KATHOLIEKE UNIVERSITEIT LEUVEN
Departement Linguïstiek

Process Linguistics

THE THEORY AND PRACTICE OF A COGNITIVE-SCIENTIFIC APPROACH TO NATURAL LANGUAGE UNDERSTANDING

PROEFSCHRIFT,
ter verkrijging van de graad
van doctor in de
Letteren en de Wijsbegeerte
ingediend door

Geert ADRIAENS

Promotor: Prof. dr. G. GEERTS
Co-promotor: Prof. dr. Y.D. WILLEMS

Leuven, september 1986

To David Marr and Hilde Smedts

Everything is process and interaction, from cognitive science to quantum mechanics, from the metascientific level of interacting sciences to the subatomic level of interacting particles in the cosmic dance. And maybe everything is uncertainty too, then, from quantum mechanics to cognitive science.

ACKNOWLEDGEMENTS

Without the help of a number of people this book would never have been finished before its moral deadline.

In the first place I want to thank my advisor, professor Geerts, for having offered me the opportunity to write a dissertation and for his quiet, tremendously stimulating trust in an enterprise of which he left the management to a great extent in the hands of an unexperienced researcher.

Next, I wish to thank professor Small (or rather, Steve) for accepting me as a visiting researcher at the University of Rochester, New York in the summer of '85. If it had not been for his friendship, his amazing zest in general and his unflagging willingness to help me with matters ranging from LISP and WEP to AI and cognitive science, I would never have been able to finish in three months abroad what I could not have dreamt of finishing in one year in Belgium.

I also extend warm thanks to dr. Geeraerts (or rather, Dirk), with whom I was very lucky to share my office during the last year of my dissertation work at the Department of Linguistics. Without his help -- especially in the areas of the philosophy of science and of the lexicon in linguistics -- many of the theoretical matters discussed would have escaped my attention or remained a lot more incomplete than they (still) are now. Beside for the serious talks, I also thank him for his pleasant company and great sense of humor (of which, alas -- or luckily?, no traces will be found in this book).

My thanks also go to professsor Willems, to whom I owe much of my training in computer science. I am also grateful for his remote -- due to an incredibly busy life at the University -- but important interest in my work, his stimulation to go to the United States, and his support in attempts to continue the research of this book in a broader perspective.

I also wish to thank the people of the Linguistics Department for their comments on earlier versions of chapters of this book or on dissertation-writing in general: Jo Rooryck, Frieda Steurs, Karel van den Eynde, Rudi Gebruers, Ludo Jacobs, Lieven Jaspaert. And, last but not least, there are the people of the Rochester Departments of Computer Science and Psychology whose hospitality and company I enjoyed very much: Chris Brown, Sue Bell, Jay Webber, Cesar Quiroz, Josh Tenenberg, Nemo, Brian Stuart, Tom Blenko, Mike Tanenhaus, Gary Dell. I hope some day I will be back to take part in one of the many parties or to have tea & chocolate chips in the middle of a sticky Rochester night.

My thanks also go to the Belgian National Fund for Scientific Research (N.F.W.O.) for its four-year financial support of my work, as well as to IBM who made my stay in the United States financially possible.

Table of Contents

CHAPTER 1 INTRODUCTION : A COGNITIVE SCIENCE FRAMEWORK

CHAPTER 2: WHY NOT GENERATIVE GRAMMAR FOR MODELS OF NATURAL LANGUAGE UNDERSTANDING ?

CHAPTER 3: TOWARD PROCESS LINGUISTICS

CHAPTER 4: PROCESS LINGUISTICS, WORD EXPERT PARSING, AND DUTCH

CHAPTER 5: WEP CONFRONTED WITH PSYCHO- AND NEUROLINGUISTIC RESEARCH

CHAPTER 1 INTRODUCTION: A COGNITIVE SCIENCE FRAMEWORK

"*If cognitive science does not exist then it is necessary to invent it.*"
(Johnson-Laird 1981, 147)

1.1. Introductory summary

This book deals with natural language understanding within a cognitive science framework. On the basis of serious doubts about the usefulness of generative linguistics for such an enterprise, a new approach to natural language understanding is suggested ("process linguistics"). A computer program is then presented which implements important aspects of the approach. Finally, the cognitive plausibility of the program is critically looked at through a confrontation with psycho- and neurolinguistic research.

Since such an approach may look uncommon (why should a linguist bother with language *understanding*? why computer programs? why "cognitive plausibility" of programs? what is cognitive science anyway?), this chapter will motivate it and explore its (philosophical) assumptions and implications.

1.2. From parsing to cognitive science

After studying linguistics (mainly Dutch linguistics) and computer science, writing a natural language understanding (further NLU) or - shorter - a parsing (1) system for Dutch seemed like an interesting topic for a book. So I started exploring the literature dealing with existing systems --

---

(1) I use "parsing" in the broad sense throughout this book, i.e. it does not just mean "assigning a (syntactic) structure to a sentence", but "analyzing a fragment of text in order to fully understand its meaning (in context)"; parsing in the narrow sense will be called "syntactic parsing". What I mean by "fully understand" will be clarified in 4.2.2.2 where the conceptual output structure of the computer model is discussed.

soon to discover that there are about as many systems as researchers. (Winograd (1983, chapter 7) gives an overview of over fifty systems developed for the analysis of English over the last twenty years.) Moreover, these researchers come from a variety of disciplines (2) and understandably have their idiosyncratic ways of presenting and motivating their systems. Linguists and psychologists try to design and implement programs that show the correctness of their theories about language and the way it is processed (computability seems to be a new kind of theory justification), whereas researchers in artificial intelligence (AI) are more concerned with writing practically useful NLU systems. (For some scientists the proof of the theory is in the program whereas for others the program *is* the theory (cp. Winograd 1977, 172).) Beside leaving me slightly disoriented, this tour of the literature raised a number of more fundamental questions than the original "simple" problem of writing a program that can understand a subset of Dutch. Although at this point these questions may seem to appear out of the blue, they led me to cognitive science and hence they have to be mentioned here.

1 can/must a linguistic theory form the basis of a full-scale NLU system?

2 can/must a psychological theory form the basis of such a system?

3 if (as most researchers cannot help doing) one claims a relationship between cognitive processes hypothesized/discovered in human beings (say, for NLU) and computational processes in NLU programs, then what is the exact nature of this relationship?

4 is it possible at all to "re-create" human behavior on a digital computer? if so, does this imply that human cognitive processes are independent of their neural substrate in that they can just as well be carried out by "beings"

---

(2) To name but a few: generative and computational linguistics (Bresnan 1982, Berwick & Weinberg 1984); psycholinguistics (Frazier & Fodor 1978); cognitive psychology (Anderson & Bower 1973); computer science and artificial intelligence (Pereira & Warren 1980, Marcus 1980, Schank & Riesbeck 1981).

that do not have an organic brain but are made of metal and plastic? (the age-old mind-body problem revisited)

In the course of this book attempts will be made to give (partial) answers to these questions (more or less starting with the last and ending with the original problem). The point I want to make here is that in my conviction anyone who wants to develop a thoroughly motivated parsing system (or apply an existing one to a different language than it was originally applied to) should at least take a position as far as these questions are concerned. Although one might easily assign them to the traditional sciences (the first to linguistics, the second to psychology, the third to AI, and the fourth to philosophy), it is clear that they are all related to each other and to the area of NLU and its computational realization. If one wants to make progress in dealing with NLU in all its complexity, it becomes necessary to cross the boundaries between disciplines and enter the realm of cognitive science (see 1.3), be it without giving up one's own scientific identity (see 1.3.4 and chapters 2 and 3).

1.3. A tour of the field

1.3.1. A definition of *cognitive science*

Drawing on a number of definitions given by others (3) I define "cognitive science" as follows:

> Cognitive science wants to be a contemporary scientific paradigm that brings together a number of existing fields (artificial intelligence, psychology, neuroscience, philosophy, linguistics and anthropology) in a concerted effort to study cognition/intelligence in its broadest sense in order
> 1) to reach a better understanding of human behavior and its relation to the mind (cognitive processes) and the brain (neural processes) and
> 2) to develop intelligent devices that can augment human capabilities in important and constructive ways.

(3) Collins 1977, Norman 1981, Pylyshyn 1984, Gardner 1985.

To reach this goal it uses the research tools recently developed in its participating sciences.

By "cognition/intelligence in its broadest sense" as the object of study is meant human and/or machine intelligence, including a.o. problems of knowledge representation, language processing, learning, reasoning and problem solving. Its most important tools come from cognitive psychology (rigorous experimentation and disciplined introspection (see Simon and Ericsson 1984)) and artificial intelligence (computer simulation of theories, using a wide variety of computer languages and formalisms such as frames, networks, etc.).

Elaborating on this definition I will first take a look at cognitive science from the perspectives of its participating disciplines (1.3.2), leaving out linguistics for separate discussion in 1.3.4. In 1.3.3 I will enumerate the most important characteristics of cognitive science in general (characteristics that will be visible throughout 1.3.2, but in a less systematic fashion) and briefly discuss them. Subsection 1.3.2 now introduces a large number of important notions and the (changing) perspective on them in cognitive science; my position with respect to some of these notions will define my own motivated view of cognitive science.

1.3.2. The hexagon of disciplines

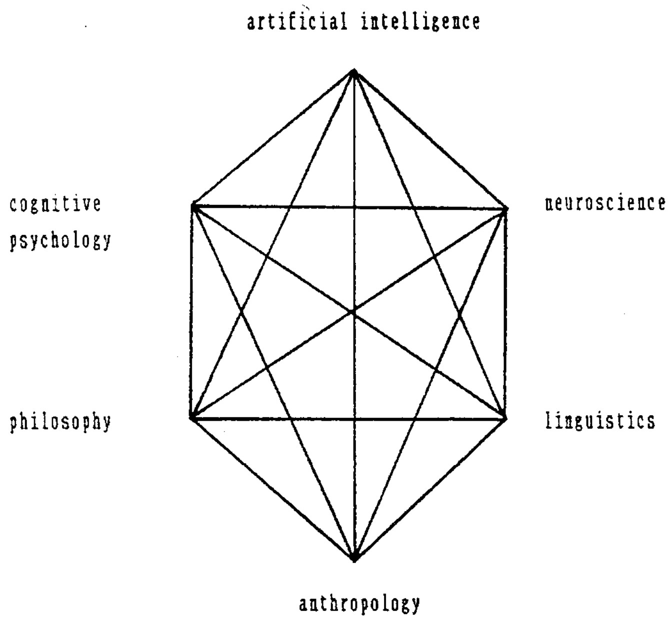


Figure I. Participating disciplines of cognitive science

Figure I shows the six participating sciences in a hexagon with the interconnecting lines indicating all the possible interdisciplinary links among them (4). Before going into some of the individual links and their strength (cp. Gardner 1985, 37) I suggest looking at the Figure as a "Gestalt" and start by putting the convergence of the disciplines in a historical perspective (5).

---

(4) It will be clear that in so far as these disciplines are in turn closely related to others (e.g. artificial intelligence to computer science, in turn related to logic), there is more to cognitive science than just these six disciplines. Hence, the hexagon is not "closed", but restricting the basic disciplines to these six helps to give cognitive science a distinct face.

(5) In what follows, I only scratch the surface of the history of cognitive science, a history that is excellently dealt with in Gardner's "The Mind's New Science. A History of

By the end of the 1950s -- where most scientists situate the birth of cognitive science -- psychology, linguistics and anthropology were becoming "cognitive" in that they all posited the necessity to study mental representations and processes in their respective fields. In psychology the behaviorist stimulus-response approach to human intelligent behavior that banished all reference to the operation of the mind was abandoned and psychologists like George Miller started studying aspects of human cognition, very much influenced by developments in information theory and computer science (see below). Thus, after the behaviorist reaction against the psychology of the early 20th century -- inspired by Wundt -- that relied heavily on introspection in its study of conscious experience, the pendulum swung back to cognitivism, but this time without the use of introspection as the most important method. Someone who *did* embrace introspection as a method, however, was the linguist Noam Chomsky, who reacted against behaviorism (the well-known review of Skinner's *Verbal Behavior*) and against American structural linguistics that was related to it (a.o. through its inductive methodology). Chomsky's commitment to mentalism consisted in positing the existence of abstract structures and rules (later "principles") in the mind; moreover, these entities were (and are) considered to be innate in a separate "mental organ", the language faculty. (I will come back to Chomsky's ideas in my discussion of generative linguistics in chapter 2.) Finally, anthropology became "cognitive" after Claude Lévi-Strauss (and a number of American anthropologists) pointed out the need to put less stress on the organizational aspects of culture and more on the mental representations of the people living in it. Cognitive anthropology (or ethnosemantics) undertook the study of the naming, classifying and concept-forming abilities of people in remote cultures and tried to describe these linguistic and cognitive practices systematically in formal terms. In short, as Gardner (1985, 238) has it, "the Zeitgeist (of cognitivism) was assiduously at work".

Before I go into the central position of the relatively new field of artificial intelligence (and indirectly introduce neuroscience), a few remarks about the place of philosophy are in order as general background to cognitive science.

---

the Cognitive Revolution" (Gardner 1985), my main source for this subsection.

The first is that the tension between behaviorism and cognitivism discussed above runs closely parallel to the tension between rationalism and empiricism in philosophy. Rationalists believe that the mind has powers of reasoning which it imposes upon our sensory experience; empiricists believe that mental processes either reflect, or are constructed on the basis of, external sensory impressions. As such, it is understandable that behaviorists have clung to empiricism while cognitivists (and cognitive scientists in particular) are likely to lean towards rationalism or a mixture of rationalism and empiricism (cp. Gardner 1985, 53). The former may go as far back as Descartes (e.g. MIT scientists like Chomsky or Jerry Fodor), whereas the latter may refer to Kant with his synthesis of rationalism and empiricism. Another remark to be made is that the advent of computers and the field of artificial intelligence have inspired philosophers to reconsider age-old conundrums like the mind-body problem (or, in its modern version, the mind-brain problem) (see e.g. Putnam 1975b), and even engage in heated debates about them (see especially Searle's "Minds, brains and programs" in *The Behavioral and Brain Sciences* (Searle 1980, reprinted as Searle 1981) plus the many reactions to it). In the context of these discussions, new and interesting philosophical ideas and theories (such as "functionalism" (Putnam 1975b) and "intentionality" (Dennett 1978), both of which will return below) have been developed. They help cognitive scientists to remain aware of the assumptions and implications of their enterprise. Introducing AI and its relationship to cognitive psychology and neuroscience will raise some of these issues. It should not come as a surprise that I have greatly benefitted from a philosopher's overview of "the science of the mind" (Flanagan 1984) to take a position in matters of "dualism", "functionalism", "materialism", "reductionism" and other *-isms* a philosopher is very familiar with.

In the hexagon of Figure I I chose to put AI in the top position since I believe that without the computer, computer science and AI that grew out of it, cognitive science might have lacked the element and the discipline that pervaded and unified its subfields (cp. Haugeland 1984 who calls AI "first among equals" in cognitive science).

Considering that AI is a fairly new discipline (it is not much older than cognitive science itself), I will start by defining the field. The classical definition (e.g. in Gardner 1985, 140) is that AI "seeks to produce, on a computer, a pattern of output that would be considered intelligent if displayed by human beings." As far as the tools are concerned to reach this goal, some AI researchers concentrate on devising programming languages suited for the enterprise (Lisp, designed by McCarthy et al. at MIT in the early sixties, is still the most widely used language in AI), others on the development of formalisms to be used in the programs themselves, especially formalisms for representing knowledge (e.g. frames, networks, etc.; see e.g. Charniak & McDermott 1985 for an overview). Thus, programs have been written that "play" chess, "understand" language, simulate consultation of experts in medicine and numerous other fields ("expert systems"), etc.; one of the most recent research areas includes attempts to make computers learn ("machine learning"), i.e. transcend the knowledge programmed into them. Interesting though all these applications may be, I will not go into them but concentrate on two notions that have to do with the foundations of AI and are equally important for cognitive psychology and cognitive science in general.

The first notion is that of "functionalism", which refers to the position in cognitive psychology and cognitive science that "it is perfectly legitimate to talk of mental events, to posit one mental event as causing another, and to do so without taking a position on whether only brain events have the properties to define mental states" (Gardner 1985, 79). It will be clear that this is an article of faith among cognitivists since it is a position that makes an independent level of psychological analysis and explanation possible (cp. Flanagan 1984, 214-221). Although many a cognitive psychologist will indeed not take a position as to whether and how cognitive states relate to the brain, a cognitive scientist cannot afford to ignore these issues (remember that neuroscience is part of the hexagon). Moreover, the possibility of AI rests on a more radical interpretation of functionalism that *does* take a position on whether only brain events can define mental states. The assumption is that this need not be the case and that devices built of entirely different material than the human being (metal and plastic) can also realize intelligent functioning. Here again, the computer has strongly influenced our thinking about cognition; more precisely, it is the distinction between software (programs) and hardware (logic gates built with transistors; memory components, etc.) and the perfect legitimacy of dealing with matters of software without bothering with hardware at all that led philosophers like Putnam (1975b) to introduce functionalism. The philosophical conundrum that hovers over this discussion is of course that of the relation between mind and body. I consider the discussion about this matter to be important because one might accuse functionalists -- and I am one myself -- of reintroducing Cartesian dualism (simply stated, the position that the mind is independent of the body/brain) and hence, of making the link with neuroscience (one of the top three sciences in the hexagon !) problematic. Flanagan (1984, 221) shares my concern about this link when he states that "it would be dangerous (...) to entertain even for a moment the belief that research in brain science is irrelevant (...) , or to take metaphysical functionalism as proof that there are no interesting mappings of psychological processes onto brain processes." Luckily, functionalism can be combined with a form of materialism (again, simply stated, the position that mind and body are dependent on one another in that there is a correspondence between mental states and physical states). Flanagan (1984, 215-221) calls it "token-physicalism" (which I will call "token-materialism" to avoid the proliferation of *-isms*); it is the position that the mapping between mental states and physical states (of the brain) is a *weak* one. A token-materialist believes that each type of mental state maps onto a variety of physical states (tokens of the type), and not, as the type-materialist believes, onto a specific type of physical state. To paraphrase Flanagan's example: a functionalist who is also a token-materialist accepts that beliefs for or against God are physical events and processes, but he doubts that they are any one particular kind of physical event or process in all people. Part of the importance of this distinction is that it allows one to stay clear of the extreme position that mental states can be reduced to exactly pinpointable physical states (reductionism). Such a position (entailing statements such as "beliefs are just xzqry-neuron firings at velocity v and rate r in sector 2304" (cp. Flanagan 1984, 215)) can (and must) only be taken by someone believing in type-materialism with its precise one-to-one mapping of mental and physical states.

