A Sentence

For the purposes of this problem, assume that the preposition on in the example below is like to in (18) in that makes no contribution to the semantics other than to pass up the INDEX and MODE values of its object NP. That is, assume it has the following lexical entry:

$$\left\langle \text{on}\ ,\ \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{HEAD} & \textit{prep} \\ \text{VAL} & \begin{bmatrix} & \langle\ \rangle \\ \text{COMPS} & \left\langle \begin{array}{c} \text{NP} \\ \begin{bmatrix} \text{CASE} & \text{acc} \\ \text{MODE} & \boxed{1} \\ \text{INDEX} & \boxed{2} \end{bmatrix} \end{array} \right\rangle \\ \text{MOD} & \langle\ \rangle \end{bmatrix} \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \text{MODE} & \boxed{1} \\ \text{INDEX} & \boxed{2} \\ \text{RESTR} & \langle\ \rangle \end{bmatrix} \end{bmatrix} \right\rangle$$

A. Draw a fully resolved tree structure for the sentence in (i). Use tags to indicate identities required by the grammar. When two feature structures are tagged as identical, you need only show the information in one place.

(i) I rely on Kim.

B. In the VP and PP nodes of your tree, indicate which aspects of the grammar constrain each piece of information (i.e. each feature value). *[Hint:Possible answers include grammar rules and the combined effect of general principles and lexical entries.]*

Problem 2: Spanish NPs II

In this problem we return to Spanish NPs (see Problem 2 in Chapter 4), this time adding adjectives. Unlike English adjectives, Spanish adjectives agree with the nouns they modify, as shown in (i)–(iv):

(i) a. La jirafa pequeiia corrió.
The.FEM.SG giraffe small.FEM.SG ran.3SG
'The small giraffe ran.'
b.*La jirafa pequeñas/pequeño/pequeños corrió.

(ii) a. Las jirafas pequeñas corrieron.
The.FEM.PL giraffes small.FEM.PL ran.3PL
'The small giraffes ran.'
b.*Las jirafas pequeña/pequeño/pequeños corrieron.

(iii) a. El pingiiino pequeiio corrió.
The.MASC.SG penguin small.MASC.SG ran.3SG
'The small penguin ran.'
b.*El pingiiino pequeña/pequeñas/pequeños corrió.

(iv) a. Los pingüinos pequeños corrieron.
The.MASC.PL penguins small.MASC.PL ran.3PL
'The small penguins ran.'
b.*Los pingüinos pequeña/pequeñas/pequeño corrieron.

A. Using the MOD feature to specify which nouns the adjective can modify, give a lexical entry for *pequeños*. Be sure to specify both SYN and SEM features.

[Hint: The semantics of adjectives is very similar to that of adverbs, so the entry for today in Chapter 5 (page 147) may be a helpful guide in doing this.]

B. Assuming the rules we have developed for English are appropriate for Spanish as well, draw a tree for the NP *los pingüinos pequeños* in (iv). Show values for all features, using tags to show identities required by the grammar.

C. Explain how the INDEX value of *pingüinos* is identified with the argument of the *predication* introduced by *pequeños*. (Your explanation should indicate the role of lexical entries, rules, and principles in enforcing this identity.)

Problem 3: English Possessives I

English uses *'s* to express possession, as in the following examples:

(i) Leslie's coffee spilled.
(ii) Jesse met the president of the university's cousin.
(iii)*Jesse met the president's of the university cousin.
(iv) Don't touch that plant growing by the trail's leaves.
(v)*Don't touch that plant's growing by the trail leaves.
(vi) The person you were talking to's pants are torn.
(vii)*The person's you were talking to pants are torn.

(While examples (iv) and (vi) are a bit awkward, people do use such sentences, and there is certainly nowhere else that the *'s* could be placed to improve them).

A. What is the generalization about where the *'s* of possession appears in English?

One traditional treatment of the possessive marker (*'s*) is to claim it is a case marker. In our terms this means that it indicates a particular value for the feature CASE (say, 'poss' for 'possessive') on the word it attaches to. If we tried to formalize this traditional treatment of *'s*, we might posit a rule along the following lines, based on the fact that possessive NPs appear in the same position as determiners:

$$\text{D} \rightarrow \begin{array}{c}\text{NP} \\ [\text{CASE poss}]\end{array}$$

Taken together with our assumption that CASE is a HEAD feature, such an analysis of *'s* makes predictions about the grammaticality of (i)–(vii).

B. Which of these sentences does it predict should be grammatical, and why?

⚠ Problem 4: English Possessives II

An alternative analysis of the possessive is to say that 's is a determiner that builds a determiner phrase (abbreviated DP), via the Head-Specifier Rule. On this analysis, *'s* selects for no complements, but it obligatorily takes an NP specifier. The word 's thus has a lexical category that is like an intransitive verb in valence.

This analysis is somewhat unintuitive, for two reasons: first, it requires that we have an independent lexical entry for 's, which seems more like a piece of a word, phonologically; and second, it makes the nonword *'s* the head of its phrase! However, this analysis does a surprisingly good job of predicting the facts of English possessives, so we shall adopt it, at least for purposes of this text.

A. Ignoring semantics for the moment, give the lexical entry for *'s* assuming its analysis as a determiner, and draw a tree for the NP Kim's brother. (The tree should show the value of HEAD, SPR and COMPS on every node. Use tags to show identities required by the grammar. You may omit other features.)

B. Explain how your lexical entry gets the facts right in the following examples:

   (i) The Queen of England's crown disappeared.

   (ii)*The Queen's of England crown disappeared.

C. How does this analysis handle recursion in possessives, for example, Robin's brother's *wife*, or Robin's brother's wife's parents? Provide at least one tree fragment to illustrate your explanation. (You may use abbreviations for node labels in the tree.)

Problem 5: English Possessives III

The semantics we want to end up with for Pat's book is the one shown in (i) (*poss* is the name of the general possession relation that we will assume provides the right semantics for all possessive constructions):[20]

(i)
$$\begin{bmatrix} \text{MODE} & \text{ref} \\ \text{INDEX} & i \\ \text{RESTR} & \left\langle \begin{bmatrix} \text{RELN} & \\ \text{NAMED} & j \\ \text{NAME} & \text{Pat} \end{bmatrix}, \begin{bmatrix} \text{RELN} & \textbf{poss} \\ \text{POSSESSOR} & j \\ \text{POSSESSED} & i \end{bmatrix}, \begin{bmatrix} \text{RELN} & \textbf{the} \\ \text{BV} & i \end{bmatrix}, \begin{bmatrix} \text{RELN} & \textbf{book} \\ \text{INST} & i \end{bmatrix} \right\rangle \end{bmatrix}$$

[20]We have chosen to use 'the' as the quantifier introduced by possessives, but this is in fact a matter of debate. On the one hand, possessive-NPs are more definite than standard indefinites such as a book. On the other hand, they don't come with the presupposition of uniqueness that tends to come with the. Compare (i) and (ii):

(i) That's the book.

(ii) That's my book.

Part (A) of this problem will ask you to give a SEM value for the determiner *'s* that will allow the grammar to build the SEM value in (i) for the phrase *Pat's book*. Recall that, on our analysis, nouns like *book* select for specifiers like *Pat's*, and the specifiers do not reciprocally select for the nouns. In order to get the correct semantics, *'s* will have to identify its BV value with its INDEX value. In this, it is just like the determiner *a* (see (9) on page 171). This constraint interacts with the constraint on all common nouns shown in (ii) to ensure that the value of BV is correctly resolved:

(ii)
$$\begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{VAL} & \begin{bmatrix} \text{SPR} & \langle [\text{SEM } [\text{INDEX } \boxed{1}]] \rangle \end{bmatrix} \end{bmatrix} \\ \text{SEM} & [\text{INDEX } \boxed{1}] \end{bmatrix}$$

A. Given the discussion above, what is the SEM value of the determiner *'s*?

B. Draw a tree for the phrase *Pat's book*, showing all SEM features on all nodes and SPR on any nodes where it is non-empty. Use tags (or matching indices, as appropriate) to indicate identities required by the grammar.

C. Describe how your analysis guarantees the right SEM value for the phrase. (Your description should make reference to lexical entries, rules and principles, as appropriate.)

Problem 6: English Possessive Pronouns

Possessive pronouns like *my*, *your*, etc. function as determiners in NPs like *my books* and *your mother*. You might think we should treat possessive pronouns as determiners that have the same AGR value as the corresponding nonpossessive pronoun. That is, you might think that *my* should be specified as:

(i)
$$\begin{bmatrix} \text{HEAD} & \begin{bmatrix} \textit{det} & \\ \text{AGR} & \begin{bmatrix} \textit{1sing} & \\ \text{PER} & \text{1st} \\ \text{NUM} & \text{sg} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

A. Explain why this analysis (in particular, the AGR value shown in (i)) will fail to provide an adequate account of *my books* and *your cousin*.

B. The semantics we want to end up with for *my book* is this:

(ii)
$$\begin{bmatrix} \text{MODE} & \text{ref} \\ \text{INDEX} & i \\ \text{RESTR} & \left\langle \begin{bmatrix} \text{RELN} & \textbf{speaker} \\ \text{INST} & j \end{bmatrix}, \begin{bmatrix} \text{RELN} & \textbf{poss} \\ \text{POSSESSOR} & j \\ \text{POSSESSED} & i \end{bmatrix}, \begin{bmatrix} \text{RELN} & \textbf{the} \\ \text{BV} & i \end{bmatrix}, \begin{bmatrix} \text{RELN} & \textbf{book} \\ \text{INST} & i \end{bmatrix} \right\rangle \end{bmatrix}$$

Formulate the SEM value of the determiner *my*.

C. Draw an explicit tree for the phrase *my book*.

[*Hint: Refer to Problem 5.*]

Problem 7: French Possessive Pronouns

Problem 6 asked you to provide an argument as to why my isn't [PER 1st, NUM sg], but didn't concern what the AGR value should be instead.

A. Provide an argument, with suitable data, that the AGR value of English possessive pronouns (e.g. my or *our)* should be left unspecified for number.

Now consider the following data from French. French nouns, like Spanish nouns, are all assigned either masculine or feminine gender. In these examples, ***pie*** is feminine and moineau is masculine.

(i) ma pie
my magpie
(ii)*mon/mes pie
(iii) mon moineau
my sparrow
(iv)*ma/mes moineau
(v) mes pies
my magpies
(vi)*ma/mon pies
(vii) mes moineaux
my sparrows
(viii)*ma/mon moineaux

B. Give the AGR values for ma, *mon,* and mes.

7

Binding Theory

7.1 Introduction

This chapter revisits a topic introduced very informally in Chapter 1, namely, the distribution of reflexive and nonreflexive pronouns. In that discussion, we noticed that the well-formedness of sentences containing reflexives usually depends crucially on whether there is another expression in the sentence that has the same referent as the reflexive; we called such an expression the 'antecedent' of the reflexive. Nonreflexive pronouns, on the other hand, often lack an antecedent in the same sentence. The issue for a nonreflexive pronoun is typically whether a particular NP could have the same referent (or, as linguists often put it, be coreferential with it) – that is, whether that NP could serve **as** the antecedent for that pronoun.

In discussing these phenomena, we will use the notation of subscripted indices to mark which expressions are intended to have the same referent and which are intended to have distinct referents. Two expressions with the same index are to be taken as coreferential, whereas two expressions with different indices are to be understood as having distinct referents.

Thus the markings in (1)indicate that himselfmust refer to the same person as John, and that the referent of her must be someone other than Susan:

(1) a. John, frightens himself,.
b.*Susan$_i$ frightens her,.
c. Susan$_i$ frightens her$_j$.

As mentioned in Chapter 5, the subscript notation is shorthand for the value of the feature INDEX.

In examples like (1a), the reflexive himself is often said to be 'bound' by its antecedent. This terminology derives from an analogy between natural language pronouns and variables in mathematical logic. The principles governing the possible pairings of pronouns and antecedents are often called BINDING PRINCIPLES, and this area of study is commonly referred to as BINDING THEORY.' The term ANAPHORIC is also used for

[1]Much of the literature on Binding Theory actually restricts the term 'binding' to elements in certain syntactic configurations. Specifically, an element *A* is often said to bind an element B if and only if: (i) they have the same index; and (ii) *A* c-commands *B.* The technical term 'c-command' has been defined in several (nonequivalent) ways in the literature; the most commonly used definition is the following:

expressions (including pronouns) whose interpretation requires them to be associated with other elements in the discourse; the relationship of anaphoric elements to their antecedents is called ANAPHORA.

With this notation and terminology in place, we are now ready to develop a more precise and empirically accurate version of the Binding Theory we introduced in Chapter 1.

7.2 Binding Theory of Chapter 1 Revisited

Recall that in Chapter 1, on the basis of examples like (2)–(9), we formulated the hypothesis in (10):

(2) a. Susan, likes herself,.
b.*Susan$_i$ likes her,.

(3) a. Susan, told herself, a story.
b.*Susan$_i$ told her, a story.

(4) a. Susan, told a story to herself,.
b.*Susan$_i$ told a story to her,.

(5) a. Susan, devoted herself, to linguistics.
b.*Susan$_i$ devoted her$_i$ to linguistics.

(6) a. Nobody told Susan, about herself,.
b.*Nobody told Susan, about her,.

(7) a.*Susan$_i$ thinks that nobody likes herself,.
b. Susan, thinks that nobody likes her,.

(8) a."Susan,'s friends like herself,.
b. Susan$_i$'s friends like her,.

(9) a.*That picture of Susan, offended herself,.
b. That picture of Susan, offended her,.

(10) Reflexive pronouns must be coreferential with a preceding argument of the same verb; nonreflexive pronouns cannot be.

Our task in this chapter is to reformulate something close to the generalization in (10) in terms of the theoretical machinery we have been developing in the last five chapters. We would also like to extend the empirical coverage of (10) to deal with examples that our informal statement did not adequately handle. Toward this end, let us divide (10) into two principles, one for reflexive pronouns and the other for nonreflexive pronouns. Our first try at formulating them using the new binding terminology is then the following:

node A in a tree c-commands node B if and only if every branching node dominating *A* dominates B. Intuitively, this means roughly that A is at least as high in the tree **as** B. Our investigations into Binding Theory will not impose any such configurational limitation, as we will be deriving a similar, arguably superior characterization of constraints on binding in terms of ARG-ST lists (see below).

Note that we are interested in determining the conditions governing the pairing of pronouns and antecedents in a sentence. We will not, however, consider what possible things outside the sentence (be they linguistic expressions or entities in the world) can serve **as** antecedents for pronouns.

(11) Principle A (version I)
A reflexive pronoun must be bound by a preceding argument of the same verb.

Principle B (version I)
A nonreflexive pronoun may not be bound by a preceding argument of the same verb.

7.3 A Feature-Based Formulation of Binding Theory

Our binding principles make use of several intuitive notions that need to be explicated formally within the theory we have been developing. The terms 'reflexive pronoun' and 'nonreflexive pronoun' have not been defined. What distinguishes reflexive pronouns is a semantic property, namely, that they require linguistic antecedents (of a certain kind) in order to be interpreted. Hence, we introduce a new value of the semantic feature MODE that we will use to distinguish reflexive pronouns; we will call that value 'ana'. Nonreflexive pronouns, like nonpronominal nouns, are [MODE ref].[2] In addition, we will assume (building on the conclusions of Problem 2 in Chapter 1) that reciprocals (that is, *each* other and perhaps one another) are [MODE ana]. This will allow us to reformulate the binding principles in terms of the feature MODE, keeping open the possibility that reflexives and reciprocals might not be the only elements subject to Principle A.

7.3.1 The Argument Structure List

Both of our binding principles contain the phrase 'a preceding argument of the same verb'. Formalizing this in terms of our theory will take a bit more work. The features that encode information about what arguments a verb takes are the valence features SPR and COMPS. Though we have not said much about the linear ordering of arguments, we have placed elements on our COMPS lists in the order in which they appear in the sentence. Hence, to the extent that precedence information is encoded in our feature structures, it is encoded in the valence features. So the valence features are a natural place to start trying to formalize the binding principles.

There is a problem, however. For examples like (2)–(5), the binding in question involves the subject NP and one of the nonsubject NPs; but our valence features separate the subject (specifier) and the nonsubject (complements) into two different lists. To facilitate talking about all of the arguments of a verb together, we will posit a new list-valued feature, ARGUMENT-STRUCTURE (ARG-ST), consisting of the sum (in the sense introduced in Chapter 5) of the SPR value (the subject) and the COMPS value (the complements)[3].

Words obey the following generalization, where '$\oplus$' again denotes the operation we have called 'sum', appending one list onto another:[4]

[2]Note that the Semantic Inheritance Principle guarantees that NPs headed by [MODE ref] nouns share that specification.

[3]MOD, which we have included among the valence features, does not list arguments of the verb. So the value of MOD is not related to ARG-ST.

[4]We will revisit and revise the Argument Realization Principle in Chapter 14.

(12) Argument Realization Principle (Version I)
A word's value for ARG-ST is $\boxed{A} \oplus \boxed{B}$, where $\boxed{A}$ is its value for SPR and $\boxed{B}$ is its value for COMPS.

So, if a verb is specified as [SPR ⟨ NP ⟩] and [COMPS ⟨ NP ⟩], then the verb's argument structure list is ⟨ NP , NP ⟩. And if some other verb is specified as [SPR ⟨ NP ⟩] and [COMPS ⟨ PP , VP ⟩], then that verb's argument structure list is ⟨ NP , PP , VP ⟩, and so on.

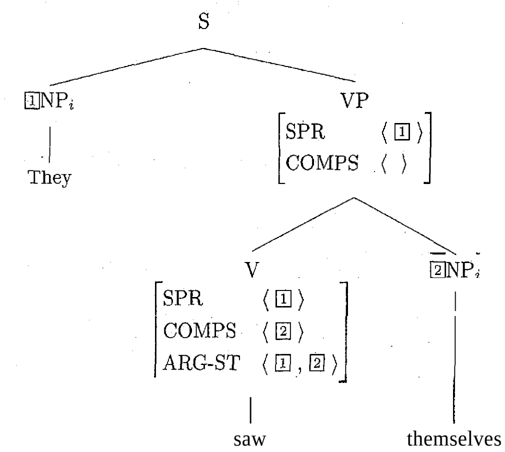
Exercise 1: Practice ARG-ST lists
What would be the value of ARG-ST in the lexical entries of each of the following verbs: devour, elapse, put, and rely? As defined, any word with valence features will have an ARG-ST value. So what would the ARG-ST values be for letter, of, today, and Venezuela?

Of course we mean real identity between the members of these lists, as shown by the specifications in (13):

(13) a.
$$\begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{VAL} & \begin{bmatrix} \text{SPR} & \langle \boxed{1} \rangle \\ \text{COMPS} & \langle \boxed{2} \rangle \end{bmatrix} \end{bmatrix} \\ \text{ARG-ST} & \langle \boxed{1}\text{NP} , \boxed{2}\text{NP} \rangle \end{bmatrix}$$

b.
$$\begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{VAL} & \begin{bmatrix} \text{SPR} & \langle \boxed{1} \rangle \\ \text{COMPS} & \langle \boxed{2} , \boxed{3} \rangle \end{bmatrix} \end{bmatrix} \\ \text{ARG-ST} & \langle \boxed{1}\text{NP} , \boxed{2}\text{PP} , \boxed{3}\text{VP} \rangle \end{bmatrix}$$

These identities are crucial, as they have the side effect of ensuring that the binding properties of the complements are actually merged into the verb's argument structure, where they will be governed by our binding principles. For example, the Head-Specifier Rule identifies a subject's feature structure with the sole member of the VP's SPR list. It follows (from the Valence Principle) that the subject's feature structure is also the sole member of the verb's SPR list. This, in turn, entails (by the Argument Realization Principle) that the subject's feature structure is the first member of the verb's ARG-ST list. Thus once the distinctions relevant to Binding Theory are encoded in the feature structures of reflexive and nonreflexive NPs, this same information will be present in the ARG-ST of the lexical head of the sentence, where the binding principles can be enforced. This is illustrated in (14):

(14)


The generalization in (12) holds only of words; in fact, it is only word structures that have the feature ARG-ST. Despite its close relationship to the valence features, ARG-ST serves a different function and hence has different formal properties. SPR and COMPS, with the help of the Valence Principle, keep track of elements that a given expression needs to combine with. As successively larger pieces of a tree are constructed, the list values of these features get shorter. By contrast, we introduced the argument structure list as a locus for stating more formal versions of the binding principles. Through a series of identities enforced by the Argument Realization Principle, the phrase structure rules and the Valence Principle, the ARG-ST list of a verb occurring in a tree contains all of the information about that verb's arguments that a precise version of the binding principles needs. It is part of neither SYN nor SEM, but rather serves to express certain relations at the interface of syntax and semantics. These relations can be stated once and for all on the ARG-ST of the lexical head. There is no need to copy the information up to higher levels of the tree, and so ARG-ST is posited only as a feature of *words*, not *phrases*.

The elements of an ARG-ST list are ordered, and they correspond to phrases in the phrase structure tree. We can thus use the ordering on the ARG-ST list to impose a ranking on the phrases in the tree. A bit more precisely, we can say:

(15) If *A* precedes B on some argument structure (ARG-ST) list, we say that A OUTRANKS *B*.

Incorporating both our characterization of reflexive pronouns in terms of MODE and our definition of 'outrank', we can now reformulate our binding principles as follows:

(16) Principle A (Final Version)
A [MODE ana] element must be outranked by a coindexed element.
Principle B (Final Version)
A [MODE ref] element must not be outranked by a coindexed element.

Notice that in this reformulation, Principle B now applies more generally, so as to govern nonpronominal elements like proper names and quantified NPs. This is a happy result, given the following examples, which are now correctly predicted to be ungrammatical:

(17) a.*Sandy$_i$ offended Jason$_i$.
b.*He$_i$ offended Sandy$_i$.
c.*He$_i$ offended each lawyer$_i$.

7.4 Two Problems for Binding Theory

These formulations have certain problems, requiring further discussion and refinement.

7.4.1 Pronominal Agreement

First, (16) says nothing about agreement between pronouns and antecedents; but we do not want Principle A to license examples like (18):

(18) a. *I enjoy yourself.
b. *He enjoys themselves.
c. *She enjoys himself.

We could rule these out by adding a stipulation to Principle A, requiring a reflexive and its antecedent to agree. But this ad *hoc* approach wouldn't explain much. It is intuitively clear why coindexed elements should exhibit a form of agreement: coindexation indicates that the expressions denote the same entity, and the properties indicated by agreement features are characteristically properties of the entity referred to (the expression's DENOTATION). Thus, for example, singular NPs normally denote single entities, whereas plural NPs denote collections. Hence a singular pronoun cannot normally be coindexed with a plural NP, because they cannot have the same denotation.

