The sausage machine:  
A new two-stage parsing model*

LYN FRAZIER and JANET DEAN FODOR  
University of Connecticut

Abstract

It is proposed that the human sentence parsing device assigns phrase structure to word strings in two steps. The first stage parser assigns lexical and phrasal nodes to substrings of roughly six words. The second stage parser then adds higher nodes to link these phrasal packages together into a complete phrase marker.

This model of the parser is compared with ATN models, and with the two-stage models of Kimball (1973) and Fodor, Bever and Garrett (1974). Our assumption that the units which are shunted from the first stage to the second stage are defined by their length, rather than by their syntactic type, explains the effects of constituent length on perceptual complexity in center embedded sentences and in sentences of the kind that fall under Kimball's principle of Right Association. The particular division of labor between the two parsing units allows us to explain, without appeal to any ad hoc parsing strategies, why the parser makes certain 'shortsighted' errors even though, in general, it is able to make intelligent use of all the information that is available to it.

1. Introduction

We will argue that the syntactic analysis of sentences by hearers or readers is performed in two steps. The first step is to assign lexical and phrasal nodes to groups of words within the lexical string that is received; this is the work of what we will call the Preliminary Phrase Packager, affectionately known as the Sausage Machine. The second step is to combine these structured phrases into a complete phrase marker for the sentence by adding higher nonterminal nodes; the device which performs this we call the Sentence

*We dedicate this paper to the memory of John Kimball, whose proposals about sentence parsing, as will become clear, have had a considerable influence on our own.  
Requests for reprints should be addressed to Janet Dean Fodor, University of Connecticut U-145, Storrs, Conn. 06268.
Structure Supervisor. These two parts of the sentence parsing mechanism have very different characteristics, and this provides an explanation for the relative processing complexity of certain types of English sentence. The Preliminary Phrase Packager (PPP) is a ‘shortsighted’ device, which peers at the incoming sentence through a narrow window which subtends only a few words at a time. It is also insensitive in some respects to the well-formedness rules of the language. The Sentence Structure Supervisor (SSS) can survey the whole phrase marker for the sentence as it is computed, and it can keep track of dependencies between items that are widely separated in the sentence and of long-term structural commitments which are acquired as the analysis proceeds.

The significant properties of this model can be brought out by comparing it with other two-stage parsing models that have been proposed. (By “two-stage model” we mean one in which the syntactic analysis of a sentence is established in two steps, one temporally prior to the other, regardless of when and how semantic properties of the sentence are determined.) Kimball (1973) proposed a model in which the first stage parser connects each lexical item as it is encountered into a phrase marker for the whole sentence. As nodes and branches are added to this phrase marker on the right, phrasal units which have been completed are snipped off from the left and shunted to the second stage parser where they are reassembled for further processing. The second stage parser may perform some semantic interpretation but it also apparently has the task of associating transformationally moved constituents with their deep structure positions in the phrase marker. Fodor, Bever, and Garrett (1974, and references therein) have made a number of proposals which, as far as we can determine, add up to the following model. The first stage parser scans the input string of lexical items for cues to the location of clause boundaries, and divides the string at these points. For each clausal unit that has been isolated, it determines the within-clause constituent structure, and then shunts it to the second stage parser whose task is to establish the configuration of these clausal units in the sentence as a whole.

These two models differ with respect to the nature of the units which are shunted between the two parsing devices, and also with respect to whether these units are attached together before they are shunted. The Fodor, Bever and Garrett parser shunts only clauses, while the Kimball parser shunts all phrasal units regardless of their syntactic type or size. In the Fodor, Bever and Garrett system, only the second stage parser is concerned with how the clausal units fit together, while in the Kimball system it is the first stage parser which determines the arrangement of all phrasal and clausal units within the phrase marker and they are shunted to the second stage parser with pointers which specify exactly how they should be reassembled. We will argue that both of these earlier models are incorrect about the type of constituent that is shunted, and that in fact the shunting unit is determined by its size rather than by its syntactic status. Our first stage parser, the PPP or Sausage Machine, will analyze a string of several (seven plus or minus two?) words at a time,¹ and these may constitute a clause (e.g., After we drove to Calais) or even two clauses (e.g., I hope you’re sorry) but only often a sub-clausal phrase (e.g., the man in the green raincoat). We will also argue that the first stage parser is responsible only for forming these phrasal/clausal packages, and that the second stage parser has to decide how to connect them together, as in the Fodor, Bever and Garrett model.

Before turning to the specific evidence for these claims, we note that there is some general motivation for a two-stage model of the human parsing mechanism, viz., that there appears to be a rather severe limit on the capacity of working memory. In a single-stage parser, which constructs and retains the phrase marker for the whole sentence, the available computation space will inevitably decrease as more and more of the lexical string is processed. But it is well known that sentence length is not a good predictor of sentence complexity for the human parser. In a two-stage parser, the demand on working memory can be kept within reasonable limits without putting the system under excessive strain towards the end of a long sentence, for partial analyses can be cleared from the first stage parsing unit as they are established. Of course, it is assumed that the second stage parser has the capacity to represent the complete sentence. But this does not undermine the argument from memory limitations, for it is a well-attested (if unexplained) fact about human memory that the more structured the material to be stored, the smaller the demand it makes on storage space. If it is assumed that the first stage parser assigns a certain amount of low level structure to a previously unstructured word string, and that the second stage parser groups the resulting units into a full phrase marker, then each stage will be handling roughly the same number of units at the relevant level of structure. The appearance of a difference in storage capacity between them would simply be due to the fact that n structural units at the first stage would subtend only a small fragment of the sentence, while n structural units at the second stage could accommodate the whole sentence.

¹The capacity of the PPP may be defined not in terms of words but in terms of syllables or morphemes or conceivably in terms of time. Its proper definition is a very interesting question, but we have not attempted to disentangle all of these alternatives. Later, we will suggest that the amount of material that can be simultaneously viewed by the PPP may be influenced by the complexity of the syntactic computations that have to be performed over it.
This general argument from capacity limitations rests in part on the assumption that the phrasal structure assigned to a lexical string is not stored separately from the decision making component of the parser, as in the ATN (augmented transition network) models of the parser proposed by Woods (1970), Kaplan (1972, and references therein). The representation of a sentence which these ATN parsers compute is stored in special registers; however large it grows, it does not hamper the processing of subsequent words in the string. It is also, however, completely inaccessible to the decision making unit which is responsible for processing subsequent words, and there is evidence that this aspect of the model is incorrect.

One of Kimball's (1973) seven parsing principles states that, where there is a choice, the human sentence parsing mechanism favors analyses in which an incoming word is attached into the phrase marker as a right sister to existing constituents and as low in the tree structure as possible. We will discuss this claim in detail in section II below, and argue that it is basically correct (though we will propose some important modifications). Its significance is that it attributes to the parser a general preference which is defined over the geometry of the phrase marker, regardless of which particular phrase types are involved. An ATN parser could certainly designed so that it would make exactly the same decision at choice points as the Kimball parser. But because its decisions are determined by the ranking of arcs for specific word and phrase types, rather than in terms of concepts like 'lowest leftmost node in the phrase marker', the parser's structural preferences would have to be built in separately for each type of phrase and each sentence context in which it can appear. Evidence that the human sentence parser exhibits general preferences based on the geometric arrangement of nodes in the phrase marker indicates that its executive component does have access to the results of its prior computations. Its input at each choice point must consist of both the incoming lexical string and the phrase marker (or some portion thereof) which it has already assigned to previous lexical items.