To summarize: I take a functionalist position, combined with token-materialism. It implies that I believe in a level of analysis and explanation that makes use of mental representations and processes but assures at the same time that a link with neuroscience remains possible; as such it avoids either of the two extreme positions of dualism and reductionism.

Since neuroscience makes its appearance here, a brief word about it. As far as the neural basis of language and the way it is processed is concerned, a number of fascinating discoveries have been made over the last two decades. They range from the more global localization of language faculties in the left hemisphere over localization of different kinds of aphasia (see Damasio & Geschwind 1984 and chapter 5) to "in vivo" research results indicating that listening to one's native language activates a different part of the brain than listening to a foreign language (6). Further dramatic progress can be expected as new techniques for brain research are developed: computerized tomography (since 1973) -- showing a sharp image of a cross-section of the living brain -- and techniques to trace how the brain uses oxygen and glucose during specific tasks. Although I take a token-materialist position here, I believe that in the long run neuroscience holds the promise (or the threat) of discoveries that may force cognitive scientists to revise or even abandon their functionalist theories and explanations. Neuroscience may some day take over the top position in the hexagon (cp. Gardner 1985, 286); in the meantime, however, carrying out psychological experiments and computer simulations of behavior remain necessary (see also chapter 5 for a critical position on a neuroscientifically inspired "connectionist" approach to NLU).

After this detour into neuroscience I return now to AI to look at a distinction that has been made between two types of AI; in light of the foregoing discussion it should be clear which of the two I accept and which I reject in order to retain internal consistency in my view of cognitive science.

In "weak AI", the computer is seen as a useful tool in the study of the mind; scientists write programs that simulate alleged psychological processes in humans in an attempt to test their hypotheses (and the predictions they entail) about these processes. "Strong AI" goes a lot further; here, "the computer is not merely a tool in the study of mind, rather the appropriately programmed computer really *is* a mind, in

---

(6) See the September 1985 issue of the "Monitor of the American Psychological Association" for an overview of recent brain research about language, learning, memory, certain diseases, etc.

the sense that computers given the right programs can be literally said to *understand* and have other cognitive states" (defined and criticized by Searle (1981, 353)). I prefer to adhere to weak or cautious AI (the NLU program presented in chapter 4 is used in the weak AI sense), because strong AI seems to push functionalism too far in the direction of dualism and may eventually turn its back irrevocably on the brain sciences (cp Flanagan 1984, 245). The adherence to the weaker version of AI does not exempt one from an exact characterization of the relationship between the computer simulation and the alleged processes though. I will go into this relationship when I confront the proposed NLU program with psycho- and neurolinguistic research in chapter 5.

The much debated equation of the human mind and the computer which is inherent in the strong view of AI brings me to the second of the two notions I started out to discuss ("functionalism" was the first). It is the notion of a "symbol processing system", central to computer science and AI; it forms the basis for the use of the computer as a metaphor for cognition in much of contemporary scientific thinking and, more radically, for the equation of cognition and computation as in strong AI (see also Pylyshyn (1980 and 1984), who tries to make the cognition-computation equation a necessary condition for cognitive science).

After the pioneers of computer science (especially Church and Turing) (7) had shown that any formal system (a set of symbols + syntactic rules for manipulating them) can be automated and as such handled by computer-like machines, Newell and Simon gave their far-reaching characterization of a "physical symbol system" in the early sixties (see Newell 1981 for the most recent description): it consists of a control unit, a memory, a set of operations, and input and output. The input consists of symbolic objects in certain memory locations; the processes regulated by the control unit are computational operations (re-creations, modifications, etc.) upon the input; the output of the system is in turn a modification or re-creation of symbolic objects in memory. A physical symbol system is seen as necessary and sufficient to carry out intelligent actions; and, conversely, any system that exhibits intelligence will prove upon analysis to be a

(7) See e.g. Hofstadter 1981, Gardner 1985, Flanagan 1984 for more precise and extensive treatments of Church's and Turing's work.

physical symbol system. In this (strong AI) view both the human being and the computer are instantiations of formal systems processing symbolic expressions (cp. Gardner 1985, 150).

There are a number of more or less related aspects and consequences of this characterization of a physical symbol system (or "information processing system", its more popular name) I want to mention and briefly discuss.

The first is that it brings in the notion of "process": symbolic objects are *processed*, i.e. manipulated by computational operations (as specified in computer programs) in a certain temporal order. Discussion of this notion is postponed till chapter 3 where I will go into its characteristics, the different kinds of processes, and of course the central role of the notion in process linguistics.

Another aspect of the physical symbol system notion is that it is almost by definition related to the classical computer and its components. The control unit is the CPU (Central Processing Unit), the input and output are instantiated in its "peripherals" (terminals, printers, etc.), it has a core memory strongly involved in the activities of the CPU, as well as extra peripheral storage facilities. It is exactly the more abstract characterization of an information-processing system (as given by Newell) combined with its concrete embodiment in the computer that has constituted the main metaphor for human cognitive functioning in cognitive psychology and even cognitive science in general (think for instance of the notions "short-term memory" and "long-term memory" and their close resemblance to core memory and peripheral memory in computers). It is also a metaphor that is being challenged more and more often because of its shortcomings vis-à-vis the changing view of human cognitive functioning (see especially Kolers & Smythe 1984, 301-302). In this context Gardner (1985) attaches great importance to what he calls "the computational paradox", the fact that working with the computer as a tool and metaphor for (the study of) human cognition has shown that human cognition is completely different from this same computer! Hence, once again, the dangers of adhering to strong AI with its equation of cognition and computation, also present in Newell & Simon's characterization of a physical symbol system as a necessary and sufficient condition for human *and* machine intelligence. It seems to me that, with the computational paradox as fundamental insight and turning point, cognitive science is entering a new era (a second generation?): whereas in the first generation the computer was used as a model for human cognition, the second generation reverses the roles and tries to model the computer after a number of hypotheses/discoveries made about human cognition during the first generation. Since the computer model presented in chapter 4 is an example of this beginning transition, it is important to take a look at the aspects of the physical symbol system considered insufficient and problematic in approaches of human cognition.

Two aspects I will go into in the course of this book are the fixedness of the memory components and the seriality in the way the symbols are processed: classical computers -- called "Von Neumann" computers after their spiritual father -- can only deal with one operation at the time whereas a human being processes e.g. different types of input (visual, linguistic, etc.) in parallel without apparent difficulty.

For the next problem with physical symbol systems I quote from Flanagan (1984, 223):

> "It is a truism that a formal system is meaningless until meaning is assigned to its elements. Until then a formal system is all syntax and no semantics. The sense of a formal system is provided by what is known as its "interpretation". The basic idea is this: manipulations of formal systems -- for example 1+1=2, or *p&q&r -> r*, or *f=ma -> f/m=a* -- are meaningless until we are told that numerals stand for numbers, that *'p'*, *'q'* can stand for any proposition in any natural language, that *'f'* means force, *'m'* means mass, and *'a'* means acceleration. The interpretation of a formal system is crucial if the system is to be about anything."

To linguists (especially those believing in a formalist approach to natural language) this is familiar ground. Here I only look at the notion "interpreted formalism" in the context of the view of human mental states as symbolic representations (implied in the view of mind as physical symbol system). The problem that arises is that of "intentionality", i.e. the property of certain mental states (e.g. beliefs, desires, expectations, intentions) to be directed at or about objects or states of affairs in the world (cp. Searle 1981, 358). As far as human intentionality is concerned, the problem is *how* it is that content gets assigned to symbolic mental representations that are so important to cognitive scientists. I offer no attempts at solving this problem (cp. the problem of how syntax and semantics can be tuned to each other in linguistics, and how it is that words are "somehow" related to phenomena in the world), but only suggest the seriousness of it by referring to two philosophers. Searle (1981) suggests that we do away with an approach to cognition that posits symbolic representations and only consider two levels of explanation, viz. the level of intentionality with a plain discussion of beliefs, wishes, expectations, etc. and the level of neuroscientific explanation of how the brain realizes intentional states (cp. Gardner 1985, 176). Fodor (1981), on the other hand, *does* believe in the necessity of a representational level but at the same time expresses a kind of agnosticism about how meaning gets assigned to abstract mental representations, and how content is dealt with by our computational systems. As far as computer intentionality is concerned (the other end of the stick): it is an ever-recurring ingredient of the debate about strong AI. Some attack the idea by stating that a computer is completely content-blind in that it merely manipulates formal symbols without "knowing" what it is doing, without "understanding" anything (needless to say that Searle is among them), whereas others (e.g. Dennett (1978)) do believe that we can talk about computer systems showing meaningful, purposeful action, hence as being intentional systems (8). Since this is again a statement fitting in with the strong version of AI I reject, I side with Searle on this last issue, but do not follow him in his abolishment of the representational level in the view of human cognition. If this level is abolished, it becomes very hard to accept the usefulness of computer simulation of such notions as "expectation" (which I consider very important in NLU, see chapters 3 and 4); since the simulation *does* use symbolic representations, it would become hopelessly problematic to relate the intentional notion and the way it is represented and dealt with computationally (the problem of the type of equivalence to assume between simulation and simulated behavior was mentioned earlier and will be discussed in chapter 5).

It is interesting to note, finally, that once the importance of intentionality (of mental states being *about* states of affairs in the world and carrying meaningful content) is

---

(8) Although the problem of human (and computer?) consciousness is certainly strongly related to what I only briefly touch upon here, going into it is beyond the scope of this introduction (see also note (9)).

recognized, two other (related) aspects of the information-processing metaphor for human cognition are challenged. The first is the view of the mind as a unified general-purpose device that performs all tasks in the same way, and that is equally competent across all domains (just like the computer with its CPU); it is usually combined with the "horizontal" view of human faculties. As Gardner (1985, 132) describes it: "On a horizontal view (...) faculties like learning, memory and perception are assumed to work in the same fashion, independent of whether the content is verbal, pictorial, musical, gustatory, or the like." Anderson (1983) holds this view and proposes a theory in which all higher-level cognitive functions can be explained by one set of principles. More and more researchers have grown skeptical of this "generalist" position, and take the "modularist" position. On this view, distinct cognitive principles are assumed to underlie the operations of distinct cognitive functions; the mind is seen as consisting of a number of modules, largely separate devices, including ones constructed to deal with language, visual processing, music, and other specific kinds of content. This skepticism about the need for some kind of CPU-like central processor is combined with a "vertical" view of human faculties: vertical faculties are computationally autonomous and deal in individual fashion with different contents. To give an example: on a horizontal view, one and the same memory faculty (controlled by a central processor) is deployed in memorizing Dutch vocabulary, telephone numbers or tastes of beer; on a vertical view, the mind has different modules for dealing with language, numbers or tastes, each with their own independent memory component. Among the defenders of this view are Chomsky (who calls the modules "mental organs" (1980, 3)), Fodor (who introduced the distinction between horizontal and vertical faculties (1983)), Gardner (who hypothesizes the existence of seven different, content-bound kinds of intelligence (1983)) and Dennett (who sees the mind as built up of intentional subsystems viewed as relatively ignorant, narrow-minded "homunculi" who form a team or committee that as a whole exhibits intelligent behavior (1978)) (9). Related to this matter of

(9) It will be clear that both the generalists and the modularists have their problems. A problem for the modularists, for instance is the question of whether there is no central processor at all, and -- if so -- what happens to human consciousness (cp. note (8))?

generality + horizontal faculties versus modularity + vertical faculties is the matter of the unified-code theory versus the six-code theory (cp. Flanagan 1984, 187-188). In a computer/physical symbol system there is only one code: all representation (in whatever computer language -- the software level) is abstract and quasi-linguistic; if, for instance, images have to be represented, they are translated into this abstract, propositional code so that their representation bears no direct resemblance whatsoever to the image or what it stands for in the world. According to the six-code theory, on the other hand, "our minds represent things in a total of six different ways. Five of these ways are tied to the sensory modalities (taste, vision, etc.); the other is abstract, propositional, and quasi-linguistic" (Flanagan 1984, 188). Hence, on this view, the abstract code is but one of the six and the others cannot be reduced to it. On a kind of intermediate position between these two extremes, one can say that the notion of "mental imagery" covers the five other codes. A lot of highly controversial and much debated research has been going on in this area (summarized in Flanagan 1984, 188-192 or Gardner 1985, chapter 11), but going into it here would lead me too far. Suffice it to say that an important aspect of this research is the attempt to show that (for certain tasks) people seem to mentally represent e.g. objects in an imagistic fashion and "manipulate" these images as if they were manipulating the real objects in space; this could be interpreted as evidence for at least an imagistic code beside the abstract propositional one. Thus, once again, a challenge to the physical symbol system notion as a metaphor for human cognition.

This concludes my critical discussion of the notion of a physical symbol or information-processing system central to AI and (with a great number of caveats) to cognitive science; it also concludes the tour of the disciplines, which I have used as "entry points" into the complex new field. To repeat: for me it combines *weak* AI, cognitive psychology that is critical of the information-processing metaphor of cognition, and neuroscience as a border discipline that may become very important in the future. Note that I do not consider anthropology in my view of cognitive science (see 1.3.3.4 for a short motivation); where and how linguistics fits in is the subject of 1.3.4 and of chapters 2 and 3. But let me first summarize five important characteristics of cognitive science as a distinct discipline by way of alternative and more systematic definition, with some implications for the methodology of the would-be new science (10).

1.3.3. Five key aspects of cognitive science

1.3.3.1. Multidisciplinarity

A good description of the necessity of multidisciplinarity (11) in cognitive science is given by Kintsch et al. (1984, ix-x):

> "Cognitive science is based on the belief that crossing the boundaries of the traditional disciplines is not merely possible, but indeed essential in the study of cognition. Without abandoning our own scientific identity, we must learn to take advantage of the results and insights obtained by researchers in other disciplines in order to progress more rapidly in the study of our exceedingly complex and difficult subject matter."

Hence, multidisciplinarity can be seen as a metascientific requirement: research results from any of the subfields (dealing with one's object of study) must not be ignored by the other fields. If predictions from one field are tested in one of the others and are proved to be correct, the hypothesis they were suggested by can be incorporated into the model they originated from; if they prove incorrect, the model has to be revised. In a science with many organically related research fields, this implies at the same time great vulnerability (constant need for model revision -- in a traditional perspective "coming from an outside field", in cognitive science perspective "coming from inside") and

(10) Gardner (1985, 38-45) also discerns five key features of cognitive science, but they do not completely overlap with mine.

(11) It will be clear that the multidisciplinarity described here goes further than the mere intersection of two of the subfields involved, as in already established fields like psycho- or neurolinguistics (of course, these subfields are by definition part of the cognitive science hexagon). This also explains why I prefer the term *multi*disciplinarity (suggesting more than two disciplines) to *inter*disciplinarity (suggesting two cooperating disciplines).

possibility of multiply confirmed hypotheses/predictions (i.e. model strength) (12). (In chapters 2 and 3 I will go into the consequences of this requirement for process linguistics.)

As a qualifying Kuhnian note to the possibility of multidisciplinarity, it should be said that the success of cognitive science will a.o. depend on the willingness of researchers to cross the boundaries between disciplines. Or, as Geschwind (1981, 30) flatly states it: "professional xenophobia and infatuation with one's own discipline are the greatest barriers to adaptation." In the meantime, an interesting aspect of attempts to achieve multidisciplinarity is that books bringing together work from different fields in a cognitive science perspective try to have their contributors make their articles understandable to nonspecialists from related fields (see e.g. Arbib et al. (1982, xv), Norman (1981, v), Hinton & Anderson (1981, vii) or Small et al. forthcoming). A to my mind very important consequence of this is the "dejargonization" of the participating disciplines: since using jargon is an ideal way to restrict legibility to insiders of a specific field, it is a practice incompatible with the multidisciplinarity goal of cognitive science. (Even if the new science itself creates its own jargon (which is probably inevitable) to replace the jargons of the subfields, it still implies an improvement to the scientific Tower of Babel...)