We will consequently refrain from any mention of agreement in the binding principles. Instead, we adopt the following general constraint:[5]

(19) Anaphoric Agreement Principle (AAP)
Coindexed NPs agree.

By 'agree', we mean 'have the same values for AGR'. Recall that AGR was introduced in Chapter 3 as a feature whose value is a feature structure specifying values for the features PER (person), NUM (number), and (in the case of *3sing* AGR values) GEND (gender). Only PER and NUM matter for the purposes of subject-verb agreement, but pronouns must also agree with their antecedents in gender, as illustrated in (18c). Since GEND is part of AGR, it is covered by the AAP.

One important advantage of leaving agreement out of the formulation of binding principles themselves is that the AAP also covers agreement between nonreflexive pronouns and their antecedents. Since Principle B only says which expressions must NOT be coindexed with nonreflexive pronouns, it says nothing about cases in which such pronouns ARE legally coindexed with something. The AAP rules out examples like (20), which are not ruled out by our formulation of Principle B.

(20) *I$_i$ thought that nobody liked him$_i$.

[5]The use of the term 'anaphoric' in (19) is intended to underscore that coindexing is used to represent the informal notion of anaphora.

It is important to realize that coindexing is not the same thing as coreference; any two coindexed NPs are coreferential, but not all pairs of coreferential NPs are coindexed. There are some tricky cases that might seem to be counterexamples to the AAP, and all of which turn out to be consistent with the AAP, once we make the distinction between coindexing and coreference. One such example is the following:

(21) The solution to this problem is rest and relaxation.

Here the singular NP *the solution to this problem* appears to refer to the same thing as the plural NP *rest and relaxation*. And indeed we would say that the two NPs are coreferential, but they are not coindexed. Thus while coindexing and coreference usually go hand in hand, they don't in this case. The whole point of identity sentences of this kind is to convey the information that two distinct (i.e. distinctly indexed) expressions refer to the same thing. If you are familiar with mathematical logic, this might remind you of situations in which two distinct variables are assigned the same value (making, e.g. '$x = y$' true). Indices are like variables; thus Binding Theory constrains variable identity, not the assignments of values to variables.

Other examples that appear to violate the AAP turn out to be cases where the pronoun isn't even coreferential with its apparent antecedent. Rather, the phrase that the pronoun is 'referring back to' only indirectly introduces the referent of the pronoun into the domain of discourse. For example, consider the sentence in (22):

(22) An interesting couple walked in. He was four foot nine; she was six foot two.

Here, the NP *an interesting couple* refers to the two people denoted by *he* and *she*, but these three expressions all have distinct indices. This is consistent with the AAP. In fact, the referent of the NP *an interesting couple* is just one entity – the couple, which is a collection of two individuals. As the collection is introduced into the discourse, however, it also makes salient each individual that is in the collection, and it is these individuals that the pronouns in the next sentence refer to. Thus in this discourse, the NP *an interesting couple*, the pronoun *he* and the pronoun *she* all refer to different things. So the AAP doesn't apply.

Similar examples involve collective nouns like *family*, which can denote a single entity, as shown by the singular verb agreement in (23), but which can, as a 'side effect', introduce a collection of entities that can serve as the antecedent for a subsequent plural pronoun:

(23) My family hates cornflakes. But they love granola.

Again there are two distinct entities being referred to by distinct indices.[6]

7.4.2 Binding in Prepositional Phrases

A second problem with our formulation of the binding principles is that reflexives and their antecedents can be objects of prepositions. A PP that consists of a prepositional head daughter like *to* or *about* and a reflexive NP object can then become a complement

[6]For some speakers, this is even possible in the context of reflexive pronouns, i.e. in examples like (i):
(i) Pat's family is enjoying themselves.
The theory we develop does not allow examples of this sort.

of the verb; and when this happens, the reflexive NP inside the PP enters into binding relations with the other arguments of the verb. Similarly, when a nonreflexive pronoun functions as a prepositional object, it can behave like an argument of the verb for purposes of binding. Thus we find the pattern of binding illustrated in (24) and (25):

(24) a. They$_i$ talk [to themselves$_i$].
b.*They$_i$ talk [to them$_i$].

(25) a. Nobody told Susan$_i$ [about herself$_i$].
b.*Nobody told Susan$_i$ [about her$_i$].

And in similar examples, the prepositional object can serve as the binder of a reflexive, but not of a nonreflexive:

(26) a. Nobody talked [to Susan$_i$] [about herself$_i$].
b.*Nobody talked [to Susan$_i$] [about her$_i$].

In examples like these, the binding principles, as formulated above, make the wrong predictions: the Argument Realization Principle (henceforth ARP) requires that the verb's ARG-ST contain the feature structure of the PP, not that of the NP within the PP. Hence if a reflexive pronoun is inside a PP that is a complement to a verb, the reflexive's feature structure will not appear on the same ARG-ST list as (the feature structures of) the verb's subject and object NPs. The Binding Theory, as formulated, thus fails to take into account the fact that certain prepositions seem to be transparent for binding purposes. That is, if prepositions such as these were simply not there and the prepositional object were an object of the verb, then Binding Theory would make just the right predictions about (24)–(26) and related examples.

This problem raises both empirical and formal questions. The empirical question is the issue of precisely when objects of prepositions can enter into binding relations with elements outside the PP. As we noted in our initial discussion of Binding Theory in Chapter 1, there is some variability about the binding possibilities of objects of prepositions. This is illustrated in (27):[7]

(27) a. The house$_i$ had a fence around $\left\{\begin{matrix}\text{it}_i\\ \text{*itself}_i\end{matrix}\right\}$.

b. To make a noose, you wind the rope$_i$ around $\left\{\begin{matrix}\text{itself}_i\\ \text{*it}_i\end{matrix}\right\}$.

c. Susan$_i$ wrapped the blanket around $\left\{\begin{matrix}\text{her}_i\\ \text{herself}_i\end{matrix}\right\}$.

[7]Some readers may have a strong preference for one version of (27c) over the other. It appears that there is some cross-speaker variation regarding such examples. For readers who do not accept both versions of (27c), here are some additional examples in which many speakers accept both reflexive and nonreflexive pronouns:

(i) Jane$_i$ put the TV remote down beside $\left\{\begin{matrix}\text{her}_i\\ \text{herself}_i\end{matrix}\right\}$.

(ii) Mary$_i$ took a quick look behind $\left\{\begin{matrix}\text{her}_i\\ \text{herself}_i\end{matrix}\right\}$.

These examples also show that it is not simply the choice of preposition that determines whether a prepositional object can be reflexive, but also the particular verb that the preposition combines with.

One possible explanation of such differences is based on the intuitive idea underlying our Binding Theory: that reflexives and their antecedents are always arguments of the same predicate. It seems plausible to claim that English prepositions have two distinct semantic functions. In some uses, they function much like verbs, introducing new predications in which they assign argument roles to the nouns they combine with. In other uses, they are simply functioning as argument markers – that is, they indicate what role their object plays in the situation denoted by the verb of the clause they appear in. The clearest examples of this argument-marking use of prepositions are sentences like (4a), *Susan$_i$* told a story to herself$_i$, in which to is used to mark what traditional grammarians called the indirect object. In these cases, the preposition can actually be omitted if the order of the complements is reversed: Susan told herself a story.

In (27a), the preposition arguably functions as a separate predicate (making the sentence mean roughly, 'The house had a fence, and the fence was around the house'), whereas in (27b), the preposition simply marks one of the arguments of the verb *wind*. Notice that nothing in the meaning of the verb *had* leads one to expect that anything is or goes around its subject. In contrast, the verb *wind* indicates that something is going around something else, so the preposition is introducing an expected participant in the situation. These remarks are intended to provide intuitive motivation for the formal distinction we make between the two types of prepositions, but the real reason we need the distinction is to account for the distribution of reflexive and nonreflexive pronouns. Cases like (27c), then, will be treated as having prepositions that are ambiguous between being independent predicates and argument markers.[8]

Let us now formalize this intuition. For the purposes of Binding Theory, nothing new needs to be said about the prepositions that function as independent predicates. If the object of such a preposition is [MODE ana], then Principle A will require it to be coindexed with something that outranks it on the preposition's ARG-ST list. This is not the case in (27a).[9] If the prepositional object is [MODE ref], it must not be coindexed with anything that outranks it on the preposition's ARG-ST list. Since the subject of the sentence in (27a) does not appear on the ARG-ST list of around, Principle B permits a nonreflexive pronoun *it* coindexed with *the* house to appear as the object of around.

For prepositions that function as argument markers, however, we need to provide some way by which they can transmit information about their object NP up to the PP that they project. In particular, in order for the binding principles to make the right predictions with respect to objects of argument-marking prepositions, we need to be able

[8]This leads in certain cases to prepositions like *around* being unintuitively treated as not directly contributing to the semantics of the sentence. A full analysis of these facts is beyond the scope of this book.

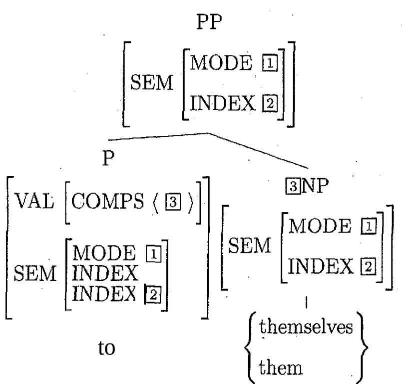
[9]We leave open for now the question of how many ARG-ST members such predicational prepositions have. If *around* in (27a) has two arguments (as seems intuitive from its relational meaning), then the first argument should be identified with *a fence*; hence, *itself* could still not be coindexed with *the house*. In Chapter 12, we will investigate mechanisms by which different ARG-ST lists can have elements with the same index.

to determine at the level of the PP both whether the object NP is a reflexive pronoun (that is, whether it is [MODE ana]) and also what its INDEX value is. If the object's MODE and INDEX values can be transmitted up to the PP, then the higher verb that takes the PP as its complement will have the MODE and INDEX information from the object NP in its ARG-ST, within the PP's SEM value. Note that without some method for transmitting this information up to the PP, the information about the preposition's object is invisible to the higher verb selecting the PP as its complement. The COMPS list of the PP, for example, is empty.

The method we use to transmit this information is straightforward: argument-marking prepositions, such as (some uses of) to, about, and of, share the MODE and INDEX values of their objects. This is illustrated in the lexical entry in (28):

(28)
$$\left\langle \text{to}, \begin{bmatrix} \text{SYN} & \begin{bmatrix}\text{HEAD} & \textit{prep}\\ \text{VAL} & [\text{SPR}\ \langle\ \rangle]\end{bmatrix}\\ \text{ARG-ST} & \left\langle \overset{\text{NP}}{\begin{bmatrix}\text{SYN} & [\text{HEAD}\ [\text{CASE}\ \text{acc}]]\\ \text{SEM} & \begin{bmatrix}\text{MODE} & \boxed{1}\\ \text{INDEX} & \boxed{2}\end{bmatrix}\end{bmatrix}} \right\rangle \\ \text{SEM} & \begin{bmatrix}\text{MODE} & \boxed{1}\\ \text{INDEX} & \boxed{2}\end{bmatrix}\end{bmatrix}\right\rangle$$

The MODE and INDEX values are projected up from the preposition to the PP by the Semantic Inheritance Principle, as shown in (29):

(29)


A PP like this can be selected by a verb like tell or wind. Hence, the PP on its ARG-ST list will contain the object NP's MODE and INDEX values within it. Put another way, the information about the object of the preposition that we need in order to apply the binding principles is available in the verb's ARG-ST list.

To get the right binding results for the objects of argument-marking prepositions, we now need to make a slight modification to our definition of 'outranks'. In particular, we need to say that an argument-marking PP and its object NP are 'of equal rank', by which we mean that they outrank exactly the same elements and are outranked by exactly the same elements. More precisely:

(30) (i) If a node is coindexed with its daughter, their feature structures are of equal rank.

(ii) If there is an ARG-ST list on which A precedes B, then A has a higher rank than (i.e. outranks) B.

Part (ii) of this definition is just the definition we gave earlier. Part (i) is needed to account for the binding facts in argument-marking PPs. Consider, for example, the case where the object of such a PP is a reflexive pronoun (e.g. The children fended *for themselves*). The reflexive's INDEX is shared by the preposition for, as is the [MODE ana] specification, as required by the lexical entry for the argument-marking for. These values are also shared by the whole PP, for themselves, as required by the Semantic Inheritance Princple. So the PP and the reflexive pronoun it contains are coindexed; hence, by part (i) of the definition above, the PP and the reflexive pronoun are of the same rank. In the ARG-ST of *fended*, the feature structure of *the children* outranks that of *for themselves.* Consequently, the feature structure of *the children* outranks that of *themselves.* Thus, if *the children* and themselves are coindexed, Principle A of the Binding Theory is satisfied. Without part (i) of the definition, the reflexive pronoun would not satisfy Principle A.[10] We will go through a similar example, as well as one with a nonreflexive pronoun, below.

The formal machinery we have just developed is designed to capture the fact that objects of prepositions in English exhibit different binding properties in different environments. It involves positing two kinds of lexical entries for prepositions: one contributes its own MODE and INDEX values; the other adopts those of its object, thereby serving as a conduit for that information to be passed on to the dominating PP. We attempted to motivate this distinction through an intuition that the two kinds of prepositions serve different semantic functions. But such intuitions vary considerably from speaker to speaker, so it would be dangerous to put too much weight on them. Our analysis provides a more reliable means of classifying prepositions as argument marking or predicational, namely, exploring their binding properties. Prepositions that are transparent for purposes of binding should be analyzed as argument markers; those whose objects cannot be bound by a preceding NP in the clause should be analyzed as predicational.

7.5 Examples

So far, this chapter has motivated several technical innovations in our theory (ARG-ST, the concept of 'outranking', and the distinction between the two types of prepositions). In this subsection, we present two examples to illustrate the formal machinery we have been discussing.

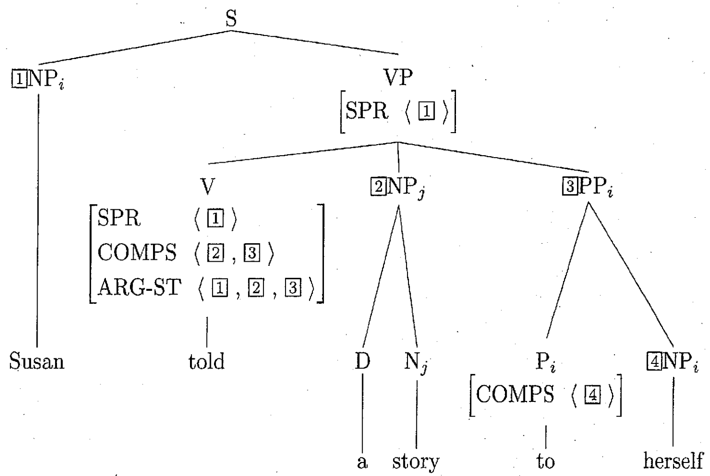
[10] As a consequence of the way we've formalized our analysis, the P *for* is also [MODE ana] and therefore subject to Principle A. It satisfies Principle A in the same way the object NP does: by part (i) of (30), its rank is equal to that of the PP and thus it is outranked by *the children*.

Consider first (4a), repeated here for convenience as (31):

(31) Susan$_i$ told a story to herself$_i$.

The structure licensed by our grammar is the following (omitting irrelevant details):

(32)



The geometry of this tree is given by our phrase structure rules in ways that are by now familiar. The aspect of the tree we are concerned with here is the coindexing of the nodes, indicated by the subscripted i and the resulting argument structure of the verb *told*, which is displayed in (33):

(33)
$$\left[\text{ARG-ST}\ \left\langle \overset{\text{NP}_i}{\left[\text{MODE ref}\right]}, \overset{\text{NP}_j}{\left[\text{MODE ref}\right]}, \overset{\text{PP}_i}{\left[\text{MODE ana}\right]} \right\rangle\right]$$

This ARG-ST conforms to the Binding Theory: the [MODE ana] PP is outranked by a coindexed NP, namely the first NP on the list. Similarly, the NP tagged $\boxed{4}$ in (32), which is also [MODE ana], is of equal rank with the PP dominating it (by the definition of rank), so it is outranked by the first NP in the list. Again, Principle A is satisfied. Notice that Principle A requires coindexing between the prepositional object and one of the other arguments, in this case, the subject. The ARG-ST list of *told* plays a crucial role in enforcing this coindexing, even though the verb is one level below the subject and one level above the prepositional object in the tree.

Principle A would also be satisfied if the anaphor were coindexed with the direct object NP:

(34)
$$\left[\text{ARG-ST}\ \left\langle \overset{\text{NP}_j}{\left[\text{MODE ref}\right]}, \overset{\text{NP}_i}{\left[\text{MODE ref}\right]}, \overset{\text{PP}_i}{\left[\text{MODE ana}\right]} \right\rangle\right]$$

Although this is implausible with *told* (because of the nonlinguistic fact that people are not the kind of thing that gets told to others), it is much easier to contextualize grammatically analogous sentences with the verb *compared*:

(35) a. We compared him$_i$ [to himself$_i$] (at an earlier age).
b. We compared them$_i$ [to each other$_i$].

Thus in both (33) and (34), the PP – and hence its NP object as well – is outranked by some coindexed element. It seems correct to say that as far as grammar is concerned, both the ARG-ST configurations in (33) and (34) are acceptable, although there are independent factors of plausibility that interact to diminish the acceptability of many grammatical examples.

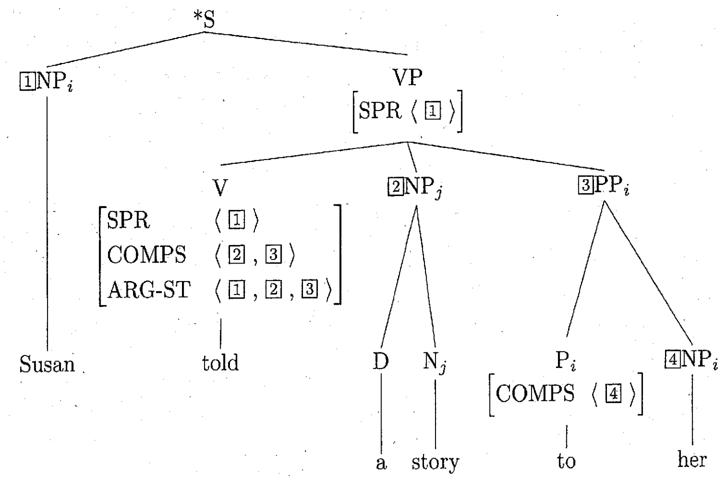
Exercise 2: The Distribution of ARG-ST
Which nodes in (32) have the feature ARG-ST?

Now consider (4b), repeated here for convenience as (36):

(36) *Susan$_i$ told a story to her$_i$.

The tree structure that our grammar must rule out is the following:

(37)



The lexical entry for *her* specifies that it is [MODE ref] – that is, that it is not a reflexive (or reciprocal) pronoun. As in the case of the previous example, the lexical entry for *to* and the Semantic Inheritance Principle pass information to the P and the PP. The verb's ARG-ST list then looks like (38):

(38) *
$$\left[\text{ARG-ST}\ \left\langle \overset{\text{NP}_i}{\left[\text{MODE ref}\right]}, \overset{\text{NP}_j}{\left[\text{MODE ref}\right]}, \overset{\text{PP}_i}{\left[\text{MODE ref}\right]} \right\rangle\right]$$

The PP in (38) violates Principle B: it is a [MODE ref] element that is coindexed with another element that outranks it – namely, the first NP on the list. Consequently, the coindexing indicated is not permitted.

7.6 Imperatives and Binding

In Chapter 1 we noted that the behavior of reflexive and nonreflexive pronouns in sentences like (39) is what one would expect if they had second-person subjects:

(39) a. Protect yourself!

b. *Protect $\left\{\begin{array}{l}\text{myself}\\ \text{himself}\end{array}\right\}$!

c. *Protect you!

d. protect $\left\{\begin{array}{l}\text{me}\\ \text{him}\end{array}\right\}$!

Sentences like these are known as IMPERATIVE sentences. Their characteristic properties are that they lack an overt subject, employ an uninflected form of the verb, and are used to express directives. Such sentences are sometimes said to have 'understood' second-person subjects. The distribution of reflexives illustrated in (39) shows that imperatives do indeed behave in at least one way as if they had second-person subjects.

Our theory provides a straightforward way of capturing the intuition that imperatives have understood subjects. First we need to allow for verb forms that lack the inflections of the verb forms we have been considering thus far. These forms, produced by a lexical rule discussed in the next chapter, have no inflectional endings and are distinguished from other kinds of verbal forms in terms of differing values for the HEAD feature FORM.[11] This basic form of a verb has the FORM value 'base'.

