A THEORY OF WORD ORDER IN CATEGORIAL GRAMMAR WITH SPECIAL REFERENCE TO SPANISH

John C. Foster

D.Phil
University of York
Department of Language and Linguistic Science
March 1990

CONTENTS

CHAPTER 5: THE CATEGORIAL ANALYSIS OF SUBJECT AND OBJECT ORDER IN SPANISH

CHAPTER 6: THE STRUCTURE OF THE CATEGORIAL LEXICON

PART THREE:
EXTENDED CATEGORIAL GRAMMAR

CHAPTER 7: CATEGORY STRUCTURE AND COMBINATORS

ACKNOWLEDGEMENTS

This thesis has had a long gestation during which time I have had the assistance of many people. Not being a native speaker of Spanish, I have used informants for the data needed to test the formal theories presented here. Señor Ronsano of Stockton-on-Tees was my original source of information. I thank him for the pains he took to teach me something of his language. My first supervisor, Professor John N. Green, struggled through many sheets of the wierdest sentences always willing to offer his opinions. His readiness to respond to my enquiries and the help he gave me in forming my early judgements were all the more remarkable considering the obvious limitations of my command of Spanish. More recently, I have had help from a number of people at Edinburgh including Pam Roderigez and her husband, Anna Gavarro, who helped with Catalan, and Antonio Sanfilippo, who helped with Italian. Most of all though I wish to thank John Beaven of the Department of Artificial Intelligence at Edinburgh. John is bilingual in English and Spanish with a background in Linguistics. He acted as the principal informant for the work undertaken in this thesis. He has responded to innumerable enquiries over the past two years and on occasions has worked his way through pages of sentences providing grammaticality judgements. At no time did he even hint that I should perhaps try working on a language he wasn't familiar with, though the temptation must have been difficult to resist. Without his continual assistance, this thesis would not have been written.

When John Green left York University, Steve Harlow took over and directed my work to its conclusion. In many ways it has not been easy for him to keep a close eye on the direction this work has taken. I was never a full time student at York and since my arrival in Edinburgh, my job has taken up a large proportion of my time. The result was that on several occasions throughout the last four years, it seemed to me that I could not complete this research. Throughout all this time Steve remained a calming influence, ready to comment in detail on anything I wrote and always willing to give advice. In the last few months especially, he went through drafts with the greatest care, helping me build a sense of confidence in what I was doing.

Others I would particularly like to mention, some of whom may be surprised at their inclusion here but who nevertheless played a role directly or indirectly in the successful completion of this thesis, are: Patrick Griffiths of York University, who was instrumental in admitting me to the postgraduate course in Linguistics at York; Ronnie Cann, who read and commented on a substantial part of this thesis; Ewan Klein, whose advice at an early stage helped me to understand better the consequences of my thinking; John Phillips, who helped me towards a better understanding of Generalised Phrase Structure Grammar; Steve Isard for his occasional advice and Colin Matheson, my colleague on the Speech Project at Edinburgh, with whom I have daily exchanges on matters linguistic.

Finally, it is one's wife and children who suffer most from one's obsessions. My family have had to live with this thesis for seven years. I can only say that I really do appreciate the support they have given me through all those years. This thesis is dedicated to Bevinda, David, Joanna and - though he isn't yet old enough to know it - Simon.

ABSTRACT

The phenomenon of semi-free word order exhibited by many languages, especially at the sentence level, poses problems for the descriptive and explanatory adequacy of many recent versions of Categorial Grammar. This thesis presents as approach to Categorial Grammar which involves factoring out the linear ordering properties of functional categories from other aspects of categorial structure. A formal construct - referred to as a *local structure* - is defined, of which two versions, unordered and ordered local structures, play a key role in the model. Unordered local structures are assigned in the lexicon; ordered local structures explicitly specify the order of application of arguments to functions and the directionality of those arguments. A mapping between the two structures is given and the linear ordering constraints, expressed in a boolean logic, are interpreted as filters on this mapping. The result is a model in which complex data from Spanish - the language primarily used to illustrate semi-free word order - can be economically described. A number of issues which arise in the application of this model are discussed.

An extended version of the model using attribute-value matrices and unification is also presented. This has the advantage that linear ordering constraints apply directly to lexical and phrasal categories without the necessity of expanding out ordered local structures. One consequence of this innovation is the need for a three-valued interpretation of the linear ordering logic in order to model the contingency of partial valency satisfaction in the grammar. Further analysis of the Spanish data is undertaken within this model with particular attention being paid to intra-linguistic word order variation and the analysis of Spanish clitic personal pronouns.

INTRODUCTION

0.1 Outline of the Issues Discussed and Principal Findings

Applicative functional calculi such as Categorial Grammar (Ajdukiewicz (1935), Lambek (1958, 1961, 1988), Moortgat (1988a, 1988b)) have recently attracted considerable attention as potential models for the syntactic and semantic analysis of natural languages (Bach (1983a, 1983b, 1984), Flynn (1983), Steedman (1985, 1987a, 1987b, 1988), Oehrle, Bach and Wheeler (1988), Moortgat (1988a, 1988b)). They have certain distinctive characteristics which bring with them advantages for the description and analysis of linguistic data, among which may be included the emphasis on lexicalism and the concomitant relocation of the "explanatory burden" (Moortgat 1988b, page 1) from the syntactic component to the lexicon, the modelling of incomplete expressions as functions with basic dependencies such as government, control and concord "defined on the function-argument hierarchy rather than on structural configurations" (Moortgat 1988b, page 1), a flexible approach to constituency structure and a strictly compositional approach to semantics with a semantic operation corresponding to each syntactic operation.

Despite these advantages, there are numerous natural language phenomena which do not readily submit to analysis within a function-argument framework, even one augmented by type shifting rules and complex compositional combinators. Among these phenomena may be included the flexible word or constituent ordering which is found in many languages, particularly at the sentence level. Most versions of Categorial Grammar that have been discussed in the literature are either too restrictive or far too liberal with respect to the permutations of word order they admit. The aim of the research reported in this thesis is an attempt to find a formal syntactic theory, fundamentally applicative in nature and lexically oriented, which will permit the expression of linguistically sensitive statements about semi-free word order which are required for the descriptively adequate treatment of natural language data. The results are recorded in the following chapters.

The first step in this enquiry required the collection of reasonably clear-cut data on flexible word ordering. Spanish is a semi-free word order language and is the source of most of the data used in the thesis. The term *semi-free* is used to indicate that at some level of analysis (notably the sentence level), major constituents can be found in different orderings, but that not all logically possible orderings are judged to be grammatical by native speakers. This is a difficult area, of course. Judgements vary considerably from region to region and speaker to speaker. It is possible, nevertheless, to isolate certain facts about Spanish sentences which virtually all native speakers are agreed upon. A simple example - which illustrates the problems experienced in obtaining the data as well as the kind of approach adopted in this thesis - is the question of the ordering of subject (S), verb (V) and direct object (O) relative to one another in simple transitive sentences. There is considerable disagreement among Spanish speakers with regard to the flexibility they are prepared to tolerate in the ordering of these constituents. Most speakers accept the orders SVO, VSO and VOS, although it was found that in the case of one informant there was a distinct preference for the SVO/VOS alternation; VSO was regarded as marginal at best. However, the judgement here was probably influenced by Catalan, generally recognised to be a more strict word order language than Spanish. Other speakers freely accept all the orderings mentioned so far - though acknowledging that some are more marked than others - and in addition accept OVS under certain circumstances. Again, judgements vary considerably. One thing *all* speakers are agreed upon is the fact that the verb final orderings of SOV and OSV are ungrammatical. Variations of this magnitude are quite typical in studies of word order phenomena and decisions have to be made by the linguist which inevitably conflict with some native speakers' intuitions. What is important is to have a formal descriptive framework capable of expressing these differences and defining the orderings which are generally accepted. The principal aim of this research has been to find a version of Categorial Grammar in which such statements can be made.

An influential approach to the analysis of word order in monostratal frameworks is the Immediate Dominance/Linear Precedence factoring developed within Generalised Phrase Structure Grammar (Pullum (1982), Gazdar, Klein, Pullum and Sag (1985)). This explicitly recognises the fact that linear ordering information is of a different kind to other syntactic information such as the dominance relation pertaining between categories in a constituent structure tree. It should, therefore, be described independently. An interesting starting point, then, for the exploration of word order in Categorial Grammar is to ask if anything analogous to Immediate Dominance/Linear Precedence factoring can be developed within an applicative functional framework.

At first sight it seems unlikely that there could be common ground between Generalised Phrase Structure Grammar and Categorial Grammar since there is no notion of a local dominance relation in Categorial Grammar over which the equivalent of linear precedence statements could be defined. This obstacle disappears, however, given two observations.

Firstly, there is an isomorphism between the conventional, so-called *curried* functional representations of Categorial Grammar categories in which the function is constrained to accept one argument at a time and *non-curried* representations in which the arguments form a list. (The demonstration of this mapping was originally due to Schönfinkel (1924)). Secondly - and following on from the first observation - the same formal abstract object underlies both Phrase Structure rules and non-curried Categorial Grammar categories. It is this second observation that forms the starting point of the work reported in this thesis.

In Gazdar, Klein, Pullum and Sag (1985, pages 52-55), an Immediate Dominance statement is given formally as an ordered pair, $<\alpha,\{\beta_1,\beta_2,\ldots\beta_n\}_m>$, in which the first element is a designated (perhaps complex) symbol and the second element is a multiset of designated (perhaps complex) symbols. Basic linear precedence statements are pairs $<\beta_1, \beta_2>$ and the set of admissible local trees induced by the grammar which derive from a single Immediate

Dominance statement is the set of trees each with root α and leaves $\beta_1,\beta_2,...\beta_n$ over which an exhaustive ordering relation is defined which is compatible with all the Linear Precedence statements. The formal representation of each of these graph-theoretic objects is given as $<\alpha,<\beta_1,\beta_2,...\beta_n>>$ where the second element is an ordered n-tuple. Notice the simple, but crucial point that these set-theoretic constructs are interpreted graph-theoretically as local trees, the well-formedness of which are guaranteed by an isomorphism between the set-theoretic and graph-theoretic constructs. These objects can, of course, be given quite different interpretations. Within Generalised Phrase Structure Grammar itself, an earlier account regarded the output of this application of linear ordering constraints to Immediate Dominance statements as Phrase Structure rules rather than local trees. Yet another interpretation - the one relevant to this thesis - involves taking the output constructs as non-curried categorial functions in which the first element in the n-tuple is the range and the second element, the domain. This observation is the first step towards the development of a formal theory of word order in Categorial Grammar.

Three generalisations of the formal objects underlying the Generalised Phrase Structure Grammar account were undertaken in order to allow for the full expression of categorial functions.

First, the ordered pairs were generalised to n-tuples. This allows the underlying construct to model further aspects of category structure by introducing new information into the n-tuple. The n-tuple whose second member is a multiset is referred to as an *unordered local structure*; its equivalent with an n-tuple as its second member is referred to as an *ordered local structure*. Collectively, they are referred to as *local structures*. The mapping between unordered and ordered local structures essentially involves taking the cross-product of all possible orderings from the unordered local structure and applying the independent ordering constraints to filter out a set of admissible ordered local structures.

Second, the basic linear ordering statements were generalised to statements about the content and form of the second element of the local structure. They are taken to be n-tuples rather than pairs with the result that $<\alpha>$, $<\alpha, \beta>$, $<\alpha, \beta, \gamma, \ldots>$ are all well-formed linear ordering statements. However, of these only *two* need to be considered as basic - $<\alpha>$ and $<\alpha, \beta>$ - since the ordering imposed by an n-tuple can be represented by n-1 pairs. If α, β are interpreted as complex feature bundles, $<\alpha>$ simply asserts that a local structure contains that feature bundle. $<\alpha,\beta>$, on the other hand, asserts that the ordered local structure contains α and β and further that the feature bundle associated with α linearly precedes the feature bundle associated with β in the second element of the ordered local structure. Again, different interpretations of this abstract linear ordering statement are available. Under the functional interpretation, which is the one of interest here, statements of the form $<\alpha>$ (or simply, α) can be used to constrain feature specification in complex feature bundles including the directionality marking associated with the arguments of functions. Statements of the form $<\alpha,\beta>$ on the other hand constrain the order of combination of the arguments with a function.

There is one complication to this generalisation of linear ordering statements. The constraints mentioned in the previous paragraph all refer to the second element of the n-tuple that forms the local structure. However, it is sometimes necessary to have the ordering statements refer to other elements in the n-tuple. In order to accommodate this, a new ordering constraint is defined of the form $f{:}\gamma$, where γ is an element in the n-tuple, $n \geq 3$. The discrepancy between this constraint type and the generalisations above is not resolved until the introduction of a full attribute-value matrix formalism for category representation later in the thesis.

Third, recall that in Generalised Phrase Structure Grammar the admissible local trees were taken to be those compatible with *all* the linear ordering statements. This effectively treats the set of linear ordering statements as a conjunctive list. It is argued here that this view of linear ordering statements needs to be generalised to arbitrary boolean combinations of basic expressions. The motivation for this came from an examination of the complex word ordering found in Spanish. Local structures are *flat* by virtue of the fact that they define an immediate relation between a single designated symbol and either a multiset or ordered n-tuple of symbols. Under a functional interpretation, a transitive verb, for example, whose range is a sentence (thereby adopting the position that sentences are projections of finite verbs) will have Noun Phrases in its domain. Without the aid of further structure, the linear ordering statements need to be more complex than simply the conjunction of basic expressions in order to describe the various orderings found in Spanish. Consequently, a Propositional Logic is introduced to express the complex word order statements, a logic which takes the set of ordered local structures as its model.

To summarise so far. The claim is made here that a formal and abstract representation called a local structure models both Phrase Structure Grammar and Categorial Grammar. In the case of Phrase Structure Grammar, this structure is interpreted in graph-theoretic terms as a local tree; in Categorial Grammar, it is interpreted as a functional category structure. Notice that quite frequently in work on Categorial Grammar the results of function-argument application are represented in terms similar to the representation of local trees. This is not, however, what is meant by local structure here. If it is desired to represent function-argument cancellation in graph-theoretic terms, then the tree resulting is the analogue of a non-local tree in Phrase Structure Grammar. But, as Moortgat (1988b) points out, it is not necessary for this notion of a tree to be imported into Categorial Grammar: the function-argument hierarchy is better interpreted as a derivational history since it is not required for semantic interpretation. This is in contrast to the graph-theoretic constructs of Phrase Structure Grammar which are the backbone upon which semantic interpretation hangs. However, despite Moortgat's observation, in this thesis partial reification of the function-argument hierarchy as a graph-theoretic object is assumed in order to provide a vehicle for coordination. This is not crucial to any argumentation in the thesis, but it does allow for the analysis of coordinated structure in traditional terms.

The outcome of this examination of the common ground between Generalised Phrase Structure Grammar and Categorial Grammar is a model which stipulates unordered local structures in the lexicon and takes (potentially) complex boolean expressions to be filters on the mapping from these unordered to a set of ordered local structures. The boolean expressions, which are given as statements in a Propositional Calculus, are interpreted against a model of potential local structures ie. the set of all possible expansions of the unordered local structures.

This framework is used for the detailed examination of a variety of word order problems in Spanish and it allows for the economical description of that data. However, despite the interest of this model of Categorial Grammar for the description of semi-free word order data, it has important limitations. Two of particular importance relate to the structure of the categorial lexicon and the mapping from unordered to ordered local structures.

The complexity of the orderings found in semi-free word order languages forces many of the linear ordering constraints to be parochial in nature: being true of one function type, but not true across the entire grammar. This parochial nature of the linear ordering constraints causes massively redundant stipulation in the lexicon. Conventionally, the categorial lexicon is taken to be a pairing of words with categories. It is easy to see that in any particular implementation of such a lexicon, a judicious use of pointers between categories might avoid some of the more obvious redundancy: the structure and ordering constraints on transitive verbs, for example, need only be given once. It is possible, however, to adopt a more radical approach to the lexicon which virtually eliminates redundant structure and specification altogether.

Consider the category for an auxiliary verb. It is analysed as a function taking (among other potential arguments) a function as argument. It returns a sentence. This functional argument will be a verbal function (such as a transitive verb). Why not regard the auxiliary not simply as taking a category that happens to be the category of a transitive verb but rather as taking the verbal function itself? In other words, auxiliaries may be analysed not so much as independent categories but as complexes of relations constructed from other categories in the lexicon. Ultimately, all categories "bottom out" as the primitive categories of the grammar such as S, NP, N. This suggests complex information sharing in the lexicon whereby linear ordering constraints may be inherited from one category to another. Not only is redundant stipulation avoided in this way, but a clearer relationship is established between lexical categories. The problem is that given the formal apparatus for the description of categories so far introduced, there is no way of precisely describing *how* this information sharing is carried out. The mechanisms for this only come with a further re-analysis of the formal machinery of Categorial Grammar.

The mapping from unordered to ordered local structures presents a variety of problems. A particularly difficult one is the fact that a single unordered local structure may be associated with a (potentially) large number of ordered local structures, all of which need to be spelled out prior to application of the combinators. While this is a matter of implementation rather than formal theory, it does raise the question whether or not a formal notation exists which would allow the grammar to be defined and used without the explicit mapping to ordered local structures. This type of question is not new, of course. Shieber (1984) investigated the direct parsing of Immediate Dominance/Linear Precedence grammars for the very same reasons.

The attribute-value matrix notation which is currently used in a number of unification-based theories of grammar (see Shieber (1986) for an introduction and overview) is a very general notation for the representation and manipulation of syntactic information. It is interesting to explore the possibility of using this notation for the representation of unordered local structures. Certain features of the attribute-value matrix notation are clearly useful. For example, the mechanism available for structure sharing may be expected to assist with the solution noted above to the problem of redundancy in the categorial lexicon. It was decided therefore to explore the extent to which this notation could be used to model the relevant aspects of a linear order factored Categorial Grammar.

When linear ordering constraints and function-argument structure are factored out, a problem arises with respect to the satisfaction conditions for function-argument application. The problem is that the ordering constraints are constraints about *all* the arguments of a function. When just one of these arguments is offered, it is not clear under what circumstances it should be accepted as a valid argument for that function. The solution to this problem turns out to be unexpectedly interesting. A three-valued logic is required to evaluate the linear ordering constraints on function categories in order to determine the suitability of a potential argument. The logic needs to be three-valued in order to model the contingency of partial valency satisfaction displayed by function categories.

The result of these investigations is that a new version of Categorial Grammar is offered which is sufficiently distinct from other Categorial Grammars to be dubbed *Extended Categorial Grammar*. This is a member of the family of applicative categorial calculi which is linguistically sensitive and capable of directly expressing the complex word order facts found in semi-free word order languages.

The formal definition of this new model also allows the programme of work undertaken here to be seen from the perspective of other categorial research. This Introduction started by pointing out that conventional directional Categorial Grammars are either too restrictive or far too liberal in their treatment of word order. Moortgat (1988b, pages 40-48) has an excellent discussion of this. He identifies a hierarchy of Categorial Grammars from the simplest applicative systems through the classical Lambek calculus (Lambek 1958) - referred to as **L** - to various extensions of **L** which include operations such as permutation over strings of categories. The Lambek system is rigid. Although it allows free bracketing of strings (it has a rule of Associativity which guarantees that if a sequence of categories reduces to a single category with one bracketing, it will reduce to that category with *any* bracketing) it does not allow permutations. Consequently, the semi-free word order facts of, say, Spanish could only be captured by having multiply different directional categories assigned to items in the lexicon. With the addition of a rule of Permutation, however, all orderings (and all bracketings, of course) become immediately possible. Moortgat (1988b) is, in part, an investigation of the categorial space between the classical system **L** and the augmented version with Permutation, **LP**.

The enterprise undertaken here can now be seen as one way of addressing these issues with respect to the linguistic sensitivity of grammars in the categorial space between **L** and **LP**. The conclusion drawn with respect to Extended Categorial Grammar is that it has the generative capacity of **LP** but it allows for the specification of individual grammars which are weaker than **LP** but more powerful than **L** (strictly, product-free **L** without Associativity). In other words, it allows for the construction of specific natural language grammars within this category space which are sensitive to the range of linear ordering facts exhibited by natural languages.

0.2 Outline of the Thesis

The thesis is divided into three parts. The First Part provides the necessary background in two chapters. Chapter One is an introduction to Categorial Grammar. This is given in some detail since it is assumed that Categorial Grammar is still a relatively unfamiliar formalism for linguists. The chapter presents a simple conventional Categorial Grammar followed by a detailed examination of the combinatorial properties of an important recent version referred to here as Generalised Categorial Grammar[1].

The chief difference between the simple and generalised versions lies in the use of multiple combinators in the latter grammar. Much recent research in this area has been devoted to explicating the formal and mathematical properties of these augmented versions of Categorial Grammar. As already indicated, Moortgat (1988b) is an important recent overview of these issues. The Chapter concludes with a brief survey of the hierarchy of Categorial Grammars as defined by Moortgat.

Chapter 2 presents some basic data from Spanish which is taken throughout this thesis as a representative of a semi-free word order language. The data are restricted to a few syntactic phenomena which are used in later chapters to exemplify the new formalism of the linear order factored grammar. These data include word order possibilities in transitive and di-transitive sentences as well as the position of the subject in various sentence frames. There is some discussion of the form and ordering of direct and indirect objects.

The Second Part of the thesis develops a theory of linear ordering in Categorial Grammars based on the proposition that the categories in such grammars can be interpreted as local structures in the sense introduced above. The problem of accounting for word order in Categorial Grammar has, of course, been addressed a number of times in the literature. Chapter 3 therefore provides a brief overview of three approaches to word order (Section 3.2), contrasting them with the approach presented here. This chapter then continues (in Section 3.3) with an informal presentation of the central notion of a local structure and its categorial interpretation. Since this notion owes so much to Generalised Phrase Structure Grammar, some background to the factoring of linear order in that model is first provided. Chapter 3 ends with a formal statement in Section 3.4 of the theory of linear order in Categorial Grammar.

Part Two continues in Chapter 4 with a detailed investigation of the consequences of applying this account to Categorial Grammar. Several aspects of linear order factored Categorial Grammars are considered. These include the interaction between the linear ordering logic and coordination (Section 4.2), the consequences of introducing limited unification into the grammar (Section 4.3), feature passing within categories (Section 4.4) and, finally, the use of metagrammatical definitions of categories (Section 4.5).

Chapter 5 applies the theory developed in the preceding two chapters to the analysis of the Spanish semi-free word order data given in Chapter 2. The aim of this chapter is two-fold: to demonstrate the complexity of word ordering facts in Spanish by considering various approaches to the basic data and, to show the power of the new categorial formalism for expressing generalisations about that data. Particular emphasis is laid on an analysis of word order in transitive sentences with reference to grammatical relations and the contrast of Given/New information (Section 5.2.2). The final section of the Chapter contrasts the ordering properties of auxiliary and subject control verbs in Spanish (Sections 5.4 and 5.5).

Part Two concludes in Chapter 6 with an examination of the structure of the categorial lexicon. This is a particularly important topic in Categorial Grammar given the fact that so much information resides in the categories associated with lexical entries. It is shown how redundancy can be removed and information sharing among categories explicitly modelled.

The Third Part of the thesis extends and modifies the theory and linguistic coverage of Part Two. There are a number of problems with the implementation of linear order factored Categorial Grammars as presented in Part Two. These are addressed at the beginning of Chapter 7 and a new formalism for category structure is introduced there. The issues surrounding the introduction of a three-valued logic are discussed in detail (in Section 7.3) and Chapter 7 ends with a suggestion for a generalised combinator fulfilling the roles of Functional Application, Composition and Substitution (Section 7.4).

Chapter 8 applies the Extended Categorial Grammar model of Chapter 7 to further Spanish data. First of all, a detailed treatment of simple sentence structure in Spanish is given, showing in particular how the particle *a* in these sentences interacts with Grammatical Relations. In addition, (in Section 8.2) a decomposition of Grammatical Relations - which have hitherto been taken as primitives - is suggested that allows for interesting generalisations in the analysis of these simple sentence types. Next, the problem of word order variation mentioned earlier is addressed and different linear ordering constraints for different word orders are compared. A suggestion for a theory of markedness is also briefly sketched (Section 8.3). Finally, (in Section 8.4) there is a brief examination of the role of object pronoun clitics in Spanish, restricted to their effects on word order.

The thesis concludes in Chapter 9 with a brief evaluation of the work presented here and a statement of what is referred to as *the categorial perspective on word order*, a view of ordering constraints as syntactic properties projected from the lexicon via the function-argument hierarchy. This embodiment of the essential lexicalism of Categorial Grammar ensures that the complex word orderings exhibited by semi-free word order languages are the product of interactions between the ordering associated with individual lexical items mediated by the combinatorial processes of the grammar.

0.3 Notes

[1] Another term frequently used in the Categorial literature is *Combinatory Grammar.* This term, associated like *Generalised Categorial Grammar* with Steedman, derives from the use of the variable-free Combinatory Logic of Curry and Feys (1958) for the semantic interpretation of the combinatory operations.

PART ONE

THE BACKGROUND

CHAPTER 1

CATEGORIAL GRAMMAR: AN INTRODUCTION

1.1 Introduction

Categorial Grammar is a monostratal theory of grammar[1] in which the combinatorial properties of words are encoded directly and explicitly in the categories associated with items in the lexicon rather than in a set of independent statements in the form of phrase structure rules. The origins of the theory lie in developments in formal logic (see Ajdukiewicz (1935)) but the theory has a long history of application to natural language analysis. Apart from Ajdukiewicz's original paper, see Bar-Hillel (1953), Lambek (1958, 1961, 1988), Bach (1983a, 1983b, 1984), Lyons (1968), Geach (1971), van Benthem (1986), Klein and van Benthem (1987) and Moortgat (1988a, 1988b). In particular, the collection of papers in Oehrle, Bach and Wheeler (1988) shows, in the editors' own words, "that work in Categorial Grammar in the broad sense has moved beyond the stage of sporadic rediscovery and reached a critical mass" (Oehrle, Bach and Wheeler, page 8).

Moortgat (1988b, pp. 1-2) characterises the programme of categorial research by drawing attention to those aspects which distinguish it from related monostratal theories of grammar such as Generalised Phrase Structure Grammar (Gazdar, Klein, Pullum and Sag (1985)) and Head-Driven Phrase Structure Grammar (Pollard (1985), Pollard and Sag (1987)). He emphasises four such aspects.

(1) *Lexicalism*: Categorial Grammar takes the tendency to "shift the explanatory burden" from the syntactic component of the grammar into the lexicon

much further than most recent theories of grammar. Syntactic information is projected entirely from the categorial structure: "In its most pure form, Categorial Grammar identifies the lexicon as the only locus for language-specific stipulation" (page 1). The ideas on word order presented in this thesis are very much in the lexicalist spirit as Moortgat defines it. The linear ordering properties of lexical items are encoded directly *within* the lexicon rather than in an independent set of configurational syntactic rules. This approach has the added advantage that the interaction of word order and other syntactic properties of lexical items can be expressed directly by the formalism presented in this thesis.

(2) *Function-Argument Structure*: incomplete expressions in Categorial Grammar are modelled as functions, syntactically and semantically. Dependencies between expressions determining grammatical phenomena such as government, control and agreement are defined over the function-argument hierarchy in the derivation of a sentence, rather than over a structural configuration. This gives a different and new perspective on these central concepts of grammatical description. It will become clear in the course of the work reported in this thesis how effective this approach can be, particularly when taken together with the use of partial information in category structures.

(3) *Flexible Constituency*: Simple directional Categorial Grammars using only Functional Application assign unique constituency structure to non-ambiguous expressions. The generalised versions of Categorial Grammar with their complex combinatory and type changing rules standardly induce multiple, semantically-equivalent derivations for a single word string. This relativisation of constituency structure has at least two justifications. Firstly, it may permit a uniform left-branching derivation of sentences. This is significant if one's interests lie in the direction of left-to-right incremental parsing of natural language input. Secondly, it provides for coordination across many more adjacent substrings than conventional Phrase Structure Grammar coordination schemata. Much of the categorial linguistic literature has, in fact, been directed at motivating the need for such a wealth of potential coordinations from the examination of complex coordination phenomena in natural

languages, in particular non-constituent coordination. The chief problem with most accounts is that they allow *too much* coordination. As with word order in the semi-free word order languages discussed later, there are limits to the freedom to coordinate. One of the strengths of the theory of word order in Categorial Grammar developed in this thesis is that it provides a means for defining extensive but systematically constrained coordinations and word orders.

(4) *Compositionality*: The semantic value of a complex expression is a function of the semantic values of its composing parts. This fact is encoded directly into the categories of Categorial Grammar just as is the homomorphic or structure-preserving relationship between the syntax and semantics. Compositionality is a central, guiding principle in much modern grammatical theory; in Categorial Grammar it is particularly transparent and clear.

Formally, a Categorial Grammar possesses three components: a (frequently implicit) list of atomic categories together with a set of formation rules for well-formed formulae or complex categories; a categorial lexicon in which atomic and complex categories are assigned directly to lexical items; and, thirdly, a set of combinatory and perhaps type changing rules defined over the categories. This arrangement ensures that syntactic entities resulting from the application of the combinatory and type changing rules are themselves well-formed formulae of the system according to the formation rules. This is an attractive property of a grammar since it encodes in a very direct way the idea of the hierarchical structure of well-formed expressions of the language induced by the grammar.

This chapter opens (Section 1.2) with a simple non-directional categorial calculus intended to illustrate these components of Categorial Grammar. The principal feature of this grammar is the fact that it possesses only the one rule of combination (Functional Application) of classical Categorial Grammar (Ajdukiewicz (1935)). Section 1.3 is an introduction to the semantics of simple Categorial Grammar and the chapter continues in 1.4 with a description of Generalised Categorial Grammar. This makes use of two innovations:

directional categories and more complex combinators. Different sets of combinators are available in Generalised Categorial Grammars depending, to some extent, on the motivation behind the grammar. The most important of these combinators are discussed with examples of their application. The Chapter concludes with a review of some of the logical properties of Categorial Grammars.

1.2 A Simple Categorial Grammar

Figure 1.1. presents a simple non-directional Categorial Grammar. With this grammar it is possible to generate and parse the sentence in 1.1a, assigning it the structure given in 1.1b.

Atomic Categories: {NP, S}

Formation Rules for well-formed formulae :

> i every atomic category is a well-formed formula of the grammar;
>
> ii every expression of the form X|Y is a well-formed formula of the grammar where X, Y are well-formed formulae of the grammar. Such an expression is referred to as a complex category. Parentheses are used to mark the scope of expressions, but are omitted where no ambiguity results.

Lexicon:	{John, Bevinda}	= NP
	{loves}	= (S|NP)|NP

Rule of Combination:

> A complex category of type X|Y in which X,Y are well-formed formulae of the grammar can combine with a category of type Y to form a category of type X.

Figure 1.1
A Simple Categorial Grammar

```
1.1   a   John loves Bevinda

      b   BA: S
                NP ∍ John
          FA:   S|NP
                    (S|NP)|NP ∍ loves
                    NP        ∍ Bevinda
```

A few comments about presentation of derivations are relevant at this point. There is no generally agreed method of laying out a derivation in Categorial Grammar. Two formats are currently employed. In the linguistic literature, a commonly used format follows that introduced by Steedman in a number of papers (Ades and Steedman (1982), Steedman (1987a, 1987b)). This involves drawing horizontal lines across constituents which combine together. The visual effect is to make a derivation resemble a traditional tree diagram except that the root is at the bottom and the terminal word string at the top. Another format, which derives from the logical literature, is used here. This is illustrated in 1.1b above. This indented list format is intended to suggest a comparison with the format used for listing derivations in logic texts. The advantage as far as this thesis is concerned is primarily a matter of convenience: it takes up less space on the page than its alternative. In this format, as can be seen from 1.1b, indentation corresponds to branching in a tree diagram with a top-down, left-to-right order on the derivation. The rule identifier before the colon is explained below. The ∍ notation introduces lexical items.

The formation rules, lexicon and rule of combination constitute a formal system for the derivation of sentences and their associated structural analyses. The similarity to the way that logical systems are presented is not a coincidence.[2]

The categorial calculus in Figure 1.1 is presented in terms of an uninterpreted category notation. In fact, as already indicated, it is usual to give the complex categories of Categorial Grammar a function-argument interpretation so that a category of the form X|Y is understood to be a function with domain a category of type Y and range a category of type X. Functions may, of course, have more than one argument. However, they are conventionally assumed to be unary. For example, in a simple, non-directional Categorial Grammar of this sort, the verb *give* will standardly be assigned the category ((S|NP)|NP)|NP reflecting its di-transitive nature, which is modelled, on this account at least, as a function from NP denotations (indirect objects) into a function from NP denotations (direct objects) into a function from NP denotations (subjects) into sentence denotations. This restriction to unary argument structure is conventional, not essential. It has been noted for some time that there is an equivalence between higher-order unary functions and first order n-ary functions (see Dowty (1982) and, originally, Schönfinkel (1924)), a fact which makes possible the use of ordered n-tuples of arguments deployed in the theory presented later in this thesis.[3]

It will have been noticed that the grammar in Figure 1.1 induces more than the sentence in 1.1a. Complex categories of the form X|Y accept their arguments from either the right or the left. Consequently, the rule of combination is standardly given in two parts to reflect this fact. In 1.2, each part carries a label used to identify the rule in derivations: FA stands for Forward Application; BA for Backward Application.

1.2	a	FA:	X|Y	Y	=>	X
	b	BA:	Y	X|Y	=>	X

If only the first version of the rule of combination were available, 1.1b could not be generated by this grammar since "John" precedes its function "S|NP". However, given both versions in 1.2, two things immediately follow: (1) all orderings of the three words in 1.1a become derivable and, (2) all derivations of each ordering are possible. These are illustrated in 1.3 below where the bracketing indicates substrings obtained by the first application of one version of the combinator and the label on the right indicates which version of the rule in 1.2 is used for this first application. The outer brackets are omitted. Given standard assumptions about constituent coordination, these substrings may be the source of conjoined structure.

1.3	a	[John loves] Bevinda	BA
	b	John [loves Bevinda]	FA
	c	John [Bevinda loves]	BA
	d	[loves John] Bevinda	FA
	e	[loves Bevinda] John	FA
	f	[Bevinda loves] John	BA
	g	Bevinda [loves John]	FA
	h	Bevinda [John loves]	BA

Clearly, this is embarrassingly liberal. Even ignoring the semantic problems posed by the different interpretations of 1.3a and 1.3f, only a few of these orders are syntactically acceptable. A linguistically motivated Categorial Grammar must provide the means for reducing the over-generation while at the same time retaining those properties of the grammar which allow for the complex coordination and word order facts of natural languages. It is considerations of this kind that led to the introduction of an indication of directionality on the arguments of functions (discussed in Section 1.4 below) and that have led to the development of the linear order factored Categorial Grammar presented in this thesis. Before augmenting the formalism with these new devices, however, it is important to consider briefly the relationship between categorial structure and semantics in Categorial Grammar.

1.3 The Semantics of Categorial Grammar

The title of McDermott (1978) neatly summarises one of the imperatives in the Tarskian universe which modern formal semantics inhabits: no notation without denotation. One of the fundamental tasks facing a theory of natural language grammar is to provide a semantic interpretation for each of the well-formed syntactic expressions of the grammar. In model-theoretic semantics, this is usually taken to mean that the semantic component of the grammar must provide (a) the denotation of primitive or basic expressions relative to a model and (b) an account of the way that the denotations of complex expressions are built up from the denotations of their constituents. In other words, the semantic component of the grammar provides a recursive definition of the semantics of all well-formed expressions of the language relative to some model.

Generally speaking, Categorial Grammars provide a very transparent relationship between syntax and semantics since the syntactic structure of complex categories directly encodes their semantic properties. This section expands on this in order to lay down the foundations for the more complex semantics of Generalised Categorial Grammars introduced later. It does not, however, attempt anything beyond a very elementary account of the formal semantics of Categorial Grammars.

Assume a function *Den* from categories to sets of denotations in a model M (cf. Gazdar, Klein, Pullum and Sag 1985, page 184). The denotations of atomic categories are given by stipulation. So, for example, *Den*(S) = {0, 1} and *Den*(NP) = E, where E is the set of entities in M. This last is an oversimplification but will serve for present purposes. For complex categories, the rule for semantic interpretation is given in 1.4.

1.4 $Den(\mathrm{X}|\mathrm{Y}) = \langle Den(\mathrm{Y}), Den(\mathrm{X})\rangle$

where the notation $\langle\alpha, \beta\rangle$ is interpreted as a function from elements of type α into elements of type β.

Since this recursive semantic definition exactly mirrors the syntactic definition of category structure for Categorial Grammar, the result is that for any arbitrary category, what it denotes in the model M follows directly from its internal structure. Categorial Grammar categories wear their semantics on their sleeve. It is quite possible, therefore, to map directly from the syntactic category structure into denotations in M, a course paralleling that adopted in Montague (1970). However, a standard way of giving the semantics, also deriving ultimately from Montague (see Montague 1973), is to use an intermediate representation as a bearer of the semantic values of Categorial Grammar categories.[4] This "logical form" has its own semantic mapping into the model. The logical form used in Montague (1973) is a higher-order typed version of the λ-calculus. Since this second approach helps to clarify how the compositional semantics works in complex derivations in Categorial Grammar, it will be assumed in this Chapter, although it is important to remember that nothing substantive hangs on the decision to construct such an intermediate representation. In the translations given here, the logical form will, however, be restricted to a simple, non-typed version of the λ-calculus.[5]

Each assignment of category to word in the categorial lexicon is associated with an expression in the λ-calculus. In addition, each combinator has associated with it a semantic value for the result of the combination. In the case of the simple Categorial Grammar under consideration, this works out in the following way. The lexical items in Figure 1.1 have the semantic values given in 1.5.

1.5	John	= John'
	Bevinda	= Bevinda'
	loves	= λxλyloves'(x)(y)

where John', Bevinda' are expressions of type e and λxλyloves'(x)(y) is an expression of type <e, <e, t>> in the λ-calculus. The combinator in 1.2 can now be re-written including the semantic translation. This is given in 1.6.

1.6	a	FA:	X|Y: f	Y: a	=>	X: f(a)
	b	BA:	Y: a	X|Y: f	=>	X: f(a)

1.6a is to be read as follows. The category X|Y is a function f of semantic type $<\alpha, \beta>$. The category Y has the semantic type α. The result, X, has the semantic type resulting from the application of the argument to the function, ie. f(a) with semantic type β.

The derivation of 1.1 can now be re-written with a parallel semantic translation, as in 1.7.

1.7	BA:	S		loves'(Bevinda')(John')
		NP ∋ **John**		John'
	FA:	S|NP		λyloves'(Bevinda')(y)
			(S|NP)|NP ∋ **loves**	λxλyloves'(x)(y)
			NP ∋ **Bevinda**	Bevinda'

This very brief account of the semantics of Categorial Grammar is adequate for present purposes. The next section introduces new combinators and for some of these an indication is given of how the compositional semantics works. Apart from that, semantics does not play a central role in the issues discussed in this thesis; it is this fact that enables the very simple account given above to stand as representing the theory of model-theoretic semantics for Categorial Grammar.

1.4 Generalised Categorial Grammar

Generalised Categorial Grammars extend the simple Categorial Grammar of the previous section in several ways, most notably in the introduction of directional categories and new combinatory or type changing rules. These two aspects of Generalised Categorial Grammar will be discussed in turn.

In the Linguistic literature, Generalised Categorial Grammars are associated with Steedman (for example, Steedman 1985, 1987a, 1987b, 1988), Bach (1981), Dowty (1988), and Moortgat (1987, 1988a, 1988b) among many others. One useful collection of recent articles is Oehrle, Bach and Wheeler (1988).

The basic components of a Generalised Categorial Grammar are very like those of a simple Categorial Grammar. An illustrative set of categories, formation rules and a lexicon are given in Figure 1.2. The set of assignments of categories to lexical items in Figure 1.2 will be used throughout the rest of this chapter. These assignments are mostly straightforward. For example, Atomic categories: {NP, N, S, S′}

Formation Rules for well-formed formulae:

> i every atomic category is a well-formed formula of the grammar;
>
> ii every expression of the form X/Y, X\Y, X|Y is a well-formed formula of the grammar where X,Y are well-formed formulae of the grammar. Such an expression is referred to as a complex category.

Lexicon:

{Harry, Betty, Mary, mushrooms}	=	NP
{loves, cook, reading, eat, file}	=	(S\NP)/NP
{the}	=	NP/N
{articles}	=	N
{will, might, must}	=	(S\NP)/(S\NP)
{that}	=	S′/S
{believe}	=	(S\NP)/S′
{and}	=	(X\X)/X
{which}	=	(N\N)/(S/NP)
{without}	=	((S\NP)\(S\NP))/(S\NP)

Figure 1.2
Categories, Formation Rules and Lexicon for a Generalised Categorial Grammar

transitive verbs are taken to be functions from noun phrase denotations into functions from noun phrase denotations into propositions. The latter part of this semantic mapping (ie. the function from noun phrase denotations into propositions) may be given the abbreviation "VP". Modal verbs are, then, functions from VPs into VPs. Notice that the term "VP" has no special status in the Categorial Grammar given in Figure 1.2 beyond being an abbreviation for a particular functional mapping. This is an attractive aspect of the formalism when dealing with the analysis of languages which arguably have no distinct verb phrase in transitive sentences.

Notice also the category associated with *and*. There are several ways of handling coordination in Generalised Categorial Grammar. The analysis here involves the use of a category variable "X". This variable ranges over the infinite set of categories induced by the formation rules and in a derivation all values of the variable are instantiated to the category of its first argument. Problems arising from the introduction of category variables into Generalised Categorial Grammar and the consequent lexical polymorphism induced are currently active areas of research.[6]

1.4.1 Directional Categories in Generalised Categorial Grammar

Generalised Categorial Grammar uses an explicit directional marking on the arguments of function categories. This idea has a long history (see Lyons (1968, pages 227-231) and, originally, Bar-Hillel (1953) for the introduction of directional marking) but it has recently been re-introduced in response to the problem of over-liberal word order flexibility mentioned at the end of Section 1.2. So, for example, the transitive verb "loves" would be represented as in 1.8 in a Generalised Categorial Grammar

1.8 {loves} = (S\NP)/NP

in which the directional marking is interpreted as follows:

1.9 a. X/Y: the function category will only accept its argument Y from its immediate right.

b. X\Y: the function category will only accept its argument Y from its immediate left.

The category X|Y may still be available as a well-formed formula of the system and is now interpreted as a function category looking for its argument either to its immediate left or its immediate right.

There are two conventions for representing directional categories using forward and backward slashes.[7] The original, introduced in Lambek (1958), places the argument *under* the slash. So, a function returning an expression of type X and looking for an argument of type Y to its left is represented as in 1.10.

1.10 Y\X

The alternative convention, adopted by Steedman, places the range of the function consistently to the left of the expression. The equivalent of 1.10 would therefore be 1.11.

1.11 X\Y

This latter convention will be adopted in this thesis.[8]

1.4.2 The Combinatory and Type Changing Rules of Generalised Categorial Grammar

A Categorial Grammar with Functional Application as its only rule of combination and without type changing rules will fail to generate many familiar natural language constructions. In particular, two types of common construction - non-constituent coordination and unbounded extraction - that separate elements of a sentence which belong together semantically cannot be accounted for in a grammar having only Functional Application.

Generalised Categorial Grammar addresses this difficulty by introducing more complex combinatory rules, which generalise the operation of category combination and extend the notion of constituent to include "nonstandard" substrings. However, from a linguistic point of view, the fewer the combinators - and consequently the more restrictive the grammar - the better. New combinators are added to a Categorial Grammar only when motivated by the needs of natural language description. Much recent work in Generalised Categorial Grammar has involved the detailed analysis of natural language constructions as motivation for the introduction of new combinatory or type changing rules. The following paragraphs review some of these arguments.

Consider the three sentences in 1.12.[9]

1.12 a Harry will cook mushrooms

b Harry will cook and might eat mushrooms

c Harry will cook mushrooms and bake a cake

1.12a requires only Functional Application for its derivation, as indicated in 1.13.

```
1.13    FA:     S
                   NP ∋ Harry
        FA:        S\NP
                         (S\NP)/(S\NP) ∋ will
        FA:               S\NP
                               (S\NP)/NP  ∋ cook
                              NP  ∋ mushrooms
```

This has a strictly right branching analysis and is strongly equivalent to the corresponding Phrase Structure Grammar analysis. However, Functional Application will not suffice for 1.12b. In this case, the direct object appears to be unavailable as an argument for the first conjunct *will cook*. A new rule of Functional Composition overcomes this problem.

This rule is given in Figure 1.3. It is so-called from its relationship to the algebraic operation of composing two functions. Standardly, if f and g are two functions such that f: $Y \rightarrow Z$ and g: $X \rightarrow Y$ then the composition of f and g (represented as $f \circ g$) is another function h: $X \rightarrow Z$ such that for any $x \in X$, $h(x) = f(g(x))$. Functional Composition therefore cancels the argument of the principal function with the range of the argument function, producing a new function category. This is indicated in the semantic value of the output in Figure 1.3.

For completeness, all the versions of the rule that have been suggested in the literature are given in Figure 1.3. These divide into Forward (F) and Backward (B) Composition, with in each case the direction of the argument of the resulting function being the same as the direction of the argument of the principal function. There are, of course, other logical possibilities but Steedman (1987a) has argued for a set of universal constraints on the order of arguments which effectively reduce the number of versions of the rule available to the grammar. These constraints are discussed in Chapter 3.

Given this new combinator, 1.12b now has the derivation in 1.14.

FC:	X/Y: f	Y/Z: g	=>	X/Z: $\lambda x f(g(x))$
FCx:	X/Y	Y\Z	=>	X\Z
BC:	Y\Z	X\Y	=>	X\Z
BCx:	Y/Z	X\Y	=>	X/Z

Figure 1.3
Functional Composition

1.14 BA: S
NP ∍ **Harry**
FA: S\NP
BA: (S\NP)/NP
FC: (S\NP)/NP
(S\NP)/(S\NP) ∍ **will**
(S\NP)/NP ∍ **cook**
FA: ((S\NP)/NP)\((S\NP)/NP)
(((S\NP)/NP)\((S\NP)/NP))
/((S\NP)/NP) ∍ **and**
FC: (S\NP)/NP
(S\NP)/(S\NP) ∍ **might**
(S\NP)/NP ∍ **eat**
NP ∍ **mushrooms**

This makes the correct claims about the sentence: the auxiliary is a function into S\NP denotations which when combined with a following lexical verb, so to speak, collects the subcategorised-for arguments of that verb. These arguments are then cancelled *after* the coordination operation.

One consequence of this analysis is that 1.12a now has two derivations: the one indicated in 1.13 and a second in which the auxiliary composes with the lexical verb before cancelling the direct object argument by Functional Application. The advantage of this is that not only is the coordination in 1.12b admitted by the grammar but also the coordination in 1.12c. Furthermore, both derivations of 1.12a receive the same semantic interpretation.

This last point is worth drawing out in some detail. In the simple extensional fragment of the λ-calculus being used here to illustrate the semantic import of the combinators, Noun Phrases are taken to be expressions of type e, *cook* translates into λxλycook'(x)(y) and *will* translates into λPλvwill'(P(v)). The translation of *cook* yields an expression of type <e,<e,t>> and *will* an expression of type <<e,t>,<e,t>>, given that v is a variable over expressions of type e and P a variable over expressions of type <e,t>. The derivation of 1.12a using only Functional Application is given in 1.15; that using both Application and Composition in 1.16.

1.15	FA:	S	will'(cook'(mushrooms')(harry'))
		NP ∋ **Harry**	harry'
	FA:	S\NP	λvwill'(cook'(mushrooms')(v)
		(S\NP)/(S\NP) ∋ **will**	λPλvwill'(P(v))
	FA:	S\NP	λycook'(mushrooms')(y)
		(S\NP)/NP ∋ **cook**	λxλycook'(x)(y)
		NP ∋ **mushrooms**	mushrooms'

1.16	FA:	S	will'(cook'(mushrooms')(harry'))
		NP ∋ **Harry**	harry'
	FA:	S\NP	λvwill'(cook'(mushrooms')(v))
	FC:	(S\NP)/NP	λxλvwill'(cook'(x)(v))
		(S\NP)/(S\NP) ∋ **will**	λPλvwill'(P(v))
		(S\NP)/NP ∋ **cook**	λxλycook'(x)(y)
		NP ∋ **mushrooms**	mushrooms'

The semantics given in 1.16 is necessarily abbreviated. *will* and *cook* combine as λPλvwill'(P(v)) ∘ λxλycook'(x)(y) yielding, in the first instance, λxλvwill'(λycook'(x)(y)(v)). This expression reduces to the one given in the derivation in 1.16 because v is an expression of the same type as y and can therefore eliminate the abstraction in the scope of will'.

Powerful as the combination of Composition and Application is, there are common constructions which cannot be analysed using these two combinatory rules only. One of these is the type of non-constituent coordination illustrated by sentence in 1.17.

1.17 [Harry will cook] and [Betty might eat] mushrooms

The auxiliary and lexical verb in the first conjunct may compose to give a function from Noun Phrases into Verb Phrases, but this will not then combine with the subject. The solution here lies in a type changing rule applied to the subject. Figure 1.4 gives the appropriate rule, where Σ is a category variable.

With this rule available, the derivation goes through as in 1.18.

```
1.18  FA:  S
      BA:        S/NP
      FC:              S/NP
      FR:                    S/(S\NP)
                                   NP ∋ Harry
      FC:                    (S\NP)/NP
                                   (S\NP)/(S\NP) ∋ will
                                   (S\NP)/NP     ∋ cook
      FA:              (S/NP)\(S/NP)
                             ((S/NP)\(S/NP))/(S/NP) ∋ and
      FC:                    S/NP
      FR:                          S/(S\NP)
                                         NP ∋ Betty
      FC:                          (S\NP)/NP
                                         (S\NP)/(S\NP)
                                               ∋ might
                                         (S\NP)/NP
                                               ∋ cook
                 NP   ∋ mushrooms
```

Again, composition is used for delaying combination with the appropriate NP as direct object. Type raising changes the type of the subject to allow the composition to go through prior to coordination.

A different approach to the analysis of 1.17 introduces a new rule type: Associativity. This is given in Figure 1.5. Associativity permutes the two most "oblique" arguments of a function, inducing the derivation in 1.19 below. Moortgat (1988b, pages 15-6) shows that the two derivations in 1.18 and 1.19 are semantically equivalent. This is another demonstration of the inherent flexibility of categorial calculi with many rules of combination.

FR: $X{:}\ f \;=>\; \Sigma/(\Sigma\backslash X){:}\ \lambda v(v(f))$

BR: $X \;=>\; \Sigma\backslash(\Sigma/X)$

Figure 1.4
Type Raising

$$\text{FAss: } (X\backslash Z)/Y: f \quad => \quad (X/Y)\backslash Z: \lambda v_1 \lambda v_2 f(v_2)(v_1)$$

$$\text{BAss: } (X/Y)\backslash Z \quad => \quad (X\backslash Z)/Y$$

Figure 1.5
Functional Associativity

1.19

FA:	S
BA:	S/NP
BA:	S/NP
	NP $\ni$ **Harry**
FAss:	$(S/NP_2)\backslash NP_1$
FC:	$(S\backslash NP_1)/NP_2$
	(S\NP)/(S\NP) $\ni$ **will**
	(S\NP)/NP $\ni$ **cook**
FA:	(S/NP)\(S/NP)
	((S/NP)\(S/NP))/(S/NP) $\ni$ **and**
BA:	S/NP
	NP $\ni$ **Betty**
FAss:	$(S/NP_2)\backslash NP_1$
FC:	$(S\backslash NP_1)/NP_2$
	(S\NP)/(S\NP) $\ni$ **might**
	(S\NP)/NP $\ni$ **cook**
	NP $\ni$ **mushrooms**

Lambek (1958), (summarised in Moortgat (1988b)), presented the classic argument for the necessity, in English at least, for both associativity and composition. This involves the English pronoun system.

Nominative and accusative personal pronouns obviously have different distributional properties. This could be modelled in a Generalised Categorial Grammar by assigning them different categories in the lexicon. Nominative pronouns combine with an S\NP (= VP) category to form a sentence. They therefore have the category S/(S\NP). This effectively excludes strings such as *likes he*. Accusative pronouns on the other hand may be assigned the category S\(S/NP). That is, they look leftwards for the subject-verb combination. Associativity makes this combination possible, and strings of the form *him loves* are not derived. Two derivations (based on Moortgat 1988b, page 17) illustrating these types are given in 1.20 and 1.21.

1.20 a he loves Mary

```
b     FA:  S
                S/(S\NP)  ∈ he
      FA:       S\NP
                     (S\NP)/NP  ∈ loves
                     NP  ∈ Mary
```

1.21 a Mary loves him

```
b     BA:  S
      BA:       S/NP
                     NP  ∈ Mary
      FAss:          (S/NP)\NP
                          (S\NP)/NP  ∈ loves
                S\(S/NP)  ∈ him
```

Unfortunately, Application and Associativity alone will not derive **he loves her**. Composition is required in addition, giving the derivation in 1.22.

1.22 a he loves her

```
b     BA:  S
      FC:       S/NP
                     S/(S\NP)  ∈ he
                     (S\NP)/NP  ∈ loves
                S\(S/NP)  ∈ her
```

Composition of the subject and verb (as in 1.22) gives a derivation relevant to coordinated sentences of the form *he hates but she loves him*. The derivation in 1.23 composes over the verb and accusative pronoun following a change in the associativity of the verb, thereby admitting the possibility of the different coordination: *he loves her but hates him*.