Two quite different explanations of relative sentence complexity can be based on the assumption that a restriction on working memory plays a significant role in the operations of the parser. One explanation relates sentence

complexity to how many nodes of the phrase marker must be stored simultaneously. The other relates sentence complexity to how many mistakes the parser makes in computing the phrase marker because it cannot store many nodes simultaneously. The model outlined in Kimball (1973) is compatible with either type of explanation. But in his 1975 paper, Kimball opted for the former, which we believe to have been the wrong choice.

Kimball noted, for example, that sentence (1) is more easily parsed with the adverb yesterday attached within the lower clause rather than beneath the S node of the main clause.

(1) Tom said that Bill had taken the cleaning out yesterday.
The higher clause attachment would require the first stage parser to retain the top S node of the phrase marker until the end of the sentence. For the preferred lower clause attachment, this S node can be shunted to the second stage parser at a much earlier point in the processing. Sentence complexity thus correlates with the load on memory in the first stage unit. The problem with this explanation is that it predicts either that the first stage parser will be totally unable to accommodate enough nodes to allow it to compute the less preferred reading of a sentence like (1), or else that both interpretations of (1) should be equally complex. Until it reaches the end of the sentence, the first stage parser has no way of knowing whether or not it will need to retain the highest S node for further attachments. If it decides to do so, then it will subsequently be in a position to make either attachment of the adverb. But whichever attachment it makes, the demands on memory will have been the same, and so the two interpretations of the sentence should be equally easy or difficult to compute. If on the other hand the parser decides not to retain the top S node, then it will subsequently only be able to attach the adverb within the lower clause. This predicts the preference for lower attachment in (1), and also the difficulty of computing the only coherent interpretation of an unambiguous example like (2).

(2) Tom said that Bill will take the cleaning out yesterday.
But notice that the explanation has now completely changed its emphasis. A sentence like (2) is difficult not because the memory of the first stage parser is overloaded with too many nodes, but precisely because it has avoided overloading itself by relinquishing the top S node.

In general, 'garden path' explanations of processing difficulty account for asymmetries between sentences where pure memory load explanations do not. Our own model is of the garden path variety. The parser chooses to do whatever costs it the least effort; if this choice turns out to have been correct, the sentence will be relatively simple to parse, but if it should turn
out to have been wrong, the sentence will need to be reparsed to arrive at the correct analysis. The fact that hearers are not always conscious of having made a mistake in the analysis of such sentences (as they are for notorious garden path sentences like *The horse raced past the barn fell*) is not, we submit, a good argument against this kind of account of perceptual complexity (see Marcus, 1978).

To summarize: the assumption of a limit on working memory enters into our model only indirectly. It is the motivation for the division of the syntactic parser into two substages. But this division then determines restrictions on the information that is available to each stage. In particular, the Preliminary Phrase Packager must make its decisions in ignorance not only of what will come later in the sentence but also of what came before, since some of the nodes that have already been established in the phrase marker will have been shunted to the Sentence Structure Supervisor. The PPP will therefore fail to recognize certain legitimate attachment possibilities for the lexical items it is processing. This, we will argue, is the source of Kimball's principle of Right Association. To establish this connection, however, it is necessary to make the shift that we have described in the division of labor between the two parsing units: the first stage parser is not responsible for constructing the whole phrase marker but only determines its lower lexical and phrasal nodes.³

Many of the parsing strategies that have been proposed in the literature are *ad hoc* in the sense that, though they do account for the relative processing complexity of certain classes of sentences, there is no explanation for why the human parsing mechanism should employ these strategies rather than some quite different ones (e.g., their inverses). What we learn from studying these strategies, therefore, is simply the values of certain parameters in the system (see Kaplan, 1972). Since a variety of quite different parsing mechanisms can apparently incorporate equivalent strategies, little light is shed on the basic structure of the system — the nature of its subcomponents, the amount and type of information transmitted between them, temporal relations between their activities, and so on. Kimball's interest was in 'strategies' which are not merely arbitrary rankings of alternative analyses but which follow from — and hence reveal — the fundamental organizing principles of the parser. Our modifications of his model permit this kind of explanation to be extended still further. That is, the parser's decision preferences can be seen as automatic consequences of its structure.

II. The Shortsightedness of the Parser

The principle of Right Association states that "terminal symbols optimally associate to the lowest nonterminal node". This predicts the preferred interpretation of sentence (1) above, the difficulty of sentence (2), and covers a wide range of other examples. The verbal particle *up* in (3) is more naturally associated with *smashed* in the lower clause than with *called* in the higher clause.

(3) Joe called the friend who had smashed his new car up.

The prepositional phrase *to Mary* in (4) tends to be associated with *the letter* rather than attached to the higher NP node which dominates the whole of *the note, the memo and the letter*, or to the VP node which is higher still.

(4) John read the note, the memo and the letter to Mary.

The possessive *'s* in (5) most naturally associates with just the lower noun phrase *Mary* rather than with the higher noun phrase *the boy whom Sam introduced to Mary*.

(5) I met the boy whom Sam introduced to Mary's friend.

The relative clause in (6) is attached beneath the NP node that dominates *the job* rather than higher up at the level of VP or S, where it would be interpreted as having been extrapoosed from the subject noun phrase.

(6) The girl took the job that was attractive.

The preference for low attachments is so strong that it persists even in unambiguous sentences where only a higher attachment would lead to a syntactically and semantically coherent phrase marker. Sentences (7) - (10) all tend to be misanalyzed on a first pass.

(7) Joe looked the friend who had smashed his new car up.
(8) John read the note, the memo and the newspaper to Mary.
(9) I met the boy whom Sam took to the park's friend.
(10) The girl applied for the jobs that was attractive.

This structural generalization over a variety of otherwise disparate sentence types is impressive, and the preferences that it accounts for are sufficiently robust to be accessible to intuition (though experimental confirmation of

³In this respect our model is similar to other 'chunking' models that have been proposed (for example, by Thorne, Bratley and Dewar, 1968, and Limber, 1970, as well as Fodor, Bever and Garrett, 1974). But we should emphasize that we follow Kimball in assuming that the boundaries of the phrasal chunks are established *in the course of* determining the details of the within-phrase structure, rather than as a priori and independent step.
them would not be amiss). We therefore accept that something like the principle of Right Association is operative within the human parsing mechanism, and turn to the question of why this should be so.

Kimball considered, but then rejected, the idea that Right Association is simply an automatic consequence of the shunting procedure which removes parts of the phrase marker from the first stage parsing unit. His principle of Closure states that “a phrase is closed as soon as possible, i.e., unless the next node parsed is an immediate constituent of that phrase”; and the principle called Processing requires that “when a phrase is closed, it is pushed down into a syntactic (possibly semantic) processing stage [the Processing Unit, or PU] and cleared from short-term memory”. Kimball considered an abstract tree structure of the form (11) to which the incoming item $k$ must be attached.

$$
\begin{array}{c}
A \\
B \quad C \\
\quad d \quad E \\
\qquad a \quad B \quad c \quad h \quad k
\end{array}
$$

It follows from Closure that by the time item $k$ is encountered, the only part of (11) which will still be represented in the first stage parsing unit will be the node $E$ and the items it dominates; the attachment of $B$ beneath $E$ would not have resulted in an immediate constituent being added beneath $A$, and therefore $A$ and all of its dependent structure would have been snipped off and shunted to the PU before $k$ was received. Therefore the only possible attachment of $k$ would be as a constituent within $E$; a higher attachment under $A$ would require $A$ to be called back from the PU and would run foul of the principle of Fixed Structure, which states that “when the last immediate constituent of a phrase has been formed, and the phrase closed, it is costly ever to have to go back to reorganize the constituents of that phrase”.  