We introduce a new grammar rule to analyze imperative sentences. This rule allows a sentence to consist of a single daughter: a VP specified as [FORM base]. In requiring that the daughter be so specified, we ensure that the lexical head of that phrase will be an uninflected verbal form, such as *be*, *get*, *run*, or *look*. The new rule we need for imperative sentences is a nonheaded rule that says a sentence may consist of a [FORM base] VP that behaves as though it had a second-person subject and is interpreted as a directive:

(40) Imperative Rule

$$\left[\begin{array}{ll} \textit{phrase} & \\ \text{HEAD} & \textit{verb} \\ \text{VAL} & \left[\begin{array}{ll}\text{SPR} & \langle\ \rangle\\ \text{COMPS} & \langle\ \rangle\end{array}\right] \\ \text{SEM} & \left[\begin{array}{ll}\text{MODE} & \text{dir}\\ \text{INDEX} & s\end{array}\right] \end{array}\right] \rightarrow \left[\begin{array}{ll} \text{HEAD} & \left[\begin{array}{ll}\textit{verb} & \\ \text{FORM} & \text{base}\end{array}\right] \\ \text{VAL} & \left[\begin{array}{ll}\text{SPR} & \langle \text{NP}\left[\text{PER}\ \ \text{2nd}\right]\rangle\\ \text{COMPS} & \langle\ \rangle\end{array}\right] \\ \text{SEM} & \left[\text{INDEX}\ \ s\right] \end{array}\right]$$

Recall that imperative sentences require their subject to be second-person, a fact that is captured by the constraint on the SPR of the daughter in (40). And though all verbs are lexically specified as [MODE prop] (which is in turn passed up to the [FORM base] VP that enters into the imperative construction), (40) ensures that any phrase it sanctions is

[11] We will have more to say about the feature FORM in Chapter 8.

specified as [MODE dir] – that is, that it has a meaning appropriate for an imperative.[12]

The Imperative Rule sanctions structures like the one depicted in (41):

(41)

$$\begin{array}{c} \text{S} \\ \left[\begin{array}{ll} \text{VAL} & \left[\begin{array}{ll}\text{SPR} & \langle\ \rangle\\ \text{COMPS} & \langle\ \rangle\end{array}\right] \\ \text{SEM} & \left[\begin{array}{ll}\text{MODE} & \text{dir}\\ \text{INDEX} & s\\ \text{RESTR} & \boxed{1}\end{array}\right] \end{array}\right] \\ | \\ \text{VP} \\ \left[\begin{array}{ll} \text{VAL} & \left[\begin{array}{ll}\text{SPR} & \langle \text{NP}[\text{PER}\ \ \text{2nd}]\rangle\\ \text{COMPS} & \langle\ \rangle\end{array}\right] \\ \text{HEAD} & [\text{FORM}\ \ \text{base}] \\ \text{SEM} & \left[\begin{array}{ll}\text{MODE} & \text{prop}\\ \text{INDEX} & s\\ \text{RESTR} & \boxed{1}\end{array}\right] \end{array}\right] \\ \triangle \\ \text{get a job} \end{array}$$

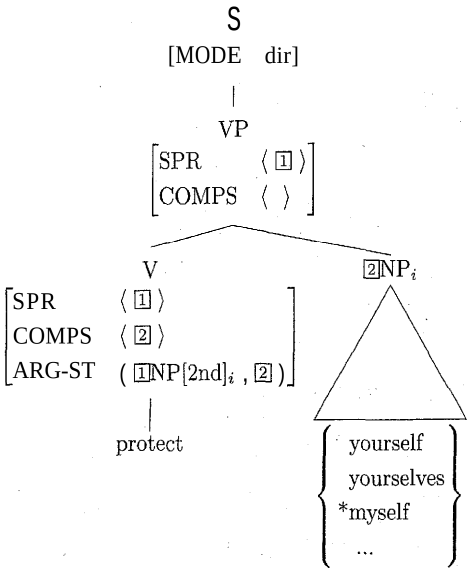
Note that, because the Imperative Rule is a not a headed rule, the Head Feature Principle, the Valence Principle, and the Semantic Inheritance Principle are not relevant to licensing the S node in (41) (though the Semantic Compositionality Principle identifies the RESTR value of the mother in (41) with the RESTR value of the daughter). Instead, the values of the features on the S node are dictated by the rule itself and/or the initial symbol.[13]

The last thing to understand about the rule in (40) is that it explains the observations we have made about anaphor binding in imperative sentences. By requiring the specifier of an imperative VP to be second-person, we constrain the first argument of the VP's lexical head (i.e. the verb) to be second-person as well, thanks to the ARP. This, in turn, entails that in a structure like the following, Principle A will require a reflexive object to be coindexed with (and hence, by the AAP, to agree with) the second person subject:

[12] This analysis of imperatives is incomplete. In a larger grammar, it would need to be scaled up to include a semantic representation for the understood subject, as well as a constraint restricting imperatives to be stand-alone sentences. For more on imperatives and English clauses in general, see Ginzburg and Sag 2000.

[13] There are further constraints on what can be a 'stand alone' clause. In Chapter 9 we will require that the 'initial symbol' of our grammar must include the specification [FORM fin], which will distinguish past and present tense verbs (e.g. *went*, *loves*) from all others. FORM values for verbs are discussed in Chapter 8. Like the specification [COMPS ⟨ ⟩], this information will be supplied to the mother node of imperatives by the initial symbol

(42)



In this way, our treatment of imperatives interacts with our treatment of ARG-ST so as to provide an account of 'understood' arguments. The ARG-ST may include elements that are not overtly expressed, that is, which correspond to no overt phrase, and these can play a role in binding relations.

Note that we can use Binding Theory to confirm whether or not a given subjectless clause should involve an understood subject. For example, it would be a mistake to analyze exclamations of the form Damn NP along the lines just employed for imperatives. If we posited an understood subject NP in the ARG-ST of damn, it would license a reflexive pronoun (of the appropriate person, number, and gender) in the position after damn. But this is not possible:

(43) *Damn { myself / yourself / herself / himself / itself / themselves }!

Hence, damn in this use will have to be analyzed as being truly subjectless, in the sense that it has only one element in argument structure (and an empty SPR list). Examples like (43) are then ruled out because the reflexive element in the ARG-ST is not outranked by any coindexed element.

We have given a preview here of the analysis of verb forms that will be developed in the next chapter. There we will address the question of how the forms are differentiated formally, and how to manage the proliferation of entries for different forms of the same word.

7.7 The Argument Realization Principle Revisited

ARG-ST lists in general, and the ARP in particular, will play an increasingly important role in the chapters to come. We will place various constraints on the ARG-ST values of particular kinds of words, yet these would be vacuous without the ARP, which relates ARG-ST values to the values of the valence features SPR and COMPS. This connection is central, if the constraints we place on lexical heads are to interact with the elements that heads syntactically combine with. The Binding Theory presented in this chapter illustrates the importance of both ARG-ST and the ARP in our theory. Note that the order of arguments on the ARG-ST list also determines their linear order, given the way our grammar works. That is, subjects precede objects and other arguments, direct objects precede other arguments except the subject, and so forth. The ordering in (44) predicts the linear order that arguments occur in reasonably well:

(44) Subject $\succ$ Direct Object $\succ$ 2nd Object $\succ$ Other Complement

ARG-ST also has other uses that we cannot examine in detail here. Many grammarians have sought to explain various regularities exhibited by subjects, objects, and other syntactic dependents of the verb by making reference to the hierarchy in (44). For example, attempts to account for regularities about the semantic roles assigned to syntactic arguments (e.g. a more 'agent-like' argument of a verb will be linked to its subject argument) have led linguists to assume an ordering of the verb's arguments like the ARG-ST ordering. Such theories (which we regrettably cannot do justice to here) are often called LINKING THEORIES.

Various other phenomena have moved linguists to posit an ARG-ST hierarchy. One has to do with what is called 'relativization' i.e. using a clause to modify a noun. In these relative clauses, there is usually a 'gap' – that is, a missing NP that is understood as coreferential with the NP containing the relative clause. For example, in the following sentences, the bracketed portion is the relative clause, and the underlining indicates the location of the gap[14]:

(45) a. I met the person [who __ left].
b. I met the person [who they visited __].

It turns out that there are languages where only subjects can be 'relativized', i.e. where the analog of (45a) is grammatical, but the analog of (45b) is not: But there are apparently no human languages where the facts are the other way around, i.e. where (45b) is grammatical, but (45a) is not. These observations also extend to examples like (46):

(46) I met the person [to whom they handed a present __].

If a language allows (46), it will also allow both (45a) and (45b). The cross-linguistic generalization then is:

(47) If a language can relativize X, then it can relativize any element that outranks X.

In addition, there are languages where a verb agrees not only with its subject, but also with its direct object or with some other argument. An examination of the agreement systems of many of the world's languages, however, will reveal the following generalization to be true:

[14] We return to the analysis of such gaps in Chapter 14.

(48) If a language has words that show agreement with X, then it also has words that show agreement with the elements that outrank X.

Thus the ARG-ST hierarchy appears to have considerable motivation beyond the binding facts that we have used it to explain, some of it cross-linguistic in nature.

The ARP is simply a constraint on the type *word* and may be formulated as follows:

(49)
$$
\textit{word}: \left[\begin{array}{ll}
\text{SYN} & \left[\text{VAL} \left[\begin{array}{ll}\text{SPR} & \boxed{A} \\ \text{COMPS} & \boxed{B}\end{array}\right]\right] \\
\text{ARG-ST} & \boxed{A} \oplus \boxed{B}
\end{array}\right]
$$

This constraint interacts with other constraints in our grammar to give appropriate values to SPR and COMPS. For example, suppose we had a lexical entry for *loves* that specified nothing about SPR and COMPS, as in (50):[15]

(50)
$$
\left\langle \text{loves}, \left[\begin{array}{ll}
\multicolumn{2}{l}{\textit{word}} \\
\text{SYN} & \left[\text{HEAD} \quad \textit{verb}\right] \\
\text{ARG-ST} & \left\langle \begin{array}{c}\text{NP}_i \\ \left[\text{AGR} \quad \textit{3sing}\right]\end{array}, \text{NP}_j \right\rangle \\
\text{SEM} & \left[\begin{array}{ll}
\text{MODE} & \text{prop} \\
\text{INDEX} & s \\
 & \left\langle \left[\begin{array}{ll}\text{RELN} & \textbf{love} \\ \text{SIT} & s \\ \text{LOVER} & i \\ \text{LOVED} & j\end{array}\right] \right\rangle
\end{array}\right]
\end{array}\right] \right\rangle
$$

The effect of the ARP is to ensure that any word structure that (50) gives rise to will also satisfy further identity conditions, for example those indicated by the tags in (51):

(51)
$$
\left\langle \text{loves}, \left[\begin{array}{ll}
\multicolumn{2}{l}{\textit{word}} \\
\text{SYN} & \left[\begin{array}{ll}\text{HEAD} & \textit{verb} \\ \text{VAL} & \left[\begin{array}{ll}\text{SPR} & \langle \boxed{1} \rangle \\ \text{COMPS} & \langle \boxed{2} \rangle\end{array}\right]\end{array}\right] \\
\text{ARG-ST} & \left\langle \begin{array}{c}\boxed{1}\text{NP}_i \\ \left[\text{AGR} \quad \textit{3sing}\right]\end{array}, \boxed{2}\text{NP}_j \right\rangle \\
\text{SEM} & \left[\begin{array}{ll}
\text{MODE} & \text{prop} \\
\text{INDEX} & s \\
\text{RESTR} & \left\langle \left[\begin{array}{ll}\text{RELN} & \textbf{love} \\ \text{SIT} & s \\ \text{LOVER} & i \\ \text{LOVED} & j\end{array}\right] \right\rangle
\end{array}\right]
\end{array}\right] \right\rangle
$$

[15] In fact, as explained in the next chapter, lists like (50), consisting of a phonological form *loves* and a feature structure of type *word*, are to be derived by an inflectional rule.

However, given what we have said so far, (51) is not the only way for both of the elements of the argument structure list in (50) to be identified with complements. The ARP would also be satisfied if both $\boxed{1}$ and $\boxed{2}$ appeared on the COMPS list (with the SPR list empty). Similarly, both $\boxed{1}$ and $\boxed{2}$ could appear on the SPR list (with the COMPS list empty). Such possibilities will need to be ruled out. In the next chapter, we introduce a constraint requiring verbs to have exactly one element on their SPR lists. This will ensure that all words and word structures that satisfy (50) will in fact also satisfy (51).

7.8 Summary

This chapter has developed an acount of anaphoric binding – that is, the association of pronouns with antecedents – within our grammatical framework. We motivated two binding principles, one licensing elements like reflexives and reciprocals and the other restricting the possible coindexing of other NPs. Formalizing this led to a number of innovations, including the feature ARG-ST, the Argument Realization Principle, and the relation 'outrank'. We saw that prepositional phrases exhibit different binding patterns, depending on whether the prepositions serve simply as argument markers or introduce their own predications. Finally, we introduced a new grammar rule for imperative sentences.

7.9 Changes to the Grammar

Most of the changes to our grammar in the remainder of the book will be additions, rather than amendments of rules, principles, or other mechanisms we have already introduced. Hence, it would be redundant and somewhat tedious to have a full grammar summary at the end of each chapter. Instead, we end this chapter and most subsequent ones with a summary of what changes to the grammar we have introduced in the chapter. We will provide two more full grammar summaries: one in Chapter 9, and one in Appendix A.

In this chapter, we added a new value of the MODE feature ('ana'). The type constraint on *sem-cat* now looks like this:

$$
\textit{sem-cat}: \left[\begin{array}{ll}
\text{MODE} & \left\{\text{prop, ques, dir, ref, ana, none}\right\} \\
\text{INDEX} & \textit{index} \\
\text{RESTR} & \textit{list(predication)}
\end{array}\right]
$$

We also added a feature ARG-ST (appropriate for feature structures of type *word*) and the Argument Realization Principle (a constraint on the type *word*) which constrains the value of ARG-ST. The value of ARG-ST is a (possibly empty) list of *expressions*. The type constraint on *word* now looks like this:

$$
\textit{word}: \left[\begin{array}{ll}
\text{SYN} & \left[\text{VAL} \left[\begin{array}{ll}\text{SPR} & \boxed{A} \\ \text{COMPS} & \boxed{B}\end{array}\right]\right] \\
\text{ARG-ST} & \boxed{A} \oplus \boxed{B}
\end{array}\right]
$$

The Binding Theory itself consists of the definition of 'outrank' and two principles:

The definition of 'outrank':

(i) If a node is coindexed with its daughter, their feature structures are of equal rank.

(ii) If there is an ARG-ST list on which A precedes B, then *A* has a higher rank than (i.e. outranks) B.

The principles of the Binding Theory:

Principle A: A [MODE ana] element must be outranked by a coindexed element.

Principle B: A [MODE ref] element must not be outranked by a coindexed element.

To account for the agreement between pronouns and their antecedents, we introduced a further principle:

The Anaphoric Agreement Principle (AAP): Coindexed NPs agree.

We also introduced a distinction between predicational and argument-marking prepositions, and an analysis of argument-marking prepositions by means of lexical entries with the following specifications:

$$\begin{bmatrix} \text{ARG-ST} & \left\langle \overset{\text{NP}}{\begin{bmatrix} \text{SEM} & \begin{bmatrix} \text{MODE} & \boxed{1} \\ \text{INDEX} & \boxed{2} \end{bmatrix} \end{bmatrix}} \right\rangle \\ \text{SEM} & \begin{bmatrix} \text{MODE} & \boxed{1} \\ \text{INDEX} & \boxed{2} \\ \text{RESTR} & \langle\ \rangle \end{bmatrix} \end{bmatrix}$$

Finally, we introduced a new grammar rule, the Imperative Rule:

$$\begin{bmatrix} \textit{phrase} & \\ \text{HEAD} & \textit{verb} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle\ \rangle \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \text{MODE} & \text{dir} \\ \text{INDEX} & s \end{bmatrix} \end{bmatrix} \rightarrow \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \textit{verb} & \\ \text{FORM} & \text{base} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \left\langle \text{NP}\begin{bmatrix} \text{PER} & \text{2nd} \end{bmatrix} \right\rangle \\ \text{COMPS} & \langle\ \rangle \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \text{INDEX} & s \end{bmatrix} \end{bmatrix}$$

7.10 Further Reading

The binding of anaphors has been the topic of an extensive literature since the late 1960s. A seminal and very readable paper is Lasnik 1976. To our knowledge, the first proposal to treat reflexive binding in terms of a hierarchy of the verb's arguments was made by Johnson (1977). The Binding Theory of Chomsky (1981) distilled many of the insights of the research of the preceding decade into three principles; this theory was developed further in a number of works within the Government and Binding Theory of grammar.

A detailed account of binding within Lexical Functional Grammar is presented by Dalrymple (1993). The theory of binding presented in this chapter is based on Pollard and Sag 1992, 1994 with terminological revision ('(out)ranking') due to Bresnan (1995). One of the most detailed attempts to date at formulating a linking theory compatible with the approach presented here is by Davis (2001), whose theory of the alignment of semantics and argument structure allows a further streamlining of all our lexical descriptions. The Argument Structure hierarchy (44) is often referred to as the 'Keenan-Comrie' Hierarchy, because of the pioneering work on this topic reported in Keenan and Comrie 1977.

7.11 Problems

⚠ Problem 1: Classifying Prepositions

We have divided prepositions into two sorts: those functioning as predicates and those functioning as argument-markers. For each of the following sentences,

(a) classify the italicized preposition into one of these two sorts (or as being ambiguously both); and

(b) justify your classification by showing (with acceptable and/or unacceptable sentences) what reflexive and nonreflexive coreferential pronouns can or cannot appear as the preposition's object.

(i) The dealer dealt an ace *to* Bo.

(ii) The chemist held the sample away *from* the flame.

(iii) Alex kept a loaded gun *beside* the bed.

(iv) We bought flowers *for* you.

(v) The car has a scratch *on* the fender.

Problem 2: Imperative 'Subjects'

There are imperative sentences that contain an NP that looks like it is the subject of the [FORM base] VP:

(i) You get out of here!

(ii) Everybody take out a sheet of paper!

But the initial NPs in these examples don't seem to participate in the normal agreement pattern with respect to reflexive pronouns. For example, we know that an NP like *everybody* is third person because of its behavior in (iii):[16]

(iii) $\text{Everybody found} \left\{ \begin{array}{l} \text{?himself} \\ \text{*yourself} \\ \text{?themselves} \\ \text{*myself} \end{array} \right\} \text{a seat.}$

[16] Following standard practice of generative grammarians, we use designations '?', '??', and '?*' to indicate different levels of naturalness between full acceptability and complete unacceptability.

Yet in imperative sentences, we still find the second-person reflexive pattern illustrated in (iv):

(iv) Everybody find { ??himself / yourself / ??themselves / *myself } a seat!

Assuming that we do not want to license examples marked '??', what minimal modification of the Imperative Rule would account for the indicated data? Make sure that your proposal still accounts for all relevant facts illustrated above for imperative sentences with no initial NP. For the purposes of this problem, don't worry about the semantics: concentrate on providing a syntactic analysis that will get the binding facts right.

Problem 3: Principle A Revisited

Picking up on an idea from Problem 2 of Chapter 1, we hinted at a couple of places in this chapter that the English reciprocal form each other might be [MODE ana] – that is; that it might obey Principle A of the Binding Theory. One immediate obstacle to this suggestion is raised by examples like (i):

(i) They acknowledged each other's contributions.

A. Explain why our current formulation of Principle A together with the assumption that each other is [MODE ana] makes the wrong prediction about (i).

At first glance, (i) might be taken to show that reciprocals are not subject to Principle A, but another possibility is that Principle A isn't formulated quite right. It turns out that there are also cases involving reflexives that do not obey Principle A:

(ii) Clinton is writing a book about himself.
(iii) We heard that embarrassing pictures of ourselves had been posted on the internet.
(iv) Pat asked Chris where they had filed the descriptions of themselves.
(v) Pat told Chris to send reminders about the meeting to everyone on the distribution list, with the exception of themselves.

Such data suggest that our formulation of Principle A is in need of revision. We could try to expand the coverage of Principle A, so that it covers such examples. But that approach does not look very promising, particularly for examples (iv) and (v). In those sentences, there is no single NP that serves as the antecedent of the reflexive. Rather, the reflexives in those examples refer to a set consisting of Pat and Chris. This indicates that determining the reference of the reflexive pronouns in these cases is not purely a matter of grammar, but involves some pragmatic inference. Consequently, it seems that the best way to deal with these counterexamples to our current Principle A is to restrict its applicability – that is, to make examples like (ii)–(v) EXEMPT from Principle A.

In doing so, however, we must be careful not to exempt too many anaphors. For example, we want Principle A to continue to account for the distinction in well-formedness between (vi) and (vii):

(vi) They read Mary's story about herself.
(vii) *They read Mary's story about themselves.

B. Reformulate Principle A so that it does not rule out (ii)–(vi), but does rule out (vii). Your formulation should likewise not rule out (i) on the assumption that each other is [MODE ana]. [Hint: Look at what *kinds* of elements (if any) outrank the [MODE ana] elements in (i)–(v), and restrict the applicability of Principle A to cases that have suitable potential antecedents. Note that the objective is simply to remove examples like (i)–(v) from the coverage of Principle A; we are assuming that the generalization that determines *how* such 'exempt' reflexives and reciprocals are interpreted is outside the domain of grammar.]

If Principle A is reformulated so as not to block (i), then it will also fail to block examples like (viii).

(viii) *You acknowledged yourself's contribution.

Let us assume the analysis of the English possessive introduced in Chapter 6, Problem 4 – that is, that *'s* is a determiner that takes an obligatory NP specifier. Notice that not all kinds of NPs can serve as specifiers for *'s*; in particular, the forms **I's*, **me's*, **you's*, *he's, **him's*, **she's*, *her's, **we's*, **us's*, **they's*, and **them's* are all ill-formed possessive determiner phrases.

C. Formulate a generalization about the possible specifiers of *'s* that will rule out (viii), independent of any facts about binding. How would this be stated formally? [*Hint:* You will need to posit a new feature (call it 'PRO') that distinguishes the kinds of NPs that cannot be specifiers of *'s* and those that can. The formal statement will *involve* the SPR value of 's.]

Your reformulation of Principle A probably also exempted examples like (ix) and (x) from its domain. (If it didn't, you should double-check to make sure that its predictions are consistent with (i)–(viii); if so, then you may have discovered a new analysis).

(ix) *Himself is to blame.
(x) *They believe that themselves will win.

D. Suggest a generalization about reflexive pronouns that will rule out (vii) and (viii) (again, without relying on binding). [Hint: Notice that the forms are himself and themselves, not **heself* or **theyself*.] How would this generalization be stated formally?

Finally, the reformulation of Principle A to exempt reflexives like those in (ii)–(v) creates problems for the analysis we gave of predicational prepositions. In particular, Principle A will no longer rule out examples like (xi) (repeated from (27a)):

(xi) *The house had a fence around itself.

E. Explain why the reflexive in (xi) is no longer ruled out.

Later in the book, we will introduce formal machinery that will allow us to bring examples like (xi) back within the purview of Principle A.

Sag. I et al. 2003. Syntactic Theory

10

The Passive Construction

10.1 Introduction

Perhaps the most extensively discussed syntactic phenomenon in generative grammar is the English passive construction. The active/passive alternation provided one of the most intuitive motivations for early transformational grammar, and it has played a role in the development of almost all subsequent theories of grammar.

In this chapter, we present an account of the English passive using the formal mechanisms we have developed in this text. Given the strongly lexical orientation of our theory, it should come as no surprise that we treat the active/passive relationship primarily as a relationship between two verb forms, and that we use a lexical rule to capture the generality of that relationship.

We begin with some data to exemplify the phenomenon in question. We then formulate our rule and explain how it works. Finally, we turn to the question of the status of the forms of the verb *be* that characteristically occur in passive sentences.

10.2 Basic Data

Consider sets of sentences (and nonsentences) like the following:

(1) a. The dog bit the cat.
b. The cat was bitten (by the dog).
c.*The cat was bitten the mouse (by the dog).

(2) a. Pat handed Chris a note.
b. Chris was handed a note (by Pat).
c.*Chris was handed Sandy a note (by Pat).

(3) a. TV puts dumb ideas in children's heads.
b. Dumb ideas are put in children's heads (by TV).
c.*Dumb ideas are put notions in children's heads (by TV).