1.23 BA: S

$$
\begin{array}{llll}
\text{BA:} & S & & \\
 & S/(S\backslash NP) \in \textbf{he} & & \\
\text{BCx:} & S\backslash NP & & \\
\text{FAss:} & & (S/NP)\backslash NP & \\
 & & & (S\backslash NP)/NP \in \textbf{loves} \\
 & & S\backslash(S/NP) \in \textbf{her} &
\end{array}
$$

This argument also motivates type raising for Noun Phrases on the standard assumption that like categories coordinate, since full lexical Noun Phrases and personal pronouns freely coordinate in English. The type raising rule already introduced changes the category of the lexical NP appropriately.

Another construction, for which a special combinatory rule seems necessary, is topicalisation. Consider the simple topicalised sentence in 1.24

1.24 Mushrooms, Harry must eat

The fronted direct object is clearly in the wrong position to combine with its verb, even when the subject and verb sequence are composed as in the earlier examples. The special rule given in Figure 1.6 overcomes this problem admitting the derivation in 1.25.

$$\textbf{Top:}\quad X\text{: } f \quad => \quad S_{top}/(S/X)\text{: } \lambda v(v(f))$$

Figure 1.6
A Topicalisation Rule

1.25 FA: S_{top}
Top: $S_{top}/(S/NP)$
    $NP \ni$ **mushrooms**
FC: S/NP
    FR: $S/(S\backslash NP)$
        $NP \ni$ **Harry**
    FC: $(S\backslash NP)/NP$
        $(S\backslash NP)/(S\backslash NP) \ni$ **must**
        $(S\backslash NP)/NP \ni$ **eat**

Notice that this rule introduces a category looking for two arguments from the same direction. Clearly, this rule is related to the type raising rule in Figure 1.4, except that in the former case the arguments came from different directions. The fundamental fact about this rule is that the harmonic relationship between the directional slashes (see Moortgat 1988b, Chapter 3 for the terms *harmonic* and *disharmonic* when applied to the directionality of slashes) allows for a permutation of standard order - the most obvious syntactic fact about topicalisation. This topic has been much discussed in the Categorial Literature. This thesis takes a rather different approach to the permutation of word order, as will become clear in what follows.

As a final example of the combinatorial extensions in Generalised Categorial Grammar, an additional combinator has been motivated for the analysis of parasitic gaps, Consider the noun phrase in 1.26 below.[10]

1.26 the articles which I file without reading

No combination of the combinatory and type changing rules introduced so far will derive this NP, since it is not possible to combine the adjunct missing its NP with the principal lexical verb. Steedman (1985) argues that a new combinator, which he calls Functional Substitution, is required. This is given in Figure 1.7.

The derivation of this sentence is given in 1.27. Again, as with all the previous additions to the armoury of combinators, this rule has generated much discussion, most of which is not immediately relevant to the concerns of this thesis. Chapter 7 develops an approach to the combinatory logic of Categorial

FS:	(X/Y)/Z: f	Y/Z: g	=>	X/Z: λxf(x)(g(x))
FSx:	(X/Y)\Z	Y\Z	=>	X\Z
BS:	Y\Z	(X\Y)\Z	=>	X\Z
BSx:	Y/Z	(X\Y)/Z	=>	X/Z

Figure 1.7
Functional Substitution

Grammar which permits a simplification in the number of combinators. It turns out that Application, Composition and Substitution, together with certain other potential combinators not discussed above, can be subsumed under one general combinatory schema.

```
1.27  FA:   NP
                   NP/N  the
      BA:          N
                         N ∈ articles
      FA:                N\N
                               (N\N)/(S/NP) ∈ which
      FC:                      S/NP
      FR:                            S/(S\NP)
                                           NP ∈ I
      FC:                            (S\NP)/NP
                                           (S\NP)/(S\NP) ∈ will
      BSx:                                 (S\NP)/NP
                                                 (S\NP)/NP ∈ file
      FC:                                        ((S\NP)\(S\NP))/NP
                                                  ((S\NP)\(S\NP))
                                                    /(S\NP) ∈ without
                                                  (S\NP)/NP
                                                    ∈ reading
```

This completes this review of some of the more important suggestions that have been made about combinators and type changing rules motivated from natural language data. The availability of these combinators, together with several not illustrated in this Section, raises the issue of the generative capacity of grammars with different sets of combinators. This is an area of active research which is briefly addressed in the next section.

1.5 A Hierarchy of Categorial Grammars

It was mentioned earlier that Categorial Grammar originated as one branch of formal logic and this logical perspective, which is in many ways quite distinct from the linguistic perspective, has carried over to the study of Generalised Categorial Grammars. Categorial grammar is a formal logic whose mathematical and metalogical properties merit attention in their own right. For example, the combinators may be viewed as axioms of a formal system and as such are subject to the same criteria of completeness and independence as the axioms of any formal logic. Although this is not the place for a review of the research programme currently under way into the logical properties of Generalised Categorial Grammar (for that, the reader is referred to Moortgat (1988b) for an excellent survey), one aspect of the formal properties of Generalised Categorial Grammar is relevant to later discussion: the hierarchy of categorial grammars induced by the introduction of new combinatory rules.

Moortgat (1988b, page 41) presents a hierarchy of grammars analogous to the well-known Chomsky hierarchy for re-writing systems and the languages they induce. Part of this is illustrated in Figure 1.9. The *locus classicus* of recent investigations into categorial logics is the Lambek calculus **L**. As mentioned earlier, the system **L** can be implemented in a variety of ways: one is via a definition of category structure and the specification of a set of combinatory rules similar but not identical to those given in the previous section; another (adopted in Moortgat (1988b)) presents **L** in the form of a sequent calculus in which the combinatory rules have the status of theorems (ie. valid inferences in the logic of the categorial combinators). For the purposes of this thesis, it is sufficient to take the former view and draw attention to the differences between **L** and the definition of Generalised Categorial Grammar given above as far as the combinators and category structures are concerned.

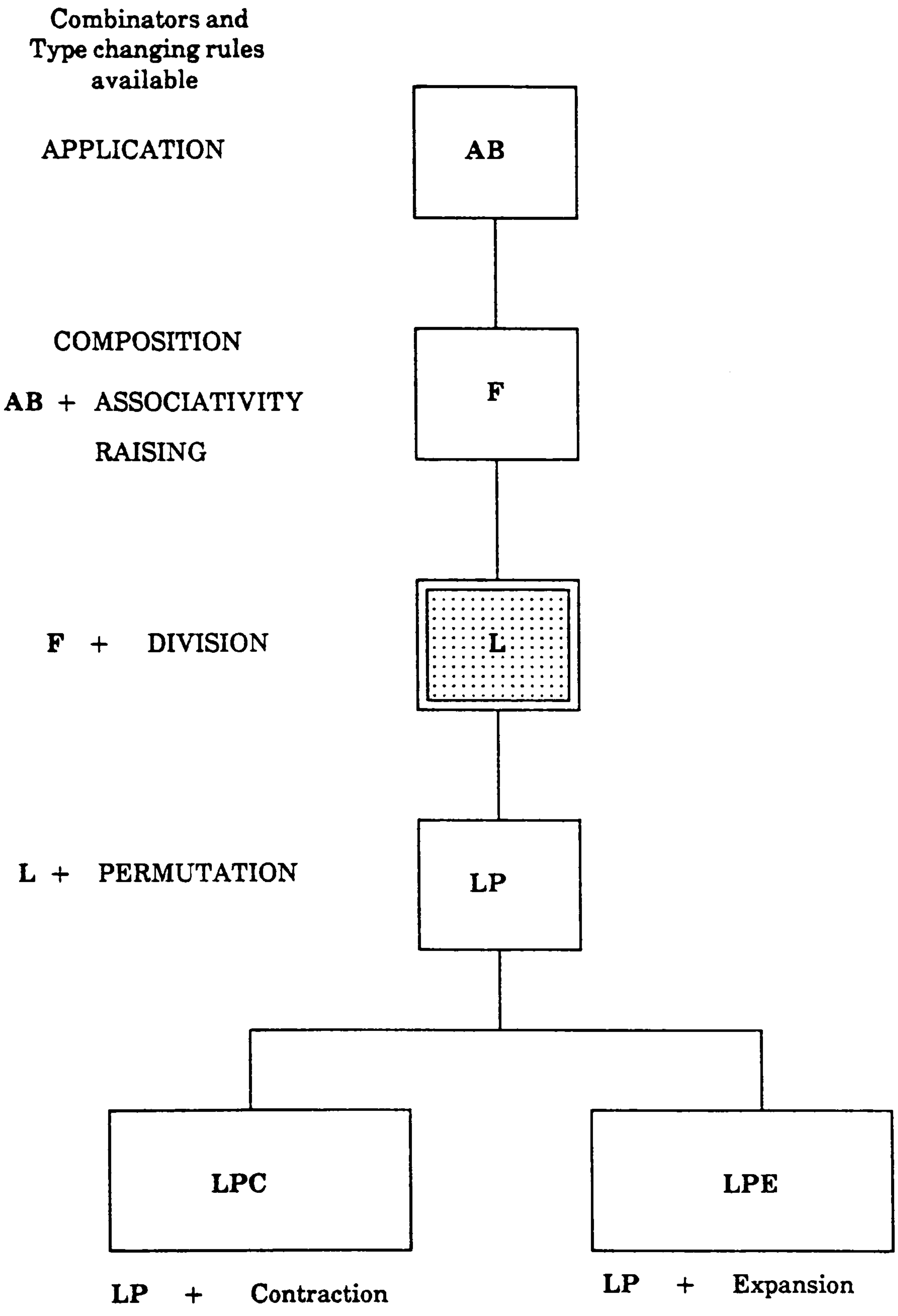


Figure 1.9
A Hierarchy of Categorial Grammars

The system L uses *three* primitive category connectives: the right and left directional slashes and a product connective. The right and left slashes have the interpretations already introduced: they form categories interpreted as functions. The product connective "•" is also a concatenative operator: an expression belongs to the product category A•B if it is the linear concatenation of category A and category B. The system introduced in previous sections is sometimes referred to as a product-free categorial calculus. The combinators of **L** include those discussed already together with two rules of *Division* which are summarised in 1.29 and 1.30 below. These will not be discussed in detail here since they play no direct part in the categorial models developed in this thesis, but for a discussion of the linguistic and logical aspects of Division see Moortgat (1988b, Chapter 1).

1.29 Division (Main function);

a $X/Y: f => (X/Z)/(Y/Z): \lambda x_1 \lambda x_2 f(x_1(x_2))$

b $X\backslash Y: f => (X\backslash Z)\backslash(Y\backslash Z): \lambda x_1 \lambda x_2 f(x_1(x_2))$

1.30 Division (Subordinate Functor):

a $X/Y: f => (Z/Y)\backslash(Z/X): \lambda x_1 \lambda x_2 (x_1(f(x_2)))$

b $X\backslash Y: f => (Z\backslash Y)/(Z\backslash X): \lambda x_1 \lambda x_2 (x_1(f(x_2)))$

The two systems **AB** and **F** which are weaker than **L** are, respectively, the classical Applicative system of Ajdukiewicz (Ajdukiewicz (1935)) and Cohen's Free Categorial Grammar (Cohen (1967)). **AB** has only Functional Application as its sole combinator and **F** has, in addition, Composition, Associativity and Type Raising. **L** is the system **F** plus the two versions of the Division Rule.

From a linguistic point of view, none of the systems **AB**, **F** or **L** are adequate for describing the full range of structural relations found in natural languages. The limitations of **L** (and, consequently, also of the weaker systems) as a model for natural language description include the following: (1) it is order preserving in the strict sense that although the associativity property allows completely free bracketing of any string (hence providing for strictly left-to-right incremental parsing and non-constituent coordination), no permutations of input are allowed; (2) there is no copying of functions or arguments. This is related to the count invariance property of **L** (van Benthem (1986)) which ensures a balance between the numbers of directional slashes on either side of a single derivation step; (3) there is no deletion or omission of types in a derivation step. It will be clear why these are limitations. It is an easy matter to cite linguistic data which *prima facie* require the use of permutation, copying and deletion. The data in Chapters 2 and 8 contain numerous such examples.

FP, **LPC**, **LPE** and **LPCE** are all augmentations of **L** with respectively, Permutation, Permutation + Contraction (deletion), Permutation + Expansion (copying) and, lastly, all three operations, the last being the most powerful of the systems. The problem with **LP** and the stronger categorial systems as models for natural language description is that they are much too liberal in allowing permutation, deletion and copying. The typical situation in natural languages is that certain permutations, for instance, are allowed but not all. One of the primary goals of Categorial Grammar research in recent years has been the investigation of the categorial and logical space between **L** and **LP**. The work reported in this thesis is offered as a contribution to this programme, driven primarily by a consideration of the complexities exhibited by semi-free word order languages.

1.6 Notes

[1] A *monostratal* theory refers to only a single level of representation. For a brief defence of monostratal over multistratal syntactic descriptions, see Gazdar, Klein, Pullum and Sag (1985, pages 10-11). The claim that Categorial Grammar uses only one level of representation is, of course, quite compatible with the metagrammar/object grammar distinction which plays such an important role both in Generalised Phrase Structure Grammar and the model of Categorial Grammar presented in this thesis.

[2] Although this is the conventional way of giving a formal calculus, a more recent interpretation, originating with Lambek (1958, 1988) and explored in Moortgat (1988b), takes "the categorial reduction system as a calculus analogous to the implicational fragment of the propositional logic sequent calculus... The sequent perspective is a particularly lucid basis for the discussion of the central notions of derivability and decidability" (Moortgat 1988b, page 27). For the exploration of word order in Categorial Grammar, where the crucial issues relate to category structure and inheritance of information between categories, the conventional representation of the calculus given here is

adequate.

[3] "An early result by Schönfinkel (1924) and, independently, by Curry, showed that it is always possible to take an n-ary function and break it down into a series of 1-ary functions.... Conversely, any complex function of the curried sort can be 'decurried'. The algebraic basis of this result is the fact that there is an isomorphism between the two function sets:

$$S^{Y \times Z} \simeq (S^{Y})^{Z}$$

given by associating a function f in the first set with a function F in the second just in case $f(y,z) = [F(z)](y)$... The linguistic consequence of this result is that theories based on the theory of functions can make use either of many-place or one-place functions (analogous to ideas about *flat* versus *hierarchical* structures in phrase-structure grammars)." (Oehrle, Bach and Wheeler (1988, pages 5-6). The relevance of this result to the theory of word order in Categorial Grammar presented in this thesis will become apparent in later chapters.

[4] For a general introduction to this approach to Montague Semantics, see Dowty, Wall and Peters (1981). For a review of the relevance of Montague's programme to Linguistics see Halvorsen and Ladusaw (1979).

[5] See Steedman (1988) for an alternative using a variable-free semantics based on the combinatory logic of Curry and Feys (1958).

[6] The limitations of this paragraph as a resumé of coordination in Generalised Categorial Grammar will be obvious. The category of conjuncts, for example, is highly problematic. For recent accounts see Moortgat (1988b, page 15), Steedman (1987b) and Wood (1988). It is not necessary in the context of the work reported in this thesis for a decision to be taken with respect to the proper theory of coordination in Categorial Grammar. The topic of coordination in relation to word ordering is discussed further in Section 4.2.

[7] Other conventions have been used in the Categorial Grammar literature. Bar-Hillel (1953) used non-directional slashes and associated directionality directly with the arguments of a function. An adaptation of this is used in Huck (1985). From the point of view of the ideas presented here, this is particularly important since it is assumed throughout that directionality is a feature associated with arguments in the domain of

a function.

[8] For an exchange between Lambek and Steedman on the relative merits of the two notations, see Moortgat, Oehrle and Wood (1987, 1988).

[9] Many of the sentences used for illustration in this Chapter bear a striking resemblance to those found in Steedman (1987b). This is not a coincidence.

[10] This is based on the sentence *Which articles did you file without reading* in Steedman (1987a), the wh-question has been suppressed as irrelevant for the purposes of this Chapter.

CHAPTER 2

SEMI-FREE WORD ORDER: SOME DATA FROM SPANISH

2.1 Introduction

Throughout most of this thesis Spanish is taken as a representative semi-free word order language. By way of introduction to this topic it will be useful to look briefly at two contrasting views of word order in Spanish and make clear the position adopted here. The following, from Ramsey (1956, page 662), a standard grammar of Spanish for English students, is representative of the first view:

> A sentence ... may be separated into sections according to meaning, and the order of the sections changed according to taste; but each section should remain unchanged:
>
> Tres-grandes-faltas-políticas | comitieron | los-árabes | al-llegar-a-nuestro-suelo
>
> (On arriving in our territory, the arabs committed three serious political mistakes)

The implication here is that all orderings of the four constituents are grammatical, the decision being free ("according to taste") or constrained by contextual and pragmatic factors.

The second view is represented here by the Spanish grammarian Gili Gaya

(1982). After noting that there are twenty four possible ordering combinations of subject, direct object, indirect object and finite di-transitive verb, Gili Gaya points out that twelve are clearly ungrammatical by the standards of modern usage, although examples may be found in poetic and "affected" styles of writing (*pueden hallarse en poesí o en estílo notoriamente afectado* (Gili Gaya 1982, page 88):

> En las doce el verbo ocupa el tercero o cuarto lugar, y la tendencia a la bipartición está visiblemente favorecida en ellas. En armonía con lo observado en las oraciones formadas por tres elementos, *el verbo no puede ir sin afectación más allá del segundo lugar.*
>
> (Gili Gaya (1982, page 88. My italics)
>
> (In the twelve [ungrammatical sequences] the verb occupies the third or fourth position and the tendency to divide into two [distinct intonational units] is very clear in them. In conformity with what is observed in sentences formed from three elements [simple transitives], the verb cannot without artificiality be further [to the right] than the second position.)

These views differ in a fundamental way: the former implies that *any* ordering of major constituents at the sentence level is grammatical in Spanish; the latter asserts explicitly that only some orderings are grammatical, that a clear demarcation can be made between those sequences of words or constituents which are grammatical and those which are not. It may be the case (and almost certainly is) that within the set of grammatical sequences there are different degrees of acceptability and markedness but the important point is the partition into two sets.

It is one of the fundamental assumptions of the work reported in this thesis that the second of these views more accurately reflects the situation in modern Spanish. All the native speaker informants who have contributed to the work reported here support the general opinion that the sequences of constituents at the sentence level can be partitioned into grammatical and ungrammatical sentences. Not all speakers agree on the contents of these sets and that poses problems for analysis but there is a core of sequences that all native speakers seem agreed upon. A difference will emerge, however, between the data presented in this Chapter and the judgements given by Gili Gaya. In particular, there seems to be strong evidence from di-transitive sentences to reject the claim that the verb can only occur as first or second constituent.

A language which displays some degree of freedom for some of its major constituents is referred to here as displaying *semi-free word order*. The intended contrasts are on the one hand with languages which show little or no flexibility with respect to linear ordering of components of major constituents and on the other hand with languages which display virtually complete freedom of ordering of components of major constituents. This topic has been of considerable interest in syntax for a number of years. Examples of rigid word order languages include English which offers very little freedom in the ordering of components of major constituents and even makes use of syntactic devices to maintain the basic underlying word order in different sentence types such as the use of *do-support* in formation of yes-no questions. The definition of mixed word order above, which is intended to exclude English as a mixed word order language, does not preclude the possibility of some degree of freedom with respect to the ordering of certain syntactic elements. So, for example, English displays some freedom with respect to the ordering of verbal particles with certain verbs and even more freedom with respect to the ordering of certain adverbials (eg. sentence adverbs) but the obligatory components of major constituents systematically exhibit fixed word order.

There are numerous recorded cases of free word order languages from Australian aboriginal languages to classical Latin and modern Finnish. A frequent observation regarding free word order languages is that the syntactic freedom correlates with the use of case marking. Spanish is an interesting language in this respect since it is mixed word order with complex ordering possibilities at the sentence level yet it lacks the fully explicit case marking found in, say, classical Latin and modern Finnish.

The purpose of this chapter is to provide some basic data against which to test the theory of word order in Categorial Grammar developed in Chapter 4. More detailed data will be discussed in Chapters 5 and 8. These data concern the position of the subject in simple transitive and di-transitive sentences, the fronting of direct and indirect objects and the ordering of the subject relative to sequences of auxiliary verbs. The chapter is divided into the following sections. Firstly, some basic information about Spanish sentences is given illustrating phenomena such as the optionality of the subject and the occurrence of lexical noun phrases and clitic object pronouns in the same sentence, a topic which will be developed further in Chapter 8. Secondly, detailed accounts are given of the orderings of subject and direct object in transitive sentences. Not all informants agree on all aspects of the data and, where relevant, areas of disagreement are highlighted. Finally, the chapter concludes with tabular summaries of the grammaticality patterns for some important sentence types which will be referred to in later chapters.

2.2 Some Basic Facts about Spanish Sentences

Consider the sentence in 2.1.

2.1	Juan	le	dio	un	libro	a	Pedro
	John	[dat]	gave	a	book	to	Peter

(John gave a book to Peter)

This sentence illustrates a number of elementary but important facts about Spanish sentences which will be relevant to later discussion.

Firstly, the subject is present lexically and has certain of its syntactic features morphologically marked on the verb. In other words, there is agreement between the verb and the subject with respect to features of number and person. The morphological marking of these features is richer in Spanish than in English.

Secondly, the indirect object is both lexically present and marked as a dative clitic - *le* - associated with the verb. The grammatical status of this clitic is a topic of some controversy. One important syntactic feature of this clitic is that it is optional. A few informants accept 2.1 without the clitic; most do not. All agree, however, that with the verb *dar* the sentence is better with the clitic. Other apparently di-transitive verbs (eg. *mandar*, to send) do not have such a strong requirement for the doubled dative clitic.

Thirdly, in the case of both the subject and the indirect object, the lexical realisation may be omitted from the sentence without causing ungrammaticality. The following are, therefore, grammatical sentences:

2.2	a le dio un libro a Pedro (he gave a book to Peter)	(omitted lexical subject)
	b Juan le dio un libro (John gave him a book)	(omitted lexical indirect object
	c le dio un libro (he gave him a book)	(omitted lexical subject and indirect object)

Fourthly, the lexical direct object in 2.1 is not duplicated morphologically or by means of a clitic. An accusative clitic can take the place of the lexical direct object, but both cannot usually be present, hence the following grammaticality judgements:

2.3	a Juan	se	lo	dio	a	Pedro		
	John	[dat]	[acc]	gave	to	Peter		
	(John gave it to Peter)							
	b *Juan	se	**lo**	dio	**un**	**libro**	a	Pedro
	John	[dat]	[acc]	gave	a	book	to	Peter
	(John gave a book to Peter)							

It will be one of the arguments of this thesis that these facts relate closely to the constituent ordering freedom characteristic of Spanish sentences. The finite verb occupies a central position in the organisation of the sentence, a fact which Categorial Grammar is particularly well suited to represent.

2.3 The Position of the Subject in Various Sentence Types

This section presents data on the ordering of the subject for various sentence types with respect to the verb and its subcategorised-for constituents. Consider the sentences in 2.4 and 2.5 in which the subject NPs are in bold font.

2.4 a **Juan** compró una casa

b Compró **Juan** una casa

c Compró una casa **Juan**
(John bought a house)

2.5 a **Juan** le dio un libro a María

b le dio **Juan** un libro a María

c le dio un libro **Juan** a María

d le dio un libro a María **Juan**
(John gave a book to Mary)

These grammatical sequences suggest the following generalisation: subjects can occur in any inter-constituent position within sentences, including the first and last position in the sentence. This is a generalisation which must be modified a little in the light of data such as the following however:

2.6 a **Juan** dijo que Pedro compró una casa

b Dijo **Juan** que Pedro compró una casa

c *Dijo que **Juan** Pedro compró una casa

d *Dijo que Pedro compró **Juan** una casa

e Dijo que Pedro compró una casa **Juan**
(John said that Peter bought a house)

The subject of the top level (matrix) verb cannot occur within an embedded sentence, although it can occur at the end of the sentence, that is, after a sentential (or infinitival) complement.

The correct generalisation is, therefore:

A subject can occur in any inter-constituent position within its own sentence.

This will be important later, but at the moment we might note in addition that sentences are usually better from a stylistic point of view when the subject is close to its verb, so that, for example, 2.6e is less good than 2.6b. This does not affect grammaticality, however.

An apparent counter-example to this generalisation is provided by subject control verbs such as *querer* (to want), the data for which are given in 2.7. 2.7c appears to have its subject *within* the "embedded" infinitival complement. It will be argued in Chapter 5 that the control verb behaves rather like an auxiliary by forming a verbal complex with its infinitival complement. The result is that the orderings found in 2.7 resemble those of a simple transitive sentence. The ordering in 2.7b arises given the linear ordering properties of the control verb itself, a topic taken up in more detail in Chapter 5.

2.7 a **Juan** quiere cantar la canción

b quiere **Juan** cantar la canción

c quiere cantar **Juan** la canción

d quiere cantar la canción **Juan**
(John wants to sing the song)

2.4 The Direct Object

Two aspects of the grammar of direct objects are relevant to the present discussion:

1. the occurrence of the particle *a* with certain direct objects;

2. the capacity of direct objects to be fronted before the verb.

2.4.1 The Marking of Direct Objects

The particle *a* is used to mark certain direct objects and most indirect objects in Spanish as well as being a true preposition. Consequently, there is room for different interpretations of the categorial status and syntactic function of this particle with respect to objects. This is an important issue but one on which there is no general consensus. In this section, examples of the use of *a* with direct objects are given; in Section 2.5 a more detailed examination of the status of this particle with respect to indirect objects is undertaken. First, the conventional wisdom. Ramsey (1956, page 38) has the following:

> The chief device in Spanish for distinguishing a noun as direct object (accusative) is by placing the preposition "a" before it. But as this preposition is the regular sign of the indirect object (dative), its application to direct objects vacillates between an endeavour on the one hand to prevent the noun from being mistaken for the subject, if the preposition were omitted, and on the other, to prevent its being mistaken for the indirect object, if the preposition were employed.

While the essential point of this quotation is certainly true (there is a tendency towards ambiguity which is avoided by using *a*), the assumption that the *a* which marks direct and indirect objects is a preposition would not be generally accepted today.

A few examples of the use of *a* with direct objects follow.

It applies primarily to nouns representing determinate, known persons or to things personified (Ramsey 1956, page 38). The use of *a* with personified direct objects will not be considered here, but an instance of the former is:

2.8 Oía detrás de él **a sus persiguidores**
(he heard his pursuers behind him)

Proper names usually denote determinate, known persons and so the use of *a* extends to names as direct objects. Compare 2.9a and 2.9b.

2.9 a Admiro mucho **a Napoleón**
(I admire Napoleon a great deal)

b *Admiro mucho **Napoleón**

A can be omitted, however, in cases where the direct object denotes a person or persons preceded by a numeral. Contrast 2.10a with 2.10b.

2.10 a El general derrotó **trescientos enemigos**
(the general defeated three hundred enemies)

b El general derrotó **a los rebeldes**
(the general defeated the rebels)

Finally, before direct objects beginning with the indefinite article there is considerable variation in usage. The following judgement is attested in Ramsey (1956, page 41):

2.11 Barraba **vio un hombre** que se movía a pie en el campo,
(Barraba saw a man moving, on foot, out in the country,

cargando con un bulto voluminoso.
loaded with a bulky object)

which can be contrasted with 2.12 (Ramsey 1956, page 41):

2.12 Por la ventana **vio a un hombre** que corría
(through the window he saw a man running

a campo traviesa.
across country)

In both 2.11 and 2.12 **un hombre** bears the same grammatical relation to its verb (direct object). There is, however, some tension between the requirement that determinate, animate objects take the particle *a* and the fact that in these instances the objects are syntactically indefinite. This tension is resolved differently in each case depending on the perception and intention of the speaker.

Suñer (1988, pages 425-431), offers a brief but very clear summary of the main issues with regard to the categorial status of this particle. As the examples above show, it does not affect the categorial status of the Noun Phrase. Suñer points out that "although its exact usages have not received a completely satisfactory analysis, the specificity or individuation and animacy of the DO [direct object] plays a role in its presence with nonquantified Noun Phrases [2.13], and with overt personal pronouns [2.14]. However, specificity is not relevant with quantified Noun Phrases [2.15], since the subjunctive relative clause in [2.15b] clearly indicates that its indefinite antecedent is nonspecific" (page 426).

2.13 a Oían a Paca a la niña a una niña
(they heard Paca the girl a girl

*a niñas a la gata *a la radio
girls the cat the radio)

b La anciana amaba los niños (generic reading)
(the old woman loved (the) children)

c Busca a una estudiante que traduce japonés
(S/he is looking for a student who translates Japanese)

2.14 Nos querían sólo a nosotros
(they loved only us)

2.15 a No vieron a ninguna persona
(they didn't see anybody)

b Buscaban a alguien que los ayudara
(they were looking for somebody who could help them)

In the face of these data, Suñer argues that the *a* particle with direct objects is "primarily a marker of animacy" (page 426). This position will be adopted here.

2.4.2 The Fronting of Direct Objects

Data on the fronting of direct objects have proved to be one of the most difficult aspects of word order to clarify because of the lack of stability and consistency in native speaker informant judgements. Basically, there appears to be a range of possibilities involving whether or not the direct object is syntactically definite or indefinite and whether or not it is marked with 'a' (as in the case of animate direct objects). Some informants are reluctant to accept *any* fronted direct objects when presented with them in isolated sentences, although most accept certain fronted objects given a sufficiently specific discourse context. In at least one instance, this reluctance to accept fronted direct objects could be attributed to influence from Catalan, a language with a more rigid word order than Spanish. It is also possible that Gili Gaya's insistence that the Spanish finite verb can only appear in first or second place in a sentence is due to influence from Catalan. Castillian speakers accept much greater freedom. For all informants, fronted direct objects constitute a marked word order which in some cases required very specific context for acceptability. This was true even of informants who quite freely accepted fronted direct objects.

There appears to be a scale of possibilities such as the following:

2.16	a Indefinite inanimates:	**un coche** compró Juan (John bought a car)
	b Definite inanimates:	**el coche** compró Juan (John bought the car)
	c Direct Objects marked with 'a':	**A la mujer** vio Juan (John saw the woman)

2.16a is the most acceptable; 2.16c the least acceptable.

In the face of this problem it is necessary to take a position even though the decision may not be acceptable to all native speakers. Here, it is assumed that all direct objects can be fronted but that discourse context disallows certain possibilities in just the same way that it disallows certain other word orders. Chapter 5 develops this analysis in detail.

One ordering possibility is universally accepted as ungrammatical: the occurrence of both the subject and direct object to the left of the verb for simple transitive sentences.

2.5 Indirect Objects

As with direct objects, two aspects of the grammar of indirect objects are of importance here: the status of the particle *a* which accompanies virtually all indirect objects, and the word order possibilities they exhibit. Each of these will be discussed in turn.

2.5.1 The Particle *a*: Preposition or Case Marker?

The discussion and example data in this section are heavily dependent on the presentation in Suñer (1988). The conclusion reached there that when it is associated with indirect objects, the particle *a* is a case marker and that consequently indirect objects are Noun Phrases rather than Prepositional Phrases is consonant with the categorial analysis of indirect objects developed in Chapter 8.

As Suñer points out (1988, page 425), there is no question that *a* can function as a true preposition in Spanish. For instance, in 2.17 *a* patterns with other true prepositions, in 2.18 it introduces an idiomatic expression and in 2.19 it is selected by the verb.

2.17 Lía se sentó a (delante, sobre, bajo) la mesa
Lía sat at (in front of, on, under) the table

2.18 Lo mataron a sangre fría
(they killed him in cold blood)

2.19 Asistió **al** (= a + el) simposio
(S/he attended the symposium)

The *a* which is found with indirect objects has none of these properties. Furthermore, there are distinct differences in the syntactic properties of prepositional phrases with *a* and indirect objects. Suñer considers three arguments for this.

Firstly, "Spanish PPs may behave as governing categories" (page 427) with the result that a pronominal form within the PP may refer back to the clausal subject, as in 2.20, where the co-indexing indicates that the pronominal must be free in its governing category.

2.20 a Ese señor$_i$ nunca se cansa de hablar [$_{pp}$ de él$_{i/j}$]
(that man never tires himself of speaking about himself/him)

b Es una pena, Pilar$_i$ sólo piensa [$_{pp}$ en ella$_{i/j}$] primero
(it is a pity, Pilar only thinks about herself/her first)

In contrast to this, indirect objects must be disjoint in reference with respect to the subject, as indicated by the co-indexing in 2.21 below.

2.21 a Mara$_i$ le$_j$ aceptó la invitación [a ella$_{*i/j}$]
(Mara accepted the invitation from her)

b Pepe$_i$ le$_j$ dará una fiesta [a él$_{*i/j}$]
(Pepe will give a party for him)

Compare the two sentences in 2.21 with those in 2.22 where reference is not required to be disjoint.

2.22 a Mara$_i$ aceptó la invitación para ella$_{i/j}$ sola
(Mara accepted the invitation for her(self alone)

b Pepe$_i$ dará una fiesta para él$_{i/j}$
(Pepe will give a party for him(self)

As Suñer says: "If IOs [indirect objects] were PPs, the pronouns should be able to co-refer with the subjects (at least in certain contexts, or with certain Vs), just as in [2.20]. That they never do argues against considering them PPs. These dissimilar patterns cannot be explained in terms of subcategorised arguments, since the bracketed PPs in [2.20] as well as the IOs in [2.22] are subcategorised" (page 428).

Secondly, there is a difference in "anaphora binding within the VP" (page 428) between direct and indirect objects on the one hand and true PPs on the other. Consider the sentences in 2.23 to 2.25 below, where 2.23 shows the co-indexing that obtains with direct objects, 2.24 the co-indexing with indirect objects, and 2.25 the co-indexing with true PPs.

2.23 El profesor (los) libró **a los estudiantes**$_j$ a sí mismos$_j$
(the professor left the students to themselves)

2.24 a Paco$_i$ (le) habló **al profesor**$_j$ de sí mismo$_{i/j}$
(Paco talked to the professor about himself)

b El padre le contó **a la niña**$_j$ de sí misma$_j$
(the father told the girl about herself)

2.25 Paco$_i$ habló **con el profesor**$_j$ sobre/de sí mismo$_{i/*j}$
(Paco talked with the professor about himself)

Direct and indirect objects may serve as binders for a VP-anaphor, while a true PP cannot (Suñer 1988, page 429). "This implies that direct and indirect objects are Noun Phrases (despite being preceded by *a*)".

Thirdly, indirect objects are "semantically unrestricted" (page 429): "It is well-known that the theta-role of a Noun Phrase governed by a preposition is determined by this preposition. In English, for example, *to* and *for* signal Goals, while *from* indicates a Source. On the other hand, prepositionless arguments such as subjects and objects are not so restricted. Thus we expect that, if IOs are Noun Phrases, the IO *a*-phrase should be semantically unrestricted". Consider the semantic roles of the bracketed phrases in 2.26.

2.26 a Entregó las flores [**a** la mucama]$_{\text{Goal}}$
(S/he gave the flowers to the maid)

b Los niños compraron chucherías [**de** la vendedora]$_{\text{Source}}$
(the children bought trifles from the vendor)

c Conseguí un empleo [**para** María]$_{\text{Goal}}$
(I obtained a job for Mary)

d Arreglaron la televisión [**de** mi madre]$_{\text{possessor}}$
(They fixed the television set for my mother)

e Lavó el auto [**por** ella]$_{\text{Beneficiary}}$
(S/he washed the car for her (= for her sake/instead of her)

All the sentences in 2.26 may be expressed with the alternative construction **le(s)....a NP** and still be interpreted with the same theta roles, as shown in 2.27.

2.27 a **Le** entregó las flores **a** la mucama

b Los niños **le** compraron chucherías **a** la vendedora

c **Le** conseguí un empleo **a** María

d **Le** arreglaron la televisión a mi madre

e **Le** Lavó el auto **a** ella

The point is not whether prepositions can specify more than one theta-role - they clearly can, as indicated by **de** in 2.26b and 2.26d - but whether by changing the preposition the meaning is affected. In this respect, the *a*-phrases in 2.27 behave very differently from true prepositions (Suñer 1988, page 430): "That it [the *a*-phrase] may be read as *Goal, Source, Possessor*, or *Beneficiary* demonstrates its semantic unrestrictedness. This suggests that these phrases are Noun Phrases and that the *a* is a dummy whose function is that of surface Case marker/spell out" (page 430). As in the case of direct objects, this position is adopted here. One consequence not commented upon by Suñer is the fact that both direct and indirect objects are categorially identical: they are both Noun Phrases.

2.5.2 The Ordering of Indirect Objects Relative to the Verb and its Arguments

The position of the indirect object is very flexible in Spanish but, as with subjects and direct objects, there are limitations on the positions that they can occupy in sentences. This section presents data illustrating the most important possibilities.

In subject-initial sentences which have only the subject before the verb, the direct and indirect objects can occur in either order after the verb. This is illustrated in 2.28 where the indirect object is in bold font.

2.28 a Juan le dío un libro a **María**

b Juan le dío **a María** un libro
(John gave a book to Mary)

It has already been demonstrated that in verb-initial transitive sentences there are no syntactic restrictions on the relative positions of subject and other arguments. This is also true of indirect objects in verb initial di-transitive sentences, relative to the subject and direct object. In other words, all the sequences in 2.29 (where S is the subject, V the finite verb, O the direct object and I the indirect object) are grammatical and constitute paraphrases of each other.

2.29	a Le dio Juan un libro a **María**	V S O I
	b Le dio Juan a **María** un libro	V S I O
	c Le dio un libro Juan a **María**	V O S I
	d Le dio un libro a **María** Juan	V O I S
	e Le dio a **María** un libro Juan	V I O S
	f Le dio a **María** Juan un libro (John gave Mary a book.)	V I S O

Furthermore, indirect objects can be fronted in the sentence quite freely, giving the grammatical sequences in 2.30 below:

2.30 a A **María** le dio Juan un libro

b A **María** le dio un libro Juan
(To Mary, John gave a book.)

Note that these sentences do not have "comma intonation" in the form of a pause after the fronted indirect object although they are emphatic on the indirect object.

There is considerable native speaker variation when considering the fronting of indirect objects together with other arguments of di-transitive verbs. For example, data from one informant indicates that although both the subject and indirect object can occur together before the verb, as in 2.31a, a fronted indirect object cannot occur with a fronted direct object. This is illustrated in 2.31b.

2.31 a Juan a **María** le dío un libro

b ?a **María** un libro le dío Juan

On the other hand, some informants accept *any* two arguments of a di-transitive verb before it. Again, as with the fronting of direct objects, a position has to be taken which will not correspond to all native speaker judgements. Here, the more liberal data indicating that any two arguments may occur before di-transitive verbs will be accepted.

Indirect objects can be extracted from infinitival complements and modal sentences but only as topicalisations, hence the judgements in 2.32.

2.32 a **A Pedro**, Juan quiere darle el libro
(to Peter, John wants to give the book)

b **A Pedro**, Juan debe darle el libro
(to Peter, John must give the book)

However, the indirect object cannot be fronted *within* the infinitival or modal sentence, as shown in 2.33.

2.33 a *Juan quiere, **a Pedro** darle el libro

b *Juan debe, **a Pedro** darle el libro

The situation with sentential complements is the reverse of this. Indirect objects cannot be extracted from embedded sentences:

2.34 a ***A María** dice Juan que le dío Pedro un libro
(to Mary, John says that Peter gave a book)

b ***A María** dice Juan que quiere darle Pedro un libro
(to Mary, John says that Peter wants to give a book)

but the indirect object can be fronted within the embedded sentence:

2.35 creo que **a Pedro** le dio Juan el libro

2.6 Summary

The purpose of this Chapter has been to provide some "bench mark" data against which to evaluate proposals for describing word order in later chapters. Basically, it is concluded here that there are certain specific restrictions on the orderings of constituents in transitive and di-transitive sentences, but that within these constraints, there are other (non-syntactic) factors which influence acceptability. Both the syntactic constraints and the other other factors which affect word order are subject to the linear ordering analysis developed in later chapters.

The tables on the following three pages summarise the key data presented in this chapter. In all the tables abbreviations are used to indicate the subject (S), direct (O) and indirect (I) objects in addition to the finite verb (V). Constituents in curly brackets are to be understood as having no linear ordering constraints on them with respect to other constituents in the same brackets. They are not, however, indicated as being optional by this notation.

Transitive sentences	Di-transitive sentences
S V O	S [dat] V {O, I}
V {S, O}	O [dat] V {S, O}
O V S	I [dat] V {S, O}
	{S, I} [dat] V O
	{S, O } [dat] V I
	{I, O } [dat] V S

Table 2.1
Word Order in Spanish Transitive and Di-transitive Sentences

(John says that Mary bought a house)

S_{matrix} V_{matrix} [COMP S V O]	**Juan** dice [que María compró una casa]
V_{matrix} S_{matrix} [COMP S V O]	Dice **Juan** [que María compró una casa]
V_{matrix} [COMP S V O] S_{matrix}	Dice [que María compró una casa] **Juan**
*V_{matrix} [COMP S_{matrix} S V O]	*Dice [que **Juan** María compró una casa]
*V_{matrix} [COMP S V S_{matrix} O]	*Dice [que María compró **Juan** una casa]

Table 2.2
The Order of the Subject Relative to Verbs Taking Sentential Complements

haber

	(John has sung the song)
S Aux_{haber} V O	Juan ha cantado la canción
Aux_{haber} V S O	ha cantado Juan la canción
Aux_{haber} V O S	ha cantado la canción Juan
*Aux_{haber} S V O	*ha Juan cantado la canción
O Aux_{haber} V S	una canción ha cantado Juan
*O Aux_{haber} S V	*una canción ha Juan cantado

estar

	(John is singing the song)
S Aux_{estar} V O	Juan está cantando la canción
Aux_{estar} V S O	está cantando Juan la canción
Aux_{estar} V O S	está cantando la canción Juan
Aux_{estar} S V O	está Juan cantando la canción
O Aux_{estar} V S	una canción está cantando Juan
O Aux_{estar} S V	una canción está Juan cantando

Table 2.3
The Order of the Subject Relative to Verbs with Auxiliaries

(John wants to sing the song)

S $V_{control}$ V O	Juan quiere cantar la canción
$V_{control}$ S V O	quiere Juan cantar la canción
$V_{control}$ V S O	quiere cantar Juan la canción
$V_{control}$ V O S	quiere cantar la canción Juan
O $V_{control}$ V S	una canción quiere cantar Juan
*S $V_{control}$ O V	*Juan quiere una canción cantar
*$V_{control}$ O V S	*quiere una canción cantar Juan
*O $V_{control}$ S V	*una canción quiere Juan cantar

Table 2.4
Word Order with Subject Control Verbs

PART TWO

A THEORY OF WORD ORDER IN CATEGORIAL GRAMMAR

CHAPTER 3

LOCAL STRUCTURE, WORD ORDER AND CATEGORIAL GRAMMAR

3.1 Introduction

This Chapter presents a linear order factored Categorial Grammar in which it is possible to describe the complex word order patterns characteristic of semi-free word order languages. It was pointed out at the end of Chapter 1 that a Categorial Grammar with a rule of Permutation will accept all word orders; it is therefore of little interest from a linguistic point of view. One of the goals of research into Categorial Grammars is to find models which allow for the word ordering found in natural languages but do not have permutation closure over the categorial calculus. Most attempts to constrain Categorial Grammar in the recent literature have involved the introduction of general principles governing either the instantiation of directionality on the arguments of functions or the range of combinators permitted by the grammar. The approach adopted here is distinct from either of these but has more in common with the former. In order to set the work reported here in context, Section 3.2 reviews three recent approaches to word order in Categorial Grammar and indicates the ways they each constrain linear ordering.

Section 3.3 introduces the basic formal structures which underpin the theory of a linear order factored Categorial Grammar. Since some of the central ideas originated with Generalised Phrase Structure Grammar, it will be useful to summarise briefly the work on linear order which has been carried out within that paradigm. This is the topic of Section 3.3.1. The emphasis is on the *formal* structures which are used in Generalised Phrase Structure Grammar to model the factoring of linear order from immediate dominance information. Section 3.3.2 generalises three aspects of this formalism: the formal object representing local structural information (Immediate Dominance statements in Generalised Phrase Structure Grammar; categories in Categorial Grammar); the basic linear ordering statements themselves; and, the logic of complex linear ordering statements. In Section 3.3.3 Categorial Grammar categories are shown to be examples of local structures in the sense defined in 3.3.2.

In Section 3.4, a formal treatment of linear ordering in Categorial Grammar is presented. A logic for the statement of linear ordering constraints (L_p) is introduced which takes as its model the local structures defined by the grammar. Finally, in Section 3.4.2 the mapping from local category structures associated with lexical items to fully instantiated directional categories is spelled out in detail.

3.2 Previous Approaches to Word Order in Categorial Grammar

An interest in word order has characterised work in Categorial Grammar from the earliest days of its application to natural languages (Bar-Hillel (1953), Lambek (1958)). This interest has tended to focus on inter-linguistic variation and typological classification. Categorial Grammar is an interesting model for the comparison of word order variation across languages. However, *prima facie*, Categorial Grammar is not well suited for handling the observed complexities of language-internal word order variation since, in its simplest form at least, those lexical items which are represented as functions take arguments whose directionality markings are fixed. Since the primary aim of the work reported in this thesis is to develop a model of Categorial Grammar in which it *is* possible to describe complex language-internal word ordering variation, it will be helpful to place the work reported here in the context of other recent research - both inter- and intra-linguistic - in order to distinguish the approach taken here.

Three distinct approaches will be taken to represent the range of recent discussion of this issue. The first, Flynn (1983), addresses the problem of motivating canonical word order and showing how the known typological correlates of that word order fall out from the statement of very general principles. Alternatively, empirical linguistic evidence may be used to suggest appropriate constraints on the combinatorial properties of the Categorial Grammar. Steedman (1987a) adopts this approach, which essentially involves constraining the grammar on two fronts: using the fewest combinator types that are compatible with the data and constraining the directionality markings on the input and output categories of complex combinators such as Functional Composition and Substitution. Finally, Moortgat (1988b) is taken to represent the approach to the problem of making Categorial Grammar sensitive to intra-linguistic word order variation.

Flynn (1983) assumes an applicative categorial system using only Functional Application (**AB** in Moortgat's hierarchy). In such a grammar, the only feature available for constraining word order is the directionality marker associated with the arguments of functions. Flynn derives a canonical word order and its typological correlates from basic ordering conventions for each language which have the following general form: if there is a non-directional category assignment in the lexicon X|Y which has certain specified properties or is of a particular type, then the category is X\Y, else it is X/Y. The typological correlates follow given reasonable assumptions about category assignments in the lexicon.

One example will be sufficient to illustrate this methodology. Flynn gives the following definition for the (universal) category set in Categorial Grammar (Flynn 1983, page 142):

> Let e and t be two fixed objects. The set of categories is the smallest set CAT such that
>
> 1. e is in CAT
> 2. t is in CAT
> 3. whenever W,Y are in CAT, $\frac{W}{Y}$ is in CAT
> 4. whenever $\frac{W}{Y}$ is in CAT, $\frac{W}{Y}\alpha$ is in CAT, where α is N, A, or V.

On the face of it, this is a different definition of the set of categories to that given in Chapter 1. However, the $\frac{W}{Y}$ notation may be read as a category *without* directionality marking and clause 4 is a way of distinguishing different lexical classes with the same categorial type, on the model of the double slash notation in Montague (1973). A *major category* is defined as any category whose resultant category is t (page 145) and with this definition in mind, Flynn gives the following Word Order Convention for English:

If some phrase φ is of category $\frac{W}{Y}$ and φ contains an expression assigned to a major category, then $\frac{W}{Y}$ is to be interpreted as Y\W. Otherwise, $\frac{W}{Y}$ is to be interpreted as W/Y.

(Flynn 1983, page 145)

Notice that this convention uses the Lambek convention for the directionality of categories. From this Convention it is possible to show that many of the typological characteristics of English as an SVO language follow given expected assignments of categories to lexical items. The details of the argumentation are not of concern here, but the consequences are summarised in 3.1 (Flynn 1983, page 157):

3.1
a determiners precede nouns
b verbs precede their complements
c subjects precede the verb phrase
d nouns precede their complements
e English is prepositional
f prepositional phrases follow the phrases they modify
g adjective phrases that do not contain major categories precede the noun
h adjective phrases that contain major categories follow the noun
i relative clauses follow the noun they modify
j English has a leftward COMP

Flynn goes on to show that different word order conventions for typologically distinct languages such as Hopi (SOV) and Malagay (VOS) carry equally effective predictive power.

Despite the impressive range of predictions which Flynn is able to demonstrate follow from the general ordering conventions for each language, there are limitations in this approach for the analysis of semi-free word order such as that exhibited by the Spanish data in Chapter 2. They are the limitations inherent in all typological theories: they assume a single, basic word order for each language.[1] To allow for semi-free word order would require a theory of markedness to show explicitly how the non-basic word orders are made available and even if such a theory were available, there remains the important problem of distinguishing grammatical from ungrammatical sequences.

Steedman, in a number of papers (1985, 1987a, 1987b, 1988), has argued for the relevance of Categorial Grammar to the analysis of a wide range of natural language phenomena. The presentation of Generalised Categorial Grammar in Chapter 1 was heavily influenced by these papers. The introduction of new combinators raises two issues: what range of combinators is required and which versions of these combinators, since each can be instantiated in numerous ways depending on the choice of directionality marking. Take Functional Composition as an example. The following version is logically possible and there is nothing in the definition of Categorial Grammar given in Chapter 1 which excludes it:

3.2 X/Y Y/Z => X\Z

In this case, the result of the composition has a different directionality to either the main or argument functors. There seems, however, to be no linguistic motivation for such a rule. What is required is a set of general constraints which induce the orderings motivated by the data. Steedman (1987a, page 407) has summarised his findings on this in a number of putatively universal principles which constrain the directionality of categories in combinatory and type changing rules. These are given in Figure 3.1.

The Principle of Adjacency may be regarded as a general programmatic or ideological principle guiding the development of Categorial Grammars. The key terms are *realised* and *adjacent*. Steedman, along with most researchers in

The Principle of Adjacency: Combinatory rules may only apply to entities which are phonetically realised and adjacent.

The Principle of Directional Consistency: All syntactic combinatory rules must be consistent with the direction of the principal function.

The 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).

Figure 3.1
Universal Ordering Principles of Generalised Categorial Grammar

Categorial Grammar, claims that descriptively and explanatorily adequate grammars can be written for complex natural language data without recourse to traces, empty categories or other phonetically unrealised grammatical elements. This contrasts sharply with other current theories such as Government and Binding which freely employ syntactically and semantically complex but phonetically null elements. Much can be said, of course, about information inheritance in the Categorial lexicon, one aspect of which is inheritance of directionality information. This topic will be discussed in more detail in Chapter 6.

Moortgat (1988b, Chapter 3) is distinguished from the previous two approaches because he directly addresses the problem of intra-linguistic word order variation. Essentially, this is a formal problem of weak generative capacity: discontinuities demand a model which is more powerful than the classical system **L** but weaker than **LP**. Consequently, an investigation is required of the theoretical space between **L** and **LP** in order to find "a calculus which is stronger than **L** in allowing empirically motivated discontinuities, but weaker than **LP**, ie. a system which retains part of the order-preserving quality of the **L**-valid type transitions" (page 81). He considers two strategies for this.

> The first strategy consists in adding specific theorems from **LP** (or the stronger systems **LPC/LPE**) as extra *axioms* for L-derivability. We show that this form of extension forces one to give up the notion of a free syntactic algebra, which is such an attractive feature of pure **L**. We demonstrate that the extra axioms motivated by discontinuities make **L** collapse into **LP**, given a transitive notion of derivability. In order to avoid degeneration into **LP**, the type transitions borrowed from **LP**, instead of being universally quantified axiom schemes, must take the form of schemes with (in)equality constraints on the type parameters. We therefore limit the use of type-restricted axiom extensions to phenomena that can be analysed in terms of lexically governed unary type transitions, in conformity with the view that lexical type assignment is the only locus for stipulation.
>
> (Moortgat 1988b, page 81)

The second strategy recasts earlier proposals by Bach (1984) and others for the use of non-concatenative or wrapping operations in syntax in Categorial Grammar terms since "Discontinuous dependencies suggest enrichment in the form of non-concatenative operations besides left- and right-division" (page 108). The two operations Moortgat adds to **L** are Extraction and Infixation. The informal definition of Extraction (symbolised $\uparrow$) is as follows:

> $C \uparrow A$ (read "C gap A") designates an incomplete expression that wants an argument of type A to form an expression of type C. But whereas fractional types C/A (or A\C [Moortgat uses the Lambek notation]) concatenate with their argument under adjacency, the type $C \uparrow A$ is assigned to incomplete expressions that have an argument expression of type A missing *somewhere*, not necessarily at the periphery, and that will yield an expression of type C in combination with such an argument.
>
> (Moortgat 1988b, page 110)

The definition of infixation (symbolised $\downarrow$) is as follows:

> A functor $C \downarrow B$ (read "C infix B") forms an expression of type C in

> combination with an argument expression of type B by being infixed *anywhere* within the sequence that makes up the argument type B.
>
> (Moortgat 1988b, page 110)

Two specialised forms of the general infix category are A>B and B<A, which are interpreted as right-infixation before the last, and left-infixation after the first element of the argument type, respectively. Moortgat demonstrates (pages 118-9) that the system **L** + {<, ↑ ,>} (referred to as **L**') induces just the four versions of Functional Composition (out of the sixteen logical possibilities) which are compatible with the data and therefore Steedman's principles fall out as theorems.

It seems clear that the first strategy that Moortgat considers is not appropriate for the analysis of the word order facts described in Chapter 2, since they are clearly syntactic and not lexically driven. The question is: How well suited is the second strategy to the Spanish data? Consider the following example of a word order problem from Flemish (Moortgat 1988b, pages 116-7).[2] The verb raising trigger, *wil* (indicated in bold in the examples), freely occupies any of the three positions indicated in the clause.