Kimball decided not to identify Right Association with Closure because he considered that these two principles are opposed to each other in a phrase like (12).

(12) old men who have small annual pensions and gardeners with thirty years of service

Both Kimball’s argument and the demonstration that it is incorrect are extremely intricate, so we will finesse this objection and concentrate on the advantages to be gained from attributing Right Association to the narrow window that the first stage parser has on the sentence.

First, the tendency towards low right association of an incoming constituent sets in only when the word is at some distance from the other daughter constituents of the higher node to which it might have been attached. As Kimball noted, there is little or no pressure in a sentence like (13) to attach the prepositional phrase for Susan as a modifier within the object noun phrase. The higher attachment (14a) even seems to be preferred to the lower attachment (14b). (We propose an explanation for this preference in section IV below.)

(13) Joe bought the book for Susan.

(14) a.  

```
S   
| NP | VP |
| V  | NP | PP |
| bought | Det | N | P | NP |
| Joe | the | book | for | Susan |
```

$^4$This formulation of Closure could be interpreted in two different ways, depending on whether the “unless” clause means that the next node parsed must be analyzed as an immediate constituent of the phrase, or whether it means that the parser will in fact analyze the next node as an immediate constituent of the phrase. On the first interpretation, Closure incorporates an attachment strategy, i.e. it requires that, where there is a choice, an item should be attached into the phrase marker in such a way that the current phrase can be closed. On the second interpretation, attachment decisions in cases of temporary ambiguity would be left to the guidance of other strategies; depending on where an incoming item was attached, Closure would merely indicate which phrases should be considered closed. We are assuming this weaker interpretation here. (See also footnote 6, page 299.)

$^5$In what follows we will argue that the point at which shunting occurs is not governed by Closure but simply by the limited capacity of the first stage parser. However, the general explanation still holds: certain attachments of items are ruled out because the relevant nodes have already been shunted to the second stage parser.

$^6$Kimball’s argument depends heavily on the details of the phrase structure rules which he assumed for English, which are highly debatable. It also depends on construing Closure as an attachment strategy. We know of no unequivocal evidence for the attachment strategy interpretation of Closure. In fact, there is considerable evidence (see Frazier, 1978) that the parser prefers to keep phrases open as long as possible, rather than closing them as soon as possible. And this preference does not conflict with, but actually follows from, the parser’s preference for low right attachments.
(14) b.

\[
\text{NP} \quad \text{VP} \quad \text{S}
\]

\[
\text{N} \quad \text{V} \quad \text{NP} \quad \text{PP}
\]

\[
\text{Det} \quad \text{N} \quad \text{P} \quad \text{NP}
\]

\[
\text{S} \quad \text{book} \quad \text{for} \quad \text{Susan}
\]

Sentence (13) contrasts in this respect with (4) and (8) above, and also with (15).

(15) Joe bought the book that I had been trying to obtain for Susan.

This dependence of right association on constituent length is exactly what would be predicted if right association is due to the shunting of previously processed material. Let us suppose for the sake of argument that the first stage parser has the capacity to retain six words of the sentence, together with whatever lexical and phrasal nodes it has assigned to them. Then in processing (13), it will still be able to ‘see’ the verb when it encounters for Susan. It will know that there is a verb phrase node to which the prepositional phrase could be attached, and also that this particular verb is one which permits a for-phrase. But in sentence (15), where a long noun phrase follows the verb bought, the first stage parser will have lost access to bought by the time for Susan must be entered into the structure; the only possible attachment will be within the long noun phrase, as a modifier to trying to obtain.

The alternative to this explanation of right association is the assumption that the first stage parser does have access to both lower and higher nodes, and that it simply chooses to make the lower attachment for some independent reason. If this were so it would be natural (though not obligatory) to predict that the difficulty of a sentence is a function of how high an incoming constituent must be attached. Kimball suggested that this was so, but all of the evidence seems to be against it. In sentence (16) there are in principle three different positions in which yesterday might be attached, but the lowest position is ruled out on semantic grounds.

(16) Joe said that Martha claimed that 1984 will be blissful yesterday.

Our informants disagree as to which of the two remaining analyses of (16) is preferable. Some favor the highest attachment for the adverb, but they tend to read the sentence with a large pause before the final word, making it an afterthought which modifies the whole sentence but is perhaps not fully integrated into it. In any case, it is generally agreed that both higher attachments are awkward, and more or less equally so; neither of them is at all comparable in acceptability to a lowest-clause attachment when this is available. Once again, this is exactly what we would expect if right association is the result of nodes being shunted out of the first stage parsing unit. Once a node has been shunted, it is unavailable to the first-stage parser for further attachments regardless of what its position in the phrase marker is or how long ago the shunting occurred.

Up to this point, the data are compatible with the kind of snipping and shunting process which relates the two stages of Kimball’s parser. But there are other sentences in which right association is not manifest which demand a redefinition of the first stage parser’s responsibilities. Sentence (17) has been constructed so that there are in principle three possible attachments of the word yesterday, two as right daughters of clauses that have already been parsed, and one as left daughter of a new clause. The lowest attachment as right daughter, which is favored by right association, is excluded by the tense; the competition is therefore between a high attachment as right daughter and an attachment as left daughter.

(17) Though Martha claimed that she will be the first woman president yesterday she announced that she’d rather be an astronaut.

As we would expect from the violation of right association, this sentence is not fully natural (especially when presented without internal punctuation or read without a major intonation break, as it must be to test the preference between competing interpretations). But it seems clear that the grouping yesterday she announced is preferred over the grouping Martha claimed ... yesterday. This preference cannot be due to a limit on the height of an attachment, because yesterday is attached even higher in the preferred phrase marker (18a) than in the alternative (18b).

This observation suggests that the important parameter may not be the height of the attachment in the phrase marker, but rather how local it is relative to other words in the lexical string. Since yesterday in (17) cannot coherently be grouped with the words immediately on its left, it is grouped with those immediately on its right rather than with more distant words on
Because a conjunction of noun phrases is a "flat" structure, rather than right branching, the attachment of to Mary as a modifier of the noun phrase would be at the same height in the phrase marker in all of (20) - (22). But as the noun phrase grows longer, the inclination to attach to Mary to just the closest sub-constituent (the letter) becomes stronger.

A preference for local association of a word with other nearby words is unexplained if the first stage parser is building up a complete phrase marker for the sentence. It is easily accommodated, however, if we assume that the first stage parser has the properties of the Sausage Machine, i.e., that its task is only to assign structure to groups of adjacent words in the lexical string, and to transmit these as separate phrasal packages to a second stage unit which will link them together with higher nodes into a complete phrase marker. A word at ambiguous phrase boundary could be incorporated into a package with the words on its left, or it could become the first word of a new package including the words on its right.

An important prediction of this model is that the preferred attachment of a constituent may differ depending on whether it is effects by the Preliminary Phrase Packager (Sausage Machine) or by the Sentence Structure Supervisor. It is the PPP which has a limited window on the sentence and which will therefore be able to make only local attachments. The SSS, which can view the whole of the phrase marker, will not be so constrained. This predicts different effects for constituents of different length. A long constituent which is packaged up as a separate unit by the PPP will be attached to other phrases by the SSS; it should therefore show no special tendency towards local attachment. Short constituents, which will be grouped together with others in a package by the PPP, are those for which a local attachment should be strongly favored.

