It seems intuitively obvious that our comprehension of sentences is “incremental,” in the sense that each successive word seems to contribute to a gradual accumulation of meaning more or less as soon as it is encountered. Not only the “sense,” or language-internal meaning, but also the referential identity of individuals and entities in the domain of discourse seems to become available immediately. That is, we feel that it is not merely semantic interpretation that is incremental, but reference itself.¹

If interpretations are available at every turn in sentence processing, then there is every reason to suppose that the local syntactic ambiguities that abound in natural-language sentences may be resolved by taking into account the appropriateness to the context of utterance of those interpretations, even when the rival analyses are incomplete. Indeed, the possibility that human language processors are able to draw on the full power of their inferential mechanism to resolve ambiguities seems to offer the only possible explanation of the fact that human languages tolerate such an astonishing profusion of local and global ambiguities, and of the fact that human language users are so rarely aware of them.

The “Strong Competence Hypothesis” of Bresnan and Kaplan (Bresnan 1982) embodies the attractive assumption that rules of natural grammar may be expected to correspond directly to the steps that a processor goes through in assembling a given analysis. Under this hypothesis, the only additional components of a processor besides the grammar itself are a mechanism for building interpretable structures according to the rules of the grammar, and a mechanism for coping with local and global ambiguities as to which rule of the grammar to apply (that is, a mechanism for deciding which analysis or analyses to pursue at any given point in a sentence). The property that the Strong Competence Hypothesis postulates is, of course, not a necessary property for a language processor, as Berwick and Weinberg (1983) have pointed out, although its psychological appeal is obvious.
However, when taken in conjunction with our overwhelming intuition that comprehension is incremental, it immediately leads to a paradox. Almost nothing else that we know about natural-language grammar seems to be as we would expect if the hypothesis held.

Take surface structure, for example. There is no particular problem about constructing a grammar that corresponds in this direct fashion with an incremental processor. Any left branching context-free (CF) grammar, generating trees of the form illustrated in (1), provides an example.

(N stands for nonterminal symbols or phrases, and T stands for terminal symbols or words.) If in addition we assume a rule-to-rule compositional semantics, then for each terminal in a left-to-right pass through the sentence, as soon as it is syntactically incorporated into a phrase, the interpretation of that phrase can be provided. And since the interpretation is complete, it may also be evaluated—for example, if the constituent is a noun or a noun phrase, then its extension or referent may be found. In contrast, a right-branching CF grammar, giving rise to trees like that in (2), does not have this property for a left-to-right processor.

In the absence of some further apparatus going beyond rule-to-rule processing and rule-to-rule semantics, all comprehension (as opposed to purely syntactic structure-building) must wait until the end of the string, when the first complete constituent is built and can be interpreted.

If human grammars are indeed incrementally interpreted, as our intuitions suggest, and if the Strong Competence Hypothesis does indeed hold, then we might expect left-branching structures to be the norm among the languages of the world. For example, on the slightly chauvinistic assumption that many or most languages include a category of transitive verbs
whose semantics defines them as first combining with objects to yield predicates which in turn combine with subjects to yield sentences, then the clause structure shown in (3) should be common.

(3)

```
S
  \-----
   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   ```

But of course, VOS languages are not at all common. Nor are OVS languages, which would have the same property. The most common order is SOV, followed closely by SVO. Neither seems as suited to incremental left-to-right interpretation under the Strong Competence Hypothesis.²

(4)

```
S
  \-----
   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   \   ```

English is an obvious example of a predominantly right-branching language, apparently ill-matched under the Strong Competence Hypothesis to the manifest intuition that incremental interpretation is available to its users.

There are a number of ways out of this apparent paradox. Perhaps the easiest is to assume that the Strong Competence Hypothesis is simply wrong, and that steps in processing are not in fact related rule-to-rule with the grammar. This tactic has been popular among linguists, because it does at least leave them with a coherent research program. The competence hypothesis is no more than a heuristic device, for the questions of the psychological reality of incremental interpretation and of the nature of grammar are logically independent. Demonstrating the reality of the former could not disprove orthodox generative grammar; there is no necessity for such a relation. Processors, including incrementally interpreting ones, can be quite independent of what we think of (say on semantic grounds) as “the” syntax of a language. As Berwick and Weinberg (1983) have pointed out, compilers frequently parse programming languages according to a “covering grammar” which is quite different from the grammar that appears in the reference manual. However, in the case of human language there is a price to pay in terms of parsimony of the theory, for to abandon the Strong Competence Hypothesis is to abandon the possibility of plasticity in development. The explanatory burden is merely shifted onto the theory of
acquisition, and hence (by arguments familiar from Chomsky 1968, passim) onto the theory of evolution.

A second way of escaping the paradox is to deny the validity of our intuition that incremental interpretation is available. This tactic has been widespread among experimental psychologists, who have tended, not surprisingly, to be more reluctant to abandon the attractions of the Strong Competence Hypothesis. The claim has been that the sensation of incremental interpretation is an artefact, brought about either by the extreme rapidity and predictive nature of syntactic processing itself, or by the way in which the context makes the message itself redundant, or by processes of semantic interpretation that are supposed to proceed entirely autonomously, without benefit of syntactic analysis. As a result, experiments claiming to investigate the mechanism that resolves local ambiguities during syntactic processing have neglected to control for the possibility that an important factor in this process may be incremental comprehension. This neglect is the more surprising in view of the fact that a few studies have in fact demonstrated such effects. These include Carroll et al. 1978, Tanenhaus 1978, Marslen-Wilson et al. 1978, and the experiments considered in sections 3 and 4 below.

It is less often remarked that there is a third and potentially less disadvantageous way out of the paradox: to retain the Strong Competence Hypothesis, and to continue to respect our intuitions concerning incremental semantics, but to reject the standard theories of grammar. One group who have advocated a move of this kind are the Lexical-Functional Grammarians (Bresnan 1982—cf. Steedman 1985b), who coined the term "competence hypothesis" and who claim that the unification component of their theory is capable of incremental semantics. Another theory of grammar that is claimed to be directly compatible with incrementation is put forward in Ades and Steedman 1982 and in Steedman 1985a, 1987a, and 1988. The theory consists in an augmentation or extension of the Categorial Grammars of Ajdukiewicz (1935), by including rules of grammar corresponding to the "combinators" used by Curry and Feys (1958) to define the foundations of all applicative systems, including the $\lambda$ calculus.

Combinatory Grammar fragments have been developed for a number of notoriously problematic constructions involving single and multiple unbounded dependencies and coordinate constructions in more than one language. One interesting consequence of the theory lies in its radical implications for the concept of surface structure (Steedman 1985a, pp. 538–540), which it implies can be analyzed as predominantly left-branching. It also allows an appropriate semantic interpretation to be
compositionally assigned to each of the nonstandard constituents that result. It is thus directly compatible with incremental comprehension under the Strong Competence Hypothesis.

Section 1 of the present chapter sketches the combinatory theory of grammar, concentrating on its relation to incrementally interpretative processing and referring the reader elsewhere for detailed linguistic argumentation and language-specific details. This is followed by a review of an experimental paradigm that has been claimed to support the view that human sentence processing is "knowledge-rich" and that it draws on the results of incremental comprehension to resolve local ambiguities during parsing, in contrast to the "knowledge-free" structural techniques that have usually been proposed. This paradigm is then used to illustrate in some detail the various possible architectures for incremental and interactive sentence processors. The chapter concludes that there is really only one such architecture that is theoretically, empirically, and computationally reasonable. This architecture is of a kind called "parallel weakly interactive," and it conforms strictly to the modularity hypothesis of Fodor (1983).