3.3 a (dat hij) haar van de stoel af **wil** duwen
that he her of the chair off wants push
(that he wants to push her off the chair)

b (dat hij) haar van de stoel **wil** af duwen

c (dat hij) haar **wil** van de stoel af duwen

Given the category assignments in 3.4 for the relevant words in 3.3

3.4	NP	PP	PRT	VP'>VP	((VP\NP)\PP)\PRT
	her	of the chair	off	wants	push

each sequence in 3.3 cancels to VP' (the prime notation is used merely to distinguish the two VP's) because the infix category of the verb raising trigger is able to penetrate into the left oriented domain of the verb producing the category ((VP'\NP)\PP)\PRT which then cancels with the other categories in the sequence.

Although **L'** gives ordering flexibility to infixation categories such as *wil* above, it is difficult to see how this can be applied to the complex word ordering data presented in the previous Chapter where, for example, in di-transitive sentences there is only the verb and its arguments with no further "special" categories present which can act as the triggers of infixation and where the complex ordering results from constraints at work on the directionality marking of the arguments in the verbal function. It seems appropriate, therefore, to attempt a rather different attack on the problem of describing complex word order phenomena. The work reported in this Chapter is an attempt to develop a theory of word order in Categorial Grammar which addresses itself directly to the problem of constraining the complex and interdependent directionality characteristics of function-internal categories.

It will be useful at this stage to set out clearly the basic assumptions and criteria of this programme of research.

The fundamental assumption of the work reported here is that it is possible (and necessary) to partition the set of strings of word in a language into the grammatical and the ungrammatical. This is an enterprise full of difficulties, often with a very grey area in the data where decisions appear arbitrary. It is nevertheless important to make the decision in any given case. The alternative is the view that there is a cline of grammaticality defined over the stringsets of a language from the most acceptable sequence to the least acceptable. The problem with this is that it fails to recognise the clear judgements that can be made in many cases with regard to the grammaticality of a string of words irrespective of context, time of utterance, who uttered it and for what purpose (see Moravcsik (1972)). In fact, of course, a fully developed syntactic theory of word order must recognise the element of truth in both these positions: it is possible to partition the string sets of a language and within the set of grammatically acceptable strings, there is a cline of acceptability for which a theory of markedness is required. The primary aim of the present thesis is to establish a well-founded model for making the former distinction, of grammaticality, within Categorial Grammar.

The criteria for a well-founded theory of word order in Categorial Grammar that have guided the work reported in this thesis may be summarised as follows. Firstly, the theory must be observationally adequate. This may seem a rather modest requirement of a syntactic theory. In the face of the complex semi-free word order data discussed here, it is argued that this is, in fact, quite an ambitious requirement. Not all the analyses presented in the following chapters achieve observational adequacy, but they all strive towards that end. Secondly, word order constraints should be as simple and general as possible. It is only through uncovering constraints with these properties that effective comparisons can be made between different dialects within one language and between different languages.

3.3 Local Structure, Categories and Linear Order: An Overview

This section presents an overview of the basic principles of a theory of word order in Categorial Grammar which explicitly factors out linear ordering information from other kinds of categorial information. It is a preview to the more formal statement of the theory in Section 3.4. The entire enterprise of factoring out linear ordering information from category structure developed out of a study of linear ordering in Generalised Phrase Structure Grammar. It is therefore relevant to review some of the underlying principles involved in factoring out linear order information in that model. Section 3.3.1 presents this background. Section 3.3.2 is an informal introduction to the formal structures which are involved in the theory for which the notion of a *local structure* is central and therefore discussed in detail. Finally, Section 3.3.3 demonstrates how this formal theory models Categorial Grammar category structure.

3.3.1 Background

Generalised Phrase Structure Grammar was one of the first monostratal models to deploy a fully explicit theory of the factoring out of linear ordering information from other syntactic information in the grammar, thereby enabling statements about linear ordering to be made independently of other syntactic components. Since it was the starting point of the research reported here and many of its fundamental ideas have been used in the account of word order in Categorial Grammar given here, it will be useful to summarise some of the main issues as they have arisen in the Generalised Phrase Structure Grammar literature.

A good starting point is Pullum (1982), the paper which first argued for a factoring out of dominance and linear precedence information in a phrase structure grammar. Fundamental to Pullum's suggestions is the distinction between a *metagrammar* and an *object grammar.* Although this distinction is now commonplace in syntactic modelling, it is worth drawing attention to the key role it plays in the model presented in this thesis. It is in the metagrammar that constraints and generalisations, particularly word order constraints and generalisations, are explicitly stated; in the object grammar, these generalisations are only implicit. The metagrammar Pullum proposes is the pair $<F,l>$ where F is a set of Immediate Dominance statements of the form

3.5 $\quad A \rightarrow \{B, C\}_m$

which is interpreted as a syntactic category A introducing (re-writing as) a multiset of two categories B and C. The multiset allows for repetitions of the same category. l is a set of Linear Precedence statements of the form

3.6 $\quad B<C$

which means that B linearly precedes C in any Phrase Structure rules or local tree deriving from the immediate dominance statements. The object grammar is then defined as the set of context free Phrase Structure rules compatible with at least one of the Immediate Dominance statements and *all* the Linear Precedence statements. Notice that the object grammar results from taking the *conjunction* of all the Linear Precedence statements, a point returned to below.

The primary source for Generalised Phrase Structure Grammar - Gazdar, Klein, Pullum and Sag (1985) - adopts the Immediate Dominance/Linear Precedence format exactly as suggested in Pullum (1982). The model is, however, given a more formal statement which makes clearer some of the underlying structures involved. Immediate Dominance rules are formally defined (Gazdar, Klein, Pullum and Sag 1985, page 53) as members of the set of pairs in 3.7.

3.7 $K \times POW_{\mathrm{m}}(K)$

where K is the set of categories defined by the grammar, POW_{m} is the multi-set powerset[3] operation and consequently, $POW_{\mathrm{m}}(K)$ is the set of all multi-sets drawn from K. Gazdar, Klein, Pullum and Sag (1985, pages 52-53) defines a mapping from pairs of this sort to pairs in which the second element is an ordered n-tuple or list. Intuitively, in Generalised Phrase Structure Grammar terms, pairs of this sort are the formal structures underlying local trees for which there is an exhaustive ordering relation defined over the leaves. The Linear Precedence statements are a filter on this mapping.

The above account contains all the ingredients for developing a fully general theory of linear ordering: the idea of a pairing of a single designated symbol and either a multiset or n-tuple of symbols; the meta-level factoring of linear ordering information; and, the constrained mapping between the multisets and the ordered n-tuples. Above all, there is the idea that these ordered pairs and multisets constitute formal objects which are open to different interpretations. In Generalised Phrase Structure Grammar these structures are interpreted as Phrase Structure rules or local trees. In what follows, a functional categorial interpretation will be developed.

3.3.2 Local Structures and Linear Order

Three generalisations of the formal theory underlying the Generalised Phrase Structure Grammar account of linear ordering are proposed in this subsection: a generalisation of the pairs in 3.7 to n-tuples; a generalisation of linear ordering statements; and, finally, a generalisation of the logic of complex linear ordering statements.

The result of generalising the pairs in 3.7 to n-tuples is a formal object which is referred to as a *local structure*. This is the key construct in the theory of word order developed in this thesis. It is defined as a set-theoretic object (an n-tuple) whose elements are categories of the grammar and (optionally) other primitive symbols or structures designated by the grammar. It has the following components:

1 a category of the grammar referred to as the head;[4]

2 a collection of categories of the grammar (either a multiset or an ordered n-tuple);

3 any number of optional, designated objects (primitives or structures) each forming one element in the n-tuple.

Two notions of "local structure" are, in fact, required for the theory developed here. The first will be termed an *unordered local structure* which is an n-tuple in which the first element is a category of the grammar and the second element is a multiset of categories of the grammar. An example unordered local structure is given in 3.8

3.8 $<h, \{a, b, c\}_m, \ldots>$

where "h" stands for head and initial lower case letters of the alphabet stand for categories of the grammar throughout. The standard notation for multisets (ie. $\{a, b, c\}_m$) will be assumed but the subscript will be omitted unless this might result in ambiguity.

The second notion will be termed an *ordered local structure.* Ordered local structures are related to ordered n-tuples or lists in exactly the same way that unordered local structures are related to multisets. Lists are objects over which a linear ordering relation is defined for every item in the list with respect to every other item in the list. The standard notation (eg. <a,b,c> for a list of three items) will be used throughout. An example ordered local structure is given below.

3.9 <h, <a, b, c>, ...>

In 3.8 and 3.9 the abbreviatory periods indicate any number of additional elements in the n-tuples. These additional elements may be thought of as representing information relevant to the *entire* local structure. So, if the local structure is interpreted graph-theoretically in some grammatical model as a tree, the additional elements of the n-tuple might contain information about the whole tree, for instance its semantic interpretation. If, on the other hand, the local structure is interpreted as a function with the head as the value returned when the arguments in the second element are accepted, then the further elements of the n-tuple may be interpreted as properties of the function itself. This will be important for later application of this approach to Categorial Grammar.

One further distinction is required: that between the set of *potential* ordered local structures and the set of *admissible* ordered local structures.

A set of potential ordered local structures is *projected* from an unordered local structure in two steps: firstly, a set of ordered n-tuples is constructed of which the second element in each is an n-tuple expressing one of the possible orderings on elements in the multiset of the unordered local structure, all orderings being represented in the set; secondly, the powerset of this is taken as the set of potential ordered local structures. An example will make this clear. Given an unordered local structure of the form

3.10 <h, {b, c}, ...>

The following is the projected set of ordered pairs:

3.11 {<h, <b, c>, ...>, <h, <c, b>, ...> }

giving, under the powerset operation, the following set of potential ordered local structures as projections of the unordered local structure:

3.12 { {}, {<h, <b, c>, ...>}, {<h, <c, b>, ...>},
{<h, <b, c>, ...>, <h, <c, b>, ...>}}

The elements of the set of admissible ordered local structures are the largest sets of the members of the set of potential ordered local structures for which each ordered local structure is consistent with the appropriate linear ordering statements of the grammar.

Just as the definition of local structure can be generalised as indicated, the definition of the linear ordering statements can also be generalised. The conventional Linear Precedence statement of the form "a<b" can be taken to be a pair <a,b>, the ordering of which must be true of any admissible local structure to which it applies. The generalisation adopted here consists in taking the linear ordering constraint to be *any* n-tuple of designated symbols (categories) in the grammar. So, the simplest linear ordering statement has the form <a>. Ordering statements formed from triples and larger lists are possible but unnecessary since any n-tuple (n > 1) can be represented by a set of n-1 pairs, as 3.13 illustrates for a 4-tuple.

3.13 <a,b,c,d> = {<a,b>, <b,c>, <c,d>}

Consequently, *two* of the basic linear ordering statements are <a> (usually written as simply "a") and <a,b>. A third type of basic expression will be introduced in the more complete formal account in the next section.

Finally, this section introduces a generalisation of the logic of complex linear ordering statements. It was pointed out earlier that in Generalised Phrase Structure Grammar, the set of admissible local trees was the largest set consistent with *all* the Linear Precedence statements. This in effect takes complex Linear Precedence statements to be conjunctive lists of basic Linear Precedence statements. It is one of the conclusions of the work carried out here into semi-free word order languages that a simple conjunction of linear ordering constraints is not sufficient for the description of the complex word orders found in such languages. Consequently, it is suggested here that arbitrary boolean combinations of linear ordering constraints are allowed by the grammar. The fundamental problem is then to show how the Propositional calculus which expresses these boolean statements (referred to as $\mathbf{L_p}$) is interpreted against the set of potential ordered local structures induced by the grammar. Some informal examples will establish the principles involved in this process. Section 3.4 will give this formally.

Assume that lower case letters stand for categories of the grammar which are themselves complex symbols of feature name/value specifications and that the unordered local structures contain partially specified categories. The linear ordering filter may be thought of as part of the mapping from partially to fully specified categories. Consider the following unordered local structure which has only two members (note that further elements in the n-tuple are ignored here):

3.14 $\quad <h, \{a, b\}>$

and a binary feature F which can be instantiated on both a and b. The following L_p expression

3.15 $\quad a[+F]$

will induce the set of admissible ordered local structures in 3.16 in which feature instantiation has taken place and the category "a" has been constrained to be +F.

3.16 $$\left\{ \begin{array}{l} <h, < \underset{+F}{a}, \underset{+F}{b} >> \\[1em] <h, < \underset{+F}{a}, \underset{-F}{b} >> \\[1em] <h, < \underset{+F}{b}, \underset{+F}{a} >> \\[1em] <h, < \underset{-F}{b}, \underset{+F}{a} >> \end{array} \right\}$$

On the other hand, the L_p expression in 3.17 induces the set of admissible ordered local structures in 3.18.

3.17 $<a, b>$

3.18 $$\left\{ \begin{array}{l} <h, < \underset{+F}{a}, \underset{+F}{b} >> \\[1em] <h, < \underset{+F}{a}, \underset{-F}{b} >> \\[1em] <h, < \underset{-F}{a}, \underset{+F}{b} >> \\[1em] <h, < \underset{-F}{a}, \underset{-F}{b} >> \end{array} \right\}$$

Finally, consider the effect of a complex L_p expression. The expression in 3.19 induces the set of ordered local structures in 3.20

3.19 $<a, b> \supset <+F, -F>$

$$3.20 \quad \left\{ \begin{array}{l} \langle h, \langle \underset{+F}{a}, \underset{-F}{b} \rangle\rangle \\[2ex] \langle h, \langle \underset{+F}{b}, \underset{+F}{a} \rangle\rangle \\[2ex] \langle h, \langle \underset{+F}{b}, \underset{-F}{a} \rangle\rangle \\[2ex] \langle h, \langle \underset{-F}{b}, \underset{+F}{a} \rangle\rangle \\[2ex] \langle h, \langle \underset{-F}{b}, \underset{-F}{a} \rangle\rangle \end{array} \right\}$$

Clearly, a different set of local structures to either of the previous sets. This is a highly schematic example but it does serve to illustrate the way that a full Boolean logic can be used to define particular sets of ordered local structures. When these abstract local structures are interpreted as categorial functions in the way explained in the following section, each member of the set of local structures realises a distinct ordering of constituents. It is in this way that L_p statements encode generalisations about the ordering characteristics of semi-free word order languages.

Having set up the formal structures required to exploit a linear order factored grammar, it is now possible to consider in exactly what ways Categorial Grammar categories relate to these structures.

3.3.3 Categories as Local Structures

Conventional Categorial Grammar categories do not appear at first sight to have the right property to be interpreted as local structures in the sense of the previous section. The functional realisation of a local structure is an n-th order n-ary function ie. a function over one or more arguments which may be basic expressions or functions. Conventional Categorial Grammar categories, on the other hand, are n-th order unary functions, that is they take only one argument which may or may not be a function. However, it has already been indicated in Chapter 1 that there is a result due originally to Schönfinkel (1924) establishing an isomorphism between n-ary functions and unary functions. Consequently, in a categorial context, for every conventional Categorial Grammar category there is a local structure representation of the kind introduced already. The result is that any category of the form given in 3.21a can be "restructured" (ignoring the vertical slash marker for the moment) into an equivalent function of the form given in 3.21b

3.21 a ((W|X)|Y)|Z

b <W, <Z, Y, X>>

where the angle bracket notation is intended to draw attention to the fact that these re-structured categories are ordered local structures in the sense of the previous section. Notice also that the second element of the pair is a stack which is popped from the left.

This re-structuring applies equally to functions taking functions as arguments, as 3.22a and 3.22b illustrate.

3.22 a (W|(A|B))|X

b <W, <X, <A, <B>>>>

It is this equivalence which underlies the claim that Categorial Grammar categories are local structures in the formal sense defined above.

An important issue for the theory of Categorial Grammar categories developed here is the nature of the directionality marking on arguments once they have been re-structured into a stack. The assumption made here is that directionality is associated with arguments of functors taken as a whole. The result is that a conventional Categorial Grammar category of the form in 3.23a is now represented as in 3.23b.

3.23 a ((W/X)\Y)/Z

b <W, </Z, \Y, /X>>

Directionality is not a syntactic feature of the same kind as other features used to analyse the complex symbols which form the basic expressions of the grammar, rather it is a *statement about* argument categories. For one thing, it occurs only with the arguments of functors, not on basic expressions in themselves. So the functor in 3.23b can combine with a category Z to its right. It makes no sense to say that it combines with a category /Z. Further, directionality marks *all* arguments in a functor, even functor arguments. So the conventional Categorial Grammar category in 3.24a (a directional variant of 3.22a) is represented as a local structure in 3.24b with a directionality feature added to the argument functor A/B.

3.24 a (W\(A/B))/X

b <W, </X, \<A, </B>>>>

The special status of directionality will be clarified later and, together with the optionality feature introduced later, will play a key role in the full theory of Extended Categorial Grammar presented in Chapter 7.

3.4 A Formal Theory of Linear Order in Categorial Grammar

This Section is in two parts. 3.4.1 gives the formal syntax and semantics of the linear ordering logic L_p. The second subsection gives a formal definition of the set of admissible ordered local structures.

3.4.1 L_p: A Language for Linear Order

L_p is a modal propositional logic which is evaluated against a well-defined model, M, of the set of sets of potential ordered local structures induced by the grammar. The set of basic expressions of L_p is given by the five clauses in 3.25.

3.25
- a every category, α, of the grammar is a basic expression of L_p;
- b every expression of the type $\backslash\alpha$, is a basic expression of L_p where α is a category of the grammar;
- c every expression of the type $/\alpha$, is a basic expression of L_p where α is a category of the grammar;
- d every pair $<\sigma, \tau>$, such that σ, τ are basic expressions of L_p, is also a basic expression of L_p;
- e every expression of the type $f{:}\xi$ is a basic expression of L_p where ξ is a designated symbol given by the grammar.

Examples of the basic expressions of L_p given by this definition (for categories a,b and designated symbol s) include: /a, \b, $<$a, b$>$, $<$\a, b$>$ and f:s. The use of the basic expression type in 3.25e will be illustrated in Chapter 5, particularly Section 5.2.2. Basically, it enables linear ordering constraints to refer to elements in the local structure n-tuple *beyond* the first two. It thereby allows constraints to refer to properties of functions taken as a whole.[5]

The **semantics of basic expressions** requires a number of the preliminary definitions.

A distinction will be made between *non-directional* and *directional* categories of the grammar. Non-directional categories are those with structure as in 3.26a below where there is no directionality marker as a member of the n-tuple defining the category. Directional categories have a directionality marker as an element of the n-tuple, they therefore have one of the forms given in 3.26b and 3.26c.

3.26 a $<\alpha, \ <...>, ...>$

b $<\alpha, \ <...>, /, ...>$

c $<\alpha, \ <...>, \backslash, ...>$

Notice that only ordered local structures are given here. It is however quite possible for directionality features to be assigned to unordered local structures explicitly in the lexicon just so long as any such assignment does not conflict with the results of the L_p filtering.

A category α is called the *non-directional equivalent* of a directional category β if α is identical to β in every way except that it has no directionality marker as a member of its n-tuple. Left directional categories are abbreviated to "$\backslash\alpha$" where α is a non-directional category; right directional categories are abbreviated to "$/\alpha$" where α is a non-directional category.

Finally, the notion of an *expansion* of an ordered pair is introduced. Given an ordered pair $\alpha = <a, b>$, an expansion of α is an ordered n-tuple ($n \geq 2$) in which the sequence $<..., a, ..., b, ...>$ appears and in which there is no sequence $<..., b, ..., a, ...>$.

So, for example, 3.27a and 3.27b are expansions of $<a, b>$, but 3.27c is not:

3.27 a $<a, b, c>$

b $<a, c, b>$

c $<a, b, c, a>$

Let M be the set of sets of potential ordered local structures induced by the grammar. Then, $\alpha \in M$ is a projection of an unordered local structure, $\beta \in \alpha$ is a set of potential ordered local structures projected from one unordered local structure, and $\gamma \in \beta$ is one potential ordered local structure.

For example, given the unordered local structure in 3.28, 3.28 <h, {b, c}, ...>

the following projection is a possible $\alpha \in$ M:

3.29 { {}, {<h, <b, c>, ...>}, {<h, <c, b>, ...>},
{<h, <b, c>, ...>, <h, <c, b>, ...>}}

In which case β could be

3.30 {<h, <b, c>, ...>, <h, <c, b>, ...>}

and γ

3.31 <h, <b, c>, ...>

Assume a notation for the denotation of expressions in which "$[\![\alpha]\!]_\beta = 1$ iff *condition*" means that the semantic value of α is true with respect to some object β under the given condition and "$[\![\alpha]\!]_\beta = 0$ iff *condition*" means that α is false with respect to β under the given condition. Further, assume a transitive set membership relation $\in_T$ defined as follows: $\alpha \in_T \beta$ iff either (1) $\alpha \in \beta$ or (2) there are sets $\sigma_1, \ldots, \sigma_n$ such that $\alpha \in \sigma_1$, $\sigma_1 \in \sigma_n$ and $\sigma_n \in \beta$.

The semantics of the basic expressions is given in 3.32 to 3.36, assuming the definitions of M and γ above and where γ:2 is the n-tuple forming the second element in each of the ordered local structures in the model M.

3.32 $[\![a]\!]_{\gamma \in_T M} = 1$

iff either there is a category "a" in γ:2 or there is some directional category "\b" or "/b" in γ:2 for which its non-directional equivalent (ie. "b") is identical to "a".

3.33 $[\![\backslash a]\!]_{\gamma \in_T M} = 1$

iff there is some category in γ:2 whose non-directional equivalent is identical to "a" and which is also $<...,\backslash,...>$.

3.34 $[\![/a]\!]_{\gamma \in_T M} = 1$

iff there is some category in γ:2 whose non-directional equivalent is identical to "a" and which is also $<...,/,...>$.

3.35 $[\![<\sigma, \tau>]\!]_{\gamma \in_T M} = 1$

iff $[\![\sigma]\!]_{\gamma \in_T M} = 1$ and $[\![\tau]\!]_{\gamma \in_T M} = 1$ and γ:2 is an expansion of $<\sigma,\tau>$.

3.36 $[\![f{:}\alpha]\!]_{\gamma \in_T M} = 1$

iff α is a designated symbol of the grammar and $\gamma = <...,\alpha,...>$.

It is easy to see from this that basic expressions will typically be true not only at several $\alpha \in M$, but also at several $\gamma \in_T M$. This point will be returned to.

Under the categorial function interpretation of local structures adopted from now on, informal explanations of these semantic rules are as follows. 3.32 says that a category "a" is true of a function if at least one of its arguments is identical to "a". This is, in effect, simply the assertion that the function must have this category in its argument list; it places no directionality constraints upon that argument. 3.33 and 3.34 are the same as 3.32 except that they additionally impose directionality on the function's arguments.

3.35 imposes an ordering on categories in the function's argument stack. Since it is a stack, popped from the left, this basic L_p expression can be used to define an ordering on the binary combination of a function with its arguments, a topic explored further below.

Finally, 3.36 allows L_p expressions to make reference to the entire function. Note that its definition refers to γ (ie. the entire local structure), not just the argument list of the local structure (ie. γ:2). This is the sole resource available for allowing L_p expressions to access the relationship between a function and its arguments.

The **complex expressions of L_p** are as follows, where φ and ψ are wffs of L_p:

3.37 $\sim\varphi$

3.38 $\varphi \vee \psi \vee \ldots$

3.39 $\varphi \ \textcircled{\vee} \ \psi \textcircled{\vee} \ \ldots$

3.40 $\varphi \ \& \ \psi \ \& \ \ldots$

3.41 $\varphi \supset \psi$

3.42 $\varphi \leftrightarrows \psi$

3.43 $\Box\varphi$

3.44 $\Diamond\varphi$

The expressions in 3.38 to 3.40 allow for concatenation of disjuncts (inclusive and exclusive) and conjuncts respectively in complex expressions of L_p, thereby avoiding the need for sequences of parentheses in long expressions.

The **semantics of the non-modal complex expressions** is as expected given the relativisation of truth values to $\gamma \in_T$ M:

3.45 $[\![\sim\varphi]\!]_{\gamma \in_T M} = 1$ iff $[\![\varphi]\!]_{\gamma \in_T M} = 0$

3.46 $[\![\varphi \vee \psi \vee \ldots]\!]_{\gamma \in_T M} = 1$ iff for at least one expression $\varepsilon \in \{\varphi, \psi, \ldots\}$, $[\![\varepsilon]\!]_{\gamma \in_T M} = 1$

3.47 $[\![\varphi \circledvee \psi \circledvee \ldots]\!]_{\gamma \in_T M} = 1$ iff for at least one but not all expressions $\varepsilon \in \{\varphi, \psi, \ldots\}$, $[\![\varepsilon]\!]_{\gamma \in_T M} = 1$

3.48 $[\![\varphi \;\&\; \psi \;\&\; \ldots]\!]_{\gamma \in_T M} = 1$ iff for every expression $\varepsilon \in \{\varphi, \psi, \ldots\}$, $[\![\varepsilon]\!]_{\gamma \in_T M} = 1$

3.49 $[\![\varphi \supset \psi]\!]_{\gamma \in_T M\gamma} = 1$ iff $[\![\varphi]\!]_{\gamma \in_T M} = 0$ or $[\![\psi]\!]_{\gamma \in_T M} = 1$

3.50 $[\![\varphi \leftrightarrows \psi]\!]_{\gamma \in_T M} = 1$ iff $[\![\varphi]\!]_{\gamma \in_T M} = [\![\psi]\!]_{\gamma \in_T M}$

The semantics of the modal operators is as follows:

3.51 $[\![\Box\varphi]\!]_{\gamma \in_T M} = 1$ iff $[\![\varphi]\!]_{\gamma \in_T M} = 1$ and for every local structure $\sigma \in \gamma{:}2$, $[\![\Box\varphi]\!]_{\sigma} = 1$.

3.52 $[\![\Diamond\varphi]\!]_{\gamma \in_T M} = 1$ iff $[\![\varphi]\!]_{\gamma \in_T M} = 1$ or for some local structure $\sigma \in \gamma{:}2$, $[\![\Diamond\varphi]\!]_{\sigma} = 1$.

Some brief comment is appropriate here. It should be noted that categories in Categorial Grammar, including this extended version, are defined recursively. This means that the linear precedence language must have the properties relevant to a recursive search of a category. The effect is that the logic used to define linear ordering in the new model of Categorial Grammar will be modal for the same reason that the category logic of Gazdar, Pullum, Carpenter, Klein, Hukari and Levine (1987) is modal: $\Box$ induces the equivalent of universal quantification over recursively defined ordered local structures; $\Diamond$ the equivalent of existential quantification. These operators therefore allow for the linear precedence constraints to be applied to successively embedded structure.

Consider the following potential ordered local structure:

3.53 $\langle h_1, \langle \langle h_2, \langle b, a \rangle \rangle, a, b \rangle \rangle$

The simple L_p statement

3.54 $\langle a, b \rangle$

will be true of 3.53 above because it is true of the "top" level of the structure. This will also be the case for

3.55 $\Diamond(\langle a, b \rangle)$

However,

3.56 $\Box(\langle a, b \rangle)$

is not true of this local structure, since this statement must be true of every relevant embedded local structure as well, and in this case the embedded local structure possesses an ordering for which the L_p statement is not true.

A "real language" example of such embedding will be discussed in Chapter 4 when considering English auxiliary verbs. In that case the local structure is that in 3.57 (omitting directionality markings).

3.57 $\langle S, \langle \langle S, \langle NPsubj, NPobj \rangle \rangle, NPobj, NPsubj \rangle \rangle$

This, of course, corresponds to 3.53 with h_1, h_2 = S, a = NPobj and b = NPsubj.

3.4.2 Admissible Local Structures

The last requirement for this theory of precedence relations is to define the set of *admissible ordered local structures*. This is the maximally large set of ordered local structures deriving from all unordered local structures and consistent with all relevant L_p statements. A particular set of admissible local structures is, of course, grammar specific in the sense that different sets of statements in L_p will induce different sets of admissible structures.

Let Λ be the finite set of statements of L_p asserted on any one occasion.[6] It was pointed out earlier that typically an expression $\lambda \in \Lambda$ will be true not only at more than one set of projections of the unordered local structures but also at more than one projection within the same set. In order to define the set of admissible local structures a notion of the maximal or largest set of a projection is needed for a given unordered local structure compatible with $\lambda \in \Lambda$.

Standardly, the truth set of a proposition in a modal logic is that set of possible worlds at which the proposition is true. Here the notion is adapted to the projections of unordered local structures. The truth set will be defined for the sets $\beta \in \alpha \in M$.

Let $|\lambda|_{M,\alpha}$ be the truth set of $\lambda \in \Lambda$, a proposition of L_p, relative to one set of potential ordered local structures $\alpha \in M$. This set will contain all $\beta \in \alpha$ for which λ is true at each $\gamma \in \beta$. Then, $MAX|\lambda|_{M,\alpha}$ is that set $\beta \in \alpha$ with the largest cardinality.

For example, given an unordered local structure as follows:

3.58 $\quad <h, \{a, b, c\}, \ldots>$

The following would be three of the sets $\beta \in \alpha \in M$:

3.59 a $\{<h, <a, b, c>, \ldots>\}$

b $\{<h, <a, b, c>, \ldots>, <h, <b, a, c>, \ldots>\}$

c $\{<h, <a, b, c>, \ldots>, <h, <b, a, c>, \ldots>,$
$<h, <a, c, b>, \ldots>\}$

$|<a, c>|_{M,\alpha}$ contains all three of these sets; $MAX|<a, c>|_{M,\alpha}$ contains only the last one.

Π_α is then defined as the intersection of $MAX|\lambda|_{M,\alpha}$ for all $\lambda \in \Lambda$,

3.60 $\Pi_\alpha = \forall\lambda \in \Lambda, \cap MAX|\lambda|_{M,\alpha}$

This gives the admissible ordered local structures for all $\lambda \in \Lambda$ associated with one unordered local structure.

Continuing with the example, the L_p statements in 3.61, defined over the unordered local structure in 3.58.

3.61 a $<a, c>$

b $<a, b>$

give respectively,

3.62 a $MAX|<a, c>|_{M,\alpha} = \{<h, <a,b,c>, \ldots>, <h, <b,a,c>, \ldots>,$
$<h, <a,c,b>, \ldots>\}$

b $MAX|<a, b>|_{M,\alpha} = \{<h, <a,b,c>, \ldots>, <h, <c,a,b>, \ldots>,$
$<h, <a,c,b>, \ldots>\}$

Therefore

3.63 $\Pi_\alpha = \{<h, <a,b,c>, \ldots>, <h, <a,c,b>, \ldots>\}$

which is the set of local structures projected from 3.58 compatible with the two precedence constraints.

Of course, Π_α might be the empty set for some α, in which case it is taken to be the largest set in α. In effect, this makes admissible all orderings for those ordered local structures to which no L_p statements apply.

Finally, the set of admissible local structures E for the grammar is the union of all Π_α:

$$3.64 \quad E = \forall\alpha \in M, \cup\Pi_\alpha$$

Because of the central role the linear precedence logic plays in all the work reported in this thesis, it is worthwhile illustrating further how it admits sets of ordered local structures.

The previous section gave a formal account of how expressions of L_p are evaluated against potential local structures ($\alpha \in M$). This involves taking the intersection of the maximal truth sets for each L_p expression at some α. The final set of admissible local structures is then the union of these values for all $\alpha \in M$.

Now consider the problem of evaluating a complex expression of L_p such as that in 3.65 below.

$$3.65 \quad \langle a, b\rangle \leftrightarrows \langle b, c\rangle$$

As in the previous section, lower case letters of the alphabet are used throughout to represent categories of the grammar. At this stage, this simplification does not affect the conclusions drawn, although in later sections the feature-based approach to category structure will be introduced to allow for more complex linear ordering statements.

Assuming the unordered local structure 3.58, $\beta \in \alpha \in M$ will include, among many other sets, the following (ignoring the possibility of other elements in the n-tuple):

3.66 a $\{\}$

b $\{\langle h, \langle a, b, c\rangle\rangle\}$

c $\{\langle h, \langle a, b, c\rangle\rangle, \langle h, \langle a, c, b\rangle\rangle\}$

d $\{\langle h, \langle c, b, a\rangle\rangle\}$

e $\{\langle h, \langle a, b, c\rangle\rangle, \langle h, \langle a, c, b\rangle\rangle, \langle h, \langle c, a, b\rangle\rangle\}$

3.65 will be true of each of these sets whenever the truth values of its composing basic expressions have the same truth value for any $\gamma \in \beta$. Both the basic expressions are true of 3.66b, this is therefore in the truth set of $\langle a, b\rangle \leftrightarrows \langle b, c\rangle$. Similarly, both basic expressions are false of 3.66d, hence this is also in the truth set of the complex expression. No other $\beta \in \alpha \in M$ is in the truth set of this complex expression.

Ignoring the designated category "h", this will induce the following as the maximally large set of potential ordered local structures:

3.67 $\{\langle a, b, c\rangle, \langle c, b, a\rangle\}$

It is clear that working through the details of the semantic rules in this way in order to retrieve the denotations of complex expressions is too laborious to be practical if an implementation of the formalism is to be considered. Fortunately, there is a more direct method of evaluating the truth values of complex expressions and obtaining the truth sets associated with them.

Again, assume the complex expression in 3.65. Further, assume the standard truth table for the evaluation of strict implication, given in table 3.1 below. The truth set for this expression relative to the unordered local structure 3.58 is given directly by evaluating each ordering of the maximally large set of ordered local structures derived from 3.58.

This is demonstrated in Table 3.2, using abbreviated ordered local structures missing their heads. The set in 3.67 is therefore induced by this complex L_p expression.

$\leftrightarrows$	T	F
T	T	F
F	F	T

Table 3.1

	<a, b>	$\leftrightarrows$	<b, c>
<a, b, c>	T	T	T
<a, c, b>	T	F	F
<b, a, c>	F	F	T
<b, c, a>	F	F	T
<c, a, b>	T	F	F
<c, b, a>	F	T	F

Table 3.2

3.5 Summary

This Chapter presented the background to a linear order factored Categorial Grammar followed by a formal presentation of the theory. It was argued that most recent attempts to constrain the power of Categorial Grammar with respect to the ordering of arguments relative to functions do not address the descriptive problems posed by semi-free word order languages. In contrast to these approaches an analysis was presented of Categorial Grammar categories as local structures akin to the Immediate Dominance statements of Generalised Phrase Structure Grammar or more traditional Phrase Structure rules. It was claimed that the same formal constructs underpin both categories and rules. Given this, it follows straightforwardly that the linear order/local structure factoring which has been developed within the Generalised Phrase Structure Grammar paradigm carries over to Categorial Grammar. This idea was presented both informally through examples and formally. Further, a logic (L_p) for the expression of linear ordering constraints was given with its semantics. The result is that the tools have been made available for the exploration of complex word order data. In the next Chapter specific aspects of this new formalism are examined and where necessary modifications are made prior to the application of the model to natural language analysis in Chapter 5.

3.6 Notes

[1] See Brody (1984) for a critique of the underlying assumption of typological theory that it is possible to uniquely identify a basic word order for every language and that the notion of a basic word order is uniform across all languages.

[2] In context, the data in 3.3 are used to distinguish Dutch and Flemish since Dutch permits only 3.3a and 3.3b. However, the relevant point for the present discussion is the account given of the flexible position of *wil*.

[3] Formally defined in Gazdar, Klein, Pullum and Sag (1985), page 53.

[4] The use of the term *head* here is quite distinct from the use of the term in Generalised Phrase Structure Grammar, for example Gazdar, Klein, Pullum and Sag (1985, page 50-52).

[5] The definition of basic L_p expressions in 3.25 makes the Categorial Grammar model of linear order factoring presented here significantly different to the Immediate Dominance/Linear Precedence factoring of Generalised Phrase Structure Grammar. In particular, basic expressions of type 3.25e do not have any obvious correlate in Generalised Phrase Structure Grammar. The reason for this is worth commenting on in some detail. The categories that populate local structures in Generalised Phrase Structure Grammar are arbitrarily complex, recursive feature name/value specifications. They encode *all* relevant syntactic information in the grammar. So, for example, whatever information needs to be given with respect to, say, verbs can be encoded within the verbal categories in the form of feature name/values. In Categorial Grammar, on the other hand, the number of basic categories is very small. It will be argued later that all the

categories in the grammar (including N, NP etc) are local structures but that the value returned by a basic category is a primitive semantic type which is represented as a feature name/value matrix. This means that any information relevant to the primitive symbols can be encoded as feature settings in the grammar but that no such representation is available for functions. Hence the introduction of n-tuples for category representation, where the elements for which n $\geq$ 3 are features associated with the entire category, and therefore the need for a basic expression of type 3.25e. A specific example will make this clear. Siewierska (1988) - an extensive survey of theories and data relating to the determination of word order - gives the following data from German (page 256) based on van Riemsdijk (1983, pages 235-237):

Er ist auf Musik erpicht
he is on music keen
(he is keen on music)

Er ist erpicht auf Musik
he is keen on music

Ein auf Musik erpichter Student
a on music keen student
(a student keen on music)

*Ein erpicht(er) auf Musik Student
a keen on music student

The point is that the prepositional complement of the adjective (*auf Musik*) may either precede or follow it when it is used predicatively; when it is used attributively (prenominally) the complement may only precede the adjective. The linear ordering expression required to express this constraint must therefore have access to the functional role of the adjective in the sentence. A Generalised Phrase Structure Grammar local structure for adjectives can encode this information directly within the category for adjectives and, in addition, bind the relevant daughter to the mother using variables, as for example

$<$AjP[AjType: α], $<$Aj[AjType: α], PP$>>$

where α ranges over {predicative, attributive}. The L_p constraint required to express the patterning above is straightforward:

<PP, Aj[AjType: attributive]>

(This discussion ignores the issue of just how well motivated analyses like this are in Generalised Phrase Structure Grammar, the point is it can be done). By contrast, in Categorial Grammar, under standard assumptions, an adjective taking a prepositional complement will be a function of category (N/N)/PP which translates into the local structure

<N, <PP[OPT], N>>

where OPT indicates that the prepositional complement is optional. The information about the function of the adjective (encoded in *AjType*) cannot be associated with any of the component categories of this local structure, rather, it is a property of the local structure as a whole. Hence the introduction of extra elements into the structure. In the theory presented here, the actual local structure would be

<N, <PP[OPT], N>, [AjType: α]>

and the L_p constraint would make use of the basic expression type in 3.25e:

f:[AjType: attributive] $\supset$ \PP

[6] The point of the word *relevant* in the previous paragraph, and *on any one occasion* here is to draw attention to the fact that in the Categorial Grammar model under development here, an admissible local structure may be induced by a *subset* of L_p expressions. In the description of semi-free word order languages, there is a general movement towards the parochial imposition of linear ordering constraints on individual functions.

CHAPTER 4

ASPECTS OF A LINEAR ORDER FACTORED CATEGORIAL GRAMMAR

4.1 Introduction

So far categorial functions have been analysed as local structures and expressions of the linear precedence logic L_p have been defined syntactically and semantically. This Chapter presents a number of examples of the way that L_p expressions can be used to describe complex word orders in Categorial Grammars. The purpose of this Chapter is, however, more than merely illustrative of various approaches to the analysis of semi-free word order. In presenting the analyses, modifications and refinements of the basic ideas are made, notably with regard to the introduction of unification and the re-statement of the semantic interpretation rules of L_p using unification in place of identity.

Section 4.2 below explores the relationship between the L_p constraints and coordination still using simplifying abbreviations for categories. It turns out that the linear precedence logic is a powerful mechanism for the description of complex coordination data. In Section 4.3 the notion of unification is introduced and with it a final version of the definition of expansion and the semantic interpretation rules of L_p basic expressions. In Section 4.4 feature passing between feature name-value lists is explored and finally, in Section 4.5, some issues in the analysis of English auxiliary verbs are discussed.

4.2 The Interaction of Linear Order and Coordination

Consider the unordered local structure in 4.1.

4.1 <h, {a, b}, ...>

Assuming only instantiation of the directionality marker on arguments in the function, the set of potential ordered local structures generated from 4.1 is that given in 4.2.

4.2 { <h, </a, /b>, ...>,
<h, </b, /a>, ...>,
<h, </a, \b>, ...>,
<h, <\b, /a>, ...>,
<h, <\a, /b>, ...>,
<h, </b, \a>, ...>,
<h, <\a, \b>, ...>,
<h, <\b, \a>, ...> }

Each of the local structures in 4.2 is associated with a particular ordering of function and its arguments in a grammatical sequence. To make this clear, the orderings of 4.2 are repeated in 4.3 with the word order that each induces indicated. Here "f" stands for the function itself.

4.3	a	<h, </a, /b>, ...>	f a b
	b	<h, </b, /a>, ...>	f b a
	c	<h, </a, \b>, ...>	b f a
	d	<h, <\b, /a>, ...>	b f a
	e	<h, <\a, /b>, ...>	a f b
	f	<h, </b, \a>, ...>	a f b
	g	<h, <\a, \b>, ...>	b a f
	h	<h, <\b, \a>, ...>	a b f

The interesting fact about this is that there is more information in the local structures than the simple word order equivalences on the right of 4.3 indicate. However, to bring this out requires a more explicit model of Categorial Grammar using the linear precedence factoring developed in this thesis. Figure 4.1 outlines the main options available in such a model.

Unordered Local Structures

|| feature instantiation
|| directionality marking
|| all orderings of argument stack
V

Potential Ordered Local Structures

||
|| L_p Statements
V

Permissible Ordered Local Structures

Curried functions	non-Curried functions	
use binary combinators	use non-binary combinators	use modified binary combinators

Categorial Grammar Derivations

Figure 4.1
Options in a Linear Precedence Factored Categorial Grammar

The steps from unordered to permissible ordered local structures were spelled out in the previous Chapter. Figure 4.1 indicates three "routes" from the set of permissible ordered local structures. Conventional Categorial Grammars use curried functions and binary combinators. By using binary combinators such grammars are able to handle many aspects of coordination that pose serious problems for Phrase Structure Grammars. The categories in the model presented here are not curried (ie. they are n-ary functions) and it would be possible to propose that non-binary combinators access all the arguments of a function simultaneously. This would, however, have serious consequences for the ability of the grammar to handle coordination. It has been decided therefore to maintain the binary nature of category combination. This can be done in either of two ways: coerce the categories of the grammar following the L_p filtering to be curried functions and apply conventional combinators or modify the combinators to accept ordered local structures. The second approach is adopted here.

Modification of the combinators, in particular Functional Application and Composition, which play such an important role in the analyses presented below, involves taking a closer look at the categories of the grammar. A distinction will be made between primitive semantic types and categories. The former are often represented as S, NP, N in Categorial Grammars and will, for the purposes of this section at least, continue to be treated as primitive atomic symbols. In the next section, they will be treated as complex symbols. The latter are local structures in the sense already defined. *All* categories of the grammar are local structures. Saturated functions are local structures with empty argument stacks.[1] Basic categories, such as the category associated with proper nouns, are also local structures with empty argument stacks. The category associated with a sentence is that given in 4.4a and the category associated with a Proper Noun (ie. a Noun Phrase) is that given in 4.4b.

4.4 a <S, <>>

b <NP, <>>

With this in mind, it is now possible to re-define the two major combinators. Functional Application is represented in 4.5. 4.5a is the conventional form using X, Y as categories of the grammar and ignoring directionality.

4.5 a X|Y Y => X

b <X, <|Y>> <Y, <>> => <X, <>>

4.5b is the non-curried version using local structures.

Functional Composition is rather more complex. Consider the case of 4.6a for which the local structure equivalent is 4.6b.

4.6 a (V|W)|X X|(Y|Z) => (V|W)|(Y|Z)

b <V, <|X, |W>> <X, <|<Y, <|Z>>>> =>
<V, <|<Y, <|Z>>, |W>>

It would seem that if the head of the second category matches with the first argument of the first category, the result is the first category with its first argument replaced by the argument stack of the second category. This is not fully general however. Consider the instance of Functional Composition in 4.7.

4.7 (V|W)|(X|Y) (X|Y)|Z => (V|W)|Z

The local structure for the first category in 4.7 is that in 4.8a and that for the second category is 4.8b.

4.8 a <V, <|<X, <|Y>>, |W>>

b <X, <|Z, |Y>>

c <V, <|Z, |W>>

The head of the second category (4.8b) does not match the first argument of 4.8a. The combinatorial algorithm required is, nevertheless, quite straightforward. Assume that "pop(category)" returns the leftmost element of the second item in the n-tuple. Functional Composition is then given by the algorithm in 4.9.

4.9 for any two categories, α, β, with α to the left of β,

if $pop(\alpha) = \beta - pop(\beta)$

then form a new category identical to α

except that $pop(\alpha)$ is replaced by $pop(\beta)$.

This gives the category in 4.8c as the result of composing the two categories in 4.8a and 4.8b, the desired result. It is easy to see that this algorithm also gives the right result for the composition in 4.6, if one recalls that the abbreviation "X" stands for the category "<X, <>>". Further, this algorithm generalises over Functional Application *and* Composition. The example of Application in 4.5 follows from 4.9 with $pop(\alpha)$ = <Y, <>> and $\beta - pop(\beta)$ = <Y, <>> since β:2 is already an empty stack.

With the conclusion in mind that Functional Application and Composition can be applied to local structures as binary combinators, consider the L_p statement in 4.10 defined over the unordered local structure in 4.1, repeated here.

4.10 /a & \b & <a, b>

4.1 <h, {a, b}, ...>

The set of permissible ordered local structures it admits is given in 4.11.

4.11 { <h, </a, \b>> }

This single local structure is a function of two arguments, the first of which is an "a" to the right and the second a "b" to the left. The conjunction of basic L_p expressions in 4.10 therefore admits a highly restricted language with just one word order, given in 4.12.

4.12 b f a

There are, of course, other (and simpler) ways of defining this "language", the most obvious being the single L_p statement in 4.13.

4.13 </a, \b>

Typically, there will be numerous logically equivalent L_p expressions defining any given word ordering.

As indicated earlier, ordered local structures contain more information than simply an indication of word order. They also indicate the *coordination potential* of word strings. There are two dimensions to the problem of coordination. One involves defining appropriate substrings over which to allow coordination to apply. That is what is meant here by coordination potential. It is essentially a matter of constraining associativity and bracketing. The second dimension involves giving an account of how conjuncts operate and the sort of category associated with them. For the purposes of the work reported here, the former is important. It is not necessary to make a decision about the categorial status of conjuncts. The simple account in Chapter 1 is sufficient.

The coordination potential admitted by the structure in 4.11 is restricted to coordination over the function and the first argument it accepts (ie. "a"). This is represented by the bracketings in 4.14 and 4.15 which follow from the fact that the function cannot combine with "b" before "a".

4.14 b [[f a] & [f a]]

4.15 *[[b f] & [b f]] a

It is the use of the basic expression <a, b> which guarantees this property of the language.

This example serves to demonstrate how complex L_p expressions can be used to define complex word order possibilities as well as restrict coordinations. This flexibility is used below in the analysis of a range of Spanish data.

4.3 Unification and L_p

So far, primitive semantic types have been treated as atomic symbols. Before applying the theory of Categorial Grammar developed here to natural language data, this over-simplification must be corrected. The previous section made clear that *all* categories of the grammar are local structures. In this section the primitive semantic types are analysed as complex feature name-value matrices in a way that is familiar from a number of current syntactic theories.[2] This use of complex symbols is motivated by the radical simplification it brings about when information is combined from two categories. It does, however, mean that some further complexity must be introduced into the statement of *expansion* and the semantic interpretation rules of L_p.

Unification is a major topic of research in current syntactic theory and this is not the place to explore the concept in detail.[3] It is necessary, however, to give a brief account of how unification works with respect to the simple feature name-value matrices which constitute the basic categories of the Categorial Grammar under development in this thesis. Chapter 7 takes the topic of unification further in the context of the more comprehensive theory of Extended Categorial Grammar.

Unification is an operation for combining two complex pieces of information to form a new, perhaps more fully specified piece of information. This operation works under the constraint that neither of the input structures must contain information that contradicts information in the other. Consider the three simple name-value lists in 4.16.

$$4.16 \quad a \quad \begin{bmatrix} \text{N} & + \\ \text{V} & - \\ \text{Num} & \text{plural} \end{bmatrix}$$

$$b \quad \begin{bmatrix} \text{N} & + \\ \text{V} & - \\ \text{GR} & \text{subject} \end{bmatrix}$$

$$c \quad \begin{bmatrix} \text{N} & + \\ \text{V} & - \\ \text{GR} & \text{object} \end{bmatrix}$$

4.16a and 4.16b will unify to produce the list in 4.17 below since neither contains information that contradicts the other.

4.17 $\begin{bmatrix} \text{N} & + \\ \text{V} & - \\ \text{Num} & \text{plural} \\ \text{GR} & \text{subject} \end{bmatrix}$

On the other hand, 4.16c cannot unify with 4.16b since the grammatical relation (GR) specification has different values in each. The operation of unification is represented here by the symbol $\sqcup$. Hence, on the assumption that the three structures in 4.16 are named respectively "a", "b", "c" the structure in 4.17 is "a$\sqcup$b", but there is no structure "a$\sqcup$c".

Notice that the unification of two structures may produce a structure which is identical to one of the inputs or, as in the case of 4.17, a new structure containing more specific information.

This summary of the main principles of unification is sufficient for present purposes. It will be clear, however, that this is a complex area especially when the structures to be unified are more than simple name-value lists and when variables and structure sharing are permitted. Chapter 7 employs some of these more complex aspects of unification for the full theory of Extended Categorial Grammar.

As stated earlier, all categories of the grammar have the structure

4.18 $<\alpha, \{...\}, ...>$

where α is a primitive semantic type of the grammar. The Categorial Grammar model developed here takes these primitive semantic types (S, NP, N) to be complex rather than atomic symbols. *All* categories of the grammar are local structures, therefore in the structure in 4.19 below, returned by a saturated verb

4.19 <S, <>, ...>

the symbol S stands for some set of features such as that given in 4.20.

4.20 $$\begin{bmatrix} \mathrm{N} & - \\ \mathrm{V} & + \\ \mathrm{Phrasal} & + \\ \ldots & \end{bmatrix}$$

The feature system adopted here derives from Chomsky (1970) and Jackendoff (1977). By decomposing syntactic classes such as verb and noun into primitive features N and V, it is possible to state generalisation across these classes. It is important to emphasise, however, that this convention is being used here simply for exemplification; no part of the analysis of word order in this or the following chapters depends essentially on this particular feature system.

L_p statements are assumed to have access to individual features when stating linear ordering constraints on the argument categories within functions. Consider an unordered local structure such as that in 4.21.

4.21 <S, {NP, NP}>

This is actually an abbreviation for the structure in 4.22

4.22 < [-N, +V, +Phrasal], { < [+N, -V, +Phrasal], <> >,
< [+N, -V, +Phrasal], <> > } >

where "NP" is an abbreviation for

4.23 < [+N, -V, +Phrasal], <> >

Now, assume a feature GR (with values "subject" and "object") which is instantiated, in addition to the directionality marking, on all NPs.[4] Using the following abbreviations:

4.24 a NPsubject = < [+N, -V, +Phrasal, GR subject], <> >

b NPobject = < [+N, -V, +Phrasal, GR object], <> >

The set of potential ordered local structures which results from this instantiation is given in 4.25.

4.25 { <S, </NPsubject, /NPobject>>,
<S, </NPobject, /NPsubject>>,
<S, </NPsubject, \NPobject>>,
<S, <\NPobject, /NPsubject>>,
<S, <\NPsubject, /NPobject>>,
<S, </NPobject, \NPsubject>>,
<S, <\NPsubject, \NPobject>>,
<S, <\NPobject, \NPsubject>> }

Now assume an L_p statement of the form "/NP". According to the semantic interpretation rules for basic expression given in the previous Chapter, /NP will be true of a local structure if and only if there is some NP in that structure that is marked "/". But none of the local structures in 4.25 contain simply "NP". The semantic rules must, therefore, refer to the fact that primitive semantic types match if they unify but all other aspects of categories must be exactly the same, including the directional marking.

Before giving a final definition of expansion and final versions of the semantic interpretation rules for basic L_p expressions it will simplify the statement of these rules to define a general notion of category matching which brings together all aspects of category structure so far touched upon.

The category matching operator $\not\curlyvee$ (read "matches") returns true whenever two local structures match in an appropriate way, otherwise it returns false. 4.26 gives a recursive definition of $\not\curlyvee$. It is complicated by the fact that local structures contain primitive semantic types, n-tuples of local structures and atomic designated symbols. In the definition, the term *head(a)*, for some n-tuple a = <X, Y, ...>, refers to "X" and the term *tail(a)* refers to the n-tuple <Y, ...>.

4.26 where a,b are categories of the grammar,

$a \forall b = T$

if head(a), head(b) are primitive semantic types

then head(a) $\sqcup$ head(b)

and

tail(a) $\forall$ tail(b)

elseif head(a), head(b) are designated symbols of the grammar

then head(a) = head(b)

and

tail(a) $\forall$ tail(b)

elseif head(a), head(b) are n-tuples

then for every element ε in head(a) and ε' in head(b)

$\varepsilon \forall \varepsilon'$

and

tail(a) $\forall$ tail(b)

else $a \forall b = F$

An expansion of an ordered pair $\Omega = \langle a, b \rangle$, where a,b are categories of the grammar, is an ordered n-tuple ($n \geq 2$) in which the sequence $\langle ..., a', ..., b', ... \rangle$ such that $a \forall a'$ and $b \forall b'$ appears but there is no sequence $\langle ..., b', ..., a', ... \rangle$.

With these augmented definitions in mind, the final versions of the semantic interpretation rules for basic expressions of L_p can now be given.

4.27 $[\![a]\!]_{\gamma \in_T M} = 1$

iff either there is a category a' in γ:2 such that $a \forall a'$ or there is some directional category "\b" or "/b" in γ:2 such that for its non-directional equivalent (ie. "b") $a \forall b$.