The b-sentences in (1)–(3) are what are standardly called 'passive'; the a-sentences are referred to as their 'active' counterparts. There is clearly a close semantic relationship between active and passive pairs. In particular, the semantic roles of the arguments are the same – in (1), the dog is the biter, and the cat is the one being bitten. To put it informally, in an active sentence and its passive counterpart, 'who does what to whom' is the same. The crucial difference between active and passive sentences is that the subject of the passive corresponds to the object of the active. The participant denoted by the subject of the active, if expressed at all in the passive, is referred to by the object of the preposition by. Consequently, the verb in a passive sentence always has one less object (that is, NP complement) than the verb in its active counterpart. This is illustrated in the c-sentences of (1)–(3). It follows that sentences with intransitive verbs, like (4a), normally do not have passive counterparts, as in (4b):

(4) a. The patient died.
b.*The patient was died (by the doctor).
c.*The doctor died the patient.

Moreover, aside from this one difference, active verbs and their corresponding passives have identical valence requirements. This is illustrated in (5), where the absence of an obligatory complement renders both the active and passive examples ungrammatical:

(5) a. Pat handed Chris *(a note).
b. Chris was handed *(a note) (by Pat).
c. TV puts dumb ideas *(into their heads).
d. Dumb ideas are put "(into their heads) (by TV).

10.3 The Passive Lexical Rule

It would not be hard to formulate lexical entries for passive forms of verbs. To capture the generalizations stated informally above, however, we need to formulate a rule that can relate actives and passives. As was the case with the rules discussed in Chapter 8, our passive rule is motivated by more than just parsimony. Faced with novel transitive verbs – either new coinages like *email* or rare words like *cark* – English speakers can (and often do) immediately use them correctly in passive sentences. Hence a rule-governed treatment of the active/passive alternation will be psychologically more realistic than a mere listing of the passive forms for all transitive verbs.

Intuitively, then, we want a rule that does the following:

- turns the first NP complement into the subject;
- allows the subject either to turn into the object of a PP headed by by or to be omitted altogether;
- leaves the valence features otherwise unchanged;
- leaves the semantics unchanged; and
- makes the appropriate morphological change in the form of the verb.

This last item is one we have not mentioned until this point. A moment's reflection should reveal that the morphology of the passive form of a verb (or 'passive participle', as it is commonly called) is always identical to that of the past participle; this is especially clear if we consider verbs with exceptional past participles, such as do (done), sink (sunk) and cut (cut). This generalization is captured easily in our framework by invoking the same morphological function, F_{PSP}, for both the Past Participle Lexical Rule and the Passive Lexical Rule.

Before writing the Passive Lexical Rule, we need to decide what type of *l-rule* it is. The morphology of English passives is inconclusive on this point: no further affixes attach to passives. As far as the morphology is concerned, the rule could be either an *i-rule* or a *d-rule*. However, the syntactic aspects of passive are only consistent with the constraints on *d-rules*. Recall from Chapter 8 that the constraints on inflectional rules (*i-rules*) and derivational rules (*d-rules*) are as in (6) and (7), respectively.

(6)
$$\textit{i-rule}: \begin{bmatrix} \text{INPUT} & \left\langle \text{X}\,, \begin{bmatrix} \textit{lexeme} & \\ \text{SYN} & \boxed{3} \\ \text{ARG-ST} & \boxed{A} \end{bmatrix} \right\rangle \\ \\ \text{OUTPUT} & \left\langle \text{Y}\,, \begin{bmatrix} \textit{word} & \\ \text{SYN} & \boxed{3} \\ \text{ARG-ST} & \boxed{A} \end{bmatrix} \right\rangle \end{bmatrix}$$

(7)
$$\textit{d-rule}: \begin{bmatrix} \text{INPUT} & \left\langle \text{X}\,, \begin{bmatrix} \textit{lexeme} & \\ \text{SYN} & /\boxed{3} \end{bmatrix} \right\rangle \\ \\ \text{OUTPUT} & \left\langle \text{Y}\,, \begin{bmatrix} \textit{lexeme} & \\ \text{SYN} & /\boxed{3} \end{bmatrix} \right\rangle \end{bmatrix}$$

In order to change the subject and complements, the passive rule must specify either different SPR and COMPS values or different ARG-ST values on the INPUT and OUTPUT. The passive rule given immediately below specifies different ARG-ST values, but either strategy would be inconsistent with the constraints on *i-rule*. Therefore, given our theory of inflectional and derivational rules, passive must be a derivational rule.[1]

The following is a lexical rule that satisfies the desiderata given above:

(8) Passive Lexical Rule

$$\begin{bmatrix} \textit{d-rule} & \\ \text{INPUT} & \left\langle \boxed{1}\,, \begin{bmatrix} \textit{tv-lxm} & \\ \text{ARG-ST} & \langle\, [\text{INDEX}\ i]\,\rangle \oplus \boxed{A} \end{bmatrix} \right\rangle \\ \text{OUPUT} & \left\langle \text{F}_{PSP}(\boxed{1})\,, \begin{bmatrix} \textit{part-lxm} & \\ \text{SYN} & \begin{bmatrix} \text{HEAD} & [\text{FORM}\ \ \text{pass}\,] \end{bmatrix} \\ \text{ARG-ST} & \boxed{A} \oplus \left\langle \left(\begin{matrix} \text{PP} \\ \begin{bmatrix} \text{FORM} & \text{by} \\ \text{INDEX} & i \end{bmatrix} \end{matrix} \right) \right\rangle \end{bmatrix} \right\rangle \end{bmatrix}$$

There are several points of explanation that need to be made here.

[1]French again confirms this conclusion: There are four inflected forms of any given passive participle, the choice depending on the number and gender of the participle's subject NP. This indicates that the passivization rule in French feeds into various inflectional rules, and hence must be derivational.

First, like the present and past participle lexical rules, the OUTPUT of this rule is of type *part(iciple)-lxm*. This is a subtype of const-lxm, so passive participles, like other participles, undergo the Constant Lexeme Lexical Rule. The only effect of the Constant Lexeme Lexical Rule is to change the type of the second member of the lexical sequence to *word*. The type word, however, is constrained to satisfy the Argument Realization Principle. As such, OUTPUTs of the Constant Lexeme Lexical Rule will be subject to the Argument Realization Principle (Chapter 7).

Second, notice that most of the effects of the rule (which applies to any lexeme belonging to a subtype of *tv-lxm*) are in the ARG-ST. At a coarse level of description, what the rule does is rearrange the elements of the ARG-ST list. Because of the ARP, these rearrangements also affect the values of the valence features. Specifically, (8) makes the second element (corresponding to the direct object) of the input ARG-ST list be the first element (corresponding to the subject) of the output's ARG-ST list. Whatever follows the second element in the input also moves up in the list. (8) also adds a PP to the end of the ARG-ST list. The specification [FORM by] on this PP indicates that the PP must be headed by the preposition by. We will abbreviate 'PP[FORM by]' as 'PP[by]' (and similarly with other values of FORM). Hence a verbal lexeme with an argument structure like (9a) will give rise to a passive lexeme whose argument structure is (9b):

(9) a. $\left[\text{ARG-ST} \;\; \langle \text{NP}_i \,,\, \text{NP} \,,\, \text{PP[to]} \rangle\right]$ (send, give, fax...)

b. $\left[\text{ARG-ST} \;\; \langle \text{NP} \,,\, \text{PP[to]} \; (, \text{PP[by]}_i) \rangle\right]$ (sent, given, faxed...)

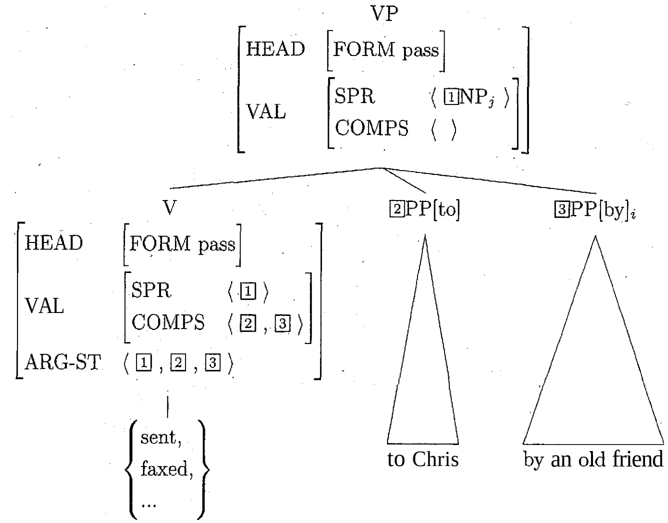
After going through the Constant Lexeme Lexical Rule, (9b) licenses two basic kinds of word structure, both constrained by the ARP. These are shown in (10):

(10) a. $\begin{bmatrix} \text{VAL} & \begin{bmatrix} \text{SPR} & \langle \boxed{1} \rangle \\ \text{COMPS} & \langle \boxed{2} \rangle \end{bmatrix} \\ \text{ARG-ST} & \langle \boxed{1}\text{NP}_j \,,\, \boxed{2}\text{PP[to]} \rangle \end{bmatrix}$ (sent, given, faxed...)

b. $\begin{bmatrix} \text{VAL} & \begin{bmatrix} \text{SPR} & \langle \boxed{1} \rangle \\ \text{COMPS} & \langle \boxed{2} \,,\, \boxed{3} \rangle \end{bmatrix} \\ \text{ARG-ST} & \langle \boxed{1}\text{NP}_j \,,\, \boxed{2}\text{PP[to]} \,,\, \boxed{3}\text{PP[by]}_i \rangle \end{bmatrix}$ (sent, given, faxed...)

Hence passive words will automatically give rise to passive VPs like (11), thanks to the Head-Complement Rule (and the HFP and the Valence Principle):

(11)



In other words, once our lexicon has passive words, our grammar already guarantees that we will have the appropriate passive VPs. These VPs can be selected as a complement by a few verbs, most notably be:

(12) A message [was [sent to Chris by an old friend]].

A third noteworty property of the Passive Lexical Rule concerns indices. Recall that subscripts indicate values of the feature INDEX; so (8) says that the optional PP[by] in the rule output has an index that is coindexed with the subject of the lexical rule input. This means that whatever semantic role the verbal lexeme assigns to its subject will be assigned to the INDEX value of the PP[by] of the passive word, and hence (since by is an argument-marking preposition) to the prepositional object within the PP[by] (see below). Likewise, since the verbal lexeme's object – the first element in the list $\boxed{1}$ – is identified with the subject of the passive word, it follows that the index of the subject of the passive word is the same as that of the verbal lexeme's direct object. Therefore, since the semantics remains unchanged by this lexical rule (because the rule says nothing to override the effect of the defeasible identity constraint), the semantic role of the active object will be the same as that of the passive subject. The overall result of this rule, then, is to shift the role assignments from subject to PP[by] and from object to subject.

Fourth, note that the passive rule does not mention case at all. Verbal lexemes do not specify CASE values for any of their arguments (in English); hence, though the lexeme's object NP becomes the subject of the corresponding passive participle, there is no need to 'unassign' an accusative case specification. All nonsubject arguments of verbs must be accusative, but the constraint that guarantees this (namely, the Case Constraint – see Chapter 8, Section 8.4.5) applies to lexical trees (word structures), not to lexemes. (See the definition of lexical licensing in Chapter 9, Section 9.2.7.) Nor does the passive rule assign nominative case to the first argument of the rule output, as one might expect on the basis of examples like (13):

(13) a. He was arrested by the police.
b.*Him was arrested by the police.

The nominative case of the subject in examples like (13) is determined by the auxiliary verb was, whose SPR value is identified with that of the passive VP, as discussed in the next section. There are in fact instances of passive verbs whose subjects are not nominative, as in (14).

(14) $\left\{ \begin{matrix} \text{Him} \\ \text{*He} \end{matrix} \right\}$ being arrested by the police upset many people.

Our passive rule achieves the desired effect in such instances by leaving the subject of the passive word unspecified for CASE. Hence, whatever case requirements the particular grammatical context imposes will determine the CASE value of a passive verb's subject.[2]

Fifth, the rule says that passive verbs are constrained to be [FORM pass].[3] The justification for having a separate value 'pass' for the FORM of passive verbs has not yet been provided; this will be addressed in the next section.

Returning to the use of the FORM feature on the PP in (8), recall that FORM has so far been used primarily for distinguishing among verb forms. But in the Agent Nominalization Lexical Rule presented in Chapter 8, we already made use of the FORM feature on PPs: a PP specified as [FORM of] was meant to be one that could only be headed by the preposition of. In fact, we want to employ the feature FORM more generally, to mark the choice of preposition in other contexts as well. Since the set of prepositions in English is a relatively small, closed set, we might (in the limiting case) have a separate value of FORM for each preposition. In this book, we'll use only the following FORM values for prepositions:

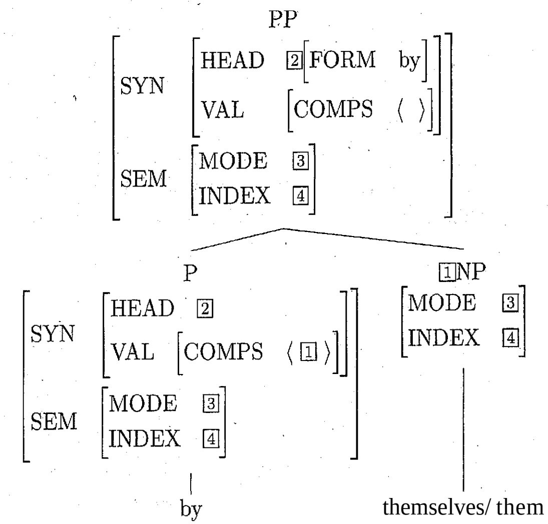
(15) of, by, to, at, in, about, on, for

Having FORM values for prepositions allows us, for example, to represent the fact that the verb rely requires a PP complement headed by either on or upon. The FORM value of the lexical preposition will be shared by the entire PP (since FORM is a head feature and hence is governed by the Head Feature Principle), as shown in the tree for a by-phrase sketched in (16):

[2] Verbal gerunds like *being* in (14), for example, might lexically specify the case of their subject (which is identified with the subject of the passive participle in (14)).

[3] Note that the passive rule, like other lexical rules applying to verbs, isn't changing the FORM value, but rather further specifying it, as verbal lexemes are generally underspecified for FORM.

(16)



Crucially, we assume *by* is an argument-marking preposition whose INDEX and MODE values are identified with those of its NP object. Thus whatever index the passive participle assigns to the PP[by] complement will be identified with the index of the NP object within that PP.

The effect of the Passive Lexical Rule, then, is to map lexemes like (17) into lexemes like (18):[4]

(17)
$$\left\langle \text{love}\ , \begin{bmatrix} \textit{stv-lxm} & \\ \text{SYN} & \begin{bmatrix}\text{HEAD} & \begin{bmatrix}\textit{verb} & \\ \text{AGR} & \boxed{1}\end{bmatrix} \\ \text{VAL} & \begin{bmatrix}\text{SPR} & \langle [\text{AGR}\ \boxed{1}] \rangle\end{bmatrix}\end{bmatrix} \\ \text{ARG-ST} & \langle\ \text{NP}_i\ ,\ \text{NP}_j\ \rangle \\ \text{SEM} & \begin{bmatrix}\text{INDEX} & s \\ \text{RESTR} & \left\langle \begin{bmatrix}\text{RELN} & \textbf{love}\\ \text{SIT} & s\\ \text{LOVER} & i \\ \text{LOVED} & j\end{bmatrix}\right\rangle\end{bmatrix}\end{bmatrix}\right\rangle$$

[4] (17)–(19) represent families of lexical sequences, each of which contains more information than is shown. The optionality of the PP in (18) and (19) is just another kind of underspecification in the description. Each of the fully resolved lexical sequences that make up these families will have a fully resolved value for ARG-ST. Some will have ARG-ST values with the PP and some will have ARG-ST values without it.

(18)
$$\left\langle \text{loved}\ , \begin{bmatrix} \textit{part-lxm} & \\ \text{SYN} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \textit{verb} & \\ \text{AGR} & \boxed{1} \\ \text{FORM} & \text{pass} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle [\text{AGR}\ \boxed{1}] \rangle \end{bmatrix} \end{bmatrix} \\ \text{ARG-ST} & \left\langle \text{NP}_j \left(, \begin{matrix} \text{PP} \\ \begin{bmatrix} \text{FORM} & \text{by} \\ \text{INDEX} & i \end{bmatrix} \end{matrix} \right) \right\rangle \\ \text{SEM} & \begin{bmatrix} \text{INDEX} & s \\ \text{RESTR} & \left\langle \begin{bmatrix} \text{RELN} & \textbf{love} \\ \text{SIT} & s \\ \text{LOVER} & i \\ \text{LOVED} & j \end{bmatrix} \right\rangle \end{bmatrix} \end{bmatrix} \right\rangle$$

The Constant Lexeme Lexical Rule then maps lexemes like (18) into words like (19):

(19)
$$\left\langle \text{loved}\ , \begin{bmatrix} \textit{word} & \\ \text{SYN} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \textit{verb} & \\ \text{AGR} & \boxed{1} \\ \text{FORM} & \text{pass} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle \boxed{2}[\text{AGR}\ \boxed{1}] \rangle \\ \text{COMPS} & \boxed{B} \end{bmatrix} \end{bmatrix} \\ \text{ARG-ST} & \left\langle \boxed{2}\text{NP}_j \right\rangle \oplus \boxed{B} \left\langle \left(, \begin{matrix} \text{PP} \\ \begin{bmatrix} \text{FORM} & \text{by} \\ \text{INDEX} & i \end{bmatrix} \end{matrix} \right) \right\rangle \\ \text{SEM} & \begin{bmatrix} \text{INDEX} & s \\ \text{RESTR} & \left\langle \begin{bmatrix} \text{RELN} & \textbf{love} \\ \text{SIT} & s \\ \text{LOVER} & i \\ \text{LOVED} & j \end{bmatrix} \right\rangle \end{bmatrix} \end{bmatrix} \right\rangle$$

Note that the effect of the ARP is seen in (19), since these lexical sequences involve *words*.

10.4 The Verb *Be* in Passive Sentences

What about the forms of *be*, which in all of our examples (so far) immediately precede the passive participle? The first thing to observe is that passive participles can also occur in environments that lack any form of *be*. Some examples are given in (20):

(20) a. The cat got bitten (by the dog).
b. Liked by many people but respected by few, Jean will have to run an aggressive reelection campaign.
c. Anyone handed a note will be watched closely.

Hence, though some form of *be* is typical in passive sentences, it would have been a mistake to try to build it into the rule introducing the passive form of verbs. Rather, we need to provide an analysis of the relevant lexical entry for *be* that links its occurrence to the presence of a passive participle.[5]

More precisely, our analysis needs to say that the passive *be* takes a complement that is a VP[FORM pass] like the one shown in (11) above. This means that the ARG-ST list of the lexeme *be* contains both an NP subject and a VP[FORM pass]. A few points are worth noting here. First, this is the first time we have considered VP arguments/complements in detail, though our Head-Complement Rule permits them, as we saw earlier (see Section 8.5.1 of Chapter 8). We will see many more examples of VP complements soon. Second, since FORM is a head feature, a verb's FORM value will show up on its mother VP node. Hence if a verb like *be* selects a VP[FORM pass] complement, that is sufficient to guarantee that the complement's head daughter will be a V[FORM pass].

The trickiest and most important aspect of our analysis of *be* in passives is how we deal with the subject (i.e. with the value of SPR). In a sentence like (1b), repeated here as (21a), the agreement indicates that *the cat* should be treated as the subject (that is, the SPR) of *was*:

(21) a. The cat was bitten by the dog.
b.*The cat were bitten by the dog.

This is further supported by the unacceptability of (21b). But in our discussion of passive participles in the previous section, we discussed the *cat* as the subject of *bitten*. This was necessary for semantic reasons (i.e. to ensure that the cat functions semantically as the thing bitten, rather than as the biter), and to capture the correspondence between the valence values of the active and passive forms.

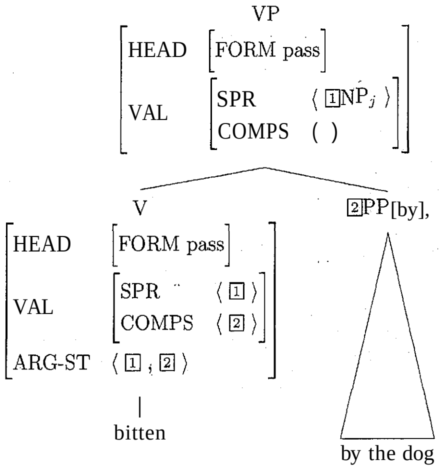
Our analysis provides a unified account of both these observations by identifying the subject of *be* with the subject of the passive verb. That is, there is only one subject NP in the sentence, but it is identified with the first member of the ARG-ST list of both *be* and the passive verb. As the subject of *be*, it is required to satisfy the agreement constraints imposed by the relevant inflected form of *be*, i.e. *was* in (21a). As the subject of the passive verb, it will also be assigned the semantic role that the object NP would take in an active sentence (the BITTEN role, rather than the BITER role that an active

[5] We'll return to the issue of whether we can analyze other uses of *be* in terms of this same lexical entry in Chapter 11.

form of *bite* would assign to its subject).

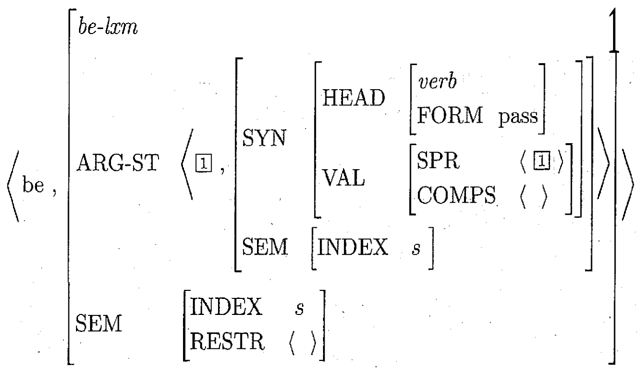
How exactly do we identify the subject of was with the subject of the passive verb bitten? First of all, it is important to see that half the job has already been accomplished by the Valence Principle, which requires that in a structure like (22), the SPR value of the passive verb is identical with that of the passive VP:

(22)



To represent the fact that be and its passive VP complement share the same subject, we need only add a constraint (using the familiar device of tagging) which specifies that the first argument of be (its subject) is identical to the SPR value of its VP[FORM pass] argument. We can now formulate the lexical entry for the passive be as follows:

(23)



What this entry says is that be belongs to a new type be-lxm (a subtype of verb-km whose properties do not yet concern us) and takes a VP argument specified as [FORM pass]. In addition, this be says that its subject must be the same as its complement's subject. This means that the subject of the sentence will also serve as the subject of the verb that heads the complement VP, according to the Valence Principle. And because be adds nothing to the meaning except the information that the complement's INDEX value is the same as that of be, (23) also guarantees that the semantics of the verb phrase headed by be is identical to the semantics of be's VP complement. (Note that be-lxm inherits the constraint [MODE prop] from the type verb-lxm.)