1 Combinatory Grammars

Applicative systems define the notion of the application of a function to an argument and the notion of the abstraction, or definition, of a function. The idea of a function is akin to the everyday notion of a "concept," and natural-language semantics could hardly be any less than a system in which concepts can be applied and new concepts defined. Many natural-language constructions are strongly reminiscent of abstractions. For example, the relation of the "topicalized" sentence (a) in (5) to the canonical sentence (b) is both semantically and structurally similar (apart from the order of function and argument) to the application of a λ expression to an argument in (c) to yield (d):

(5) a. Harry, I know you like!
   b. I know you like Harry
   c. λx[know'(like' x you')]I' Harry
   d. know'(like' Harry' you') I'

However, the absence in (a) of any explicit linguistic realization of a variable binding operator like λ or a bound variable like x in (c) makes it interesting to ask whether some less familiar variety of applicative system might bear a more direct relation to natural language than the λ calculus. Such a system offers the interesting possibility that we might thereby explain the
notorious “constraints” on the unbounded dependencies or “extractions” illustrated by sentence (a) above, and the related idiosyncracies of coordinate constructions. The interest of Curry and Fey’s (1958) work in this connection is that their combinators allow the definition of applicative systems using operations that are entirely local and operate on adjacent entities, entirely without the use of bound variables.³

**Categorial Grammar**
Categorial grammars consist of two components. The first is a categorial lexicon, which associates each word of the language with at least one syntactic and semantic category. This category distinguishes between elements like verbs, which are syntactically and semantically functions, and elements like NPs and PPs, which are syntactically and semantically their arguments. The second component is a set of rules for combining functions and arguments, which are here called *combinatory rules* because of their close relation to Curry’s combinatory logic. In the “pure” categorial grammar of Bar Hillel (1953), this component was restricted to rules of functional application, and it made the grammar context-free, equivalent to more familiar phrase-structure grammars. Later versions, starting with those of Lambek (1958, 1961) and Geach (1972), have included more abstruse combinatory operations (see Steedman 1987a for a more detailed bibliography). Many of these extensions conform to the following limiting principle, as does the original operation of functional application:

(6) Principle of Adjacency:

Combinatory rules may only apply to entities that are linguistically realized and adjacent.

This principle embodies the central assumption of the present theory. It expressly excludes the postulation of “empty” categories, and it embodies a very strong form of localism. We are thereby already committed to some class of combinators. Unbounded operations, such as abstraction operators, movement, and coindexing, are excluded under the Principle of Adjacency.

**The Categorial Lexicon**
Some syntactic categories (such as nouns) which are naturally thought of as arguments, bear straightforward categories such as N. Others (like verbs) are naturally thought of as functors. In the present theory, functions that combine with arguments to their right bear a category of the form X/Y, denoting a rightward-combining function from category Y into category X. For example, determiners are NP/N and transitive verbs are VP/NP.
Other functions that combine with their arguments to the left are distinguished by the use of a backward slash, and bear categories of the form X\Y, denoting a leftward-combining function from Y into X. For example, VP-adverbia! phrases, such as quickly, bear the category VP\VP, and predicate phrases, such as arrived, bear the category S\NP.

Both types of function may of course have more than one argument, and may mix the two types of slashes, combining with different arguments in different directions. However, all function categories are unary or "curried." For example, the ditransitive verb give will bear the category (VP/NP)/NP—a (rightward-combining) function from (indirect) objects into (rightward-combining) functions from (direct) objects into VPs. This restriction has no great significance. Unary n-th-order curried functions are equivalent to n-ary first-order functions, as was first noted by Schönfinkel (1924). Where convenient, I shall assume this equivalence without comment, referring to a function like (VP/NP)/NP as "binary" and to VP as its "range."

The categories of all expressions, including the lexical categories, conform to the following principle, which embodies an assumption of the strongest possible "type-driven" relationship between syntax and semantics (cf. Klein and Sag 1985):

(3) Principle of Transparency:

The information in the syntactic type of an expression includes the information in its semantic type.

In other words, if an expression bears the syntactic category of a function from objects of syntactic type α into those of type β, then it is also semantically a function over the corresponding semantic types. The syntactic type will also of course determine a number of additional factors, such as linear order and agreement, which are not represented semantically. Such categories can be represented by a single data structure uniting syntactic type and semantic interpretation in unification-based implementations like those of Zeevat, Klein, and Calder (1986), Pareschi and Steedman (1987), and Pareschi (in preparation). (See Karttunen 1986, Uszkoreit 1986, Shieber 1986, and Wittenburg 1986 for related approaches.) However, the principle embodies some strong assumptions about the nature of semantic representations in the present grammar. For example, if we wanted to follow Montague in accounting for ambiguities of quantifier scope by changing the type of NP arguments into functions, then we would probably have to do it at some other level of semantic representation. (See Kang 1988 for an interesting discussion of this point.)
The combinatory rules govern the combination of function categories with other adjacent categories to yield new categories conforming to the above principle. The simplest such rules are the ones that apply a function to an argument.

**Functional Application**

Because of the assumption enshrined above in the Principle of Transparency, we can write the syntactic and semantic combinatory rules as one, associating each syntactic category in a rule with the semantic interpretation that it transparently reflects. Indeed, the only point of distinguishing the two at all is to make explicit the relation that the interpretation of the result bears to that of the inputs. The following obvious rules are required:

(8) a. \( X/Y:F \ Y:y \Rightarrow X:Fy \ (>) \)

b. \( Y:y \ X/Y:F \Rightarrow X:Fy \ (<) \)

In this and the other combinatory rules that follow, \( X \) and \( Y \) are variables that range over any category, including functions, so \( X/Y \) is any rightward-combining function and \( X\backslash Y \) is any leftward-combining function. Uppercase \( F \), \( G \), etc. are used for the interpretations of functions; lower-case \( x \), \( y \), etc. are used for the interpretations of arguments. The application of a function \( F \) to an argument \( x \) is represented by left-to-right order; e.g., \( Fx \). Semantic interpretations appear to the right of the syntactic category that identifies their type, separated by a colon.

The first of these rules, called Forward Application, allows rightward-combining functions such as transitive verbs to combine with arguments to their right. Its application in a derivation is indicated by underlining the operands and indexing the underline with the symbol \( > \). The second instance of the rule of functional application allows a leftward-combining function to combine with an argument to its left, and is indicated in derivations by an underline indexed by \( < \). Both rules determine the interpretation of the result, whose type is written underneath. It seems natural to assume that tensed verb phrases bear the category \( S\backslash NP \), so that tensed transitive verbs such as *eat* are \( (S\backslash NP)/NP \), ditransitives are \( ((S\backslash NP)/NP)/NP \), and so on, giving rise to derivations like the following:

(9) Harry eats apples

\[
\begin{array}{l}
\text{NP} \quad (S\backslash NP)/NP \quad \text{NP} \\
\text{---------------------------} \\
\text{S\backslash NP} \\
\text{---------------------------} < \\
\text{S}
\end{array}
\]
This derivation assigns an interpretation of type S, which we might write as eat’ apples’ harry’, where functional application "associates to the left," so that the result is equivalent to ((eat’ apples’) harry’). It is the interpretation of the verb, eat’, that determines the grammatical relations of the first argument (harry’) and the second (apples’) as subject and object, respectively.6

The above two rules are the only two rules of functional application that the theory allows. In particular, application of a function to an argument is by definition subject to their left-to-right order being consistent with the directionality of the function, because that is what the slashes mean. Obvious though this restriction is, it will be useful to enshrine it under the title of the principle of Directional Consistency, as follows:

(10) Principle of Directional Consistency:

All syntactic combinatory rules must be consistent with the directionality of the principal function.

The "principal" function is the one whose range is the same as the range of the result. (Since there is only one function concerned in functional application, the adjective is redundant in the case at hand.)

The above principle is not a stipulation, for it could be shown to follow from the semantics of the metalanguage in which the grammar is couched. Informally, directionality is a property of the argument of a function. The direction of a slash on a particular argument of a function denotes the relative position of the entity with which it may combine. The consistent rules are limited by the categories themselves.

Combinators, Right Node-Raising, and Leftward Extraction

The two central problems for any theory of natural-language grammar are posed by "deletions" and "ex extractions" in coordinate and other constructions, exemplified in (11).

(11) a. [I know Harry will cook] and [I think Betty might eat] the mushrooms we picked in the dank meadows behind the Grange.

b. These mushrooms, I think Betty might eat.

Both constructions appear to separate elements like objects and verbs which belong together in semantics. Both may separate them by unbounded amounts, including clause boundaries. They therefore appear to force us to abandon simple assumptions like the Principle of Adjacency (3), or the assumption that rules of grammar should apply to constituents. According to the present theory, however, both of these phenomena can be explained
without abandoning either assumption, under some simple extensions to
the combinatory rules and a consequent extension of the concept of a
constituent to include entities corresponding to strings like *might eat* and
*I think Betty might eat*.

**Coordination**

Coordination can be accommodated in a categorial grammar using the
following syncategorematic schema, inherited from Dougherty (1970) via
Gazdar (1981), which can be paraphrased as “Conjoin like categories.”