4.28 $[\![\backslash a]\!]_{\gamma \in_T M} = 1$

iff there is some category in γ:2 which has a non-directional equivalent, b, such that $a \mathrel{\forall\!\!\!\!-} b$, and which is also $\langle \ldots, \backslash, \ldots \rangle$.

4.29 $[\![/a]\!]_{\gamma \in_T M} = 1$

iff there is some category in γ:2 which has a non-directional equivalent, b, such that $a \mathrel{\forall\!\!\!\!-} b$, and which is also $\langle \ldots, /, \ldots \rangle$.

4.30 $[\![\langle \sigma, \tau \rangle]\!]_{\gamma \in_T M} = 1$

iff $[\![\sigma]\!]_{\gamma \in_T M} = 1$ and $[\![\tau]\!]_{\gamma \in_T M} = 1$ and γ:2 is an expansion of $\langle \sigma, \tau \rangle$.

4.31 $[\![f{:}\alpha]\!]_{\gamma \in_T M} = 1$

iff α is a designated symbol of the grammar and $\gamma = \langle \ldots, \alpha, \ldots \rangle$.

The set of permissible local structures admitted by the L_p statement. /NP, given above is, therefore,

4.32 { <S, </NPsubject, /NPobject>>,
<S, </NPobject, /NPsubject>>,
<S, </NPsubject, \NPobject>>,
<S, <\NPobject, /NPsubject>>,
<S, <\NPsubject, /NPobject>>,
<S, </NPobject, \NPsubject>> }

Giving the word order possibilities in 4.33

4.33 a [f NPsubject] NPobject

b [f NPobject] NPsubject

c NPobject [f NPsubject]

d [NPobject f] NPsubject

e [NPsubject f] NPobject

f NPsubject [f NPobject]

where the bracketing indicates the *first* argument to combine with the function.

4.4 Feature Passing in Categorial Grammar

This section addresses the issue of passing morphosyntactic features between categories in an L_p factored Categorial Grammar. The principles are straightforward and reminiscent of those used in Generalised Phrase Structure Grammar. Consider the sentence in 4.34.

4.34 The book is lost

Unordered and ordered local structures for each of the words in 4.34 are given in Figure 4.2 below. The first entry in each case is the abbreviated unordered local structure assigned in the categorial lexicon. Variable assignments are suppressed because of the use of abbreviations, but they correspond to those given in the third entry for each word. The second entry is an abbreviated form of the ordered local structure. The third is the fully specified category with explicit feature marking.

As indicated above, the use of the X-bar style N, V features in the feature lists for primitive semantic types is not crucial in any way, rather merely illustrative of the possibilities. The use of these features carries the same advantages and disadvantages as it does in other syntactic theories. The singular noun *book* has number and person features specified. Adjectives, on the other hand, which are taken to be functions from noun denotations into noun denotations, have variable number and person features which are passed between the argument and the head. The assumption behind the notation is that whenever an argument is accepted for which the function possesses features with variable values (indicated with lower case Greek letters) those values are instantiated (if possible) throughout the function, including the head and all arguments in the argument stack. In this way information can be passed between arguments to the function or from an argument to the head of the function. This reflects closely, the feature passing conventions found in Generalised Phrase Structure *the* a <NP, {N}>

b <NP, </N>>

c <[+N, -V, +phrasal, Num α, Pers β],
< <[+N, -V, -Phrasal, Num α, Pers β],
<>, /> >>

book a <N, {}>

b <N, <>>

c <[+N, -V, -Phrasal, Num sg, Pers 3], <> >

is a <S, {<N, {N}>, NPsubject}>

b <S, </<N, </N>>, \NPsubject>>

c <[-N, +V, +Phrasal],
< <[+N, -V, -Phrasal],
< <[+N, -V, -Phrasal], <>, />>, / >
<[+N, -V, +Phrasal, Num sg, Pers 3,
GR subject], <>>, \ >>

lost a <N, {N}>

b <N, </N>>

c <[+N, -V, -Phrasal, Num α, Pers β],
< <+N, -V, -Phrasal, Num α, Pers β], <>, />

Figure 4.2
Ordered Local Structures from a Sample Lexicon

Grammar where information can be passed between the daughters and between the daughters and the mother of local trees.

The copula is analysed as taking an adjective to its right and a noun phrase subject to its left. Agreement with the subject is guaranteed by having the relevant number and person values on the subject NP argument.

Finally, the definite article is, like the adjective, a feature passing category.

It is a straightforward matter to see how the sentence in 4.34 would be analysed with these lexical entries and a rule of Functional Application. A full derivation, showing instantiation of feature variables in bold is given in 4.35 below.

4.35

FA: <[-N, +V, +Phrasal], <> >

FA: <[+N, -V, +phrasal, Num **sg**, Pers **3**], <> >

<[+N, -V, +phrasal, Num **α**, Pers **β**],
< <[+N, -V, -Phrasal, Num **α**, Pers **β**],
<>, /> >> ∋ the

<[+N, -V, -Phrasal, Num sg, Pers 3], <> > ∋ book

FA: <[-N, +V, +Phrasal], < <[+N, -V, +Phrasal,
Num sg, Pers 3, GR subject],
<>>, \ >>

<[-N, +V, +Phrasal],
< <[+N, -V, -Phrasal],
< <[+N, -V, -Phrasal], <>, />>, / >
<[+N, -V, +Phrasal, Num sg,
Pers 3, GR subject],
<>>, \ >> ∋ is

<[+N, -V, -Phrasal, Num **α**, Pers **β**],
< <+N, -V, -Phrasal, Num **α**, Pers **β**],
<>, /> ∋ lost

4.5 Metacategories and English Auxiliaries

The purpose of this section is to explore some natural language data with the theory so far introduced and to show how metacategory definition helps in the formation of generalisations over category structure. The section focuses on the analysis of auxiliary verbs and auxiliary sequences in English transitive sentences. A small lexicon for transitive verbs and their arguments is given in Figure 4.3 using abbreviated category structures. Relevant L_p statements for this grammar are given in 4.36

John, Mary	NP
the	<NP, {N}>
hall, bathroom	N
painted, decorated	<S, {NP, NP}>

Figure 4.3
A Small Categorial Lexicon for English Transitive Sentences

4.36 a /N

b \NPsubject & /NPobject

These are straightforward. The only aspect of them to which attention needs to be drawn is the fact that 4.36b imposes no ordering over the argument stack for transitive verbs. It simply constrains the subject to precede the verb and the direct object to follow it. Consequently, two ordered local structures are admitted for transitive verbs. These are spelled out in 4.37.

4.37 a <S, <\NPsubject, /NPobject>>

b <S, </NPobject, \NPsubject>>

4.37a is the category for the verbal function which combines first with a subject Noun Phrase to the left and then with an object to the right, giving a derivation [SV]O in which coordination of the verb and subject is possible. 4.37b gives a derivation S[VO] in which coordination of the verb and object is possible.

Two aspects of the syntax of auxiliaries are of interest in this section:

(1) the ordering of auxiliary verbs in sequences; and

(2) the combinatorial properties of auxiliary verbs and the L_p statements

required to describe them.

The approach adopted here to the well-known ordering constraints on auxiliary verbs is adapted from the account that has been developed within the Generalised Phrase Structure Grammar paradigm. This is a morphosyntactic rather than semantic theory which expressed the fact that each auxiliary verb constrains the verb following, whether an auxiliary or main verb, to a particular morphosyntactic realisation. The sequence for English is: finite - base - past participle - present participle. 4.38 illustrates this ordering.

4.38 John may have been painting the hall

Principal sources for this account within the Generalised Phrase Structure Grammar framework are Gazdar (1982), Gazdar, Pullum and Sag (1982) and Warner (1984). No attempt is made here to present the details of this theory since that is fully covered in the references cited. The account transfers over to the present categorial theory in the following way. An auxiliary verb takes a verbal function (ie. a function category headed by S) to its right and an NP to its left. The relevant morphosyntactic features are instantiated on the head of the auxiliary category and the head of the verbal argument. This is illustrated in 4.39 for the category associated with *may*.

4.39 may = <S[finite], </<S[base], <\NPsubj>>, \NPsubj>>

The category in 4.52 will combine first with a verbal function looking for a subject NP to its left and whose morphosyntactic realisation is a base form, and then with the subject NP to the left. By stipulating these features in the way indicated, the correct constraints on sequences of auxiliaries can be imposed.

The second issue - the combinatorial properties of auxiliary verbs - raises a number of interesting problems. Figure 4.4 contains some simple data for subject-auxiliary-verb sequences with *have*. The square brackets are used to indicate coordination possibilities. All coordination possibilities are in fact grammatical. Awkwardness in some of the sentences with coordination can be

a [John has] painted the hall
[John has and Mary has] painted the hall

b John [has painted] the hall
John [has painted and has decorated] the hall

c John has [painted the hall]
John has [painted the hall and decorated the bathroom]

d [John has painted] the hall
[John has painted and Mary has decorated] the hall

e John [has painted the hall]
John [has painted the hall and has decorated the bathroom]

Figure 4.4
The Combinatorial Properties of "have" in Subject-Auxiliary-Verb sequences

explained by a general stylistic inhibition on the repetition of the same auxiliary within a single coordinated phrase. When the auxiliaries are different, the sentences are much more acceptable, so that the sentence in Figure 4.4a is much better as 4.40.

4.40 John may have and Mary certainly has contributed to the fund

Consider what is required for each sentence in Figure 4.4. The first requires the auxiliary to combine with the subject before it combines with the verb or verb phrase. Hence the category in 4.41 is required (ignoring the morphosyntactic features).

4.41 $<S, <\backslash NPsubj, /<S, <\backslash NPsubj>> >>$

The auxiliary verb in the second sentence combines directly with the transitive verb, thereby requiring the category in 4.42.

4.42 <S, < /<S, <\NPsubj, /NPobj> >, \NPsubj, /NPobj> >

Clearly, however, the auxiliary can combine with an intransitive verb or with a verb phrase having the same category as an intransitive verb. Consequently, the category required for the auxiliary verb in the third sentence is that in 4.43.

4.43 <S, < /<S, <\NPsubj> >, \NPsubj> >

The fourth sentence has two derivations: one involving the auxiliary combining with the subject NP followed by the main verb; the other involving combination with the main verb first. The first of these requires the category in 4.44.

4.44 <S, < \NPsubj, /<S, <\NPsubj, /NPobj> >, /NPobj> >

The second requires the category in 4.42.

Likewise, the fifth sentence has two derivations: one requires the category in 4.43; the other, the new category in 4.45.

4.45 <S, < /<S, <\NPsubj, /NPobj> >, /NPobj, \NPsubj> >

These orderings can be summarised in the following way. Assume the abbreviation IVP (intransitive verb phrase)[5] for the category <S,<\NPsubject> > and the abbreviation TVP (transitive verb phrase) for the category <S,<\NPsubject,/NPobject> >. The allowed orderings are given in 4.46 using these abbreviations.

4.46 a <\NPsubject, /IVP> = 4.41

b </TVP, \NPsubject, /NPobject> = 4.42

c </IVP, \NPsubject> = 4.43

d <\NPsubject, /TVP, /NPobject> = 4.44

e </TVP, /NPobject, \NPsubject> = 4.45

It will be clear that these are only a small fraction of the logically possible combinations of these categories with arbitrary directionality markings.

The categories in 4.41 to 4.45 have a significant characteristic in common. They are all functions which take as one of their arguments a verbal function and the category with that verbal function removed has the same arguments as the function argument (though not necessarily in the same order on the stack). In other words, there is a dependency between the verbal function argument of the auxiliary and the auxiliary function itself.[6] To accommodate this observation assume a meta-definition of unordered category structure to the effect that one type of category - *C* - admitted by the grammar has the structure given in 4.47.

4.47 *C* = <h, {..., *K*, ...}>

where, *K* is a complex category and
head(*K*) = head(*C*)
argset(*K*) = argset(*C*) - *K*

The interpretation of the notation here is as follows: ***head(X)*** refers to the first element in a category pair; ***argset(X)*** refers to the second element in the category pair.

English auxiliary verbs are a special case of this category type, being assigned the following unordered local structure in the lexicon:

4.48 <S, {<S, {NPsubj, W}>, NPsubj, W}>

where the variable "W" ranges over a subset of arguments in the argsets. Using the notation in 4.47 above, K = <S, {NPsubj, W}>, head(K) = S, argset(K) = {NPsubj, W} and argset(C) = {<S, {NPsubj, W}>, NPsubj, W}.

On the assumption that the variable "W" is available to L_p statements, the expression in 4.49 below defines the relevant constraint on the instantiation to ordered local structures.

4.49 < /<S, <\NPsubj, W>> , W >

This constraint says that in any local structure for auxiliaries, the verbal argument must come from the right and combine with the auxiliary function before any other non-subject argument combines with the auxiliary function. The subject must come from the left and may combine with the auxiliary before the verbal argument. The direction of the object - consistently from the right - is determined by the linear ordering constraints on the transitive verbal argument and is "picked up" by W.

This L_p statement captures exactly the data in 4.46 and therefore induces only the categories in 4.41 to 4.45.

To illustrate the effects of this, 4.50 is a derivation for the fourth sentence in Figure 4.4.

```
4.50  FA:   <S, <>>
      FA:        <S, </NPobject>>
                      NP ∋ John
      FA:             <S, <\NPsubject, /NPobject>>
                           <S, </<S, <\NPsubj, W>>,
                                \NPsubj, W>> ∋ has
                           <S, <\NPsubject, /NPobject>>
                                                 ∋ painted
      FA:        NP
                   <NP, </N>> ∋ the
                   N ∋ room
```

4.6 Summary

This Chapter has demonstrated that the factoring out of linear ordering information is made possible in Categorial Grammars by treating the categories as local structures over which L_p statements are defined. Such L_p statements act as a filter on the mapping from unordered to ordered local structures. In order to establish this fact about Categorial Grammars it was necessary to demonstrate that Categorial Grammar categories can be re-interpreted as local structures without loss of information (Section 4.2) and that L_p statements can be defined over the set of ordered local structures. The required basic expressions of L_p together with a preliminary version of their semantic interpretation was given in Section 4.3.

Given this restructuring of categories and the new L_p logic, a number of specific issues needed to be addressed.

There is an interesting interaction between the L_p logic and coordination potential in the new new model which was discussed and illustrated in Section 4.4.1. In the same section new definitions of the two most frequent combinators were given which did not assume Curried functions. An algorithm for the combination of categories which generalises over Functional Application and Composition was presented. This is in fact the first step towards the fully general statement of categorial combination presented in Chapter 6.

Unification was discussed in Section 4.4.2. This is an important topic but in this section was limited to the unification of simple feature name value matrices representing primitive semantic types. The use of complex symbols to represent primitive semantic types in this way, while complicating the definition of the L_p logic, allows for more radical generalisation over linear ordering to be made than would otherwise be possible.

With the introduction of complex symbols representing the primitive semantic types of the grammar it is natural to consider ways that morphosyntactic information might be passed from category to category during a derivation. A simple feature passing convention was introduced in Section 4.4.3 which allows for this. An example of this mechanism was presented.

Finally, data involving the auxiliary *have* in English was used to illustrate the advantage of meta-category definition, a mechanism for generalisation which will be more fully exploited in the next Chapter.

4.7 Notes

[1] This idea coincides with the empty SUBCAT feature for "saturated" categories such as S, NP in Head-Driven Phrase Structure Grammar (see Pollard and Sag 1987, page 68). For a related idea in Generalised Phrase Structure Grammar, where SUBCAT takes a set rather than a stack see Gunji (1986a, 1986b).

[2] See Gazdar, Klein, Pullum and Sag (1985, Chapter 2) for an overview of a theory of syntactic features which, though different in many respects to the account offered here, is essential background.

[3] Shieber (1986) is an excellent introduction to the unification formalism. See Karttunen (1984), Shieber (1984) and Pereira and Shieber (1984) for background to computational and semantic aspects of unification. The following theories make extensive or exclusive use of unification over attribute-value matrices: Generalised Phrase Structure Grammar (Gazdar, Klein, Pullum and Sag (1985)); Head-Driven Phrase Structure Grammar (Pollard and Sag (1987)); Functional Unification Grammar (Kay (1983, 1985)). And within the family of Categorial Grammars: Unification Categorial Grammar (Zeevat (1987)) and Categorial Unification Grammar (Uszkoreit (1986)). See Haddock, Klein and Morrill (1987) for a relevant collection of papers.

[4] Throughout Chapters 4 to 7 Grammatical Relations are taken as primitve features on Noun Phrase arguments to functions. See Bresnan (1982, pages 344-348) for a defence of Grammatical Relations as primitives in the context of Lexical-Functional Grammar. In Chapter 8 a decomposition of Grammatical Relations into two more primitive binary features is suggested motivated by the desire to express generalisations over the personal and dative *a* in Spanish.

[5] The use of VP in these abbreviations is not to be taken as meaning that a Verb

Phrase category or constituent is required for the analyses of auxiliaries. It will be clear from the tenor of the present discussion (and even more so when auxiliaries in Spanish are discussed in Chapter 5) that no commitment is made to the existence of Verb Phrases.

[6] This observation is consistent with a typology of Categorial Grammar categories which includes categories of the form X|X (see Bach 1983a, 1983b). The general idea is that the category structure itself encodes the notion that the features of a phrase will frequently be identical to the features on the head of the phrase. See Bouma (1988) for a detailed examination of categories of the form X[F]|X[F] - referred to as modifiers - in a Categorial Unification grammar framework.

CHAPTER 5

THE CATEGORIAL ANALYSIS OF SUBJECT AND OBJECT ORDER IN SPANISH

5.1 Introduction

This chapter analyses the data presented in Chapter 2 within the framework of an L_p factored Categorial Grammar. The focus of interest is the set of L_p statements required to define the constituent orderings for verbal functions of various types. In all cases, the standard assumption is made that nominal arguments are basic categories of the grammar and that verbs are complex functions. Other complex categories which appear in the data - such as those associated with clitics, prepositions and the direct object marker - are not discussed in detail in this chapter but are deferred to Chapter 8.

In Section 5.2 two approaches to the analysis of word order in transitive and di-transitive sentences are discussed. The first of these involves a simple L_p statement for the word order appealing only to conventional syntactic categories and Grammatical Relations. The second considers the problem of accounting for word order *in context* through appeal to discourse saliency information. This analysis demonstrates how a set of complex ordering facts can be expressed through a single L_p expression.

In the following sections a number of aspects of the data from Chapter 2 are discussed individually. First of all, in Section 5.3, an L_p statement for verbs taking sentential complements is given which takes into account the ordering of both subject and complement relative to the verbal function. Secondly, in

Section 5.4, the ordering constraints on auxiliaries are accounted for with two L_p statements. This analysis develops the notion of a meta-category definition introduced in relation to English auxiliaries in Chapter 4. Finally, in Section 5.5, subject control verbs are briefly discussed. The Chapter ends with a summary and comparison of the various ordering constraints on verbal complexes.

5.2 Transitive and Di-transitive Sentences

This section examines word order from two perspectives: syntactic and pragmatic. Both use the formal apparatus for factoring out linear ordering information developed in the last chapter.

The syntactic constraints on word order at the sentence level can be expressed very simply. This statement about the ordering of major constituents using syntactic categories together with grammatical relations says little, however, about the pattern of acceptability in terms of discourse saliency. Consequently, a detailed investigation of one informant's responses using a restricted preceding dialogue context (referred to here as *micro-dialogues*) was carried out. The results of this investigation are presented in Section 5.2.2 below together with an analysis in terms of the L_p factored Categorial Grammar.

5.2.1 A Syntactic Analysis

Figure 5.1, repeated from Chapter 2, below summarises the word order possibilities for transitive and di-transitive sentences in Spanish, where the curly brackets indicate that the constituents they enclose may occur in any order relative to one another.

There is clearly a regularity here to the effect that the verb cannot occur as the last major constituent in a sentence although this generalisation is only applicable when there is a direct object present.[1] For intransitive sentences, the

Transitive sentences	Di-transitive sentences
S V O	S [dat] V {O, I}
V {S, O}	O [dat] V {S, O}
O V S	I [dat] V {S, O}
	{S, I} [dat] V O
	{S, O } [dat] V I
	{I, O } [dat] V S

Figure 5.1
Word Order in Spanish Transitive and Di-transitive Sentences

order SV is, of course, perfectly acceptable, as is the order VS. The precise manner of stating this constraint depends upon the category associated with the indirect object in Spanish. Following the evidence from Suñer presented in Chapter 2, it is assumed here that this category is NP marked for the feature [GR: iobj]. Given this assumption, the L_p statement in 5.1 induces the correct grammaticality patterns.

5.1 NPobject $\supset$ /NP

This says that if there is a Noun Phrase marked [GR: dobj] in the argument stack of the verbal function, there must be at least one Noun Phrase with directionality marking "/" (ie. post-verbal). The effects of this ordering constraint on simple transitive verbs are illustrated in Table 5.1 below, where S = NPsubject, O = NPobject and the results of evaluating the L_p constraint against each Argument Stack are given. The last column indicates the informant grammaticality judgement summarised in Figure 5.1 above. It can clearly be seen that these judgements correspond with the truth-value assignment for the L_p expression.

Argument Stack	Surface Order	L_p constraint			
		NPobject	$\supset$	/NP	
/S /O	[V S] O	T	T	T	ok
/S \O	O [V S]	T	T	T	ok
\S /O	[S V] O	T	T	T	ok
\S \O	O [S V]	T	F	F	*
/O /S	[V O] S	T	T	T	ok
/O \S	S [V O]	T	T	T	ok
\O /S	[O V] S	T	T	T	ok
\O \S	S [O V]	T	F	F	*

Table 5.1
The Effects of the L_p Constraint in 5.1
for Simple Transitive Sentences

5.2.2 A Discourse-Based Analysis

The simple L_p statement in 5.1 provides the basic syntactic constraint for Spanish intransitive, transitive and di-transitive sentences. However, it leaves a great deal unexplained. This section will explore the possibility of defining word order using pragmatic and syntactic concepts in combination. The purpose is twofold. On the one hand, the analysis presented in this section serves to illustrate just how effective the use of L_p is in systematically reducing complex data to a few very general statements of constraints on order. Secondly, the section draws attention to an important aspect of Spanish syntax: that word order and pragmatic/discourse functions interact in complex ways.[2] The use of an applicative system such as Categorial Grammar with a factored linear ordering component allows various syntactic and discourse-oriented influences on word order to be clearly identified.

It is easy to demonstrate that conventional syntactic distinctions are not adequate for the analysis of word order taken within a discourse context. Consider the micro-dialogue in 5.2 below. (Note that throughout this section immediately preceding dialogue context will be given in English.)

5.2 a -- Who arrested Mary yesterday?

b -- Los soldados arrestaron a María ayer.
(The soldiers arrested Mary yesterday)

In 5.2b, *los soldados* is NEW information, the other constituents - *arrestaron* and *a María* - are GIVEN information since both are directly referred to in the immediately preceding context. The discourse saliency pattern of 5.2b (ie. the assignment of NEW and GIVEN to constituents in the sentence) is therefore as given by 5.3.

5.3 Los soldados	arrestaron	a María ayer.
N	G	G

Where N = NEW and G = GIVEN information.

A simple generalisation might take this pattern of NEW preceding GIVEN to be characteristic of Spanish transitive sentences generally. Such a conclusion would, however, be premature. Consider the micro-dialogue in 5.4.

5.4 a -- What did John do?

b -- ?Golpeó Juan al soldado
(John struck the soldier)

In 5.4b, the discourse saliency pattern is that in 5.5.

5.5	?Golpeó	Juan	al soldado
	--------	-------	--------------
	N	G	G

This is the same pattern as in 5.3, but here the sequence is doubtfully grammatical in context. This does not, of course, follow from the order of the grammatical relations since the sequence V S O is perfectly acceptable at the syntactic level in Spanish. Rather, it follows from the interaction of grammatical relations and discourse saliency information: the V S O sequence which carries the information assignment N G G is only marginally grammatical.

The purpose of this section is to show that this interaction of different kinds of linguistic information can be described in an L_p factored Categorial Grammar of the sort under development in this thesis. Further, it is possible to express generalisations about the data using the linear precedence logic. This section demonstrates the power of the current formalism through an experiment in the analysis of informant-supplied data.

As a preliminary to the investigation, a body of raw data was collected from an informant. This consisted of grammaticality judgements on sets of potential answers to questions, given a restricted preceding context. The answers were designed to cover all the syntactically acceptable sequences of S, V and O, that is: SVO, VSO, VOS, and OVS. No clitic pronouns were included in the sentences. The informant reponses are presented in detail in the Appendix 1.

The first step in analysing the raw data was to reduce the sequences of answers to patterns indicating the grammatical relations of the noun phrases to the finite verb and at the same time assign NEW, GIVEN information to each constituent in each sentence. The results of this step are given in Appendix 2.

The next step in uncovering underlying patterns involves reorganising the data to allow for comparison of the different information saliency patterns corresponding to each word order possibility. The results of this reorganisation of the data are presented in Tables 5.2 to 5.5 below.

Discourse saliency Pattern			Judgement	References to Contexts
N	N	N	ok	A
N	N	G	ok	B
N	G	N	ok	C
N	G	G	ok	D, E
G	N	N	ok	F, G, H
G	N	G	ok	I
G	G	N	ok	J, K, L
G	G	G	ok	M

Table 5.2
S V O Order

Discourse saliency Pattern			Judgement	References to Contexts
N	N	N	ok	A
N	N	G	*	B
N	G	N	? to *	F, G, H
N	G	G	?	I
G	N	N	*	C
G	N	G	ok to ?	D, E
G	G	N	ok	J, K, L
G	G	G	ok	M

Table 5.3
V S O Order

Discourse saliency Pattern			Judgement	References to Contexts
N	N	N	ok	A
N	N	G	?*	F, G, H
N	G	N	*	B
N	G	G	?	I
G	N	N	*	C
G	N	G	ok	J, K, L
G	G	N	ok to ?	D, E
G	G	G	?	M

Table 5.4
V O S Order

Discourse saliency Pattern			Judgement	References to Contexts
N	N	N	?*	A
N	N	G	?*	F, G, H
N	G	N	?*	C
N	G	G	ok	J, K, L
G	N	N	*	B
G	N	G	*	I
G	G	N	? to *	D, E
G	G	G	*	M

Table 5.5
O V S Order

In the case of the SVO pattern, there are no constraints on the assignment of discourse saliency information: an SVO pattern is acceptable in *any* dialogue context. The VSO and VOS grammaticality patterns show exactly the same discourse saliency: the verb and at least one of its arguments must be GIVEN by the context. The same constraint applies to the OVS pattern which means that in this case the verb and its following subject must be GIVEN.

The results of this analysis are summarised in Table 5.6 below. This summary covers most of the data given in the tables above except for the following:

5.6	a	VSO	NNN is acceptable
	b	VOS	NNN is acceptable
			GGG is not entirely acceptable
	c	OVS	GGG is not acceptable

All of these are rather special situations for either initiating discourse or responding to yes/no questions. It is assumed here that these situations involve other pragmatic factors which override the generalisations in Table 5.5.

These facts may be expressed in the theory of Categorial Grammar presented in the last chapter in the following way. Assume the following lexical entry for transitive verbs:

5.7 <S, {NP, NP}>

Further, assume that the GR (ie. grammatical relation) feature is instantiated

S V O	all distributions of discourse saliency information are acceptable.
V S O	the verb and at least one argument must be GIVEN information.
V O S	the verb and at least one argument must be GIVEN information.
O V S	the verb and its following subject must be GIVEN information.

Table 5.6
Summary of Word Order Possibilities

freely on the NP arguments before the application of the L_p constraints and that *one* of the designated symbols GIVEN, NEW is freely instantiated on all function categories. The following L_p statement constrains the ordered local structures to just those compatible with the generalisations in Table 5.5 above.

5.8 /NPsubject ⊃ *f*:GIVEN & /<..., GIVEN, ...>

This expression is read as follows. If the function's argument stack contains a rightward looking subject NP then the discourse saliency information on the function itself must be GIVEN and there must be at least one rightward looking GIVEN constituent in the function's argument stack. Otherwise, all distributions of grammatical relations and discourse saliency information are acceptable.

Table 5.7 illustrates the way this L_p statement filters out unacceptable instantiations of the unordered local structure associated with transitive verbs for just three examples. Appendix 3 gives tables for all 48 word orders which are grammatical with respect to the first L_p statement in 5.1. In table 5.7, the argument stack is given in the first column with the assignment of Given (G)

Argument Stack	**Surface Order**	**L_p Constraint**				
		/NPsubject	⊃	*f*:GIVEN	&	/<.., GIVEN,..>
<\S, /O,> N G N	[S V] O G N N	F	**T**	F	F	F
</S, /O,> G G N	[V S] O G G N	T	**T**	T	T	T
</S, /O,> N G N	[V S] O N G N	T	**F**	F	F	T

Table 5.7
Truth-table Evaluation of the L_p Statement
in 5.8 Against Some Potential Ordered Local Structures

and New (N) beneath each argument. There is a similar assignment immediately following the angle brackets which marks the discourse saliency of the function itself. The second column, as before, shows the surface orderings and distributions of discourse saliency information associated with the particular argument stack.

5.3 Verbs Taking Sentential Complements

Figure 5.2 summarises the data for sentential complements from Chapter 2 and adds one further complication: the fronting of the sentential complement before the matrix verb. The last two sentences in Figure 5.2 illustrate these possibilities.

(John says that Mary bought a house)

Smatrix Vmatrix [COMP S V O]	**Juan** dice [que María compró una casa]
Vmatrix Smatrix [COMP S V O]	Dice **Juan** [que María compró una casa]
Vmatrix [COMP S V O] Smatrix	Dice [que María compró una casa] **Juan**
*Vmatrix [COMP Smatrix S V O]	*Dice [que **Juan** María compró una casa]
*Vmatrix [COMP S V Smatrix O]	*Dice [que María compró **Juan** una casa]
[COMP S V O] Vmatrix Smatrix	[que María compró una casa] dice Juan
[COMP S V O] Smatrix Vmatrix	[que María compró una casa] Juan (lo) dice

Figure 5.2
The Order of the Subject Relative to Verbs Taking Sentential Complements

The only order which is not permitted is verb final when the subject linearly precedes the sentential complement. Otherwise, the generalisation given in Chapter 2 with respect to the order of the subject is clearly apparent here: the subject can occur in any position relative to its matrix verb but cannot separate constituents in the embedded clause. Notice that the clitic pronoun in the last sentence in Figure 5.2 is optional. This will not be taken into account in the linear ordering constraint offered below.

The full grammaticality pattern for these data taking sentential complement fronting into account but leaving out the ungrammatical sequences where the matrix subject is embedded within the complement since these could not be generated by the grammar is shown in Figure 5.3.

The L_p constraint required for these data is given in 5.9.

5.9 $(\backslash\text{SComp} \ \& \ \langle\text{SComp, NPsubject}\rangle) \supset /\text{NPsubject}$

Table 5.8 gives the truth-table evaluation of this constraint against the orderings in Figure 5.3 where S = NPsubject and S' = SComp.

NPsubject V SComp

*NPsubject SComp V

V NPsubject SComp

V SComp NPsubject

SComp V NPsubject

SComp NPsubject V

Figure 5.3
Grammaticality Pattern for
Verbs Taking Sentential Complements

Argument Stack	Surface Order	L_p Constraint				
		(\S'	&	<S',S>)	⊃	S
\S, /S'	[S V] S'	F	F	F	**T**	F
/S', \S	S [V S']	F	F	T	**T**	F
\S', \S	*S [S' V]	T	T	T	**F**	F
/S, /S'	[V S] S'	F	F	F	**T**	T
/S', /S	[V S'] S	F	F	T	**T**	T
\S', /S	[S' V] S	T	T	T	**T**	T
/S, \S'	S' [V S]	T	F	F	**T**	T
\S, \S'	S' [S V]	T	F	F	**T**	F

Table 5.8
Truth-table Evaluation of the L_p Statement in 5.9 Against Some Potential Ordered Local Structures

5.4 Subjects and Auxiliary Verbs

Auxiliary verbs pose more difficult and interesting problems. The key data, again repeated from Chapter 2, are given in Figure 5.4. The important distinction between *haber* and *estar* is with respect to the position of the subject.

Section 4.5 discussed the analysis of English auxiliary verbs drawing attention to constraints on their morphosyntactic realisation in sequences and their combinatorial potential. Since Spanish auxiliaries show similar morphosyntactic constraints to English auxiliaries, this aspect will not be pursued here. The combinatorial and word order potential of auxiliary verbs are, however, of particular interest.

Auxiliary verbs in Spanish are taken to be instantiations of the metacategory definition given in 4.5 and repeated here as 5.10.

haber	
	(John has sung the song)
S Aux_{haber} V O	Juan ha cantado la canción
Aux_{haber} V S O	ha cantado Juan la canción
Aux_{haber} V O S	ha cantado la canción Juan
*Aux_{haber} S V O	*ha Juan cantado la canción
O Aux_{haber} V S	una canción ha cantado Juan
*O Aux_{haber} S V	*la canción ha Juan cantado
estar	
	(John is singing the song)
S Aux_{estar} V O	Juan está cantando la canción
Aux_{estar} V S O	está cantando Juan la canción
Aux_{estar} V O S	está cantando la canción Juan
Aux_{estar} S V O	está Juan cantando la canción
O Aux_{estar} V S	una canción está cantando Juan
?O Aux_{estar} S V	una canción está Juan cantando

Figure 5.4
The Order of the Subject Relative to Verbs with Auxiliaries

5.10 <S, {<S, {NPsubject, W}>, NPsubject, W}>

Where "W" ranges over a subset of the arguments in the argument stack of the function. The way this works out is indicated in 5.11 below. Sample ordered local structures associated with each verb are given in 5.11b and 5.11c. The value of the variable "W" is given in 5.11d. The result is given in 5.11e.

5.11 a ha cantado
(has sung)

b ha = <S, </<S, </NPsubject, W> >, /NPsubject, W> >

c cantado = <S, </NPsubject, /NPobject> >

d W = /NPobject

e <S, </NPsubject, /NPobject> >

In order to establish the L_p constraints for these two auxiliaries it is necessary to examine the grammatical status of all relevant surface orderings. The details of this are presented in Table 5.9.

As 5.10 makes clear, auxiliaries are interpreted as functions taking three arguments. One argument is the verbal function which the auxiliary always expects to its right; another is the subject NP of that verbal argument; and the last is a variable ranging over all the other arguments of the main verb with which the auxiliary combines. There are two dimensions of freedom with respect to these arguments: the directionality marker they take and their relative ordering on the argument stack of the auxiliary. The first column in Table 5.9 represents the case where an auxiliary combines with a transitive verb, therefore "W" is set to "NPobject". "S" refers to the NPsubject, "f" refers to the main verb function and "O" refers to the NPobject. The sets in this column illustrate all the possible orderings of these constituents given the fact that the auxiliary *always* looks for its main verb function to its right and the L_p constraints on transitive verbs discussed earlier are such as to disallow any other combinations on directionality marking on the subject and object.

The second column shows the surface ordering of the constituents and, in addition, the bracketing indicates the order of combination, thereby indicating coordination potential. The last two columns give grammaticality judgements from a native speaker informant for the sort of simple sentence shown in Figure 5.4

Argument Stack	Surface Order	*haber*	*estar*
Group 1			
/S /f /O	[[Aux S] V] O	*	ok
/S /f \O	O [[Aux S] V]	*	ok
\S /f /O	[[S Aux] V] O	ok	ok
Group 2			
/f /S /O	[[Aux V] S] O	ok	ok
/f /S \O	O [[Aux V] S]	ok	ok
/f \S /O	[S [Aux V]] O	ok	ok
Group 3			
/S /O /f	[[Aux S] O] V	*	*
/S \O /f	[O [Aux S]] V	*	?*
\S /O /f	[[S Aux] O] V	*	*
Group 4			
/f /O /S	[[Aux V] O] S	ok	ok
/f \O /S	[O [Aux V]] S	ok	ok
/f /O \S	S [[Aux V] O]	ok	ok
Group 5			
/O /f /S	[[Aux O] V] S	*	*
\O /f /S	[[O Aux] V] S	*	?*
/O /f \S	S [[Aux O] V]	*	*
Group 6			
/O /S /f	[[Aux O] S] V	*	*
\O /S /f	[[O Aux] S] V	*	?*
/O \S /f	[S [Aux O]] V	*	*

Table 5.9
Comparison of Judgements for "haber" and "estar"

and illustrating the constituent ordering in the second column.

Note that a surface order can appear more than once in Table 5.9. Each occurrence is, however, associated with a different coordination potential and therefore with possibly different grammaticality judgements. For example, the sequence O Aux S V is acceptable with *estar* in Group 1, which admits the coordinated sentence in 5.12.

5.12 Una melodía está Juan cantando y está María tocando
(A melody is John singing and is Mary playing)

But not in Group 3, because the coordinated sentence in 5.13 is not acceptable.

5.13 *Una canción está cantando y una melodía está tocando Juan
(A song is singing and a melody is playing John)

The L_p statements required for these data are given in 5.14 and 5.15 for *haber* and *estar* respectively.

5.14 *haber*:

< /<S, <NPsubject, W>>, W >
&
(/NPsubject ⊃ < /<S, <NPsubject, W>>, NPsubject>)

5.15 *estar*:

< /<S, <NPsubject, W>>, W >

These two L_p statements can be described as follows. Recall that, given the meta-category definition of the auxiliaries, W is instantiated to the non-subject arguments of the verbal argument. In the case under consideration this is the direct object. The L_p constraints therefore allow both *haber* and *estar* to combine with their verbal argument before combining with any non-subject subcategorised elements of that argument. This is the only constraint required in the case of *estar* since it alone suffices to exclude groups 3, 5 and 6 in Table 5.9 where the object precedes the verbal argument on the stack. For *haber* there is the further constraint to the effect that it must combine with its verbal argument before combining with a subject NP from the right. Tables 5.10 and 5.11 give the truth-tables for a selection of the potential ordered local structures. Certain abbreviations have been introduced into the L_p constraint in Table

5.10. The verbal argument is reduced to V and the NPsubject to Subj. Otherwise the constraint is exactly that given in 5.14.

The account given here also allows for fronted direct objects. In fact, all orderings which are permitted for lexical verbs are also allowed when those verbs occur with auxiliaries since the relevant information about ordering constraints is passed from the rightmost verbal argument to the combination of auxiliary and verb.

The analysis also accounts for the behaviour of sequences of auxiliaries given this passing up of information. So that, for example, the subject in a sentence such as that in 5.16a can be moved to any of the positions indicated in 5.16b-e. All of these are predicted by the analysis above including the fact that the subject cannot occur to the immediate right of *haber*, as indicated in 5.16f.

Argument Stack	**Surface Order**	**L_p Constraint**				
		/V	&	(/Subj	⊃	</V, Subj>)
<S,</<S,</NPsubject, W>>, /NPsubject, W> >	[[aux V] S] O	T	**T**	T	T	T
<S,</<S,<\NPsubject, W>>, \NPsubject, W> >	[S [aux V]] O	T	**T**	F	T	T
<S,</NPsubject, /<S, </NPsubject,W>>, W> >	[[aux S] V] O	T	**F**	T	F	F
<S,<\NPsubject, /<S, </NPsubject,W>>, W> >	[[S aux] V] O	T	**T**	F	T	F

Table 5.10
Evaluation of the L_p Statement for "haber"
against some Potential Ordered Local Structures

Argument Stack	Surface Order	L_p Constraint </<S, <...>>, W>
<S, < /<S, </NPsubject, W>>, /NPsubject, W> >	[[aux V] S] O	**T**
<S, < /<S, <\NPsubject, W>>, \NPsubject, W> >	[S [aux V]] O	**T**
<S, < /NPsubject, /<S, </NPsubject,W>>, W> >	[[aux S] V] O	**T**

Table 5.11
Evaluation of the L_p Statement for "estar"
against some Potential Ordered Local Structures

5.16 a **Juan** puede haber estado cantando la canción
(John may have been singing the song)

b Puede **Juan** haber estado cantando la canción

c Puede haber estado **Juan** cantando la canción

d Puede haber estado cantando **Juan** la canción

e Puede haber estado cantando la canción **Juan**

f *Puede haber **Juan** estado cantando la canción

5.5 Subject Control Verbs

The data for subject control verbs in Spanish is interestingly different to auxiliaries. Some of the data are shown in Figure 5.5, again repeated from Chapter 2.

Table 5.12 presents the details of the correlation between grammaticality

(John wants to sing the song)

S $V_{control}$ V O	Juan quiere cantar la canción
$V_{control}$ S V O	quiere Juan cantar la canción
$V_{control}$ V S O	quiere cantar Juan la canción
$V_{control}$ V O S	quiere cantar la canción Juan
O $V_{control}$ V S	una canción quiere cantar Juan
*S $V_{control}$ O V	*Juan quiere una canción cantar
*$V_{control}$ O V S	*quiere una canción cantar Juan
*O $V_{control}$ S V	*una canción quiere Juan cantar

Figure 5.5
Word Order with Subject Control Verbs

judgements and the ordering of items on the argument stack of the control verb exactly in parallel to Table 5.9 for the auxiliary verbs. The L_p statement required to describe these data is given in 5.17.

5.17 < /<S, <NPsubject, W>>, W>
&
(<NPsubject, /<S, <NPsubject>> $\supset$ /NPobject)

The first conjunct excludes the sequences in groups 3, 5 and 6 of Table 5.12; the second conjunct excludes the ungrammatical sequence in group 1 since it asserts that if the subject precedes the verbal argument on the stack, the object must be taken from the right. This is not the case for the ungrammatical sequence in group 1.

Argument Stack	Surface Order	*querer*
Group 1		
/S /f /O	[[Aux S] V] O	ok
/S /f \O	O [[Aux S] V]	*
\S /f /O	[[S Aux] V] O	ok
Group 2		
/f /S /O	[[Aux V] S] O	ok
/f /S \O	O [[Aux V] S]	ok
/f \S /O	[S [Aux V]] O	ok
Group 3		
/S /O /f	[[Aux S] O] V	*
/S \O /f	[O [Aux S]] V	*
\S /O /f	[[S Aux] O] V	*
Group 4		
/f /O /S	[[Aux V] O] S	ok
/f \O /S	[O [Aux V]] S	?
/f /O \S	S [[Aux V] O]	ok
Group 5		
/O /f /S	[[Aux O] V] S	*
\O /f /S	[[O Aux] V] S	??
/O /f \S	S [[Aux O] V]	*
Group 6		
/O /S /f	[[Aux O] S] V	*
\O /S /f	[[O Aux] S] V	*
/O \S /f	[S [Aux O]] V	*

Table 5.12
Grammaticality Judgements for "querer"

5.6 Summary: L_p Statements for Verbal Complexes

This Chapter has shown how a range of complex ordering data can be described effectively within the linear order factored Categorial Grammar presented in Part Two of this thesis.

5.18 below brings together the L_p statements that have been put forward in this Chapter for the analysis of English and Spanish auxiliary verbs and subject control verbs in Spanish. Each has its own ordering and combinatorial potential requiring different L_p constraints. There is, nevertheless, considerable overlap indicated by the first conjunct for the Spanish auxiliaries and the English auxiliary category. In all case, the auxiliary takes a rightward verbal argument *before* taking a direct object or other complement which might instantiate the variable W.

5.18 a *English Auxiliaries*:

</<S, <\NPsubject, W>>, W>

b *Spanish Auxiliaries*:

haber: </<S, <NPsubject, W>>, W>
&
(/NPsubject ⊃
</<S, <NPsubject, W>>, NPsubject>)

estar: </<S, <NPsubject, W>>, W>

c *Spanish Subject Control Verbs*:

querer: < /<S, <NPsubject, W>>, W>
&
(<NPsubject, /<S, <NPsubject, W>>> ⊃
/NPobject)

5.7 Notes

[1] This generalisation has frequently been noted. Groos and Bok-Bennema (1987) argue for the following structure for the Spanish sentence: (XP) V XP*, where the first XP (NP in the examples here) is optional and the verb is "followed by a series of constituents that may appear in any order" (page 68). This has also been suggested in Torrego (1984) in terms of a "free inversion rule" for Spanish. Groos and Bok-Bennema observe that this makes Spanish look very like Hungarian and so-called "topic" languages that have one or two pre-verbal positions which are characterised as topic and focus and a fixed position for the verb. See Abraham and Meij (1986) for a collection of papers discussing the issues of topic, focus and configurationality with particular reference to Hungarian and German. It is the basic claim here that, simple as the L_p constraint offered below is, it represents the facts of ordering in Spanish more comprehensively and rigorously than the schema suggested in Groos and Bok-Bennema.

[2] See Contreras (1978) for a study of Spanish word order in terms of Given/New information.

CHAPTER 6

THE STRUCTURE OF THE CATEGORIAL LEXICON

6.1 Introduction

The structure of the lexicon is frequently not given the prominence it deserves in Linguistic applications of Categorial Grammar. This is rather surprising in the light of the lexicalist orientation of the theory indicated at the beginning of Chapter 1. The standard view of the lexicon which appears in most published accounts of Categorial Grammar analyses is a simple pairing of words with categories. The obvious redundancies in this idea which result, for example, from the fact that one category is frequently associated with more than one word, can be overcome in straightforward ways. But in the theory presented here, where linear ordering information is factored out of the category structures, there are other and more serious forms of redundancy which cannot be eliminated by simply re-grouping words into sets associated with single categories. Two of these redundancies are particularly relevant to considerations of the structure of the categorial lexicon:

(1) Because each category of the grammar is associated with its own L_p statements, any claim to generality for L_p constraints seems to be lost. Since this was one of the original motivations in Generalised Phrase Structure Grammar for introducing the factoring of linear order information from structural constituency information, this appears to be a serious weakness in the present theory of word order;

(2) L_p statements need to be defined recursively over complex categories. This

involves considerable repetition of constraint information from one category to another. Furthermore, additional metatheoretical restrictions on category structure are required in order to define the argument structure of individual categories, an approach adopted earlier in the analysis of auxiliary verbs in English and Spanish.

This Chapter presents an account of the categorial lexicon which avoids these two redundancies. Fundamentally, the problems that need to be addressed in considering possible ways of organising the categorial lexicon are *structure sharing* and *inheritance of information* from one category to another. This chapter presents one model, albeit a relatively simple one, of how structure sharing and inheritance might operate in the categorial lexicon.

6.2 Information Sharing in the Categorial Lexicon

Section 4.4.4 introduced the idea of metacategory definition as a general structural constraint on subsets of well-formed categories of the grammar. This idea proved to be useful in accounting for the behaviour of auxiliary verbs in English and Spanish. However, it is now necessary to look more closely at metacategories in order to see what effect their introduction has on the structure and operation of the grammar.

A metacategory is a way of imposing a dependency between the arguments of a function. In the particular case of the auxiliary verbs examined in the previous chapters, this was a dependency between a function argument and other non-function arguments of a function category. This dependency specification turns out to be a substitute for the recursive specification of L_p constraints within a category. Consider the unordered local structure in 6.1 below.

6.1 $\quad <S, \{ <S, \{NP\}>, NP\}>$

Assume that the grammar requires that the ordered local structures derivable from this lexical entry are in the set given in 6.2.

6.2 { <S, </<S, </NPsubject>>, /NPsubject>>

<S, </<S, <\NPsubject>>, \NPsubject>> }

There are two ways of achieving this constraint on the mapping from unordered to ordered local structures. Firstly, the structure in 6.1 might be given a metacategory definition similar to that given for auxiliaries in Section 4.4.4. This is repeated here for convenience as 6.3.

6.3 *C* = <h, {..., *K*, ...}>

where, *K* is a complex category and
head(*K*) = head(*C*)
argset(*K*) = argset(*C*) - *K*

In which case, the L_p statement in 6.4 induces the set in 6.2.

6.4 </<S, <...>>, NP>

Without a metacategory definition, the L_p statement in 6.5 is required to induce the set in 6.2.

6.5 /<S, <...>>
&
(□/NPsubject V □\NPsubject)

The difference between these two approaches lies in the fact that the first makes a claim about the lexical category, viz. that it belongs to a family of categories which have the same general properties and in which information is shared between the arguments of a function. By making this assumption the L_p statements can be simplified considerably. In the second case, no such assumption is made and the L_p statements must recursively constrain the arguments of the ordered local structures. What is really at stake here is the problem of the inheritance of information between categories, in particular information relating to linear ordering constraints. Categories are not isolated structures associated with lexical items as the simple view of the categorial lexicon implies; rather, they are inter-related structures whose individual properties are intimately bound up with the properties of other categories in the grammar. This is particularly clear in the case of linear order constraints.

Each category in a Categorial Grammar is associated with a set of expressions of the language. These may be single lexical items, as in the case of the category associated with transitive verbs, or all the well-formed sentences of the language as in the case of the category <S,< > >, or, indeed, any substring of the language associated with categories that in a phrase structure context would be referred to as non-lexical or non-terminal. Call the set of such expressions the *expression set* of a category. One way of looking at the inter-relationships that exist between categories is to see them as inter-relationships between elements in the expression sets of categories. So, for instance, when a category takes <S, <NPsubject> > as one of its arguments, it can be thought of as taking any one of the members of the expression set of that category as one of its arguments.

This suggests that categories in a Categorial Grammar might be thought of as being constituted from other categories. This idea can take one of two equivalent forms: either a function category literally *is* built up from other categories or a function category is made up of a number of pointers to other categories in the grammar. Since all complex categories might be thought of in this way, the resulting pattern of pointers to other structures must terminate in some structures which do not point at anything else. These will be primitives of the grammar and it can easily be seen that they will be the basic categories which have empty argument stacks. Taking up the idea that categories are structures which contain pointers to other structures in some well-defined logical space, it can be seen that such a network could be modelled in formal mathematical terms by lattice theory or in computational terms by a programming language like LISP. The point to emphasise is that a strong claim is being made to the effect that function categories really consist only of pointers to other structures in the space of potential categories of the grammar. The rest of this chapter is an explication and development of this simple notion.

As a first step towards describing the categorial lexicon, a distinction can be made between the logical space of category structures and the inheritance of information from one category to another. The recursive definition of category structure for Categorial Grammar induces an infinite set of categories. This is something that has often troubled linguists who are more familiar with the traditional notion that any given language has a finite (even quite small) set of syntactic categories associated with words in the lexicon and a relatively small set of non-terminal category types for deployment as phrasal nodes in Phrase Structure trees. Of course, feature-based grammar formalisms such as Generalised Phrase Structure Grammar have challenged this idea by employing feature sets which when expanded out result in a very large set of categories. The use of partially specified categories makes this innovation more tractable from a grammatical point of view though it in no way reduces the potential number of categories available to the grammar. The important point to emphasise is the fact that even these very large sets of categories remain finite whereas the category set of a Categorial Grammar is infinite. The suggestion put forward here is that the lexicon consists of a set of category pointers defined over the infinite space of potential categories. The set of pointers is finite and, although there is no absolute requirement that every category consists of pointers to other categories, the set of *active* categories for a given language (ie. those categories having expression sets associated with them) *is* finite.

The picture emerging so far is of a structured lattice of categories, the lattice representing information shared between the category structures of the language. This basic idea becomes more interesting and useful when combined with the two-level representation of the lexicon proposed in Chapter 4: a lexicon in which there is a mapping between unordered local structures and ordered local structures via L_p constraints.

Consider the level of unordered local structures. The claim here is that these categories are related to each other by way of pointers to shared structure. The term "shared structure" here is to be taken quite literally: a category that shares structure with another inherits all the linear ordering properties of that structure. When the ordered local structures are projected from the unordered local structures, all shared structure has the linear ordering properties of its origin. Ultimately, all structures resolve themselves into basic categories of the grammar which have no L_p constraints active upon them since they have nothing in their category sets. Figure 6.1 shows a sample lexicon structured in this way.

The two planes of the lexicon are indicated. The lower plane represents the level of unordered local structures. The numbered lines indicate shared category structure. The basic unordered local structures of the grammar such as <NP,{}>, labelled **A** in Figure 6.1, maps into a single ordered local structure since there can be no L_p constraints associated with a category with an empty argument list. The function category **B** takes two NP's and returns an S. It has an L_p statement associated with it and maps into a set of ordered local structures, as indicated. The hypothetical category **C** is more complex. Its first argument is category **B**, as pointer 5 indicates. It therefore inherits the L_p constraints which apply to **B**. The dependencies indicated by pointers 6 and 7 show that the second and third arguments of **C** are the arguments of **B**. They therefore inherit the L_p constraints from that category. The mapping from the unordered local structure **C** to the set of ordered local structures is therefore mediated via a complex interaction of L_p constraints drawn directly from **C** itself and from the constraints it inherits from other categories in the grammar, in this case **B**.

The result of this set of interactions is that the linear ordering properties of the arguments of **C** are such that there is a tight dependency between the ordering of arguments in the first function argument and the ordering of the NP arguments. It would not be possible, for example, to have a category such as that in 6.6 below where the NPsubject has a different directionality marking in the embedded function and elsewhere in the category.