We will see in the next two chapters that the idea of having a verb and its argument share a subject is extremely useful in describing a number of phenomena. In Chapter 13, we will see in addition how using lexical types can simplify lexical entries such as these.

Exercise 1: Shared Subjects

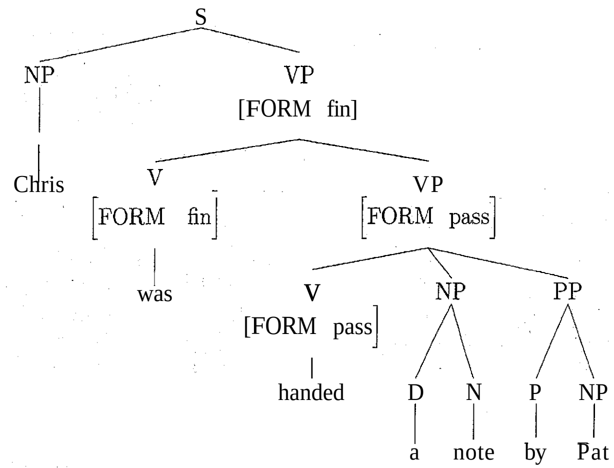
Why doesn't the lexical entry in (23) license sentences like (i)?

(i)*A cat was a cat bitten by the dog.

10.5 An Example

We conclude this chapter with a detailed analysis of example (2b). The phrase structure we need to license is the following:

(24)



In this phrase structure, the word was is part of a family of lexical sequences constrained as shown in (25):

(25)
$$\left\langle \text{was}\ , \left[\begin{array}{ll} \textit{word} & \\ \text{SYN} & \left[\begin{array}{ll} \text{HEAD} & \left[\begin{array}{ll} \textit{verb} & \\ \text{AGR} & \boxed{5} \\ \text{FORM} & \text{fin} \end{array}\right] \\ \text{VAL} & \left[\begin{array}{ll} \text{SPR} & \langle \boxed{1}[\text{AGR } \boxed{5}] \rangle \\ \text{COMPS} & \langle \boxed{2} \rangle \end{array}\right] \end{array}\right] \\ \text{ARG-ST} & \left\langle \left[\text{AGR} \ \ \textit{3sing}\right] , \boxed{2}\left[\begin{array}{ll} \text{HEAD} & \left[\begin{array}{ll} \textit{verb} & \\ \text{FORM} & \text{pass} \end{array}\right] \\ \text{VAL} & \left[\begin{array}{ll} \text{SPR} & \langle \boxed{1} \rangle \\ \text{COMPS} & \langle\ \rangle \end{array}\right] \\ \text{INDEX} & s \end{array}\right] \right\rangle \\ \text{SEM} & \left[\begin{array}{ll} \textbf{MODE} & \text{prop} \\ \text{INDEX} & s \\ \textbf{RESTR} & \langle \dots \rangle \end{array}\right] \end{array}\right] \right\rangle$$

This is the same as (23), except that it includes constraints contributed by the Past-Tense Verb Lexical Rule. In particular (25) ensures that *was* is finite (i.e. [FORM fin]) and that it has past-tense semantics (suppressed here) and a third-person singular subject.[6] Note that the subject in (25) is identical to the complement's subject (as was the case in (23)). Further, the verb's SPR value is constrained to be identical to the first member of the ARG-ST list. This, together with the COMPS value, is the result of the ARP, which (25) must obey.

So now let us consider more closely the VP[pass], whose head is the passive participle *handed*. The lexical entry for hand is the following:

(26)
$$\left\langle \text{hand}\ , \left[\begin{array}{ll} \textit{dtv-lxm} & \\ \text{ARG-ST} & \langle X_i , Y_j , Z_k \rangle \\ \text{SEM} & \left[\begin{array}{ll} \text{INDEX} & s \\ \text{RESTR} & \left\langle \left[\begin{array}{ll} \text{RELN} & \textbf{hand} \\ \text{SIT} & s \\ \text{HANDER} & i \\ \text{RECIPIENT} & j \\ \text{HANDED} & k \end{array}\right] \right\rangle \end{array}\right] \end{array}\right] \right\rangle$$

[6]The verb *be* is unique among English verbs in distinguishing different forms (*was* and ***were***) in the past tense. See note **34** of Chapter 8.

The lexical sequences satisfying this lexical entry all obey (27):

(27)
$$\left\langle \text{hand}\ , \left[\begin{array}{ll} \textit{dtv-lxm} & \\ \text{SYN} & \left[\begin{array}{ll} \text{HEAD} & \left[\begin{array}{ll} \textit{verb} & \\ \text{AGR} & \boxed{6} \end{array}\right] \\ \text{VAL} & \left[\text{SPR} \ \ \langle [\text{AGR } \boxed{6}] \rangle\right] \end{array}\right] \\ \text{ARG-ST} & \langle NP_i , NP_j , NP_k \rangle \\ \text{SEM} & \left[\begin{array}{ll} \text{MODE} & \text{prop} \\ \text{INDEX} & s \\ \text{RESTR} & \left\langle \left[\begin{array}{ll} \text{RELN} & \textbf{hand} \\ \text{SIT} & s \\ \text{HANDER} & i \\ \text{RECIPIENT} & j \\ \text{HANDED} & k \end{array}\right] \right\rangle \end{array}\right] \end{array}\right] \right\rangle$$

In addition, they may undergo the Passive Lexical Rule, yielding lexical sequences like the following:

(28)
$$\left\langle \text{handed}\ , \left[\begin{array}{ll} \textit{part-lxm} & \\ \text{SYN} & \left[\begin{array}{ll} \text{HEAD} & \left[\begin{array}{ll} \textit{verb} & \\ \text{AGR} & \boxed{6} \\ \text{FORM} & \text{pass} \end{array}\right] \\ \text{VAL} & \left[\text{SPR} \ \ \langle [\text{AGR } \boxed{6}] \rangle\right] \end{array}\right] \\ \text{ARG-ST} & \left\langle NP_j , NP_k \left(, \begin{array}{c} \text{PP} \\ \left[\begin{array}{ll} \text{FORM} & \text{by} \\ \text{INDEX} & i \end{array}\right] \end{array} \right) \right\rangle \\ \text{SEM} & \left[\begin{array}{ll} \text{MODE} & \text{prop} \\ \text{INDEX} & s \\ \text{RESTR} & \left\langle \left[\begin{array}{ll} \text{RELN} & \textbf{hand} \\ \text{SIT} & s \\ \text{HANDER} & i \\ \text{RECIPIENT} & j \\ \text{HANDED} & k \end{array}\right] \right\rangle \end{array}\right] \end{array}\right] \right\rangle$$

And these may undergo the Constant Lexeme Lexical Rule to give sequences like (29): (Note that as words, these are subject to the ARP.)

(29)

$$\left\langle \text{handed}\ , \begin{bmatrix} \textit{word} & \\ \text{SYN} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \textit{verb} & \\ \text{AGR} & \boxed{6} \\ \text{FORM} & \text{pass}\end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle \boxed{1}[\text{AGR}\ \boxed{6}] \rangle \\ \text{COMPS} & \boxed{B} \end{bmatrix}\end{bmatrix} \\ \text{ARG-ST} & \langle \boxed{1}\text{NP}_j \rangle \oplus \boxed{B} \left\langle \text{NP}_k \left(, \overset{\text{PP}}{\begin{bmatrix} \text{FORM} & \text{by} \\ \text{INDEX} & i\end{bmatrix}}\right)\right\rangle \\ \text{SEM} & \begin{bmatrix} \text{MODE} & \text{prop} \\ \text{INDEX} & \text{s} \\ \text{RESTR} & \left\langle \begin{bmatrix} \text{RELN} & \text{hand} \\ \text{SIT} & s \\ \text{HANDER} & i \\ \text{RECIPIENT} & j \\ \text{HANDED} & k \end{bmatrix}\right\rangle \end{bmatrix}\end{bmatrix}\right\rangle$$

Lexical sequences like (29) form the basis for word structures like (30), where the optionality of the PP is resolved, and the Case Constraint and the Binding Theory come into play:

(30)

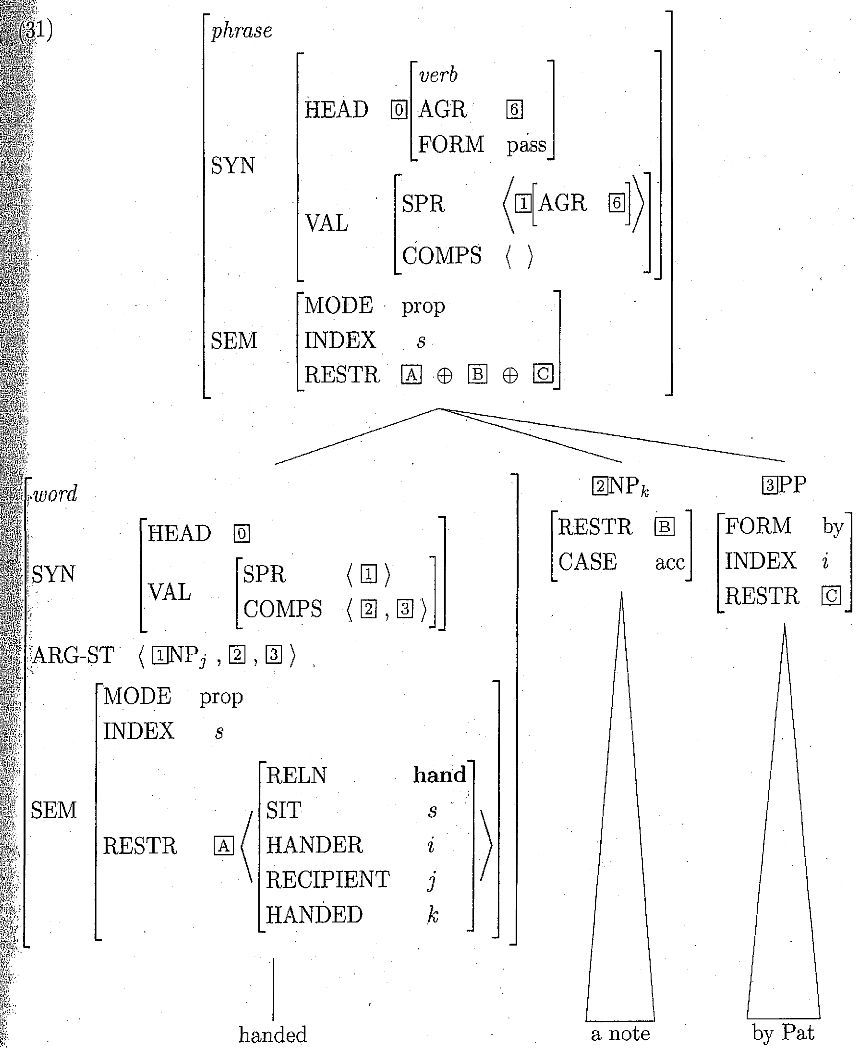
$$\begin{bmatrix} \text{word} & \\ \text{SYN} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \textit{verb} & \\ \text{AGR} & \boxed{6} \\ \text{FORM} & \text{pass}\end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle \boxed{1}[\text{AGR}\ \boxed{6}] \rangle \\ \text{COMPS} & \langle \boxed{2}, \boxed{3} \rangle \end{bmatrix}\end{bmatrix} \\ \text{ARG-ST} & \left\langle \boxed{1}\text{NP}_j\ , \boxed{2}\text{NP}_k[\text{acc}]\ , \overset{\boxed{3}\text{PP}}{\begin{bmatrix} \text{FORM} & \text{by} \\ \text{INDEX} & i\end{bmatrix}}\right\rangle \\ \text{SEM} & \begin{bmatrix} \text{MODE} & \text{prop} \\ \text{INDEX} & \text{s} \\ \text{RESTR} & \left\langle \begin{bmatrix} \text{RELN} & \textbf{hand} \\ \text{SIT} & s \\ \text{HANDER} & i \\ \text{RECIPIENT} & j \\ \text{HANDED} & k \end{bmatrix}\right\rangle \end{bmatrix}\end{bmatrix}$$

$$|$$

handed

This is consistent with the use of *handed* in (24). (30) fits into the larger tree corresponding to the VP[pass] shown in (31):

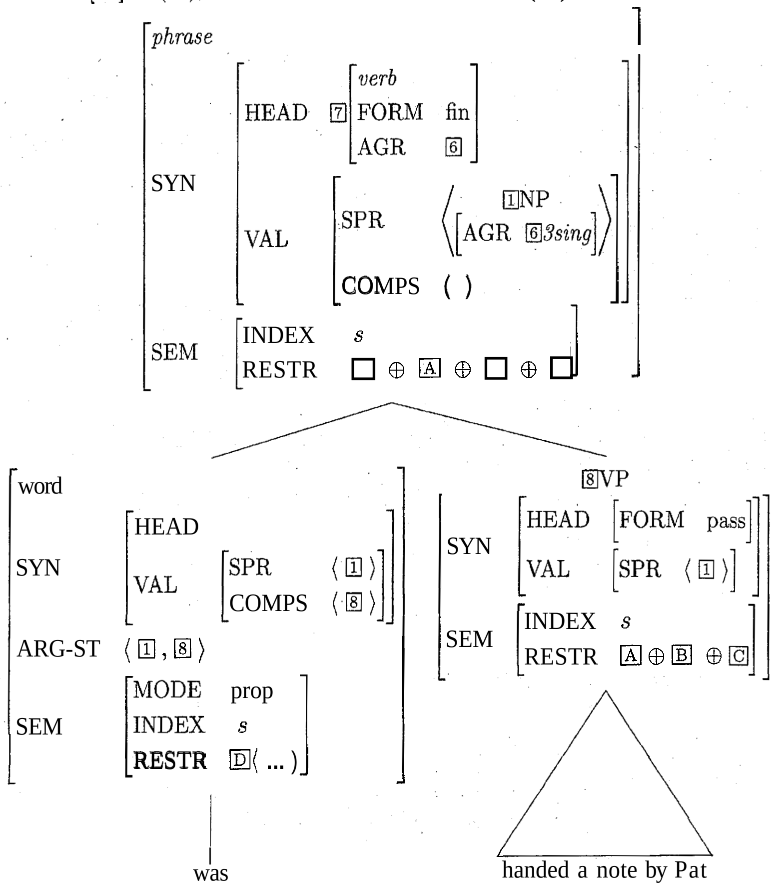
(31)



As usual, the HEAD, SPR, and INDEX values of the mother are the same as those of the head daughter (courtesy of the HFP, the Valence Principle, and the Semantic Inheritance Principle, respectively), and the mother's RESTR value is the sum of the daughters' RESTR values (courtesy of the Semantic Compositionality Principle).

This VP[pass] combines with a word structure licensed by the lexical sequence in (25) to form the VP[fin] in (24), which is shown in more detail in **(32):**

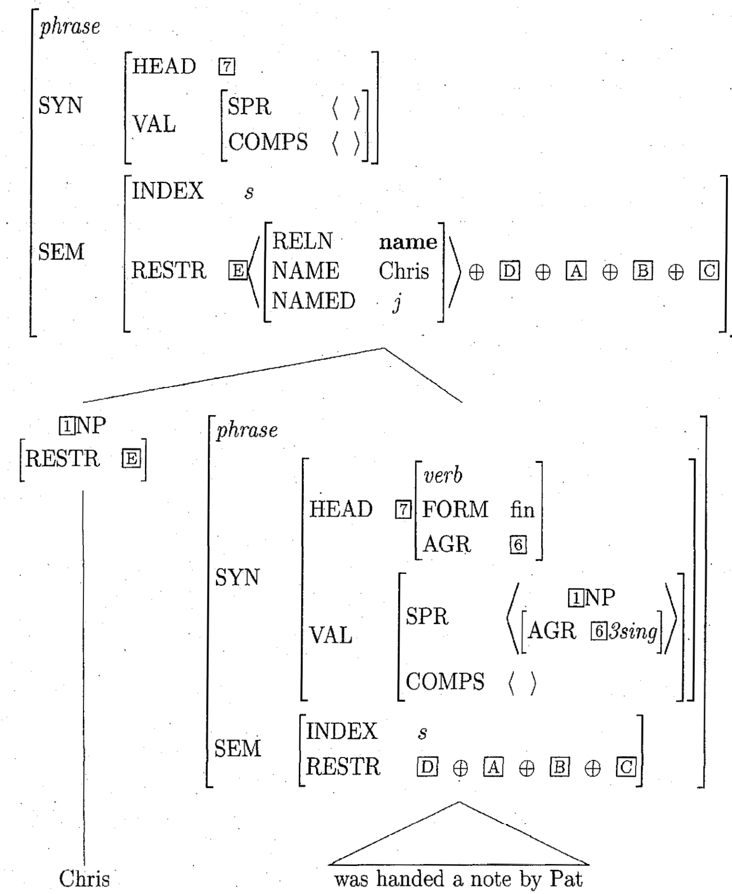
(32)



Again note the effect of the HFP, the Valence Principle, the Semantic Compositionality Principle, and the Semantic Inheritance Principle.

And finally, this VP combines with the subject NP, as shown in **(33):**

(33)



Since the NP dominating Chris is singular, it is consistent with the SPR specification in (33). Because of the identity of subjects established in *be-lxm*, *Chris* (more precisely the NP dominating Chris) is the subject of both was and handed. This assigns the correct semantic interpretation to the sentence: *Chris* plays the recipient role of the handing relation. The other two roles are straightforwardly determined by the indexing shown in (31).

10.6 Summary

Our treatment of the active/passive alternation in English is based on a relationship between verb forms. We formalize this with a derivational lexical rule that modifies the lexeme type, the morphology, the argument structure, and some details of the HEAD values. Passive participles usually follow a form of be; this chapter introduced a lexical entry for this use of be. Passive participles and the form of be that precedes them share the same subject. Our lexical entry for be encodes this fact, anticipating a central topic of Chapter 12.

10.7 Changes to the Grammar

In this chapter, we added the following lexical rule to the grammar:

Passive Lexical Rule

$$\left[\begin{array}{ll} \textit{d-rule} & \\ \text{INPUT} & \left\langle \boxed{1}, \left[\begin{array}{l} \textit{tv-lxm} \\ \text{ARG-ST} \;\; \langle [\text{INDEX } i] \rangle \oplus \boxed{A} \end{array}\right] \right\rangle \\ \text{OUPUT} & \left\langle \text{F}_{PSP}(\boxed{1}), \left[\begin{array}{ll} \textit{part-lxm} & \\ \text{SYN} & [\text{HEAD} \;\; [\text{FORM} \;\; \text{pass}]] \\ \text{ARC-ST} & \square \oplus \left\langle \left(\begin{array}{c} \text{PP} \\ \left[\begin{array}{ll} \text{FORM} & \text{by} \\ \text{INDEX} & i \end{array}\right] \end{array} \right) \right\rangle \end{array}\right] \right\rangle \end{array}\right]$$

We also added a lexeme be, which is distinguished from other verb lexemes we've seen so far in that it identifies the first member of its ARG-ST list with the SPR of the second member:

$$\left\langle \text{be}, \left[\begin{array}{ll} \textit{be-lxm} & \\ \text{ARG-ST} & \left\langle \boxed{1}, \left[\begin{array}{ll} \text{SYN} & \left[\begin{array}{ll} \text{HEAD} & \left[\begin{array}{l} \textit{verb} \\ \text{FORM} \;\; \text{pass} \\ \text{FOF} \end{array}\right] \\ \text{VAL} & \left[\begin{array}{ll} \text{SPR} & \langle \boxed{1} \rangle \\ \text{COMPS} & (\;) \end{array}\right] \end{array}\right] \\ \text{SEM} & [\text{INDEX} \;\; s] \end{array}\right] \right\rangle \\ & \left.\begin{array}{l} s \\ \langle\;\rangle \end{array}\right] \end{array}\right] \right)$$

The constraints in **(33)** will be revised somewhat in Chapters 11 and 13, but this key property will remain constant.

10.8 Further Reading

The English passive has been analyzed and reanalyzed throughout the history of generative grammar. Among the most influential works on the subject are: Chomsky 1957, 1965, and 1970; Perlmutter and Postal 1977; Wasow 1977; Bresnan 1982c; Burzio 1986; and Postal 1986.

10.9 Problems

⚠ Problem 1: Passive and Binding Theory

The analysis of passive just sketched makes some predictions about binding possibilities in passive sentences. Consider the following data:[7]

(i) She_i was introduced to herself_i (by the doctor).
(ii)*She_i was introduced to her_i (by the doctor).
(iii) The barber_i was shaved (only) by himself_i.
(iv)*The barber_i was shaved (only) by him_i.
(v) The students_i were introduced to each other_i (by Leslie).
(vi)*The students_i were introduced to them_i (by Leslie).
(vii) Kim was introduced to Larry, by himself,.
(viii)*Kim was introduced to himself, by Larry,.

Assuming that to and by in these examples are uniformly treated as argument-marking prepositions, does the treatment of passives sketched in the text correctly predict the judgements in (i)–(viii)? If so, explain why; if not, discuss the inadequacy of the analysis in precise terms.

An ideal answer should examine each one of the eight sentences and determine if it follows the binding principles. That is, the analysis of passive presented in this chapter associates a particular ARG-ST list with the passive verb form in each example and these lists interact with the binding principles of Chapter 7 to make predictions. Check to see if the predictions made by our Binding Theory match the grammaticality judgements given.

Problem 2: Pseudopassives

Consider the following passive sentences:

(i) Dominique was laughed at by the younger kids.
(ii) This bed was slept in by the ambassador to Dalmatia.
(iii) This problem is talked about in every home.

A. Explain why our current passive rule does not allow sentences like (i)–(iii) to be generated.

[7]It may require a little imagination to construct contexts where such examples have a plausible meaning, e.g. a doctor dealing with an amnesia victim. Being able to construct such contexts is an essential part of being able to understand what conclusions to draw from the fact that some sentence you are interested in doesn't sound completely acceptable.

We know of cases where grammatical deviance has not been separated with sufficient care from semantic implausibility. For example, examples like ?*I smell funny* to myself have on occasion been cited as ungrammatical. However, a bit of reflection will reveal, we think, that what is strange about such examples is the message they convey, not their grammar. If one needed to convey that one's own olfactory self-impression was strange (in whatever odd context such a need might arise), then *I smell funny to myself* is probably the most straightforward way the grammar of English has for allowing such a meaning to be expressed.