This is exactly what the data show. The examples with which Kimball motivated his Right Association principle involve the attachment of verbal particles, possessive 's, single word adverbs, and short prepositional phrases. But there are innumerable sentences in which whole clauses are attached high in the phrase marker without any apparent ill effect on perceptual complexity. In (22), for example, the but-clause has to be Chomsky-adjoined to

---

(19) John read the letter to Mary.
(20) John read the memo and the letter to Mary.
(21) John read the note, the memo and the letter to Mary.
(22) John read the postcard, the note, the memo and the letter to Mary.
the highest S node over the preceding clause; the same is true of the adverbial clause in (24).

(23) We went to the lake to swim but the weather was too cold.
(24) Grandfather took a long hot bath after he had finished his chores.

Our explanation for these cost-free violations of the Right Association principle is that the PPP takes *but* and *after* to be the first word of a new phrasal package and leaves it to the SSS to attach this package into the phrase marker. The same applies to (17) above; if the PPP discovers that it cannot coherently connect *yesterday* into its current package (which will be the complement clause within the *though*-clause), it will then make *yesterday* the first word of a new package, which the SSS can attach without difficulty at the highest level of the phrase marker, as in (18a).

What we have shown is that right (or local) association is sensitive not only to the length of the intervening constituent but also to the length of the constituent to be attached. Notice the striking contrast between (25) and (26).

(25) John threw the apple that Mary had discovered was rotten out.
(26) John threw the apple that Mary had discovered was rotten out of the window and into the rosebush.

The one-word locative in (25) shows a strong inclination to group with the preceding words *had discovered was rotten* even though this makes no sense; the longer locative phrase in (26) is much more easily analysed as modifying the higher and more distant verb *threw*. It might be objected that these sentences differ in their syntax, since *out* is a particle in (25) but a preposition in (26). However the length effect can also be observed in (27) and (28) where this difference is controlled for.

(27) a. Ellen brought the pies that she had spent the whole morning baking to the party.
   b. Ellen brought the pies that she had spent the whole morning baking to the potluck supper at the church hall.

(28) a. Joe got the idea that the coastguard was going to send a liferaft across.
   b. Joe got the idea that the coastguard was going to send a liferaft across to the two men clinging to the sinking canoe.

Theorem 9.3 (The Sausage Machine Model)

The human parsing mechanism is generally very efficient, as long as the sentence provides it with sufficient information to govern its decisions at the time they must be made. But we have argued that it exhibits one specific kind of 'stupid' behavior, which is explicable if the major phrases of a sentence are packaged up by a first stage parser with very limited access to structure that has already been established. Let us consider in detail how this preliminary phrase packager will operate.

We assume that the PPP's 'viewing window' shifts continuously through the sentence, and accommodates perhaps half a dozen words. The degree
of structure which has been assigned to this string of words will be greatest on the left and will decline towards the right, where items may not yet even have been lexically categorized. We will concentrate on an item $X$, roughly in the middle of the window, which has had a lexical node assigned to it but has not yet been linked to other words by means of nonterminal nodes.\footnote{The phrases formed by the PPP do not have to be snipped off from any larger structure in order to be transmitted to the SSS. They might be shunted to the SSS, as in Kimball’s model and in the Fodor, Bever and Garrett model. However, a picture that we find more natural is that of the PPP shuttling its narrow viewing window through a sentence, forming its phrasal packages, and depositing these in the path of the SSS which is sweeping through the sentence behind it. This would have the advantage of increasing the decision lag of the SSS, by permitting the SSS to ‘look over the shoulder’ of the PPP, while the latter is forming its phrasal packages, and start considering their possible attachments before they are complete. It also leaves open the question of whether the PPP’s viewing window shifts in a truly continuous fashion through the sentence, or whether it jumps from one phrasal package to the next. We have not attempted to distinguish these two possibilities here, though they may prove to make different predictions about the details of the PPP’s sensitivity to constituent length.}\
\footnote{We leave open here the question of whether lexical category decisions are based entirely on information retrieved from the mental lexicon or whether they can be influenced by syntactic information. However, in assuming that lexical categorization typically occurs quite early relative to the postulation of higher nonterminal nodes, we are rejecting the idea that category decisions are made solely on the basis of the syntactic structure that has been assigned to prior words. (See footnote 16 on page 217.)}

We assume that the task of the PPP is to group as many items as it can into a single phrasal package. If it were to form only very small packages, there would be more of them relative to the number of words in the sentence; the SSS would therefore be left with more attachment decisions to make, and it would have to make them at a greater rate. Efficient functioning of the system demands that each of the two stages should do as much of the work as it is able to within the limits on capacity at its level of analysis. The item $X$ is therefore to be grouped, if possible, with other nearby items. Grouping it with items on its left should be optimal, for these items have already been assigned some structure into which $X$ could be incorporated. If this is not possible, the PPP’s second choice would be to terminate the package it has been constructing and to start forming a new one, in which $X$ will be the leftmost daughter. What the PPP cannot do, of course, is to attach $X$ as a sister to items on the left which have already passed out of its viewing window.

As we have noted, this characterization of the first stage parsing unit explains a number of related observations: the preference for low right attachments; the sensitivity of this preference to the length of the prior constituent; its sensitivity to the shortness of the constituent to be attached; and the preference for local attachment as a left daughter rather than distant attachment as a right daughter. We now show that this model offers a new explanation for the extreme processing difficulty of center embedded sentences.

The correct analysis of a sentence like (29) goes against the grain of the PPP, because adjacent phrases must not be grouped together into the same phrasal package.

(29) The woman the man the girl loved met died.

If the PPP did package the first six words of (29) together as a phrase, they could only be interpreted as constituting a conjoined noun phrase (missing its and). This, of course, is exactly the kind of misanalysis which Blumenthal (1966) showed that people tend to impose on such sentences.\footnote{The conjunction analysis (i) will also be favored over the (correct) relative clause analysis (ii) by the Minimal Attachment principle discussed in Section IV below.}

Notice also that even if this sentence-initial garden path is avoided in an example like (30), because its noun phrases are not conjoinable, one is still inclined to analyse the sequence of verb phrases as a list or conjunction.

(30) The woman someone I met loved died.

If the whole phrase marker were computed by a single parsing device, this would be difficult to understand. For, having correctly imposed right branching structure on the noun phrases, the parser should be able to predict the subsequent appearance of a tier of three verb phrases. But if it is the PPP which is packaging the verb phrases, we would expect its analysis to be quite local and unaffected by the structure of the preceding noun phrase sequence. The SSS will be aware of the nested structure, of course, but it will be helpless because it will inherit an incorrectly structured VP sequence from the PPP.

The only correct phrasal packages which the PPP could usefully transmit to the SSS, in the case of the center embedded sentence (29), are those shown in (31).
A safe but extremely uneconomical solution would be for the PPP simply to group each determiner with its following noun, and send six separate packages to the SSS – three noun phrases and three verb phrases. For the PPP to form package a (the girl loved) would be a minimal improvement; it would still be under-using its capacity, and there would be five separate packages in this nine-word sentence for the SSS to cope with. Package b (the man the girl loved) would therefore be preferable; it would combine five words, and further reduce the decision density for the SSS by chopping the nine words into only four packages. Packages c, d and e would obviously be better yet, but only if the PPP could achieve them without error. These packages are clearly in danger of exceeding the capacity of the PPP, and if they did, the results would be disastrous. That is, if the PPP were to begin forming one of these packages and then discovered that it could not squeeze in the last word or two, the result would be a non-constituent and the SSS would not inherit the correct units of the sentence to work with.