6.6 <S, < <S, <\NPsubject, /NPobject> >, /NPsubject, \NPobject> >

It will be observed that this dependency has achieved the same result that the metacategory definition achieved in previous chapters. The structured lexicon is, therefore, a way of establishing dependencies within and between categories

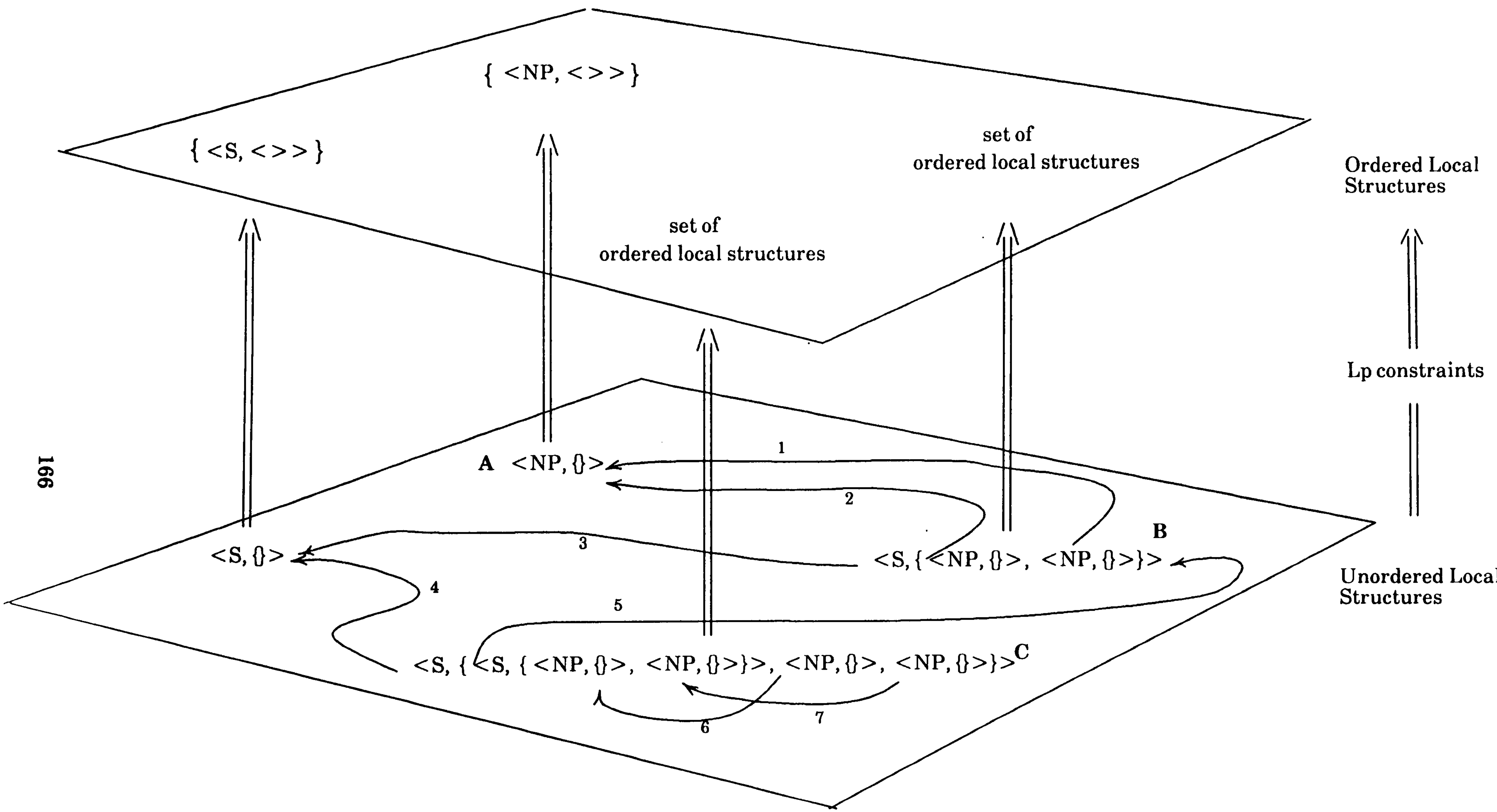


Figure 6.1

Information Inheritance in the Categorial Lexicon

which make "higher level" specification of category relations unnecessary.

There are, of course, certain consequences of adopting this approach to the lexicon. An advantage is that the L_p statements associated with individual unordered local structures in the grammar no longer need be recursive: they apply only to the top level of the arguments in the function's stack. This considerably simplifies presentation of the linear ordering constraints in the grammar. On the other hand, the mapping from unordered local structures to ordered local structures is considerably more complex, since this now has to take into account the fact that function arguments, so to speak, bring with them their own L_p constraints. The overall complexity of the lexicon has not been reduced by this re-interpretation of the relationship between categories, rather the burden has just been moved from the L_p constraints themselves to the mapping from unordered local structures to ordered local structures. The next chapter addresses this issue and suggests a way of allowing the combinatory rules to access the unordered local structures direct thereby eliminating an entire level in the categorial lexicon and consequently avoiding the need for a complex mapping between the two levels.

The second issue mentioned in the Introduction was the apparent failure of a linear order factored Categorial Grammar to accommodate generalisations about word order on a global basis across a language. Such generalisations are, of course, entirely compatible with the idea that a language shows semi-free word order. So, for example, it is quite feasible for the L_p constraint in 6.7 to be true of *every* category in a given language which otherwise shows very complex semi-free word order.

6.7 \NPsubject

The problem lies in how to model obvious generalisations of this kind in the categorial lexicon.

Consider the cases of intransitive and transitive verbs in a language for which 6.7 is true. Figure 6.2 illustrates the relationships that might exist between

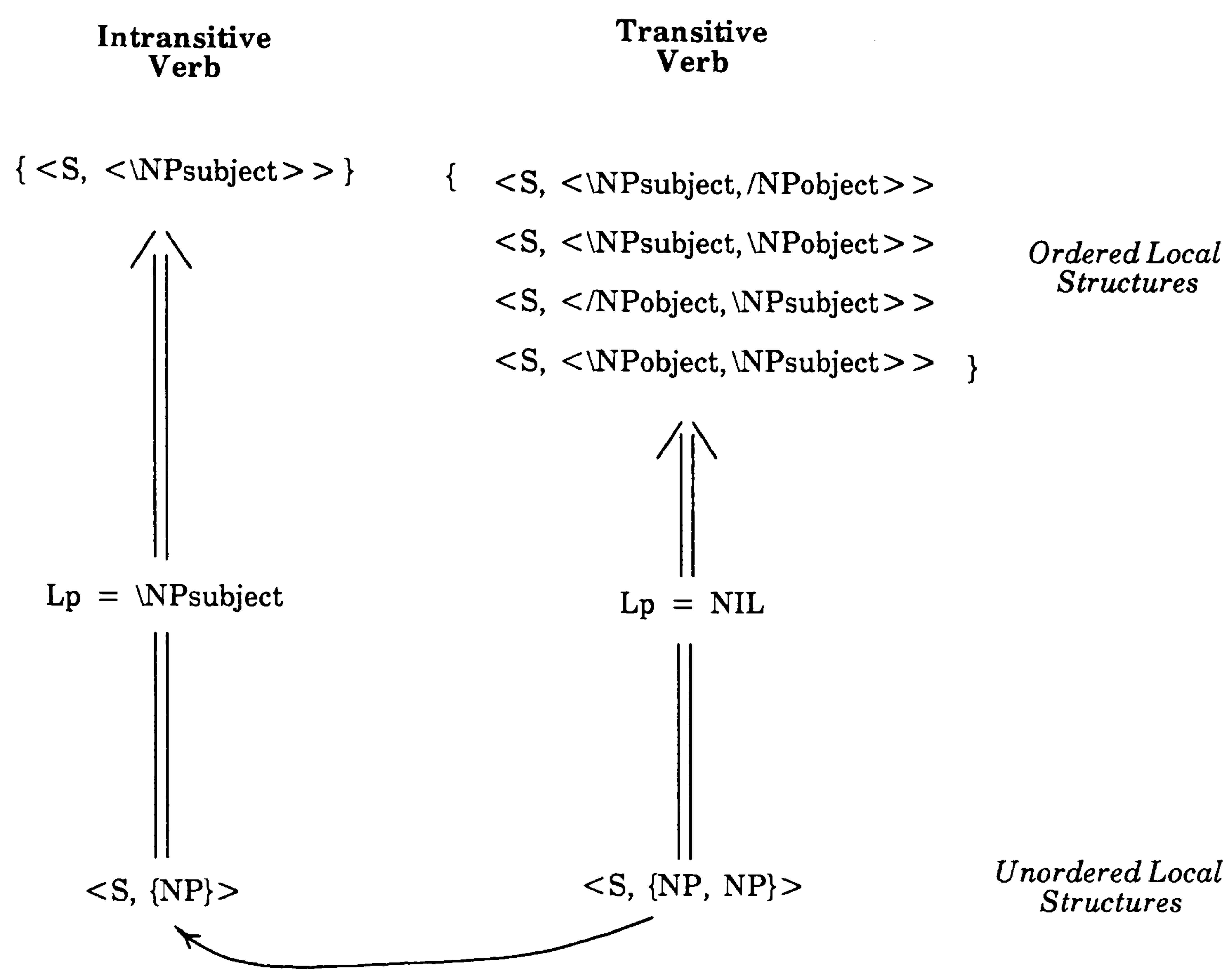


Figure 6.2
Inheritance of L_p Statements in the Categorial Lexicon

these two classes of verbs. The mapping from unordered local structures to ordered local structures is shown, but the structure sharing between the unordered local structures shown and other (basic) categories of the grammar is omitted for simplicity. The L_p constraint in 6.7 is defined as part of the mapping between the two category levels for intransitive verbs. Transitive verbs are interpreted in the standard manner as structures which take two arguments and return a sentence. One of these arguments is a basic category of the grammar, the other is a pointer to the argument of intransitive verbs. This last therefore inherits the linear ordering constraints of the intransitive verb's argument. In fact, given the ordered local structures which require to be associated in this language with transitive verbs, *no* L_p constraint operates in the mapping from unordered local structure to ordered local structures for this verb class.

This scheme makes the correct predictions about the subject argument of intransitive and transitive verbs: they are the same category in the grammar. Furthermore, redundancy in the statement of L_p constraints is eliminated through the use of structure sharing.

PART THREE

EXTENDED CATEGORIAL GRAMMAR

CHAPTER 7

CATEGORY STRUCTURE AND COMBINATORS

7.1 Introduction

The Second Part of this thesis presented a theory of word order in Categorial Grammar based on a generalised notion of local structure. The aim was to describe a linear order factored grammar which remained close in spirit and fairly close in form to Generalised Categorial Grammar. It was shown how this approach could be used for the analysis of a range of semi-free word order data in Spanish. The model presented in Part Two does, however, have certain limitations. Perhaps the most serious is associated with the mapping from unordered local structures to ordered local structures. The problem here is analogous to that faced by proposals to generate fully instantiated Phrase Structure rules from Immediate Dominance rules in early Generalised Phrase Structure Grammar: by spelling out the features and linear orderings, thousands of rules can be generated from each immediate dominance rule. The problem in the case of the linear order factored Categorial Grammar described in Part Two is not as serious as that, but the sets of ordered local structures associated with a single unordered local structure may be quite large.

Further, as was pointed out in Chapter 6, the mapping from unordered to ordered local structures becomes complex when categories are regarded as inheriting linear ordering information from other categories in the grammar. The only alternative is to structure the lexicon as a simple list and to employ complex, recursive L_p constraints, thereby losing generality and unacceptably increasing the redundancy in the lexicon. The obvious way to avoid these problems is to do away with the level of ordered local structure altogether. This would eliminate the need to impose order on the local structures and thereby do away with the two level structure in the lexicon while at the same time retaining the property of information sharing since this operates at the level of unordered local structures. One consequence would be that L_p statements would no longer be interpreted as filters on an expansion of categories but rather as descriptions of the linear ordering properties of categories. Furthermore, the combinators would have to be re-structured in order to allow them to combine with the equivalent of unordered local structures.

This Chapter addresses these issues in the following way. First of all, in Section 7.2, a formalism for the representation of category structure in linear order factored Categorial Grammars is presented: attribute-value matrices. The use of attribute-value matrices for the representation of syntactic information is, of course, not new. Theories such as Functional Unification Grammar (Kay (1983, 1985)), Lexical-Function Grammar (Bresnan (1982), Kaplan and Bresnan (1982), Halvorsen (1983)) and Head-Driven Phrase Structure Grammar (Pollard (1985), Pollard and Sag (1987)) are among recent theories to use this formalism extensively or exclusively for modelling syntactic information. Furthermore, the use of attribute-value matrices in Categorial Grammar is not new. Both Categorial Unification Grammar (Uszkoreit (1986)) and Unification Categorial Grammar (Zeevat, Klein and Calder (1987), Calder, Klein and Zeevat (1988)) in various versions have made use of this formalism. The approach adopted here was developed independently of either of these recent theories, however, and differs from them in a number of important respects. Section 7.2 does more than simply present a new way of representing categories. It addresses a number of issues such as information sharing through re-entrancy, the use of functional operations over values within matrices and the typing of categories. Through these devices, a number of difficulties with the formalism in the previous chapters are overcome. The new representation of category structure put forward in this section leads to a version of Categorial Grammar which is sufficiently distinct from other current versions that it is dubbed *Extended Categorial Grammar*.

In Section 7.3, the problem of working with a single level of unordered local structures is addressed. The central issue is how to ensure that the combinators can operate on two unordered structures by taking one argument of a function at a time when the linear ordering information is represented globally for all arguments of the function. To solve this problem a truth-functional operation - *T-Eval* - and a three-valued logic are introduced. In Section 7.4 the combinators are discussed in detail and the suggestion is made that a number of the combinators introduced in Chapter 1 and retained throughout the first two parts of the thesis can in fact be merged into a single combinatory operation. The Chapter concludes in section 7.5 with a discussion of the place of Extended Categorial Grammar in the hierarchy of Categorial Grammars discussed in Chapter 1.

7.2 Extended Categorial Grammar: Towards a New Category Structure

As mentioned in the Introduction, this section describes each of the components of a theory of category structure for Extended Categorial Grammar. These include the following:

1 the use of attribute-value matrices to model all aspects of category structure;

2 the generalisation of attribute values to specified data types;

3 the use of re-entrancy to model information sharing and inheritance both within categories and in the lexicon;

4 category typing;

5 the introduction of operators over attribute values;

Each of these aspects of the formalism is discussed with examples.

7.2.1 Attributes, Values, Re-entrancy and Information Sharing

Attribute-value matrices were introduced in Section 4.3. At that point they were applied only to the primitive semantic types of the grammar which occur as the heads of local structures. Atomic symbols were used for both attributes and their values with the result that the category for plural subject NP, for example, had as its head the attribute-value matrix given in 7.1.

7.1 $$\begin{bmatrix} \text{N} & + \\ \text{V} & - \\ \text{Phrasal} & + \\ \text{Num} & \text{plural} \\ \text{GR} & \text{subject} \end{bmatrix}$$

In this section the attribute-value notation is generalised to model all aspects of category structure, thereby unifying the treatment of categories in Extended Categorial Grammar.

One standard generalisation of attribute-value matrices involves attributes taking attribute-value matrices as values. In other words, matrices are defined recursively. Figure 7.1 (from Shieber 1986, p.13) illustrates an attribute (Agreement) taking an attribute-value matrix for its value. Extended

$$\begin{bmatrix} \text{Category:} & \text{NP} \\ \text{Agreement:} & \begin{bmatrix} \text{Number:} & \text{singular} \\ \text{Person:} & \text{third} \end{bmatrix} \end{bmatrix}$$

Figure 7.1
A Recursively Defined Category in Attribute-Value Matrix Notation

Categorial Grammar takes this generalisation of matrix structure one step further by permitting attributes to take named data types as values. The data types available are: atoms, attribute-value matrices, sets and boolean expressions. 7.2a illustrates the notation used for set-valued attributes and 7.2b, the notation for boolean-valued attributes where F is the attribute name and x, y are arbitrary categories of the grammar.

7.2 a [F: {x, y}]

b [F: $((x \supset y) \vee z)$]

The last two will be discussed later in relation to the function categories of Extended Categorial Grammar where it is shown that boolean expressions can in fact be reduced to conventional attribute-value matrices and a reformulation of category matching over set-valued attributes as sets of matchings over non-set-valued attributes is presented.

A property of attribute-value matrices which plays an important role in Extended Categorial Grammar is *re-entrancy*. This is the term used when two attributes share the *same* value and is the formal mechanism available in attribute-value matrix notation for handling information sharing *within* category structures. It is important to distinguish the situation where "same" in the previous sentence means the very same token, as opposed to two identical but distinct tokens of the same type: re-entrancy refers to the former, stronger relation. In Figure 7.2, for example, (adapted from Shieber 1986, p.13) the two categories are not the same: the top one contains two attributes (F and G) having distinct values; the bottom one has the attributes sharing the very same value. This is a notion of structure sharing that will be familiar to anyone who has worked with symbolic languages such as LISP. The shared value is indicated with coindexed integers in boxes. Other notations use lines linking substructures or equality signs. The first notation will be used throughout this and the following chapter.

$$\begin{bmatrix} \text{F:} & \begin{bmatrix} \text{H:} & \text{a} \end{bmatrix} \\ \text{G:} & \begin{bmatrix} \text{H:} & \text{a} \end{bmatrix} \end{bmatrix}$$

$$\begin{bmatrix} \text{F:} & \begin{bmatrix} \text{H:} & \text{a} \end{bmatrix}^{\boxed{1}} \\ \text{G:} & \boxed{1} \end{bmatrix}$$

Figure 7.2
Re-entrancy in Attribute-Value Matrices

7.2.2 Category Typing

An Extended Categorial Grammar defines an association between each attribute given by the grammar, the data type of its value and the range of values it can take. So, for example, an attribute GR (Grammatical Relation) is atomic valued taking "subject" as a possible value, but not, say, "plural". However, the notation introduced so far places no restrictions on the attributes themselves with respect to their use in categories. Placing restrictions on the attributes that can occur in categories is referred to as *category typing* and the definition of well-formedness for categories in Extended Categorial Grammar makes reference to the fact that categories are typed with respect to the attributes they can take. Where required for clarity, categories are labelled with their type, indicated on the bottom right of category brackets.

To illustrate the role of category typing in Extended Categorial Grammar three important types of category are introduced here: *primitive*, *basic* and *function*. These correspond, respectively, to the semantic primitives, basic and functor categories used in previous chapters. The semantic primitives of the grammar are syntactically complex expressions formed from syntactic primitives. Attributes such as "N" and "V" are appropriate to primitive categories but not to basic or function categories. The category in 7.1 is an example of a primitive category; it could therefore be labelled as such at the bottom right.

Figure 7.3 shows schematic representations of function and basic categories of

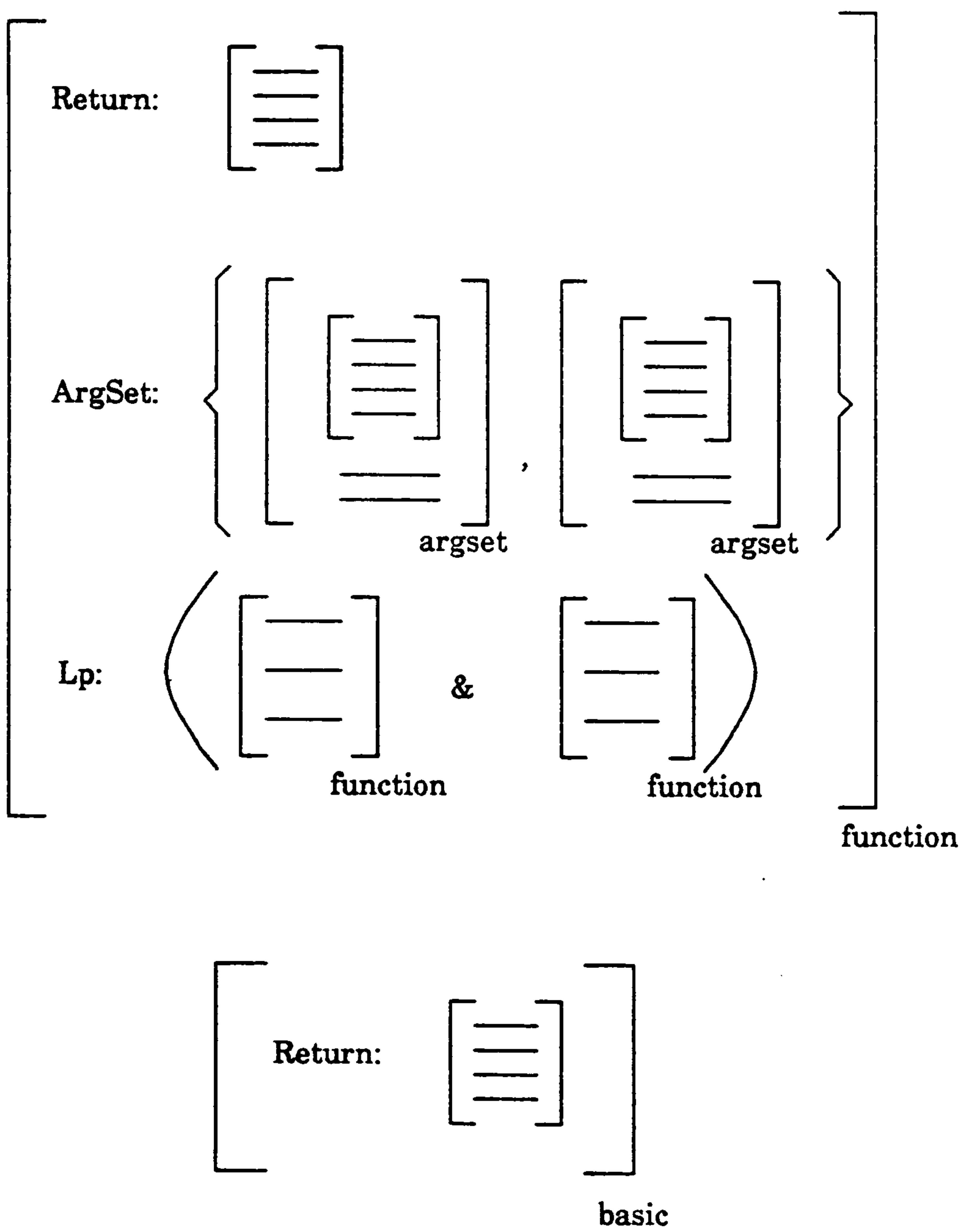


Figure 7.3
Schematic Representations of Function and Basic Categories in Extended Categorial Grammar

Extended Categorial Grammar. Both take, among other possibilities, the attributes Return, ArgSet and L_p. The Return attribute of a function takes either a basic or function category as its value; the Return attribute of a basic category takes a primitive category as its value. The value of ArgSet in a function is a set of categories each of type *argset*; in a basic category it is the empty set (as indicated). The L_p attribute of a function is more complex. Briefly, it takes a truth value (ternary rather than binary, as discussed in Section 7.3) but this truth value is normally mediated by a boolean expression (such as that shown in Figure 7.3) each basic expression of which is of type *function* and which is evaluated in a way explained in Section 7.3 to return a truth value.

7.2.3 Operations over Attribute Values

One further extension to the attribute-value matrix notation which is crucial for understanding the combinatorial process of Extended Categorial Grammar and which will play a prominent role in the description of natural language data in the next chapter is the introduction of operations over attribute values. Consider the schematic category in 7.3 below.

7.3

$$\begin{bmatrix} \text{F:} & \rho\left(\begin{bmatrix} \text{G:} & \text{x} \\ \text{H:} & \text{y} \end{bmatrix}\right) \\ & \\ \text{K:} & \text{z} \end{bmatrix}$$

ρ is an operation defined over the value of F, in this case a category. It is unconstrained except for the fact it must return a data type appropriate to the attribute and is typically the locus of language-specific category changing operations. An example of a category using two operations is given in Figure 7.4. Firstly, there is an operation - OP - over the value of the Return attribute; secondly, there is the operation *T-Eval* over the value of L_p. OP is taken here to illustrate the way that Return operators work in general; *T-Eval*, which is

$$\begin{bmatrix} \text{Return:} & \text{OP}(\ \boxed{3}\ ) \\ \text{ArgSet:} & \left\langle \begin{bmatrix} \text{Return:} & \text{S} \\ \text{ArgSet:} & \{\ \text{NPsubject}\ \boxed{1},\ \ \text{NPobject}\ \boxed{2}\ \} \\ \text{Lp:} & /\ \boxed{1}\ \ \&\ \ /\ \boxed{2} \end{bmatrix}^{\boxed{3}} \right\rangle \\ \text{Lp:} & \text{T-Eval}\left(/\ \boxed{3}\right) \end{bmatrix}$$

Figure 7.4
An Example of a Category with an Operation over its Return Value

an obligatory operator in all function categories, is discussed in detail in Section 7.3. If the operation OP is as given in 7.4 below where the arrow in the representation of the operation indicates a change in the value of the directionality marker in the L_p statement for the NPobject argument, the category in Figure 7.4 will be interpreted as a function that takes a function of two NP arguments to its right and returns a similar function as value except that the NPobject argument is now looked for from the left instead of the right.

7.4 $\quad L_p$: /NPobject $\rightarrow$ \NPobject

In Chapter 8 it is shown that the individual characteristics of certain clitic personal pronouns in Spanish can be localised in the different Return operations each clitic possesses.

7.2.4 Example Extended Categorial Grammar Categories

Having presented the individual components of Extended Categorial Grammar category structure, it will be useful at this point to bring them together and give some examples of Extended Categorial Grammar categories. Figure 7.5 is the category for a transitive verb in English taking two NP arguments and returning a sentence. Unlike the representations used so far, this spells out in detail the full structure of the category. It will be too cumbersome to continue to represent categories in this way, therefore various abbreviatory conventions will be periodically introduced. Figure 7.4 illustrated some standard abbreviations. Later sections will use more highly abbreviated representations.

Most categories in Figure 7.5 are labelled with their type, and typical attributes associated with the category types are shown. For example, *argset* categories take three attributes: *Category*, *OPT* and *DIR*, the first two of which are shown in Figure 7.5. The *Category* attribute takes either a function or a basic category (as here) and *OPT* is a binary attribute over the values NIL and T. T means that the argument is optional and works in a way explained in the next chapter to ensure that a function can be saturated even if it has not accepted an argument of that particular type. NIL, on the other hand, means that the function must accept an argument of this type if it is to be fully saturated. *DIR* takes a left or right directional slash as value. Partial category specification plays an important role with respect to this attribute. The argset categories of the function in Figure 7.5 have no specifications for this attribute. This is interpreted as meaning that the function is unsaturated with respect to its arguments. The results of combining two categories will be discussed in more detail in later sections. For the present it should be noted that in Extended Categorial Grammar arguments are not eliminated from function categories in the way that is commonly assumed in Categorial Grammars. The function in Figure 7.5 would at no time during the parsing of a sentence in which it is employed lose any of its arguments in the sense that argset categories are removed from the set value of ArgSet. Rather, the *DIR* attributes in the argset categories are used for two purposes: they mark the fact that an argument of the function has been accepted and they mark the

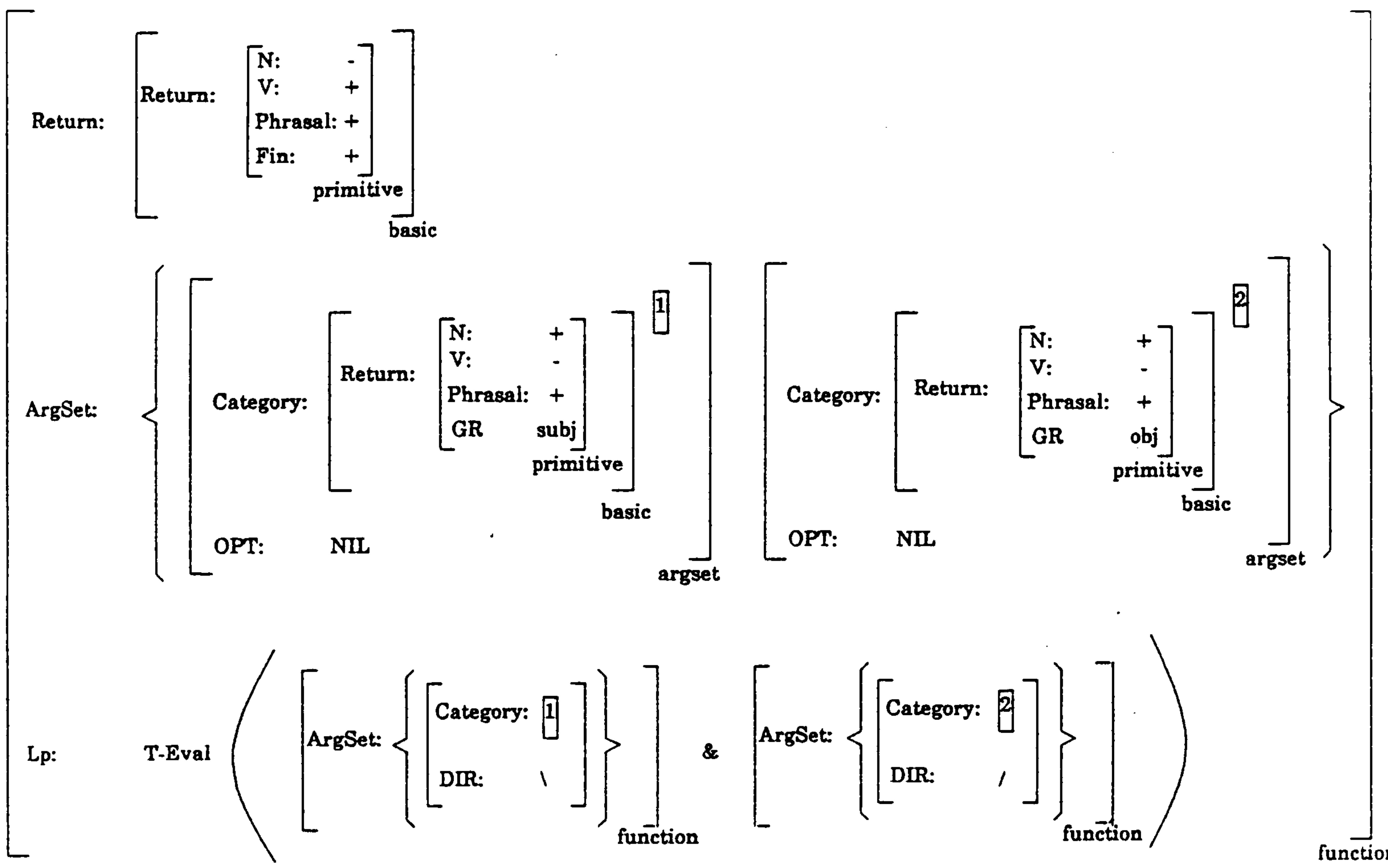


Figure 7.5
An Illustrative Extended Categorial Grammar Category

direction from which the argument came. This innovation greatly simplifies the category matching operation since without it the L_p statements would also need to be changed and this would be difficult to model.

The L_p attribute takes a boolean expression with *T-Eval* as an operator, as already mentioned. Notice that the basic expressions of the boolean are partially specified function categories.

Attention should be drawn to the use of re-entrancy in Figure 7.5. Both the arguments in Figure 7.5 share structure with basic L_p expressions. Re-entrant structure is frequently used in this way in Extended Categorial Grammar but it is *not* a requirement of the theory that a basic expression of L_p must be coindexed with an argument in ArgSet. Consider the category in Figure 7.6. This is similar to the category in Figure 7.5 except that the L_p constraint is not re-entrant with any argset category. The interpretation of the L_p expression in Figure 7.6 is potentially ambiguous. It can have either a universal or existential interpretation. On the former reading, it means that every category with which it matches must have the appropriate directionality marking; on the latter reading, it means that the basic expression must be true of at least one

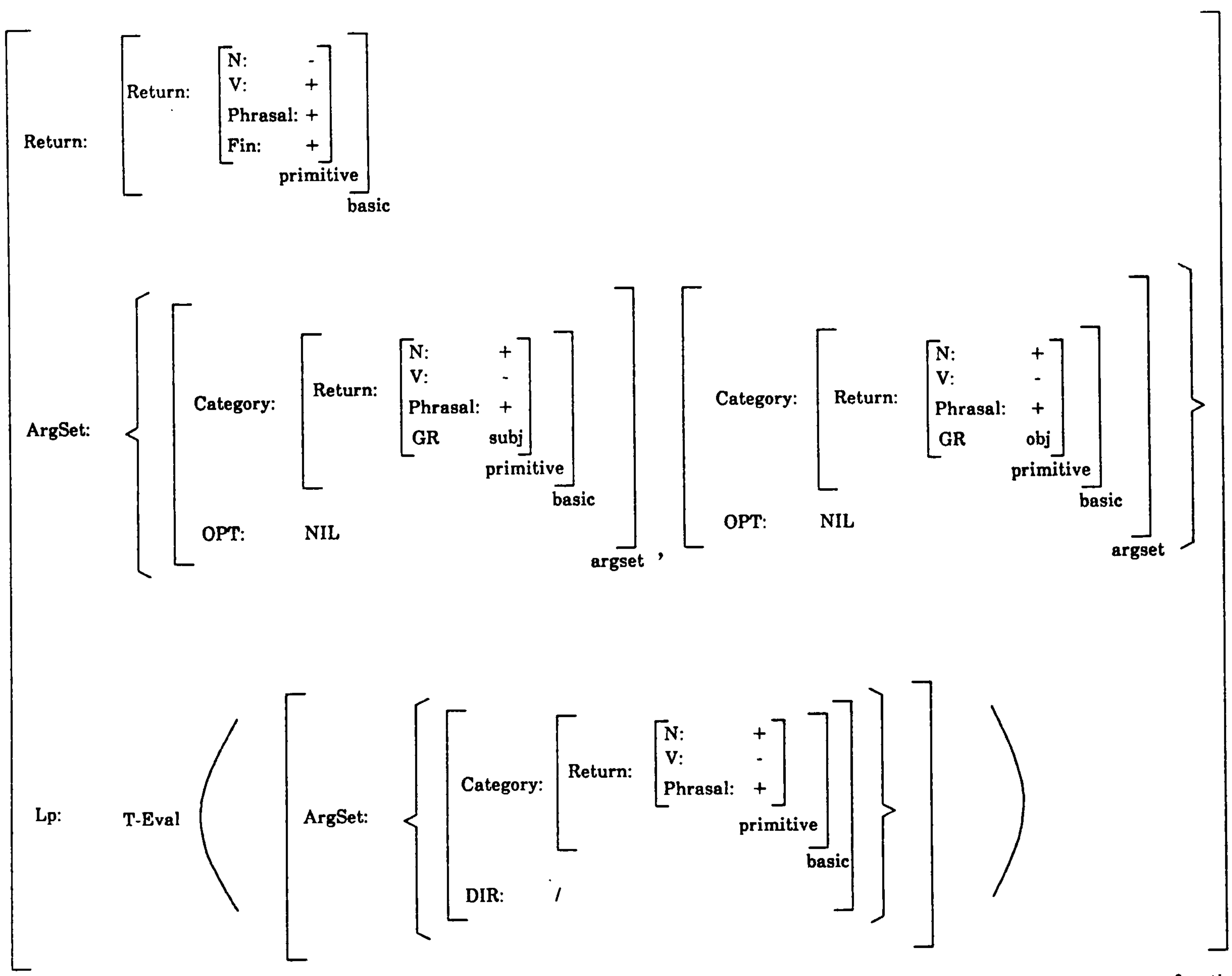


Figure 7.6
An Extended Categorial Grammar Category without Re-entrancy

category in the function's ArgSet. As explained later, the second reading is adopted in this chapter.

As a last example, Figure 7.7 shows a category which takes a function as argument. In conventional Categorial Grammar this is the function S/(S\NP), ie. one of the categories for type-raised noun phrases.

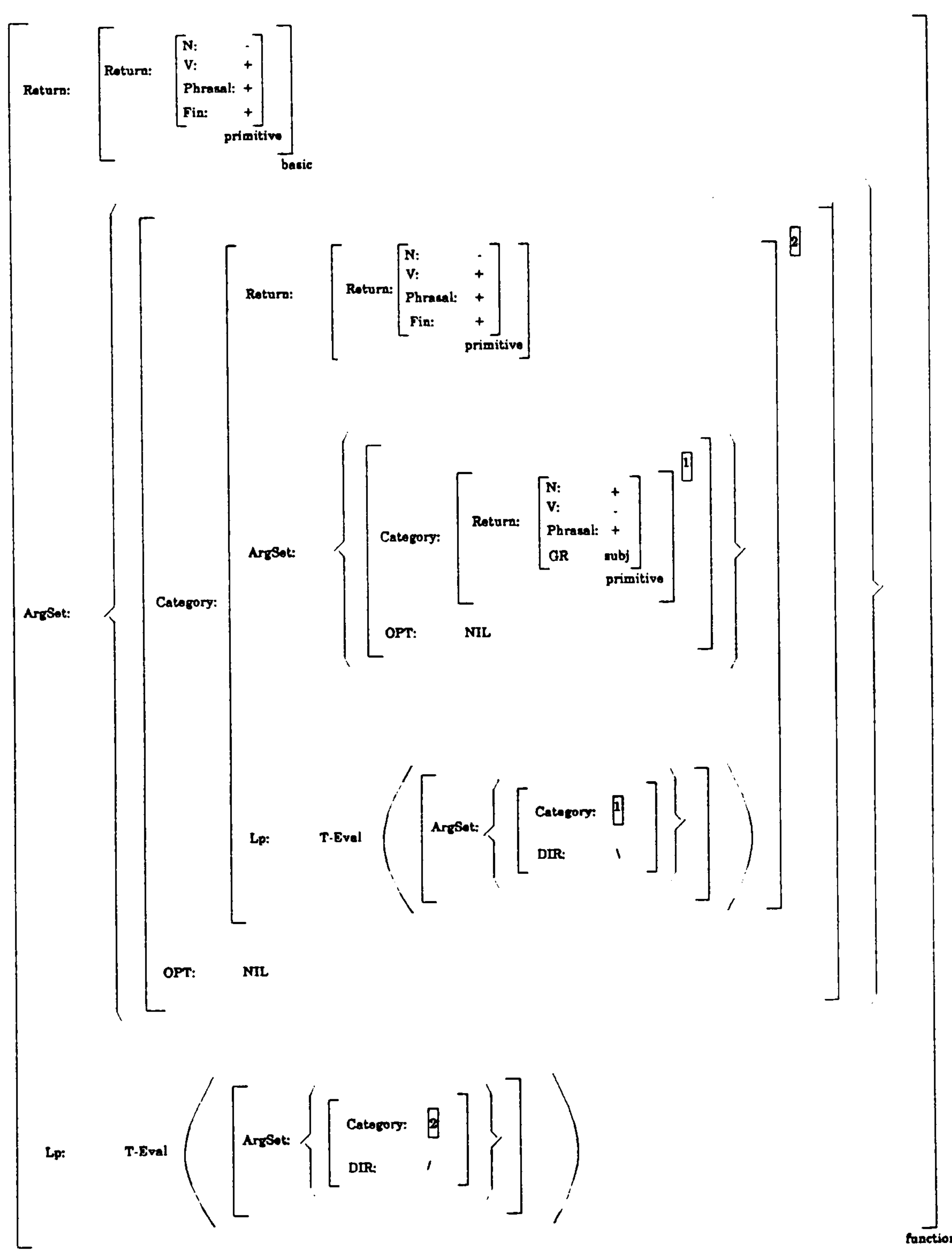


Figure 7.7
A Category Taking a Function as Argument

To conclude this section, a comment can be made about the structure of the categorial lexicon in Extended Categorial Grammar. In Chapter 6, a view of the structure of the categorial lexicon was put forward in which categories were seen as inheriting information, particularly linear ordering information, from other categories in the lexicon. The concept of typing introduced in this section may be used to model this aspect of the lexicon. The category space can be represented as a single attribute-value matrix (of type "lexicon") which takes category* for its attributes (the kleene star notation is used to indicate that "lexicon" can take any number of attributes of type "category"). Within the lexicon, structure sharing can operate between categories of the grammar in the way indicated in Section 7.2.1, a fact which allows for simplification of the L_p constraints. For ease of exposition, however, L_p statements will continue to be represented as one element in categories.

7.3 T-Eval and the Interpretation of L_p Constraints

T-Eval is an operator over the boolean expressions which are the values of the L_p attribute. It evaluates L_p expressions and returns a truth value. One of the conditions for successful application of the combinators of Extended Categorial Grammar is that *T-Eval* of the function returns a non-false value; it therefore plays a crucial role in the formal grammar model presented here. Two questions need to be considered with regard to *T-Eval*: (1) what sort of truth value does it return, and (2) how are the basic L_p expressions evaluated? Each of these is considered in this section. It is shown in 7.3.1 that a binary truth value interpretation is not adequate for modelling the contingency of partial information displayed by Extended Categorial Grammar function categories. 7.3.2 goes on to argue that a three-valued logic does provide the means to model this partial information and examples of the application of the three-valued logic are given. The second question with respect to *T-Eval* is addressed in 7.3.3 which provides a detailed account of how basic L_p expressions are evaluated and presents the formal algorithms.

7.3.1 The Inadequacy of a Two-Valued Interpretation of L_p Statements

Boolean values of the L_p attributes are evaluated by *T-Eval* to return a truth value. It might be thought that a two-valued interpretation of the boolean expression would be adequate, but this turns out not to be the case. This subsection examines the reasons for this.

To assist with the presentation of the argument, a shorthand notation for function categories is introduced, an example of which is given in 7.5 below.

7.5 $\{ X^1, \; Y^2 \}$
$\backslash 1 \;\; \& \;\; /2$

This is function category missing its Return attribute and value (since these are not relevant to the interpretation of *T-Eval*) and the attributes ArgSet and L_p. The values of these last two attributes are, however, represented (although only in abbreviated form) and re-entrant values are marked by coindexing. The category in 7.5 is a function looking for an X to its left and a Y to its right.

Combination of a function and argument takes place in two stages. Firstly, when a potential argument is offered to a function it is required to match with one of the categories in the function's ArgSet. If there is a successful match, it is assumed that the DIR feature of the relevant argset category in ArgSet is set appropriately and remains set if the second stage of the combination goes through, that is, if *T-Eval* returns a non-false value for the L_p expression of the function. So, if the category in 7.5 is offered an X from the left, the result will be the category to the right in 7.6. Note that neither at this stage nor later is the X argument removed from the function; rather, it is given the appropriate directionality marking and is assumed to be "neutralised" as far as further combination is concerned.

7.6 X $\{\backslash X^1, \; Y^2\}$
$\backslash 1 \;\; \& \;\; /2$

Now consider the evaluation by *T-Eval* of the boolean expression in 7.6, given a binary truth value. It seems clear that the first conjunct of the L_p statement should evaluate to T since the DIR values are the same. Equally, it seems obvious that if the DIR values had been contradictory, it should have evaluated to F. What about the cases where the DIR attribute in the relevant argset category is absent, as with the second conjunct in 7.6? Should this cause the basic L_p expression to evaluate to T or F? The obvious answer seems to be T. This would then model the intuition that a basic L_p expression may be true, and certainly should not be false, of an argument which the function has not yet encountered. In the case under consideration, this would indeed produce the desired result, as indicated in 7.7, where the truth values are indicated beneath the L_p expression for the category.

$$
\begin{array}{lll}
7.7 & X & \{\backslash X^1, \; Y^2\} \\
 & & \backslash 1 \;\&\; /2 \\
 & & \\
 & & T \;\&\; T \\
 & & \quad T
\end{array}
$$

Both conjuncts in 7.7 evaluate to T and therefore the argument X to the left would be combinatorially acceptable. Not only that, but the category abbreviated in 7.7 in which the X argument of ArgSet retains its leftward looking DIR value can itself accept a Y from the right following the same algorithm for evaluation of the boolean expression. Notice, that another X category cannot be accepted given the assumption mentioned above that an argument may not unify with a category in ArgSet which has a DIR value.

Now consider the function in 7.8.

$$
\begin{array}{ll}
7.8 & \{X^1, \; Y^2\} \\
 & \backslash 1 \;\&\; \neg /2
\end{array}
$$

The proposed binary version of *T-Eval* will *not* return the correct result for this category. Assume that 7.8 is presented with an X from the left, as in 7.9.

7.9 X $\{\backslash X^1, \quad Y^2\}$
$\backslash 1 \;\&\; \neg /2$

$$\begin{array}{c} T \;\&\; \neg T \\ T \;\&\; F \\ F \end{array}$$

The first conjunct will return T, as required, but the second will return F given the earlier assumption that basic expressions of L_p return T for categories in ArgSet which have no specific setting for their DIR attribute. The result is that the function in 7.8 will not combine with an X to the left, which is wrong. Negation, therefore, causes problems for the evaluation of L_p expressions. It is not only negation, however, that causes problems. Implication also fails to produce the correct results. Consider the category in 7.10.

7.10 $\{X^1, \quad Y^2\}$
$\backslash 1 \supset /2$

If 7.10 is offered a Y from the left, as in 7.11, *T-Eval* will evaluate the L_p expression to F.

7.11 Y $\{X^1, \quad \backslash Y^2\}$
$\backslash 1 \supset /2$

$$\begin{array}{c} T \supset F \\ F \end{array}$$

This is wrong, since the category in 7.10 can accept a Y argument from the left, although only on the condition that the X argument it receives comes from the right. This is demonstrated in Table 7.1 below where the relevant sequence is highlighted.

What is the nature of the problem here? The L_p expression of the function encodes *all* information about the the relative orderings of its arguments. In Extended Categorial Grammar, however, as in conventional categorial grammar, arguments are presented to functions one at a time. There must therefore be a way of encoding partial information in functions with regard to the

linear order		\X	⊃	/Y
1	X f Y	T	T	T
2	X Y f	T	F	F
3	**Y f X**	**F**	**T**	**F**
4	Y X f	T	F	F
5	f X Y	F	T	T
6	f Y X	F	T	T

Table 7.1

validity of the combination. When an argument is presented to a function there are three possible outcomes: the combination may be successful, it may fail or it may be undecidable without further information concerning future arguments the function might receive. This suggests the usefulness of a three-valued logic to model the contingency of partial information in which the intermediate truth value represents the undecidability of the truth conditional status of the function with respect to its putative argument. This forms the topic of the next subsection.

7.3.2 Three-valued Logic in Extended Categorial Grammar

Three valued logics have a long history and have been used for modelling uncertainty in a number of different domains. One of the earliest systems (Łukasiewicz (1920)) was introduced to model future contingency in which the third, intermediate truth value was intended to be read as "indeterminate" or "possible". A more recent system due to Kleene (1952) was introduced to model "undecidability" in which the intermediate truth value was intended to be taken by statements which are neither true nor false. It is the kleene matrices that are relevant for modelling the undecidability of categorial combination in Extended Categorial Grammar. These matrices are presented in Figure 7.8. They differ from the original Łukasiewicz tables only with respect to material implication. Whereas Łukasiewicz set the result of an implication to T when both the antecedent and consequent are indeterminate, Kleene set that result to indeterminate. In general, "Kleene's matrices are thus constructed on the principle that where the truth or falsity of one component is sufficient to decide the truth or falsity of a compound, the compound should take that value despite having (an)other, undecidable component(s); otherwise, the compound is itself undecidable" (Haack 1978 p. 207).

In order to understand the operation of these truth tables in the present context, consider again the schematic Extended Categorial Grammar function category given in 7.8 above. It is assumed, as seems reasonable, that a basic expression of L_p will return i if it unifies with an argset category which has no DIR attribute. Consequently, if 7.8 is presented with an X from its left (as in 7.9), it will return the intermediate value i as indicated in 7.12 below.

$$
\begin{array}{llccc}
7.12 & X & \{\backslash X^1, & & Y^2\} \\
 & & \backslash 1 & \& & \neg /2 \\
 & & & & \\
 & & T & \& & \neg i \\
 & & T & \& & i \\
 & & & i &
\end{array}
$$

This is the desired result since it means that 7.8 can accept an X from the left but only contingently. If a Y argument later comes from the right, this category cannot accept it and further combination is blocked.

The introduction of three-valued logic also overcomes the problem with conditional L_p expressions, as 7.13 shows.

¬	
T	F
i	i
F	T

&	T	i	F
T	T	i	F
i	i	i	F
F	F	F	F

V	T	i	F
T	T	T	T
i	T	i	i
F	T	i	F

⊃	T	i	F
T	T	i	F
i	T	i	i
F	T	T	T

Figure 7.8
Truth Tables for a Three-Valued Logic

$$\begin{array}{lll} \text{7.13} & Y & \{X^1, \quad \backslash Y^2\} \\ & & \backslash 1 \ \supset \ /2 \\ & & \\ & & \mathrm{i} \ \supset \ F \\ & & \quad \mathrm{i} \end{array}$$

The combination of these two categories is, therefore, undecidable as indicated earlier until the X argument is received. Again, 7.14 shows the case of an X argument to the left.

$$\begin{array}{lll} \text{7.14} & X & \{\backslash X^1, \quad Y^2\} \\ & & \backslash 1 \ \supset \ /2 \\ & & \\ & & T \ \supset \ \mathrm{i} \\ & & \quad \mathrm{i} \end{array}$$

Orders 1 and 4 in Table 7.1 show that the validity of this combination is not decidable until further information about the direction of the second argument becomes available.

One side-effect of using a three-valued logic in this way is that further information becomes available (especially at parse time) about the status of a potential argument of a function. There are, in fact, four outcomes of an attempted combination. Firstly, *T-Eval* might return F, in which case the combination does not go through. Secondly, *T-Eval* might return T with the function saturated (ie. all the DIR attributes of the argset categories are set to values). Thirdly, *T-Eval* might return T with the function being unsaturated. This is an interesting case; it means that the argument is acceptable to the function no matter what other arguments may be accepted at a later stage. Such situations arise in cases where a function always looks in a certain direction for an argument of a particular type no matter what other arguments it might accept. Fourthly, *T-Eval* might return i, which means that the combination of function and argument is undecidable given current knowledge.

7.3.3 The Evaluation of Basic L_p Expressions

The previous two sections motivated the introduction of a three-valued logic for the interpretation of L_p expressions by *T-Eval* and showed how this works to make the right predictions about the combinatorial properties of Extended Categorial Grammar categories. There remains the task of showing in detail how basic expressions of L_p receive their truth values. This is quite complex but the leading idea is as follows: a basic L_p expression receives its truth value via an algorithm which assumes an attempted match of the basic expression against the function category of which it is a part. Depending on the success or failure of this attempted match certain information is made available to the algorithm to determine the truth value associated with the basic expression.

Recall that the L_p logic of Part Two defined three basic expression types: a category (with or without a directionality marking), and expressions of the form

$<\alpha,\beta>$ and $f{:}\alpha$, where α, β are categories of the grammar. In the terms of Part Two, the first type of basic expression asserted the existence of a category (often with a directionality marking) in the stack of arguments of the ordered n-tuple. The second type of expression asserted an order on the stack, viz. α is to the left of β. The third type asserted the presence of some feature elsewhere in the n-tuple. These expressions need to be re-interpreted in Extended Categorial Grammar given the fact that function categories take a set rather than a stack of arguments. The first and third expression types both assert the presence of certain attribute-value structure in the function, they can therefore be amalgamated. This satisfies the claim made earlier that the use of attribute-value matrices permits significant generalisation not available in the model developed in Part Two. There remains, however, the expression of form $<\alpha,\beta>$. Clearly, the interpretation of this expression needs to be quite different in Extended Categorial Grammar compared with its interpretation in the model in Part Two. Instead of asserting an order on a stack of arguments, in Extended Categorial Grammar it is interpreted to mean that the function must combine with an α category *before* combining with a β category. It will be clear, however, that this new interpretation is simply a reflex of the fact that it evaluates against a set rather than a stack, the underlying effect is the same.

The outcome of these changes is that Extended Categorial Grammar is left with *two* basic expression types in its L_p constraints: one is a partially specified function category (usually with a directionality marking), the other is an ordered pair of partially specified function categories (usually without directionality markings). The fundamental operation used by the algorithms for assigning truth values to basic expressions is a category matching operation of the basic expression against the function category in which it is contained. The details of this are given in the following paragraphs.

Category matching and unification are closely related operations. The only difference between them is that with category matching no new category structure is created by the operation: it is simply a matter of checking one category against another. Certain information is made available following an attempt to match two categories, which is used by *T-Eval*, as explained below. Category matching over simple and recursive attribute-value matrices is exactly like the unification operation introduced in 4.3, ie. two categories match if they do not contain contradictory information. However, the generalisation of attribute values to named data types requires a fuller account than that given in Chapter 4. Recall that the four data types for values are atoms, attribute-value matrices, sets and boolean expressions. Matching over boolean expressions reduces to matching over atoms and attribute-value matrices given the fact that arbitrary boolean expressions can be structured as conventional attribute-value matrices, as indicated in Figure 7.9. The problem case is matching over set-valued attributes. The proposal made here is not intended as a general solution to the problem of matching or unification over set-valued attributes, but it is a solution in the context of Extended Categorial Grammar. Set-valued unification or matching reduces to a set of unifications or matchings

$$(a \supset b) \vee \neg(a \;\&\; b)$$

$$\begin{bmatrix} \text{CAT:} & \begin{bmatrix} \text{CAT:} & a \\ \text{CONN:} & \supset \\ \text{CAT:} & b \end{bmatrix} \\ \\ \text{CONN:} & \vee \\ \\ \text{CAT:} & \begin{bmatrix} \text{CONN:} & \neg \\ \text{CAT:} & \begin{bmatrix} \text{CAT:} & a \\ \text{CONN:} & \& \\ \text{CAT:} & b \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

Figure 7.9
A Boolean L_p Expression as an Attribute-Value Matrix

of categories *without* set-valued attributes where every combination of potential pairs is represented under the constraint that each value must appear at least once. With respect to the evaluation of basic L_p expressions, the evaluation function returns a set of truth values, one for each of the attempted matches in the set of possibilities. Some examples will make this clear.

The reduction of category matching over two set-valued attributes each with two element sets to sets of category matches is illustrated in 7.15.

7.15 $[F\!: \{x, y\}] \not\forall [F\!: \{a, b\}]$

reduces to:

$\{ [F'\!: x] \not\forall [F'\!: a], [F'\!: y] \not\forall [F'\!: b] \}$

OR

$\{ [F'\!: x] \not\forall [F'\!: b], [F'\!: y] \not\forall [F'\!: a] \}$

where

x, y, a, b are categories of the grammar;

$\not\forall$ is the general category matching operation;

F' is exactly like F except that it is not set-valued;

The interpretation of this is as follows. If two categories each with a set-valued attribute containing two categories are matched, either of two situations can obtain, indicated by the disjunct. In each of these there will be enough pairings to cover all possibilities with every value being represented at least once. The match works if, for at least one of the disjunctive sets, every match in the set succeeds.