For the present purposes, the semantics of the rule can be ignored, except
to note that it is obtained from that of the conjuncts by applying some
functional $\Phi$ to the notoriously problematic interpretation of the conjunc-
tion (here written as “&”) and the conjuncts themselves:

(12) Coordination:

$$X:F \text{ conj } X:G \Rightarrow X:\Phi \& FG$$

Using such a rule schema, transitive verbs could coordinate as follows:

(13) I cooked and ate the beans

$$\text{NP (S/NP)/NP conj (S/NP)/NP NP}
\text{ coord}
\text{ (S/NP)/NP}
\text{ S/NP}
\Rightarrow
\text{ S}$$

**Functional Composition**

The following sentence will block in a pure categorial grammar:

(14) I will cook and might eat the mushrooms we picked

$$\text{NP (S/NP)/VP VP/NP conj (S/NP)/VP VP/NP NP}
\text{ Functional application will not help us here, but the earlier papers propose}
\text{ a comparably simple rule that will. It is the following (we will ignore the}
\text{ question of how the interpretations are related for the moment, and just}
\text{ treat it as a rule relating syntactic/semantic categorial types):}
\text{ Functional application will not help us here, but the earlier papers propose}
\text{ a comparably simple rule that will. It is the following (we will ignore the}
\text{ question of how the interpretations are related for the moment, and just}
\text{ treat it as a rule relating syntactic/semantic categorial types):}

(15) Forward Functional Composition

$$X/Y \text{ Y/Z } \Rightarrow X/Z$$

This rule, which has the appearance of the “cancellation” rule of fractional
multiplication, for reasons that will be apparent directly, will be indexed
“$> B$”. It allows the following derivation for the coordinate sentence (14):
The categories of the adjacent functors \( \text{will}_{(S\backslash NP)/VP} \) and \( \text{cook}_{VP/NP} \) match the rule, as do the parallel categories \( \text{might} \) and \( \text{eat} \). The result has the same type—\( (S\backslash NP)/NP \)—as the transitive verb in (9), so the rest of the derivation is the same as the earlier one.

Semantically, this rule is almost as simple as functional application. It is, in fact, functional composition. The combinator that composes two functions \( F \) and \( G \) was called \( B \) by Curry, and can be defined by the following equivalence:

\[
(17) \quad \text{BFG}x = F(Gx).
\]

A convention that application associates to the left is again followed, so that the left-hand side is equivalent to \((BF)G)x\). It follows that we can consider the application of \( B \) to \( F \) and \( G \) as producing a new function equivalent to abstracting on \( x \) in the above expression, thus:

\[
(18) \quad \text{BFG} = [x]F(Gx).
\]

(Curry's "bracket abstraction" notation "\([x] \langle \text{expression} \rangle\)" means much the same as the \( \lambda \) notation \( \lambda x \langle \text{expression} \rangle \). It is used here to remind us that the combinators are the primitives, not the abstraction operator.) We can therefore write the forward composition rule more completely as follows:

\[
(19) \quad \text{Forward Functional Composition} \quad X/Y:F \; Y/Z:G \Rightarrow X/Z:BFG
\]

The fact that semantics of the rule is functional composition ensures that the interpretation of the result will be the one we want. The fact that composition is an associative operation ensures that such will be the case for any order of composition of the component functions.

Repeated application of Forward Composition to the verb sequences in examples like (20) will allow coordination of indefinitely long strings of
verbs, on the assumption that each is a function over the result of the one to its right.

(20) She [may have seemed to have wanted to meet.]_{(S \backslash NP)/NP} but [actually turned out to dislike.]_{(S \backslash NP)/NP} the man you brought to the party at Harry’s

On the other hand, violations of the “across the board” condition on right node-raising and all other movement out of coordinate structures are still disallowed as a consequence of the composition mechanism and the assumption that coordination is an operation on adjacent constituents of like type. Thus, sentence 21 is not accepted, because \( (S \backslash NP)/NP \) and \( S \backslash NP \) are not of the same category.

(21) * I will cook and might eat beans potatoes

\[
\text{NP (} S \backslash \text{NP)/NP conj } S \backslash \text{NP } \text{NP} \]

From what general class of rules is the Forward Composition rule chosen? For the purpose of answering this question, it will be convenient to distinguish the two functions \( F \) and \( G \) in the above example as the “principal” and the “subsidiary” function, respectively. Like the rule of Functional Application (8), this rule is subject to the Principle of Directional Consistency (10): The subsidiary function must occur to whichever side is consistent with the slash on the principal function. The rule is also subject to a less obvious principle, which is claimed in Steedman 1987 to limit all combinatory rules in Universal Grammar that produce a function as their output, as follows:\(^{10}\):

(22) Principle of Directional Inheritance:

If the category that results from the application of a combinatory rule is a function category, then the slash defining directionality for a given argument in that category will be the same as the one defining directionality for the corresponding argument(s) in the input function(s).

(There is only one argument of the function that results from composition, and it is only inherited from one input function, so we can ignore the plural possibilities for the present purposes.)

The functional composition rule therefore potentially gives rise to four instances, distinguished by the left-to-right order and directionality of the principal and subsidiary functions, as follows:
(23) a. \( X/Y: F \ Y/Z: G \Rightarrow X/Z: BFG \) \( (> B) \)
    b. \( X/Y: F \ Y/Z: G \Rightarrow X/Z: BFG \) \( (> Bx) \)
    c. \( Y/Z: G \ X/Y: F \Rightarrow X/Z: BFG \) \( (< B) \)
    d. \( Y/Z: G \ X/Y: F \Rightarrow X/Z: BFG \) \( (< Bx) \)

As with all combinatory rules, natural languages are free to include rules on any of the four patterns, to restrict variables in any given rule to certain categories (such as S or maximal categories), or even to entirely exclude some of them. All four rules have been used to account for various phenomena in English.\(^{11}\)

**Type-Raising**

The following examples require more than functional composition alone:

(24) a. [I will cook] and [Betty may eat] the mushrooms we picked in the dismal glens above the Grange.
    b. [I think I will cook] and [you think that Betty may eat] the mushrooms etc.

The problem with examples like these is that the subject(s) cannot combine with the tensed verb(s) or the composed verb group(s), whose categories dictate that they have to combine with something else first:

(25) \[
\begin{align*}
\text{I will cook} & \ldots \\
\text{NP} & \quad (S/\text{NP})/VP \quad VP/NP \\
\text{---------} & \quad \Rightarrow B \\
\text{---------} & \quad (S/\text{NP})/NP \\
\end{align*}
\]

Functional composition alone does not help—composition is for combining *functions*, not arguments. However, there is an operation of “type-raising,” which is widely used in the Montague Grammar literature to map arguments (such as subjects) into functions over functions-that-take-such-arguments (such as predicates).\(^{12}\) Type raising is indexed \( C_\ast \), for reasons given below, and the instance of the rule that is relevant here is the following “forward” version:

(26) Subject Type-raising:
    \( \text{NP} \Rightarrow S/(S/\text{NP}) \) \( (> C_\ast) \)

It will permit the following derivation for (25):
The subject NP in the example raises into the category S/(S\NP). This category can, in turn, compose with the verb by the standard forward composition rule. Further iteration of composition and application completes the derivation. No other raised category for the subject NP will allow this or any other derivation. The more complex unbounded across-the-board right node-raised example (24b) is accepted in a parallel manner, since the embedded subjects can also raise under the rule, and repeated composition can again assemble two constituents of type S/NP. However, violations of the across-the-board condition are still not permitted, because the grammar does not yield categories of like type:

(28) *[I will cook]_{S/NP} and [Betty might eat potatoes]_{S}
            [the mushrooms ...]_{NP}

Even across-the-board extraction may not combine subject and object extraction, for the same reason:

(29) *[I will meet]_{S/NP} and [will marry Mary]_{S\NP} [your best friend]_{NP}

And of course, the following example is excluded, because only adjacent categories can coordinate:

(30) *[I will cook]_{S/NP} [the mushrooms]_{NP} and [Betty will eat]_{S/NP}

Like composition, the type-raising rules have a simple and invariant semantics. The semantics corresponds to another of Curry’s basic combinators, called \( C_* \), defined by the following equivalence:

(31) \( C_*x\ F = Fx \).

It follows that \( C_* \) applies to an argument creates the following abstraction over the function (again, Curry’s “bracket abstraction” notation is used):

(32) \( C_*x = [F]Fx \).
Type raising is here assumed to be subject in English to a "direction-preserving" property proposed by Dowty, such that arguments may only raise into rightward-looking functions over leftward-looking ones, or into leftward-looking functions over rightward ones. (I assume that this restriction would follow from a formal statement of the Principles of Inheritance and Consistency.) There are therefore just two possible direction-preserving type-raising rules, of which (26) is one special case, and which can be written as follows:

(33) Direction-preserving Type-raising

a. $X:x \Rightarrow \Sigma/(\Sigma/X):C_\ast x$ \quad (> C_\ast)

b. $X:x \Rightarrow \Sigma\backslash(\Sigma/X):C_\ast x$ \quad (< C_\ast)

The rule uses a polymorphic type variable $\Sigma$, and must be restricted, if only for the purposes of the parser. I shall assume for the present purpose that $X$ must be the sort of thing that is subcategorised for by a verb, and that the functions that $X$ raises over are (finite, nonfinite, participial, or whatever) *verbs*—that is, functions of any arity into (finite, nonfinite, participial, etc.) $S$. In derivations, the variable $\Sigma$ will be expanded to the category that instantiates it, as in (27).