1.3.3.2. <u>Centrality of representation and process</u>

As a consequence of cognitivism (versus behaviorism) and functionalism (versus reductionism) cognitive scientists posit a separate level of analysis and explanation called "the representational level". It implies the use of representational entities such as symbols, schemas, rules or images and the way they are manipulated by cognitive processes (i.e. transformed, joined, contrasted, re-created, etc.) in an attempt to explain the variety of human behavior (plans, intentions, beliefs, actions, etc.). In accordance

(12) See the work of David Marr (1982) on the early phases of visual perception as an example of truly cognitive-scientific research in the sense described here; Marr tries to rigorously combine insights from perceptual psychology, neuroscience and artificial intelligence in his computational approach to vision.

with token-materialism this level is considered independent of though not unrelated to the neuroscientific level of explanation. As far as this book is concerned: the stress will be on processes (the dynamic aspect) rather than on the representations (the static aspect, fixed structures) they manipulate (see chapter 3 for an elaboration of this issue).

An interesting characterization of the way cognitive scientists work with representations and processes is Dennett's (1978, chapter 1) distinction between the "design" and "intentional" stances on the one hand, and the "physical" stance on the other. Researchers taking the design stance look at the way systems are made up of smaller functional subsystems or mechanisms. An approach to NLU from the design stance might look at the mechanism that encodes visual information (in reading), the mechanism that organizes the access of the mental lexicon, the mechanism responsible for syntactic comprehension, etc. The intentional stance "involves the use of ordinary mental concepts: belief, desire, hope, expectation, imagining, and the like" (Flanagan 1984, 179). From this stance, it is more important for an explanation of NLU to refer to e.g. the expectations the language user has. Cognitive scientists often mix these two styles of explanation; I will do the same when I consider the importance of expectations (be it as an intentional *process* rather than an intentional *state*, the way it is usually referred to) on all levels of language comprehension (syntactic, semantic, pragmatic,...). Unlike the design and intentional stances, the physical stance implies use of actual physical and chemical properties of the human organism in analysis and explanation; this is the stance taken in the brain sciences and is not our main concern here, as explained in 1.3.2. In the context of methodological matters, Dennett's distinction will prove very useful in the course of this book. In chapter 2 it will be used to criticize the way some researchers in computational linguistics sloppily mix the design and physical stances, and in chapter 5 I will use it to take my distance from a specific development in the computer model presented in chapter 4 (5.3.2).

1.3.3.3. Importance of computers

In 1.3.2 we have encountered a lot of traces of the pervasive influence of the computer (and the physical symbol system it instantiates) on (cognitive-)scientific thinking: the software-hardware distinction inspired philosophers dealing with the mind-body problem (with the important notion of functionalism as a result), the way the computer processes information has been (and still is) the dominating metaphor in the approach to human cognition -- be it a metaphor that is the target of a lot of criticism nowadays (cp. the computational paradox). But an aspect of this ingeniously conceived machine that no one will doubt about is its usefulness as a tool for scientists trying to understand our largely inaccessible cognitive functioning. Mandler (1984, 307) sees the metascientific implications of using computers in theory construction very clearly when he considers

> "the test of implementation (...) to be a useful tool for keeping social and psychological theorists honest. If their theories are so vague that their assumptions, axioms, and postulates cannot even be properly stated for possible implementation, or if their theoretical statements, once implemented, lead to internal contradictions and lacunae of indeterminacy, then the most advisable watchword would be: Back to the drawingboard!"

Needless to say that the same applies to linguists (see 1.3.4 and chapters 2 and 3)...

1.3.3.4. De-emphasis on affect, culture and history

In their stress on the mind and its functioning, cognitive scientists consider it a legitimate form of abstraction not to deal with human emotions and the way the human being participates in a certain culture and history. Hence, for one thing, the absence of anthropology (whether it stresses the individual within his culture -- the cognitive orientation -- or the culture around the individual) in my "cognitive science mix".

To linguists the whole issue of bracketing the way language is used in context is familiar enough; in chapter 2 I will discuss how generative linguistics seems to have gone too far in these matters through its competence-performance distinction. I only mention here that my critique concentrates more on the overemphasis on linguistic structure over cognitive process with respect to the *individual* language user; for an approach that criticizes the bracketing of language as used *in society* I refer to the well established research tradition of sociolinguistics (see e.g. Hymes 1974, Labov 1975).

1.3.3.5. <u>Two conflicting research poles</u>

Miller et al. (1984, 2-3) characterize the dual origin of cognitive science as follows:

> "Cognitive science, as it is practiced today, has two distinct historical roots. It derives from one scientific tradition that emphasizes objectivity and the study of behavior from the outside, and from another that is subjectively oriented and that has proposed to study mental life from the inside (...). The information processing tradition of cognitive psychology on the one hand, and action theory and purposive or intentional descriptions of behavior on the other, are representative examples of these incompatible trends within cognitive science."

Hence, a conflict between two points of view: cognition as mechanistic information-processing and cognition as intentional behavior. To Miller et al. "the problem for cognitive science is to find the right synthesis of these approaches", but they sound skeptical about the possibility of such a synthesis when they deplore the "confusion" between the languages of both traditions in cognitive science literature:

> "As if it were the most natural thing in the world, purposive terminology has been imported into an information-processing framework: subgoals are stored in short-term memory; unconscious expectations are processed in parallel; opinions are represented propositionally; the mind contains schemata. Is it a sign of conceptual weakness, or merely an excusable sloppiness in the use of language? Or is it no confusion at all, but a new synthesis that cognitive science has achieved?" (Miller et al. 1984, 6)

As a researcher who is optimistic about the cognitive science enterprise, I would simply answer "no" to the first and "yes" to the second question. Expressing a more differentiated opinion, I would say that the *awareness* of both research traditions and the way they have to complement each other is the sign of the complete cognitive scientist. *How* these traditions can be reconciled is one of the most challenging aspects of cognitive science, and it is AI with its methodology of computer simulation of purposive human behavior that inspires optimism about the success of the enterprise.

To conclude 1.3.3.5 (and 1.3.3) I present Figure II which brings together a number of notions that have been discussed in 1.3.2 and 1.3.3 and that can be grouped together around the two research poles of cognitive science. On the left, the mechanistic information-processing pole; on the right the intentional behavior pole (see also chapter 3 where a somewhat similar dichotomy will return when I oppose generative linguistics to process linguistics).

- formal, content-blind, context-free models	- stress on content, intentionality, context
- computer as model for human cognition	- skepsis about the computer metaphor
- unified-code theory	- six-code theory
- unity of mind (generalism)	- modularity of mind (modularism)
- horizontal faculties	- vertical faculties
- design stance more important than intentional stance	- intentional stance more important than design stance

Figure II. The two conflicting research traditions in cognitive science.

1.3.4. <u>Linguistics: ugly duckling</u>?

The presentation of cognitive science in the two preceding subsections may suggest that a smooth integration of its subfields is a fairly unproblematic matter, with the fruitful interactions of cognitive psychology and AI as the best example. This picture changes, however, when we look at the place of linguistics in cognitive science.

In so far as linguists study natural language they share a common interest with researchers in the other subfields of cognitive science, two of which I will focus on here: cognitive psychologists (which I take to include psycholinguists) are interested in the way human beings process language, and AI researchers are interested in simulating comprehension (and production) on computers. Yet, the views of researchers from one field on the contributions of the other fields to the general enterprise of understanding language and the way it is processed vary to such large extents that scientists have engaged in heated debates about the "contributions issue". In this subsection I mainly want to give an idea of how heated the debates are by quoting some pretty strong statements from them; in chapters 2 and 3 I will take my own position in the debate and critically discuss the issues involved in a more systematic way.

Already in the early days of cognitive science there were signs that linguistics (and more exactly mainstream, i.e. generative linguistics) seemed an unwelcome participant in the enterprise. The first issue of the journal *Cognitive Science* (January 1977) bore the subtitle "A multidisciplinary journal of artificial intelligence, psychology, and *language*" (my emphasis) (13). So one gets the impression that the object of study of linguistics was considered relevant, but not the discipline itself ! Rumor has it that the explanation must be sought in the fact that Roger Schank (an important AI figure and one of the then editors of the journal) could not get his (linguistic) theory of conceptual dependency (see e.g. Schank 1972) published in a linguistic journal and decided to start his own (with the bizarre but understandable subtitle...). Whatever the truth of the story, it shows that AI and linguistics are not the best of friends. Beside stories like these, an important background factor in the discussions (viz. in the United States) has certainly been the problem of research funding. As Lakoff expresses it: "With government funding sources running low and with a

(13) A light historical note: the journal has carried this subtitle until the first issue of 1985 when it simply became "A multidisciplinary journal" after the incorporation of another journal (*Cognition and Brain Theory*); an interesting aspect of this incorporation is that it seems to show the (growing) importance of the neurosciences in cognitive science.

decision by the Sloan Foundation to pour millions of dollars into Cognitive Science, the competition for research funding has been keen" (1978, 267; see also Gardner 1985, 35-38). Hence the attempts to stress the importance of one's own approach at the cost of someone else's.

The main question researchers disagree about is whether a linguistic theory (as developed in generative linguistics) can/must form the basis of a psychological and/or computer model of natural language processing. Generative linguists are very positive about the answer to this question:

> "(...) it seems that the development of an adequate theory of language use will depend on a firm characterization of linguistic knowledge, a grammar. One cannot build a theory of language use directly: the theory of language use will emerge out of a theory of competence, a theory of algorithms, a theory of implementation, and a theory of the proper mapping between these explanatory levels." (Berwick & Weinberg 1983, 53)

or, even more directly,

> "(...) linguistic grammars should form the abstract foundation of psychological parsing models" (Berwick & Weinberg 1985, 193)

Cognitive psychologists and AI researchers, for their part, sometimes express a firm *no* in answering the question:

> "In short, psycholinguistics should be interested in the grammars linguists develop so that we may describe observed speech adequately, but be wary of taking any grammar, especially structure-based grammar, as a model of performance. The primary function of performance is communicating semantic content, not producing grammatical structures." (Taylor 1976, 143; (14))

---

(14) See also Stabler (1983 and 1984), a strong opponent of the Berwick & Weinberg view.

> "From the perspective of artificial intelligence (AI), it is unlikely that a purely linguistic theory could be in any sense adequate. By a purely linguistic theory, we here mean a theory created to account solely for linguistic phenomena. The attempt to create such a theory is based on the presupposition that language can in some way be isolated from other elements of thought. But our successes and failures in trying to construct computational models capable of performing significant linguistic tasks seem to point in another direction: they indicate that language and thought are inextricably bound together." (Schank & Birnbaum 1984, 211)

But the game does not end here, since generative linguists go even further and express their doubts about the scientific nature of AI when it deals with natural language:

> "In this paper, we will show that (...) current work in AI does not in any way address the central questions that any scientific inquiry into language ought to address. Furthermore, we will argue that most of this work, though purporting to simulate aspects of human linguistic performance, is of virtually no psychological -- as opposed to technological -- interest because it is totally devoid of any principles which could serve as even a basis for a serious scientific theory of human linguistic behavior" (Dresher & Hornstein 1976, 322; (15))

No wonder then that researchers doubt the fruitfulness of interactions with other disciplines, and decide to retreat within their own territories. A group of generative linguists and a cognitive psychologist word their retreat as follows:

---

(15) The paper Dresher & Hornstein agressively set out to present in this quotation was (has been?) the start of the most heated debate between generative linguists and AI researchers in the short history of cognitive science. See Schank & Wilensky (1977) and Winograd (1977) for replies to the paper, and Dresher & Hornstein (1977a, 1977b) for replies to the replies. See also Lakoff 1978 or Berwick 1983 for contributions to the debate.

"In view of the fact that the packaging and public relations of much recent linguistic theory involves constant reference to questions of psychology (...), it is appropriate for us to make a few remarks about the connections between the claims we make and issues in the psychology of language. We make no claims, naturally enough, that our grammatical theory is *eo ipso* a psychological theory (...). Thus we feel it is possible, and arguably proper, for a linguist (*qua* linguist) to ignore matters of psychology" (Gazdar et al. 1985, 5; (16)).

"The proper task for the psycholinguist is not, at the moment, to determine the relationship between linguistic theory and psychological processes, but to try to acquire the kind of psychological processing data which will allow the construction of a genuinely psychological theory of sentence recognition." (Tyler 1980, 58)

In the context of the relationship between linguistics and psychology, this last quotation is also interesting in a historical perspective: whereas in the late 60s and the early 70s the "state of the art" in generative linguistics dominated psycholinguistic research (see e.g. Fodor et al. 1974 for an overview or Weimer 1974), psycholinguistics seems to have declared its independence and is conducting research without reference to linguistic theories (see some of the research discussed in chapter 5).

Beside these different views across disciplines, the divergent approaches *within* the field of linguistics are even more striking: there is the general opposition of "formalist" and "functionalist" approaches (see e.g. Bresnan & Kaplan versus Givón in Kintsch et al. 1984, 103-190), there is the rivalry between government-and-binding theory (e.g. Berwick & Weinberg 1984) on the one hand, and lexical-functional

---

(16) It should be noted though, that Gazdar et al. *do* go on to claim that their theory should have implications for psycholinguistics; whether psycholinguists will accept this type of interaction is of course another matter. See also Soames 1984 for a suggestion that psychology and linguistics should go separate ways.

grammar (Bresnan 1982) and generalized phrase structure grammar (Gazdar et al. 1985) on the other within generative linguistics, etc. In short, as Lakoff (1978, 274) makes the long list brief, "there are getting to be almost as many approaches to linguistics as there are linguists."

Thus, trying to be a linguist *and* a cognitive scientist looks like a very hard position to hold. In the rest of this book I will try to show that it is not impossible; moreover, that in spite of all the quarrels and debates across disciplines and of the diverging approaches within the field there is still a discipline called "linguistics" seems to show (paradoxically enough !) that there is reason for optimism about the cognitive science enterprise in general. To end chapter 1 gracefully with a hopeful quote expressing this optimism:

> "Looking at the diversity within established academic disciplines -- linguistics, in this case -- ought to reassure us. If linguistics can live with such differences as we have seen in the two chapters of this volume (i.e. a formalist versus a functionalist approach, see above (G.A.)), cognitive science can learn to expect and tolerate diversity, too. Eventually, of course, history will make its own choices, and the right way will be there for all to see" (Polson et al. 1984, 288).

CHAPTER 2 : WHY NOT GENERATIVE GRAMMAR FOR MODELS OF NATURAL LANGUAGE UNDERSTANDING ?

"I begin to wonder whether the whole field of linguistics has not lost its senses" (Derwing 1973, 6)

2.1. Introduction

At the end of chapter 1 I have tried to evoke the atmosphere that surrounds the way linguistics fits in with the other disciplines of cognitive science in the context of natural language processing. As announced, this chapter will be devoted to the issues involved. The discussion will take the form of a critique of generative linguistics (mainly the type of linguistics that has been done at MIT since Chomsky 1957 (e.g. Dresher & Hornstein 1976, Chomsky (1981, 1982), Berwick & Weinberg 1984) or that grew out of MIT (Bresnan 1978, 1982) (1). But why pick on generative linguistics if there seem to be a hundred ways of doing linguistics ? For one thing, it is (still) viewed as an important paradigm (if not the most important) within linguistics, with other approaches (often rejecting generative linguistics and its formalist approach) considered peripheral to this central paradigm (2). Now, since generative linguists in the Chomskyan tradition have always made strong claims about the necessity to incorporate a generative grammar as a basic and central component into a model of language use (the realization of a "competence" model into a "performance" model, see below), a linguist interested in NLU can hardly ignore these claims (especially if they come from a well-established

---

(1) See 2.2.1 for a further delineation of the type of generative linguistics taken under fire here.

(2) To name but a few: Langacker 1983, Hudson 1984, Dik 1978, Givón 1984, Gross 1984, Starosta 1978, Moore & Carling 1982. In chapter 3 I will come back to some of these approaches to show how they are related to process linguistics. I will also show that in spite of all the divergence both generative and non-generative approaches seem to converge on the importance of the lexicon, which also plays a central role in process linguistics.

paradigm). Moreover, these claims are often made within a cognitive science perspective (i.e. linguistic theories are put to the test by psycholinguistic experimentation with and computer simulation of language processing).

In this chapter I will try to show that there is no reason to take a generative grammar as a central component of a model of language understanding. The critique consists of two parts. First, I will show that there is an unbridgable gap between the notion of competence as defined in the Chomskyan tradition and the characteristics of human linguistic behavior in producing and understanding language (2.2). This part of the argument is certainly not new. It was the conclusion reached by a lot of critics of generative grammar as a basis of performance models in the early 70s (mostly psychologists) (3); they have gradually been joined by AI researchers showing the same skepticism about the realization of competence models (henceforth C-models) in performance models (P-models) (4). In the second part of the argument I take a closer look at the concrete attempts that have been made to supplement the claims with evidence (2.3). This part will take the form of a historical overview in which I concentrate on the most recent attempts to integrate C-models in P-models (Bresnan (1978, 1982) and Berwick & Weinberg (1983, 1984)). It will be shown that in spite of all the sophistication (borrowed from computer science) and elaborate argumentation these attempts cannot be considered successful.

Finally, in 2.4 I will draw a number of conclusions about how to do linguistics outside of or within cognitive science. In chapter 3 I will present process linguistics as an example of cognitive-scientific linguistics. Many of its notions follow from the points of critique formulated in this chapter, which implies that the critique is not gratuitous but supplemented by an alternative.