Category matching over two set-valued attributes, where one has a single element set, is given in 7.16 below.

7.16 $[F: \{x, y\}] \forall [F: \{a\}]$

reduces to:

$\{ [F': x] \forall [F': a], [F': y] \forall [F': a] \}$

where

x, y, a are categories of the grammar;

$\forall$ is the general category matching operation;

F' is exactly like F except that it is not set-valued;

This reduction is fully general insofar as it provides a way of matching any two categories each with any number of arguments. The two cases of interest in Extended Categorial Grammar are, however, the two cases shown in 7.15 and 7.16 above. The former covers cases where a function takes a function argument which must, therefore, match with a category in the function's ArgSet; the latter covers the case of L_p statements.

One further piece of notation is required before describing how *T-Eval* works in detail. When unification fails, a *failure path* is constructed terminating at the point of failure. This may be illustrated using the abbreviated categories in 7.17a and 7.17b. These two categories will not unify because of the clash between the respective values of the GR attribute.

7.17 a

$$\left[\text{ArgSet:}\ \left\{ \begin{bmatrix} \text{Category:} & \begin{bmatrix} \text{Return:} & \begin{bmatrix} \text{N:} & + \\ \text{V:} & - \\ \text{GR:} & \text{subj} \end{bmatrix} \\ \text{ArgSet:} & \{\} \\ \text{Lp:} & \text{NIL} \end{bmatrix} \\ \text{DIR:} & / \end{bmatrix} \right\} \right]$$

b

$$\left[\text{ArgSet:}\ \left\{ \begin{bmatrix} \text{Category:} & \begin{bmatrix} \text{Return:} & \begin{bmatrix} \text{N:} & + \\ \text{V:} & - \\ \text{GR:} & \text{obj} \end{bmatrix} \\ \text{ArgSet:} & \{\} \\ \text{Lp:} & \text{NIL} \end{bmatrix} \\ \text{DIR:} & / \end{bmatrix} \right\} \right]$$

The failure path is a list of those attributes the two categories have in common between the highest level attribute and the lowest (or most deeply embedded) attribute whose values fail to unify. In the case of the two categories in 7.17, the failure path is that given in 7.18 below.

7.18 <ArgSet, Category, Return, GR>

Given the basic notions of category matching and failure path, it is possible to present the algorithm for the evaluation of the first type of basic L_p expression. However, before doing that, it will be useful to discuss the evaluation of these expressions informally.

7.16 above showed the case relevant to L_p expressions where a category with a set-valued attribute with a single element in the set (the basic L_p expression) is matched against a set-valued attribute with two elements in the set (the containing function). Of course, this last attribute may have one or more elements in its set, but so far all the examples in this chapter have made use of functions of two arguments. The principles outlined here are not affected by the number of arguments in a function's ArgSet. As indicated above, this set matching reduces to a single set of matches. The idea behind the algorithm is that each of these matches evaluates to a truth value. Consequently, a set of truth values is initially returned. Three situations can obtain. The truth value set might be empty. In this case *T-Eval* returns F. The truth value set might contain one value, in which case that is the truth value returned by *T-Eval* for the basic expression. Finally, there may be more than one value in the truth value set, in which case *T-Eval* returns the *highest* value on the scale "F $<$ i $<$ T". The reason for this choice is discussed in relation to one of the examples below but, in effect, it amounts to existential quantification over the categories in ArgSet. It is possible to write L_p constraints on Extended Categorial Grammar categories using the equivalent of either existential or universal quantification but the constraints given in Chapter 5 for Spanish assumed the weaker, existential form.

The range of possible basic L_p expressions and ArgSet configurations can be illustrated using an ArgSet of one element. The possibilities are shown in 7.19 below.

7.19	a	$\{X^1\}_{1}$	$\Rightarrow$	T
	b	$\{X^1\}_{\backslash 1}$	$\Rightarrow$	i
	c	$\{\backslash X^1\}_{1}$	$\Rightarrow$	T
	d	$\{\backslash X^1\}_{\backslash 1}$	$\Rightarrow$	T
	e	$\{\backslash X^1\}_{/1}$	$\Rightarrow$	F
	f	$\{X^1\}_{2}$	$\Rightarrow$	F
	g	$\{X^1\}_{\backslash 2}$	$\Rightarrow$	no truth value

7.19a and 7.19c each has a non-directional L_p expression and an argument in ArgSet with which it unifies. In the case of 7.19a, the argset category is unsatisfied (has no directionality marking); in 7.19c the function has already accepted an argument of type X. Standardly, basic L_p expressions of the first type are directional since their primary purpose is to define and constrain the ordering of arguments relative to the function. However, non-directional expressions of this sort can be used simply to assert the existence of a particular category type in the function's ArgSet. If a match is found for such an argument category, whether or not it has a DIR value, *T-Eval* should return T for this sort of basic expression. It is possible, of course, for a non-directional basic expression to find no category in ArgSet with which it matches. This would imply a badly formed L_p expression and the truth value set associated with this expression would then be the empty set, in which case *T-Eval* will return F. This case is illustrated by 7.19f.

7.19b illustrates the case where a directional basic L_p expression matches an argument in ArgSet which has not yet been satisfied. In this case *T-Eval* returns i. 7.19d is the case where the basic L_p expression matches an argument which has a DIR value. The very fact that the match is successful indicates that the DIR values on the basic expression and the argument must be the same. *T-Eval* returns T in this instance. 7.19e illustrates a case similar to 7.19d except that the two DIR values clash. In this case *T-Eval* returns F. Finally, 7.19g stands for all the cases where a directional basic L_p expression fails to match with an argument because of a clash between the categories rather than between the DIR values. In all such cases, no entry is made in the truth value set associated with the basic expression since the argument is not relevant to the evaluation of the expression.

Given this informal background, the algorithm for assigning truth values to the first type of basic expression is presented in Figure 7.9. Different actions are taken by *T-Eval* depending on the success or failure of the category match. If it succeeds, two tests are made of the containing function category (f_p) and the basic expression category (f_{lp}), respectively. If f_{lp} contains structure not in f_p this can only mean that f_{lp} has a value for the DIR attribute which is

---

For each attempted match, let f_{lp} be the basic L_p expression and f_p be the function category in which it is contained.

if *category matching succeeds*, then

> **if** f_{lp} contains structure not in f_p
>
> > **then** enter i in the truth value set associated with the basic expression;
>
> **else** enter T in the truth value set associated with the basic expression;

if *category matching fails*, then

> **if** <ArgSet, Category, ...> is in the failure path
>
> > **then** make no entry in the truth value set associated with the basic expression;
>
> **else** enter F in the truth value set associated with the basic expression;

Figure 7.9
Algorithm for the Evaluation of Basic L_p Expressions of the First Type

---

not instantiated in f_p. In this case i is entered in the truth value set associated with the basic expression. If neither of these situations hold, the basic expression is evaluated to T.

If the category match fails and that failure is due to a clash between the values of the Category attribute no entry is made in the truth value set since f_p is not relevant to the evaluation of f_{lp}. On the other hand, if the failure to match is due to any other reason, F is entered in the truth value set associated with the basic L_p expression.

Some examples of the application of the algorithm will show how it works with actual function categories. The abbreviatory convention for function categories adopted earlier is used throughout the examples.

Consider the category in 7.19.

7.19 NP {\NPsubj^1, NPobj^2}
\1 & /2

Evaluation of the first conjunct reduces to the set of category matches in 7.20

7.20 { [ArgSet: \NPsubj^1] $\not\forall$ [ArgSet: \1] ,
[ArgSet: NPobj^2] $\not\forall$ [ArgSet: \1] }

where the first category in each matching pair is a partial specification of f_p and the second is f_{lp}. The first match in this set succeeds, f_{lp} does not contain structure not in f_p, therefore T is entered in the truth value set associated with the basic expression. The second match in this set fails and the failure path contains <ArgSet, Category, ...>, therefore no entry is made in the truth value set. The result is that the truth value set associated with the first conjunct is {T} and therefore T is returned by *T-Eval*.

Evaluation of the second conjunct reduces to the set of category matches in 7.21.

$$7.21\quad \{\,[\text{ArgSet: }\backslash\text{NPsubj}^{1}] \not\forall [\text{ArgSet: }/2]\,,$$

$$[\text{ArgSet: NPobj}^{2}] \not\forall [\text{ArgSet: }/2]\,\}$$

The first match in this set fails giving a failure path containing <ArgSet, Category, ...> and therefore making no entry in the truth value set. The second match succeeds with f_{lp} containing structure not present in f_p. Consequently, i is added to the truth value set. The result is that the truth value set associated with the second conjunct is {i} and therefore i is returned by *T-Eval*. This means that the entire L_p expression in 7.19 becomes "T & i", which evaluates to i, the desired result since, clearly, the function in 7.19 is one which can accept an NP to the left so long as it can match with the subject NP of the functor's ArgSet, as here.

Next, consider the function in 7.22 below.

$$7.22\quad \underset{\backslash 1\ \supset\ /2}{\{\text{NPsubj}^{1},\ /\text{NPobj}^{2}\}}\qquad \text{NP}$$

Evaluation of the antecedent reduces to the set of category matches in 7.23.

$$7.23\quad \{\,[\text{ArgSet: NPsubj}^{1}] \not\forall [\text{ArgSet: }\backslash 1]\,,$$

$$[\text{ArgSet: }/\text{NPobj}^{2}] \not\forall [\text{ArgSet: }\backslash 1]\,\}$$

The first match in this set succeeds, f_{lp} contains information not in f_p (ie. the DIR attribute and value), therefore, i is entered in the truth value set. The second match fails giving a failure path containing <ArgSet, Category, ...> and therefore no entry is made in the truth value set. Consequently, i is returned as the truth value of the antecedent.

Evaluation of the consequent reduces to the set of category matches in 7.24.

$$7.24\quad \{\,[\text{ArgSet: NPsubj}^{1}] \not\forall [\text{ArgSet: }/2]\,,$$

$$[\text{ArgSet: }/\text{NPobj}^{2}] \not\forall [\text{ArgSet: }/2]\,\}$$

The first match in this set fails with a failure path containing <ArgSet, Category, ...>, therefore no entry is added to the truth value set. The second match succeeds with f_{lp} not containing structure not in f_p and T is therefore entered into the truth value set for this basic expression. The resulting truth value set for the consequent is {T} with T therefore being returned by *T-Eval*. The L_p expression in the category in 7.22 therefore becomes i $\supset$ T which evaluates to T.

This is an interesting outcome. Consider the category in 7.22 in more detail. It is not a fully saturated function, having accepted only one argument to its right. It nevertheless evaluates to T. The reason is that this function will accept a direct object NP to its right no matter from which direction its subject finally comes. This is illustrated in 7.25 where the sequences accepted and not accepted by this function (labelled f) are indicated.

7.25 NPsubj f NPobj
*NPsubj NPobj f
NPobj f NPsubj
*NPobj NPsubj f
f NPsubj NPobj
f NPobj NPsubj

The sequence f NPobj is decidable, as indicated by the first and last orderings in 7.25, no matter where the subject comes from. This is not true of other sequences, such as NPobj f. The acceptability of this depends on where the subject will finally come from, as shown by comparing the second and third orderings in 7.25.

Now consider the function category in 7.26.

7.26 {NPsubj, /NPobj} NP
\NP

This category does not use re-entrant values in its L_p constraint and since the function takes two arguments, it might be expected that *T-Eval* would return a truth value set containing more than one truth value, and this is in fact the case. There are two possible interpretations of the L_p constraint in 7.26: either it constrains *all* NPs to be accepted from the left, or it constrains at least one NP to be accepted from the left. In order to maintain consistency with the model developed in Part Two, the latter interpretation is adopted in Extended Categorial Grammar. So, word orders involving the function in 7.26 will be well-formed if at least one NP argument is to the left of the function. The effect of this in the present context, as pointed out earlier, is that the highest truth value in the truth value set is returned as the final truth value for basic expressions. If universal quantification were required, this would mean that the lowest truth value in the truth value set would have to be selected.

Evaluation of the basic expression reduces to the set of category matches in 7.27 below.

7.27 { [ArgSet: NPsubj] $\not\forall$ [ArgSet: \NP] ,
[ArgSet: /NPobj] $\not\forall$ [ArgSet: \NP] }

The first match succeeds, f_{lp} contains structure not in f_p, therefore i is entered in the truth value set. The second match fails with <ArgSet, Category, ...> not in the failure path, therefore F is entered in the truth value set of the L_p expression. The resulting truth value set is {i, F} and since the highest value is selected, i is the final evaluation returned by *T-Eval* for this attempted match, as required.

The examples above illustrated the way *T-Eval* evaluates basic L_p expressions of the first type. The second type of basic L_p expression requires a different algorithm, although it builds on the category matching of the algorithm for the first type of basic expressions. As explained earlier, the form $<\alpha,\beta>$ is interpreted to mean that the function must combine with an α category before combining with a β category. *T-Eval* has access to the information that a function has accepted an argument through the presence of the DIR attribute in the relevant argset category in ArgSet. The algorithm therefore needs to check for this attribute in categories.

Consider the function in 7.28 below.

7.28 {NPsubj^{1}, NPobj^{2}}
$<1, 2>$

This function will combine with an NPsubj and NPobj categories from any direction just so long as it combines with the NPsubj before combining with the NPobj. This means that the grammaticality pattern associated with this function is that given in 7.29 below, where the brackets indicate the first combination and therefore the coordination potential of the function.

7.29	a [NPsubj f] NPobj	*NPsubj [f NPobj]
	b *NPsubj [NPobj f]	
	c NPobj [f NPsubj]	*[NPobj f] NPsubj
	d NPobj [NPsubj f]	
	e [f NPsubj] NPobj	
	f *[f NPobj] NPsubj	

Each of the component categories α, β of the basic expression must return a value indicating whether or not a DIR attribute has been found in any category in the function's ArgSet with which the component category can match. Call a positive return *Dir* and a negative return *NoDir*. The four possibilities for the basic expression together with the truth value returned for the expression are given in 7.30.

7.30	a $\langle Dir, Dir \rangle$	$\Rightarrow$	**T**
	b $\langle Dir, NoDir \rangle$	$\Rightarrow$	**T**
	c $\langle NoDir, Dir \rangle$	$\Rightarrow$	**F**
	d $\langle NoDir, NoDir \rangle$	$\Rightarrow$	**i**

7.30b and 7.30c are self explanatory. For example, the values in 7.30c mean that the function has combined with a β category before combining with an α category: this must therefore return F. The two *Dir* values in 7.30a evaluate to T because there is only one way that a function could arrive at this distribution of *Dir* values: by having first combined with a α category. This distribution therefore returns T. In the case of 7.30d, the fact that neither of the category elements of the basic expression produce *Dir* values means that the function has yet to combine with these two category types and therefore i is the only possible value to return.

The algorithm for evaluating basic expressions of the second type is given in Figure 7.10. To see how this works out, consider the category in 7.28 again. With $f_{\mathrm{lp}} = \alpha$ in the basic expression, the set matching reduces to the set of matches in 7.31 below.

7.31 { [ArgSet: NPsubj^1] $\not\forall$ [ArgSet: 1]
[ArgSet: NPobj^2] $\not\forall$ [ArgSet: 1] }

The first match succeeds but there is no *Dir* attribute in f_{p}; the second match fails. Consequently, no truth value is returned for this setting of f_{lp}.

With $f_{\mathrm{lp}} = \beta$ in the basic expression, the set matching reduces to the set of matches in 7.32.

7.32 { [ArgSet: NPsubj^1] $\not\forall$ [ArgSet: 2]
[ArgSet: NPobj^2] $\not\forall$ [ArgSet: 2] }

In this case also, no truth value is returned. Consequently, i is returned as the evaluation of the basic expression in 7.28, as required.

Let f_{lp} be the basic L_p expression and f_p be the function category in which it is contained.

for $f_{lp} = \alpha$ in $<\alpha, \beta>$

if *category matching succeeds* then

if f_p has *Dir* value

then return T

for $f_{lp} = \beta$ in $<\alpha, \beta>$

if *category matching succeeds* then

if f_p has *Dir* value

then return F

if neither of these return a truth value, return i

Figure 7.10
Algorithm for the Evaluation of Basic L_p Expressions of the Second Type

It is easy to see how the introduction of DIR values into the argset categories of the function in 7.28 would produce a different evaluation.

This section concludes with a re-interpretation in Extended Categorial Grammar terms of the word order constraints for Spanish sentences involving discourse saliency information presented in Chapter 5. Recall that two linear ordering constraints were proposed for verbal functions. The first constrained the verb to a non-final position if a direct object were present. This L_p statement is repeated as 7.33a below. The second constrained the appearance of discourse saliency information. The L_p statement is repeated as 7.33b.

7.33 a NPobject $\supset$ /NP

b /NPsubj $\supset$ (f:GIVEN & /<..., GIVEN, ...>)

The interpretation of 7.33b given in Chapter 5 was as follows: if the function's argument stack contains a rightward looking subject NP, then the discourse saliency information on the function itself must be GIVEN and there must be at least one rightward looking GIVEN constituent in the function's argument stack. Otherwise, all distributions of grammatical relations and discourse saliency information are acceptable. Clearly, some of the terms of this interpretation need to be changed in Extended Categorial Grammar.

The Extended Categorial Grammar equivalent of 7.33a is 7.34.

7.34

$$\left[\text{ArgSet: } \left\{ \left[\text{Category: NPobj} \right] \right\} \right] \supset \left[\text{ArgSet: } \left\{ \begin{bmatrix} \text{Category:} & \text{NP} \\ \text{DIR:} & / \end{bmatrix} \right\} \right]$$

The equivalent of 7.33b is 7.35.

7.35

$$\left[\text{ArgSet: } \left\{ \begin{bmatrix} \text{Category:} & \text{NPsubj} \\ \text{DIR:} & / \end{bmatrix} \right\} \right] \supset$$

$$\left[\text{DS: GIVEN} \right] \quad \& \quad \left[\text{ArgSet: } \left\{ \begin{bmatrix} \text{DIR:} & / \\ \text{DS:} & \text{GIVEN} \end{bmatrix} \right\} \right]$$

Two examples will illustrate the way these constraints work.

Consider the partial category specification in 7.36 below.

7.36 $\{\text{NPsubj}^1, \text{NPobj}^2\}$
[DS: NEW]

This is to be read in conjunction with the two L_p constraints in 7.34 and 7.35 which are assumed to be conjoined. Further, the NPsubj and NPobj are taken to be re-entrant with the categories bearing those labels in the two L_p constraints. The function in 7.36 is, therefore, a verbal function looking for two arguments constrained in the ways defined by the statements in 7.34 and 7.35 and carrying a discourse saliency (DS) specification of NEW.

What happens when this function is offered an NP which is new information to the left? Firstly, either of the arguments will unify with the NP. There are, therefore, two potential output categories, reflecting potential indeterminacy in the parsing. Restricting attention to the case where the NP unifies with the NPsubj in the function's ArgSet, the category in 7.36 is instantiated to that in 7.37.

7.37	NP [DS: NEW]	{\NPsubj^1[DS: NEW], NPobj^2} [DS: NEW]

The evaluation of the L_p expressions proceeds as follows. The antecedent of 7.34 evaluates to T since it matches with NPobj and contains no more information than NPobj. The consequent returns a truth value set with two values : {F, i}. The first, because f_{lp} and f_p fail to match for the reason that the DIR clash, and the second because f_{lp} and f_p match with f_{lp} containing structure not in f_p. The result is that i is returned for the consequent. *T-Eval* therefore returns i for 7.34.

The antecedent of 7.35 evaluates to F which means that the entire expression evaluates to T. For completeness, the first conjunct of the consequent in 7.35 evaluates to F since the function has a DS value of NEW, and the second conjunct evaluates to i. The consequent therefore evaluates to F.

The conjunction of 7.34 and 7.35 evaluates to i which is the desired result since the function in 7.37 can accept an NP[DS: NEW] from the left but only conditionally. Notice that the condition does *not* depend on the discourse saliency constraint embodied in 7.35 since all settings of discourse saliency for the SVO order are acceptable, as established in Chapter 5. The constraint in 7.34 does, however, impose a conditionality on the acceptability of SV orders since the

OSV order is ungrammatical.

As a last example, consider the function and argument in 7.38 below.

7.38 {/NPsubj^1[DS: GIVEN], NPobj^2} NP
[DS: NEW] [DS: GIVEN]

The function has accepted an NP[DS: GIVEN] from the right as a potential subject. 7.34 evaluates to T since both the antecedent and consequent evaluate to T. The second L_p constraint is more complex. Its antecedent evaluates to T since the NPsubj is to the right. The first conjunct of the consequent evaluates to F, however, since the verb is [DS: NEW]. Of course, this alone is enough to ensure that the entire expression evaluates to F and consequently the conjunction of the two L_p constraints evaluates to F. This sequence of categories with this particular distribution of discourse saliency information is not grammatical, a fact established in Chapter 5.

7.4 The Combinators of Extended Categorial Grammar

This section addresses itself to the combinators of Extended Categorial Grammar. It would be possible to use the range of combinators available in Generalised Categorial Grammar that were discussed in Chapter 1, though modified to take account of the fact that functions take sets of arguments. However, it is suggested here that several of these combinators can be collapsed into a single rule schema. Motivation for this view and the issues arising from it are the principal topics of this section.

The issue of generalising over the combinators can be looked at from two angles: from the point of view of conventional Generalised Categorial Grammar using curried functions; or, from the point of view of the Extended Categorial Grammar model. The advantage of examining the issue from the viewpoint of curried functions is that it allows certain of the logical underpinnings of the generalisations to be drawn out more clearly. Of course, for the work reported in this thesis, it is the generalisation in terms of Extended Categorial Grammar that is ultimately of importance. This section is consequently divided into three parts: in 7.4.1 a generalisation over Functional Application, Composition and Substitution is suggested within the framework of Generalised Categorial Grammar with curried functions; in 7.4.2 the same generalisation is presented and discussed from an Extended Categorial Grammar perspective; finally, in 7.4.3 some worked examples of the generalised combinator are offered.

7.4.1 A Generalised Categorial Grammar Rule Schema

It is argued here that Functional Application, Composition and Substitution are each special cases of the same combinatorial principle. This can be demonstrated using the curried versions of functions and by considering the conditional equivalences of categorial categories.

The category "X|Y" can be interpreted (as it standardly is) to mean that a category "X" can be asserted if there is a category "Y" (adjacent). This interpretation can be expressed using a conditional proposition as follows:

7.39 $Y \supset X$

In general, *any* complex Categorial Grammar category can be represented in a conditional form by reversing the two constituent categories and joining them with hook. Such a statement will be referred to as the *conditional equivalent* of the categorial category. So, the complex category in 7.40a can be represented as its conditional equivalent given in 7.40b.

7.40 a $(S|NP_1)|(S|NP_2)$

b $(NP_2 \supset S) \supset (NP_1 \supset S)$

The rule of functional application can therefore be stated as an expression of propositional logic in the following way:

7.41 $((Y \supset X) \ \& \ Y) \supset X$

which is a tautology of the propositional calculus if taken as a theorem or the rule of *modus ponens* if interpreted as a rule of inference.

Functional Composition is also a tautology of propositional logic, given by the expression in 7.42.

7.42 a $X|Y \ \ Y|Z \ \Rightarrow \ X|Z$

b $((Y \supset X) \ \& \ (Z \supset Y)) \supset (Z \supset X)$

(Standardly, this is one form of the rule of the syllogism).

Likewise, the rule of Functional Substitution introduced in Chapter 1 is a tautology given by the expression in 7.43.

7.43 a $(X|Y)|Z \ \ Y|Z \ \Rightarrow \ X|Z$

b $((Z \supset (Y \supset X)) \ \& \ (Z \supset Y)) \supset (Z \supset X)$

It turns out that *all* the combinatory rules which have been suggested for Generalised Categorial Grammar in the literature have, in fact, conditional equivalents in the propositional calculus which are tautologies. This is not surprising, of course, given the fact that it has been demonstrated that Generalised Categorial Grammar can be expressed as a sequent calculus (see Moortgat 1988b for detailed discussion). What is of interest here is the observation that the conditional form of Functional Application can be seen to be a special case of the conditional form of Functional Composition: that case where the second conjunct (either a basic expression or a conditional) is identical to the antecedent of the first conditional of the conjunct.

Consider the sequence of schematic non-directional combinatory-style rules in 7.44 below.

7.44 a $X|Y \quad Y \Rightarrow X$

b $X|Y \quad Y|Z \Rightarrow X|Z$

c $X|Y \quad Y|Z \Rightarrow (X|Z)|Z$

d $(X|Y)|Z \quad Y \Rightarrow X|Z$

e $(X|Y)|Z \quad Y|Z \Rightarrow X|Z$

f $(X|Y)|Z \quad Y|Z \Rightarrow (X|Z)|Z$

g $(X|Y)|Z \quad Y|W \Rightarrow (X|Z)|W$

Each of these rules has a conditional equivalent which is a tautology of the Propositional Calculus. 7.44a and 7.44b are, of course, Functional Application and Composition respectively. 7.44c is a variant of Composition which is a tautology because of the following strict equivalence in Propositional Logic:

7.45 $(Z \supset (Z \supset X)) \Leftrightarrow (Z \supset X)$

The number of categories of type Z which can be added in this way is unlimited so that the categorial forms $X|Y$, $(X|Y)|Y$, $((X|Y)|Y)|Y$ for instance are all logically equivalent though categorially distinct. This is far too liberal for natural language analysis and, although there are cases where a constrained use of this copying can be motivated (see Moortgat 1988b, page 47), in general, the output of a combinator should not allow redundant multiplication of identical arguments in this way. 7.44d is a rule that has not been put forward in the literature in precisely this form but can be seen as a variant of Functional Application where the Y argument is picked out internally rather than peripherally from the arguments of the function. The effect of this rule can be achieved in conventional Categorial Grammars by judicious use of Associativity. 7.44e is the rule of Functional Substitution and 7.44f is the same rule but with one extra Z category added to the output. It therefore parallels 7.44c and the same comments apply. 7.44g, which differs from 7.44e and 7.44f only with respect to the non-identity of the principal and dependent functor arguments, has not been suggested in the categorial literature but might be useful for the incremental parsing of VSO languages. Consider the Spanish sentence in 7.46

below.

7.46 vio el hombre a la mujer
(the man saw the woman)

The type assignments for this sequence of words is given in 7.47.

7.47	vio	$(S/NP_1)/NP_2$
	el	NP/N
	hombre	N
	a	NP/NP
	la	NP/N
	mujer	N

The incremental parsing strategy involves parsing a sentence in a strictly left to right manner where each new input category is fully integrated into the existing parse structure before proceeding to the next input. Clearly, the structure built will not correspond to conventional phrase structure and so any claim to be able to use constituent structure, say for coordination, will have to be given up. However, one of the strengths of Categorial Grammar is that is provides a mapping directly from the lexical type assignments to the semantic interpretation of the string via the semantics of the combinators without recourse to an "intermediate" phrase structure and that it is possible to build partial incremental semantic representations in this way. Incremental parsing is an area of active research in Categorial Grammar at the present time. The position taken in this thesis has been that coordination is evidence for constituent structure. It has therefore been decided to retain a notion of constituent structure as the domain for coordination.

The first two words of the sentence in 7.46 can be combined using Functional Composition to yield a category of type $(S/NP_1)/N$ looking for an N and NP. It is often assumed in curried Categorial Grammar representations that the order of the arguments reflects the grammatical relation of the NP arguments to the verb so that, for example, the last NP to combine with the verb is the subject, thereby incorporating the theory of grammatical relations in Keenan (1974). This is a way of deriving grammatical relations from the functional hierarchy rather than taking them as primitive notions as has been done in this thesis. However, notice that Functional Composition will give the wrong result under these assumptions since the definite article has composed with the NP which would standardly be regarded as the direct object NP of the verbal function. The rule in 7.44g, however, will give the correct results as indicated in 7.48.

7.48 $(S/NP_2)/N$
$\quad (S/NP_1)/NP_2 \ni$ **vio**
$\quad NP/N \ni$ **el**

This interesting set of rule types, which covers all the standard combinators as well as some extra rules which may well have independent motivation under certain parsing and processing assumptions but which also has the undesirable property of allowing unconstrained copying in the output, can be generalised by the schema in 7.49 below.

7.49 $(X|Y)|\alpha \quad Y|\beta \quad \Rightarrow \quad (X|\alpha)|\beta$

The way this works out for various values of α and β is shown in Figure 7.11. The characteristic of the schema is that the argument's Return value cancels an argument of the function with other arguments (represented by α and β) being carried over to the output. When $\alpha, \beta = \varnothing$ then the instantiation of the schema is Functional Application since the Y argument is simply cancelled from the function. Notice that in this schema variables stand for *any* arbitrary categories of the grammar. Consequently, the application in 7.50 below will go through with X, Y = (S/NP).

7.50 $(S/NP)/(S/NP) \quad S/NP \quad \Rightarrow \quad S/NP$

The second version, where α = Z, is the variant of Functional Application mentioned above where the argument is removed from *within* the function rather than peripherally. The third version of the schema where β = Z induces Functional Composition. The version where both α and β are instantiated to the

General Rule Schema: $(X|Y)|\alpha \quad Y|\beta \quad \Rightarrow \quad (X|\alpha)|\beta$

where X, Y, α, β are categories of the grammar.

Instantiations:

1 $\alpha, \beta = \varnothing$:

$X|Y \quad Y \quad \Rightarrow \quad X$

2 $\alpha = Z; \beta = \varnothing$:

$(X|Y)|Z \quad Y \Rightarrow \quad X|Z$

3 $\alpha = \varnothing; \beta = Z$:

$X|Y \quad Y|Z \quad \Rightarrow \quad X|Z$

4 $\alpha, \beta = Z$:

$(X|Y)|Z \quad Y|Z \Rightarrow \quad (X|Z)|Z$

5 $\alpha = Z, \beta = W$:

$(X|Y)|Z \quad Y|W \quad \Rightarrow \quad (X|Z)|W$

Figure 7.11
A Schema for Generalised Categorial Grammar Rules of Combination

same value induces a rule which resembles Functional Substitution but is not identical to it since it has too many arguments in the output. Within the conventional framework of curried category structure used in this section, Functional Substitution proper remains outside the scope of this generalisation. In 7.4.2 it will be shown how it can be brought within the scope of this generalisation in Extended Categorial Grammar.

Finally, the last version of the rule schema where both α and β have values but $\alpha \neq \beta$ has already been discussed in connection with the brief remarks above about incremental parsing strategies.

7.4.2 A Combinator for Extended Categorial Grammar

The purpose of this section is to adapt the observations above about the possibility of generalising over several of the combinatory rules of Generalised Categorial Grammar to the Extended Categorial Grammar framework. Transferring the generalisation of the rule schema in Figure 7.11 to Extended Categorial Grammar involves taking into account the set-valued ArgSet attribute of function categories. Consider the function-argument pair in 7.51.

7.51 [X { Y }] Y ⇒ [X { /Y }]

The abbreviatory conventions used in 7.51 are as follows. X and Y are basic categories of the grammar. Functions are represented by their Return value and the contents of their ArgSet. The function on the left in 7.51 is, therefore, looking for a Y argument. The output category shows that the Y has been accepted by marking the DIR value on the relevant argument. Further, recall that the combination of the two categories on the left in 7.51 would not have gone through unless *T-Eval* over the L_p constraint of the function (not given) had returned a non-false value.

7.51 is a case where the function has one argument, the argument has no arguments itself and therefore the only possible match is between the entire argument and the function's ArgSet category. Now consider 7.52 below.

7.52 [X {Y, Z}] Y ⇒ [X {/Y, Z}]

Here the function has two arguments, one of which matches with Y, resulting in the output shown. Clearly, both 7.51 and 7.52 are instances of Functional Application. The only difference between them being that the function in 7.52 may be taken as an example of 2 in Figure 7.11.

What happens when the argument itself has categories in its ArgSet? Consider 7.53.

7.53 [X {Y}] [Y {Z}] $\Rightarrow$ [X {/Y, Z}]

In terms of Figure 7.11, this is the third instantiation of the rule schema, with $\alpha = \varnothing$ and $\beta = Z$. It is, of course, an instance of Functional Composition. If α has a value in the function, 7.54 is the result.

7.54 [X {Y, Z}] [Y {Z}] $\Rightarrow$ [X {/Y, Z}]

This is related to the fourth instantiation of the rule schema but with one important difference: there is only one Z argument in the output. It therefore models Functional Substitution. This is valid on the assumption that the output category is constructed by taking the *union* of all the arguments not to be marked with a DIR value. If the two remaining arguments are different, like the fifth example in Figure 7.11, the result will be that shown in 7.55 below.

7.55 [X {Y, Z}] [Y {W}] $\Rightarrow$ [X {/Y, Z, W}]

The picture which emerges from this discussion is that two categories combine if either the entire argument category or its Return value match a category in the functions ArgSet. The output has this category neutralised by the introduction of a DIR value and the union of the remaining categories in both function and argument is taken.

This generalisation covers the examples above but is not complete. The situation is rather more complex than this. Consider the two categories in 7.56 below.

7.56 [X {Y, Z}] [Y {W, M}]

According to the combinatory algorithm in the previous paragraph, these should combine to produce the category in 7.57.

7.57 [X {/Y, Z, W, M}]

But this is wrong. The curried versions corresponding to the rightmost category in 7.56 (taken as the argument category) would be

7.58 a (Y|W)|M

b (Y|M)|W

and those corresponding to to the leftmost category in 7.56, taken as the function would be

7.59 a (X|Y)|Z

b (X|Z)|Y

Neither version of the function can accept either version of the argument. The function would, of course, have to have either Y|W or Y|M as a category in its ArgSet for the combination to go through. The change to the algorithm to ensure that all cases are covered is straightforward: a category in the function's ArgSet must combine with an argument category formed by taking either the entire argument or the category formed by removing at most one category from the argument's ArgSet. This is summed up in the combinator schemata given in Figures 7.12 and 7.13. Two versions of the combinator are given to reflect the directionality of the potential argument.

The schemata show that a category in the function's ArgSet must match with the potential argument and that the argument can have at most one category in its own ArgSet which is not part of the match. The output is a category in which the matched argument is given an appropriate DIR value and the union of the other categories in the two ArgSets is taken, thereby eliminating repetitions of identical categories. The L_p constraint for the output category consists of a conjunction of the L_p constraint for the function with the results of applying the operator f over the constraints of the argument. This operator simply selects the constraint relevant to the β category in the argument, filtering out constraints relating to other categories in the argument's ArgSet.

$$
\begin{bmatrix} \text{Return:} & Y \\ \text{ArgSet:} & \cup \; W, \beta \\ \text{Lp:} & e(\gamma) \end{bmatrix}
+
\begin{bmatrix} \text{Return:} & X \\ \text{ArgSet:} & \cup \begin{bmatrix} \text{Return:} & Y \\ \text{ArgSet:} & W \\ \text{Lp:} & e(\alpha) \end{bmatrix}, \alpha \\ \text{Lp:} & e(\delta) \end{bmatrix}
=>
$$

$$
\begin{bmatrix} \text{Return:} & X \\ \text{ArgSet:} & \cup \begin{bmatrix} \text{Category:} & \begin{bmatrix} \text{Return:} & Y \\ \text{ArgSet:} & W \\ \text{Lp:} & e(\alpha) \end{bmatrix} \\ \text{DIR} & \backslash \end{bmatrix}, \alpha , \beta \\ \text{Lp:} & e(\delta) \;\&\; e(\gamma) \end{bmatrix}
$$

Figure 7.12
The Extended Categorial Grammar Combinator Schema: Leftward Looking Version

$$
\begin{bmatrix}
\text{Return:} & X \\
\text{ArgSet:} & \cup \begin{bmatrix} \text{Return:} & Y \\ \text{ArgSet:} & W \\ \text{Lp:} & e(\alpha) \end{bmatrix}, \alpha \\
\text{Lp:} & e(\delta)
\end{bmatrix}
+
\begin{bmatrix}
\text{Return:} & Y \\
\text{ArgSet:} & \cup\ W, \beta \\
\text{Lp:} & e(\gamma)
\end{bmatrix}
\Rightarrow
$$

$$
\begin{bmatrix}
\text{Return:} & X \\
\text{ArgSet:} & \cup \begin{bmatrix} \text{Category:} & \begin{bmatrix} \text{Return:} & Y \\ \text{ArgSet:} & W \\ \text{Lp:} & e(\alpha) \end{bmatrix} \\ \text{DIR} & / \end{bmatrix}, \alpha, \beta \\
\text{Lp:} & e(\delta)\ \&\ e(\gamma)
\end{bmatrix}
$$

Figure 7.13
The Extended Categorial Grammar Combinator Schema: Rightward Looking Version

7.4.3 Parsing with the Combinator Schemata in Extended Categorial Grammar

Having presented the general schemata for function-argument combination in Extended Categorial Grammar, this section illustrates their operation through two concrete examples: Application and Composition.

Consider the abbreviated category in Figure 7.14. This is a function looking for either an NP or another function. Assume it is offered the category in Figure 7.15 as an argument. These two categories combine according to the rule schema with the following settings for the variables.

$$
\begin{bmatrix}
\text{Return:} & S \\[2ex]
\text{ArgSet:} & \left\{ \begin{bmatrix} \text{Category:} & \text{NPsubj}^{1} \end{bmatrix},\ \begin{bmatrix} \text{Category:} & \begin{bmatrix} \text{Return:} & S \\ \text{ArgSet:} & \left\{ \begin{bmatrix} \text{Category:} & \text{NPsubj}^{2} \end{bmatrix} \right\} \\ \text{Lp:} & \backslash 2 \end{bmatrix} \end{bmatrix}^{3} \right\} \\[2ex]
\text{Lp:} & \backslash 1 \supset /3
\end{bmatrix}
$$

Figure 7.14
A Hypothetical Function Category

$$
\begin{bmatrix}
\text{Return:} & S \\[2ex]
\text{ArgSet:} & \left\{ \text{NPsubj}^{1} \right\} \\[2ex]
\text{Lp:} & \backslash 1
\end{bmatrix}
$$

Figure 7.15
A Potential Argument for the Function in Figure 7.14

7.60 $X = S$
$Y = S$
$W = NP$
$\alpha = \text{NPsubj}$
$\beta = \varnothing$

The output category is given in Figure 7.16. The satisfied argument has an appropriate DIR value and the L_p constraint on the output is taken from the function only since no category remains in the ArgSet of the argument. It might be helpful at this point to spell this combinatorial process out using curried representations. The function in 7.13 has the set of curried functions in 7.61 as its equivalent.

7.61 a (S\NPsubj)/(S\NPsubj)
b (S/(S\NPsubj))\NPsubj
c (S/NPsubj)\(S\NPsubj)
d (S\(S\NPsubj))/NPsubj
e (S/NPsubj)/(S\NPsubj)
f (S/(S\NPsubj))/NPsubj

The output in Figure 7.16 is a function looking for an NP to be saturated. It can find that NP from either the left or right. This is the desired result since

$$
\begin{bmatrix}
\text{Return:} & \\
\text{ArgSet:} & \left\{ \begin{bmatrix} \text{Category:} & \begin{bmatrix} \text{Return:} & \text{S} \\ \text{ArgSet:} & \left\{ \text{NPsubj}^{2} \right\} \\ \text{Lp:} & \backslash 2 \end{bmatrix}^{3} \\ \text{DIR:} & / \end{bmatrix} , \begin{bmatrix} \text{Category:} & \text{NPsubj}^{1} \end{bmatrix} \right\} \\
\text{Lp:} & \backslash 1 \supset /3
\end{bmatrix}
$$

Figure 7.16
Output Category for the Combination of Categories in Figures 7.14 and 7.15

the two curried versions in 7.61a and 7.61c have as their output categories looking for their NP argument in either direction. The category in Figure 7.16 is simply a generalisation over these two curried functions.

Functional Composition is illustrated using the same function in Figure 7.14 and the function in Figure 7.17.

The settings of the variables are given in 7.61 below.

$$
\left[\begin{array}{ll}
\text{Return:} & S \\
\text{ArgSet:} & \left\{ \left[\text{Category: } NPsubj\right]^{1},\ \left[\text{Category: } \left[\begin{array}{ll} \text{Return:} & S \\ \text{ArgSet:} & \left\{\left[\text{Category: } NPsubj\right]^{2}\right\} \\ \text{Lp:} & \backslash 2 \end{array}\right]\right]^{3} \right\} \\
\text{Lp:} & \backslash 1 \supset /3
\end{array}\right]
$$

Figure 7.17
A Function to Compose with the Category in Figure 7.14

7.61 $X = S$

$Y = S$

$W = [\text{Category: NPsubj}^3]$

$\alpha = [\text{Category: NPsubj}]$

$\beta =$

$$\left[\text{Category:} \; \left[\begin{array}{ll} \text{Return:} & S \\ \text{ArgSet:} & \left\{ \text{NPsubj}^{1} \right\} \\ \text{Lp:} & \backslash 1 \end{array}\right]\right]$$

The output category is given in Figure 7.18.

$$\left[\begin{array}{ll} \text{Return:} & \\ \text{ArgSet:} & \left\{ \left[\begin{array}{ll} \text{Category:} & \left[\begin{array}{ll} \text{Return:} & S \\ \text{ArgSet:} & \left\{ \text{NPsubj}^{1} \right\} \\ \text{Lp:} & \backslash 1 \end{array}\right] \\ \text{DIR:} & / \end{array}\right]^{3}, \; \text{NPsubj}^{2}, \; \left[\text{Category:} \; \left[\begin{array}{ll} \text{Return:} & S \\ \text{ArgSet:} & \left\{ \text{NPsubj}^{4} \right\} \\ \text{Lp:} & \backslash 4 \end{array}\right]\right]^{6} \right\} \\ \text{Lp:} & (\backslash 2 \supset /3) \;\&\; /6 \end{array}\right]$$

Figure 7.18
Output of Functional Composition of the Categories in Figures 7.14 and 7.17

7.5 Extended Categorial Grammar and the Hierarchy of Categorial Grammars

Sections 7.2 to 7.4 presented the outline of a monostratal linear order factored version of Categorial Grammar. The intention was to show how certain problems with the simpler model presented in Part Two can be overcome by the use of formal devices such as category typing and matching. Part of the intention of this chapter was also that the formalism should be explicit enough to point towards an implementation. It is appropriate now to stand back and try to view the model presented here in the context of the hierarchy of Categorial Grammars outlined in Section 1.5.

Section 1.5 introduced a hierarchy of Categorial Grammars as it is presented in Moortgat (1988). It was pointed out there that the classical system **L** is too weak for natural language description but that the permutation closure of **L** - **LP** - is far too unconstrained since for any sequence of categories that can be parsed to a given type, *any* ordering of those categories will parse to that type. One of the principal aims of this thesis has been to display and analyse natural language data which calls for a degree of flexibility from the grammar formalism with respect to word order but where that flexibility is strictly constrained. One of the tasks of any syntactic model must be to provide a mechanism for describing such semi-free word order. **LP** cannot do that.

It will be useful at this point to categorise the properties of **L** which are relevant to the problem of placing Extended Categorial Grammar in the hierarchy of Categorial Grammars. The combinators of the product-free version of **L** which has been the subject of this thesis may be classified as in 7.62 below.

7.62 a Application
Composition

b Associativity

c Raising
Division

7.62a are the combinatory rules to which can be added Functional Substitution. The previous section showed how it was possible to generalise over these three rule types. The rule in 7.62b is particularly important in the present context. It is the source of the structural completeness of **L**, the property that guarantees that a sequence of categories which parses to a particular type will parse to that type for any bracketing of the category pairs. Finally 7.62c is the group of type changing rules which have been motivated from both a logical and linguistic perspective. They will not be discussed here since they are not the focus of attention in this thesis.

Moortgat (1988) investigates ways of defining Categorial Grammars in the logical space between **L** and **LP**. He considers two approaches to the problem. The first involves introducing specific theorems from **LP** into **L**. He shows, however, that the axioms motivated by linguistic phenomena such as discontinuous dependencies cause collapse into **LP**. The second approach involves the introduction of non-concatenative operations into **L**. The work reported here addresses itself to the same issue but from a different perspective: Extended Categorial Grammar has the generative power of **LP** but the L_p constraints permit the definition of grammars with generative powers between **L** and **LP**.

L_p constraints employing basic expressions of the form $\backslash\alpha$ and $/\alpha$ define the permutation possibilities of categories whereas expressions of the form $<\alpha, \beta>$, on the other hand, constrain the associativity of categories.

CHAPTER 8

THE EXTENDED CATEGORIAL GRAMMAR ANALYSIS OF SEMI-FREE WORD ORDER

8.1 Introduction

The purpose of this chapter is to examine a range of data from Spanish using the Extended Categorial Grammar model developed in the previous chapter. The analysis of word order in Spanish transitive sentences in Chapter 5 left a number of issues unaddressed: in particular, the categorial status of the particle *a*, the role of the determiners and the interaction of these with the process of Grammatical Relation assignment and word order constraints. As a first step towards a more complete analysis of Spanish, it is suggested in Section 8.2 that appropriate generalisations can be expressed about the particle *a* by factoring the conventional Grammatical Relation assignment of subject, direct and indirect object into two more primitive features. This innovation is then used as the basis for a detailed examination of the Spanish data.

In Sections 8.3 the problem of describing word order variation within languages in the Extended Categorial Grammar framework is briefly addressed. The word order data presented for Spanish in Chapter 2, which formed the basis of the analysis in Chapter 5, involved certain decisions being made about the acceptable sequences in Spanish. Basically, it was decided to take the broadest view of grammaticality that had been found among informants. This amounted to accepting all orderings of S, V and O except the verb final ones. However, this is an area where there is considerable variation between speakers. Extended Categorial Grammar should be able to model these differences and thereby allow comparisons to be made. These are discussed and L_p statements for the various orderings are offered.

Finally, the presence in sentences of personal pronoun clitics (referred to here as pronominal clitics or, simply, clitics) has significant effects on the ordering of arguments relative to verbal functions and so has important implications for the statement of linear ordering constraints. These issues, together with the issue of the categorial status of clitics form the topic of Section 8.4.

8.2 The Analysis of Basic Sentences in Spanish: The Roles of *a* and the Determiners

The analysis of word order in Chapter 5 was conducted primarily on the basis of simple transitive sentences. The underlying idea was that the Noun Phrase argument specifications in the verbal function's ArgSet would carry information about the Grammatical Relations that potential arguments might take. This relieves the Noun Phrase arguments themselves from having to have an explicit Grammatical Relation assignment. There are two drawbacks to this approach. Firstly, it leads to indeterminacy in parsing. When a Noun Phrase becomes available as an argument to a transitive verb, it is assumed to be underspecified with regard to the categories in the function's ArgSet, particularly for Grammatical Relations. Consequently, it may match with either the subject or direct object if both are unsaturated, a process potentially involving the hypothesising of two output categories. Secondly, the claim that Noun phrases in Spanish are neutral with respect to Grammatical Relation assignment is counterfactual since the Grammatical Relation a Noun Phrase may bear to a verb is to some extent predictable independently of any specific context in which the Noun Phrase occurs. Showing how this second point resolves itself in Spanish also shows how the problem over indeterminacy can be reduced or even eliminated in many situations. The particle *a* plays a key role in the assignment of Grammatical Relations and the presence of the definite and indefinite determiners is also relevant.

In Chapter 2 evidence was presented from Suñer (1988) to show that the particle *a* when used with indirect objects is *not* a preposition, as is sometimes assumed. The conclusion to be drawn from that discussion is that both direct and indirect objects are Noun Phrases and that the particle *a* has the dual role of assigning direct and indirect object Grammatical Relations to Noun Phrases. Without further analysis this would mean positing two homophonous items which assign different Grammatical Relations. The discussion following shows how the Extended Categorial Grammar framework, together with a new approach to factoring Grammatical Relations into more primitive concepts, provides the means of analysing the particle *a* as essentially a single functional category.

As indicated above, the Grammatical Relation that a Noun Phrase may bear to a verb is to some extent predictable in Spanish independently of any specific context in which the Noun Phrase may occur. For example the Noun Phrase *María* can only be the subject of a sentence, as in 8.1a, never the direct object (hence the judgement in 8.1b) or indirect object (hence 8.1c).

8.1 a **María** cantó la canción
(Mary sang the song)

b *el hombre vio **María**
(the man saw Mary)

c *Juan dio el libro **María**
(John gave the book to Mary)

An animate Noun Phrase can only be the direct or indirect object of a verb if it is preceded by the particle *a*, a fact illustrated in Chapter 2. The corollary of this is that no Noun Phrase preceded by the particle *a* can be the subject of a sentence. The inanimate Noun Phrase *La canción*, however, is potentially either a direct object, as in 8.1a, or a subject, as in 8.2.

8.2 **La canción** pareció sin melodía y poco interesante
(the song seemed tuneless and uninteresting)

This array of facts suggests that the tripartite distinction between subjects, direct and indirect objects will prove inadequate when it is necessary to state generalisations over direct and indirect objects on the one hand (eg. *a María*) and subject and direct objects on the other (eg. *la canción*). To overcome this problem, it is suggested here that the Grammatical Relations used so far are decomposed into two primitive binary features called *Primary* and *Objective*.

The Primary feature serves to distinguish subjects and direct objects on the one hand from indirect objects and more peripheral arguments of verbs on the other hand. The motivation for this distinction derives from the linguistic typology literature. Andrews (1985) is an excellent survey of the issues involved. In common with many authors in this field, he derives Grammatical Relations from more fundamental semantic and semiotic functions of language:

> The semantic roles and pragmatic functions of the NPs in a sentence may be called their "semiotic" functions, since they have to do with the meaning of the sentences. Semiotic functions are ultimately signalled by "overt coding features" such as word order, case marking and cross-referencing (agreement). But it is difficult to provide a coherent account of how this occurs in terms of a direct connection between coding features and the semiotic functions they express. Rather it seems better to posit an intervening level of grammatical structure: the coding features indicate the grammatical structure of the sentence, and the grammatical structure determines the semiotic functions.
>
> (Andrews 1985, page 63)

The "grammatical functions of the NPs are the relationships in this grammatical structure which participate in determining the semantic and pragmatic functions of the NPs ... and are involved in governing the form of sentence structure" (page 63). What Andrews calls "grammatical functions" play a pivotal role in determining the structural and "semiotic" properties of sentences.

One class of sentences is frequently taken as basic because of its universality: the class of two-argument verbs taking an Agent and Patient. Andrews refers to these as "primary transitive verbs (PTVs)".

> Languages always seem to have a standard way or small set of ways in which they normally express the Agent and Patient of a PTV. If an NP is serving as argument of a two-argument verb, and receiving the morphological and syntactic treatment normally accorded to an Agent of a PTV, we shall say that it has the grammatical function A; if it is an argument of a verb with two or more arguments receiving the treatment normally accorded to the Patient of a PTV, we shall say that it has the grammatical function O. A sentence is called "transitive" if it has A and O functions in its syntactic structure, "intransitive" if one or both functions is missing... An NP in an intransitive sentence that is receiving the treatment normally accorded to the single argument of a one-argument predicate will be said to have S function.
>
> (Andrews 1985, page 68)

These grammatical functions are related to Grammatical Relations in different ways in different languages: "Most often, one finds one Grammatical Relation associated with A and S, and another with O. The former sort of Grammatical Relation we will call *subject*, the latter *object*" (Andrews 1985, page 69). Since Spanish is a Nominative/Accusative rather than an Ergative language, these associations will be adopted here.

It is clear that, from this perspective, the subject and direct object share a mutual significance in the mapping between semiotic function and grammatical structure. Other arguments of the verb do not have this centrally important role. It seems legitimate, therefore, to distinguish the subject and direct object from other subcategorised roles and this is exactly the purpose of the Primary feature.

The Objective feature is motivated in a rather different way. Many languages exhibit so-called subject-object asymmetries with respect to grammatical processes such as agreement and control of reflexivisation. From a typological perspective, Tomlin (1986) provides extensive evidence for this asymmetry sufficient to raise it to one of three fundamental principles that "shape the grammars of natural languages" (Tomlin 1986, page 73):

The principle of Verb-Object Bonding is presented in (1):

> (1) Verb-Object Bonding (VOB): the object of a transitive verb is more tightly bonded to the verb than its subject.

> This principle claims that a transitive verb and its object form a more cohesive, unified syntactic and semantic whole than do a transitive verb and its subject. The data ... show that it is more difficult to interfere with the syntactic unity of the verb and its object, by attempting syntactic insertions, movements and so on, than to interfere with any such possible unity between the verb and its subject. They show further that there is also a greater semantic unity between verb and object than between verb and subject.
>
> (Tomlin 1986, page 74)

This insight is also encoded explicitly in some recent theories such as Government and Binding by treating the subject differently to the strictly subcategorised constituents of verbs: subjects have "external" θ-roles (Agent, Patient etc) while other subcategorised objects, including the direct and indirect object, are said to have "internal" θ-roles (Cook 1988, page 115). The Objective feature identifies the non-subject subcategorised arguments of the verb.

The assignment of Grammatical Relations to the distribution of values for Primary and Objective is given in Table 8.1. It is clear from Table 8.1 how this distribution of the two features will allow for the generalisations mentioned above: direct and indirect objects are [+Objective]; subjects and direct objects are [+Primary].

The assignment of these features Primary and Objective to different Noun Phrase types is shown in Table 8.2. Bare animate Noun Phrases such as *María* can only be subjects, they are therefore [+Primary, -Objective]. The particle *a* can only occur with animate Noun Phrases to form Noun Phrases which are either direct or indirect objects. The resulting Noun Phrase is consequently [+Objective] with no restriction on the Primary feature.