Let us consider what would be required for the PPP to form package b. First, it would have to avoid grouping NP1 with NP2 and NP3. But it would refrain from making this grouping only if NP1 were long enough to qualify as a separate package of its own. This predicts that a center embedded sentence with a long NP1 should be easier to parse, other things being equal, than sentence (29). Similar arguments apply to the verb phrases. VP1 is to be packaged with the preceding two noun phrases, but VP2 and VP3 are not. Also, VP2 and VP3 must not be packaged together. The optimal situation, therefore, should be one in which both VP2 and VP3 are long enough to be formed into separate packages. Apart from this, all the other constituents (viz., NP2, NP3 and VP1) should ideally be short, to facilitate their being packaged together. These predictions are confirmed by a comparison of (29) with (32), which is considerably easier to parse.

(32) The very beautiful young woman the man the girl loved met on a cruise ship in Maine died of cholera in 1962.

Two alternative explanations of this effect of constituent length can be rejected. The first is that lengthening a constituent inevitably increases its semantic content, and thus provides semantic constraints which can facilitate the analysis. Note, however, that the extra material in (32) in no way restricts the possible pairings of noun phrases with verb phrases. Semantic facilitation may be what is at work in (33), but it is not in (32).

(33) The snow the match the girl lit heated melted.

should be noted that this ratio plays only a tangential role in our own explanation of center embedded sentences. Furthermore, it looks as if there may be a conflict between assuming that complexity increases as this ratio increases, and the fundamental finding that more verbal material can be stored the more structured it is.

The second possible factor affecting package size (which is not open to this objection) is the complexity of arriving at the correct structure for a package. This would be greater, for example, if the PPP were to adopt a faulty hypothesis which it then had to revise. This will be the case in sentence (31) (even in packages like b and c which contain only two of the three noun phrases), because the Minimal Attachment principle (as noted in footnote 10 above) will tend temporarily to garden path the PPP in relative clause constructions.

Assuming, therefore, that the parser’s memory capacity and computational capacity are not entirely independent of each other, it seems plausible to suppose that the PPP’s package size is not constant. And if so, then the center package in a sentence like (31) may be limited to only four or five words even though packages may be perhaps seven or eight words long in other contexts.
In some cases the value of a specific lexical signal for a nonterminal node is clear. A sentence such as He knew the girl in the bakeshop was hungry contains a temporary ambiguity. Because know (like many other verbs in English) can take either a simple direct object or a complement clause, the parser will not know whether to attach the girl in the bakeshop directly beneath the VP node, or beneath an S node subordinate to the VP. If the parser tends not to insert the S node in these circumstances (see the discussion of Minimal Attachment in Section IV below), then the phrase marker will have to be corrected subsequently when the verb phrase was hungry is encountered. The complementizer that before the noun phrase would have resolved the temporary ambiguity and permitted the parser to avoid this error. Hakes' (1972) study of sentences with and without complementizers confirmed that their presence does indeed facilitate parsing.

It is much less clear for many other grammatical words what special role they might play in parsing (over and above their obvious linguistic role of distinguishing singular from plural, conjunction from disjunction, the various spatial relations from each other, and so on). A determiner, for example, requires an NP node to be introduced into the phrase marker, but this only rarely resolves a temporary ambiguity, and in any case the need for an NP node is just as clear if a noun phrase begins with a noun as if it begins with a determiner. Prepositions undeniably signal PP nodes, but they do not resolve the very common uncertainty about whether a prepositional phrase should be attached within a noun phrase or directly under the VP. A conjunction such as and or or demands a superordinate node over both conjuncts, but it arrives after the first conjunct and will often be too late to prevent that conjunct from being incorrectly attached into the phrase marker without the superordinate node above it. Similarly, as Kimball observed, grammatical words typically follow the phrases to which they are adjoined in SOV languages, which suggests – somewhat surprisingly – that natural languages do not try very hard to save the parser from having to go back and insert nodes over constituents that have already been processed.

Despite these problems, some part of New Nodes can be salvaged in the sausage machine model. It is not implausible that grammatical words should have a special signalling function that content words lack. Grammatical words are members of small closed lexical classes. A parser which is able to distinguish these words in the lexical string13 therefore has access to a very superficial source of information about phrasal structure, which it could make use of before any extensive syntactic analysis has been performed.

13 Bradley (1978) provides evidence that “closed class” lexical items are contacted through a special lexical retrieval system.
A second possibility is that lengthening the sentence (without increasing its degree of nesting) simply gives the parser more time to consider possible analyses and select the correct one. But it is easy to show that lengthening constituents is not a general palliative. Sentence (34) has as many long constituents as (32), but they are just the wrong ones from the point of view of the PPP. As predicted, this sentence is more difficult to process than (32), and perhaps even more difficult than (29). 12

(34) The woman the sad and lonely old man the pretty little schoolgirl loved with all her heart met died.

It should be noted that there is no need, in this account of center embedded sentences, to set an arbitrary top limit of two on the number of S nodes that the parser can store simultaneously, or on the number of simultaneous applications of the clause parsing subroutine. The fact that processing difficulty rises so sharply with the degree of embedding falls out automatically. If only the inner two noun phrases and the first verb phrase can be packaged together, there will have to be two packages for a two-clause sentence, four packages for a three-clause sentence, six packages for a four-clause sentence, and so on. As the number of packages increases, so does the problem of establishing the correct package boundaries, the decision pressure on the SSS, and the chance that a package will be attached incorrectly (e.g. as a conjunct).

Previous proposals about the source of the difficulty in center embedded sentences have emphasized memory limitations. Miller and Isard (1960) suggest that there is a limit on memory for re-entry addresses for the clause processing subroutine; others (such as Yngve, 1960, and Kimball, 1975) propose that memory is overloaded by retention of words and nodes in the early clause fragments while they await completion. We cannot review all such proposals in detail here, but we will illustrate, in the framework of a phrase shunting model, what we believe to be a general defect that they all share.

Right branching constructions, unlike center embedded constructions, are easy to parse. In a right branching structure, by definition, a higher clause is complete when the clause embedded in it begins. But the parser does not always have evidence that the higher clause is complete. For all the parser can tell, a right branching sentence such as (35) might have continued as in the center embedded sentence (36).

(35) I saw a boy who dropped the delicate model airplane he had so carefully been making at school.
(36) I saw a boy who dropped the delicate model airplane he had so carefully been making at school into a puddle cry.

An obvious point, which any explanation of center embedded sentences should accommodate, is that, up to the word school, sentence (36) is just as easy to parse as sentence (35); it seems clear that both sentences are processed in exactly the same way. If it is assumed that clause fragments are retained in memory until it is certain that they are complete, then the need to retain clause fragments cannot be the source of difficulty in (36), for just the same fragments will be being retained in (35). If it is assumed instead that incomplete clauses can be shunted to the second stage parser (either generally, or at least when they could be complete clauses), then the memory explanation for (36) also fails. The only special characteristic of (36) would be that it requires that extra phrases can be attached into incomplete clauses after they have been shunted. It might be claimed, of course, that this is what makes (36) difficult (cf. Kimball's Fixed Structure principle), but there is really no evidence that this is so. As we argued in connection with sentences (23) and (24) above, extra material can be attached into an already shunted constituent without any noticeable difficulty - as long as the phrase lengths are not such as to trick the PPP into forming incorrect phrasal packages. (In sentence (36) the last two constituents are too short for comfort; parsing is easier if they are lengthened, as in I saw a boy who dropped the delicate model airplane he had so carefully been making at school into the puddle of mud beside the back door reach down and pick it up by its broken tailfin.)