---

(3) See Fodor, Bever & Garrett 1974, Greene 1972, Levelt 1974[xxx], Anderson & Bower 1973; Derwing 1973, Parret 1974. More recent criticism can be found in Tyler 1980, Marslen-Wilson & Tyler 1980, Moore & Carling 1982, Kintsch 1984.

(4) See Wilks 1973 and passim, Schank 1975 and passim, Schank & Birnbaum 1984 (quoted in chapter 1), Winograd (1977, 1983).

2.2. The gap between generative grammar and NLU

2.2.1. Introduction

When asked to define the discipline of generative linguistics (or generative grammar -- the term is used in many senses, one of which is to denote the discipline), a researcher will nowadays be forced to ask in turn: whose generative grammar do you mean ? Within the Chomskyan paradigm (transformational generative grammar) theories have undergone rapid changes over the last twenty years: from standard theory (Chomsky 1965) to extended standard theory (Jackendoff 1972), to revised extended standard theory (Chomsky 1976) to government-and-binding theory (Chomsky (1981, 1982), further abbreviated to gb). In the meantime Bresnan (1978, 1982) left the paradigm and developed lexical-functional grammar (lfg). Finally, Gazdar et al. (1985) created their own brand of generative grammar (generalized phrase-structure grammar, gpsg) as a reaction against the Chomskyan paradigm (5). It will be clear that it is not my intention to go into the specific content of these theories with their resemblances and differences. I am more interested in the claims that are made concerning the mapping of their models to models of language use and the assumptions on which this mapping rests. Actually, this immediately reduces the applicability of the critique in 2.2 in that it does not apply to Gazdar et al. for the central issue of the competence-performance distinction. Recall the quotation in chapter 1 where Gazdar et al. take their distance from matters of psychology (6). And indeed, in their introduction (p. 1-16) the competence-performance distinction (central to the Chomskyan paradigm

---

(5) I restrict myself in this chapter to government-and-binding, lexical-functional grammar and generalized phrase-structure grammar (the three most important generative approaches). See e.g. Gazdar et al. (1985, 6) for other approaches considered generative.

(6) Actually, though in the 1985 book they take this non-psychological position, earlier statements about gpsg (in the context of the defense of an approach to natural language by using context-free grammars) did have a strong psychologizing character (see the discussion by Sampson (1983b)). I will come back to the argumentation involved in 3.3.2; here I only mention it for completeness' sake.

and to lfg) is not mentioned at all. I will come back to the status of gpsg and its relationship to cognitive science in 2.4.

Let me now briefly characterize the field in a way that covers the Chomskyan tradition *and* lfg that grew out of it; some aspects will be treated in greater detail when the discussion necessitates this. One of the basic assumptions of generative linguistics is that natural languages can be compared to formal languages and studied with notions from formal language theory. A language is seen as an infinite set of sentences that can be characterized or defined ("generated") by a finite set of basic elements and rules, i.e. by a generative grammar (in a more narrow sense). This system of rules determines which sentences are in the language (i.e. grammatical) and which are not (i.e. ungrammatical). (In so far as most of the rules considered deal with the syntax of languages, the stress is on this aspect and not so much on semantics or pragmatics.) Rather than dealing with the rules for specific languages, the generative linguist concentrates on general principles that constrain possible grammars (and as such possible languages). The set of these principles/constraints is called "universal grammar" (a third sense of "grammar"). All this shows the importance of formal language theory, an importance that is also stressed by Gazdar et al. (see the quotation below). The whole approach is given a mentalistic and psychological flavor (this is where Gazdar et al. part company with gb and lfg) by the stress on the notions of competence and language acquisition. An ideal speaker-hearer is assumed to possess the principles of universal grammar as an innate endowment allowing him to learn the rules of his language, i.e. to internalize a generative grammar; both the universal grammar and the particular rule-system are considered part of the ideal speaker-hearer's "competence", his knowledge of language. This competence allows him to produce and understand an infinite number of sentences and accounts for his "rule-governed creativity" (see further below). By looking at these mentalistic aspects of generative grammar, its goals can alternatively be stated as characterizing the competence of the ideal speaker-hearer, or -- and this receives most of the stress -- as explaining how people are able to learn their language. It is the observed capacity of a child to learn language pretty quickly in the face of a variety of data that leads generativists to posit innate principles (the universal grammar) and to concentrate on giving an account of these.

With this brief characterization in mind, the important point I want to make in this chapter (and in the alternative approach sketched in the next) can be restated more explicitly.

The critique is aimed at refuting the claims that a generative approach is a necessary and sufficient condition for "processual psychological adequacy", i.e. for smooth integration of a generative characterization of language and the processes of production, understanding and learning. (In this book I mainly consider the understanding process.) Hence, the need for an alternative approach that *does* allow for such an integration.

But the claims about processual adequacy are not the whole generative story. As said above, in the context of language acquisition there are also claims about "structural psychological adequacy". By this I refer to the idea that the presence of universal characteristics of language structure explains the ease of acquisition. The universals are given psychological reality by the statement that they are innate. This part of the psychological claims is not the focus of attention here (indirectly because language learning is not in focus), but I will briefly expose a view on these matters that is consistent with the ideas of this book (especially chapter 3).

First, there seems to be no urgent reason to deal with the language universals (such as *subjacency*, further discussed in 2.3.4) studied by generativists, since their truly universal status has as yet insufficiently been shown. Empirical research into specific languages constantly calls for "de-universalization" or revision of the universals. But, granting for a moment that the universal characteristics or constraints on grammars/languages are real (also psychologically), it would be necessary for any approach to language to incorporate them somehow. Yet, in that case the self-explanatory nature of the universals (they are *there*, innate, psychologically real by virtue of their mere existence) is very unsatisfactory. As will be explained in chapter 3 (3.2), if we take a process view of language it should be possible to have a richer explanatory model by trying to show that the structural universals are actually just epiphenomena of the characteristics of the cognitive *processes* responsible for our linguistic abilities (processes that can in turn ultimately be explained in neurophysiological terms, cp. 1.3). I believe that in order to show the correctness of this hypothesis it is necessary to look at language from the process view first (see further chapter 3), with a better chance of success in achieving truly psychological adequacy (and a truly cognitive-scientific account). And in fact, as we will see in 2.3.4, even generativists have recently tried to give a reductionistic account of universals by showing that they necessarily follow from the characteristics of the processing (viz. parsing) mechanisms. In short, trying to derive universals of language structure from characteristics of the processes of verbal behavior would allow for a much richer explanatory model than merely positing the existence of these universals as "innate axioms of grammar". And in order to achieve this, a logical approach is to look at the processes first.

By way of transition to the critique of the claims of processual adequacy of the generative model, I want to introduce a notion that is closely linked to what has just been discussed. We will see that in the generative tradition attempts have often been made to rationalize purely theory-internal notions by trying to give them theory-external importance (especially by psychologizing them, nowadays often in a cognitive science perspective). Examples further dealt with below are "creativity", "reliability" and "judgments about grammaticality". I will call these attempts instances of the *closed level fallacy.* An additional danger of this type of reasoning is also that when the theory-external phenomena are just theory-internal phenomena "in disguise", the closed level fallacy has an aspect of circularity to it. Maybe this danger can be avoided if one attempts a truly reductionistic approach, introducing an extra level in one's methodology as suggested above (7).

2.2.2. *Competence* and related notions

2.2.2.1. Generative grammar: idealized P-model?

The quotation from Chomsky's *Aspects of a Theory of Syntax* that one can usually find in a discussion of "competence", "performance" and the relationship between both is the following:

---

(7) See also Clark & Malt (1984, 191-214), a brief but clarifying commentary by psychologists on the linguistic trade.

"No doubt, a reasonable model of language use will incorporate, as a basic component, the generative grammar that expresses the speaker-hearer's knowledge of the language; but this generative grammar does not, in itself, prescribe the character of a perceptual model or a model of speech production" (1965, 9).

"Competence" is the speaker-hearer's linguistic knowledge characterized by a generative grammar; it is the basis or central component of "performance", i.e. language use (production and comprehension).

Before I analyze this central component view of competence more carefully (2.2.2.2) I want to consider the caveat ("but...") first. As Derwing (1973, 259-270) points out, it seems to have been inspired by a misinterpretation of the distinction suggested by Chomsky himself in other texts, viz. the view of competence as being in itself an idealized model of linguistic performance. In this sense, competence is seen as the *ability* to produce and understand an infinite number of sentences (the creativity of the language user); the idealization lies in abstracting away from shifts of attention, hesitations, distractions, errors and the like. Consider the following statements:

1) "A grammar, in the traditional view, is an account of competence. It describes and attempts to account for the ability of a speaker to understand an arbitrary sentence of his language and to produce an appropriate sentence on a given occasion. If it is a pedagogic grammar, it attempts to provide the student with this ability; if a linguistic grammar, it aims to discover and exhibit the mechanisms that make this achievement possible" (Chomsky 1966, 3).

2) "The most striking aspect of linguistic competence is what we may call the 'creativity of language', that is, the speaker's ability to produce new sentences, sentences that are immediately understood by other speakers although they bear no physical resemblance to sentences which are 'familiar'. The fundamental importance of this creative aspect of normal language use has been recognized since the seventeenth century at least" (Chomsky 1966, 4).

Although Chomsky clearly takes a distance from this interpretation in *Aspects* (the central component interpretation given there is also the one advocated by researchers in the Chomskyan tradition and by Bresnan), there are two reasons why I want to go into it still. The first is that my own redefinition of competence as processing competence (3.3.3) resembles the interpretation of competence suggested by 1) and 2) in that it stresses the *ability* more than the grammar. Although processing competence is defined in a completely different context than generative grammar, it is interesting to look at the arguments that make the ability interpretation of Chomsky's notion impossible since they suggest what such an "ability competence" could (should) look like. The second reason (related to the first) is that the interpretation suggested by 1) and 2) has been a source of confusion (by linguists and psycholinguists) about the *direct* usefulness of a generative grammar as a model of language processing. I want to go into the reason why this confusion could occur since it inspires a warning about mixing notions from formal language theory and notions from psychology (cp. 1.3.3.5 and 2.4).

Because I will need some of the notions of the Chomskyan brand of generative grammar (transformational generative grammar, tgg), let me first describe it in a little more detail (without going into the many changes the model has undergone) (8). A transformational theory of language contains three components (syntactic, semantic and phonological) that operate on two types of syntactic structure (deep structure and surface structure). The deep structure is an abstract underlying form of the actual sentence (the surface structure). It is generated by a base component consisting of a lexicon and context-free phrase structure rules, and transformed into its surface structure by transformational rules (nowadays just one rule, "move-α", with α an arbitrary phrasal category). The semantic component contains rules for interpreting the deep structure semantically, and the phonological component interprets the surface structure to give it a phonetic interpretation. The focus of attention has always been on the syntactic component studied independently of the other components and dealing with structures independent of their semantic and/or phonological interpretations (see further below for a discussion of this "autonomy of syntax thesis"). As said above, this model is taken to characterize

(8) This description does not apply to Bresnan's lfg (see 2.3.4).

the speaker-hearer's competence, his knowledge of language.

Coming back now to the interpretation of competence as an idealized model of performance (the ability interpretation), some of the reasons why a generative grammar fails under such an interpretation will be clear from the characterization above. According to quotation 1) the model should account for the *mechanisms* that allow a language user to produce appropriate sentences on given occasions and to understand any sentence of his language. However, the model with its structure-manipulating rule components nowhere says anything about the processes involved in actual language use. As far as production is concerned, the model is certainly suggestive by its "generative" character (though "to generate" in its formal sense and "to produce" as linguistic behavior are totally unrelated notions). Yet, nothing is said about the way a language user comes to produce an occasion-appropriate sentence; the syntactic component

> "enumerates the infinite set of sentoids in an order and in a way that must be considered essentially random from the viewpoint of actual speech production and comprehension. The phonological and semantic components cannot change this fact, because they are merely interpretative devices which assign interpretations to sentoids in whatever order those sentoids are given to them by the syntactic component" (Katz & Postal 1964, 166).

Hence generative grammars lack the feature of *selectivity* or non-randomness of production (cp. Derwing 1973, 267). (See e.g. Hymes' definition of "communicative competence" that does want to account for the choice of occasion-appropriate utterances (Hymes 1972)).

A second shortcoming in the context of language production concerns the account of "creativity" (stressed in quotation 2)) given by the model. This account (briefly mentioned on p. 31) considers the notion of recursion to be central to the creativity of language production. In the context of grammars with a finite set of basic elements and rules, recursion is that quality of the rules that allows them to generate an infinite set of sentences (9). It is clear that this type of

---

(9) If a rule (e.g. S -> aSb) contains the same symbol on the left and the right hand side (which makes the symbol a recursive symbol and the rule a recursive one) it can gen-

creativity is more a property of rules and as such of a grammar, not of a language user (cp. Parret 1974, 324-325). It makes more sense to relate the creativity of the language user to his creativity of thought based to a large extent on the infinitely many things he experiences throughout his life. To say that creativity (of a rule system) is "rule-governed" (Chomsky 1965, 59) is tautological. Of course, this does not mean that creativity as described here can be expressed at will, but the rules about the way to express it do not restrict creativity itself. The notion of recursion is also problematic when it comes to models of the language user. For Chomsky it was indispensable within a formal account of natural language (Chomsky 1963); however, human beings are hardly capable of understanding recursive (especially center-embedded) sentences like "The dog the cat the mouse hates likes went out" because of the limited memory we have and the incapacity to interrupt understanding mechanisms repeatedly. The solution within a formal language account: endow the *ideal* speaker-hearer with an unlimited memory capacity (part of his competence) and the formal notions of recursion and creativity can be retained. This explains why for Chomsky "distractions, shifts of attention, errors and hesitations" are on the same level as factors to be abstracted away from in competence as "memory limitations": the theory of generative grammar needs it (cp. Greene 1972, chapter 4). Yet, it is not necessary to assume that languages are recursive (as Chomsky suggests himself nowadays (see note (9))) to account for creativity. Moreover, if one takes limited memory capacity to be a fundamental constraint of human cognition (as I do in the context of processing competence, see 3.3.3) the plausibility of recursion is strongly reduced: if we cannot handle it mentally, why assume that it is a property of natural language (10)? Note, in the

---

erate infinitely many sentences when combined with other rules (e.g. S -> ab). The type of recursion exemplified by the two rules given here is the type that played a central role in Chomsky's discussion of what types of grammars can be used for generating natural languages. I will not go into Chomsky's argumentation here (see Chomsky 1963), the more so as Chomsky nowadays considers the notion to be much less important than in his earlier work (Chomsky 1980b, 119-123).

(10) See also 2.3.4 for a similar discussion of "creativity" and "finite capacity" as revived in Bresnan 1982.

passing, that this treatment of creativity (and recursion with it) is a first instance of the closed level fallacy.

A last shortcoming of the competence model (in the context of language production) I want to discuss concerns the organization of the components of a generative grammar. As said above, the syntactic component generates a (syntactic) deep structure which is then interpreted semantically. (Hence the term "interpretative" semantics; I will consider its opponent within generative grammar right away.) The following quote states clearly enough how such a model fares as a performance model:

> "The semantic component only serves to interpret what the syntactic component generates. As a psychological model of the causal sequence by which a sentence is generated, this scenario is utterly ridiculous. It would claim that we first decide what utterance we are going to say and then decide what meaning we want to convey, which is surely just the wrong way around" (Anderson & Bower 1973, 113).

The interpretative view of semantics was challenged in the early 70s by the generative semanticists. They stated that the deep structure should itself not be syntactic in nature, but semantic. Thus a base component directly generates a semantic structure, which is then transformed into a surface structure; no semantic interpretation rules are involved. Although this theory is more appealing as a model of performance (viz. production), at the time it was put forward hardly any serious model of language production existed. Now that these models start to emerge, the generative semanticists have left the generative scene. Hence its usefulness in production models has never been tested. (Moreover, as a transformational theory it suffers from the same flaw as its competitor when it comes to models of language understanding, viz. the nonreversibility of transformations, discussed below.) In short, a generative grammar does not account for language production, as Chomsky himself stressed in his caveat above.

As far as comprehension is concerned, the idealized P-model interpretation of competence fares even worse. Here again, a generative grammar says nothing about the *processes* involved in language understanding. *If* one wants to grant a dynamic aspect to its rules (phrase-structure or transformational) then one can at the most say that they are unidirectional (cp. Derwing 1973, 269) and only suggestive of a production process. The rules themselves (as opposed to their interpretation by automata or (possibly) a human being (see 2.3.2)) say nothing about how they can be applied "in reverse". Whereas context-free phrase-structure rules can fairly easily be applied to syntactically parse the language they generate, the reversibility of transformations (i.e. to go from surface to deep structure) has proved to lead to insurmountable problems (11).

In short, to let Chomsky himself bury the interpretation of competence as an idealized model of performance:

> "To avoid what has been a continuing misunderstanding, it is perhaps worth while to reiterate that a generative grammar is not a model for a speaker or a hearer. It attempts to characterize in the most neutral possible terms the knowledge of the language that provides the basis for actual use of language by a speaker-hearer" (1965, 9).

As for the reasons why this misunderstanding could occur: beside the fact that Chomsky suggested it himself it is true that generative grammars have some kind of "deceptive" dynamic quality. I already discussed the suggestiveness of the notions of creativity and recursion. The notions "generate" and "produce" (both having a specific meaning within formal language theory, cp. e.g. Lewi et al. 1982, chapter 2) are even more suggestive for the process of language production by human beings (cp. Quillian 1968, 263-264). Further, transformations are seen as processes for manipulating structures (see 2.3.4 for the consequences of a psychological interpretation of these "processes"). Finally, though grammars as systems of rules are by themselves *not* algorithms (in the computer science sense of completely specified procedures for solving problems), the rules themselves suggest how they can be applied to "produce" or "understand" sentences from the language they generate. For problems other than language generation or recognition, e.g. determining a move in a game of chess, the rules of how the pieces can move about certainly do not suggest so directly what move will be made.