	Subject	Direct Object	Indirect Object
Primary	+	+	-
Objective	-	+	+

Table 8.1
The Decomposition of Grammatical Relations into Primary and Objective Features

Noun Phrase Type	Examples	Possible GRs	Primary Feature	Objective Feature
NP[+Animate]	María	subject	+	-
a NP[+Animate]	a María	direct object indirect object	+/-	+
Det N[+Animate]	la mujer	subject	+	-
Det N[-Animate]	la canción	subject direct object	+	+/-

Table 8.2
Assignment of Features to the Principal Noun Phrase Types

The definite and indefinite determiners are more complex. Table 8.2 shows that the potential Grammatical Relations of a Noun Phrase with a determiner at its leading edge depends on the animacy of the Noun. Such a Noun Phrase which is [+Animate] can only have the Grammatical Relation subject whereas if the Noun Phrase is [-Animate], it can be either subject or direct object. Notice that a Noun Phrase such as *a la mujer*, which is not explicitly given in Table 8.2, can be thought of as composed of a Noun Phrase, *la mujer*, which would be a potential subject, prefixed by *a* which causes it to take on the

Grammatical Relations of direct or indirect object, exactly analogous to *María* and *a María*.

How can these distributions be secured in the grammar of Spanish without resort to excessive feature stipulation in the lexicon? Firstly, *a* and the definite and indefinite determiners are taken to be functions over Noun Phrases and Nouns respectively; each function returning a Noun Phrase. The category for *a* is given in Figure 8.1. The abbreviatory conventions used here and throughout this Chapter are as follows: the alias NP stands for a basic category of the form in 8.3.

8.3 $\quad \text{NP} \quad = \quad \left[\begin{array}{ll} \text{Return:} & \left[\begin{array}{ll} \text{N:} & + \\ \text{V:} & - \\ \text{Phrasal:} & + \end{array} \right] \end{array} \right]$

$$\left[\begin{array}{ll} \text{Return:} & \text{NP[+Animate, +Objective]} \\ & \\ \text{ArgSet:} & \left\{ \left[\text{Category: NP[+Animate]} \quad \boxed{1} \right] \right\} \\ & \\ \text{Lp:} & / \; \boxed{1} \end{array} \right]$$

Figure 8.1
The Category for the Particle *a*

The attribute-value assignments (where ±F is used as a short form for [F: ±]) following the alias are regarded as part of the primitive return category. The abbreviated form NP[+animate, +Objective] is therefore the category in 8.4 below.

8.4

$$\begin{array}{l}\text{NP}\\ \text{[+Animate]}\\ \text{[+Objective]}\end{array} = \begin{bmatrix} \textbf{Return:} & \begin{bmatrix} \text{N:} & + \\ \text{V:} & - \\ \text{Phrasal:} & + \\ \text{Animate:} & + \\ \text{Objective:} & + \end{bmatrix} \end{bmatrix}$$

The function in Figure 8.1 therefore generalises over the use of *a* with direct objects as well as indirect objects, in keeping with the discussion of data from Suñer (1988) mentioned earlier.

The determiners might be analysed as two categories (one returning a definite Noun Phrase, the other an indefinite), each with the feature analysis of the returned Noun Phrase being in part dependent on the animacy of the Noun argument. A different solution is adopted here, however. The general feature constraint in 8.5 below is added to the grammar.

8.5 NP[+Animate] $\Rightarrow$ NP[+Primary, -Objective]

The interpretation of this is as follows: whenever a Noun Phrase is instantiated for features, if it is [+Animate] it is also [+Primary, -Objective] *unless this clashes with features already instantiated on the category.* This is primarily a default constraint which applies to categories at parse time, since that is when values of these features are added to categories. Notice also that this constraint is thought of as applying to *all* Noun Phrase categories including those which form the Return and ArgSet values of functions. The examples below will illustrate its effects.

The results of introducing this constraint are twofold: lexical entries can be simplified and a single category each for the definite and indefinite determiners suffices. An example of simplification in the lexicon is offered by the category for *a* in Figure 8.1. The argument to this function is any NP[+Animate] but *a* will not combine with *any* NP[+Animate]. Essentially, it is a Grammatical Relation changing particle, changing subjects to direct or indirect objects. It will, therefore, only combine with an NP[+Animate, +Primary, -Objective]. This is ensured by the default in 8.5 since the Noun Phrase argument will default to these values. By contrast, the Return value of this function is given as NP[+Animate, +Objective]. Since this feature complex conflicts with the output of 8.5, the default instantiation does not apply and the Return value is a partial specification of direct and indirect object categories, as required.

The category for the Definite Determiner is given in Figure 8.2 below where α is a variable over binary atomic values.

Lexical entries for some Proper and Common Nouns and the transitive verbs *cantar* and *ver* are given in Figure 8.3. Again, the default in 8.5 will

$$\left[\begin{array}{ll} \text{Return:} & \text{NP[}\alpha\text{Animate, +Definite]} \\ \text{ArgSet:} & \left\{\left[\text{Category: NP[}\alpha\text{Animate]} \quad \boxed{1}\right]\right\} \\ \text{Lp:} & / \; \boxed{1} \end{array}\right]$$

Figure 8.2
The Category for the Definite Determiner

instantiate María with [+Primary, -Objective] features. The verbs have two NP arguments with full Grammatical Relation specification. The L_p statements for these transitive verbs are left unspecified here since they are not relevant to the present discussion and have been fully described in earlier chapters.

Details of the analysis of some basic sentence types showing how the Grammatical Relations of Noun Phrases are correctly predicted from the lexical entries and default in 8.5 are discussed in the following paragraphs.

Consider the sentence in 8.6 below.

8.6 María cantó la canción
(Mary sang the song)

María (Mary): NP[+Animate]

canción (song): N[-Animate]

mujer (woman): N[+Animate]

cantar (to sing)
ver (to see):

$$\left[\begin{array}{ll} \text{Return:} & \text{S} \\ \\ \text{ArgSet:} & \left\{ \left[\text{Category:}\ \ \begin{array}{c}\text{NP}\\ \text{[+Primary]}\\ \text{[-Objective]}\end{array}\right], \left[\text{Category:}\ \ \begin{array}{c}\text{NP}\\ \text{[+Primary]}\\ \text{[+Objective]}\end{array}\right] \right\} \\ \\ \text{Lp:} & \text{e(y)} \end{array}\right]$$

Figure 8.3
Some Lexical Entries for an Extended Categorial Grammar

In this sentence *María* can only be the subject whereas *la canción* is potentially a subject or direct object. There is consequently only one possible distribution of Grammatical Relations over the NP arguments. From the point of view of the present discussion, an interesting word order variant of this sentence is the one given in 8.7 which is, of course, acceptable according to the L_p constraints for Spanish.

8.7 cantó la canción María

During the parsing of the sentence in 8.7, the assignment of Grammatical Relation to the first Noun Phrase will be non-deterministic since it can be subject or direct object. Only when the second Noun Phrase is accepted is a non-ambiguous assignment of Grammatical Relations possible. Of course, other features (such as animacy) are relevant to the assignment of Grammatical Relations and these may well conspire to exclude *la canción* as a suitable subject for *cantar*. This, however, involves drawing on semantic and possibly real-world knowledge which is not being considered here.

The sentence in 8.8 below offers different problems.

8.8 ?*a la mujer cantó la canción
(?*the song sang the woman)

a la mujer cannot be the subject of the sentence but *la canción* can be subject or direct object. This means that this sentence can parse with an OVS assignment. This syntactic possibility, however, breaks two semantic constraints on the choice of subject and direct object. Contrast this with the sentence in 8.9.

8.9 A Juan vio María
(Mary saw John)

This has the same Grammatical Relation assignment as 8.8 with an OVS order but does not break any semantic selectional restrictions. It is therefore acceptable syntactically and semantically.

The sentences in 8.10 are both unacceptable.

8.10 a *María cantó a la canción

b *María vio Juan

The particle *a* cannot occur with inanimate Noun Phrases (hence the ungrammaticality of 8.10a) and in 8.10b there are two conflicting subjects with no potential direct object.

Finally, consider the sentences in 8.11.

8.11 a María (le) dio el libro a Pedro
(Mary gave the book to Peter)

b el gobernador envió a la esclava al emperador
(the governor sent the slave to the emperor)

The assignment of Grammatical Relations in 8.11a is unambiguous even though the Grammatical Relations of both *el libro* and *a Pedro* are ambiguous: the former can be subject or direct object, the latter direct or indirect object. The unambiguous distribution is locked into place by the fact that *María* can only be the subject. On the other hand, although a single translation has been given for 8.11b, it is in fact ambiguous. Both phrases introduced by *a* can be either direct or indirect objects. Grammatical Relation assignment therefore results from non-syntactic information such as the tendency in unmarked contexts to order direct before indirect objects re-enforced by the real-world knowledge that the slave is more likely to be sent to the emperor than the other way round.

8.3 Word Order in Spanish: Some Further Observations

In Chapter 2 certain decisions were made about the word order possibilities in simple transitive sentences in Spanish. Basically, these involved the generalisation that all orderings are acceptable so long as a Noun Phrase argument follows the verb. There is considerable support for this among the informants used for collecting data for this study. However, it is important to recognise that this generalisation is by no means acceptable to all Spanish speakers and that other constraints are frequently cited in the literature. Armed with a model for the description of complex word orderings, it is now possible to address this problem of word order variation directly and this is explored in Section 8.3.1.

Another aspect of the problem of word order in transitive sentences is that of *markedness*. Within the set of grammatical sentences some are clearly more marked than others. Again, the Extended Categorial Grammar model provides the foundations of a theory of markedness. This is explored in Section 8.3.2.

8.3.1 Alternative Data on Word Order in Transitive Sentences

Green (1988) offers opinions about word order in Spanish which are explicitly contrary to the simple generalisation of Chapter 2. Green asserts (pages 103-4) that Spanish is a VO language, having many of Greenberg's (1966) typological characteristics of VO languages. These include prepositions rather than postpositions (Universal 3), the genitive following the governing noun (Universal 2), interrogative words coming first in interrogative word questions (Universal 12), and inflected auxiliaries preceding main verbs (Universal 16).

In addition, "It is certainly possible to topicalise an object consisting of a definite Noun Phrase or a Proper Noun by moving it to the front of the sentence, but when this happens there is an intonation break after the topic, and an object clitic is obligatorily inserted before the verb: *el coche, lo compró Elena* "(as for) the car, Helen bought it". The result is no longer a simple sentence; *lo compró Elena* is a complete structure in its own right" (Green 1988, page 114).

Green goes on in the same article to argue that Spanish has acquired the reputation of a "comparatively free" word order language largely because of the mobility of the subject rather than because of any other degree of freedom at the sentential level. This is clearly in contradistinction to the data based on informants presented in Chapter 2. It serves to show how variable judgements can be on the matter of word order freedom in languages.

If Spanish is VO, as Green suggests, then the L_p constraint in 8.12 defines the correct orderings.

8.12 $[+\text{Objective}] \supset \neg\backslash[+\text{Objective}, +\text{Primary}]$

This constraint says that if a function takes an argument which is [+Objective] then it cannot take a [+Primary, +Objective] argument from the left. This covers all three classes of verbs which have been the main focus of this study ie. intransitive, transitive and di-transitive.

In addition to the view expressed above, there are speakers of Spanish who accept an ordering which is freer than VO but not so free as that adopted in Chapter 2. These speakers accept fronted direct objects in relatively unmarked contexts just so long as they are indefinite. Again, this can be modelled in Extended Categorial Grammar with the L_p constraint in 8.13.

8.13 $[+\text{Objective}] \supset \neg\backslash[+\text{Objective}, +\text{Primary}, +\text{Definite}]$

The only difference between 8.12 and 8.13 is in the additional specification of the definiteness feature in the latter case.

Notice that the *freer* word order described by the L_p statement in 8.13 does not imply a simpler L_p statement. There are six logically possible orderings of S, V and O. Of two languages L_1, L_2, if L_1 admits three of these orderings and L_2 admits four, including the three of L_1, L_2 might be described as having *freer* word order but the L_p constraint necessary to define the pattern for L_2 might be much more complex than that required for L_1.

This section has shown how L_p constraints can be used to model the varying judgements by Spanish speakers of the limits of intra-linguistic grammaticality. It is naturally inviting to speculate on cross-linguistic variability as well. During the course of this study data were collected for several Romance languages and comparisons were made with regard to the linear ordering constraints operative in those languages. These data were, however, partial and there was little opportunity for verifying the judgements in detail. It has therefore been decided to present the data and analysis in the form of an Appendix rather than as main text. As a result, Appendix 4 is offered as a preliminary examination of the problem of analysing cross-linguistic data within the Extended Categorial Grammar framework.

8.3.2 Word Order and Markedness

The L_p constraints in 8.12 and 8.13 are consistent with the general assumptions and criteria for linear order factored Categorial Grammar laid out in the Introduction: they define the limits of grammaticality for different classes of speakers. However, each set of orderings induced by these L_p statements can be further analysed in terms of the relative markedness each ordering exhibits. Not only that, but whether the claim is that Spanish is VO or the weaker claim that it is not verb final, judgements about markedness remain fairly consistent across different speakers. So, for example SVO is generally agreed to be the least marked order and OVS to be highly marked for those who accept it.

Markedness is a difficult topic but the following comment from Comrie provides a guide to the use of the term in the present context:

> The intuition behind the notion of markedness is Linguistics is that, where we have an opposition between two or more members, ... it is often the case that one member is felt to be more usual, more normal, less specific than the others.
>
> (Comrie 1976, page 111)

In other words deviations from a norm tend to be marked in some way. Word order is clearly a way of marking utterances and deviations from a basic or normal order will be perceived as less usual, more specific and less flexible as far as the contexts in which it may naturally appear are concerned. The interesting question is whether the linear order factored version of Categorial Grammar developed in these chapters can be used to model this intuition.

A good starting point is the work carried out by Uszkoreit within the Generalised Phrase Structure Grammar framework presented in a series of papers deriving from his thesis (Uszkoreit 1984, 1986a, 1986b). Uszkoreit's theory of linear precedence was designed to accommodate ordering constraints which appear to be less of the "all-or-nothing" variety suggested by conventional linear precedence rules and more in the way of a scale of possibilities, a scale which reflects the degree of acceptability of the utterance to native speakers.

Uszkoreit's primary data come from the so-called *Mittelfeld* (middle field) in German. The characteristic of which he describes as follows:

> In main clauses [complements and adjuncts] follow the finite verb, with the possible exception of a single fronted constituent. They can in turn be followed by a nonfinite main verb and nonfinite auxiliary verb. Since the string of complements and adjuncts might be surrounded by other material, it is traditionally referred to as the "middle field" (Mittelfeld) of the clause.
>
> (Uszkoreit 1986b, pages 886-7)

An example of the phenomenon is given in 8.14. Note here that the brackets indicate the *Mittelfeld*, not the constituent structure.

8.14 Dann wird [der Doktor dem Patienten die Pille] geben
NOM DAT ACC

(then will the doctor the patient the pill give)

All six orderings of the noun phrases in 8.14 are grammatical. However, "if we consider the full range of phrases that can fill the roles of subjects and objects, we immediately find numerous ungrammatical sequences" (Uszkoreit 1986b, page 887). So, for example, compare

8.15 Dann wird [es ihm einen Ball] geben
NOM DAT ACC
it him a ball

with

8.16 *Dann wird [einen Ball ihm es] geben
a ball him it

where the only difference lies in the ordering of elements within the middle field. Uszkoreit argues that the standard interpretation of linear precedence statements cannot capture these facts, nor the degree of acceptability judgements found with respect to the various orderings within the middle field.

However, Uszkoreit claims that a number of generalisations fall out from the data given an analytical vocabulary extended to include grammatical and semantic relations, as well as pragmatic roles. The principal generalisations he draws from his study are given in 8.17 (Uszkoreit 1986b, page 889).

8.17 a The agent precedes the theme

b The agent precedes the goal

c The goal precedes the theme

d Focused constituents follow other constituents

e Personal pronouns precede non-pronominal constituents

Without going into the detailed interpretation of notions such as "focus", it is clear that each of these is readily converted into a linear precedence statement. So, 8.18 repeats 8.17 in conventional Generalised Phrase Structure Grammar terms.

8.18	a	TR:AGENT	<	TR:THEME
	b	TR:AGENT	<	TR:GOAL
	c	TR:GOAL	<	TR:THEME
	d	FOCUS:-	<	FOCUS:+
	e	PPRN:+	<	PPRN:-

where TR, FOCUS and PPRN are features for thematic role, focused constituent and personal pronoun respectively.

Now, if these linear precedence statements were simply added to a Generalised Phrase Structure Grammar as a conjunctive list, many of the grammatically acceptable sequences in the middle field would not be induced. There appear to be two facts true about these generalisations: firstly, not all of them hold in every grammatical local structure; secondly, some are stronger constraints than others. So, for example, in the latter case, breaking linear precedence statement 8.18e almost always results in ungrammaticality whereas breaking linear precedence statement 8.18a alone rarely does so.

Uszkoreit's solution is to postulate a new type of linear precedence statement. He treats sets of linear precedence rules such as those in 8.18 as clauses in a complex disjunction. Because it is a disjunction, not every clause has to hold of every given local structure. He does, however, add that, roughly speaking, the more clauses that hold of a local structure, the more likely it is to be grammatical. Further, the clauses are ordered with respect to one another, this order reflecting the fact that some are more constraining than others.

Not surprisingly, many of these ideas carry over to the linear order factored Categorial Grammar presented here. Recall that the basic linear ordering constraint for Spanish transitive sentences is that given in 8.19 below.

8.19 NP[+Objective] $\supset$ /NP

This is a strong constraint on Spanish sentences since it prevents verbal final sequences in transitive sentences, something all informants have been agreed upon. At the other end of the scale could be put the constraint in 8.20.

8.20 \[-Objective]

This simply says that subjects appear to the left of the verb. Clearly, this is a very weak constraint since VOS and VSO orders are frequent in Spanish. However, it does encode the fact that the least marked order is SVO. These two constraints, together with those in 8.12 and 8.13 can be put in an ordered list following Uszkoreit's suggestion. If a penalty is associated with each order, where the higher the number, the more important the constraint, the results are shown in Tables 8.3 and 8.4.

The numbers assigned to the penalties are, of course, merely illustrative. The ordering of the constraints is important, however. It can be seen that the VO orders have the smallest penalties and may therefore be interpreted as the

Penalty	L_p Constraint			Orderings		
				SVO	VOS	VSO
1		\[-Objective]		T	F	F
4	[+Objective]	⊃	¬\[+Objective] [+Primary] [+Definite]	T	T	T
6	[+Objective]	⊃	¬\[+Objective] [+Primary]	T	T	T
10	[+Objective]	⊃	/NP	T	T	T
			TOTAL	0	1	1

Table 8.3
Orderings for Spanish Transitive Sentences

Penalty	L_p Constraint			Orderings		
				OVS [-Def]	OVS [+Def]	SOV/ OSV
1		\[-Objective]		F	F	T
4	[+Objective]	$\supset$	$\neg$\[+Objective] [+Primary] [+Definite]	T	F	T/F
6	[+Objective]	$\supset$	$\neg$\[+Objective] [+Primary]	F	F	F
10	[+Objective]	$\supset$	/NP	T	T	F
			TOTAL	7	11	16/20

Table 8.4
Orderings for Spanish Transitive Sentences

least marked. For those accepting OV orders, the two OVS orders in Table 8.4 are clearly distinguished. The two verb final orders have a penalty which is greater than that assigned to the strongest L_p constraint, which may be interpreted as indicating that these two orderings lie well outside the set of grammatically acceptable sequences.

8.4 Spanish Clitic Pronouns

The syntax and semantics of clitic personal pronouns in the Romance languages is an immensely complex matter. It would be out of the question to review here all the issues that are currently discussed in the literature under the headings of the theory of clitic pronouns and cliticisation processes. It is, however, germane to examine aspects of this topic insofar as they are relevant to the analysis of word order constraints in Spanish.

8.4.1 The Categorial Status of Pronominal Clitics

Cross-linguistically, clitics satisfy the subcategorisation requirements of verbs and are frequently found in complementary distribution to Noun Phrases. Both of these are characteristics which are typically stated in terms of phrasal constituents (see Borer 1986, page 2) and at first sight it seems reasonable, therefore, to take clitics to be Noun Phrases. However, there are difficulties with this view. For one thing, it means that phrasal status is being assigned to items which enter into word formation processes; for another, as Borer (1986, pages 2-3) points out, clitics have no internal structure or other NP-like properties: they cannot take specifiers, modifiers or complements. Of course, they have these characteristics in common with non-clitic pronouns and proper names but, in addition, clitics cannot be conjoined whereas non-clitic pronouns and proper names can be.

These arguments weigh heavily against the claim that pronominal clitics are Noun Phrases. An alternative view available in Categorial Grammar, and the one adopted here, is to treat clitics as functions over verbal functions which return a verbal function identical to their argument except for certain systematic changes in the valency and linear ordering properties of the argument. Through the mechanisms of re-entrant structure and operations over Return values, Extended Categorial Grammar is able to model this behaviour in interesting ways. The essential point about this analysis is that *all* pronominal clitics are categorially related (ie. they are functions over verbal functions) and the idiosyncratic behaviour distinguishing dative from accusative clitics, for example, is located entirely within the operations over the Return values. Figure 8.4 shows the category assigned to accusative clitics in Extended Categorial Grammar; Figure 8.5, the category for dative clitics. Both these category representations are provisional at this stage. The simple L_p constraint assigned in each case is not adequate. The details of this are discussed below and a final version of each category with the full L_p constraint is given later.

Given this category structure for clitics, it is possible to offer an Extended Categorial Grammar account of, for example, a transitive sentence containing a

$$
\left[\begin{array}{ll}
\text{Return:} & \text{ACC}(\ \boxed{1}\ ) \\[2ex]
\text{ArgSet:} & \left\{ \left[\begin{array}{ll} \text{Category:} & \left[\begin{array}{ll} \text{Return:} & \text{S} \\ \text{ArgSet:} & \cup\ \text{W},\ \ \underset{\substack{[+\text{Primary}] \\ [+\text{Objective}]}}{\text{NP}} \end{array} \right]^{\boxed{1}} \end{array} \right] \right\} \\[2ex]
\text{Lp:} & /\ \boxed{1}
\end{array}\right]
$$

Figure 8.4
The Category for the Accusative Pronominal Clitic:
Preliminary Version

pronominal clitic direct object, such as that in 8.21 below.

$$
\left[\begin{array}{ll}
\text{Return:} & \text{DAT}(\ \boxed{1}\ ) \\[2ex]
\text{ArgSet:} & \left\{ \left[\begin{array}{ll} \text{Category:} & \left[\begin{array}{ll} \text{Return:} & \text{S} \\ \text{ArgSet:} & \cup\ \text{W},\ \ \underset{\substack{[-\text{Primary}] \\ [+\text{Objective}]}}{\text{NP}} \end{array} \right]^{\boxed{1}} \end{array} \right] \right\} \\[2ex]
\text{Lp:} & /\ \boxed{1}
\end{array}\right]
$$

Figure 8.5
The Category for the Dative Pronominal Clitic:
Preliminary Version

8.21 Juan lo compró
(John bought it)

The Rightward Looking version of the Extended Categorial Grammar combinator schema in Chapter 7, repeated here as Figure 8.6, will allow the combination of the clitic and verb in 8.21.

The category for the clitic is that given in Figure 8.4 and the category for *compró* is given in 8.22. It is assumed that the inflectional affix adds an OPT value to the subject NP and relevant features to the Return value of the category.

$$
\begin{bmatrix}
\text{Return:} & X \\
\text{ArgSet:} & \cup \begin{bmatrix} \text{Return:} & Y \\ \text{ArgSet:} & W \\ \text{Lp:} & e(\alpha) \end{bmatrix}, \alpha \\
\text{Lp:} & e(\delta)
\end{bmatrix}
+
\begin{bmatrix}
\text{Return:} & Y \\
\text{ArgSet:} & \cup \; W, \beta \\
\text{Lp:} & e(\gamma)
\end{bmatrix}
=>
$$

$$
\begin{bmatrix}
\text{Return:} & X \\
\text{ArgSet:} & \cup \begin{bmatrix} \text{Category:} & \begin{bmatrix} \text{Return:} & Y \\ \text{ArgSet:} & W \\ \text{Lp:} & e(\alpha) \end{bmatrix} \\ \text{DIR} & / \end{bmatrix}, \alpha, \beta \\
\text{Lp:} & e(\delta) \;\&\; e(\gamma)
\end{bmatrix}
$$

Figure 8.6
The Extended Categorial Grammar Combinator Schema:
Rightward Looking Version

8.22

$$
\left[\begin{array}{ll}
\text{Return:} & \left[\begin{array}{ll}\text{Return:} & \left[\begin{array}{ll}\text{N:} & - \\ \text{V:} & + \\ \text{Phrasal:} & + \\ \text{Pers:} & 3 \\ \text{Num:} & \text{sg} \\ \text{Tense:} & \text{pret}\end{array}\right]\end{array}\right] \\[2ex]
\text{ArgSet:} & \left\{ \left[\begin{array}{ll}\text{Category:} & \begin{array}{c}\text{NP} \\ [+\text{Primary}] \\ [-\text{Objective}]\end{array} \\ \text{OPT:} & \text{T}\end{array}\right], \left[\begin{array}{ll}\text{Category:} & \begin{array}{c}\text{NP} \\ [+\text{Primary}] \\ [+\text{Objective}]\end{array}\end{array}\right] \right\} \\[2ex]
\text{Lp:} & e(\gamma)
\end{array}\right]
$$

The settings for the variables in the combinator schema are given in 8.23.

8.23
$$
\begin{array}{lcl}
X & = & \text{ACC}(1) \\
Y & = & S \\
W & = & \{\text{NP}[+\text{Primary}, +\text{Objective}]\} \\
\alpha & = & \emptyset \\
\beta & = & \emptyset
\end{array}
$$

The index in ACC(1) is used here as an abbreviation for the structure the number indicates.

The result of the combination is given in 8.24 below.

8.24

$$
\left[\begin{array}{ll}
\text{Return:} & \text{ACC}(\boxed{1}) \\
\text{ArgSet:} & \left\{ \left[\begin{array}{ll}
\text{Category:} & \left[\begin{array}{ll}
\text{Return:} & \left[\begin{array}{ll}
\text{Return:} & \left[\begin{array}{ll}
\text{N:} & - \\
\text{V:} & + \\
\text{Phrasal:} & + \\
\text{Pers:} & 3 \\
\text{Num:} & \text{sg} \\
\text{Tense:} & \text{pret}
\end{array}\right]
\end{array}\right] \\
\text{ArgSet:} & \left\{ \left[\begin{array}{ll}
\text{Cat} & \left[\begin{array}{c} \text{NP} \\ [+\text{Primary}] \\ [-\text{Objective}] \end{array}\right] \\
\text{OPT:} & \text{T}
\end{array}\right], \begin{array}{c} \text{NP} \\ [+\text{Primary}] \\ [+\text{Objective}] \end{array} \right\} \\
\text{Lp:} & e(\gamma)
\end{array}\right] \boxed{1} \\
\text{DIR:} & /
\end{array}\right] \right\} \\
\text{Lp:} & / \ \boxed{1}
\end{array}\right]
$$

Recall that whenever *T-Eval* of an Extended Categorial Grammar category evaluates to T, as it does for the category in 8.24, the Return value of the function is also made available as a category. In this case the Return value is subject to the operation ACC. The details of this operation are discussed below but for the present it can be assumed that the category returned is that given in 8.25.

8.25

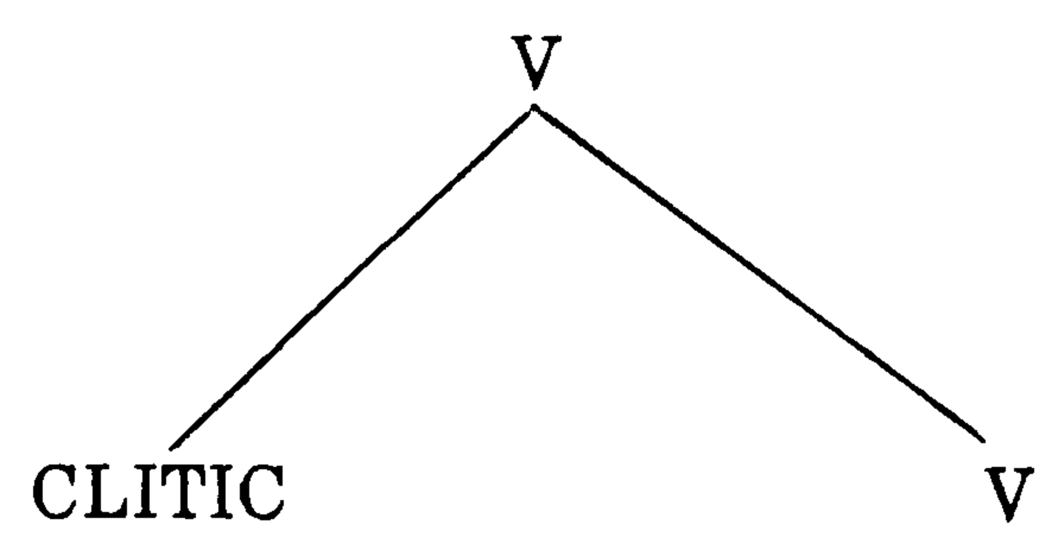
$$
\left[\begin{array}{ll}
\text{Return:} & \left[\begin{array}{ll}
\text{Return:} & \left[\begin{array}{ll}
\text{N:} & - \\
\text{V:} & + \\
\text{Phrasal:} & + \\
\text{Pers:} & 3 \\
\text{Num:} & \text{sg} \\
\text{Tense:} & \text{pret}
\end{array}\right]
\end{array}\right] \\
\text{ArgSet:} & \left\{ \left[\begin{array}{ll}
\text{Cat} & \left[\begin{array}{c} \text{NP} \\ [+\text{Primary}] \\ [-\text{Objective}] \end{array}\right] \\
\text{OPT:} & \text{T}
\end{array}\right], \left[\begin{array}{ll}
\text{Cat} & \left[\begin{array}{c} \text{NP} \\ [+\text{Primary}] \\ [+\text{Objective}] \end{array}\right] \\
\text{OPT:} & \text{T}
\end{array}\right] \right\} \\
\text{Lp:} & e(\gamma)
\end{array}\right]
$$

This category has an optional marking for the direct object NP indicating that it may combine with a lexical NP but does not need to do so for the sentence of which it is a constituent to be regarded as fully saturated.

This simple example indicates the general approach adopted in Extended Categorial Grammar for the analysis of the combination of clitics with their *hosts* (see Zwicky (1977) for this term). Two questions immediately arise with respect to this proposal. Firstly, what is the categorial status of the host? Secondly, how can it be ensured that the clitic combines *immediately* with its host to the right with no intervening lexical material? As indicated below, the answers to these two questions are interconnected.

Clitics are commonly assumed to take *lexical* categories as hosts. In the Phrase Structure Grammar literature this has led to the proposal that clitics occur in structures such as the following (Borer 1986, page 3):

8.26



Since the clitic is dominated by a (pre)-terminal or lexical category, this tree can be regarded as specifying word internal structure. However, the rule which induces this local tree breaks standard X-Bar conventions since neither daughter has a bar level lower than the mother and there seems to be little motivation for a special class of rules of this kind apart from the fact that they are required to introduce clitics. It remains, therefore, a statement of an exception, rather than a generalisation.

In Categorial Grammar there is no clear distinction between terminal and non-terminal categories. Consequently, if clitics take *only* categories which have accepted no other arguments as hosts, this might prove difficult to model. The Spanish data clearly show, however, that clitics can combine with verbal functions which have already accepted arguments. Consider the sentences in 8.27 below.

8.27 a Juan la canta y toca
(John sings it and plays it)

b la canta Juan y toca María
(John sings it and Mary plays it)

In both 8.27a and 8.27b the second verb (*tocar* in each case) may be interpreted transitively, so that the song John plays in 8.27a is the same as he sings and the song Mary plays in 8.27b is the same as John sings. In 8.27a, the verbs are conjoined. It would be difficult to see how to represent the semantics of the transitive reading if the clitic combined with *cantar* before the conjunction operation. 8.27b is even more interesting: both verbs must combine with their subjects and then conjoin before combining with the clitic in order to secure the transitive reading. The category of the clitic host is, therefore, *any* verbal category which carries an argument relevant to the clitic. In the categories given in Figures 8.4 and 8.5 the variable W ranges over all arguments in the host's ArgSet except the direct and indirect objects respectively. The result of this is that the clitic will combine with verbs which have already accepted some other arguments, such as subjects.

The second question raised above concerned the fact that clitics attach themselves *immediately* to a verbal host. This is difficult but the Extended Categorial Grammar model does offer a solution. Consider the data in 8.28 below.

8.28 a Juan la canta
(John sings it)

b la canta Juan

c *la Juan canta

Nothing in the above account of the categorial status of the clitic host predicts that 8.28c is ungrammatical. Indeed, the contrary is the case since the category associated with the sequence *Juan canta* is identical to that associated with *canta Juan* except for the value of the DIR attribute. Consider the clitic category in Figure 8.4 again. This must combine with a verbal category with no intervening lexical material although the host may well have accepted arguments already, as the examples in 8.27 indicate. In other words, if the host has already accepted arguments, they must have been taken from the right before it combines with the clitic. To enforce this constraint requires that the category in the ArgSet of the clitic in Figure 8.4 can only unify with a host if there is no leftward instantiated DIR value in the host's ArgSet. The L_p constraint of the clitic category, as it is given in Figure 8.4 at least, refers only to the direction of the host category in its entirety, not to the host's ArgSet. There is, however, a mechanism available within Extended Categorial Grammar for imposing constraints of this kind. Recall from Chapter 3 the basic L_p expression of the form $\Box\alpha$. This was taken to mean that α returns T only if it is true of every category *within* the category in which the expression is given. Recall also that Extended Categorial Grammar categories combine only if *T-Eval* of the function returns a non-false value *after* the putative argument has unified with one of the categories in the function's ArgSet. This unification is important since the resulting category may contain new information drawn from either the function or argument or both.

The result of these observations is that it is possible to define the L_p constraints of the clitic in such a way that they constrain the arguments of the putative host which has unified with the clitic. The revised category structures for accusative and dative clitics are given in Figures 8.7 and 8.8. The first conjunct of the L_p statement is identical to the earlier version; the second conjunct ensures that no Noun Phrase category *within* the ArgSet of the clitic's argument (ie. the host verb) can have been found from the left.

$$\begin{bmatrix} \text{Return:} & \text{ACC}(\boxed{1}) \\ \text{ArgSet:} & \left\{ \begin{bmatrix} \text{Category:} & \begin{bmatrix} \text{Return:} & S \\ \text{ArgSet:} & \cup\ W,\ NP_{\substack{[+\text{Primary}]\\ [+\text{Objective}]}} \end{bmatrix}^{\boxed{1}} \end{bmatrix} \right\} \\ \text{Lp:} & /\ \boxed{1}\ \&\ \Box \neg\backslash NP \end{bmatrix}$$

Figure 8.7
The Category for the Accusative Pronominal Clitic:
Revised Version

$$\begin{bmatrix} \text{Return:} & \text{DAT}(\boxed{1}) \\ \text{ArgSet:} & \left\{ \begin{bmatrix} \text{Category:} & \begin{bmatrix} \text{Return:} & S \\ \text{ArgSet:} & \cup\ W,\ NP_{\substack{[-\text{Primary}]\\ [+\text{Objective}]}} \end{bmatrix}^{\boxed{1}} \end{bmatrix} \right\} \\ \text{Lp:} & /\ \boxed{1}\ \&\ \Box \neg\backslash NP \end{bmatrix}$$

Figure 8.8
The Category for the Dative Pronominal Clitic:
Revised Version

8.4.2 Clitic Doubling and the Return Operations

The previous Section dealt with problems relating to the categorial status of the clitic host and the L_p statements required by the clitic to ensure that it combines directly with its verbal host without intervening lexical material. There is one remaining aspect of the category structure given in Figures 8.7 and 8.8 which must be considered: the operations ACC and DAT over the Return values of the functions. Consideration of these operations must take into account the phenomenon of clitic doubling which is so common in Spanish (see Jaeggli (1986) and Beaven (1989) for recent discussions of this phenomenon). Before discussing this, however, it is important to observe that there can be no question that pronominal clitics do satisfy the valency requirements of verbs exactly like Noun Phrases, so that the sentence in 8.29, for example, is a fully grammatical sentence.

8.29 Juan la compró
(John bought it)

Notice further that 8.29 is verb final, apparently in contravention to the L_p constraints for transitive sentences discussed in earlier chapters.

Since the subject in 8.29 can also come to the right of the verb, it might be thought that the operation of the accusative clitic (ACC in Figure 8.7) could simply involve the removal of the direct object argument from the ArgSet of the verb. Recall that the basic L_p statement for intransitive, transitive and di-transitive verbs accepted in Chapter 5 is that given in 8.30a, which is now better represented as 8.30b.

8.30 a NPobj $\supset$ /NP

b NP[+Objective] $\supset$ /NP

With the removal of an NP[+Primary, +Objective] argument from its ArgSet, the accusative clitic/transitive verb combination becomes, in effect, an intransitive verb. In the case of di-transitive verbs, the clitic/verb combination is one which will allow further combination with a subject and indirect object, one of which must occur to the right of the verb. This captures the data judgements in 8.31 below.

8.31 a Juan le dio un libro
(John gave him/her a book)

b *Un libro Juan le dio

c *Juan un libro le dio

The fact of clitic doubling complicates this picture and makes this simple version of ACC untenable. It is commonly argued that the accusative clitic does not double with a Noun Phrase equivalent. This is not strictly correct. The data in 8.32 shows the possibilities.

8.32 a *La compró una casa

b Una casa la compró
(s/he bought a house)

The direct object can co-occur with the accusative clitic but only if accepted from the left. Notice that 8.32b is not a dislocated construction since there is only a single intonation contour. The ACC operation is therefore required to do two things: to mark the direct object as optional (rather than to remove it), and to change the L_p constraints so that the verb accepts its direct object (if any) from the left only. 8.33 spells out this operation in Extended Categorial Grammar terms.

8.33 ACC:

1 ArgSet: NP[+Primary, +Objective]

$\Rightarrow$ NP[+Primary, +Objective]
[OPT: T]

2 L_p: NP[+Objective] $\supset$ /NP

$\Rightarrow$ NP[+Objective] $\supset$ \NP[+Primary, +Objective]

The first operation in 8.33 changes the direct object category in ArgSet, placing the attribute-value pair [OPT: T] in the argset category. The OPT feature was introduced in Chapter 7 but not discussed at that point. It is, however, important for the evaluation of basic L_p expressions. Essentially, it guarantees that a basic expression which matches with an argset category will be evaluated to T if that argset category is marked as optional, whether or not it has accepted an argument. The introduction of this feature is more than just a mechanism for marking the optionality of arguments. It has major consequences for the definition of sentencehood in Categorial Grammar. In a conventional Categorial Grammar, a sentence is a sequence of categories which reduces to S, leaving no argument in any function in the sequence unsatisfied. In other words, sentencehood is defined in terms of valency satisfaction. Clearly, however, in a language where there are optional components to sentences, the definition of sentencehood and the notion of valency satisfaction need to be treated separately. The proposal adopted here regards sentencehood in Spanish not solely as a syntactic concept but also as a pragmatic one. OPT is, in effect, a feature marking this pragmatic aspect of functions.

The second component of the operation in 8.33 is the change to the L_p expression. This ensures that if a direct object is accepted, it must come from the left. It also allows for the fact that transitive sentences with clitic objects appear to be verb final, since the L_p constraint applies only to the direct object and leaves the ordering of the subject free.

The dative clitic is rather different to the accusative. The phrasal indirect object freely co-occurs with the dative clitic, which places no ordering constraints on it. Consequently, DAT is given by 8.34 below.

8.34 DAT:

ArgSet: NP[-Primary] ⇒ NP[-Primary]
[OPT: T]

It is easy to see how these two clitic operations will feed each other in the case of a sentence like that in 8.35.

8.35 Juan se lo dio
[DAT] [ACC]
(John gave it to him/her)

ACC ensures that the direct object argument of *dar* is marked as optional but, if present, found only to the left. DAT will then take the output of the combination of the accusative clitic and verb to mark its indirect object as optional. Hence admitting the sequence in 8.35 as a sentence.

8.4.3 Clitic Climbing

As the previous discussion has made clear, clitics normally occur in close proximity to the verb they are semantically related to. There is, however, a particular type of clitic behaviour in Spanish that has long been discussed and which is commonly referred to as *clitic climbing* (see Fish (1961), Roldán (1972), Monzon (1979), Luján (1980), Bok-Bennema (1982) and Beaven (1989)). This behaviour involves the clitic occurring to the left of its semantic host attached to a verb which does not necessarily take a Noun Phrase argument. This is very difficult to model in most current syntactic frameworks and Extended Categorial Grammar is no exception. The nature of the problem is illustrated by the data in 8.36.

8.36 a quiero comprarlo
(I want to buy it)

b lo quiero comprar

c lo quiero poder comprar
(I want to be able to buy it)

In 8.36b the clitic, which satisfies an argument requirement of *comprar*, is found apparently attached to the "higher" control verb. 8.36c shows that the clitic can be separated by an arbitrary number of verbs from its semantic host, although sequences of more than three are extremely difficult to construct for semantic reasons.

The first step towards addressing this problem in Extended Categorial Grammar is to observe the position of the clitic relative to its host. In 8.36a it is enclitic, whereas in a sentence like 8.37 it is proclitic.

8.37 Juan lo compró
(John bought it)

The orderings in 8.38 below are absolutely ungrammatical.

8.38 a *quiero lo comprar

b *Juan compró lo

In general, pronominal clitics are enclitic with present participles and infinitival (basal) forms of the semantic host but proclitic with past participles and finite forms of the host (Ramsey 1956, pages 74 and 94-95). This pattern must simply be taken as a raw fact needing to be explicitly stipulated in the L_p constraints for clitics. Consequently, the full L_p constraint for clitics in Extended Categorial Grammar is given in 8.39.

8.39 $([\text{Return: } [+\text{PRP}]] \ \vee \ [\text{Return: } [+\text{BASE}]]) \supset (\backslash 1 \ \& \ \Box\neg\backslash\text{NP})$

V

$([\text{Return: } [+\text{PSP}]] \ \vee \ [\text{Return: } [+\text{FIN}]]) \supset (/1 \ \& \ \Box\neg\backslash\text{NP})$

Here standard abbreviations are used for Present Participle (PRP), Past Participle (PSP), the base form of the verb used as the infinitive in Spanish (BASE), and finite (FIN).

Given this characteristic of clitic ordering relative to their hosts, some of the clitic climbing facts fall out without the need for further principles. Thus, the enclitic form in 8.36a is in accordance with the constraint in 8.39 and the form in 8.38a is predicted to be ungrammatical. Likewise, the proclitic in 8.37 is predicted to be grammatical, the sentence in 8.38b to be ungrammatical.

There are further complications, however. Data from coordination shows that the predictions based on stipulating the proclitic and enclitic behaviour relative to morphosyntactic characteristics of the host are not descriptively adequate.

Consider the sentences in 8.40 and 8.41 below, where the bracketing indicates the first two categories to combine.

8.40 a quiero [comprar lo]
(I want to buy it)

b *[quiero comprar] lo

8.41 a lo [quiero comprar]
(I want to buy it)

b [lo quiero] comprar

The sentence in 8.40a is straightforward. The clitic combines with its semantic host and the result becomes the argument for *querer*. This means that the coordination in 8.42 is correctly predicted to be grammatical.

8.42 quiero [[comprarlo] y después [venderlo]]
(I want to buy it then sell it)

The combination in 8.40b is correctly predicted to be ungrammatical. It was established earlier that clitics can combine with verbs which have already accepted some of their arguments. *Querer* takes a basal form of the verb (here, *comprar*) and returns a finite verb looking for the arguments *comprar* requires. The ungrammaticality of the sentence in 8.40b is, therefore, not due to the fact that the two verbs combine first, rather it is due to the fact that the category resulting from the combination is finite and therefore must take its clitic from the left. This is illustrated in 8.41a where the verbs combine and form a suitable argument for the clitic. The availability of the bracketing in 8.41a is shown by the acceptability of 8.43 below.

8.43 lo [[quiero comprar] y [quiero vender]]
(I want to buy (it) and sell it)

Here both verbs, *comprar* and *vender*, can have transitive readings. It seems therefore that at least one version of the clitic climbing in 8.36b and 8.36c can be explained by assuming that the verbal categories combine first and the position of the clitic is determined by the constraint in 8.39.

The problem sentence is 8.41b. This is clearly an acceptable order of combination, as shown by the grammaticality of the sentence in 8.44.

8.44 [[lo quiero] y [lo debo]] comprar
(I want to and must buy it)

This sentence shows quite clearly that the clitic, when it has climbed over a "higher" control verb, can combine with that verb *before* the result of that operation combines with the clitic's semantic host. This is a serious problem.

It is tempting to suggest that this behaviour finds its proper explanation at the level of parsing and linguistic processing. So, for example, assuming a left-to-right incremental parsing of sentences, the dislocated appearance of the clitic for which there is no immediate host might result in the clitic being "stacked" until a suitable host becomes available. An account along these lines would, however, have to offer a quite different approach to coordination to the one assumed throughout this thesis, since the stacked clitic and the control verb are available for coordination even though they have not formed a single category.

A more syntactically oriented explanation is tentatively offered here, though one which requires much further research. Recall that control verbs belong to the class of metacategories defined in Chapter 4 and defined in Extended Categorial Grammar terms in 8.45 below.

$$8.45 \quad C = \begin{bmatrix} \text{Return:} & X \\ \text{ArgSet:} & \{\ldots, K, \ldots\} \\ \text{Lp:} & e(\gamma) \end{bmatrix}$$

where, K is a complex category and
Return value of K = Return value C
ArgSet of K = ArgSet of C - K

Specifically, the category for *querer* is that given in 8.46.

8.46

$$\begin{bmatrix} \text{Return:} & S \\ \text{ArgSet:} & \left\{ \begin{matrix} NP \\ [+\text{Primary}] \\ [-\text{Objective}] \end{matrix} \;,\; \begin{bmatrix} \text{Return:} & S \\ \text{ArgSet:} & \left\{ \begin{matrix} NP \\ [+\text{Primary}] \\ [-\text{Objective}] \end{matrix} \;,\; W \right\} \\ \text{Lp:} & e(\delta) \end{bmatrix} \;,\; W \right\} \\ \text{Lp:} & e(\gamma) \end{bmatrix}$$

Two new assumptions need to be made. Firstly, that the variable W in 8.46 may be distributed over any number of potential arguments by division of W into $w_1, w_2, \ldots w_n$, in which case the category in 8.47 is equivalent to the category in 8.46.

8.47

$$\begin{bmatrix} \text{Return:} & S \\ \text{ArgSet:} & \left\{ \begin{array}{c} \underset{\text{[-Objective]}}{\overset{\text{NP}}{\text{[+Primary]}}} \end{array} , \begin{bmatrix} \text{Return:} & S \\ \text{ArgSet:} & \left\{ \underset{\text{[-Objective]}}{\overset{\text{NP}}{\text{[+Primary]}}} , w_1 , w_2 \right\} \\ \text{Lp:} & e(\delta) \end{bmatrix} , w_1 , w_2 \right\} \\ \text{Lp:} & e(\gamma) \end{bmatrix}$$

Secondly, β in the combinator schema is taken to range over sequences of categories, not a single category.

The result of adopting these two assumptions means that the sequence *lo quiero* is parsed as a single category with the following settings for the combinator schema:

8.48 $X = S$

$Y = S$

$W =$

$$\begin{bmatrix} \text{Return:} & S \\ \text{ArgSet:} & \left\langle \begin{matrix} \text{NP} \\ [+\text{Primary}] \\ [-\text{Objective}] \end{matrix} , \begin{matrix} \text{NP} \\ [+\text{Primary}] \\ [+\text{Objective}] \end{matrix} , w_1 \right\rangle \\ \text{Lp:} & e(\delta) \end{bmatrix}$$

$\alpha = \varnothing$

$\beta =$

$$\left\langle \begin{matrix} \text{NP} \\ [+\text{Primary}] \\ [-\text{Objective}] \end{matrix} , \begin{matrix} \text{NP} \\ [+\text{Primary}] \\ [+\text{Objective}] \end{matrix} , w_1 \right\rangle$$

The output category is shown in 8.49.

8.49

$$\begin{bmatrix} \text{Return:} & S \\ \text{ArgSet:} & \left\langle \begin{matrix} \text{NP} \\ [+\text{Primary}] \\ [-\text{Objective}] \end{matrix} , \begin{bmatrix} \text{Cat:} & \begin{bmatrix} \text{Return:} & S \\ \text{ArgSet:} & \left\langle \begin{matrix} \text{NP} \\ [+\text{Primary}] \\ [-\text{Objective}] \end{matrix} , \begin{matrix} \text{NP} \\ [+\text{Primary}] \\ [+\text{Objective}] \end{matrix} , w_1 \right\rangle \\ \text{Lp:} & e(\delta) \end{bmatrix} \\ \text{DIR} & / \end{bmatrix} , \begin{matrix} \text{NP} \\ [+\text{Primary}] \\ [+\text{Objective}] \end{matrix} , w_1 \right\rangle \\ \text{Lp:} & e(\gamma) \end{bmatrix}$$

This analysis allows for the combination of a pronominal clitic with its host which does not take a Noun Phrase argument in its own right but which can accept a category following combination with the clitic which does take Noun Phrase arguments.

8.5 Conclusion

The aim of this chapter was to apply the Extended Categorial Grammar model to a variety of natural language problems involving complex semi-free word order. The first of these involved a detailed examination of the structure of Spanish transitive sentences, paying particular attention to the roles of the determiners and the particle *a*. This led into a discussion of alternative data on word order which had been encountered among informants and in the literature with respect to the ordering of the direct object and finite verb. It was shown how the linear ordering logic of Extended Categorial Grammar can model these different views of the limits of grammaticality in Spanish.

An interesting approach to the problem of defining variable word ordering has been developed in the Generalised Phrase Structure Grammar framework by Uszkoreit. This was briefly discussed and a method of implementing these insights in a linear order factored Categorial Grammar was suggested.

Finally, an analysis was offered of Spanish clitic pronouns with respect to their effects on word order. The well-known phenomena of clitic doubling and clitic climbing were addressed from this point of view. It was shown how the Extended Categorial Grammar model provides a powerful tool for the exploration of these issues.

CHAPTER 9

THE CATEGORIAL PERSPECTIVE ON WORD ORDER

The work reported in this thesis began with the apparently straightforward aim of factoring out linear ordering constraints from other syntactic information in a Categorial Grammar on the model of the Immediate Dominance/Linear Precedence factoring in Generalised Phrase Structure Grammar. The first step in this programme involved demonstrating the existence of a common formal ground between Phrase Structure rules and Categorial Grammar categories. The common structure which emerged - referred to as a local structure - was then used to develop a model of Categorial Grammar in which explicit word ordering statements could be expressed independently of other structural aspects of the calculus.

It had already become obvious that the full range of word orders found in Spanish transitive sentences could not be uniformly analysed in an Immediate Dominance/Linear Precedence format grammar. The same difficulties naturally carried over to a linear order factored Categorial Grammar. It was therefore necessary to introduce a full propositional logic for the description of semi-free word order. The important point is that this extension to the formal power of the grammar was motivated entirely by the data: the facts of semi-free word order force the introduction of such a logic. This generalisation in the power of the linear ordering component of the grammar, together with a more principled account of the basic linear ordering expressions, made it possible to account for the variations in word order in Spanish. The details of this were presented in Part Two.

While the motivation for these generalisations of the formalism were fundamentally empirical, it became clear during the investigation of this model that a linear order factored grammar using boolean combinations of ordering constraints addressed in a rather direct way the problem of defining grammars in the categorial space between L and **LP**, something that has been the concern those approaching Categorial Grammar from a more logical and mathematical point of view. The weakest grammars admitted by the model presented in Part Two are, in fact, slightly weaker than **L** (strictly they are product-free versions of **L** without Associativity); the strongest grammars are **LP**. The linear ordering constraints can be used to define explicitly the limits of grammaticality for languages between these extremes. This approach has its drawbacks as well as its advantages. The formalism of the model presented in Part Two places no inherent restrictions on linear ordering in this categorial space. All restrictions must be explicitly given in the L_p expressions associated with functions. Different attitudes can be taken towards this. On the one hand, the formalism clearly does not carve out a subset of Categorial Grammars weaker than **LP** corresponding to the class of natural languages. On the other hand, from an empirical point of view, it is not at all obvious that it is possible, given current knowledge, to define such a subset. Simply not enough is yet known about the limits of word order freedom in languages. It may be very unlikely that there exist languages requiring full permutation closure over their stringsets, but the limits of scrambling have not yet been empirically determined. Consequently, it seems premature to require of a grammatical model that it carve out some *a priori* determined subset of languages between **L** and **LP**.

Part Three extended the formalism presented in Part Two by the use of attribute-value matrices to represent category structure, the introduction of unification as part of a general category matching operation, and the elimination of the expanded level of ordered local structure. The motivation for this last innovation was originally implementational. There are well-known problems with the expansion of partially specified Immediate Dominance statements to fully instantiated Phrase Structure rules; a very similar problem arises with the model in Part Two. The consequences of making this move proved to be unexpectedly interesting. It provoked the introduction of a three-valued logic for the interpretation of the L_p constraints. Furthermore, new opportunities arose for categorial interaction following the adoption of the attribute-value matrix formalism with its operations over the Return values of functions. It is obvious, however, that the extended model as it it presented in Chapter 7 has serious limitations. Although it was originally intended as a preliminary to an implementation of a linear order factored Categorial Grammar, it is clear that certain aspects of the model could prove very difficult to implement efficiently. In particular, the shift in computational burden from the precompilation of ordered lexical categories to run-time evaluation of L_p constraints is likely to be problematic. There is also an unresolved question about the types of combinators required in the new model. Given the use of variables in categories, it is not at all clear that the full range of functional combinators is required. Much closer examination of the relevant data, particularly that which has been used to motivate Composition and Substitution, would be required before the status of the Combinator Schema as it is presented in Chapter 7 could be finalised.