**Nonstandard Constituent Coordination**

As is pointed out in Steedman 1985a and in Dowty 1988, Combinatory Grammars offer an account not only of coordinations of verb groups but also of the "nonconstituent" coordination of sequences of arguments, because arguments are allowed to type-raise into functions, and then to compose, as in the following example, adapted from Dowty:

(34)

\[
\begin{array}{ccccccc}
\text{give} & a \text{ dog} & a \text{ bone} & \text{and} & a \text{ policeman} & a \text{ flower} \\
\text{(VP/NP)/NP} & \text{(VP/NP)/((VP/NP)/NP)} & \text{conjunction} & \text{VP/((VP/NP)/NP)} & \text{VP/((VP/NP)/NP)} & \text{coord} \\
\text{VP/((VP/NP)/NP)} & \text{VP/((VP/NP)/NP)} & \text{VP/((VP/NP)/NP)} & \text{VP/((VP/NP)/NP)} & \text{VP/((VP/NP)/NP)} \\
\text{VP/((VP/NP)/NP)} & \text{VP/((VP/NP)/NP)} & \text{VP/((VP/NP)/NP)} & \text{VP/((VP/NP)/NP)} & \text{VP/((VP/NP)/NP)} \\
\end{array}
\]

While this derivation invokes the second instance, $< C_\ast$, of type-raising, and another instance, $< B$, of composition, the constraints of Adjacency, Consistency, and Inheritance will not permit rules that would allow the following examples to yield the same interpretation:
Dowty points out that a combinatorial grammar also allows such right node-raised nonstandard constituents to "strand" prepositions, just as standard constituent coordinates can, as in (36).

Dowty also points out that the acceptability of such strandings appears to be precisely parallel to island constraints on leftward extraction, as would be predicted on the present model, in which both rightward and leftward extraction depend on the possibility of assembling the residue into a single entity via the composition rule.

**Leftward Extraction**

The two combinatorial syntactic rules of functional composition and type raising provide almost all that we need in order to solve the second problem exemplified in example (11) namely that of leftward extraction in wh-movement constructions. Thus, in example 37 the subject NP can again raise over the predicate category, and iterated composition can again assemble the subject and all the verbs in the entire sequence *Harry must have been eating* to compose into a single function.
The important result is that the entire clause has been assembled into a single function adjacent to the extracted argument. Technically, this function still cannot combine, because the directionality of the slash forbids it. There are a number of ways of handling this detail in a manner consistent with the Principle of Adjacency, and we will pass over the matter here.\textsuperscript{14}

The rules of functional composition and type-raising provide a general mechanism for unbounded extraction. On the assumption that one category for the tensed verb \textit{believe} is (S/\NP)/S', and that the complementizer \textit{that} is S'/S, repeated application of the forward composition rule allows extractions across clause boundaries:

\begin{equation}
\text{(38) Apples, I believe that Harry eats}
\end{equation}

\[
\begin{array}{c}
\text{NP} & S/(S/\NP) & (S/\NP)/S' & S'/S & S/(S/\NP) & (S/\NP)/NP \\
\hline
S/S' & \rightarrow B \\
S/S & \rightarrow B \\
S/\NP & \rightarrow B \\
\end{array}
\]

The corresponding subject extractions can be excluded on the assumption that the slash-crossing instance \( \rightarrow Bx \) of forward composition is excluded from English, thus capturing the Fixed Subject Constraint or \textit{that}-trace filter of Bresnan (1972) and Chomsky and Lasnik (1977) and excluding sentences like (39) from English.

\begin{equation}
\text{(39) *Harry, I believe that eats apples.}
\end{equation}

Other parametrically related properties of English, in particular its fixed word order, depend upon the exclusion of this rule (Steedman 1987a). The possibility of exceptional subject extraction in examples like (40)

\begin{equation}
\text{(40) Harry, I believe eats apples}
\end{equation}

is captured by assigning a special lexical category to bare-complement verbs like \textit{believe} (Steedman 1987a, section 3.2.2). Once again, such details are passed over here.
At this point, the question naturally arises whether any other combinatorial operations than these two are implicated in natural-language grammars. The construction in (41) is of a type that Taraldsen (1979) and Engdahl (1981, 1983) have talked of as including a "parasitic" gap or empty category.

(41) (articles) which I will file \_ without reading \_p

The important properties of the sentence are that it has more than one gap corresponding to a single extracted item (which articles) and that one of these gaps (indicated by subscript p) is in a position from which extraction would not normally be allowed. In Steedman 1987a it is proposed to accommodate this construction by including rules corresponding to one more of Curry's combinators, the one he called S. The possible rules are again correctly limited by the principles of Consistency and Inheritance. The implications of such rules for the grammar of English are explored more fully there.

2 Syntactic Processing and Incremental Interpretation in Combinatory Grammar

The introduction of functional composition and other combinatorial rules into the grammar has radical implications for the concept of surface structure, and for syntactic processing. Most important, in return for a simple account of coordination, we are forced to conclude that the surface syntax of natural sentences is much more ambiguous than under traditional accounts. It will be recalled that many strings that in classical terms would not be regarded as constituents have that status in the present grammar. For example, the unbounded extraction in example 38 implies that the surface structure of the sentence "Those cakes I can believe that she will eat" includes constituents corresponding to the substrings I can, I can believe, I can believe that, I can believe that she, I can believe that she will, and I can believe that she will eat. In fact, since there are other possible sequences of application and composition that will accept the sentence, the theory implies that such sequences as can believe that she will eat, believe that she will eat, that she will eat, she will eat, and will eat may also on occasion be constituents. Since these constituents are defined in the grammar, it necessarily follows that the surface structure of the canonical "I can believe she will eat those cakes" may also include them, so that (42) represents only one of several possible surface-structure alternatives to the orthodox right-branching tree:
The proliferation of possible analyses that is induced by the inclusion of function composition seems at first glance to have disastrous implications for processing efficiency, because it introduces an explosion of what Wittenburg (1986) has called "spurious" ambiguities—that is, of alternative surface analyses that do not differ in interpretation. I will pass over this problem, noting that Pareschi (forthcoming) and Pareschi and Steedman (1987) argue that an efficient solution to this problem is offered by two properties of combinatory categorial grammar. The properties in question are the associativity of functional composition and the procedural neutrality of combinatory rules. The first property means that, as already noted, all these derivations yield identical results. The second property means that the constituents of any derivation can be directly recovered from the interpretation that results from any other derivation in the same equivalence class of "spuriously ambiguous" derivations. These properties suggest that unification-based parsers can thereby cope with both "spurious" and genuine attachment ambiguities with a uniform chart-based apparatus with no significant additional overheads to those engendered by more traditional grammars.

It is more important for present purposes to note that the effect of the Functional Composition rule has been to convert the right-branching structure that would result from simple functional application of the categories in the above example into a left-branching structure. It was remarked in the introduction that left branching allows incremental interpretation of sentences by left-to-right rule-to-rule processors under the Strong Competence Hypothesis. In the example, such a processor could, as it encountered each word of the sentence, build a single constituent corresponding...
to the prior string up to that point. And since the composition rule corresponds to semantic as well as syntactic composition, each of these constituents can immediately be interpreted. Indeed, as the earlier papers point out, there is no reason for any autonomous syntactic representation, as distinct from the interpretation itself, to be built at all. And if such fragments can be interpreted, then the results of evaluating them with respect to the context can be used to resolve local syntactic ambiguities. Thus, combinatory grammars seem to offer to formalism for natural-language grammar that is directly compatible with such processing, under the Strong Competence Hypothesis, without the addition of any extra apparatus.

Adopting the weak version of the interaction hypothesis, in turn, suggests a variety of processing strategies that will reduce the proliferation of semantically equivalent analyses induced by the combinatory rules. For example, in an implementation of the present grammars as "shift and reduce" parsers (Ades and Steedman 1982), a (nondeterministic) "reduce first" strategy will tend to produce left-branching derivations of the kind illustrated above. Such derivations have the property that, at any given point in the left-to-right pass of the processor, an interpretation for each analysis for the entire prior string is produced, which can then be evaluated with respect to the context (Steedman 1985a, p. 539). Moreover, this reduce-first strategy, by choosing the left-branching analyses that are permitted by the combinatory grammar over the right-branching ones that are proposed under traditional accounts of surface grammar, has the further desirable property of minimizing the depth of the shift-reduce processor's stack, thus countering Kimball's (1973) and Frazier and Fodor's (1978) objections to bottom-up parsing. Such strategies are under investigation by Pareschi (forthcoming) and Haddock (1986, 1987, forthcoming).