---

(11) This can be considered as one of the reasons why transformational grammar has come to downplay the importance of transformations and to reduce their number and power (cp. Berwick & Weinberg 1984, 17-34).

The importance I attach to not jumping from formal language theory to psychology will be considered again in 2.3.4 and 3.2 in the broader context of the easy equation of parsing algorithms (as implemented on computers) and understanding processes in the human being (recall the computer metaphor discussed in 1.3.2 and also the closed level fallacy). Let me run ahead of the story by saying that it will lead me to a view of linguistics, computational linguistics and cognitive science linguistics in which certain approaches are considered incommensurable (Kuhn 1962) and should be kept separate (see 2.4).

2.2.2.2. Gg: central component of an idealized P-model?

Introduction

To show that "competence" cannot be considered as an idealized model of performance did not require too close a look at the content given to the notion; it was enough to point out that it has nothing to say about the processes involved in linguistic behavior (as Chomsky notes himself). The canonical view of competence as a central component of an idealized performance model (let me further abbreviate it as the "competence hypothesis", the term used by Bresnan & Kaplan (1982, xvii) to refer to this interpretation) does call for a closer look at its content *and* at the way it is supposed to interact with the other ("peripheral") components of a P-model. In this section I will try to show that "competence" is given so many ill-defined meanings that it almost becomes an empty notion. Hence my doubts about its usefulness and about the claim that it should be central to performance. (The notion of processing competence defined in 3.3.3 will be the alternative.) Beside this more theoretical argument, there is the matter of practice. Since Chomsky's central-component claim there have been a number of attempts to realize a C-model in a P-model; these attempts will be discussed in 2.3. There I will show that beside the fact that no attempts are made to clarify the notion of competence the models are not convincing in their claim that a competence model as defined in the Chomskyan tradition should be the central component of a P-model.

Knowledge and intuitions

Competence, as we have seen, is the speaker-hearer's knowledge of his language. A generative grammar purports to be a description of this competence, and as such of the underlying system of rules that has been mastered by the speaker-hearer. The data used to build and evaluate the system of rules and principles are linguistic intuitions, i.e. introspective judgments about grammaticality, ambiguity, paraphrase, synonymy, etc. These intuitions themselves are also called the competence of the native speaker:

> "(...) the grammar is justified to the extent that it correctly describes its object, namely the linguistic intuition -- the tacit competence -- of the native speaker" (Chomsky 1965, 27).

Hence, 1) competence is knowledge, 2) competence is a system of rules, 3) competence is linguistic intuition. All this could reasonably be brought together by saying that linguistic intuitions are an important aspect of our knowledge and that they help us discover the system of rules and principles that is also in our competence. However, things are not so simple as they may seem. The speaker-hearer is said not to be aware of the rules of grammar and even to be incapable of becoming aware of them (Chomsky 1965, 8); his knowledge is called "tacit" (1965, 19 21 27), unconscious, not open to introspection. Yet the introspective judgments are also in the knowledge, and, moreover, they are the *data* for the theory of this knowledge. This ambiguity never seems to have been resolved, but attempts have been made to clarify the issue. Harman (1967) suggested a dichotomy that has become popular again in cognitive science: the distinction between knowing *that* (declarative knowledge of facts) and knowing *how* (procedural knowledge, a skill or ability -- like riding a bicycle). In this distinction only the first kind of knowledge would be open to introspection, whereas the second is not (by now an article of faith of many a cognitive psychologist). Chomsky rejected Harman's distinction though, suggesting that knowledge of language can neither be characterized as knowing that or knowing how, without clarifying the issue (Chomsky 1969). Derwing (1973, 251-258) suggests a distinction between knowledge1 -- the potentially overt knowledge referred to as native speaker intuitions (judgments about sentences) -- and knowledge2 -- the inaccessible knowledge of a grammar or knowledge of a language. However, he is not clear about whether there could be any relations between both types, suggesting rather that a speaker-hearer has nothing to say about the generative grammar (theory) that is supposed to underly his competence. The problems with the notion of knowledge even lead Derwing to "wonder whether the term knowledge is even appropriate" (1973, 253).

I think Derwing is right in suggesting that knowledge is not such a good term here. Linguistic competence (as I see it) is an ability, a skill, a set of processes (processing competence, see 3.3.3) we have at our disposal. These processes (with reference to generative grammar: these innate processing universals) are simply *there*; we use them but they are not accessible, we cannot bring them to consciousness (just like we cannot describe in any way how it is that we can ride a bicycle). In this regard, I would not call these processes "knowledge", because to me knowledge implies accessibility of what it is about. Thus, "knowing how" is simply "being capable of" and draws upon such elusive phenomena as processes. However, for linguistic behavior (as opposed to bicycle riding, for instance) we are lucky that there is *output* of the process (viz. of production), output we can describe or characterize by conscious reflection. This output is accessible, hence there is knowledge of it, of how it is structured. Formulating rules about the output is one way to express knowledge about it, but these rules need not bear any direct relationship to how language is processed. Someone may perfectly master his language, but be incapable of formulating a linguistic rule. As I see it, this is not a question of "the rules are there (in our heads, even innately so) but we cannot become aware of them" but simply of not being trained in dealing with languages as objects of study (in education, science,..). But is there no relationship at all then between the rules (we have knowledge about) and language processing (we have no knowledge about) ? There is, but a very indirect one. In 3.3.3 I will come back to this matter in more detail. Let me point out here that knowledge of rules can help normal linguistic processing when necessary (as in linguistic judgments, wordplay, conscious resolution of ambiguities). Processing competence can run in two different modes to make this possible: in normal mode no conscious effort to process language is required, whereas in "metamode" conscious knowledge (in the form of awareness of rules about language, for instance) is appealed to. The two modes can interact, with metamode helping normal mode if e.g. correct understanding requires this. (See also 2.3.2 and 3.3.3 for the issue of rules and how they are or are not "involved" in processing.)

So much for the difficulties with the notion of knowledge. But the matter of linguistic intuitions also deserves closer attention. Many linguists have deplored the use of intuitions as data for linguistics (instead of actually occurring spoken or written utterances) because of their arbitrariness and incompleteness, especially for the description of particular languages. To show that they are right and that there are many "holes" in generative characterizations of particular languages would force me to go into the content of generative grammars, which is not the purpose of this critique. I refer the reader to Gross' article on the failure of generative grammar where especially this matter of empirical correctness is dealt with in great detail (Gross 1979). More important here is that competence is equated with linguistic intuition (see the last quotation from Chomsky), implying that intuitions should be at the heart of performance according to the competence hypothesis. Levelt has criticized this aspect of the competence-performance distinction most thoroughly:

> "The theory of any kind of linguistic behavior, namely, metalinguistic judgment (see note (12), g.a.) on such things as grammaticality and paraphrase, would then as a whole be built into theories on other forms of linguistic behavior such as speaking and understanding (...). The priority, given in this way to the theory of linguistic intuitions, has no empirical basis whatsoever. On the contrary, if we wish to think in terms of primary and derived forms of verbal behavior, the speaking and the understanding of language fall precisely into the category of primary forms, while *metalinguistic judgments will be considered highly derived, artificial forms of linguistic behavior, which, moreover, are acquired late in development.* (...) We (...) do not know the psychological factors which determine the formation of such intuitions. It

would be foolish to make linguistic virtue of psychological necessity by concluding that these factors are unimportant simply because they are unknown, but this is precisely what is done when linguistic intuitions are made the key to linguistic competence" (Levelt 1974xxx, 5-6, my emphasis (12)).

Beside pointing out these general (psychological) problems with linguistic intuitions, Levelt also studied their unreliability (acknowledged by Chomsky (1965, 8) but not considered at all as a reason for doubting their usefulness in linguistics). He reports a little experiment in which fourteen sentences from the generative literature (some judged grammatical, others ungrammatical) were presented to twenty-four trained linguists. The results showed that sentences marked as ungrammatical by the authors had less than half as much chance of being judged ungrammatical by the linguists as those marked grammatical by the authors, which is the opposite of what one would expect (see Levelt 1974xxx, 14-21 for a full discussion). But in the end it is even the question whether intuitions (especially about grammaticality) play any decisive role at all in the construction of generative theories:

4 waren ambigue zinnen!

"A linguist trained in the transformational grammar of the type presented in *Syntactic Structures* (Chomsky 1957, g.a.) will judge the string *colorless green ideas sleep furiously* as grammatical (in the restricted syntactic sense of the word), although it is semantically abnormal. A linguist trained in the *Aspects* theory will find the same string ungrammatical, because (syntactic) lexical insertion rules have not been respected in its derivation. The linguist trained in generative seman-

(12) Note that for Levelt intuitive linguistic judgments are *meta*linguistic judgments (because they are linguistically expressed judgments about linguistic objects; see Levelt 1974xxx, 7-10). To avoid possible confusion, I point out here that my use of the term "*meta*mode of processing competence" (mentioned above and further explained in 3.3.3) to refer a.o. to linguistic intuitions and judgments implies a different use of *meta* than Levelt's, although intuitively there is certainly some similarity.

tics, on the other hand, will in turn judge the string as grammatical, because the selection restrictions which have been violated are purely semantic in nature. We see here that the same phenomenon is alternately called semantic and syntactic, independently of the form of the theory, and this in turn determines the nature of the criterion of judgment. In this regard, judgments can only confirm the theory. If we hold the convention that selection restrictions are semantic, *it is the theory which decides that* colorless green ideas sleep furiously *is syntactically correct. The judgment of the linguist adds nothing to this*" (Levelt 1974[xxx], 20, my emphasis in the last two sentences).

We find the same well-worn sentence in Gazdar et al. (1985, 10), and in the context of their particular theory the sentence is simply "claimed" to be grammatical. In short, it is the theory, the grammar that determines by virtue of its rules and principles which sentences are grammatical. Here again, formal language theory and psychological notions of introspection and intuition are mixed in a non-illuminating way; stating that *intuitions* (viz. about grammaticality) are central to the approach (rather than simply admitting that grammaticality is purely theory-internal) is another instance of the closed level fallacy. In the context of intuitions and grammaticality in generative linguistics, a final deplorable fact (especially about the search for constraints that exclude ungrammatical sentences) is that the ungrammatical sentences excluded by the constraints are themselves mere artifacts of the theory they are supposed to support. "Sentences" like

*John seems that feeding himself will be difficult.

uit context getrokken!

or

*What did John believe the claim Harry would like to eat ? (Berwick & Weinberg 1984, 155).

are important within a framework that hypothesizes certain locality constraints on the movement of constituents (i.c. subjacency, see e.g. Chomsky 1981), but they are simply nonsensical to "naive" language users. (In this regard, generative linguistics looks more like a theory of linguistic artifacts than of linguistic facts.) In short, this again raises the question of how a model concentrating on (judgments about) grammaticality, paraphrase, etc. can be at the center of a P-model. It is not clear to me how the gap between derived linguistic behavior (metalinguistic judgments) and primary behavior (speaking and understanding) could be bridged. Here again: processing competence underlying performance will concentrate on the abilities underlying primary behavior, with derived behavior considered much less important (see 3.3.3) (13). If according to these primary abilities sentences like those above are completely incomprehensible, they have no role to play whatsoever in a linguistic approach that focusses on these abilities.

The single representation hypothesis

Beside the problems with knowledge and intuitions for the competence hypothesis, there is the problem of the single representation hypothesis implied by it. Mainly on methodological grounds of simplicity and generality (discussed further below) it is assumed that the *same* stored knowledge structure of competence underlies *all* forms of linguistic behavior (speaking, understanding, learning) (cp. Bresnan & Kaplan 1982, xviii-xix). I will not deny that *some* form of stored linguistic knowledge is involved in performance, but that it should be the same for all behavior (and, moreover, a generative grammar) is not so certain. Hence, in response to the provocative challenge of Berwick & Weinberg (1983, 198) that

> "the most highly valued theory would be one that could account for all of the functional demands on language (parsing, learning, production) by a single, uniform representation scheme. If someone doesn't like the uniform representation story, then the burden of proof is on them to come up with evidence to counter it",

---

(13) I will also go into the notions of acceptability (a more useful notion from Chomsky 1965, but relegated to performance and not important to his definition of competence) and understandability, important notions in the context of processing competence.

here are some suggestions of that kind of evidence (cp. Clark & Malt 1984, 200-203). There are numerous examples that suggest that we can understand a lot more on all levels of language structure than we can produce. We understand a variety of accents with which our native language (and, for that matter, the foreign languages we know) is spoken, yet we cannot produce them without conscious effort. Moreover, we can understand archaic language or idiosyncrasies of certain writers without having productive control over the structures or words involved. In general, our recognition vocabulary seems much larger than our production vocabulary, etc. In short, "deficiencies in production lie in syntax, vocabulary, morphology, phonology, and semantics, suggesting that at all levels of language structure, the process of listening has access to more "knowledge" than does the process of speaking" (Clark & Malt 1984, 200). Of course, these observations can still be considered compatible with the single representation hypothesis. One can suppose that there is simply a body of knowledge people can access in understanding but not in speaking. One can even assume that there are different access routes to the same body of knowledge without giving up the single representation hypothesis (14). In contrast to these assumptions to save the single representation hypothesis, one can take a more radical view and

> "suppose that comprehension and production access *distinct* representations of linguistic knowledge, even though, in normal people, the two representations code much the same information and are closely coordinated: people use their comprehension system to monitor and adjust what they produce, bringing production into line with comprehension. *Under this view, the single-representation assumption is incorrect*" (Clark & Malt 1984, 200-201; my emphasis).

This is the view I take, a view that is of capital importance to process linguistics since it justifies a linguistic approach that only deals with one of the linguistic modalities (viz. understanding, not production or learning) (see 3.3.1). Clark & Malt also present evidence for the more

(14) See also chapter 5 for a discussion (c.q. rejection) of the idea of different access routes to function and content words, a hot topic in psycho- and neurolinguistic research.

radical view:

> "As evidence for the more radical view, consider phonology. The processes of hearing speech sounds -- all the acoustic, phonetic, and phonological processes that investigators of speech perception have learned so much about -- bear little resemblance at any level of abstraction to the processes of sound production -- planning phonetic sequences, creating articulatory programs, and executing these programs. The first involves the ear and theories of auditory perception, and the second, the mouth and tongue and theories of motor movements. The two processes appear to involve distinct parts of the cortex as well. All that theories of phonetic perception and phonetic production need have in common is that the phonemes identified in perception, when veridical, are the same phonemes the speaker intended to articulate. Even the intention to produce a phoneme, and the recognition of that intention, need not make reference to the same representation, as long as they are coordinated in some way. In any case, the language representations that the two processes make reference to in realizing and recognizing these intentions don't need to look alike" (1984, 201).

Another item of evidence comes from neuroscientific research: experiments with story understanding and re-telling showed that different parts of the brain are involved in understanding and (re)producing (Turkington 1985, 12). (Even understanding itself seems to activate different parts of the brain depending on the language involved (native or foreign) (ibid.; cp. 1.3.2).)

But what about the preferability of the single representation hypothesis on methodological grounds of simplicity ? Granting that simplicity is an epistemologically desirable feature of a theory, it should never be invoked as a substitute for empirical data. If there is evidence against the single representation hypothesis, then it must be revised or abandoned. Simplicity cannot be assumed to guarantee the correctness of a theory by itself. Once again, the closed level fallacy lurks in the background: by positing that simplicity (a theory-bound notion) is the ultimate evaluation measure for a theory (see e.g. Chomsky 1957, 55-56), the theory encapsulates itself and is shut off from relevant data. Saying that an element of the theory correctly describes and explains something because it is a simple and generally applicable element is then a theory-internal circular reasoning. See also Derwing (1973, chapter 7), Parret (1974, 327-329), Winograd (1977), Gross (1979), and Stabler (1984) on the issue of simplicity and its abuse in the context of theory justification and explanation.

The implications of this motivated rejection of the single representation hypothesis can be summarized as follows. Intuitively, nobody will deny that production and understanding are different processes and need separate treatment. Yet, one can assume that there is an invariant body of knowledge (a set of rules and/or principles, constraints, etc.) that is used in these processes (and in learning as well). This assumption supports the competence hypothesis. Now, if there are clear indications that the linguistic knowledge used in the different processes varies with these processes (however unappealing this is to scientists looking for simplicity and generality), it is clear that trying to discover an invariant body of knowledge *a priori* is a wrong-headed enterprise. Closely related to the rejection of the single representation hypothesis is a central thesis of this book that will frequently be repeated in different guises, viz. the *anteriority of process thesis*. If (static) knowledge *varies* with the (macro)process (understanding, production, learning) manipulating it, then that process is anterior to the knowledge structures. In a generative framework the linguistic structures are the rules and principles mentioned throughout the discussion and even the tree structures (deep and surface) they apply to (cp. Dresher & Hornstein (1976, 378); they are considered to be in the speaker-hearer's competence and the central object of research. This, together with the fact that processes are relegated to performance, implies an anteriority of structure over process in generative grammar. Needless to repeat that rejecting the single representation hypothesis and the anteriority of structure over process it implies entails a rejection of the competence hypothesis that encompasses these assumptions.