One further aspect of the presentation of Extended Categorial Grammar in Chapter 7 also requires comment. The relation between the declarative and procedural aspects of the grammar is unclear as it stands. On the one hand, procedural definitions are given for category matching and the evaluation of basic L_p expressions; on the other hand, the foundations of the linear ordering component of the grammar are given as a logic which is inherently declarative. This inadequacy in the presentation would be corrected by a stricter separation between a formal, declarative definition of Extended Categorial Grammar in line with the Categorial Grammars given in Chapter 1 and independent descriptions of the algorithms for implementation of the grammar.

From this exploration of categorial models suitable for the expression of complex linear ordering facts there gradually emerged what might be called the categorial perspective on word order: an approach to the problems of describing semi-free word order which gives substance and form to the lexicalist claims of Categorial Grammar made at the beginning of Chapter 1. The essential ingredient in this perspective is the word: more than a Saussurean sign, rather a matrix of syntactic, semantic and pragmatic properties which fan outwards left and right influencing the behaviour and properties of other words and being influenced by them. The conflicting restrictions and constraints that words impose upon their neighbours are resolved in the combinatorial processes of the grammar until ultimately, at the sentence level (perhaps more accurately, the discourse level) the conflicts are resolved. This is what is meant here by the categorial perspective. Syntactic properties, especially those relating to word order, project out from the lexicon up the function-argument hierarchy changing and being changed in a systematic manner. This is quite a different perspective on linear ordering from that encouraged by models where the ordering constraints are defined over structural configurations; it has more in common with the perspective encouraged by dependency grammar with the constraint that only adjacent categories may cancel in this version of Categorial Grammar.

Two clear examples of the propagation of linear ordering properties up the function-argument hierarchy have been presented here. Auxiliary verbs combine with a verbal function, returning a verbal function with the same linear ordering characteristics as the argument. From the point of view of the ordering restrictions across the sentence, auxiliaries are invisible. This does not *have* to be the case. Recall that the meta-category definition of auxiliaries, as it was given in Chapter 5, allowed them three arguments: a subject, the verbal function argument and a variable W. The auxiliary therefore has its own L_p constraints governing the behaviour of these arguments. In the examples studied, from English and Spanish, the L_p constraint required that the auxiliary combine first with its verbal function argument from the right. It may be that under different circumstances, this restriction should not hold. There is the obvious issue of the categorial analysis of yes-no questions in both English and Spanish; there is the possibility that in other languages this restriction simply does not operate. In other words, the auxiliary may, in certain circumstances, influence the linear ordering characteristics of a sentence through its own L_p constraints. Clearly, this is an area which is wide open for further investigation but the categorial perspective does offer an interesting way of looking at the interaction of word order constraints projected from the lexicon.

A second and contrasting example of the propagation of linear ordering information up the function-argument hierarchy involves Spanish clitic pronouns. Clitics are *not* transparent transmitters of the linear ordering behaviour of their verbal hosts. The accusative clitic, for example, systematically changes the linear ordering properties of the verb so that it must collect its direct object (if at all) from the left rather than the right. Again, as with auxiliaries, much more work needs to be done on the range of functions which can be made available to the grammar to change the behaviour of other functions. The passive auxiliary in English may be of the same type of category as clitics: it takes as an argument a verbal function with an L_p constraint to the effect that the subject comes from the left and (for simple transitives at least) the direct object from the right. It returns a verbal function looking for its subject from the right, which is marked as optional and as, say, NP[+by]. The object is not optional and is required from the left. The agentive *by* is itself a function returning a suitably feature-instantiated Noun Phrase. Furthermore, it would be possible to avoid the redundant repetition of information on verbs to the effect that they require a Noun Phrase to their left by the use of a meta-level statement over L_p constraints.

Finally, what conclusions may be drawn from this study and what directions might future research take?

The first conclusion is the simplest. It has been demonstrated in this thesis that it is possible to provide a way of factoring out explicitly stated constraints on the ordering of arguments relative to functions in the formal definition of a Categorial Grammar. A considerable proportion of the thesis was given over to the exposition and illustration of this idea. A full propositional logic was chosen to express the linear ordering constraints although issues relating to the power of the logic required for the description of natural languages and the possibility of introducing further meta-level generalisations over the linear ordering statements themselves were left for further investigation. It is a fundamental claim of the approach adopted here that this factoring out of linear ordering information from other aspects of category structure provides one way of addressing the important issue of defining linguistically sensitive grammars in the categorial space between **L** and **LP**.

A second conclusion to be drawn from this study concerns the analysis of semi-free word order data. The version of Categorial Grammar developed here allows for the concise description of the complex word ordering restrictions found in semi-free word order languages. Examples of the kinds of statements that can be made covered major constituent ordering in Spanish transitive sentences, the constrasting behaviour of the two auxiliary verbs *haber* and *estar*, and the effects on ordering of accusative clitic pronouns. Furthermore, the linear ordering logic used to express these complex ordering restrictions allows cross-linguistic comparisons to be made, as was indicated in the brief study of the Southern Romance languages in Appendix 4. The actual analyses offered in the thesis remain programmatic; there is no intention here to claim, for example, that the last word has been said on the nature of the constraints influencing the ordering of subjects and direct objects in simple transitive sentences in Spanish. Far from it, the actual influences on word order are more complex and subtle than those discussed in this thesis. However, it has been demonstrated that, for one informant at least, ordering constraints that refer to grammatical relations and some discourse information such as the distinction between given and new require complex conditional linear ordering statements. It can therefore be concluded that a formalism using a propositional logic such as L_p is required for the concise and economic description of semi-free word order data.

With regard to future research, two features of the model presented here merit further investigation. Firstly, it was pointed out above that the categorial perspective on word order suggests a picture of language in which potentially complex word order constraints are passed up the function-argument hierarchy from the lexicon. These constraints will tend to be local rather than global, insofar as they apply to the immediate arguments of functions. The picture is complicated by the introduction of more complex combinators some of which (for example the disharmonic versions of functional composition) are implicated in word order variation. However, the general picture which is compatible with the model presented here is one in which semi-free word order flexibility operates locally over the arguments of individual functions rather than being a global operation over the entire sentence. It is a matter for further investigation whether such a claim is compatible with the facts of word order variation in natural languages.

Secondly, throughout the thesis the assumption has been made that basic L_p statements of the form \a, /b take an existential interpretation when evaluated as constraints over the argument set of unordered local structures or Extended Categorial Grammar categories. The effect of this is to make a basic L_p expression such as /NP evaluate to T if *at least* one NP in the argument set has a rightward looking directionality marking. This existential interpretation was adopted since it fitted the Spanish semi-free word order data. It is not, however, the only interpretation that can be adopted. Universal quantification of the L_p statement would require that *every* NP in the argument set should carry the appropriate directionality marking. The existential/universal distinction may therefore be viewed as a parameter of language variation. For example, a strictly verb final language such as Japanese would require a universal interpretation of the basic L_p constraint \NP over transitive verbs. It is interesting to note in this context that only one of Greenberg's (1963) universals dealt with alternative basic word orders in the sense that has been explored in this thesis: Universal 6 states that "All languages with dominant VSO order have SVO as an alternative or as the only alternative basic order". This does not exclude the possibility of a language with word orders in the set {VSO, VOS} where VOS is the dominant order. However, it does suggest that universal quantification over rightward looking nominal arguments to verbs is less likely to occur than universal quantification over leftward looking arguments. Much more research needs to be carried out of course to evaluate this suggested parameter.

APPENDIX 1

DATA FOR WORD ORDER IN SPANISH TRANSITIVE SENTENCES

This appendix presents the raw data on word order in simple transitive sentences in Spanish based on an informant's responses. It consists of grammaticality judgements on sets of potential answers to questions given a restricted preceding context. The answers were designed to cover all the syntactically acceptable sequences of S, V and O, that is: SVO, VSO, VOS, and OVS. No clitic pronouns were included in the sentences.

Immediately Preceding Context and Question	Response	Judgement
[Initiating a discourse]	a. unos soldados han arrestado a María (some soldiers arrested Mary)	ok
	b. han arrestado unos soldados a María	ok
	c. han arrestado a María unos soldados	ok
	d. a María han arrestado unos soldados	?*
Comments		

Table A1.1
Data for Context A

Immediately Preceding Context and Question	Response	Judgement
Peter escaped into the countryside. *What happened to John?*	a. la policía arrestó a Juan (the police arrested John)	ok
	b. arrestó la policía a Juan	*
	c. arrestó a Juan la policía	*
	d. a Juan arrestó la policía	*
Comments		

Table A1.2
Data for Context B

Immediately Preceding Context and Question	Response	Judgement
There was a scuffle in the street. *And someone got stabbed?*	a. sí, Pedro apuñaló a Juan (yes, Peter stabbed John)	ok
	b. sí, apuñaló Pedro a Juan	*
	c. sí, apuñaló a Juan Pedro	*
	d. sí, a Juan apuñaló Pedro	?*
Comments		

Table A1.3
Data for Context C

Immediately Preceding Context and Question	Response	Judgement
who arrested Mary?	a. los soldados arrestaron a María (the soldiers arrested Mary)	ok
	b. arrestaron los soldados a María	?
	c. arrestaron a María los soldados	?
	d. a María arrestaron los soldados	?
Comments		
sentence d. is better with *a María* stressed or emphatic.		

Table A1.4
Data for Context D

Immediately Preceding Context and Question	Response	Judgement
Who hid the gun?	a. Juan escondió el fusil (John hid the gun)	ok
	b. escondió Juan el fusil	ok
	c. escondió el fusil Juan	ok
	d. el fusil escondió Juan	*
Comments		
sentence d. is ok with the accusative clitic *lo* before the verb.		

Table A1.5
Data for Context E

Immediately Preceding Context and Question	Response	Judgement
There was a raid on our quarter last night. The police and army were working together. *What did the soldiers do?*	a. los soldados arrestaron a María (the soldiers arrested Mary)	ok
	b. arrestaron los soldados a María	?
	c. arrestaron a María los soldados	*?
	d. a María arrestaron los soldados	*?
Comments		

Table A1.6
Data for Context F

Immediately Preceding Context and Question	Response	Judgement
Peter ran away from the oncoming tanks. *What did John do?*	a. Juan agredió a un grupo de soldados (John attacked a group of soldiers)	ok
	b. agredió Juan a un grupo de soldados	?*
	c. agredió a un grupo de soldados Juan	?*
	d. a un grupo de soldados agredió Juan	?*
Comments		

Table A1.7
Data for Context G

Immediately Preceding Context and Question	Response	Judgement
Peter later killed a a soldier and was arrested. *What about John?*	a. Juan mató a un guardia (John killed a policemen)	ok
	b. mató Juan a un guardia	*
	c. mató a un guardia Juan	?*
	d. a un guardia mató Juan	?*
Comments		
sentence d is acceptable in contrastive contexts ie. *John killed a policeman so he didn't get arrested.*		

Table A1.8
Data for Context H

Immediately Preceding Context and Question	Response	Judgement
Peter tripped the soldier up. *What did John do (to the soldier)?*	a. Juan golpeó al soldado (John struck the soldier)	ok
	b. golpeó Juan al soldado	?
	c. golpeó al soldado Juan	?
	d. al soldado golpeó Juan	*
Comments		

Table A1.9
Data for Context I

Immediately Preceding Context and Question	Response	Judgement
The army was on the streets all night. Some soldiers broke into our house looking for people *Who did they/the soldiers arrest?*	a. los soldados arrestaron a María (the soldiers arrested Mary)	ok
	b. arrestaron los soldados a María	ok
	c. arrestaron a María los soldados	ok
	d. a María arrestaron los soldados	ok
Comments		
sentences b, c and d require *a María* to be stressed.		

Table A1.10
Data for Context J

Immediately Preceding Context and Question	Response	Judgement
What did John hide?	a. Juan escondió un fusil (John hid a gun)	ok
	b. escondió Juan un fusil	ok
	c. escondió un fusil Juan	ok
	d. un fusil escondió Juan	ok
Comments		

Table A1.11
Data for Context K

Immediately Preceding Context and Question	Response	Judgement
[There are several items laid out on a table] *Which one of these did John hide*	a. Juan escondió el fusil (John hid the gun)	ok
	b. escondió Juan el fusil	ok
	c. escondió el fusil Juan	ok
	d. el fusil escondió Juan	ok
Comments		
sentences b, c require *el fusil* to be stressed.		

Table A1.12
Data for Context L

Immediately Preceding Context and Question	Response	Judgement
Did John visit Mary yesterday?	a. sí, Juan visitó a María ayer (yes, John visited Mary yesterday)	ok
	b. sí, visitó Juan a María ayer	ok
	c. sí, visitó a María Juan ayer	?
	d. sí, a María visitó Juan ayer	*
Comments		

Table A1.13
Data for Context M

APPENDIX 2

PATTERNS INDICATING GRAMMATICAL RELATIONS AND GIVEN/NEW INFORMATION IN SENTENCES FROM APPENDIX 1

This Appendix reduces the information in Appendix 1 to a series of tables allowing comparison of Grammatical Relations and GIVEN/NEW information in the sentences.

[Initiating a discourse]				
Constituents	**Pattern**			**Judgement**
NEW information:	S *N*	V *N*	O *N*	ok
S = *unos soldados* (some soldiers)				
V = *han arrestado* (have arrested)	V *N*	S *N*	O *N*	ok
O = *a María* (Mary)	V *N*	O *N*	S *N*	ok
	O *N*	V *N*	S *N*	?*

Table A2.1
Context A

Preceding Question: *What happened to John?*				
Constituents	**Pattern**			**Judgement**
GIVEN information:	S *N*	V *N*	O *G*	ok
O = *a Juan* (John)				
NEW information:	V *N*	S *N*	O *G*	*
S = *la policía* (the police)	V *N*	O *G*	S *N*	*
V = *arrestó* (arrested)	O *G*	V *N*	S *N*	*

Table A2.2
Context B

Preceding Question: *And someone got stabbed?*				
Constituents	**Pattern**			**Judgement**
GIVEN information:	S *N*	V *G*	O *N*	ok
V = *apuñaló* (stabbed)				
NEW information:	V *G*	S *N*	O *N*	*
S = *Pedro* (Peter)	V *G*	O *N*	S *N*	*
O = *a Juan* (John)	O *N*	V *G*	S *N*	?*

Table A2.3
Context C

Preceding Question: *Who arrested Mary?*				
Constituents	**Pattern**			**Judgement**
GIVEN information: V = *arrestaron* (arrested)	S *N*	V *G*	O *G*	ok
O = *a María* (Mary)	V *G*	S *N*	O *G*	?
NEW information:	V *G*	O *G*	S *N*	?
S = *los soldados* (the soldiers)	O *G*	V *G*	S *N*	?

Table A2.4
Context D

Preceding Question: *Who hid the gun?*				
Constituents	**Pattern**			**Judgement**
GIVEN information: V = *escondió* (hid)	S *N*	V *G*	O *G*	ok
O = *el fusil* (a gun)	V *G*	S *N*	O *G*	ok
NEW information:	V *G*	O *G*	S *N*	ok
S = *Juan* (John)	O *G*	V *G*	S *N*	*

Table A2.5
Context E

Preceding Question: *What did the soldiers do?*				
Constituents	**Pattern**			**Judgement**
GIVEN information: S = *los soldados* (the soldiers)	S *G*	V *N*	O *N*	ok
NEW information:	V *N*	S *G*	O *N*	?
V = *arrestaron* (arrested)	V *N*	O *N*	S *G*	?*
O = *a María* (Mary)	O *N*	V *N*	S *G*	?*

Table A2.6
Context F

Preceding Question: *What did John do?*				
Constituents	**Pattern**			**Judgement**
GIVEN information: S = *Juan* (John)	S *G*	V *N*	O *N*	ok
NEW information:	V *N*	S *G*	O *N*	?*
V = *agredió* (attacked)	V *N*	O *N*	S *G*	?*
O = *un grupo de* soldados (a group of soldiers)	O *N*	V *N*	S *G*	?*

Table A2.7
Context G

Preceding Question: *What about John?*				
Constituents	**Pattern**			**Judgement**
GIVEN information:	S *G*	V *N*	O *N*	ok
S = *Juan* (John)				
NEW information:	V *N*	S *G*	O *N*	*
V = *mató* (killed)	V *N*	O *N*	S *G*	?*
O = *a un guardia* (a policeman)	O *N*	V *N*	S *G*	?*

Table A2.8
Context H

Preceding Question: *What did John do (to the soldier)?*				
Constituents	**Pattern**			**Judgement**
GIVEN information:	S *G*	V *N*	O *G*	ok
S = *Juan* (John)				
O = *al soldado* (the soldier)	V *N*	S *G*	O *G*	?
NEW information:	V *N*	O *G*	S *G*	?
V = *golpeó* (struck)	O *G*	V *N*	S *G*	*

Table A2.9
Context I

Preceding Question: *Who did the soldiers arrest?*				
Constituents	**Pattern**			**Judgement**
GIVEN information:	S	V	O	ok
S = *los soldados* (the soldiers)	*G*	*G*	*N*	
V = *arrestaron* (arrested)	V	S	O	ok
	G	*G*	*N*	
NEW information:	V	O	S	ok
O = *a María* (Mary)	*G*	*N*	*G*	
	O	V	S	ok
	N	*G*	*G*	

Table A2.10
Context J

Preceding Question: *What did John hide?*				
Constituents	**Pattern**			**Judgement**
GIVEN information:	S	V	O	ok
S = *Juan* (John)	*G*	*G*	*N*	
V = *escondió* (hid)	V	S	O	ok
	G	*G*	*N*	
NEW information:	V	O	S	ok
O = *un fusil* (a gun)	*G*	*N*	*G*	
	O	V	S	ok
	N	*G*	*G*	

Table A2.11
Context K

Preceding Question: *Which one did John hide?*				
Constituents	**Pattern**			**Judgement**
GIVEN information:	S *G*	V *G*	O *N*	ok
S = *Juan* (John)	V *G*	S *G*	O *N*	ok
V = *escondió* (hid)				
NEW information:	V *G*	O *N*	S *G*	ok
O = *el fusil* (the gun)	O *N*	V *G*	S *G*	ok

Table A2.12
Context L

Preceding Question: *Did John visit Mary yesterday?*				
Constituents	**Pattern**			**Judgement**
GIVEN information:	S *G*	V *G*	O *G*	ok
S = *Juan* (John)	V *G*	S *G*	O *G*	ok
V = *visitó* (visit)				
O = *a María* (Mary)	V *G*	O *G*	S *G*	?
	O *G*	V *G*	S *G*	*

Table A2.13
Context M

APPENDIX 3

EVALUATION OF THE LINEAR ORDERING CONSTRAINT IN 5.8 AGAINST LOCAL STRUCTURES FOR TRANSITIVE SENTENCES WITH GIVEN/NEW INFORMATION

This appendix tabulates the local structures for transitive sentences with all possible assignments of GIVEN/NEW discourse saliency information. The L_p constraint given in the text is evaluated against each local structure. The orderings which are excluded by the L_p constraint in 5.1 are not included in the tables.

The following abbreviations are used:

S = NPsubject

O = NPobject

N = NEW

G = GIVEN

Argument Stack	Surface Order	L_p Constraint				
		/NPsubject	⊃	(*f*:GIVEN	&	/<.., GIVEN,..>)
<\S, /O,> N N N	[S V] O N N N	F	T	F	F	F
<\S, /O,> N N G	[S V] O N N G	F	T	F	F	T
<\S, /O,> N G N	[S V] O G N N	F	T	F	F	F
<\S, /O,> N G G	[S V] O G N G	F	T	F	F	T
<\S, /O,> G N N	[S V] O N G N	F	T	T	F	F
<\S, /O,> G N G	[S V] O N G G	F	T	T	T	T
<\S, /O,> G G N	[S V] O G G N	F	T	T	T	F
<\S, /O,> G G G	[S V] O G G G	F	T	T	T	T

Table A3.1
Truth-table Evaluation of the L_p Statement
in 5.8 against SVO Orders Where S Precedes O on the Argument Stack

Argument Stack	Surface Order	L_p Constraint				
		/NPsubject	$\supset$	(f:GIVEN	&	/<.., GIVEN,..>)
</O, \S> N N N	S [V O] N N N	F	T	F	F	F
</O, \S> N N G	S [V O] G N N	F	T	F	F	F
</O, \S> N G N	S [V O] N N G	F	T	F	F	T
</O, \S> N G G	S [V O] G N G	F	T	F	F	T
</O, \S> G N N	S [V O] N G N	F	T	T	F	F
</O, \S> G N G	S [V O] G G N	F	T	T	F	F
</O, \S> G G N	S [V O] N G G	F	T	T	T	T
</O, \S> G G G	S [V O] G G G	F	T	T	T	T

Table A3.2
Truth-table Evaluation of the L_p Statement
in 5.8 against SVO Orders Where O Precedes S on the Argument Stack

Argument Stack	Surface Order	L_p Constraint				
		/NPsubject	⊃	(f:GIVEN	&	/<.., GIVEN,..>)
</S, \O> N N N	O [V S] N N N	T	F	F	F	F
</S, \O> N N G	O [V S] G N N	T	F	F	F	F
</S, \O> N G N	O [V S] N N G	T	F	F	F	T
</S, \O> N G G	O [V S] G N G	T	F	F	F	T
</S, \O> G N N	O [V S] N G N	T	F	T	F	F
</S, \O> G N G	O [V S] G G N	T	F	T	F	F
</S, \O> G G N	O [V S] N G G	T	T	T	T	T
</S, \O> G G G	O [V S] G G G	T	T	T	T	T

Table A3.3
Truth-table Evaluation of the L_p Statement
in 5.8 against OVS Orders Where S Precedes O on the Argument Stack

Argument Stack	Surface Order	L_p Constraint				
		/NPsubject	$\supset$	(f:GIVEN & /<.., GIVEN,..>)		
<\O, /S> N N N	[O V] S N N N	T	F	F	F	F
<\O, /S> N N G	[O V] S N N G	T	F	F	F	T
<\O, /S> N G N	[O V] S G N N	T	F	F	F	F
<\O, /S> N G G	[O V] S G N G	T	F	F	F	T
<\O, /S> G N N	[O V] S N G N	T	F	T	F	F
<\O, /S> G N G	[O V] S N G G	T	T	T	T	T
<\O, /S> G G N	[O V] S G N N	T	F	T	F	F
<\O, /S> G G G	[O V] S G G G	T	T	T	T	T

Table A3.4
Truth-table Evaluation of the L_p Statement
in 5.8 against OVS Orders Where O Precedes S on the Argument Stack

Argument Stack	Surface Order	L_p Constraint				
		/NPsubject	$\supset$	(f:GIVEN	&	/<.., GIVEN,..>)
</S, /O> N N N	[V S] O N N N	T	F	F	F	F
</S, /O> N N G	[V S] O N N G	T	F	F	F	T
</S, /O> N G N	[V S] O N G N	T	F	F	F	T
</S, /O> N G G	[V S] O N G G	T	F	F	F	T
</S, /O> G N N	[V S] O G N N	T	F	T	F	F
</S, /O> G N G	[V S] O G N G	T	T	T	T	T
</S, /O> G G N	[V S] O G G N	T	T	T	T	T
</S, /O> G G G	[V S] O G G G	T	T	T	T	T

Table A3.5
Truth-table Evaluation of the L_p Statement
in 5.8 against VSO Orders

Argument Stack	Surface Order	L_p Constraint				
		/NPsubject	⊃	(*f*:GIVEN	&	/<.., GIVEN,..>)
</O, /S> N N N	[V O] S N N N	T	F	F	F	F
</O, /S> N N G	[V O] S N N G	T	F	F	F	T
</O, /S> N G N	[V O] S N G N	T	F	F	F	T
</O, /S> N G G	[V O] S N G G	T	F	F	F	T
</O, /S> N N N	[V O] S G N N	T	F	T	F	F
</O, /S> N N G	[V O] S G N G	T	T	T	T	T
</O, /S> N G N	[V O] S G G N	T	T	T	T	T
</O, /S> N G G	[V O] S G G G	T	T	T	T	T

Table A3.6
Truth-table Evaluation of the L_p Statement
in 5.8 against VOS Orders

APPENDIX 4

A PRELIMINARY EXTENDED CATEGORIAL GRAMMAR INVESTIGATION OF CROSS-LINGUISTIC WORD ORDER VARIATION

While collecting the Spanish data for this thesis, data on other Romance languages were encountered and collected, though not so systematically. The languages for which data became available from published sources and native speaker informants are members of the Southern Romance Group, namely, Catalan, Italian and Sardinian. These form an interesting group (along with Spanish) since they exhibit a number of common syntactic characteristics - such as subject drop and extensive use of clitic doubling - which distinguish them from other groups in the Romance family while at the same time they display significant differences in their word order freedom. The aim of this Appendix is to indicate how the L_p factoring of Extended Categorial Grammar may be used to compare and contrast the different word order possibilities in each of these languages.

Data were collected for two variations of basic transitive sentences. Firstly, the sentence with a single intonation contour. This has a single tone group with one tonic nucleus. It is referred to here as the *neutral* form of the sentence. Secondly, sentences which are divided into two distinct intonation contours with two tone groups, each with its own tonic nucleus. This is often rendered by a comma in the written form of the sentences and therefore frequently referred to as *comma intonation*. There are two subtypes of comma intonation sentences: the first has the argument marked with comma intonation on the left periphery of the sentence (called here *preposed*), the second has the argument marked with comma intonation on the right periphery of the sentence (called here *postposed*). Collectively, these will be referred to as *adposed* orders. The logical possibilities for the S, V, O components of these sentences are given in A4.1 below.

A4.1 a neutral:

SVO, SOV, VSO,
VOS, OSV, OVS

b preposed:

S, OV   O, SV

S, VO   O, VS

c postposed:

VO, S   VS, O

OV, S   SV, O

Notice that all the data in the following sections are of transitive sentences *without* pronominal clitics. Doubled clitics can radically alter word order possibilities, a topic discussed in Section 8.4.

In Sections A4.1 to A4.3 below the data for the three languages are briefly presented and a tabular summary of the results is offered. Section A4.4 is a discussion of the L_p constraints required to describe and compare these data.

A4.1 Catalan

Catalan has a much more restricted word order than Spanish, as already noted. Basically, only the VSO order illustrated in A4.2 below is clearly grammatical.

A4.2 l'home va cantar la cançó
(the man sang the song)

The VOS order in A4.3a is at best marginal; the VSO order is not acceptable.

A4.3 a ?va cantar la cançó l'home

b *va cantar l'home la cançó

The exceptions to this are intransitive sentences which can exhibit the VS order, as an alternative to the unmarked SV order.

The data are more interesting when adposed versions of the sentence in A4.2 are considered. Two of the logically possible types of preposing are grammatical; they are given in A4.4.

A4.4 a la cançó, va cantar l'home (O, VS)

b la cançó, l'home va cantar (O, SV)

The preposed direct object can be either definite or indefinite.

No clear example of postposing as defined above is grammatical in Catalan. There are cases in which the sentence can carry *three* intonation contours such as the two variants in A4.5 below. These possibilities require much further research, however.

A4.5 a va cantar, l'home, la cançó (V, S, O)

b l'home, la cançó, va cantar (S, O, V)

A4.2 Italian

At first sight, Italian looks very like Catalan. It has a very restricted word order for neutral, single intonation contour sentences. This is given in A4.6.

A4.6 l'uomo cantó la canzone (SVO)
(the man sang the song)

Informants differ with regard to the acceptability of the VOS order. As with

Catalan, it seems at best marginally grammatical. Clear differences between Catalan and Italian do appear, however, in the data for preposed and postposed variants: Italian is much freer than Catalan.

The data for these patterns is taken from Lepschy and Lepschy (1977, page 155). The acceptable preposed patterns are given in A4.7.

A4.7 a il gatto, ha mangiato la carne (S, VO)
(the cat, has eaten the meat)

b il gatto, la carne ha mangiato (S, OV)

c la carne, il gatto ha mangiato (O, SV)

d la carne, ha mangiato il gatto (O, VS)

The data for the postposed pattern are given in A4.8.

A4.8 a ha mangiato la carne, il gatto (VO, S)

b il gatto ha mangiato, la carne (SV, O)

A4.3 Sardinian

The data for Sardinian is taken from Jones (1988). These data are far from complete but they provide an interesting contrast to Catalan and Italian. Basically, two orders of the neutral transitive sentence are grammatical. They are given in A4.9.

A4.9 a as tunkatu su barkone (VO)
(you-have shut the window)

b su barkone a'tunkatu Juanne (OVS)
(the window has-shut John)

These are strictly not comparable, since A4.9b has an overt lexical subject which is missing in A4.9a. However, the source observes that these are examples of the two grammatical orderings which obey the following generalisations (Jones 1988, page 339)

> 1 the fronting of objects only occurs if the subject is unspecified or inverted;
>
> 2 subject inversion only occurs if there is no explicit postverbal complement (of any sort).

Together, these ensure that only two orders are grammatical: SVO and OVS.

The data on preposing and postposing is sparse in the source and not substantial enough to take as the basis of generalisations. However, two possibilities are recorded and given in A4.10a and A4.10b below (Jones 1988, page 339), and two further orderings can be deduced (though they are not explicitly given in the source) given the following comment: "Definite subjects of finite clauses can also be dislocated with the null subject (or personal inflection on the verb) functioning as the resumptive pronominal element" (Jones 198, page 338). These are given in A4.10c and A4.10d.

A4.10 a Juanne, su barkone a'tunkatu (S, OV)

b a'tunkatu su barkone, Juanne (VO, S)

c Juanne, a'tunkatu su barkone (S, VO)

d su barkone a'tunkatu, Juanne (OV, S)

These data, and those for Catalan and Italian, are summarised in Figure A4.1 below.

Neutral Orders			Adposed Orders	

Catalan

	SVO		*S, VO	*VO, S
*SOV		?VOS	*S, OV	*OV, S
*OSV		*VSO	O, SV	*SV, O
	*OVS		O, VS	*VS, O

Italian

	SVO		S, VO	VO, S
*SOV		?VOS	S, OV	OV, S
*OSV		*VSO	O, SV	SV, O
	*OVS		*O, VS	*VS, O

Sardinian

	SVO		S, VO	VO, S
*SOV		*VOS	S, OV	OV, S
*OSV		*VSO	*O, SV	*SV, O
	OVS		*O, VS	*VS, O

Figure A4.1
Summary of the Word Order Data for Southern Romance languages

A4.4 Linear Ordering Constraints: a Comparison

As indicated above, this section is necessarily tentative. There are enough data in Figure A4.1, however, to provide some preliminary indications of how Extended Categorial Grammar might describe the variations found among these languages. In particular, the data on Italian are sufficiently detailed to allow some account of the word ordering constraints in that language and hence a comparison with Spanish.

Catalan and Italian neutral orders are very restricted, with the subject to the left and direct object to the right. The L_p constraint in A4.11 expresses this.

A4.11 \[-Objective] & /[+Objective]

If the VOS order is to be included in the the L_p constraint, this poses interesting questions about the existence of the analogue of a Verb Phrase in the categorial analysis of these languages. It is possible that a constraint such as that in A4.12 is required where the verb combines with the direct object from the right before combining with the subject.

A4.12 </[+Objective], [-Objective]>

This constraint guarantees that the VSO order is not admitted, as seems appropriate.

The neutral word order for Sardinian is more problematic. The two grammatically acceptable orders are mirror images of one another and require a complex disjunctive constraint to exclude other sequences. The use of variables over DIR values in the L_p constraint would allow for a degree of generalisation over the data, as shown in A4.13, but the effects of introducing this kind of variable would have to be studied further.

A4.13 α[-Objective] & $\neg\alpha$[+Objective]

The adposed orders for these languages are more complex and interesting. Catalan and Italian may have very similar ordering constraints for the neutral orders but their adposed orders differ significantly, as the summary table above indicates. The analysis of these orders requires some preliminary working assumptions and definitions.

Assume a binary feature P which is taken by a Noun Phrase argument of a verb if it is to be preposed or postposed in a sentence. This feature can be understood as the marker at the syntactic level of the intonational independence of the Noun Phrase; any Noun Phrase carrying this feature will have an independent intonation contour. Further, assume that only one Noun Phrase in the sentence carries this feature and that it is always peripheral. The orderings in A4.14 are consistent with these assumptions; the orderings in A4.15 are not.

A4.14 a O[$+P$], S V

b O[$+P$], V S

A4.15 a S, O[$+P$] V

b S[$+P$], O[$+P$] V

These two assumptions are counterfactual since it it certainly possible in the languages under investigation here to have sentences with *three* intonation contours, in which case all three constituents would be [$+P$]. Data is not available for these cases, however, therefore these simplifying assumptions are introduced to allow for a presentation of the constraints appearing in the data available.

The L_p constraint for Catalan is given in A4.16 below.

A4.16 [$+P$] $\supset$ \[+Objective, $+P$]

This states that if there is an adposed argument to the verb, it must be the object and come from the left, thereby capturing the data in Figure A4.1 above.

The L_p constraint for Italian is given in A4.17.

A4.17 $[+P] \supset \neg(/[\text{-Objective}]\ \&\ [+\text{Objective}, +P]$

This is the most complex constraint of the three languages under discussion. It disallows a subject from the right if the object is adposed.

Again, the Sardinian data are different. The L_p statement in A4.18 expresses the necessary constraint.

A4.18 $[+P] \supset \neg[+\text{Objective}, +P]$

It has been pointed out already that there may be many ways of expressing a single set of ordering facts. The three L_p statements in A4.16 to A4.18 have each been expressed with an implication with the same antecedent, which is simply a check that the sentence for which the constraint might be true really does have an adposed constituent. The consequents of the three L_p expressions all refer to the adposed object [+Objective, $+P$] but place different constraints on the acceptability (or non-acceptability) of this constraint. Italian is clearly more complex given the data available. It is these data which are taken as reinforcement of the observation that, at least with respect to what is at present known, an ordering logic of the power of L_p is required. Weaker boolean systems of constraints may prove eventually to be adequate and, if so, this would be an interesting finding, but the complexity of the Italian facts suggests that it is premature to restrict the descriptive power of the ordering logic at this stage.

Finally in this section it is interesting to examine the data for Italian taking one further piece of information into account. Lepschy and Lepschy (1977), from which the Italian data above are taken, also give the orderings acceptable when the assignment of NEW and GIVEN information is included.

The data are summarised in Figure A4.2, where the *N* is used to indicate new information. According to Lepschy and Lepschy (1977), these are the only grammatical sequences allowed in Italian. However, further field work needs to be done to confirm these findings.

The assignments appear complex and indeed Lepschy and Lepschy make no attempt to comment on them except to list them as possibilities. It is, nevertheless, possible to define the constraints on these orderings within the Extended Categorial Grammar framework. Two constraints characterise the data in Figure A4.2: the subject is never NEW information and adposed arguments are never NEW. These constraints are expressed by the L_p statements in A4.19.

A4.19 a ¬[-Objective]
[DS: NEW]

b ¬[+P]
[DS: NEW]

S,	V	O		V	O,	S
		N			*N*	
S,	V	O		V	O,	S
	N			*N*		
S,	O	V		O	V,	S
	N			*N*		
O,	S	V		S	V,	O
		N			*N*	

Figure A4.2
NEW/GIVEN Information in Italian Adposed Transitive Sentences

The first two rows of Figure A4.2 show that with a VO order and adposed subject, *all* assignments of discourse saliency information are acceptable, subject to the constraints in A4.19. The bottom row of the Figure contains all possibilities for adposed direct objects since the L_p constraint in A4.17 prohibits the VS order in this case and the subject cannot be NEW information according to A4.19a.

The third row of Figure A4.2 poses a problem. The orderings in A4.20 below are unacceptable but no combination of the constraints so far given will exclude them.

A4.20 a $*S, O\ \underset{N}{V}$

b $*O\ \underset{N}{V}, S$

The generalisation required is that when a non-adposed object comes from the left, the verb must be GIVEN information. This is expressed in A4.21.

A4.21 \[+Objective, -P] $\supset$ [DS: GIVEN]

These constraints work together to predict the grammatical relations of the Noun Phrases to the verb together with the distribution of discourse saliency information.

REFERENCES

Abraham, W. and Meij, S. de (1986), *Topic, Focus, and Configurationality: Papers from the 6th Groningen Grammar Talks*, Benjamins, Amsterdam.

Ades, A. E. and M. Steedman (1982), On the Order of Words. *Linguistics and Philosophy*, 4, pages 517-558.

Ajdukiewicz, k. (1935), Die syntakische konnexitä. *Studia Philosophica*, 1, pages 1-27. Translated in **McCall, S. (ed.)** (1967), *Polish Logic 1920-1939*, Oxford.

Andrews, A. (1985), The major functions of the noun phrase. In **Shopen, T. (ed)** (1985), *Language Typology and Syntactic Description. Volume 1: Clause Structure*, Cambridge University Press, Cambridge, pages 62-154.

Bach, E. (1979), Control in Montague Grammar. *Linguistic Inquiry*, 10, pages 515-531.

Bach, E. (1981), Discontinuous Constituents In Generalized Categorial Grammar, in *Proceedings of the Eleventh Annual Meeting of the North Eastern Linguistic Society*, Graduate Linguistics Student Association, University of Massachusetts at Amherst, pages 1-12.

Bach, E. (1983a), On the relationship Between Word-Grammar and Phrase-Grammar. *Natural Language and Linguistic Theory*, 1, 65-89.

Bach, E. (1983b), Generalised Categorial Grammars and the English Auxiliaries. In **Heny, F and B. Richards (eds.)** (1983), *Linguistic Categories* Vol II, Reidel, Dordrecht, pages 161-202.

Bach, E. (1984), Some Generalisations of Categorial Grammar. In **Landman, F. and F. Veltman (eds.)** (1984),*Varieties of Formal Semantics: Proceedings of the Fourth Amsterdam Colloquium, September 1982*, Foris, Dordrecht, pages 1-23.

Bar-Hillel, Y. (1953), A quasi-arithmetical notation for syntactic description. *Language*, 29, pages 47-58.

Beaven, J. L. (1989), *A UCG Treatment of Spanish Clitics*. Manuscript, Department of Artificial Intelligence, University of Edinburgh.

Benthem, J. van (1986), Categorial Grammar. In **Benthem, J. van** (1986), *Essays in Logical Semantics*, Reidel, Dordrecht, pages 123-150.

Bok-Bennema, R (1982), Clitics and Binding in Spanish. In **May, R. C. and J. Koster (eds.)** (1982), *Levels of Syntactic Representation*, Reidel, Dordrecht , pages 9-32.

Borer, H. (ed.), *Syntax and Semantics, Vol 19: The Syntax of Pronominal Clitics*, Academic Press, London.

Bresnan, J. (ed.) (1982), *The Mental Representation of Grammatical Relations*, MIT Press, Cambridge, Massachusetts.

Brody, J. (1984), Some Problems with the Concept of Basic Word Order. *Linguistics*, 22, pages 711-736.

Bouma, G. (1988), Modifiers and specifiers in categorial unification grammar. *Linguistics*, 26, pages 21-46.

Calder, J., E. Klein and H. Zeevat (1988), Unification Categorial Grammar: A Concise, Extendible Grammar for Natural Language Processing. In *Proceedings of the 12th International Conference on Computational Linguistics and the 24th Annual Meeting of the Association of Computational Linguistics, Budapest, August 1988.*

Chomsky, N. (1970), Remarks on nominalization. In **Chomsky, N.** (1976), *Studies on Semantics in Generative Grammar*, Mouton, The Hague, pages 11-61.

Cohen, J. (1967), The equivalence of two concepts of categorial grammar. *Information and Control*, 10, pages 475-484.

Comrie, B. (1976), *Aspect*, Cambridge University Press, Cambridge.

Contreras, H. (1978), *El Orden de Palabras en Español*. Cátedra, Madrid.

Cook, V. J. (1988), *Chomsky's Universal Grammar: An Introduction*, Blackwell, Oxford.

Curry, H. B. and R. Feys (1958), *Combinatory Logic: Vol I*, North Holland, Amsterdam.

Dowty, D. R. (1982), Grammatical Relations and Montague Grammar. In **Jacobson, P. and G. K. Pullum (eds.)** (1982), *The Nature of Syntactic Representation*, Reidel, Dordrecht, pages 79-130.

Dowty, D. R. (1988), Type Raising, Functional Composition and Non-Constituent Coordination. In **Oehrle, R., E. Bach and D. Wheeler (ed.) (1988)**, *Categorial Grammars and Natural Language Structure*, Reidel, Dordrecht, pages 153-197.

Dowty, D. R., L. Karttunen and A. Zwicky (eds.) (1985), *Natural Language Parsing: Psychological, Computational and Theoretical Perspectives*, Cambridge University Press, Cambridge.

Dowty, D. R., R. E. Wall and S. Peters (1981), *Introduction to Montague Semantics*, Reidel, Dordrecht.

Fish, G. T. (1961), "Lo puede hacer" vs. "Puede hacerlo". *Hispania*, 44(2), pages 137-139.

Flynn, M. (1983), A Categorial Theory of Structure Building. In **Gazdar, G., E. Klein and G. Pullum (eds.) (1983)**, *Order, Concord and Constituency*, Foris, Dordrecht, pages 139-174.

Gazdar, G. (1982), Phrase Structure Grammar. In **Jacobson, P. and G. K. Pullum (eds.) (1982)**, *The nature of Syntactic Representation*, Reidel, Dordrecht, pages 131-186.

Gazdar, G., E. Klein, G. Pullum and I. Sag (1985), *Generalised Phrase Structure Grammar*, Blackwell, Oxford.

Gazdar, G., G. K. Pullum, R. Carpenter, E. Klein, T. E. Hukari and R. D. Levine (1987), *Category Structures*, CSLI Technical Report, Stanford University, California.

Gazdar, G., G. K. Pullum and I. A. Sag (1982), Auxiliaries and related phenomena in a restrictive theory of grammar. *Language*, 58, pages 591-638.

Geach, P. (1971), A program for syntax, *Synthese 22, pages 3-17. Also in* **Davidson, D. and G. Harman (eds.) (1972)**, *Semantics of Natural Language*, Reidel, Dordrecht, pages 483-497.

Gili Gaya, S. (1982), *Curso Superior de Sintaxis Española*, Vox, Barcelona.

Green, John N. (1988), Spanish. In **Harris, M. and N. Vincent (eds.) (1988)**, *The Romance Languages*, Croom Helm, London, pages 79-130.

Greenberg, J. H. (1963), Some Universals of Grammar with Particular Reference to the Order of Meaningful Elements. In **Greenberg, J. H. (1963)**, *Universals of Language*, MIT Press, Massachusetts, pages 73-113.

Groos, A. and R. Bok-Bennema (1987), The structure of the sentence in Spanish. In **Bordelois, B. (1987)**, *Syntactic Theories of Spanish*, Foris.

Gunji, T. (**1986a**), *Japanese Phrase Structure Grammar*. Reidel, Dordrecht.

Gunji, T. (**1986b**), Subcategorisation and Word Order. Paper delivered at *The International Symposium on Language and Artificial Intelligence*, Kyoto University, Japan, March 16-21, 1986.

Haack, S. (**1978**), *Philosophy of Logics*, Cambridge University Press, Cambridge.

Haddock, N., E. Klein and G. Morrill (**1987**), *Working Papers in Cognitive Science, Volume 1: Categorial Grammar, Unification Grammar and Parsing*. Centre for Cognitive Science, University of Edinburgh.

Halvorsen, P. -K. (**1983**), Semantics for Lexical-Functional Grammar. *Linguistic Inquiry*, 14(4), pages 567-616.

Halvorsen, P. -K. and W. Ladusaw (**1979**), Montague's "Universal Grammar": An Introduction for the linguist. *Linguistics and Philosophy*, 3, pages 185-223.

Huck, G. J. (**1985**), *Discontinuity and Word Order in Categorial Grammar*, Indiana University Linguistics Club, Bloomington, Indiana.

Jackendoff, R. (**1977**), *X-bar Syntax: A Study of Phrase Structure*, MIT Press, Cambridge, Massachusetts.

Jaeggli, O. A. (**1986**), Three Issues in the Theory of Clitics: Case, Doubled NPs, and Extraction. In **Borer, H. (ed.)**, *Syntax and Semantics, Vol 19: The Syntax of Pronominal Clitics*, Academic Press.

Jones, M. (**1988**), Sardinian. In **Harris, M. and N. Vincent** (**eds.**) (**1988**), *The Romance Languages*, Croom Helm, London, pages 314-350.

Kaplan, R. and J. Bresnasn (**1982**), Lexical-Functional Grammar: A Formal System for Grammatical Representation. In **Bresnan, J.** (**ed.**) (**1982**), *The Mental Representation of Grammatical Relations*, KIT Press, Cambridge, Massachusetts.

Karttunen, L. (**1984**), *Features and Values*. In **Shieber, S. M., L. Karttunen and F. C. N. Pereira** (**eds.**) (**1984**), *Notes from the Unification Underground: a Compilation of Papers on Unification-based grammar Formalisms*. SRI International Technical Note 327, SRI, Menlo Park, California, Chapter 2, pages 17-36.

Kay, M. (**1983**), *Unification Grammar*, Technical Report, Xerox Palo Alto Research Center, Palo Alto, California.

Kay, M. (**1985**), Parsing in Functional Unification Grammar. In **Dowty, D., L. Karttunen and A. Zwicky** (**eds.**) (**1985**), *Natural Language Parsing: Psychological, Compu-*

tational and Theoretical Perspectives, Cambridge University Press, Cambridge, Chapter 7, pages 251-278.

Keenan, E. L. (1974), The functional principle: generalising the notion of "subject of". In **La Galy, M. W., R. A. Fox and A. Bruck (eds)** (1974), *Papers from the 10th Regional Meeting of the Chicago Linguistics Society*, pages 298-309.

Kleene, S. C. (1952), *Introduction to Metamathematics*, North Holland.

Klein, E. and J. van Benthem (eds.) (1987), *Categories, Polymorphism and Unification*, Centre for Cognitive Science and Institute for Language, Logic and Information, Edinburgh and Amsterdam.

Lambek, J. (1958), The Mathematics of Sentence Structure. *American Mathematical Monthly*, 65, pages 154-169.

Lambek, J. (1961), On the Calculus of Syntactic Types. In **Jakobson, R (ed.)** (1961), *Structure of Language and its Mathematical Aspects*, Providence, pages 166-178.

Lambek, J. (1988), Categorial Grammars and Categorical Grammars. In **Oehrle, R., E. Bach and D. Wheeler (ed.)** (1988), *Categorial Grammars and Natural Language Structure*, Reidel, Dordrecht, pages 297-317.

Lepschy, A. L. and G. Lepschy (1977), *The Italian Language Today*, Hutchinson, London.

Luján, M. (1980), Clitic promotion and mood in Spanish verbal complements. *Linguistics*, 18, pages 381-484.

Łukasiewicz, J. (1920), On 3-valued logic. In **McCall, S. (ed.)** (1967), *Polish Logic 1920-1939*, Oxford.

Lyons, J. (1968), *Introduction to Theoretical Linguistics*, London and New York, Cambridge University Press.

McDermott, D. (1978), Tarskian semantics, or no notation without denotation, *Cognitive Science*, 2(3), pages 277-282.

Montague, R. (1970), Universal grammar, *Theoria* 36, pages 373-398. Reprinted in **Thomason, R. H.** (1974), *Formal Philosophy: Selected Papers of Richard Montague*, New Haven, Yale University Press, pages 222-246.

Montague, R. (1973), The proper treatment of quantification in ordinary English. In **Hintikka, J., J. M. E. Moravcsik and P. Suppes (eds.)** (1973), *Approaches to Natural Language*, Reidel, Dordrecht. Reprinted in **Thomason, R. H.** (1974), *Formal Philo-*

sophy: Selected Papers of Richard Montague, New Haven, Yale University Press. pages 247-270.

Monzon, M. C. **(1979),** *A Constituent Structure Rule Grammar of the Spanish Clitic Positioning in Complex and Simple Sentences.* MA Thesis presented at the University of Texas at Austin, December 1979.

Moortgat, M. **(1988a),** Mixed Composition and Discontinuous Dependencies. In **Oehrle, R., E. Bach and D. Wheeler (ed.)** **(1988),** *Categorial Grammars and Natural Language Structure*, Reidel, Dordrecht, pages 297-317.

Moortgat, M. **(1988b),** *Categorial Investigations: Logical and Linguistic Aspects of the Lambek Calculus.* Foris, Dordrecht.

Moortgat, M., R. Oehrle and M. M. Wood **(1987),** *Categorial Grammar Newsletter, Issue 1.* Foris, Dordrecht.

Moortgat, M., R. Oehrle and M. M. Wood **(1988),** *Categorial Grammar Newsletter, Issue 2.* Foris, Dordrecht.

Moravcsik, J. M. E. **(1972),** *Understanding Language*, Mouton, The Hague.

Oehrle, R., E. Bach and D. Wheeler (ed.) **(1988),** *Categorial Grammars and Natural Language Structure*, Reidel, Dordrecht.

Pereira, F. C. N. and S. M. Shieber **(1984),** *The Semantics of Grammar Formalisms Seen as Computer Languages.* In In **Shieber, S. M., L. Karttunen and F. C. N. Pereira (eds.)** **(1984),** *Notes from the Unification Underground: a Compilation of Papers on Unification-based grammar Formalisms.* SRI International Technical Note 327, SRI, Menlo Park, California, Chapter 3, pages 37-54.

Pollard, C. **(1985),** *Lectures on HPSG.* Unpublished ms, Stanford University, California.

Pollard, C. and I. A. Sag **(1987),** *Information-Based Syntax and Semantics. Volume 1: Fundamentals*, CSLI Lecture Notes Number 13, Stanford University, California.

Pullum, G. **(1982),** Free Word Order and Phrase Structure Rules. In **Pustejovsky, J. and P. Sells (eds.)** **(1982),** *Proceedings of the Twelfth Annual Meeting of the North Eastern Linguistic Society*, Graduate Linguistics Student Association, University of Massachusetts at Amherst, pages 209-220.

Ramsey, M. M. **(1956),** *A Textbook of Modern Spanish*, Holt, Rinehart and Winston, New York.

Riemsdijk, H. C. van (1983), The case of the German adjective. In In **Heny, F and B. Richards (eds.) (1983)**, *Linguistic Categories: Auxiliaries and Related Puzzles*, Reidel, Dordrecht, pages 223-252.

Roldán, M. (1972), Clitic Climbing and Unrelated Matters. *Linguistic Inquiry*, 4(2), pages 342-345.

Schönfinkel, M. (1924), Über die Bausteine der Mathematischen Logik, *Mathematische Annalen*, 92, pages 305-316.

Shieber, S. M. (1983), Direct parsing of ID/LP grammars. *Linguistics and Philosophy*, 7, pages 135-154.

Shieber, S. M. (1984), *The Design of a Computer Language for Linguistic Information.* In In **Shieber, S. M., L. Karttunen and F. C. N. Pereira (eds.) (1984)**, *Notes from the Unification Underground: a Compilation of Papers on Unification-based grammar Formalisms.* SRI International Technical Note 327, SRI, Menlo Park, California, Chapter 1, pages 4-16.

Shieber, S. M. (1986), *An Introduction to Unification-Based Approaches to Grammar*, CSLI Lecture Notes, Number 4, Stanford, California.

Siewierska, A. (1988), *Word Order Rules*, Croom Helm, London.

Steedman, M. (1985), Dependency and Coordination in the Grammar of Dutch and English, *Language*, 61, pages 523-568.

Steedman, M. (1987a), Combinatory Grammars and Parasitic Gaps. *Natural Language and Linguistic Theory*, 5, pages 403-439.

Steedman, M. (1987b), Gapping as Constituent Coordinations. Draft ms, October 7, 1987, University of Edinburgh.

Steedman, M. (1988), Combinators and Grammars. In **Oehrle, R., E. Bach and D. Wheeler (ed.) (1988)**, *Categorial Grammars and Natural Language Structure*, Reidel, Dordrecht, pages 417-442.

Suñer, M. (1988), The Role of Agreement in Clitic-Doubled Constructions, *Natural Language and Linguistic Theory*, 6, pages 391-434.

Thomason, R. H. (ed) (1974), *Formal Philosophy: Selected Papers of Richard Montague.* New Haven, Yale University Press.

Tomlin, R. S. (1986), *Basic Word Order: Functional Principles*, Croom Helm, London.

Torrego, E. (1984), On Inversion in Spanish and Some of Its Effects. *Linguistic Inquiry*, 15 (1), pages 103-129.

Uszkoreit, H. (1984), *Word Order and constituent structure in German*. Unpublished Ph.D. Thesis, University of Texas at Austin.

Uszkoreit, H. (1986a), Categorial Unification Grammars. In *Proceedings of the 11th International Conference on Computational Linguistics and the 23rd Annual Meeting of the Association of Computational Linguistics*, Institut für Kommunikationsforschung und Phonetik, University of Bonn, 25-29 August, 1986, pages 187-194.

Uszkoreit, H. (1986b), Constraints on order. *Linguistics*, 24, 883-906.

Warner, A. R. (1984), *The Structuring of English Auxiliaries: A Phrase Structure Grammar*. manuscript, University of York, January 1984.

Wood, M. M. (1988), *A Categorial Syntax for Coordinate Constructions*. Unpublished Ph.D Thesis, University College, London.

Zeevat, H., E. Klein and J. Calder (1987), Unification Categorial Grammar. In **Haddock, N., E. Klein and G. Morrill** (1987), *Working Papers in Cognitive Science, Volume 1: Categorial Grammar, Unification Grammar and Parsing*. Centre for Cognitive Science, University of Edinburgh, pages 195-222.

Zwicky, A. (1977), On clitics. In **Dressler, W. and O. Pfeiffer** (eds.) (1977), *Phonologica 1976*, Institut fur Sprachwissenschaft der Universität Innsbruck, Innsbruck, pages 29-39.