3 Evidence for Incremental Semantics and Interactive Processing

There is a well-established psycholinguistic tradition of accounting for the resolution of local syntactic ambiguities by purely structural criteria, rather than by any appeal to semantics. Much of this work can be summarized as having identified two main families of structural criteria. The first is a criterion that amounts to choosing the analysis that produces fewest nodes. For example, the VP see the boy with a telescope has (given some assumptions about constituent structure) the following two analyses, because of the ambiguity of the PP with a telescope between a VP argument and a Chomsky-adjointed NP modifier:
Under a criterion that has been most succinctly defined by Frazier (1978) under the name “Minimal Attachment,” analysis (a) is supposedly preferred to (b) by the human sentence-processing mechanism, because it has one fewer node.\(^{16}\)

The second strategy applies to the case where a category such as a VP modifier could attach to more than one VP in the sentence structure, as in the ambiguous VP *say Harry died last Saturday*, where the unambiguous adverbial modifier *last Saturday* could modify the saying or the dying:

The two analyses do not differ in complexity, but there is general agreement that the preferred reading is for the dying to have occurred last Saturday—that is, for attachment as low down and far to the right of the structure as possible, an observation variously enshrined in strategies of “right association” (Kimball 1973), “late closure” (Frazier 1978), and “final arguments” (Ford et al. 1982).

The unacceptable effect of sentences like

(45) The horse raced past the barn fell

is commonly attributed to purely structural properties of the two structures that are locally possible at the word *raced*, arising from its ambiguity
between a transitive verb and a past-participial NP modifier, coupled with a limit on human ability to recover from the ensuing “garden path” (Bever 1970; Kimball 1973; Frazier 1978). The preference for the garden-path-inducing VP reading of the substring *raced past the barn* over the non-garden-pathing participial-phrase reading is argued by these authors to arise from one or other of the “Minimal Attachment” strategies.

It has always been known that semantics can on occasion override such effects. Bever himself noted that the garden-path effect in (46) could be overridden by sense-semantic effects arising from the involvement of different kinds of arguments, as in (47):

(46) The authors read in the garden stank
(47) The articles read in the garden stank

It is important to be clear from the start that this purely sense-semantic effect turns out to be comparatively weak; Crain (1980), reported in Crain and Steedman 1985, found a rather slight effect of lexical materials, and Rayner, Carlson, and Frazier (1983) found no effect.

However, the weakness of the sense-semantic effect should not be taken to imply a similar ineffectiveness for all effects of meaning and context upon sentence processing. Crain and Steedman (1985) pointed out that complex NPs differ from simple ones not only in structural complexity but also in presuppositional complexity—that is, in the assumptions that they embody concerning the entities that are established in the context of utterance, and by virtue of which the NPs actually refer. In particular, postmodifiers (such as participial phrases, prepositional phrases, and restrictive relative clauses) all presuppose that there is a set of entities of the type denoted by the head noun, which the modifier restricts.

A second experiment (Crain 1980, reported in Crain and Steedman 1985) will help to make the point. Consider the following pair of sentences:

(48) a. Complement Target Sentence:
The psychologist told the wife that he was having trouble with her husband.

b. Modifier Target Sentence:
The psychologist told the wife that he was having trouble with to leave her husband.

Each of these sentences contains a local ambiguity at the word *that* between an analysis of the following string as a complement clause, as in (a), or as a relative clause, as in (b). In the “neutral” or “null” context, readers resolve the ambiguity in favor of the complement analysis, so that version (b), which
demands the relative-clause analysis, engenders a garden-path effect. Frazier has argued that this garden path is a consequence of the Minimal Attachment strategy, just as the earlier horse raced examples are. However, the use of a restrictive relative clause modifier in (b) presupposes that there is more than one wife in the context of discourse, and that one of these wives is distinguished by being troublesome to the psychologist. By contrast, the absence of a modifier in (a) presupposes that just one wife is established in the context of discourse. Crain argued that if incremental comprehension were indeed characteristic of human processors, then a context supporting the presuppositions of either the simple or the complex NP analyses of the locally ambiguous substrings in the earlier examples could make the processor favor that analysis, overriding the classic garden-path effect or creating novel garden paths where there were none before. Moreover, he argued, the fact that the so-called null context supports the presuppositions of neither analysis did not mean that it would be neutral with respect to the two analyses.

To demonstrate that garden-path effects could be controlled by referential context, Crain preceded the presentation of “complement” and “modifier” target sentences like (48a) and (48b) above with contexts that supported the presuppositions of the respective analyses. For the “psychologist” sentences, such contexts were the following:

(49) a. Complement-supporting context:
    A psychologist was counseling a married couple. One member of
    the pair was fighting with him but the other one was nice to him.

b. Modifier-supporting context:
    A psychologist was counselling two married couples. One of the
    couples was fighting with him but the other one was nice to him.

These contexts were claimed by Crain to constitute a minimal pair, differing only in the number of participants, and in particular in the number of wives that could be inferred to be involved. The first of these contexts supports the presupposition of sentence 48a that there is one wife in the context, and therefore demands the complement-clause analysis. It fails to support the presuppositions of the relative sentence 48b that there are several wives under discussion. If the classic garden-path effect is under the control of context, then that effect should be unaffected or even exacerbated. By contrast, the second of these contexts supports the presuppositions of the relative-clause sentence 48b and denies those of the complement-clause target 48a. On the same assumption, this second context should therefore reduce or even eliminate the classic garden-path effect. The two types of
contexts illustrated in (49) can be crossed with the corresponding pairs of target sentences illustrated in (48) to yield four conditions: complement-supporting context, complement target (CC); complement-supporting context, modifier target (CM); modifier-supporting context, modifier target (MM); and modifier-supporting, complement target (MC). The prediction is that the “crossed” conditions CM and MC should engender garden paths, whereas the “uncrossed” conditions CC and MM should not. Although some details of Crain’s actual experiment have been criticized by Altmann, a number of experimental studies on a pattern closely related to the above (Crain 1980; Crain and Steedman 1985; Altmann 1986, 1987; Altmann and Steedman 1988), using either grammaticality judgments (Crain), global reading times, or subject-paced reading times (Altmann), show the effect of context on local syntactic attachment ambiguities of the kind whose processing characteristics have been claimed to support Minimal Attachment and related structural strategies. Crain’s and Altmann’s experiments have used a wide variety of presuppositionally loaded grammatical parameters and constructions, including definiteness (Crain), participial modifiers (Crain), restrictive relatives (Crain, Altmann) and prepositional-phrase post-modifiers (Altmann).

Ferreira and Clifton (1986) have claimed that Crain’s result does not appear when a measure more related to on-line processing, namely eye movements, is used. They claim that Crain’s and Altmann’s results arise from post hoc reanalysis, an argument which is criticized at length on theoretical grounds in section 4 below. The materials and methods of Ferreira’s study have been criticized by Altmann and Steedman (1988), on the grounds that their materials do not constitute minimal pairs of contexts and targets. An experiment controlling for these factors and using subject-paced phrasal reading time as an on-line measure is reported by Altmann and Steedman (1988, expt. II). This experiment replicates the earlier results in all relevant respects, and shows that there is a substantial effect of context on reading time within the critical phrase. Indeed, this experiment suggests that, once the materials and the context are properly controlled, any residual effect is in the direction opposite to that predicted by the minimal-attachment hypothesis.

Crain and Steedman proposed a mechanism for such context-based decisions of local syntactic ambiguities that invoked a “Principle of Referential Success,” according to which successfully referring analyses were to be selected in favor of unsuccessful ones. Altmann (1986, p. 201) and Altmann and Steedman (1988) have reformulated this principle in terms of partial NP analyses and the failure of their presuppositions in an incre-
mental constraint-satisfaction model suggested by Haddock (1987). The principle can be defined informally as follows:

(50) Principle of Referential Support:
A (possibly incomplete) NP analysis whose referential presuppositions are supported by the context will be favored over one whose referential presuppositions are not supported by the context.

We will return to the important question of how such processors might actually be organized, and to what architectures are possible or sensible for them. It is more important for the moment to note that this proposal also has implications for the processing of such sentences in the so-called null context, typical of the psycholinguistic sentence-processing experiment, in which no analysis refers. We have been very vague so far about what exactly we mean by saying that a referent or set of referents is present in the context. But it clearly means much more than mere prior presence or representation of the entities in question in some model which we may think of as some variety of propositional database in the hearer's mind, to which expressions refer anaphorically. We often encounter definite referring expressions whose referents are not previously established in the recipients mind—for example, in fictions such as novels, or in intensional contexts as in (51).