Our explanation of center embedded sentences does not face this problem. It accepts that both right branching and center embedded sentences can be parsed and shunted phrase by phrase, and it attributes the difficulty of center embedded sentences not to memory overload but to the problem of establishing the correct phrasal units. This has the additional advantage that it allows for the fact that center embedded constructions of different types differ considerably in perceptual complexity. Just how difficult a particular example is will depend on how many opportunities it offers for misassignment of phrase boundaries, how persistently its structure garden paths the
This could be of value to a 'detective' style parser, which gathers up all the words in a clause and processes them together as a sort of structural anagram. Determiners, prepositions, conjunctions and the like would provide immediate clues as to what sorts of phrases the clause contains. These superficial signals would also be useful to the PPP of our model, which needs to be able to look at the next word or two in the sentence and make a rapid decision about whether or not to try to squeeze it into the current phrasal package. As an injunction to the PPP, New Nodes could be reformulated as: if the phrasal package under construction is approaching the limit of (for example) six words, close it at (i.e., to the left of, in English, but to the right of, in Japanese or Turkish) the next grammatical word, and group subsequent words into a new package. Our suggestion, then, is that whether or not grammatical words signal the existence of higher nodes, they signal likely points at which to chunk the lexical string into packages.

This reinterpretation of New Nodes does not imply that every grammatical word triggers the closure of a phrasal package. Rather, the signal will be made use of just in case closure is independently going to be necessary quite soon. Thus phrases like the man in the top hat or that Sam may scream could be packaged up as units by the PPP despite the presence within them of determiners, prepositions, and auxiliary verbs. In fact, we see no reason to restrict the PPP to introducing only certain types of nodes (e.g., only lexical and non-clausal phrase nodes). It will very often happen in practice that higher nodes such as S are introduced only by the SSS. But as long as a sentence or clause is short enough to fall within the scope of the PPP, we assume that the PPP can supply its S node.

The distinction between the two parsing units is therefore not a matter of what kinds of operation each can perform, but only of how much computation each is able to perform on any given sentence. As noted earlier, the sausage machine model differs in this respect both from Kimball's, in which the first stage parser assigns all nodes in the phrase marker, and from Fodor, Bever and Garrett's, in which the first stage parser assigns all and only within-clause phrasal structure. The effects of constituent length on the right association phenomenon suggest that the first stage parser's operations are indeed governed by constituent length rather than by constituent type. And the experiments which have been taken to show the special significance of clausal boundaries (Caplan, 1972, and others reported in Fodor, Bever and Garrett, 1974) must at least be interpreted in the light of Carroll and Tanenhaus's (1975) demonstration of similar effects for long non-clausal noun phrases. 14

IV. The Farsightedness of the Parser

In this section we will do our best to defend two very strong claims about the human parsing mechanism: (a) that the division of the parser into the PPP and the SSS is the only source of constraint on the structural hypotheses it considers; (b) that this division of the parser is the only source of constraint on the sequence of attachment operations it performs in building up a phrase marker. (a) amounts to the claim that there are no special strategies that the parser uses to select a hypothesis to pursue at a choice point in the sentence; (b) amounts to the claim that there is no special schedule that requires the parser to enter some nodes in a phrase marker before the others. Later, we will consider some apparent counter-evidence to both of these claims, but we will argue that it is simply an automatic consequence of the time and memory pressures that the parser is subject to.

Both (a) and (b) are intended to apply to both the PPP and the SSS. The PPP has its weaknesses, as we have argued, but these can all be attributed to its limited capacity. So far we have had little to say about the SSS, except that it does not make the kinds of shortsighted decisions characteristic of the PPP. The assumption that the SSS has virtually unlimited capacity is motivated by the fact that some part of the human parsing mechanism is extremely good at keeping track of syntactic dependencies which span longer stretches of the sentence than the PPP can accommodate. For example, a question beginning with a WH-phrase must have a corresponding 'gap' in the deep structure position of that phrase. If there is no gap, as in (39), the ungrammaticality of the sentence is easy to detect.

(39) *Which student did John take the new instructor to meet the dean?

People are also quick to notice the ungrammaticality of a sentence like (40) which contains a gap where it should not, i.e., in which an obligatory constituent is absent, and its absence is unaccounted for by any transformational rule.

(40) *John took the new instructor to meet.

A parser which did no more than attach incoming words into the phrase marker in accord with the rules of the grammar would serenely analyze a non-sentence like (40) as if it were well-formed. The receipt of each lexical item would occasion the assessing of grammatical rules to determine what nodes may legitimately appear above it, and any item which could not be integrated into the phrase marker in accord with these rules would be readily detected. But the absence of a lexical item would not occasion any rule assessing, and so the parser would be unaware that a constituent required by

14There is also, of course, the possibility that clausal units appear to play a special role in syntactic parsing only because propositional units are of special importance in the semantic analysis of a sentence.
the rules was missing. The fact that people are not misled by sentences like (40) shows that any such model is incorrect. The human parsing mechanism not only processes what it does receive but also makes predictions concerning what it is about to receive.

We note in passing that ATN parsing models have structural predictions built into their network of arcs. The network simply cannot be traversed if an obligatory constituent is missing, because the arc corresponding to that constituent will be the only 'bridge' across part of the network. ATN parsers are therefore efficient at detecting ungrammaticalities of omission as well as those of other kinds. However, in Section I we argued that current ATN models are deficient insofar as the phrase marker that is being constructed is not available as input for decisions about subsequent words in the sentence. It also seems likely that an ATN parser, with its rigidly sequenced arcs, will have difficulty in recovering from ungrammaticalities and making sense of the sorts of partially scrambled sentences that are so common in every day conversation. The human parsing mechanism can detect ungrammaticalities but is not devastated by them. We believe it to be one of the assets of our model that it can reconstruct this resilience.

The PPP will continue to form its phrasal packages, and the SSS will continue to find appropriate locations for them in the overall predictable structure of the phrase marker, even if some aspects of the structure are indeterminate because earlier packages were misordered or absent or of the wrong type. We propose to permit both the PPP and the SSS to postulate obligatory nodes in the phrase marker as soon as they become predictable, even if their lexical realizations have not yet been received. The PPP, for example, upon receiving a preposition or an obligatorily transitive verb can enter an NP node as its right sister before any elements of the noun phrase have been located in the lexical string. A prediction which is more likely in practice to be made by the SSS is that a sentence beginning *Either John is ...* has a coordinate structure and must therefore contain a second S node as sister to the S node over the first clause. If these predicted nodes should continue to dangle for lack of any corresponding lexical items in the sentence, they will signal ungrammaticalities of omission. They will also sometimes serve to reserve what would otherwise be temporary ambiguities in sentences.

The role of the word *was* in the sentence fragment *That the youngest of the children was proved to ...* is unambiguous. The complement clause must contain a verb phrase, and its position in the phrase marker requires that in the lexical string this verb phrase should precede the verb phrase of the main clause. Therefore *was proved to...* can only be attached within the subordinate clause, not within the main clause. But either attachment would *appear* to be legitimate to a parser which did not enter the predictable subordinate VP node *before* attempting to connect *was* into the phrase marker. This is only one of innumerable examples in which node prediction can save a parser from the danger of being garden pathed by potential attachment ambiguities. And informal evidence suggests that people do not respond to such sentences as if they were temporarily ambiguous; the impossible attachment seems never to be contemplated. (Note that in the example above this cannot be accounted for in terms of semantic constraints, and the constituent lengths are such that right association by the PPP is an unlikely explanation.)

It seems, therefore, that the human parsing mechanism does anticipate predictable aspects of the phrase marker.