B. Give the ARG-ST and RESTR values for one of the passive participles in (i)–(iii), along with the ARG-ST and RESTR values of the corresponding active form.

C. Propose an additional lexical rule that will produce appropriate lexical sequences for the passive participles in these sentences.

[Hints: Your new rule should be similar to our existing *Passive* Lexical Rule. *Assume* that the prepositions involved in examples of this sort are all argument-marking prepositions – that is, they all share INDEX and MODE values with their object *NPs*. Your rule will need to use these INDEX values (and the FORM values of the prepositions) in producing the passive *lexemes* needed to license examples like *(i)–(iii)*.]

D. Explain how your lexical rule relates the ARG-ST values you gave in (B) to each other.

E. Assuming the lexical entry in (iv), does the rule you formulated in (C) predict that both (iii) and (v) are grammatical?

(iv) $\left\langle \text{talk}\ ,\ \begin{bmatrix} \textit{new-v-lxm} \\ \text{ARG-ST} & \langle\ \text{NP}\ (,\ \text{PP[to]})\ (,\ \text{PP[about]})\ \rangle \end{bmatrix} \right\rangle$

(v) This person was talked to by every teacher.

Explain your answer.

Problem 3: The Dative Alternation

In Chapter 8, we mentioned the possibility of formulating a lexical rule describing the 'dative alternation' – that is, a class of verbs that appear in both of the valence patterns exemplified in (i) and (ii):

(i) $$\text{Dale} \left\{ \begin{array}{l} \text{gave} \\ \text{handed} \\ \text{sold} \\ \text{loaned} \\ \text{mailed} \end{array} \right\} \text{Merle a book.}$$

(ii) $$\text{Dale} \left\{ \begin{array}{l} \text{gave} \\ \text{handed} \\ \text{sold} \\ \text{loaned} \\ \text{mailed} \end{array} \right\} \text{a book to Merle.}$$

A. Is this alternation productive? Justify your answer with at least two examples. [Hint: See the discussion of productive lexical rules at the end of Section 8.1 of Chapter 8.]

B. Formulate a lexical rule for the dative alternation.

[Hint: Consider which kind of *l-rule* (*i-rule* or d-rule) this should be, based on the kind of constraints you need to write. You can choose either of the valences illustrated in *(i)* and (ii) as the input and the other as the output. It should not be easier one way than the other.]

C. Show how your rule interacts with the Passive Lexical Rule to make possible the generation of both (iii) and (iv). Your answer should include ARG-ST values showing the effect of applying the rules. [Hint: First consider which order the rules apply in, based on the types of the *INPUT* and OUTPUT values of each rule.]

(iii) Merle was handed a book by Dale.

(iv) A book was handed to Merle by Dale.

D. Explain why your rule correctly fails to license (v) (or, more precisely, fails to license (v) with the sensible meaning that the book was the thing handed to Merle).

(v) ?*A book was handed Merle by Dale.

14

Long-Distance Dependencies

14.1 Introduction

One of the principal tasks of a theory of grammar is to provide mechanisms that allow economical formulations of the sorts of co-occurrence restrictions that exist in natural languages. In earlier chapters, we developed techniques for analyzing such aspects of syntax as differences in the valence of particular verbs, agreement between subject and verb, agreement between determiner and head noun, and restrictions on the distribution of dummy NPs. All of these co-occurrence restrictions are quite local, in the sense that they involve limitations on what can occur together as elements of a single clause. We extended this locality slightly with our analysis of raising, which in effect permits the co-occurrence restrictions of one verb to be transmitted to a higher verb.

The present chapter introduces a new type of construction in which the locality of co-occurrence restrictions appears to be violated in a more radical way. In these cases, two elements (say, an NP and a verb) appear far from one another in a sentence, despite the existence of a syntactic dependency (such as case marking or agreement) between them. Handling these 'long distance dependencies' (or LDDs, as we will call them) will require several changes to our theory:

- two new features,
- reformulation of the constraints on the types word, *lexeme* and *1-rule,* and on the initial symbol (in reference to the new features),
- a minor reformulation of some of our grammar rules,
- a new principle,
- a new grammar rule, and
- a new lexical rule.

14.2 Some Data

Our current grammar correctly rules out examples like the following:

(1) a.*They handed to the baby.
b.*They handed the toy.
c.*You have talked to.
d.*The children discover

Because the lexical entry for hand specifies that its COMPS list has both an object NP and a PP, (la–b) are ruled out through the interaction of the lexicon, the headed grammar rules, the Argument Realization Principle, and the Valence Principle. Similarly, (1c–d) are ruled out because both the preposition to and the verb discover require an object NP, which is absent from these examples.

So it's interesting to find that there are grammatical sentences that contain exactly the ungrammatical strings of words in (1). For example, there are questions containing wh-words ('wh-questions') such as following:

(2) a. What did they hand to the baby?
b. To whom did they hand the toy?
c. Who(m) should you have talked to?
d. What will the children discover?

There are also NPs modified by RELATIVE CLAUSES which contain the same ungrammatical strings:

(3) a. The toy which they handed to the baby...
b. The baby to whom they handed the toy ...
c. The people who(m) you have talked to...
d. The presents that the children discover...

Another sort of example is a kind of sentence that is used for a certain sort of emphasis that is usually called a 'topicalized' sentence. In such sentences, a topicalized element can be followed by one of those same ungrammatical word sequences in (1):[1]

(4) a. That toy, they handed to the baby.
b. To the baby, they handed a toy.
c. That kind of person, you have talked to (many times).
d. Presents that come from grandma, the children (always) discover.

And finally, there are certain adjectives like easy and hard whose infinitival complements may contain a verb or preposition lacking a normally obligatory object:

(5) a, That toy would be easy to hand to the baby.
b. You are easy to talk to.
c. The presents from grandma were hard for the children to discover.

In each of the examples in (2)–(5), there is a dependency between a phrase or 'filler' at the beginning of a clause and a 'gap' somewhere within the clause. In questions, relative clauses, and topicalized sentences, the filler appears to be an extra phrase in that position; in examples like (5), the subject of the clause also serves as the filler.

[1]When examples like (4) are first presented, some students claim that they find them unacceptable, but examination of actual usage indicates that topicalization is quite common, e.g. in examples like the following:

(i) Me, you bring an empty food dish; him, you bring a leash. (from a cartoon)
(ii) The film clips you're going to see tonight, no one's ever seen before. (Carol Burnett radio ad, November 26, 2001)

The name 'topicalization' is actually rather misleading. To be sure, the fronted element refers to an entity whose role in the discourse is distinguished in some way, but that entity need not correspond to the 'topic of discussion' in any straightforward way, as (i) indicates.

In short, we see that elements whose presence is usually required in a clause are allowed to be absent if there is an appropriate filler in the right place. Likewise, if there is a filler, then there must be a gap somewhere within the sentence that follows the filler:

(6) a.*What did Kim hand the toys to the baby?
b.*The dolls that Kim handed the toys to the baby....
c.*The dolls, Kim handed the toys to the baby.
d.*The dolls are easy to hand the toys to the baby.

In such constructions, the filler can be separated from the gap by extra clauses, as indicated in (7)–(10). To help readers identify the location of the gaps, we have marked them with an underlined space.

(7) a. What did you say they handed __ to the baby?
b. Who(m) did he claim that they handed the toy to __ ?
c. Who(m) do you think you have talked to __ ?
d. What will he predict that the children discover __ ?

(8) a. The toy which we believe they handed __ to the baby...
b. The baby that I think they handed the toy to __ ...
c. The person who(m) everyone thinks you have talked to __ ...
d. The presents that it annoys me that the children discover __ ...

(9) a. That toy, I think they handed __ to the baby.
b. This baby, I know that they handed a toy to __ .
c. That kind of person, you know you have talked to __ .
d. Presents that come from grandma, I know that the children (always) discover __ .

(10) a. This toy isn't easy to try to hand __ to the baby.
b. The baby is easy to ask someone to hand a toy to __ .
c. That kind of person is hard to find anyone to talk to __ .
d. Presents from grandma are easy to help the children to discover __ .

In fact, there can be multiple extra clauses intervening:

(11) What did you think Pat claimed I said they handed __ to the baby?

14.3 Formulating the Problem

We want to be able to build clauses with elements missing within them. But somehow we have to keep track of the fact that something is missing. Furthermore, as the following contrasts show, we need to keep track of just what is missing:

(12) a. This, you can rely on.
b.*This, you can rely.
c.*On this, you can rely on.
d. On this, you can rely.
e.*On this, you can trust.

(13) a. Him, you can rely on.
b.*He, you can rely on.

(14) a. The twins, I can't tell the difference between.
b.*That couple, I can't tell the difference between.

Exercise 1: Long-Distance Selectional Dependencies
What exactly is wrong with the starred examples in (12)–(14)? Which element is selecting for the missing (or 'gapped') element, and which requirement of the selecting head does the filler not fulfill?

We can think of this as an information problem. We have to make sure that the phrases within the sentence keep track of what's missing from them as they are built. This has to be done just right, so that sentences missing a phrase of category X (no matter how deeply embedded that gap may be) combine with a filler of category X, and that fillers are allowed only when there is a gap for them to fill (cf. (6)).

14.4 Formulating a Solution

Our solution to this information problem will involve breaking it down into three parts: the bottom, the middle and the top. The bottom of an LDD is where the gap is 'introduced' – i.e. the smallest subtree where something is missing. Many theories handle the bottom by positing an empty element in the tree. We will avoid using empty elements in this way and instead handle the bottom by means of a feature (GAP) and a revision to the ARP that allows ARG-ST elements to show up on GAP instead of on the COMPS list. This is the topic of Section 14.4.1. The middle of an LDD is the 'transmission' of the information about what is missing from bottom to top (alternatively, the 'transmission' of what is available as a filler from top to bottom). We will handle this by means of a principle that relates the GAP values of phrases to the GAP values of their daughters. This is the topic of Section 14.4.2. The top of an LDD is where the filler is introduced, and the GAP requirement cancelled off. How exactly this happens depends on the particular kind of LDD. In Section 14.4.3, we will consider two kinds: 'topicalized' sentences, which we analyze in terms of a new phrase structure rule, and LDDs with easy-class adjectives, where the lexical entry for the adjective handles the top of the LDD.

14.4.1 The Feature GAP

. We introduce the feature GAP (on syn-cat) to encode the fact that a phrase is missing a certain kind of element. There are examples of clauses where more than one phrase is missing,[2] a phenomenon we will return to in Problem 5 below:

(15) a. Problems this involved, my friends on the East Coast are hard to talk to __ about __ .
b. Violins this well crafted, these sonatas are easy to play __ on __ .

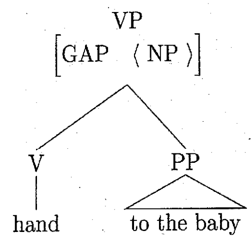
[2] Or, as linguists sometimes say (though it is somewhat of an oxymoron): 'where more than one gap appears)

Note that the two gaps in each of these sentences have distinct fillers. In (15a), for example, the filler for the first gap is my *friends* on *the* East Coast, and the filler for the second one is problems **this** *involved*. Such examples are rare in English and sound a bit awkward, but there are other languages (for example several Slavic and Scandinavian languages) that allow multiple gaps more freely.

Given the existence of sentences with multiple gaps, we need a mechanism that can keep track of multiple missing elements. This suggests that the value of GAP is a list of feature structures, like the values of COMPS, SPR, MOD, and ARG-ST.

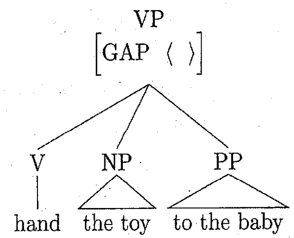
The intuitive significance of a phrase specified as, say, [GAP ⟨ NP ⟩] is that it is missing exactly one NP. The trick will be to make GAP have the right values in the right places. What we want is to allow a transitive verb or preposition to build a VP or PP without ever combining with an object NP. Furthermore, we want to ensure that it is only when an NP is absent that the relevant phrase is specified as [GAP ⟨ NP ⟩], as illustrated in (16):

(16)



When nothing is missing, we want the relevant phrase to be [GAP ⟨ ⟩], as in (17):

(17)



We will deal with this kind of 'missing element' as an instance of something that is present in argument structure but absent from the valence features. We could accomplish this by means of a lexical rule, but a more general solution is to modify the Argument Realization Principle. Our current version of the principle says that a word's SPR and COMPS lists add up to be its argument structure (ARG-ST) list. We now want to allow for the possibility that some element or elements of ARG-ST are on neither the SPR list nor the COMPS list, but on the GAP list instead.

To make this modification precise, we will introduce a kind of subtraction operation on lists, which we will mark with the symbol ⊖. Intuitively, if A and B are lists, then A ⊖ B is a list that results from removing the elements of B from A. A couple of caveats are in order here. First, we want A ⊖ B to be defined only when the elements of B all occur in A, and in the same order. So there are many pairs of lists for which this kind of list subtraction is undefined. This is unlike our form of list addition (⊕), which is defined for any pair of lists. Second, when A $\ominus$ B is defined, it need not be unique. For example, if A = $\langle$NP, PP, NP$\rangle$ and B = $\langle$NP$\rangle$, then there are two possible values for A $\ominus$ B, namely $\langle$NP, PP$\rangle$ and $\langle$PP, NP$\rangle$. We will interpret an equation like A 8 B = C to mean that there is some value for A $\ominus$ B that is identical to C.

With this new tool in hand, we can restate the Argument Realization Principle as follows:

(18) Argument Realization Principle:

$$word: \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{VAL} & \begin{bmatrix} \text{SPR} & \boxed{A} \\ \text{COMPS} & \boxed{B} \ominus \boxed{C} \end{bmatrix} \\ \text{GAP} & \boxed{C} \end{bmatrix} \\ \text{ARG-ST} & \boxed{A} \oplus \boxed{B} \end{bmatrix}$$

The revised ARP guarantees that any argument that could appear on a word's COMPS list can appear on its GAP list instead. (We will deal with gaps that correspond to subjects, rather than complements, in Section 14.5) Further, (18) guarantees that whenever an argument is missing, any co-occurrence restrictions the word imposes on that argument will be registered on the element that appears on the GAP list.

Because the result of list subtraction ($\ominus$), as we have defined it, is not always unique, when we specify the ARG-ST in a verb's lexical entry without also specifying its SPR, COMPS, and GAP values, we are actually providing an underspecified lexical entry that will give rise to a family of words that differ with respect to how the ARP is satisfied. Consider, for example, the lexical entry for the lexeme hand, as specified in (19):

(19)

$$\left\langle \text{hand}, \begin{bmatrix} \textit{ptv-lxm} & \\ \text{ARG-ST} & \left\langle \text{X}_i, \text{Y}_k, \begin{bmatrix} \text{FORM} & \text{to} \\ \text{INDEX} & j \end{bmatrix} \right\rangle \\ \text{SEM} & \begin{bmatrix} \text{INDEX} & \text{s} \\ \text{RESTR} & \left\langle \begin{bmatrix} \text{RELN} & \textbf{hand} \\ \text{SIT} & s \\ \text{HANDER} & i \\ \text{HANDED-TO} & j \\ \text{HANDED-ITEM} & k \end{bmatrix} \right\rangle \end{bmatrix} \end{bmatrix} \right\rangle$$

This can undergo the Non-3rd-Singular Verb Lexical Rule presented in Chapter 8, which gives rise to lexical sequences which satisfy the following description:

(20)

$$\left\langle \text{hand}, \begin{bmatrix} \textit{word} & \\ \text{SYN} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{FORM} & \text{fin} \end{bmatrix} \end{bmatrix} \\ \text{ARC-ST} & \left\langle \begin{bmatrix} \text{CASE} & \text{nom} \\ \text{AGR} & \textit{non-3sing} \end{bmatrix}, \overset{\text{NP}}{\begin{bmatrix} \text{CASE} & \text{acc} \end{bmatrix}}, \begin{bmatrix} \text{FORM} & \text{to} \end{bmatrix} \right\rangle \end{bmatrix} \right\rangle$$

Since the second member of these lexical sequences is of type *word*, it is subject to the ARP. But now there are multiple ways to satisfy the ARP. In particular, the family of lexical sequences described in (20) includes lexical sequences meeting each of the following (more detailed) descriptions:

(21)

$$\left\langle \text{hand}, \begin{bmatrix} \textit{word} & \\ \text{SYN} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{FORM} & \text{fin} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle \boxed{1} \rangle \\ \text{COMPS} & \langle \boxed{2}\text{NP[acc]}, \boxed{3}\text{PP[to]} \rangle \end{bmatrix} \\ \text{GAP} & \langle\ \rangle \end{bmatrix} \\ \text{ARG-ST} & \left\langle \overset{\boxed{1}\text{NP}}{\begin{bmatrix} \text{CASE} & \text{nom} \\ \text{AGR} & \textit{non-3sing} \end{bmatrix}}, \boxed{2}, \boxed{3} \right\rangle \end{bmatrix} \right\rangle$$

(22)

$$\left\langle \text{hand}, \begin{bmatrix} \textit{word} & \\ \text{SYN} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{FORM} & \text{fin} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle \boxed{1} \rangle \\ \text{COMPS} & \langle \boxed{3}\text{PP[to]} \rangle \end{bmatrix} \\ \text{GAP} & \langle \boxed{2}\text{NP[acc]} \rangle \end{bmatrix} \\ \text{ARG-ST} & \left\langle \overset{\boxed{1}\text{NP}}{\begin{bmatrix} \text{CASE} & \text{nom} \\ \text{AGR} & \textit{non-3sing} \end{bmatrix}}, \boxed{2}, \boxed{3} \right\rangle \end{bmatrix} \right\rangle$$

(23)

$$\left\langle \text{hand}, \begin{bmatrix} \textit{word} & \\ \text{SYN} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{FORM} & \text{fin} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle \boxed{1} \rangle \\ \text{COMPS} & \langle \boxed{2}\text{NP[acc]} \rangle \end{bmatrix} \\ \text{GAP} & \langle \boxed{3}\text{PP[to]} \rangle \end{bmatrix} \\ \text{ARG-ST} & \left\langle \overset{\boxed{1}\text{NP}}{\begin{bmatrix} \text{CASE} & \text{nom} \\ \text{AGR} & \textit{non-3sing} \end{bmatrix}}, \boxed{2}, \boxed{3} \right\rangle \end{bmatrix} \right\rangle$$

All of these are legitimate lexical sequences: (21) shows hand's feature structure in sentences like (24a); (22) is the way *hand* appears in the tree our grammar assigns to sentences like (24b); and (23) shows *hand* as it appears in the tree we assign to sentences like (24c):[3]

(24) a. You handed the toy to the baby.
b. What did you hand to the baby?
c. To whom did you hand the toy?

The prepositional lexeme in (25) will now give rise to the word structures sketched in (26) and (27) (omitting what is not directly relevant):

(25)
$$\left\langle \text{to}, \left[\begin{array}{ll} \textit{argmkp-lxm} & \\ \text{SYN} & \left[\begin{array}{ll} \text{HEAD} & \left[\begin{array}{ll}\textit{prep} & \\ \text{FORM} & \text{to}\end{array}\right] \\ \text{VAL} & \left[\begin{array}{ll}\text{SPR} & \langle\,\rangle\end{array}\right]\end{array}\right] \\ \text{ARG-ST} & \langle \boxed{1}\text{NP[acc]}\rangle \end{array}\right] \right\rangle$$

(26)
$$\left[\begin{array}{ll} \textit{word} & \\ \text{SYN} & \left[\begin{array}{ll} \text{HEAD} & \left[\begin{array}{ll}\textit{prep} & \\ \text{FORM} & \text{to}\end{array}\right] \\ \text{VAL} & \left[\begin{array}{ll}\text{SPR} & \langle\,\rangle \\ \text{COMPS} & \langle \boxed{2}\text{NP[acc]}\rangle\end{array}\right] \\ \text{GAP} & \langle\,\rangle \end{array}\right] \\ \text{ARG-ST} & \langle \boxed{2}\rangle \end{array}\right]$$

|
to

(27)
$$\left[\begin{array}{ll} \textit{word} & \\ \text{SYN} & \left[\begin{array}{ll} \text{HEAD} & \left[\begin{array}{ll}\textit{prep} & \\ \text{FORM} & \text{to}\end{array}\right] \\ \text{VAL} & \left[\begin{array}{ll}\text{SPR} & \langle\,\rangle \\ \text{COMPS} & \langle\,\rangle\end{array}\right] \\ \text{GAP} & \langle \boxed{2}\text{NP[acc]}\rangle \end{array}\right] \\ \text{ARG-ST} & \langle \boxed{2}\rangle \end{array}\right]$$

|
to

This last lexical tree is the one that allows for sentences like (28):

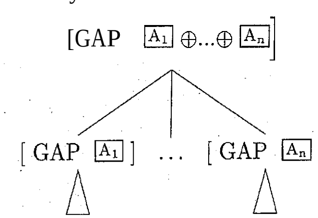
(28) Which baby did you hand the toy to?

14.4.2 The GAP Principle

The GAP feature tells us which of a word's arguments is missing. The Argument Realization Principle, as we have reformulated it, permits us to instantiate gaps freely (other than elements that must be on the SPR list). Now we need some way of passing the information in the GAP value up[4] from words like those just illustrated so that the phrases that they head will register the fact that something is missing, and from those phrases to larger phrases. To do so, we adopt the principle shown in (29):

(29) The GAP Principle (Preliminary Version)

A local subtree Φ satisfies the GAP Principle with respect to a headed rule ρ if and only if Φ satisfies:



In other words, in a headed structure, the GAP values of all the daughters must add up to be the GAP value of the mother. That is, a phrase whose daughter is missing something is missing the exact same thing. There is one exception to this generalization, and that is the case where the larger phrase also contains the filler. We'll return to these cases directly.

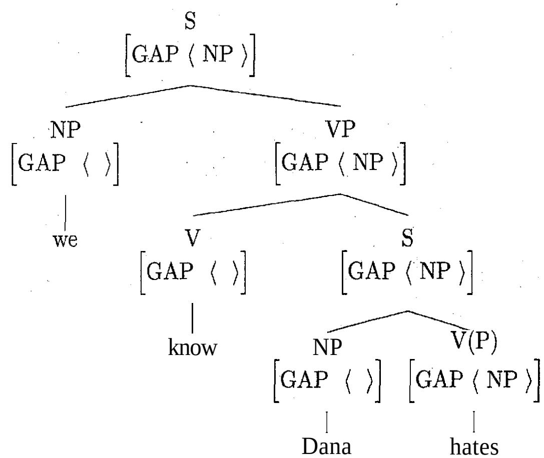
The notion of lists 'adding up to' something is the same one we have employed before, namely the operation that we denote with the symbol '$\oplus$'. In most cases, most of the

[3]The ARP also allows for a family of lexical sequences in which both the NP and PP complements are in the GAP list, rather than the COMPS list. We will return to multiple-gap sentences in Problem 5 below.