The autonomy of syntax thesis

Up to now the discussion has centered around more general theoretical problems with the competence hypothesis. Let me take a closer look now at how the linguistic knowledge is organized into subcomponents, concentrating on the syntactic and semantic components. Recall that in the main tradition of Chomskyan linguistics the syntactic component is the focus of attention: it is studied independently of the other components and deals with syntactic structures without reference to their semantic interpretation (assigned to the sentence after syntactic generation). This is the autonomy of syntax thesis. It is against this aspect of the competence hypothesis that psychologists and AI researchers have often rolled in the big guns, suggesting that the semantic component should dominate the syntactic one. However, as generativists have rightly pointed out, the autonomy of syntax thesis within the competence hypothesis does not necessarily imply the autonomy (and priority) of syntactic parsing (Dresher & Hornstein 1976, 331-332; Berwick & Weinberg 1983, 36-44) (15). It is suggested that a generative grammar can be built into a variety of P-models, even if they attach more importance to (or allow more freedom in) the interactions of the components than a generative grammar suggests. However, although this argument is often used in a theoretical defense of the competence hypothesis (and sounds plausible enough), practice shows that it is fallacious. The P-models that *do* incorporate a generative grammar (Bresnan 1982, Marcus 1980, Berwick & Weinberg 1984) also stick very closely to the autonomy (and priority) of syntactic parsing. Lip service is paid to a possible contribution of semantics when syntactic parsing runs into trouble, but nowhere is semantics allowed in when the parser is used in a defense of the correctness of the syntactic rules or principles according to which it operates. To the extent that these performance models copy the componential organization of the competence model quite

(15) Much of the criticism of the autonomy of syntax thesis is in fact directed against the interpretation of competence as an idealized model of performance, and not against the canonical interpretation (cp. the quotation from Anderson & Bower in 2.2.2.1). Hence the usefulness of clearly distinguishing both interpretations.

directly (see 2.3.3 for a more precise characterization of this copying), they also imply a return to the idealized performance interpretation of competence and fall prey to the strong criticism this interpretation is subject to (see 2.2.2.1).

At this point it is interesting to introduce two general models of language understanding proposed by psychologists because one of the models strongly adheres to the autonomy of syntax thesis, whereas the other rejects it. The way this other model works is perfectly compatible with process linguistics and with the computer model that simulates aspects of the approach, so a closer look at it is certainly worth while.

Forster's P-model is an example of a model that is inspired by the organization of a generative grammar (Forster 1979). Let me call it the autonomous component model. In Forster's model (both for spoken and written language understanding) the language processor consists of a linear chain of three separate and autonomous processing systems: a lexical processor locating the input elements in the lexicon, a syntactic processor assigning syntactic structures to the input and a semantic processor (called "message processor") building conceptual structures. (Note the correspondence between levels of processing and levels of linguistic description.) Thus, the input enters the lexical processor, whose output enters the syntactic processor, whose output in turn enters the message processor; no other communication among the processors is allowed. All three processors have access to a "language-oriented data storage", the lexicon (16).

The alternative model (Marslen-Wilson & Tyler 1980 for spoken language; Just & Carpenter 1980 for written language) is the interactive model (17). Rather than viewing knowledge

---

(16) Forster's complete model also includes a general problem solver (GPS), but I leave it out here since it has no special role to play in the analysis of linguistic stimuli and is not part of the language processor proper (1979, 32).

(17) I immediately mention here that *interactive* will be used in a broader and more technical sense than its more common computer-world meaning of "involving the user as an active participant in a program/system". Both meanings are related, though, in that in the everyday meaning of *interactive* the computer program/system and the user can be seen as two interacting "processes" (see further in the text).

as organized in neatly separated components and processors, it stresses the purposeful integration of knowledge of all kinds in the understanding process (implying multiple interactions of knowledge sources). Marslen-Wilson & Tyler's model of spoken language understanding (partly a critique of Forster's model) starts from the claim that a listener tries to fully interpret the input as he hears it, on a word-by-word basis. The processing (viz. recognition) of the words is directly influenced by the contextual environment in which they occur; this implies that lexical, structural (syntactic) and interpretative knowledge sources communicate and interact freely in an optimally efficient and accurate way during language comprehension, without any delays in availability of information. The same view of the comprehension process is held by Just & Carpenter in their model of written language understanding: their "immediacy assumption" posits that all knowledge sources in working and long-term memory aid undelayed interpretation of the words of a textual fragment as they are read. Both interactive models also stress that the words themselves are the primary information sources the language user has; thus, bottom-up (data-driven) processes triggered by the words are more important than the top-down (hypothesis-driven) ones that further aid interpretation.

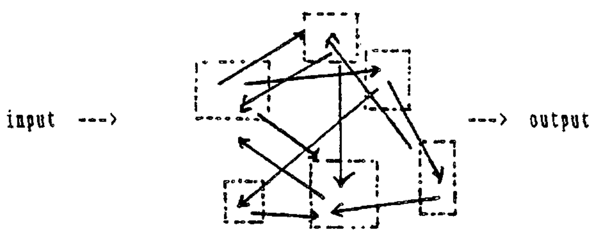
Both types of models are supported by experimental evidence; critically reviewing all the experiments involved is beyond the scope of this discussion though, so I cannot make a choice between both models on the basis of attested empirical correctness. Let me only note that the autonomous component model is often chosen for methodological reasons (Forster 1979; Berwick & Weinberg 1983, 39), viz. the possibility of concentrating on the structure and content of one component (a form of abstraction) rather than on the processes that "move to and fro" across component boundaries:

> "It seems to me that if we begin by postulating such a model (viz. the interactive one, g.a.), then there is very little hope of discovering interesting structural properties at all, and, consequently, we would be reduced to merely noting and cataloguing the kinds of problem-solving strategies that are (or can be) employed in various kinds of tasks. This may ultimately be the correct view to adopt, but it seems preferable to first thoroughly explore the alternatives to this view" (Forster 1979, 36).

Indeed, when we look at Figure I (an attempt to visualize the difference between the two models), it is understandable that researchers prefer the neat organization of the autonomous component model over the "messy" interactive model. (In the autonomous component model the (black) boxes dominate the picture, with the arrows showing the near-absence of interactions; in the interactive model the arrows - suggesting multiple interactions - dominate and the boxes disappear in the background.)

input ---> syntax ---> semantics ---> pragmatics ---> output

Autonomous component models



Interactive models

Figure I. Autonomous component models versus interactive models.

Yet, my choice of the interactive model is motivated by other reasons than methodological ones. First, there is its intuitive appeal: understanding seems indeed to be an efficient process in which all knowledge that helps understanding is brought to bear right away without delays. Further, there are more theoretical reasons. As I noted above, adherents to the autonomy of syntax thesis admit that knowledge of other kinds should influence the understanding process, but they leave it to others to deal with *where* and *when* exactly this other knowledge must enter the picture. It is a well-known fact to researchers dealing with natural language (understanding) that sometimes a syntactic analysis can simply not proceed without semantic information (18) (just like a semantic analysis may need syntactic information at times). Now, within the autonomous component model it is hard to accommodate these interactions, so they are often merely acknowledged or get swept under the rug. Needless to say that the interactive model is perfectly suited for dealing with the interactions (it *has* to deal with them). This has an important implication for process linguistics as an adherent to the interactive model. In the discussion of syntax-first versus semantics-first the "firstness" is less important and should depend on the concrete data to be analyzed (the bottom-up priority). Moreover, a middle ground position stating that we need both syntax and semantics is not enough. Important is when and where the interactions have to occur, and to look for systematicity in these interactions (rather than calling them "random" *a priori*, as Berwick & Weinberg do (1983, 39)). (In chapter 4 I will go into how the computer model supporting process linguistics tries to achieve this.)

Let me also point out that whereas the autonomous component model fits in with the autonomy of syntax thesis (and the competence hypothesis), the interactive model fits in with the anteriority of process thesis (interactions are more important than "structural properties" (Forster 1979, 81) of components and their content), which in turn is a corrolary of the redefinition of competence as processing competence (see 3.3.3). To drive the point home, I refer to Winograd here, who describes an approach to the study of language which he calls the "computational paradigm" (opposing it to the Chomskyan tradition in the study of language); in this approach -- to which I will return in 3.2 -- the centrality of process also implies a rejection of the autonomous component models:

---

(18) The correct interpretation of prepositional phrases is a well-known example. To analyze "I saw the dog with a black tail" versus "I saw the dog with a telescope" syntax alone cannot decide about whether the pp belongs together with the np in front of it (the first sentence) or whether it is by itself an adjunct of manner (the second sentence).

> "There is a strong basic belief that the best methodology for the study of language is to reduce the language facility to a set of largely independent "components", and assign different phenomena to each of them. This is in direct contrast to a system-centered approach which sees the phenomena as emerging from the interactions within a system of components. Much of the work in the computational paradigm has taken this more systemic viewpoint, emphasizing the mechanisms of interaction between components and concentrating on "process structures" -- those aspects of logical and temporal organization which cut across component boundaries" (Winograd 1977, 169).

In support of the anteriority of process thesis chapters 3 and 4 will show some examples of how structural properties of language (syntactic properties) can be seen as falling out of, or emerging from semantics-based interactive processes involved in language understanding.

Let me now summarize the criticism of the autonomy of syntax thesis. In the Chomskyan tradition it was suggested mainly as a methodological principle of abstraction in the study of competence: the syntactic component is central, the others are peripheral. Now, whereas it is claimed in the context of the competence hypothesis that the organization of the C-model does not need to bear a direct relation to the organization of the P-model, the models themselves show that the contrary is true. The same methodological preferences are the ultimate argument to justify this approach. The consequence is that we have a return to the interpretation of competence as an idealized model of performance, an interpretation subject to strong criticism and believed to have been abandoned by generativists themselves. The question arises whether it is at all possible to incorporate the C-model with its componential organization into a P-model that is not organized in the same componential way. Hence, the unclarity of what meaning has to be given to the claim that the competence model is the "central component" of the P-model. As we will see below, I believe this problem has led generativists interested in P-models to *weaken* the competence hypothesis and to suggest very indirect ways of realizing a C-model in a P-model (see 2.3.4).

Conclusion

Derwing's conclusion at the end of a similar argument against the competence hypothesis was that the C-model could only be seen as "an independent abstract entity remote from linguistic performance" (1973, 281). Parret reaches the same conclusion and suggests that the competence-performance dichotomy can in the end only be interpreted as the dichotomy of "grammar and linguistic reality or, even more barely, of theory and givenness" (1974, 331; my translation). Derwing goes on to plead for a cognitive science approach ("avant la lettre") to natural language:

> "(..) the first order of business for linguistic theory is the construction of tentative models of linguistic performance. It is a matter of indifference, really, who does the work, though I suspect that the best approach would be for linguists and psychologists to collaborate on the problem" (1973, 281).

I wholeheartedly agree with Derwing, and suggest we also let the AI researchers in on the enterprise. It was suggested above how psychological models (the interactive models) can form a good basis to start from, together with the anteriority of process thesis implied by the models and further developed in the next chapter in a cognitive-scientific linguistic perspective.

This concludes the theoretical part of the argument against the competence hypothesis. I will now give an overview of the attempts that have been made to incorporate generative grammars into performance models.

2.3. Generative grammars in P-models

2.3.1. Introduction

Before I give my overview of P-models incorporating generative grammars, some terminology has to be introduced. The possible kinds of mappings from C-models to P-models range from isomorphism over homomorphism to idiomorphism; these terms will be clarified in 2.3.3. In these mappings the notion of "rule" plays a very important role. Since rules are the object of some controversy in the context of models of language use, I will sketch this controversy first in 2.3.2, the more so as the status of rules in process linguistics is linked to this debate. 2.3.4 contains the overview itself.

2.3.2. The status of rules

To many scientists (and especially linguists, whatever their politico-linguistic conviction) the rule has always been the device *par excellence* to characterize or describe their object of study in an attempt to "capture generalizations" about it. Grammars, as already mentioned a number of times, are systems of rules and/or constraints on them. Whereas nobody will deny that rules are useful (if not indispensable) devices for the description of languages, problems arise when the status of these rules is considered in the models of language use that incorporate rule-system type descriptions of languages. The question that arises is: to what extent are the rules (central to the business of the linguist) (19) involved in the processing of language by humans ? In the Chomskyan tradition (see 2.2 and 2.3.4 below) the hypothesis is usually that the rules specified in the theory are mentally represented and used in the exercise of linguistic abilities. Hence, to say that linguistic behavior is "rule-governed" is not merely to say that it externally conforms to the rules of grammar, but that the rules are internalized, represented and causally engaged in linguistic processing (see especially Chomsky 1980a, p. 13, p. 54-55). As Stabler summarizes it: "the processing conforms to the rules [of grammar] *because* the rules are *encoded* ("represented") and *used* ("followed")" (1983, 396).

Yet, this view has recently become the target of criticism, especially by psychologists and AI researchers. Attempts are being made to respond to Chomsky's challenge that

(19) See 2.3.4 (Berwick & Weinberg 1984) for a discussion of the way the status of rules has become unclear in Chomskyan generative linguistics (viz. government-and-binding theory).

> "The critic's task is to show some fundamental flaw in principle or defect in execution, or to provide a different and preferable account of how it is that what speakers do is in accordance with certain rules -- an account that does not attribute to them a system of rules (rules which in fact appear to be beyond the level of consciousness)" (1980, 12).

Searle points out that the rules-are-used view is itself a hypothesis for which no hard evidence exists:

> "The claim that the agent is acting on rules involves more than simply the claim that the rules describe his behavior and predict future behavior. Additional evidence is required to show that they are rules the agent is actually following, and not mere hypotheses or generalizations that correctly describe his behavior; there must be some independent reason for supposing that the rules are functioning causally" (1980, 37).

In the "evidence" adduced for the hypothesis the reasoning usually goes along the following lines (cp. Stabler 1983, 396-398):

A) The elements of our theory (the rules of grammar and/or the constraints/principles of universal grammar) capture important generalizations about the structure of language.

B) The observation of human linguistic abilities (intuitive judgments, production, understanding, learning) shows that linguistic behavior respects the rules/principles of the theory.

C) Hence, that the rules/principles of the theory are used in linguistic behavior explains why the speaker-hearer's performance respects the rules/principles.

It will be clear that especially psychologists are not so happy about this type of reasoning. Granting that there *are* mentally encoded rules/principles, then "we can think of them as generalizations that are true of the *computations* or *operations* performed on the linguistic structures posited by the theory. We do not need to think of them as generalizations about the syntax or vocabulary of rules as they are encoded in the human sentence encoding-decoding mechanism or anywhere else" (Stabler 1983, 396). The reasoning above is one more instance of the closed level fallacy, and it is circular, as Clark & Malt (1984, 196-197) also point out. To Clark and Malt, linguists should be very careful in appealing to facts about language structure as an "explanation" for linguistic behavior (a psychological matter):

> "A feature found in all languages is prima facie evidence that there *may* be a psychological constraint leading to that feature, but the feature itself doesn't constitute the constraint. To claim that it does would be to fall prey to the fallacy *post hoc, ergo propter hoc*" (1984, 197).

After all, we should not forget either that i.c. rules only characterize the *output* of a process, which by no means implies that they should be directly related to the process that led to this output. In the context of the type of reasoning above, Clark & Malt make a distinction between weak and strong psychological constraints. One of the characteristics of a strong constraint is that it must not be derived from facts about the structure of language but from (empirically grounded) facts about processes in language use (the constraint of structure-independence (196-197)). Note that this constraint is just another way of stating the anteriority of process thesis (as opposed to the anteriority of linguistic structure adhered to by generativists). To Clark & Malt linguists can at the most be said to deal with weak psychological constraints, i.e. constraints that are

> "motivated not so much by psychological concerns -- by examining psychological theories to see how they might constrain grammar -- as by linguistic concerns -- by trying to rationalize the constraints that languages seem[..] obviously subject to" (1984, 205).

Chomsky himself seems to be aware of the structure-dependence of the psychological explanations in generative grammar: "Challenged to show that the constructions postulated in [our] theory have "psychological reality", we can do no more than repeat the evidence and the proposed explanations that involve these constructions" (1980, 191).

I will not go deeper into this debate about the explanatory value of structural characteristics of language for linguistic behavior; I side with Stabler and Clark & Malt in the insistence on the anteriority of process thesis. In this view, reasoning from structure to process is indeed fallacious and circular. To claim that linguistic constraints explain the behavior that conforms to them is another attempt at making "linguistic virtue of psychological necessity" (Levelt 1974[xxx], 6) by injustly extending the scope of linguistic theory to the domain of cognitive processing, a complex domain that we are only just beginning to explore.