The prediction of nodes is an accomplishment usually associated with top down parsers. We have proposed a model in which phrase marker construction proceeds bottom up, in the very special sense that the PPP supplies the nodes immediately above lexical items before the SSS connects the resulting phrasal package together with others by means of higher nodes. But claim (b) states that, *within* each stage of the parser, processing is not necessarily bottom up, nor is it necessarily top down or governed by any externally imposed schedule.

Pure bottom up parsing is governed by what we will call a No Incomplete Nodes principle, which stipulates that a node may not be entered into the phrase marker until all of its daughter nodes have been established. This concentrates the work of building the phrase marker rather late in the processing of the sentence, and it also precludes all sorts of potentially useful predictive activity. Since higher nodes cannot be entered until after all the words they dominate have been received, these nodes cannot be made use of for the 'forwards' prediction of words or nodes within that portion of the lexical string.

Pure top down parsing is governed by what we might call a No Orphaned Nodes principle, which stipulates that a node may not be entered into the phrase marker unless all of the higher nodes which connect it to the top S node have also been postulated. This concentrates structural decisions in an extreme form, top down parsing is governed by a constraint on hypothesis formation rather than just a constraint on the entry of nodes into the phrase marker. That is, it requires that the parser should predict the next lexical item and all of the nodes which link it into the phrase marker, on the basis of the nodes which are already present. The only role of the lexical item which actually occurs next in the lexical string is to confirm or disconfirm this prediction.

(Continued overleaf.)
towards the beginning of the parsing process. Since it demands 'upwards' prediction of higher nodes, it does permit considerable 'forwards' prediction of their forthcoming daughter nodes. But it has the disadvantage that errors of analysis may result from the parser's being forced to make decisions about higher nodes before the lexical string makes clear what the right decision is. For example, the parser must decide how to attach the word that in the sentence fragment John told the girl that silly old-fashioned... before processing subsequent words (such as... joke yesterday, or... joke had offended a more amusing story, or... jokes had become the latest craze) which could indicate whether that was a demonstrative determiner, a complementizer introducing a relative clause, or a complementizer introducing a complement clause.

Kimball (1975) rejected both pure bottom up and pure top down schedules. He observed that top down parsing cannot be error-free for natural languages since these contain left-recursive structures, such as noun phrases with other noun phrases as left daughters. A noun phrase may therefore be recognizable as a noun phrase, but carry no indication about how many nodes intervene between it and the top S node. Kimball then considered two intermediate schedules. A Predictive Analyzer (somewhat misleadingly named) enters higher nodes above a lexical item only up to some specified height (e.g., first node above, first two nodes above, all nodes up to and including the first S node above). An Over the Top parser makes forwards as well as upwards predictions; it enters not only a dominating node but also that node's next daughter node. But neither of these schedules gets to the heart of the deficiencies of pure bottom up or top down parsing schedules. They retain some of the disadvantages of each, postulating fewer nodes than they could in some contexts, and more nodes than is safe in others.

The characteristic property of natural language sentences is the variable predictability of parts of their phrase markers. There are no fixed parameters n and m such that exactly n upward nodes and exactly m forward nodes are predictable for each lexical item in each sentence. It is illuminating to consider the sentence fragment (41).

(41) John put the mustard in the ...

A parser making full use of the rules of English grammar could establish with certainty the partial phrase marker (42).

That is, the parser knows that the lexical string must contain another noun as daughter to the NP within the PP, but it does not know how many more words will precede this noun. It knows that the verb phrase must contain an NP and some sort of locative phrase, but it does not know whether these predictable nodes can be identified with the nodes over the mustard and in the... . (The PP node might instead turn out to be a sister to the NP node, with another NP node above them both; this higher NP node might be conjoined with another, under yet another higher NP node; and so on.) Thus the parser can establish three nodes above the word put but only two nodes
above the first the; it can establish the next sister node to put but not the next sister to the second the; it can establish the existence of a second sister to put but not its identity except within certain limits; it can establish the identity of a sister to the second the but not its serial position.

It is hard to imagine any sort of fixed schedule which could permit a parser to represent all the secure facts here without simultaneously forcing it to make dangerous guesses about other unpredictable aspects of the phrase marker. To cope with natural languages with the maximal blend of reliability and efficiency a parser must be purely information-driven, permitted to build up the phrase marker in any order at all and to enter a given node no sooner and no later than it can confidently do so on the basis of the lexical string.

Efficiency and reliability are notable characteristics of the human parsing mechanism but they are not the only relevant considerations. The information-paced parser will not attach a node until it is certain how to attach it. But there is no guarantee that a temporary attachment ambiguity ever will be resolved by the words in the sentence. And even when it is, the parser has no way of knowing how long it must wait for disambiguating evidence. (This is especially true for attachment ambiguities at higher levels of the phrase marker, because the higher a node is, the longer the stretch of the lexical string it can span.) Given that unstructured verbal material is more costly to store in memory than structured material, the waiting strategy may turn out to be even more dangerous than a guessing strategy. Unless nodes are attached together in some fashion, they may be lost to the parser altogether. And there is abundant evidence in the psycholinguistic literature that the human parsing mechanism does make structural decisions in advance of the evidence. The guesses that it favors are usually attributed to specific strategies, or rankings of alternative hypotheses, which guide the parser's activities at choice points in the sentence. In fact, most or all of the familiar examples can be accounted for by one very general strategy, and this strategy can then be explained away in terms of the demand characteristics of the parser's task.

The general strategy is what we will call the Minimal Attachment principle. This stipulates that each lexical item (or other node) is to be attached into the phrase marker with the fewest possible number of nonterminal nodes linking it with the nodes which are already present.18 Minimal Attachment accounts for the preference, noted in Section II, for the attachment of for Susan directly beneath the VP node in a sentence like John bought the book for Susan, where the verb phrase is short enough so that the PPP's tendency towards local attachment into the object noun phrase is not operative. Minimal Attachment accounts for the preference, noted in the discussion of New Nodes, for the direct object analysis of the noun phrase in a sentence fragment like We knew the girl... even though this phrase might equally well be the subject of a complement clause. It accounts for the preference, observed in center embedded sentences, for a conjunctive analysis of an NP NP sequence, rather than an analysis in which the second NP begins a relative clause modifying the first one. It accounts for the preference (noted by Bever, 1970, Bever and Langendoen, 1971, Chomsky and Lasnik, 1977) for analyzing a clause as main rather than subordinate wherever possible. It accounts for the preference (noted by Wanner, Kaplan and Shiner, 1975) for the analysis of a that-clause which follows a noun phrase as a complement clause rather than a relative clause. It even predicts certain lexical category decisions (though perhaps these are also influenced by the nature of lexical access operations). For example, it predicts that the word that is more easily interpreted as a determiner than as a complementizer in a context like That silly old-fashioned joke/jokes .... And it predicts that the verb raced in The horse raced past the ... is more readily interpreted as an active intransitive verb in the main clause than as the passive participle of a transitive verb in a reduced relative clause modifying the horse. Frazier (1978) provides experimental evidence for the operation of Minimal Attachment in a variety of different constructions.

These preferences of the parser are less extreme, perhaps, than those due to the PPP's packaging routines; they can be swayed to some extent by the content of individual sentences. But their overall direction is very clear. Regardless of what sort of constituent is to be attached, or what the alternative attachments are, the simplest attachment is always the one that is favored.