[4]The metaphor of passing information between nodes should again not be taken literally. What the principle in (29) does is similar to what the Head Feature Principle and Valence Principle do, namely, enforce a particular relationship between certain feature values in mothers and daughters in phrase structure trees. That is, it is simply part of our definition of phrase-structure well-formedness.

GAP lists that are added up in this way are in fact empty, because most constituents don't contain gaps, so the addition is quite trivial. The effect of (29), then, given our lexical entries (and the word structures they sanction in virtue of our revision of the ARP), is illustrated in (30):

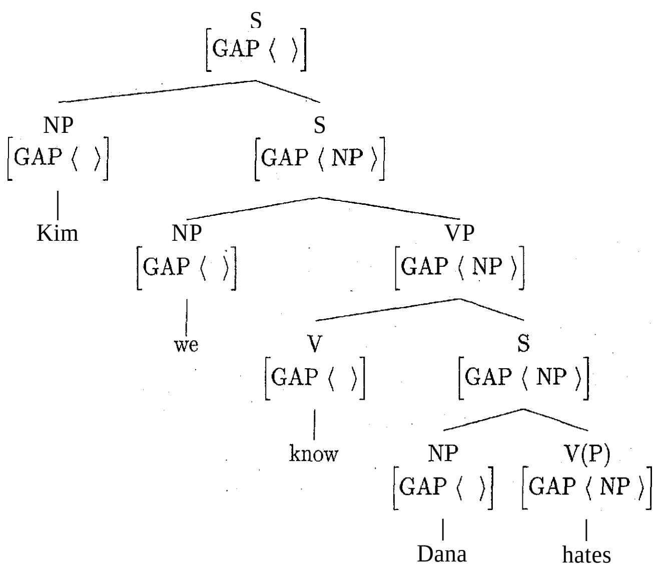
(30)



Note that each local tree in (30) satisfies the GAP Principle. That is, in each tree, the GAP values of the daughters add up to the mother's GAP value: (⟨ ⟩ $\oplus$ ⟨NP⟩) = ⟨NP⟩

We now return to the exception (mentioned above) to the GAP Principle, as stated in the preliminary version: At the top of the LDD, where the gap is filled, we want the mother node to be [GAP ⟨ ⟩]. This is illustrated in (31):

(31)



We have not yet seen the phrase structure rule which licenses the topmost subtree of (31). It will be introduced in the next subsection. Here, we are concerned with the GAP values in that subtree. We want the mother to be [GAP ⟨ ⟩] as shown, because, intuitively, the NP *Kim* is 'filling' the gap. That is, the tree structure shown in (31) is no longer 'missing something', and this should be reflected in the GAP value of the root node in (31).

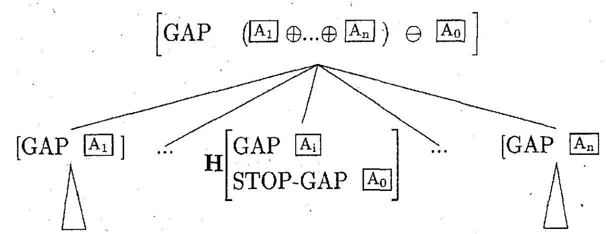
Adjectives like *hard* and *easy*, which we discussed earlier, also perform a gap-filling function, even though they also serve as the head daughter of a head-complement phrase. The VP in (32a) is 'gappy' – it is missing an NP and hence should be specified as [GAP ⟨ NP ⟩], while the AP in (32b) is not gappy and should be specified as [GAP ⟨ ⟩], like all other APs that we have encountered.

(32) a. [to talk to ___]
b. [easy to talk to ___]

We will provide a unified account of gap filling by introducing a new list-valued feature called STOP-GAP. Like GAP, STOP-GAP is a feature of *syn-cats*. This feature signals what gap is to be filled in the local subtree where it appears. Most nodes will be [STOP-GAP ⟨ ⟩], but where a gap is associated with its filler, the feature has a non-empty list as its value. In particular, the lexical entries for gap stoppers like *easy* and *hard* will specify a non-empty value for this feature, as will the grammar rule we introduce for the topicalization construction. Making use of this new feature, we can reformulate the GAP Principle so that it passes up GAP values only if they are not filled. This is shown in (33):

(33) The GAP Principle (Final Version)

A local subtree Φ satisfies the GAP Principle with respect to a headed rule ρ if and only if Φ satisfies:



What this revision says is that the GAP value of the mother node in a headed structure is determined by adding up the GAP values of all the daughters and then subtracting any gaps that are being filled, as indicated by the head daughter's STOP-GAP value.

14.4.3 The Head-Filler Rule and *Easy*-Adjectives

We have dealt with the bottom of LDDs, where non-empty values for GAP are introduced, and the middle of LDDs where those GAP values are propagated through the tree (until they meet their fillers). Now we turn to the top of LDDs: the filling of the gap. As noted above, we will consider two types of gap-filling here: topicalized sentences and *easy*-adjectives.

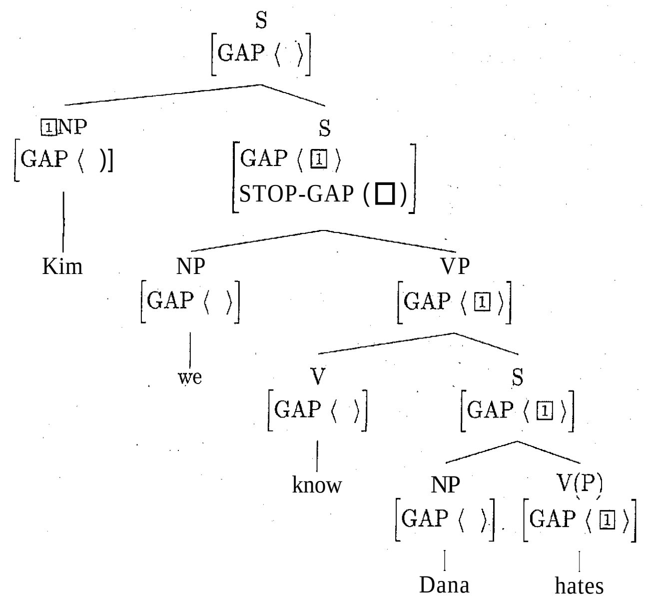
To deal with topicalized sentences, we now introduce a new grammar rule, formulated as follows:

(34) Head-Filler Rule

$$[phrase] \rightarrow \boxed{1}\big[\text{GAP}\ \langle\ \rangle\big] \quad \mathbf{H}\begin{bmatrix} \text{HEAD} & \begin{bmatrix} \textit{verb} & \\ \text{FORM} & \text{fin} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle\ \rangle \\ \text{COMPS} & \langle\ \rangle \end{bmatrix} \\ \text{STOP-GAP} & \langle \boxed{1} \rangle \\ \text{GAP} & \langle \boxed{1} \rangle \end{bmatrix}$$

This rule says that a phrase can consist of a head with a gap preceded by an expression that meets whatever requirements the head places on that gap.[5] The Head-Filler Rule licenses the topmost subtree in (35), and it enforces the identity between the NP *Kim* and the element on the GAP list of the gappy S *we know Dana hates* ($\boxed{1}$). Because that GAP element is identified with the GAP element of the V *hates* (and therefore also with an element of its ARG-ST list), any requirements that *hates* places on its complement (that it be a [CASE acc] NP, that its INDEX be identified with the HATED value in the **hate** predication) must be satisfied by the filler *Kim*.

(35)



The topmost node of (35) is [GAP ⟨ ⟩], indicating that the gap has been filled, thanks to the GAP Principle: The Head-Filler Rule in (35) specifies that the head daughter's GAP list and STOP-GAP list both contain the filler daughter, so that element is subtracted from the head daughter's GAP value in determining the GAP value of the

[5]And further that the filler must not be gappy.

mother: $(\langle\ \rangle \oplus \langle \boxed{1} \rangle) \ominus \langle \boxed{1} \rangle = \langle\ \rangle$.

It is important to see that our analysis entails that a filler NP can appear before a clause only when that clause is gappy, i.e. when that clause is missing an NP that would normally appear there. Moreover, the Head-Filler Rule does not require the filler to be an NP, but it does require that the filler's *synsem* be identified with the unique member of the head daughter's GAP list. From this it follows that topicalized sentences may contain PP fillers (and perhaps fillers of other categories) just as long as the gap within the clause matches the synsem of the filler. That is, if the filler is a PP, then the missing element must be a PP, not an NP. This is a consequence of the many identities triggered by the Head-Filler Rule and the GAP Principle, interacting with the Argument Realization Principle and particular lexically specified ARG-ST values.

We now turn to our other example of gap filling, adjectives like *easy* or hard. Most words don't fill gaps, so we will posit the following defeasible constraint on the type *lexeme*:

(36) $\quad \textit{lexeme} : \big[\text{STOP-GAP}\ /\ \langle\ \rangle\big]$

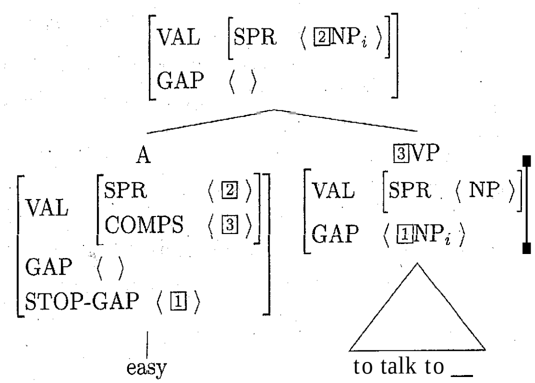
Adjectives like *easy* or *hard* are the exceptions. We give them lexical entries which override this constraint, as shown for *easy* in (37):

(37)

$$\left\langle \text{easy}\ ,\ \begin{bmatrix} \textit{adj-lxm} & \\ \text{SYN} & \big[\text{STOP-GAP}\ \langle \boxed{1} \rangle\big] \\ \text{ARG-ST} & \left\langle \text{NP}_i\ ,\ \begin{bmatrix} \text{VP} & \\ \text{INF} & + \\ \text{GAP} & \langle \boxed{1}\text{NP}_i\ , \ldots \rangle \end{bmatrix} \right\rangle \end{bmatrix} \right\rangle$$

Because the member of the STOP-GAP list in (37) is identified with the first member of the VP argument's GAP list, adjectives of this type must perform gap stopping of the sort shown in (38):

(38)



Notice that the GAP list is empty at the top node of this subtree. That is, the AP *easy to talk to* is treated as having no gap, even though the infinitival VP *to* talk *to* inside it has an NP gap. This may seem puzzling, since easy to talk to seems to be missing the same NP as to talk to. But at the level of the AP, the referent of the missing NP is fully determined: it is the same as the subject of the AP. Hence, the GAP list at the AP level no longer needs to register the missing NP. Instead, the first argument (that is, the subject) of the AP is coindexed with the NP in the GAP list.[6] This guarantees that, in a sentence like (39), the Pat is understood as the person who is followed:

(39) Pat is easy to continue to follow __ .

14.4.4 GAP and STOP-GAP in the Rest of the Grammar

We have added two features to our grammar (GAP and STOP-GAP) which are involved in passing information around the tree. As such, we must pause and ask whether the rest of our grammar (in particular, lexical rules, the rest of our grammar rules and the initial symbol) are currently doing the right thing with respect to these new features. The answer is (unsurprisingly) that we will need to make a few modifications.

First, with respect to the feature GAP: Nothing we have said so far ensures that all gaps ultimately get filled. We make sure that SPR and COMPS requirements are ultimately fulfilled by requiring that both be empty on the initial symbol. We can do the same for GAP. That is, our initial symbol is now the following:

(40)
$$\begin{bmatrix} \text{phrase} & \\ \text{SYN} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \textit{verb} & \\ \text{FORM} & \text{fin} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \\ \text{COMPS} & \langle\ \rangle \end{bmatrix} \\ \text{GAP} & \langle\ \rangle \end{bmatrix} \end{bmatrix}$$

Without this specification, we would license examples like (1), repeated here for convenience, as stand-alone utterances:

(41) a.*They handed to the baby.
 b.*They handed the toy.
 c.*You have talked to.
 d.*The children discover.

The other consideration with respect to the feature GAP is whether its value is sufficiently constrained. The GAP Principle applies to headed phrases, but not non-headed phrases. Thus, in our discussion so far, we have not constrained the GAP value of coordinate phrases or imperatives. We will return to coordination in Section 14.6 below. As for imperatives, in order to ensure that we don't allow gappy VPs as the daughter (as in (42)), we can identify the mother's and daughter's GAP values, as shown in (43). Since imperative phrases must also satisfy the initial symbol, they must be [GAP ⟨ ⟩] on the mother.

(42)*Hand the toy!

[6] More precisely, with the NP in initial position in the GAP list.

(43) Imperative Rule (Revised Version)

$$\begin{bmatrix} \textit{phrase} & \\ \text{SYN} & \begin{bmatrix} \text{HEAD} & \textit{verb} \\ \text{VAL} & [\text{SPR}\ \langle\ \rangle] \\ \text{GAP} & \boxed{A} \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \text{MODE} & \text{dir} \\ \text{INDEX} & s \end{bmatrix} \end{bmatrix} \rightarrow \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \textit{verb} & \\ \text{INF} & - \\ \text{FORM} & \text{base} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle \text{NP[PER 2nd]} \rangle \\ \text{COMPS} & \langle\ \rangle \end{bmatrix} \\ \text{GAP} & \boxed{A} \end{bmatrix} \\ \text{SEM} & [\text{INDEX}\ s] \end{bmatrix}$$

Thanks to the GAP Principle and the two modifications given above, GAP values are now sufficiently constrained throughout our grammar. We haven't said much about STOP-GAP values, however, except to say that they are non-empty in two places: on the head daughter of a head-filler phrase, and in the lexical entries for adjectives like (easy). In addition, the defeasible constraint given in (36) above and repeated here ensures that all other lexical entries are [STOP-GAP ⟨ ⟩]:

(44) $\textit{lexeme} : [\text{STOP-GAP}\ /\ \langle\ \rangle]$

Since we want the STOP-GAP values given on lexemes to be reflected in the word-structures they license, we need to make sure that all lexical rules preserve that information. We do that by adding the following non-defeasible constraint to the type *l-rule*:

$$\textit{l-rule} : \begin{bmatrix} \text{INPUT} & \langle \text{X}\ ,\ [\text{STOP-GAP}\ \boxed{A}] \rangle \\ \text{OUTPUT} & \langle \text{Y}\ ,\ [\text{STOP-GAP}\ \boxed{A}] \rangle \end{bmatrix}$$

When STOP-GAP is non-empty, the GAP Principle subtracts the relevant element from the GAP list being passed 'up' the tree. It follows that we want to ensure that STOP-GAP is empty when there is no gap-filling going on. Gaps are never filled in head-specifier or head-modifier phrases, so we constrain the head daughters of the Head-Specifier and Head-Modifier Rules to be [STOP-GAP ⟨ ⟩]:

(45) Head-Specifier Rule (Revised Version)

$$\begin{bmatrix} \textit{phrase} & \\ \text{SPR} & \langle\ \rangle \end{bmatrix} \rightarrow \boxed{1}\quad \mathbf{H}\begin{bmatrix} \text{SPR} & \langle \boxed{1} \rangle \\ \text{COMPS} & \langle\ \rangle \\ \text{STOP-GAP} & \langle\ \rangle \end{bmatrix}$$

(46) Head-Modifier Rule (Revised Version)

$$[\textit{phrase}] \rightarrow \mathbf{H}\boxed{1}\begin{bmatrix} \text{COMPS} & \langle\ \rangle \\ \text{STOP-GAP} & \langle\ \rangle \end{bmatrix} \begin{bmatrix} \text{COMPS} & \langle\ \rangle \\ \text{MOD} & \langle \boxed{1} \rangle \end{bmatrix}$$

Gap-filling sometimes occurs in head-complement phrases (in particular, when the head is an adjective like easy), so we do not want to constrain the head daughter of the Head-Complement Rule to be [STOP-GAP ⟨ ⟩]. However, since the head daughter of this rule is always a word, the STOP-GAP value will be appropriately constrained by the lexical entries.

This completes our discussion of complement gaps.[7]

14.5 Subject Gaps

We have covered only the basic cases of long-distance dependencies. There are many additional complexities. For example, we have not discussed cases in which the gaps are not complements, but rather subjects or modifiers. In addition, we have not discussed the distribution of wh-words (such as who, *what*, which, etc.) in questions and relative clauses, nor the obligatory inverted order of subject and auxiliary verb in many wh-questions. There is a rich literature investigating these and many other questions associated with LDDs, but such matters are beyond the scope of this text. In this section we sketch the basics of an account of what is subject extraction – that is LDDs in which the gaps are in subject position.

Our present account does not yet deal with examples like (47):

(47) a. Which candidates do you think like oysters on the half-shell?
b. That candidate, I think likes oysters on the half-shell.

This is because of an interaction between the ARP and the constraints (including the SHAC, inherited from *infl-lxm*) that all verb lexemes have SPR lists of length one. Together, these constraints require that the first member of a verb's ARG-ST list must appear on its SPR list. It may not belong to the rest of the list – i.e. to the list of elements that can appear on either COMPS or GAP, according to the ARP.

Rather than attempt to revise the ARP to handle these cases, we will treat them in terms of a post-inflectional lexical rule which provides [SPR ⟨ ⟩] lexical sequences for verbs, and puts the right information into the GAP list:

(48) Subject Extraction Lexical Rule

$$\left[\begin{array}{ll} \textit{pi-rule} & \\ \text{INPUT} & \left\langle \text{X}, \left[\begin{array}{ll} \text{SYN} & \left[\begin{array}{ll} \text{HEAD} & \left[\begin{array}{ll} \textit{verb} & \\ \text{FORM} & \text{fin} \end{array}\right] \\ \text{VAL} & [\text{SPR} \ \langle \text{Z} \rangle] \end{array}\right] \\ \text{ARG-ST} & \boxed{A} \end{array}\right] \right\rangle \\ \text{OUTPUT} & \left\langle \text{Y}, \left[\begin{array}{ll} \text{SYN} & \left[\begin{array}{ll} \text{VAL} & [\text{SPR} \ \langle\ \rangle] \\ \text{GAP} & \langle \boxed{1} \rangle \end{array}\right] \\ \text{ARG-ST} & \boxed{A} \langle \square, \ldots \rangle \end{array}\right] \right\rangle \end{array}\right]$$

This rule maps any finite verb form into a word with an empty SPR list and a GAP list containing an element identified with the first argument – the subject of the verb. The lexical sequences that are the outputs of this rule are illustrated by the description in (49):

(49)

$$\left\langle \text{likes}, \left[\begin{array}{ll} \textit{word} & \\ \text{SYN} & \left[\begin{array}{ll} \text{HEAD} & \left[\begin{array}{ll} \textit{verb} & \\ \text{FORM} & \text{fin} \end{array}\right] \\ \text{VAL} & \left[\begin{array}{ll} \text{SPR} & \langle\ \rangle \\ \text{COMPS} & \langle \boxed{2} \rangle \end{array}\right] \\ \text{GAP} & \left\langle \boxed{1} \left[\begin{array}{ll} \text{CASE} & \text{nom} \\ \text{AGR} & \textit{3sing} \end{array}\right] \right\rangle \\ \text{STOP-GAP} & \langle\ \rangle \end{array}\right] \\ \text{ARG-ST} & \langle \boxed{1}, \boxed{2}\text{NP[acc]} \rangle \end{array}\right] \right\rangle$$

Note that the ARP (inherited from the type word) is satisfied in (49): the SPR list is empty, and the rest of the ARG-ST list (i.e. the whole ARC-ST list) is appropriately related to the list values of COMPS and GAP. That is, the COMPS value (⟨NP[acc]⟩) is just the ARG-ST value (50a) minus the GAP value (50b):

(50) a. $\left\langle \left[\begin{array}{ll} \text{CASE} & \text{nom} \\ \text{AGR} & \textit{3sing} \end{array}\right], \begin{array}{c} \text{NP} \\ [\text{CASE acc}] \end{array} \right\rangle$

b. $\left\langle \left[\begin{array}{ll} \text{CASE} & \text{nom} \\ \text{AGR} & \textit{3sing} \end{array}\right] \right\rangle$

14.6 The Coordinate Structure Constraint

One of the most discussed topics related to LDDs concerns restrictions on possible filler/gap associations. Although the position of filler and gap may be arbitrarily far apart, there are certain configurations that do not permit LDDs. Such configurations are known as 'islands' (a term due to Ross (1967)), and a major goal of syntactic research over the past three decades has been to understand where and why islands occur. In this section, we will look at one type of island and show how our grammar correctly predicts its existence and its properties.

The following examples illustrate what Ross called the 'Coordinate Structure Constraint':

(51) a.*Here is the student that [the principal suspended [__ and Sandy]].
b.*Here is the student that [the principal suspended [Sandy and __]].

(52) a.*Here is the student that [[the principal suspended __] and [the student council passed new rules]].
b.*Here is the student that [[the student council passed new rules] and [the principal suspended __]].

(53) a.*Apple bagels, I can assure you that [[Leslie likes __] and [Sandy hates lox]].
b.*Apple bagels, I can assure you that [[Leslie likes lox] and [Sandy hates __]].

[7] There are further constraints governing complement gaps that we will not treat here. For example, an ADV$_{pol}$ like not or accented SO, which were analyzed as complements in Chapter 13, cannot serve as a topicalization filler:

(i)*Not, Kim will go to the store.
(ii)*So, Kim will go to the store.

This contrasts with the behavior of adverbial modifiers (left untreated in this text), which may be topicalized:

(iii) Tomorrow, (I think) Kim will go to the store __ .

Translating Ross's transformation-based formulation of the constraint into the language of fillers and gaps that we have been using, it can be stated as follows:

(54) Coordinate Structure Constraint (CSC)
In a coordinate structure,
(a) no conjunct can be a gap,
(b) nor can a gap be contained in a conjunct if its filler is outside of that conjunct.

(54a) is often referred to as the CONJUNCT CONSTRAINT, while (54b) is sometimes called the ELEMENT CONSTRAINT.

Ross also noticed a systematic class of exceptions to the Element Constraint, illustrated by (55):

(55) a. This is the dancer that we bought [[a portrait of __] and [two photos of __]].
b. Here is the student that [[the school suspended __] and [we defended __]].
c. Apple bagels, I can assure you that [[Leslie likes __] and [Sandy hates __]].