Beside this general criticism on the "use" of linguistic rules in language processing, cognitive scientists have proposed performance models that account for rule-governed behavior with no reference to rules at all. Rather than resigning themselves to Demopoulos & Matthews' statement that "we as postbehaviorists know that explanations of behavior must advert to internal processes and we know of no other way of characterizing these processes except in terms of mentally represented rules" (1983, 406), researchers have developed "connectionist" or "interactive activation" models (20). These models reject the computer metaphor for cognition (for our discussion: the Von Neumann computer with its CPU manipulating rules in some programming language, cp. 1.3.2), advocating instead a radically different approach inspired by brain research. VanLehn gives the following characterization of connectionist models:

> "Connectionist models of cognition feature a network of nodes, whose typology is assumed to be relatively permanent (cp. the neuronal network of the brain, g.a.). Computation (i.e. thinking) is represented by fluctuations of the activation levels of nodes and by transmission of excitation and inhibition along connections. More elaborate formulations equip nodes with small state registers instead of activations, and connections pass small messages instead of an excitatory or inhibitory quantities (*sic*). The main architectural principles are (1) information transmission along connections happens in parallel, (2) there is little, if any global control (i.e. no central processor), and most importantly, (3) a cognitive model may use as many nodes and connections as it needs, but there are severe limitations on the amount of information stored in

(20) Feldman & Ballard 1982, Cottrell & Small 1983, Cottrell 1985, McClelland & Rumelhart (1981, 1982), to name but a few.

nodes or transmitted by connections" (1984, 74).

A hypothesis associated with connectionism is that for some tasks the best models are those that achieve rule-like behavior without rules, by using a large finite store of templates activated by the parallel process described above. Without going into the many details of the models and their computer simulation, let me give an (often mentioned) example. Rumelhart & McClelland (1981, 1982) used a connectionist model to account for research findings with the recognition of letters in words (using a store of about 1250 words, and *no rules*). The task given to subjects is the following. They are briefly shown four-letter words; after the stimulus word has disappeared the subjects are tested on a single letter in it. They must answer the question whether a certain letter occurred in a certain position of the word. Three important effects are observed in these experiments:

1) when the stimuli are English words (WORK, TRIP, CART), answers are correct about 17% more often than when they are non-words (XLQJ or ACUU)

2) when the stimuli are pseudowords (e.g. MAVE, SPET -- possible but non-occurring in English), answers are correct about 15% more often than they are with non-words

3) when the stimuli are consonant strings obtained by replacing the vowel in a four-letter word by a consonant (e.g. SPCT from SPOT, orthographically regular but unpronouncable), answers were again 15% more accurate than with non-word stimuli

Now, whereas 1) and 2) had been accounted for in other models, it is 3) that constituted a very important finding. The way the connectionist model works predicted this (counterintuitive !) result (not considered or accounted for in any theory based on stored orthographical or phonological rules) and it was *experimentally confirmed* (McClelland & Rumelhart 1982). Hence, the superiority of the connectionist model accounting for all the findings by its stored word templates and its unified process of interactive activation of these templates. It should be added here that these models have proved useful for simple tasks, but for complex behavior (like language understanding) their usefulness remains to be seen. (I have to come back to this issue in chapter 5 because a further development of the computer model presented in chapter 4 takes the connectionist direction, a direction whose success I am skeptical about (see 5.3.2).) Yet, they *do* show that we do not have to think there are no alternatives to rule-based approaches.

In 3.3.3 I will return to the status of rules in process linguistics; needless to say that their role in language understanding will be a minor one.

2.3.3. Possible mappings from C-models to P-models

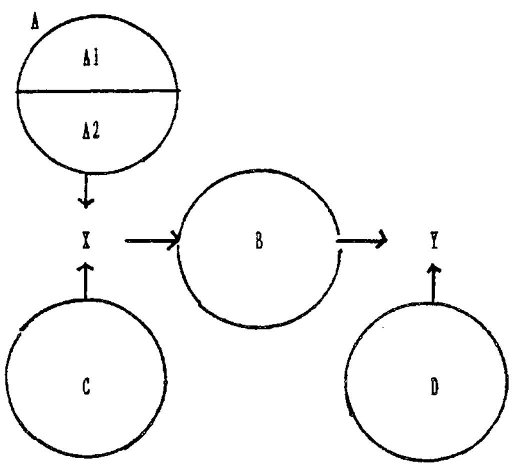


Figure II. A prototypical scheme of generative grammars.

If we take Figure II to represent a schematic prototypical generative grammar, we can say that it distinguishes four important types of elements. Globally speaking, there are components (A, B, C, D). These components contain rules (A1, B, C, D) and/or lexical elements (A2). Fourth, there are the syntactic tree structures (X, Y) generated and manipulated by the rules; with the P-model in focus, the most important structure is the one output by the parsing process. (In a variation on the model, X and Y might be a single structure, which would also alter the component buildup of the model). I suggest that for each of the types of elements we consider three types of mappings: isomorphism, homomorphism and idiomorphism (21). The first two terms have exact meanings

(21) Compare Levelt 1974[xxx], 68-73 or Berwick & Weinberg (1983, 1984) for similar but not identical distinctions.

in mathematics, but I only preserve the spirit of those meanings here; the third term is a neologism. We then get twelve theoretically possible mappings:

	isomorphism	homomorphism	idiomorphism
component	(1)	(2)	(3)
rule	(4)	(5)	(6)
lexical element	(7)	(8)	(9)
structures	(10)	(11)	(12)

Figure III. Possible mappings from C- to P-models.

By isomorphism I mean a strict one-to-one mapping between elements of the C-model (the grammar) and elements of the P-model (the parser); rule isomorphism ((4)), for instance, means that if the grammar contains a specific rule, the parser must have an operation that matches this rule. Homomorphism is a weaker type of mapping allowing for a number of possible gradations. It implies that distinctions made in the grammar are preserved in the parser, but the parser is allowed to make its own distinctions on top of those. Component homomorphism ((2)), for instance, will usually mean that the P-model contains a phonological, semantic and syntactic processing component (like the C-model) plus a pragmatic component. Rule homomorphism ((5)) similarly implies that one grammar rule may correspond to more than one parser operation; in a weaker sense, it can also be said to imply that rule *types* map to distinct types of operations (e.g. phonological versus semantic rules). (For structure-homomorphism -- which I will not deal with -- see Berwick & Weinberg 1984, 78-82.) Idiomorphism, finally, means the absence of any correspondence: C-model and P-model elements have their own characteristics (cp. the conclusion of 2.2). Structure idiomorphism ((12)), for instance, means that the (output) structure the parser builds bears no resemblance to the structure(s) distinguished in a C-model (most often, they will be semantic structures then rather than syntactic ones).

Instead of defining the twelve possibilities exactly at this point, I refer to the overview in 2.3.4 where some of the actually occurring ones are discussed. Mappings between components were already discussed in 2.2.2; let me repeat here that little is said about the overall component organization in most models below, because they concentrate on the syntactic component/processor.

2.3.4. A closer look at some realization attempts

In 1973 Derwing deplored the lip service paid to P-models by generative linguists as follows:

> "Most linguists seem satisfied with a schematic suggestion as to how a generative grammar *might in principle* be incorporated into a workable model of linguistic performance, and do not seem concerned with the question whether or not such a program can actually be carried out" (1973, 272).

And indeed, as we will see, the early models were very sketchy, incomplete or psychologically implausible. Since most of them have been abandoned nowadays, I will not spend too much time on them. However, to the extent that they form the point of departure (or are even revived) in more recent serious attempts at realizing C-models in P-models, they have to be briefly discussed. For fuller discussions of the early models I refer to Fodor, Bever & Garrett 1974 (henceforth FBG) and Levelt 1974[xxx], chapter 3; the recent attempts I will discuss here are Bresnan (1978, 1982), Marcus (1980) and Berwick & Weinberg (1983, 1984).

Miller & Chomsky 1963

In the early 60s the collaboration of George Miller and Chomsky gave a new impulse to psycholinguistic research. The competence hypothesis implied that grammars had to play an important role in verbal behavior. Hence, research into the psychological reality of the elements of generative grammars was enthousiastically undertaken. Miller & Chomsky's *Finitary Models of Language Users* (Miller & Chomsky 1963) was a seminal paper in this context. There are three elements in the article I want to single out:

1) Completely in the spirit of the strong belief in the usefulness of formal language and automata theory, it looks at how existing formal models relate to the human language user. The stress is on how the limited memory capacity of the user can be accounted for in those models. (This was discussed in 2.2.2; I will not go into it again here.)

2) There is a suggestion of how a P-model incorporating a C-model could look like.

3) It announces a theory that would be the focus of psycholinguistic research for the next ten years, viz. the Derivational Theory of Complexity.

Ad 2): the P-model Miller & Chomsky describe (1963, 465) is an improved version of the "analysis-by-synthesis" model. In such a parsing model, the grammar is used to generate a search space of candidate structural descriptions which are tested one by one against the input string. The comparison procedure halts when a match is found between the internally generated signal (sentence) and the input. The structural analysis of the input is determined by reference to the rules applied in generating the successful matching signal. The improvement consists in cutting down the size of the search space by using heuristics (smart shortcuts, cp 3.2). (The search space may be too large for any device to effect a match within reasonable time limits; hence the implausibility of the pure analysis-by-synthesis model.) In Miller & Chomsky's model the heuristics are in a separate component and allow the device to "guess" about the input (by preanalyzing it) and to gradually reduce the discrepancy between the guess and the input. Although the analysis-by-synthesis model has been abandoned nowadays, let me point out what it implies for Miller & Chomsky's approach. First, the literal embedding of the grammar whose rules are slavishly applied to recover the (grammar-defined) structural description of the input means that the model is rule and structure isomorphic (22). As far as the components are concerned, the

(22) Because in early transformational models lexical elements were also introduced by rules (such as N -> chair, table, dog, etc.), rule isomorphism also implies lexical element isomorphism here.

model is very vague except in its clear statement that the grammar and a limited memory space constitute the main component (actually the only one studied). Other suggested components are the ones needed for the analysis-by-synthesis procedure: a preanalysis component, a heuristic component, a comparator, "and perhaps others" (1963, 465). Moreover, there must be "components that reflect various semantic and situational constraints suggested by the context of the sentence" (ibid.). Although their importance is acknowledged, they are not studied "as an unfortunate consequence of limitations in our current knowledge and understanding" (466). In short, we have a vague component homomorphism with the autonomy of syntax principle simply extended to the P-model, a feature that -- as already suggested -- can still be found in the recent models too.

Beside the analysis-by-synthesis model, there is another rule and structure isomorphic model to be mentioned. FBG call it analysis-by-analysis, Levelt calls it the onion model (1974[xxx], 94). Here again, the rules of the grammar are literally applied, but "in reverse". The grammar is run backward, starting with the words, computing the intermediate representations in reverse, and terminating with the sentence symbol S (normally the start symbol of the grammar). This model has also been abandoned: whereas it is easy to run context-free rules backward (by a bottom-up recognition algorithm), it has proved very hard to run transformations backward. First, they are only defined over *trees*, not over strings of words. This means that a preanalysis has to determine the correct labeled bracketing of the string before we can tell whether a certain transformation can be applied to it in reverse. Here again (as with the analysis-by-synthesis model), a preanalyzer extraneous to the grammar has to do all the work. Beside the cognitive implausibility of several analyses performed serially on the input (we understand in one go, cp. Levelt 1974[xxx], 72), the question arises what role the grammar still plays if all the work is done by an extraneous preanalyzer. The second problem with running transformations in reverse (already mentioned in 2.2.2.1) is simply that they are not defined to be applied that way. They are unidirectional (from deep structure to surface structure and not the other way around). A specific example of the problems associated with this is the non-recoverability of deleted elements. If a transformation applied to a deep structure deletes something, it is impossible to tell from the surface structure whether or when this transformation was applied.

To come back to Miller & Chomsky now, both the analysis-by-synthesis and the analysis-by-analysis models for transformational grammars can be subsumed under the Derivational Theory of Complexity (point 3), henceforth DTC) suggested by Miller & Chomsky in the following quotation:

> "The psychological plausibility of a transformational model of the language user would be strengthened, of course, if it could be shown that our performance on tasks requiring an appreciation of the structure of transformed sentences is some function of the nature, number, and complexity of the grammatical transformations involved" (1963, 481).

Thus, the "Theory" holds that the complexity of a sentence is directly proportional to the number of grammatical rules (viz. transformations) employed in its derivation; this is rule isomorphism in its purest form. Researchers doing experiments to prove the correctness of the DTC did not worry too much about the processual problems with applying rules, but simply took it for granted that the rules *were* applied to sentences. Hence, for instance, tests to see whether sentences to which a number of transformations had been applied (according to the grammar, e.g. passive sentences) were harder to understand (i.e. took longer to respond to) than untransformed sentences (e.g. actives) or sentences to which fewer transformations had been applied.

Fodor, Bever & Garrett 1974

The by now well-known conclusion about the incorrectness of the DTC brings us to the next station in the overview, viz. FBG. Fodor, Bever and Garrett critically review the experiments done in relation to the DTC and conclude that it is untenable. I briefly summarize the main points in their argument (see FBG 1974, 320-328 for the details):

1) As one might intuitively expect, semantic factors proved to play a more important role in ease or difficulty of understanding sentences, whether they be "transformed"

(e.g. passives) or "untransformed". For example, it was found that only reversible passives (i.e. passives that make sense when their *agens* and *patiens* are interchanged, e.g. "The boy was hit by the girl") proved "more complex" than their active counterparts, whereas non-reversible passives ("The kite was flown by the child") did not. In the latter case it is clearly the semantic nature of the sentence (the verb and its semantic cases) that makes it no more difficult to understand than its active counterpart. Thus, the DTC predicting greater difficulties with *any* passive is refuted. Moreover, when we consider the complete transformational model (i.e. *with* its semantic component), it is not even possible to decide whether complexity is a matter of syntax (transformations) or semantics. Recall that the canonical deep structure is supposed to be the one that contains the grammatical relations among the parts of the sentence in the easiest form for semantic interpretation. Now, transformations "destroy" the canonical form, making semantic interpretation more difficult as well. Hence, complexity can just as well be a matter of semantic difficulty as implied by the transformational model itself.

2) Even more importantly, FBG showed that the DTC made a large number of wrong predictions about complexity. An example: in standard transformational grammar adjectives are transformationally derived from relative clauses; hence "The small cat is on the dirty mat" should be more complex than "The cat which is small is on the mat which is dirty"... Another example (from a slightly different angle): sometimes additional transformations make sentences *easier* to understand. "The shot fired by the soldier missed" seems easier than "The shot the soldier fired missed", thanks to its additional passive transformation.

Hence, FBG buried the DTC (and its associated analysis-by-synthesis and analysis-by-analysis recognition models). Their general conclusion:

"(..) experimental investigations of the psychological reality of grammatical rules, derivations, and operations -- in particular, investigations of DTC -- have generally proved equivocal. This argues against the occurrence of grammatical derivations in the computations involved in sentence recognition and hence against a concrete employment of the grammar by the sentence recognizer" (FBG 1974, 368).

Needless to say that this conclusion (i.e. the refutation of the competence hypothesis on the basis of psycholinguistic research) complements the refutation of the hypothesis on theoretical grounds (2.2.2.2).

Before having a look at how generative linguists have reacted to FBG's verdict, let me briefly mention the alternative model sketched by FBG. As a consequence of the rejection of the competence hypothesis, their model is component and rule idiomorphic. The sentence recognition system has a complex structure of its own to which grammatical knowledge only makes a small and indirect contribution. (This is exactly the position process linguistics takes, see 3.3.3.) The system consists of a number of heuristic strategies (23) whose task it is to recover the (transformational-generative) deep structure from the surface structure. Note that although FBG's model is component and rule idiomorphic, it remains structure isomorphic to the transformational C-model by retaining surface and deep structure as defined in tgg (24).

To summarize the overview so far: after early enthousiasm about the psychological reality of the elements of generative grammar (based on the competence hypothesis), more and more experimental evidence was adduced against the idea that a

---

(23) I will not go into FBG's heuristics here, because they are mainly aimed at recovering syntax, not meaning. In 3.3.3 and 3.3.4 meaning-searching heuristics as advocated by process linguistics will be discussed in greater detail. (See also Kimball 1973, Levelt 1974[xxx] or Clark & Clark 1977 for a description of possible heuristic strategies in comprehension.)

(24) In process linguistics structure isomorphism is rejected; the structure that is built as a result of the understanding process is not related to tgg. Hence, the model is idiomorphic on all levels. See 4.2.2.2 for a more precise characterization of the output structure produced by the computer model complementing process linguistics.

generative grammar is an isomorphically employed central component in performance. This was the main point of FBG, the most complete account of psychological reality research in the late 60s and early 70s. Now, one can at least imagine four ways to react to this verdict:

1) reject the competence hypothesis and build an idiomorphic model of performance (as FBG do, and as I do)

2) stick to the competence hypothesis, but revise transformational grammar to make it a more plausible central component of performance (Bresnan 1978, 1982)

3) stick to transformational grammar, but "revise" the competence hypothesis by rejecting the necessity of an isomorphic mapping (Berwick & Weinberg 1983, 1984)

4) try to show that the FBG conclusion was premature (Berwick & Weinberg 1983, 1984)

5) take no position in the competence hypothesis and do generative linguistics (transformational or not) without too many psychological claims (Gazdar et al. 1985)

Bresnan 1978, 1982

Bresnan (1978) accepts the FBG verdict about the psychological unreality of transformations but does not want to give up the competence hypothesis (she motivates this latter point in Bresnan & Kaplan 1982, see below):

> "If a given model of grammar cannot be successfully realized within a model of language use, it may be because it is psychologically unrealistic in significant respects and therefore inadequate in those respects as an empirical theory of the human faculty of language" (1978, 2).