(51) Did you see the man who just walked past the window?
Such definites refer to entities which the speaker regards as established. But for the hearer they may or may not be so. If they are not, then the hearer must introduce appropriate representations in their database. If a referring expression not only denotes a novel entity but also carries additional presuppositions which are also not yet satisfied, then the corresponding modifications must also be made.

Arguments for the Parallel Weak Interaction

Strong and Weak Interaction

According to the strong interaction hypothesis, semantics and context can "prescribe" specific courses of action to syntactic processing, actively restricting the search space within which it operates—say, by affecting the order in which rules of grammar are to be tried, or even by entirely excluding some of them.

The weak version of the interaction hypothesis is much more restrictive. According to this version, syntax autonomously "proposes" analyses, whereas semantics and context merely "dispose" among the alternatives
that it offers. The only interaction that is allowed is that interpretative processes may deliver judgment on the contextual appropriateness of the alternatives that are proposed by syntax, causing it to abandon some and continue with others. Some authors have since preferred to talk of "instructive" versus "selective" interaction rather than strong versus weak, but I will continue to use the latter terms.

Crain and Steedman (1985) argue that the only coherent form of interaction is the weak or selective one, according to which the results of evaluation can suspend a line of analysis on the grounds that its interpretation is contextually inappropriate but cannot predispose syntactic processing toward any particular construction, a proposal which is tentatively endorsed by Fodor (1983, p. 78n). This is the mechanism that we were tacitly assuming when arguing that the presence in a hearer's mind of several potential referents (say several wives in Crain's example) will cause a simple NP analysis (e.g. "the wife") to be rejected in favor of a complex NP analysis ("the wife that he was having trouble with"), whereas other contexts that do not support the presuppositions of restrictive adjuncts (including the so-called "null context") will support the simple NP analysis.

It will be convenient to refer to the two uses of definite referring expressions contrasted above as "given" and "new," respectively. In the traditional psycholinguistic experiment, in which sentences are presented out of context, definite expressions are by definition of the new variety. However, they still carry their presuppositions, which must be supported or accommodated by the hearer. Crain and Steedman (1985) argued that in a case of local ambiguity of this kind, it was reasonable to assume that the reading carrying fewest presuppositions, and therefore requiring fewest modifications to the database, would be favored under a "Principle of Parsimony":

(52) Principle of Parsimony:

A reading that carries fewer unsatisfied presuppositions will be favored over one that carries more.

It should be obvious that the above principle subsumes the earlier criterion of Referential Support as a special case: A reading that fails to refer by definition carries more unsatisfied referential presuppositions than one that does refer, since the latter carries no such unsatisfied presuppositions.

4 Architectures for Incrementally Interpreting Interactive Processors

We can identify a number of parameters that determine possible architectures for interactive processors of the kind that are implied by these
experimental accounts. The first parameter concerns the nature of the interaction itself. Crain and Steedman distinguish two versions of the interactive hypothesis, which they call the “strong” and “weak” versions. Weakly interactive processors may be further divided into “parallel” and “serial” versions. A third parameter concerns the “intimacy” (or fineness of grain) of the interaction, that is, the question of whether the comprehension process can only adjudicate over large units, such as clauses, or whether appeal can be made at each successive word. This parameter has a rather different character than its predecessors. In a processor that is consistent with the Strong Competence criterion, it is a property of the grammar rather than the processor per se. A fourth parameter concerns the “top-down” or “bottom-up” direction of application of the rules of the grammar. Any advantages also depend on the particular grammars that are involved.

An early proposal for incremental semantics and the weak interaction in processing was made by Winograd (1972), who proposed to allow reference to the context or the database to decide ambiguities in NP analyses in examples like (53).

(53) Put the block in the box on the table.

Winograd’s actual architecture was serially weakly interactive. That is, syntax proposed a single analysis for the NP (actually, the program started with the longest alternative, the block in the box. Only if that failed to refer—perhaps because there was no block in the box—would the processor backtrack for another analysis.)

This proposal appears to have influenced the work of Bobrow and Webber (1980, 1981) on the RUS parser, although the weak interactions that they allow from semantics are strictly sense-semantic and do not involve reference at all. More specifically, they allow information about the sense-semantic type of arguments such as NPs and PPs, and a rich form of case-frame/subcategorization information associated with verbs, to interact to disambiguate verb senses and grammatical relations in an incremental fashion. However, reference plays no part in this model, although some of the problems that Bobrow and Webber propose to handle purely on the basis of sense-semantics seem to be likely to require reference.

The program of Bobrow and Webber uses sense-semantics in the form of a rich set of semantic argument types and case frames—for example, to rule out a reading of the door as an agent in ergative sentences like (54).

(54) The door opened.

However (Bobrow and Webber 1980), such strategies will tend to yield no reading at all for sets of sentences like (55).
(55) a. I am wearing my birthday present.
    b. I am driving my birthday present.

("Apparel-NP" and "vehicle-NP" are among their sense-semantic categories subcategorized for by verbal case-frames.) They offer a generalization based on a hierarchically structured type system for the criterion of sense-semantic compatibility in order to handle this problem (as well as a number of other cases, including pronominals and NPs like *the thing* whose sense-semantics is underspecified). But they also point out that something more is needed. The type system will still not handle certain cases of metonymy that they mention, as when a customer in a restaurant is referred to as "the hamburger". A specific domain of reference seems to be inextricably involved in such cases.

Crain and Steedman (1985) argue on metatheoretical grounds against the alternative "strong" interaction hypothesis, according to which the referential context might predispose the processor toward certain constructions. However, they note (p. 326) that, while some versions of the strong interaction hypothesis are empirically distinguishable from the weak variety, some are not. A version that says that the presence in a hearer’s mind of several wives predisposes them toward complex NP analyses in general—not just *the wife that he was having trouble with* but also *the horse raced past the barn*—certainly makes a strong prediction, although it is so absurd as to be hardly worthy of experimental investigation. But a version that says that on encountering the word *wife* the presence of several referent wives "switches on" the complex NP analysis and switches off the simple one could probably not be distinguished experimentally from the alternative weak hypothesis, according to which the analyses would be developed first and then adjudicated by appeal to the context. The arguments against this version of the strong hypothesis rest on its theoretical complexity, and its probable computational costliness, in comparison with the weak interaction. As Fodor has pointed out, there seem to be just too many ways to say what we mean for this effect to be very helpful, and it is noticeable that the strong interaction, though frequently advocated, has rarely if ever been implemented as a program.

**Serial vs. Parallel Interaction**

Once one is committed to the weak interaction, it is clear that there is a second parameter that must be fixed in specifying an interactive architecture, namely whether the proposal of alternative analyses by the syntactic component for disposal by the comprehension process occurs serially, one by one, or in parallel.
If the alternative analyses of ambiguous substrings were to be proposed in series, then a weakly interactive parser would have to specify the order on other than contextual grounds—say, by rule-ordering. If the simple analysis were to be ordered before the complex, then we would have something closely related to a parser embodying Minimal Attachment in the form of rule ordering, such as the ATN-based processor proposed by Wanner (1981). The only difference would be that the present theory would allow interpretation to precipitate reanalysis in the “two-wife” context, as opposed to the “one-wife” context, under the criterion of Referential Failure, thus predicting Crain’s and Altmann’s experimental results. In particular, we might still expect the increase in processing load for the nonminimal analysis predicted by the Minimal Attachment strategy, despite the avoidance of the garden path itself.

However, when it comes to the null context, for which the rival NP analyses are new rather than given and the ambiguity is resolved under the more general criterion of parsimony, it is clear that the mechanism must be parallel rather than serial. We cannot reject an innovating noun phrase merely because it carries a great many previously unsatisfied presuppositions, for it may be the only possible analysis. We can only reject it in comparison with some other analysis carrying fewer unsatisfied presuppositions. In general, the same is true for the criterion of referential failure, since the alternative to a referentially failing “given” interpretation is always the “new” analysis. It follows that the mechanism implicit in the earlier informal account must be one in which syntax offers alternatives in parallel.

There is further evidence to suggest that alternative analyses must be developed in parallel by human sentence processors. This evidence is quite independent of whether these alternatives are disposed of by semantics or not. Consider the alternatives that a processor must consider in dealing with the following set of sentences:

(56) a. Who do you want _?
   b. Who do you want to have _?
   c. Who do you want _ to have dinner?
   d. Who do you want to have dinner with _?
   e. Who do you want _ to have dinner with the president?
   f. Who do you want to have dinner with the president on behalf of _?
g. Who do you want to have dinner with the president on behalf of your government?

h. Who do you want to have dinner with the president on behalf of your government for ...