To handle examples like these, he appended an additional clause to the constraint, which we can formulate as follows:

(56) 'Across-the-Board' Exception (addendum to CSC):
...unless each conjunct properly contains a gap paired with the same filler.

As presented, the Coordinate Structure Constraint seems quite arbitrary, and the Across-the-Board Exception is just an added complication. And most analyses of these phenomena – specifically those that handle LDDs transformationally – have never come to grips with the full range of facts, let alone derived them from general principles.

Note first of all that the Conjunct Constraint is already explained by our grammar. Examples like (51) are ungrammatical for the simple reason that the elements on GAP lists must also be on ARG-ST lists, and coordinate conjunctions like and have empty ARG-ST lists. Unlike many other analyses (in particular, transformational approaches) our grammar does not employ empty elements (usually referred to as 'traces') to occupy the position of the gap in the syntactic structure. Since there are no empty NPs in our analysis, there is no empty element that could serve as a conjunct in a coordinate structure. That is, the Conjunct Constraint follows directly from the decision to treat the bottoms of LDDs in terms of an unrealized argument, rather than the presence of an empty element.

Now reconsider the grammar rule for coordination last updated in Chapter 8:

(57) Coordination Rule (Chapter 8 Version)

$$\begin{bmatrix}\text{FORM} & \boxed{1}\\ \text{VAL} & \boxed{0}\\ \text{IND} & s_0\end{bmatrix} \rightarrow$$

$$\begin{bmatrix}\text{FORM} & \boxed{1}\\ \text{VAL} & \boxed{0}\\ \text{IND} & s_1\end{bmatrix} \ldots \begin{bmatrix}\text{FORM} & \boxed{1}\\ \text{VAL} & \boxed{0}\\ \text{IND} & s_{n-1}\end{bmatrix} \begin{bmatrix}\text{HEAD} & conj\\ \text{IND} & s_0\\ \text{RESTR} & \langle[\text{ARGS}\ \langle s_1 \ldots s_n\rangle]\rangle\end{bmatrix} \begin{bmatrix}\text{FORM} & \boxed{1}\\ \text{VAL} & \boxed{0}\\ \text{IND} & s_n\end{bmatrix}$$

As stated, this rule doesn't say anything about the GAP values of the conjuncts or of the mother. (Note that the GAP Principle doesn't apply to subtrees licensed by this rule, as it is not a headed rule.) In our discussions of coordination so far, we have seen that some features must be identified across conjuncts (and with the mother) in coordination and that others should not. The Element Constraint examples cited above in (52) and (53) show that GAP is one of the features that must be identified. We thus modify our Coordination Rule slightly to add this constraint:

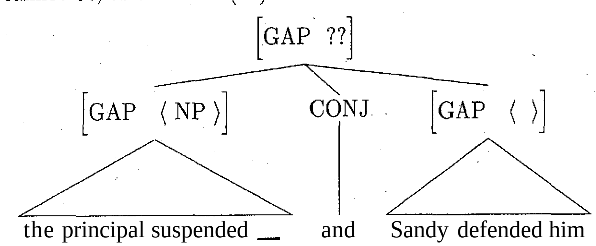
(58) Coordination Rule (Final Version)

$$\begin{bmatrix}\text{FORM} & \boxed{1}\\ \text{VAL} & \boxed{0}\\ \text{GAP} & \boxed{A}\\ \text{IND} & s_0\end{bmatrix} \rightarrow$$

$$\begin{bmatrix}\text{FORM} & \boxed{1}\\ \text{VAL} & \square\ \boxed{A}\\ \text{GAP} & \\ \text{IND} & s_1\end{bmatrix} \ldots \begin{bmatrix}\text{FORM} & \boxed{1}\\ \text{VAL} & \boxed{0}\\ \text{GAP} & \square\\ \text{IND} & s_{n-1}\end{bmatrix} \begin{bmatrix}\text{HEAD} & conj\\ \text{IND} & s_0\\ \text{RESTR} & \langle[\text{ARGS}\ \langle s_1 \ldots s_n\rangle]\rangle\end{bmatrix} \begin{bmatrix}\text{FORM} & \boxed{1}\\ \text{VAL} & \boxed{0}\\ \text{GAP} & \boxed{A}\\ \text{IND} & s_n\end{bmatrix}$$

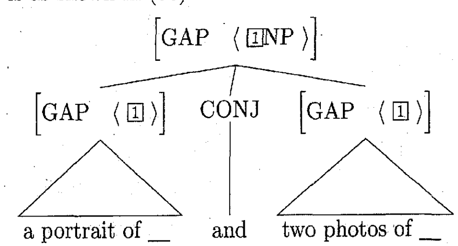
This revision guarantees that two conjuncts in a coordinate structure cannot differ in their GAP value. If one has an empty GAP list and the other has a nonempty GAP list (as in (51)–(53)), then the structure is not licensed. The GAP values that must be identical cannot be, as shown in (59):

(59)



On the other hand, it is possible for conjuncts to have nonempty GAP lists if they are all nonempty and all share the same value. This is what is illustrated in (55), whose structure is as shown in (60):

(60)



In short, both the Element Constraint and the Across-the-Board exceptions to it are treated properly in this revision of our analysis of coordination.

We close this discussion with one final observation about LDDs and coordinate structures. There is an exception to (56), illustrated by (61):

(61)*Which rock legend would it be ridiculous to compare [[__] and [__]]?

Our statements of the generalizations in (54) and (56), like Ross's original formulations of them, would in fact permit (61), whose deviance should have a syntactic (rather than a semantic) explanation, it would appear, because the meaning of this putative sentence could certainly be expressed as (62):

(62) Which rock legend would it be ridiculous to compare __ with himself?

But our analysis correctly rules out any sentences in which a gap constitutes a full conjunct. As noted above, this is because nonempty GAP values in the lexicon are licensed by the Argument Realization Principle, which allows ARG-ST elements not to be expressed as complements, rather than allowing them to appear as a phonetically empty element, or 'trace'. The difference is subtle, but the predictions are quite striking: our traceless analysis of gaps provides an immediate account of the deviance of (61) as well as an explanation of the examples in (51)–(53), which motivated Ross's Conjunct Constraint. The Coordinate Structure Constraint and its exceptions are thus properly accounted for in the analysis of coordination we have developed. Many alternative approaches – particularly those involving movement transformations to account for LDDs – have been unable to account for them at all.

14.7' Summary

Deducing the Conjunct Constraint from the interaction of our analyses of coordination and LDDs is an elegant result, providing significant support for our general approach to syntax. We also showed that we could extend our account of coordination in order to account for the Element Constraint as well.[8]

We will not examine other island constraints in this text. As with the Coordinate Structure Constraint, linguists have not been content to catalog the environments in which filler-gap pairings are impossible. Rather, a great deal of effort has gone into the search for explanations of syntactic islands, either in terms of the interaction of independently motivated elements of the theory (as in the example given above), or in terms of such factors as the architecture of the human language-processing mechanisms. This is a fertile area of research, in which definitive answers have not yet been found.

14.8 Changes to the Grammar

In this chapter, we developed an analysis of long-distance dependencies involving 'fillers' and unrealized elements, or 'gaps'. Our analysis involved two new features, GAP and STOP-GAP, both appropriate for feature structures of type syn-cat:

$$\text{syn-cat}: \begin{bmatrix} \text{HEAD} & \textit{pos} \\ \text{VAL} & \text{val-cat} \\ \text{GAP} & \textit{list(expression)} \\ \text{STOP-GAP} & \textit{list(expression)} \end{bmatrix}$$

[8]Essentially this account was first developed by Gazdar (1981), within the framework of Generalized Phrase Structure Grammar.

We treated the introduction of gaps at the bottom of LDDs in terms of the following modification of the Argument Realization Principle:

Argument Realization Principle:

$$\text{word}: \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{VAL} & \begin{bmatrix} \text{SPR} & \boxed{A} \\ \text{COMPS} & \boxed{B} \ominus \boxed{C} \end{bmatrix} \\ \text{GAP} & \boxed{C} \end{bmatrix} \\ \text{ARG-ST} & \square \oplus \square \end{bmatrix}$$

To introduce subject gaps, we created the following lexical rule:

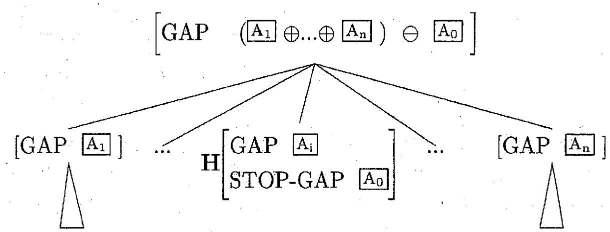
Subject Extraction Lexical Rule

$$\begin{bmatrix} \textit{pi-rule} & \\ \text{INPUT} & \left\langle \text{X}, \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \textit{verb} \\ \text{FORM} & \text{fin} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle \text{Z} \rangle \end{bmatrix} \end{bmatrix} \\ \text{ARG-ST} & \end{bmatrix} \right\rangle \\ \text{OUTPUT} & \left\langle \text{Y}, \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{VAL} & \begin{bmatrix} \text{SPR} & \langle\ \rangle \end{bmatrix} \\ \text{GAP} & \langle \boxed{1} \rangle \end{bmatrix} \\ \text{ARG-ST} & \boxed{A}\langle \boxed{1}, \ldots \rangle \end{bmatrix} \right\rangle \end{bmatrix}$$

We also introduced a new principle, which has the effect of passing GAP specifications from daughter to mother within headed phrase structures, while subtracting out any GAP elements that are bound within the phrase:

The GAP Principle

A local subtree Φ satisfies the GAP Principle with respect to a headed rule ρ if and only if Φ satisfies:



The value of STOP-GAP is assigned in the lexicon by the following defeasible constraint that is overridden by the lexical entries for adjectives like easy and hard:

$$\textit{lexeme}: [\text{STOP-GAP} \ / \ \langle\ \rangle]$$

$$\left\langle \text{easy}, \begin{bmatrix} \textit{adj-lxm} & \\ \text{SYN} & [\text{STOP-GAP} \ \langle \boxed{1} \rangle] \\ \text{ARG-ST} & \left\langle \text{NP}_i, \begin{bmatrix} \text{VP} \\ + \\ \text{GAP} \ \langle \boxed{1}\text{NP}_i, \ldots \rangle \end{bmatrix} \right\rangle \end{bmatrix} \right\rangle$$

And we added the following nondefeasible constraint on the type l-rule:

$$\textit{l-rule}: \begin{bmatrix} \text{INPUT} & \left\langle \text{X}, [\text{STOP-GAP} \ \boxed{A}] \right\rangle \\ \text{OUTPUT} & \left\langle \text{Y}, [\text{STOP-GAP} \ \boxed{A}] \right\rangle \end{bmatrix}$$

To handle the top of LDDs, we introduced lexical entries for *easy*-type adjectives and we also introduced one new grammar rule – the Head-Filler Rule, which licenses sentences with a 'topicalized' initial element:

Head-Filler Rule

$$[\textit{phrase}] \rightarrow \boxed{1}[\text{GAP} \ \langle\ \rangle] \quad \mathbf{H}\begin{bmatrix} \text{HEAD} & \begin{bmatrix} \textit{verb} \\ \text{FORM} & \text{fin} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle\ \rangle \\ \text{COMPS} & \langle\ \rangle \end{bmatrix} \\ \text{STOP-GAP} & \langle \boxed{1} \rangle \\ \text{GAP} & \langle \boxed{1} \rangle \end{bmatrix}$$

To properly constrain the values of GAP and STOP-GAP throughout our trees, we made minor revisions to the 'initial symbol' clause of the definition of well-formed tree structure and to the Head-Specifier, Head-Modifier and Imperative Rules:

Φ is a Well-Formed Tree Structure according to G if and only if:

the label of Φ's root node satisfies the constraint:

$$\begin{bmatrix} \textit{phrase} & \\ \text{SYN} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \textit{verb} \\ \text{FORM} & \text{fin} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle\ \rangle \\ \text{COMPS} & \langle\ \rangle \end{bmatrix} \\ \text{GAP} & \langle\ \rangle \end{bmatrix} \end{bmatrix}$$

Head-Specifier Rule

$$\begin{bmatrix} \textit{phrase} & \\ \text{SPR} & \langle\ \rangle \end{bmatrix} \rightarrow \boxed{1} \quad \mathbf{H}\begin{bmatrix} \text{SPR} & \langle \boxed{1} \rangle \\ \text{COMPS} & \langle\ \rangle \\ \text{STOP-GAP} & \langle\ \rangle \end{bmatrix}$$

Head-Modifier Rule

$$[\textit{phrase}] \rightarrow \mathbf{H}\boxed{1}\begin{bmatrix} \text{COMPS} & \langle\ \rangle \\ \text{STOP-GAP} & \langle\ \rangle \end{bmatrix} \begin{bmatrix} \text{COMPS} & \langle\ \rangle \\ \text{MOD} & \langle \boxed{1} \rangle \end{bmatrix}$$

Imperative Rule

$$\begin{bmatrix} \textit{phrase} & \\ \text{SYN} & \begin{bmatrix} \text{HEAD} & \text{verb} \\ \text{VAL} & [\text{SPR} \ \langle\ \rangle] \\ \text{GAP} & \boxed{A} \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \text{MODE} & \text{dir} \\ \text{INDEX} & s \end{bmatrix} \end{bmatrix} \rightarrow \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \textit{verb} \\ \text{FORM} & \text{base} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle \text{NP}[\text{PER} \ \text{2nd}] \rangle \\ \text{COMPS} & \langle\ \rangle \end{bmatrix} \\ \text{GAP} & \boxed{A} \end{bmatrix} \\ \text{SEM} & [\text{INDEX} \ s] \end{bmatrix}$$

Finally, to deal with the Element Constraint (part of Ross's Coordinate Structure Constraint), we modified the Coordination Rule as follows:

Coordination Rule

$$\begin{bmatrix} \text{FORM} & \boxed{1} \\ \text{VAL} & \boxed{0} \\ \text{GAP} & \boxed{A} \\ \text{IND} & s_0 \end{bmatrix} \rightarrow$$

$$\begin{bmatrix} \text{FORM} & \boxed{1} \\ \text{VAL} & \square \\ \text{GAP} & \boxed{A} \\ \text{IND} & s_1 \end{bmatrix} \ldots \begin{bmatrix} \text{FORM} & \boxed{1} \\ \text{VAL} & \boxed{0} \\ \text{GAP} & \boxed{A} \\ \text{IND} & s_{n-1} \end{bmatrix} \begin{bmatrix} \text{HEAD} & \textit{conj} \\ \text{IND} & s_0 \\ \text{RESTR} & \langle [\text{ARGS} \ \langle s_1 \ldots s_n \rangle] \rangle \end{bmatrix} \begin{bmatrix} \text{FORM} & \boxed{1} \\ \text{VAL} & \boxed{0} \\ \text{GAP} & \boxed{A} \\ \text{IND} & s_n \end{bmatrix}$$

14.9 Further **Reading**

Ross 1967 is probably the most influential work to date on the topic of long-distance dependencies. Chomsky (1973, 1977, 1986a) developed one of the most influential approaches to analyzing these constructions, using transformations. The treatment presented here is based loosely on that developed in Pollard and Sag 1994, which is compared with transformational approaches in Levine and Sag 2003. This analysis is unusual in not positing an empty category (a trace) in the position of the gap. Arguments for such a traceless analysis are discussed by Sag and Fodor (1994). Other nontransformational treatments are presented in Gazdar 1981, Kaplan and Zaenen 1989, Steedman 2000, and Bouma et al. 2001.

14.10 Problems

Problem 1: A Tree with a Gap

Draw a tree for (9b). Use abbreviations for the node labels, and show the value of GAP on all nodes. Show the value of STOP-GAP on any node where it is non-empty.

Problem 2: Blocking Filled Gaps

Examples (i) and (ii) are well-formed, but example (iii) is ungrammatical:

(i) Pat thinks that I rely on some sort of trick.
(ii) This mnemonic, Pat thinks that I rely on.
(iii) *This mnemonic, Pat thinks that I rely on some sort of trick.

Explain in detail why the mechanisms that license (i) and (ii) do not also permit (iii).

Problem 3: Subject Gaps

This problem is to make sure you understand how our analysis accounts for examples like (47).

A. Sketch the family of lexical sequences for *likes* that is the input to the Subject Extraction Lexical Rule.

B. Sketch the family of lexical sequences for likes that is the corresponding output to the Subject Extraction Lexical-Rule.

C. Sketch the tree for the sentence in (47b). Use abbreviations for node labels, but show the value of GAP on all nodes and the value of STOP-GAP on any node where it is non-empty. You may abbreviate the structure over the NP *oysters on the* half-shell with a triangle.

D. Does our analysis correctly predict the contrast between (47b) and (i)?
(i)*Those candidates, I think likes oysters on the half-shell.
Explain why or why not.

Problem 4: Irish Complementizers

Consider the following example that shows the typical word order pattern of Modern Irish (data from McCloskey 1979):

(i) Shíl mi! goN mbeadh si! ann.
thought I COMP would-be he there
'I thought that he would be there.'

Irish is a VSO language. One way of analyzing such languages is to posit a Head-Specifier-Complement Rule that introduces both kinds of dependents as sisters of the lexical head. In addition, we'll need a Head-Complement Rule that realizes only complements, and requires the head daughter to be [SPR ⟨ ⟩].

A. Formulate these two rules and show the structure for sentence (i). You may use abbreviations such as NP, S, etc., but be sure to show the value of COMPS and SPR on each node.

Now consider some further Irish data:

(ii) Dúirt mé gurL shil mé goN mbeadh sé ann.
said I goN.PAST thought I COMP would-be he there
'I said that I thought that he would be there.'

(iii) an fear aL shil mé aL bheadh ann
the man COMP thought I COMP would-be __ there
'the man that I thought would be there'

(iv) an fear aL dúirt mi! aL shil mi! aL
the man COMP said I COMP thought I COMP
bheadh ann
would-be __ there
'the man that I said I thought would be there'

(v) an fear aL shil goN mbeadh sé ann
[the man]$_j$ COMP thought __ COMP would-be he, there
'[the man], that thought he$_j$ would be there'

(vi) an fear aL dúirt sé aL shil goN
the man COMP said he COMP thought __ COMP
mbeadh sé ann
would-be he there
'the man that he said thought he would be there'

The complementizers *goN* and *aL* are in complementary distribution. That is, wherever *goN* is possible in these examples, *aL* is not, and vice versa.[9] Assume that both these elements are heads of CPs similar to those headed by that complementizers in English. If we then make the further assumption that LDDs in Irish work much as they do in English, we have all the tools we need to analyze the contrasts in (i)–(vi).[10]

B. Provide lexical entries for these two complementizers. [Note: You may assume for purposes of this problem that the type *comp-lxm* which we proposed for English is applicable to Irish as well.]

C. Show how your analysis successfully explains the distributional differences between the two complementizers. Be sure to cite the data given in the problem.

[9] For the purposes of this problem, you should ignore the difference between *gurL* and *goN*.
[10] Examples (iii)–(vi) involve relative clauses, which we have not discussed in much detail. Assume that the complementizers are the same whether they appear in relative clauses or in CP complements to verbs.

Problem 5: Nested Dependencies

We have made GAP a list-valued feature, which leaves open the possibility of multiple GAPs. This problem considers sentences which instantiate this possibility, such as (i) and (ii):

(i) Problems this involved, my friends on the East Coast are hard to talk to __ about __

(ii) Violins this well crafted, these sonatas are easy to play __ on __ .

A. Indicate which NP is interpreted as the filler for each of the gaps in (i) and (ii).

B. Draw a tree for sentence (i), indicating the value of GAP and STOP-GAP on every node. You do not need to include other features, though you should have a node label (e.g. VP, PP, etc.) on each node, and use tags and coindexing as appropriate. You may abbreviate the structure of the NPs *problems this involved* and *my friends on the East Coast* with triangles.

The PP complements of *talk* can actually appear in either order:

(iii) Dana tried to talk about it to everyone in the building.

(iv) Dana tried to talk to Leslie about this problem.

For the sake of this problem, we will assume that this is dealt with by allowing two distinct ARG-ST lists for *talk*: ⟨ NP , PP[about] , PP[to] ⟩ and ⟨ NP , PP[to] , PP[about] ⟩.

However, when we switch the order of the PPs in the multiple gap example, we get a sentence with a bizarre meaning (in which someone is talking to problems about people):

(v) Problems this involved, my friends on the East Coast are hard to talk about __ to __

C. Is this predicted by our analysis of LDDs? Why or why not?
[Hint: Remember that the value of GAP is a list, and the order of the GAP list on phrasal nodes is determined by the GAP Principle.]

Problem 6: Binding and LDDs

Assuming that reciprocals are [MODE ana], does our analysis of LDDs interact with the Binding Theory to predict that (i) should be grammatical? Why or why not?

(i) [Those people]$_i$ I tend to believe will tell [each other]$_i$ everything.

15

Variation in the English Auxiliary System

15.1 Introduction

English auxiliaries constitute a particularly interesting syntactic system, involving a small set of words that exhibit many intricate interactions and some fascinating idiosyncrasies. This system is peculiar to English, though many other languages have elements with intriguingly parallel properties.

English auxiliaries have changed considerably over the last thousand years or so, and their evolution has been well documented, making them a natural domain for studying syntactic change.[1] Change begins with variation, and the auxiliary system is also the locus of some fascinating differences among varieties of English. Variation is interesting in its own right, but studying it also helps us to ascertain which properties of our grammar we should formulate as or deduce from general principles, and which ones we should treat as essentially accidental.

In this chapter, we provide two examples of variation in English auxiliaries. The first example concerns the behavior of *have*, whose syntax is that of an auxiliary verb in some instances but not in others. The second example deals with a much studied phenomenon in the dialect known as African American Vernacular English (AAVE, for short). In both cases, we will explore how the variation might be handled within our theory. Our examples and discussion are intended only as samples of how data on variation might be relevant to work on syntactic theory. Syntactic variation is a topic worthy of a textbook in its own right, and we make no pretense of doing it justice here.

15.2 Auxiliary Behavior in the Main Verb *Have*

Our first example of variation in the English auxiliary system is one that occurs in a number of varieties of English, and can even vary for individual speakers, depending on the formality of the context: In some circumstances, certain uses of *have* as a main verb exhibit auxiliary-like behavior. The examples in (1) were judged acceptable by the speakers of British English we consulted: (Speakers of other varieties may well have heard such examples, even if they wouldn't produce them.)[2]

[1]Discussion of the history of the English auxiliary system, though an exciting topic, would take us too far afield.

[2]Our small survey on these data also included some speakers from Australia, New Zealand and India. These sentences were most uniformly accepted by the British speakers, however, and it is their judgments