For a parser which is obliged by memory limitations to make structural decisions in the absence of sufficient evidence, Minimal Attachment would be a very rational strategy to adopt. For one thing, the minimal attachment analysis will make the least demand on memory; even if it does turn out to be wrong, it will have been less costly than some other wrong analysis. Furthermore, trying the simplest attachment first ensures that revisions will be uniform – they will all consist of adding extra nodes to the phrase marker. (A Maximal Attachment strategy would also permit orderly revision procedures, but maximal attachments are of course not well-defined in a grammar with recursion.) Minimal Attachment also presupposes minimal rule

---

18 A similar, though weaker, principle is proposed in Kimball (1975, p. 164). "In formal terms, the machine is seeking a path (from some postulated mode) to the root [S node], where certain symbols may be repeated. Let us define an equivalence relation between paths, so that two paths are equivalent if they differ only in repetition of a given symbol. Thus, the paths Det NP S, Det NP NP S, and Det NP NP S S S are all equivalent. We can then pick a canonical representative of such an equivalence class to be the most collapsed string, in this case the first." It is this most collapsed string of nodes that the parser is assumed to postulate when there is a choice. Note that, unlike Minimal Attachment, this principle would not favor the string Det NP S over the longer string Det NP PP S, and it is unclear whether it is intended to favor Det NP VP S over Det NP S VP S.
accessing. If the well-formedness conditions are mentally represented in the form of phrase structure rules, each node between a lexical item and the top S node of the phrase marker will require the accessing of another rule, to determine in what configurations it may properly appear. For example, if an NP node has been established at the beginning of a sentence, Minimal Attachment will require it to be entered as immediate daughter to the top S node, by reference to the rule $S \rightarrow NP \cdot VP$. If additional intervening nodes were postulated instead, they would have to be checked against further rules such as $NP \rightarrow NP \cdot conj \cdot NP$, or $NP \rightarrow Det \cdot N$ and $Det \rightarrow NP \cdot 's$.

This observation about Minimal Attachment is what permits it to be dispensed with as an independent strategy. We need only suppose that the structural hypothesis which the parser pursues is the first one that it recognizes. Establishing the legitimacy of the minimal attachment of a constituent will take less time than establishing the legitimacy of a long chain of linking nodes. In normal conversational contexts sentence parsing has to be performed very rapidly, with little leeway provided by the constant arrival of new words to be processed. It is therefore not at all ad hoc for the parser to pursue whichever structural hypothesis most rapidly becomes available to be pursued, quite apart from the fact that this will also be the easiest one to store and the easiest one to correct if wrong.

We have argued that, when making its subsequent decisions, the executive unit of the parser refers to the geometric arrangement of nodes in the partial phrase marker that it has already constructed. It then seems unavoidable that the well-formedness conditions on phrase markers are stored independently of the executive unit, and are accessed by it as needed. That is, the range of syntactically legitimate attachments at each point in a sentence must be determined by a survey of the syntactic rules for the language, rather than being incorporated into a fixed ranking of the moves the parser should make at that particular point, as in an ATN parser.

We have no direct argument to offer in favor of the further assumption that the well-formedness conditions for the language are stored in the form of phrase structure rules, though such rules have, of course, proved particularly suitable for the linguistic description of natural languages. They serve, however, to characterize only deep structure phrase markers, not the surface phrase markers which determine the actual sequence of lexical items in a sentence. Fodor (1978, in press) argues that these rules are nevertheless applied by the parser to the surface forms of sentences and that discrepancies between surface and deep structures are resolved by restoring constituents which have been deleted or moved from their original deep structure positions.

It is worth noting that transformational dependencies often extend across more words in a sentence than the PPP can accommodate, and that the major burden of determining how to fill in transformationally induced gaps in sentences will therefore fall on the SSS. The prior operations of the PPP would therefore have to be guided by a superset of the phrase structure rules for well-formed deep structures. These rules would allow for transformationally moved and deleted constituents in the surface forms of sentences, but would also, inevitably, let through some similar structures which happen not to be legitimate. This may account for the fact that the PPP, at least when under pressure, is apparently capable of forming some very strange phraseal packages. For example, in the sentence He took the hat, the gloves, the coat and the vest off, there is a tendency to group the words the vest off together even though the lexical and syntactic constraints of English do not allow this as a possible phrase.

It is particularly interesting that for this explanation to go through, it is not even necessary to suppose that the human parsing mechanism considers alternative hypotheses in serial rather than in parallel. Its goal might be to pursue all the legitimate hypotheses simultaneously. But because the alternatives are recognized at different speeds, its parallel processing of them would be staggered. Only the first of them might be available by the time the very next words in the sentence had to be processed. And if further computation is cut short by the arrival of new items, demanding yet more structural decisions, there is the possibility that the minimal attachment will be the only one which is ever recognized. (Without committing ourselves to this idea, we offer it as a possible explanation for the conflicting experimental data on the serial or parallel processing of ambiguous sentences.)

We have shown that claims (a) and (b) are both false as they stand. The structural hypotheses that the parser pursues are systematically restricted by the Minimal Attachment principle as well as by the shortsightedness of the PPP. The sequence in which the parser postulates nodes in the phrase marker is governed by more than the bottom up relation between the two stages, since there is some pressure towards early postulation of higher non-terminal nodes as in a top down system. Nevertheless, we have argued that these further restrictions emerge naturally or inevitably from general limits on the memory and time available for sentence parsing under normal circumstances. Even with this modification, our claims for the explanatory value of the two stage model will no doubt turn out to be too strong, but it is remarkably difficult to come by any clear evidence for further constraints on what the parser may do and when.

The model that we have proposed falls somewhere between the 'detective' model of Fodor, Bever and Garrett, and the more rigidly constrained models inspired by the development of parsing systems for computer languages. Detective models are also driven primarily by the availability of information in the sentence rather than by an externally imposed schedule. But the Fodor, Bever and Garrett parser must have a very considerable decision lag if the internal structure of a clause is not decided until all the words of a clause are available to be juggled into a best-fit structure. Detective models also appear to presuppose some kind of internal attention shifting mechanism, which is governed by strategies of its own, and determines which clues to the structure of the sentence will be attended to first or should be given more weight in cases of conflict. The greater flexibility of such models makes it considerably more difficult to predict exactly what moves the parser will make in response to a given lexical string. These richer models are therefore very difficult to put to a detailed empirical test. The sausage machine model, as we have tried to show, makes some rather precise predic-
tions, and accounts, with the fewest number of ad hoc assumptions, for the peculiar mix of blindness and intelligence that is observed in the human parsing mechanism.

References


Résumé

Dans cet article on propose un mécanisme de segmentation des énoncés qui assigne en deux étapes une structure syntagmatique aux suites de mots. La première méthode de segmentation assigne des noeuds lexicaux et syntagmatiques à des suites de 6 mots environ. La seconde ajoute des noeuds à un niveau supérieur pour lier ces blocs syntagmatiques et obtenir ainsi un marqueur syntagmatique complet.

Ce modèle de segmentation est comparé d'une part aux modèles ATN et d'autre part au modèle en deux étapes de Kimball (1973) et Fodor, Bever et Garrett (1974). Nous pensons que les unités qui passent du 1er au 2e niveau sont caractérisées par leur longueur plutôt que par leur forme syntaxique. Ceci expliquerait les effets de la longueur des constituants sur la complexité perceptuelle des phrases encadrées et des phrases du type de celles qui tombent sous le principe de l'association à droite de Kimball.

La distinction spécifique du travail entre les deux unités de segmentation permet d'expliquer, sans faire intervenir des stratégies ad hoc, certaines erreurs de segmentation même si, en général, il est possible de faire un usage intelligent de toutes les informations disponibles.