Sentences b, d, f, and h are "equi NP-deleted"; a, c, e, and g are not. Thus there is a local ambiguity in all members of the set after the word want. If either the equi or the non-equi analysis were consistently chosen first, say by ordering the respective lexical entries (Ford, Bresnan, and Kaplan 1982—see Steedman 1985b, p. 378, for a critique of this proposal), or by a "most recent filler" strategy (Frazier, Clifton, and Randall 1983), then we would expect the sentences in the other set to show an increased processing load. What is more, we might expect this increased load to grow larger as the sentences grew longer. Intuitively this seems unlikely; there is no obvious difference in difficulty between the two sets, even for the longer examples. This kind of lexical ambiguity is rather different from the attachment ambiguities considered above. The two analyses just seem to be maintained in parallel indefinitely, and neither leads to a garden path in the null context.\footnote{22}

Frazier, Clifton, and Randall (1983), who took the serial non-interactive proposal to extreme lengths, claimed to find experimental evidence in support of a two-stage parser in which all long-range dependencies, whether lexically determined (as in sentences with equi and control verbs) or not (as in the case of wh-dependencies), are determined after structural analysis is complete. They claimed that even when a verb does not permit an equi analysis (as in the case of force), the first, structure-building pass of the parser will construct an equi analysis with an empty category, under a strategy which they call the "most recent filler" strategy. That is, in a sentence like

(57) Everyone liked the woman who the little child forced to sing those stupid French songs last Christmas

the processor will incorrectly build a structure including

(58) \ldots the child$_1$ forced [PRO$_1$ to sing \ldots].

By contrast, when faced with

(59) Everyone liked the woman who the little child started to sing those stupid French songs for last Christmas

the processor will correctly build a structure including

(60) \ldots the child$_1$ started [PRO$_1$ to sing \ldots].
It is only at the later stage of associating fillers (e.g. subjects and relative pronouns) with gaps (e.g. empty categories) that the processor will access the lexical information that force does not permit this construction, forcing the first stage to restart.

We noted earlier that this proposal is surprising, and that it does not conform to our intuitions as to what goes on when we process such sentences. Nevertheless, it would constitute a very strong argument against the parallel model proposed here, were it robust. However, Crain and Fodor (1985) have pointed to a number of problems with the materials and method of this experiment. In an experiment with revised materials and a self-paced reading-time measure (as opposed to Frazier et al.’s measure of latency between the end of the sentence and a “got it” comprehension response), there was no evidence of increased processing time due to revision of anomalous equi analyses for unambiguously non-equi verbs such as force.

This result appears to contradict Frazier’s two-stage parser. Crain and Fodor did find some evidence of a processing advantage of the equi analysis for ambiguous verbs such as beg, and admit the possibility of serial processing with a “most recent filler” strategy of some other kind. However, in the absence of an actual garden-path effect, or at least of some increase in the overhead when disambiguation points occur very late after the disambiguation point, the processor proposed here, using parallel processing with structure sharing, is not excluded. The mere fact that there is a preference for the equi analysis in the null context tells us nothing.

**Intimacy of Evaluation in Incremental Semantics and Interactive Processing**

A further parameter of incremental semantics and the architecture of a (serial or parallel) weakly interactive processor is the “intimacy” of the interaction, or the frequency with which syntax can appeal to interpretation for guidance. At one extreme, one might believe (as certain early psycholinguists who managed the considerable mental feat of believing that transformations corresponded to operations of the processor had to), that such interactions apply only to complete clauses or even sentences. Or one might believe that smaller units, such as noun phrases, can be evaluated as soon as they are complete, without waiting for the rest of the matrix clause. Such more intimate incremental semantics and interactive processing was a feature of Winograd’s (1972) program. At the other extreme, one might believe that even more intimacy is possible. If, as our intuitions suggest, the most incomplete fragments of constituents can be interpreted, then an
interactive parser could appeal very frequently indeed—perhaps as each new word is processed.

However, there is an important restriction on this aspect of the processor. If a rule-to-rule grammar and processor are to conform to the Strong Competence Hypothesis, then only constituents, as defined by the grammar, can receive an interpretation, or be evaluated, or be used to guide interactive processing. It is this corollary of the Strong Competence Hypothesis that makes the predominance in natural languages of right-branching constructions so surprising. But if one adheres to that hypothesis, then the limit on intimacy of incrementation and interaction is simply a question of grammar.

Top-Down vs. Bottom-up
A further parameter governing the architecture of any language processor, whether incremental and interactive or not, concerns the direction of application of the rules of the grammar. Does the processor work top-down and predictively through the rules of the grammar, starting from the distinguished symbol S, or does it work bottom-up, starting from the words in the string? There are potential advantages to both tactics, and it turns out that this parameter also depends on the particular grammar that the processor embodies.23

There is a considerable intuitive appeal to the bottom-up strategy. Data-driven perceptual models always have the psychological edge over top-down models, because they tie the mechanism to the properties of the domain, with obvious desirable consequences for theories of development and evolution. In the syntactic domain, part of their appeal also arises from the fact that almost any well-formed constituent—as well as many strings which are apparently non-constituents—counts as a complete utterance, not just complete sentences. Although nothing about this phenomenon is incompatible with top-down processing,24 it is natural to think of this property as arising from a processor that starts from the words and builds whatever it can. A further intuitive appeal of such processors is that they appear more straightforwardly compatible with incremental semantic interpretation, allowing each successively larger constituent to be interpreted as soon as it is built.

A more practical advantage of bottom-up processors is that they are not necessarily vulnerable to the special problems which (left) recursive rules of grammar create for (left-to-right) top-down processors.25 Such rules, in the absence of special mechanisms (which, though simple enough, compromise the Strong Competence Hypothesis), cause infinite recursion. Nevertheless, they are widespread in natural languages.
However, bottom-up processors as models of human sentence comprehension have been criticized on a number of grounds. Kimball (1973) and Frazier and Fodor (1978) assert that they do not have the same "predictive" capacity as top-down processors, which allow the entire "left context," or the portion of the sentence to the left of the point in the string which the analysis has reached, to limit the search space for the remainder (but see Kimball 1975). They also make the predominance of right-branching structures puzzling for a second reason: Bottom-up processors such as those related to "shift-reduce" or LR parsers (Aho and Johnson 1974; Shieber 1983; Pereira 1985) require an indefinitely large stack for right-branching structures, whereas left-branching ones keep the stack depth constant.

These objections have been countered by Pereira (1985), who has also argued for bottom-up shift-reduce parsing, within a rather different framework than the one presented here. However, the combinatory grammars discussed in sections 1 and 2 of the present chapter provide a grammatical framework that avoids both problems, by embedding information equivalent to predictive "reachability tables" (Kay 1980) in the grammar itself and by inducing left-branching structures on what would, in pure categorial grammar, be right-branching sentences.

Modularity and Incremental Interactive Processing

Nothing in the proposal that human sentence processors are bottom-up, and weakly parallel-interactive, with as much intimacy as the grammar allows under the Strong Competence Hypothesis, conflicts in any way with the modularity hypothesis of Fodor (1983, pp. 78 and 135). Though an unusually high degree of parallel structure is here claimed to hold between syntax, processing, semantics, and even the inference system, with a consequent reduction of the theoretical burden upon innate specification, these components are all formally and computationally autonomous and domain-specific.

On a Supposed Alternative to Parallel Weak Interaction

The weakly interactive incremental model has received some support from the psychology and artificial intelligence communities (Bobrow and Webber 1980a,b; Carroll et al. 1978; Marslen-Wilson et al. 1978; Sanford and Garrod 1981; Steedman and Johnson-Laird 1978; Swinney 1979; Tanenhaus 1978; Winograd 1972). However, the dominant model of sentence processing in both the psychology and the AI literature has not been this one. Both groups have predominantly opted for a serial non-interactive architecture, driven by or embodying nonsemantic parsing "strategies" predominantly
drawn from the two families of "Minimal Attachment" and "Right Association" (Bever 1970; Kimball 1973; Fodor and Frazier 1978; Frazier 1978; Wanner 1980; Ford, Bresnan, and Kaplan 1982; Marcus 1980). The realizations of this model vary a great deal. Some are "one-stage" parsers; others operate in two stages, separating structure-building from assignment of dependencies (Frazier, Berwick). Some more or less explicitly include the strategies as rules (Frazier, Ford); others embody them directly in rule-orderings (Wanner) or less directly in other aspects of the parser (Marcus). However, for present purposes these distinctions are not as important as the fact that a number of crucial predictions arise from the comparison of these models with the one advocated here.

**Serial Non-Interactive Architecture**

By far the most commonly proposed alternative to the architecture we are proposing here has been the non-interactive or "autonomous" variety depending on parsing strategies or preferences (see the earlier references). Usually, the very existence of garden-path phenomena has been taken to indicate that, in cases of local ambiguity, analyses are proposed singly and in series, recovery from inappropriate analyses being on occasion impossible.

Of course, none of these parsers is entirely non-interactive. Everyone admits that, once syntactic analysis is complete and an interpretation can be obtained, the context may reveal an anomaly and precipitate a re-analysis. (This amounts to a very weak form of the weak interaction, with the clause or sentence as the unit.) It is often assumed (Ferreira and Clifton 1986; Frazier 1987) that Crain's and Altmann's results are compatible with such processors, since most of their results do not use on-line measures of processing difficulty, but only overall measures, such as reading times. (However, Altmann and Steedman [1988] replicated the earlier results with an on-line measure.) It seems not to be generally realized that, under any reasonable set of assumptions about how such a serially non-interactive processor could work, the proposal leads to a logical contradiction, since it immediately forces an evidently false prediction, as follows.

Consider the earlier examples of contextual control of garden paths. Crain showed that a context containing several wives would eliminate the classical garden-path effect of (61).

(61) The psychologist told the woman that he was having trouble with * to leave her husband.

The asterisk marks the first point at which the serially autonomous processor can detect an anomaly. Thanks to the new proposal, the processor
can restart syntactic analysis when it detects the syntactic anomaly and the mismatch between the context and the presuppositions of the simple NP that it has produced via Minimal Attachment. It can therefore avoid the garden path, by reanalyzing at a cost in processing that we might hope would show up on some appropriate measure. All seems well with the serially non-interactive theory.

Unfortunately, it isn’t. Without appealing to reanalysis, we wouldn’t be able to explain how context can prevent the garden path. But now it is hard to see why the sentence should ever garden-path. Consider the information the processor has available at the end of its first, minimally attaching, pass through the sentence. It has blocked, grammatically, and it can presumably tell that the context does not support the NP analysis that it has produced. But what kind of serial autonomous processor can recover to autonomously produce a syntactic analysis just in the case when the context will in fact turn out to be consistent with the alternative analysis, but fails in this purely syntactic process when the context will not in fact support it (for example, when it is the null context)? It is a very strange serial autonomous processor indeed, for it appears to have foresight about the contextual compatibility of an analysis which it has not yet built because it is serial. And it appears to be able to allow the success of its purely syntactic processes to be determined by semantic and referential facts to which it can have no access because it is autonomous.

The serial non-interactive model therefore runs into a paradox when faced with Crain and Altmann’s results showing that local syntactic-ambiguity resolution is under the control of context. There seem to be only two ways out of this paradox for those who wish to maintain the hypothesis of serial non-interactive processing. One is to assume that the effect on the serial autonomous processor of this kind of post hoc assessment of the context is to predictively affect the future operations of the processor when it restarts. However, such a proposal amounts to incorporating not only interaction, but the strong interaction. The difficulties of even seeing how this kind of interaction could work have already been noted, and presumably this is not what the proponents of serial autonomy intend. The only alternative appears to be to adopt the parallel weakly interactive theory, as being the only class of processor that is straightforwardly compatible with Crain’s and Altmann’s experimental results. Under the Strong Competence Hypothesis, the combinatory grammars discussed in sections 2 and 3 seem to be uniquely suited to this kind of processing.
Acknowledgments

This paper owes a lot to conversations and collaborations over a number of years with Gerry Altmann, Stephen Crain, Nick Haddock, Remo Pare-schi, and Henry Thompson. Section 4 is an expanded cousin to some portions of the introduction to Altmann and Steedman 1988. Thanks to William Marslen-Wilson, Henry Thompson, and Bonnie Lynn Webber for reading and commenting on the draft. Part of the work was supported under a grant from ESPRIT (project 393).

Notes

1. By “reference”, I mean something like “evaluation to the level of discourse entities,” or to Webber’s (1978) level 2 representations, rather than evaluation to the level of entities in the world or the model.

2. Mallinson and Blake (1981, p. 148) cite 2 VOS and 1 OVS languages in a sample of 100 languages. In the same sample there are 76 languages which are either SOV or SVO.

3. Some possible computational advantages of applicative systems that avoid the overheads of variable-binding via the use of combinators are discussed in Turner 1979a, and in Steedman 1987b.

4. The present theory follows Lambek and Steedman 1987a in this respect, and differs from the earlier versions in Ades and Steedman 1982 and Steedman 1985, which used nondirectional slashes, constraining order in the combinatory rules.

5. Some authors, including Lambek, use other conventions.

6. The last, subject, argument of the verb must be defined for plural or singular number by the inflection of the verb, and the subject argument must be compatible with this specification, to capture basic subject-verb agreement using an obvious unification-based mechanism of the kind proposed for this problem by Shieber (1986). See Steedman 1987a for further remarks on agreement, and Pareschi and Steedman for a discussion of unification-based combinatory grammars.

7. The notion of “like category” is of course problematic for any theory of syntax, as well-known examples like “Pat is a Republican and proud of it” (Sag et al. 1985) reveal. We shall not discuss such problems here, assuming that some finer-grained feature-based categorization of atomic categories like NP such as the one offered by them can be applied to the present theory.

8. It might appear more natural for a categorically based approach to eschew such syncategorematic rules, and drive coordination from the lexical category of the conjunct by associating the following categorial type with sentential conjunctions: and := (Σ \ Σ)/Σ:Φ&. However, such a category requires notationally tedious elaboration if overgeneration is to be prevented, so we will continue to use the syncategorematic rule as a convenient abbreviation.

9. Assuming a proper semantics for the conjunction rule.
10. Steedman (in press) argues that this principle follows from other principles of
the theory, rather than by stipulation.
11. Besides the forward rule (a), Dowty (1988) has used the backward rule (c) in his
account of English “non-constituent” coordination (see below). While Dowty sug-
ggested that “slash crossing” composition should be excluded, the slash-crossing
backward rule (d) is introduced in Steedman 1987a, and has also been proposed by
Moortgat (1985) and Morrill (1987) to account for right extraposition. A very
restricted version of the forward crossing rule (b) is proposed for English “gapping”
by Steedman (in press).
12. I do not intend to suggest that the present syntactic use of type raising is related
in any way to Montague’s (1973) account of quantifier scope phenomena. See
Steedman 1987a for remarks on subject-verb agreement and type-raised subject
category.
13. Different notations are used in Steedman 1985a and in Steedman 1987a, but
they all amount to the same thing. The category of NPs in Karttunen 1986 is related
to a type-raised category.
See Szabolcsi 1987 for extensions to “pied piped” wh-items.
15. However, the interpretation in question must be intensional rather than ex-
tensional, and it should be thought of as a structural object, related to a traditional
deep structure or “logical form.”
16. In Kimball’s (1973) account, the explanation is similar but arises under a
number of his “principles,” the most important being (early) “closure.”
17. The question of what exactly is meant by an “entity” here, and of what is meant
by its being “established in the context,” is one to which we return below.
18. The possibility that this relative is non-restrictive is presumably excluded by
the absence of punctuation, and by the fact that in most dialects that, unlike
who(m), cannot introduce nonrestrictives anyway.
19. The terms are from Halliday 1967, although they are used informally here.
Presumably the distinction is merely one example of the general tendency of
referring expressions to depend for their effect upon the intensional characteristics
of the context. Other related distinctions are referential and attributive definiteness
and de dicto/de re.
20. If it is the only analysis, a serial processor is going to have to either reanalyze
when it fails to find an alternative or save the analysis against the possibility of such
failure. Either strategy amounts to simulating parallelism.
21. “Empty categories,” indicated by _, are used (as usual) merely for expository
clarity.
22. Of course, in order to avoid redundancy in the analyses of the remainder of the
string, it follows that an efficient processor will have to resort to some trick such
as “structure sharing” between the two analyses.
23. This use of the parsing term-of-art “top-down” should not be confused with
the (unfortunately) widespread use of the phrase to describe what we call in this
chapter the “strong” or instructive interaction between high- and low-level processes such as semantics and syntax.

24. A top-down processor could simply define all the nodes in question as possible start symbols.

25. The word necessarily is important. Some ways of handling nondeterminism, such as chart parsing, may reintroduce the problem of left-recursion into a bottom-up parser.

References