Hence it has to be revised. A cognitive science perspective is chosen to back up the feasibility of the enterprise: "new developments in transformational linguistics, together with independent developments in computational linguistics and the psychology of language, make it feasible to begin to construct realistic grammars" (1978, 2-3). The model Bresnan proposes is the lexical-interpretive theory of transformational grammar (a precursor of lexical-functional grammar, see below): the transformational component of the grammar is drastically reduced and the lexicon and semantic component are greatly enlarged. Nontransformational rules (lexical and interpretive) play the main role. Bresnan's revision of tg is motivated both by theory-internal developments (25) and by a view of human cognitive capacity: the assumption is that it is easier for us to look up information (as in a large lexicon) than to compute information (as the application of transformations requires); the idea behind this is that if retrieval takes less time than computation, then the lexical-interpretive theory is in keeping with the rapidity of language comprehension.

In the 1978 article, the stress was on the revision of tg, and not so much on the way the lexical-interpretive theory could be integrated in a P-model. A brief note about the revision: I consider the shift in focus of attention to the lexicon a healthy development. The lexicon has always been neglected in generative grammar (see Gross 1979, Taylor 1980), which is comprehensible if the stress is on "capturing (syntactic) generalizations" and not on the many (real but troublesome) idiosyncrasies in natural languages, idiosyncrasies that can be traced back to the lexicon. This lexicon will play a central role in process linguistics, be it from a different perspective than Bresnan's (see 3.3.4). As to the integration of the new model in a P-model, it is interesting to look at the type of mapping Bresnan has in mind.

> "These realizations should map distinct grammatical rules and units into distinct processing operations and informational units in such a way that different rule types of the grammar are associated with different processing functions. If distinct grammatical rules were not distinguished in a psychological model under some realization mapping, the grammatical distinctions would not be "realized" in any form psychologically, and the grammar could not be said to represent the knowledge of the language user in any psychologically interesting sense" (1978, 3).

A first thing to note is that the statements about the

(25) I will not go into these; see Bresnan 1978 or Hoekstra et al. 1980.

realizability of C-models in P-models have become much more tentative than in the early days of transformational grammar. It is no longer taken for granted that there should be a very direct, isomorphic mapping. On the other hand, generativists are fully aware that the acceptability of the competence hypothesis depends on the strength of the mapping one has in mind: on a large scale, no isomorphism is proposed, but on a small scale (e.g. one specific grammatical phenomenon) anything that shows evidence for an isomorphic mapping is treated as very important to the theory. Note that with the competing theories in generative grammar, isomorphism is not merely important as "evidence" for the competence hypothesis, but even more important as evidence for the specific theory that can claim the strongest mapping. When we take a closer look at the quotation above, Bresnan fluctuates between isomorphism and homomorphism. She starts with "distinct rules" (implying rule isomorphism), but switches to "rule types" (implying rule homomorphism) in the very same sentence. Moreover, in the next sentence, she merely talks about "some realization mapping". The last sentence also reflects the importance attached to finding some mapping: of what use is a C-model that cannot be realized in a P-model? Important is also that Bresnan is the first to stress lexical element isomorphism (see the first sentence, "informational units"), which is understandable if the theory she has in mind considers the lexicon as a central component.

At the end of her article, Bresnan comes back to the realization problem, suggesting that the lexical-interpretive theory of grammar can be embedded in a syntactic pattern-recognition system like augmented transition networks (ATNs) (see Bresnan 1978, 50-58). Yet, it is clear that she is not so happy about this realization because it is only useful for the (small) phrase-structure rule component of lexical-interpretive grammar, and not for the much more important lexicon:

> "Indeed, the realization outlined here suggests one respect in which ATN systems may model linguistic comprehension inadequately: in recognizing sentences, transition network systems appear to make insufficient use of lexical information" (1978, 57).

In the 1982 book on lexical-functional grammar (Bresnan 1982) -- the further developed successor to lexical-interpretive grammar -- the discussion of the realization problem (Bresnan & Kaplan, Introduction) has become more extensive, and more work has been done on realizing lfg in acquisition, comprehension and production models. I restrict myself to the introduction, chapter 4 (where the grammar is explained in detail) and chapter 11 (where the realization of lfg in a comprehension model is proposed), the three parts that are the most relevant to the discussion here.

In the Introduction, Bresnan & Kaplan discuss the realization problem. The competence hypothesis is defended again (see above), by stressing that its associated single-representation hypothesis is preferable on methodological grounds of simplicity of explanation (see 2.2.2.2 or Clark & Malt 1984 for a critical reply to this defense). They go on to regret that Chomsky himself seems to have abandoned his strong conception of the psychological reality of grammars (which they adhere to) when he no longer requires "that we take responsibility not only for characterizing the abstract structure of the linguistic knowledge domain, but also for explaining how the formal properties of our proposed linguistic representations are related to the nature of the cognitive processes that derive and interpret them in actual language use and acquisition" (1982, xxii). Bresnan & Kaplan try to determine what criteria other than psychological reality *do* play a role in the development of tg then, and they reject them. Strangely enough, simplicity of explanation is among the rejected criteria; this is in contradiction with their own defense of the competence hypothesis on the same grounds.

> "(..)simplicity is itself a theory-bound notion; as Chomsky (..) has argued, the choice of a simplicity metric is made on the same empirical grounds as the choice of a theory. Moreover, it is easy to imagine even highly elegant and deductively satisfying rule systems that lack psychological reality in the sense we would like" (1982, xxii).

As I discussed in 2.2.2.2 the same reasoning applies just as well to the single-representation hypothesis. If the content of the competence component varies with the process involved, then the single-representation hypothesis (and the competence hypothesis with it) is false in spite of its simplicity (26).

---

(26) Let me note here that Chomsky's stress on idealization in science is also criticized by Bresnan & Kaplan; they

In short, the psychological reality of grammars is a strong criterion for their evaluation. Bresnan & Kaplan go deeper into this problem by trying to provide answers to two complementary questions:

1) which constraints on the representation of (linguistic) knowledge affect the processes that can manipulate them?

2) which constraints on processing linguistic knowledge affect the structure of grammars?

The answer to the first question comes down to a defense of lfg at the cost of the transformational theories it grew away from and some general suggestions of how an lfg could be embedded in P-models. Whereas the 1978 article was not so clear about how direct the mapping from C- to P-model had to be, the 1982 Introduction is more explicit. According to the "strong competence hypothesis" there is a component of stored linguistic knowledge that prescribes certain operations a processor is to perform in parsing (e.g. manipulating phrases); this component is called the representational basis of the processing model. The model satisfies the hypothesis if and only if its representational basis is isomorphic to the competence grammar. Now, it is clear that to them the competence grammar is an lfg, but it is not clear what they mean by their isomorphism. Consider their more detailed description:

> "(The representational basis is the "internal grammar" of the model.) Since not all components of the internal grammar are necessarily utilized in every linguistic behavior, we do not require all information in the representational basis to be interpreted by every processing model. However, we do require that every rule of the representational basis be interpreted in a model of *some* behavior; thus, the internal grammar cannot contain completely otiose rules" (xxxi).

---

call it "suspicious" and unmask it as a means "used mainly to restrict the kind of evidence that may be brought to bear on representational issues" (xxiii) (cp. the closed level fallacy). The evidence they have in mind is especially psycholinguistic evidence (but see Berwick & Weinberg 1984 for a reply to this critique).

Under such a characterization of an "isomorphic" mapping, they should at least make clear what components are involved in what behavior; moreover, that every rule *must* be used in *some* process is more easily stated than shown. In short, to call this an isomorphic mapping is mere wishful thinking. They can at the most say that the mapping is homomorphic: *some* components and *some* rules are used in *some* type of linguistic behavior. To the extent that the mapping can only be interpreted as much weaker than claimed, the "strong competence hypothesis" only gets weak support as well (see also below).

In answer to the second question, Bresnan & Kaplan propose five psychological constraints that should restrict the forms of grammars. The first two are creativity and finite capacity (familiar from Chomsky 1965). Creativity implies that the grammar must be capable of generating an infinite number of strings. Finite capacity means that 1) the means to generate sentences are finite (words + syntactic relations) and 2) people's mental capacity for storing knowledge must be finite. As was already discussed in 2.2.2.2, creativity and part 1) of finite capacity are mere theory-internal or structure-dependent principles and not psychological constraints (cp. Clark & Malt 1984, 202). Part 2) of finite capacity is a true psychological constraint, but the effect it is allowed to have on the theory of grammar is minimal. The theory *assumes* that the set of words and of grammatical relations is finite; finite mental capacity is only invoked postfactum to "constrain" the theory here. On the other hand, finite mental capacity is not allowed to constrain e.g. depth of recursion in languages, since recursion is too important to the theory (it is needed to account for creativity). The third constraint ("reliability") states that syntactic analysis of sentences must correspond to an "effectively computable characteristic function", i.e. it is an automatic, fully specifiable (algorithmic) process. I will come back to this view of the comprehension process when I oppose algorithms to heuristics (which do not constitute such a characteristic function) (see 3.2), but let me briefly point out a few things about this constraint here. To me, it is more a computational constraint imposed by the type of parsing program one has in mind than a true psychological constraint. To call it "reliability" makes it sound psychological, but only obscures its true nature. It is not because language users are assumed to *reliably* classify sentences as grammatical or ungrammatical (Bresnan & Kaplan, xl), that they also use a failsafe algorithm in (syntactic) parsing (cp. also the critique on the use of (often unreliable) intuitions in 2.2.2.2). Hence, just like "creativity", "reliability" is another instance of the closed level fallacy. The fourth constraint is "order-free composition". It states that grammatical relations derivable from an arbitrary fragment of a sentence -- like *not told that* -- must be included in the grammatical relations derivable from the entire string -- like "I was not told that she was here". Indeed, people seem capable of interpreting arbitrary fragments of text out of context. However, the question arises what the meaning of "order-free composition" is as a psychological constraint, since it is never required of a human being (except in tests to prove that it exists; even then, the fact that we understand can be a mere consequence of our capacity to imagine the rest of the sentence). The motivation for this constraint is again computational rather than psychological. As Bresnan & Kaplan point out themselves:

> "the order-free composition constraint asserts that sentential context may determine the *choice* of one of a set of locally computed grammatical relations for a segment, but the computation of grammatical relations for a segment may not involve the computation of the grammatical relations of the context. In other words, this postulate severely constrains the role of context-sensitive operations in the syntactic mapping" (1982, xlvii).

If we consider that an lfg uses context-*free* phrase structure rules (with a minimum of context-sensitive information attached to them), the computational motivation of the locality constraint order-free composition is, clearly follows from the theory (plus its computational realization) and is not psychological at all. The last constraint then is "universality", stating that the procedure for grammatical interpretation is assumed to be the same for all natural language grammars. As with the related single-representation hypothesis, the methodological motivation of simplicity of explanation is allowed in again. I refer to the criticism in 2.2.2.2 and to Clark & Malt (1984, 205-206) for a refutation of the acceptability of this constraint as psychological. Suffice it to say here that it is just as plausible that users of completely different languages (e.g. free word order versus fixed word order) might also develop different strategies for analyzing their languages (an assumption that is compatible with the anteriority of process thesis).

That I have spent some time on the constraints proposed by Bresnan & Kaplan is because it seems like a laudable initiative of linguists to be willing to constrain their theories psychologically; in fact this implies that performance is allowed to constrain competence, something which Chomsky himself would not allow. However, we have seen that either the constraints are not psychological at all and inspired by the competence theory or they are psychological but their impact is kept small enough to leave the theory intact.

In chapter 4 of Bresnan (1982) (Kaplan & Bresnan 1982) then, lfg is presented in extensive detail, and the discussion opens with the competence hypothesis. I quote it once again, because it stresses another aspect I want to go into briefly:

> "We assume that an explanatory model of human language performance will incorporate a theoretically justified representation of the native speaker's linguistic knowledge (*a grammar*) as a component separate both from the computational mechanisms that operate on it (*a processor*) and from other nongrammatical processing parameters that might influence the processor's behavior (..) To a certain extent the various components that we postulate can be studied independently, guided where appropriate by the well-established methods and evaluation standards of linguistics, computer science, and experimental psychology. However, the requirement that the various components ultimately must fit together in a consistent and coherent model imposes even stronger constraints on their structure and operation" (1982, 173).

Note that this is a much weaker statement again than the "strong competence hypothesis" discussed above: instead of stressing the dependence of the processor (the P-model) on the grammar (the C-model), the independence of both models is now stressed. But what I want to point out here is the part about the "components" involved. I have repeatedly mentioned that the attempts at realization never come any further than a vague component homomorphism. This seems to be the consequence of overstressing the design stance (see 1.3.3.2) in scientific research: researchers ultimately concentrate on one component (lfg remains a formalism for representing *syntactic* knowledge), and leave the others for what they are (black boxes), as well as the ultimate fitting together of the components. It is never made clear how this fitting together will eventually be done. (Note that beside showing the vagueness of the component mapping this reasoning can also be used as a motivation for choosing the interactive model of language comprehension over the autonomous component one with its largely unspecified components and their restricted interactions; see 2.2.2.2.)

I will not go into how lfg's are analyzed in a parsing model here, but briefly return to this matter when I present the computational model that accompanies process linguistics (4.3.3.4). Let me only point out that at the end of the presentation of lfg, Kaplan & Bresnan weaken their competence hypothesis even further when they discuss the generative power of their model:

> "If our system turns out to have full context-sensitive power, then there are no known solutions to the recognition problem that require less than exponential computational resources in the worst case. It might therefore seem that, contrary to the Competence Hypothesis, lexical-functional grammars cannot be naturally incorporated into performance models that simulate the apparent ease of human comprehension" (Kaplan & Bresnan 1982, 271)(27).

They go on to state that there will probably have to be some more constraints on the theory, and also that nongrammatical heuristic strategies will have to guide the processor's computations.

In chapter 11 (Ford et al. 1982) a detailed example is worked out of how an lfg could be involved in the comprehension of structural ambiguities, as in "(the woman) (wanted) (the dress on that rack)" versus "(the woman) (wanted) (the dress) (on that rack)". (The prepositional phrase can be a postmodifier of the noun phrase "the dress" or it can be an adjunct by itself.) A detailed analysis of their approach is beyond the scope of the overview here; I will return to it in 5.2.3 when I discuss psycholinguistic research into lexical expectation. Just two brief remarks. First, it is assumed

---

(27) And indeed, Berwick & Weinberg (1984 chapter 4) have shown that lfg's can generate languages whose recognition time is computationally intractable.

that structural ambiguities are solved by using syntactic knowledge alone, which is comprehensible considering that lfg is a theory of syntax, but which is intuitively not very plausible. Meaning in context seems more important than syntax. Second, in order to account for the solution of ambiguities much more is needed than a bare lfg. A central assumption is that different lexical forms of i.c. verbs have different "strengths", and that the strongest form determines the preferred analysis. For example, *want*<(SUBJ)(OBJ)> is assumed to be stronger than *want*<(SUBJ)(OBJ)(PCOMP)>, which would lead to a preference for the first of the two sentences above. What this strength is and where it comes from is left open; moreover, it is assumed to be contextually neutral, which comes down to saying that (semantic) context is not considered. To me, this is again an illustration of how wide the gap is between a theory plus its formal objects and the way this theory has to be used in practice.

Admitting that this discussion of Bresnan 1978 and 1982 is incomplete, I still believe it is clear that it is not shown in a convincing way that a C-model fits nicely in a model of performance. Lfg remains a theory of syntax (28); the need of bringing in semantic information and heuristics is acknowledged but these elements are not studied seriously; competence remains a "store of knowledge" on which true performance constraints hardly have an impact; statements about the type of mapping range from optimistic isomorphism to doubt about the analyzability of lfg's within reasonable time limits. Although Bresnan explicitly takes a cognitive science perspective, it remains a formal linguist's perspective. Considerations that dominate the approach come from generative grammar and a computer science approach to (formal) language recognition; psychology and AI are hardly allowed in. For psychology, I refer to Clark & Malt's critique of Bresnan & Kaplan's constraints; for AI, it is e.g. deplorable that for the idea of "lexical preference", Wilks' work on preference semantics is not even mentioned (Wilks 1973, 1976 and passim). But then Wilks' approach is semantic, whereas Bresnan et al.'s is syntactic...

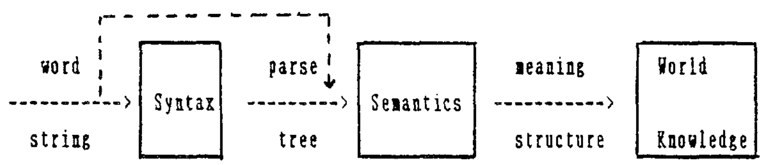
---

(28) In this respect, it is deplorable for a theory that crucially makes use of highly specified lexical elements that the whole 1982 book does not contain one single fully specified lexical entry.

Marcus 1980

By way of transition to the last station in the overview (Berwick & Weinberg 1983, 1984), the important work by Marcus (Marcus 1980) has to be discussed first. Although Marcus is not a generative linguist (he is an AI researcher at MIT), the parser he developed is closely linked to the theory of generative grammar. (Berwick & Weinberg use the Marcus parser as their concrete example of what a P-model incorporating the gb theory of generative grammar could look like, see below.)

Marcus' parser (called Parsifal) is a rule-based syntactic parser, meant as the first stage of an autonomous component model of performance that looks like this (Marcus 1982, 117):



The parser assigns a syntactic structure to a string in accordance with the extended standard theory variant of tgg, and uses two important data structures to achieve this. The "active node stack" contains constituents that are not yet completely determined, and the "three-place buffer" contains complete constituents that have to be attached to the incomplete ones in the active node stack. (An incomplete VP, for instance, can sit in the active node stack until an NP is processed in the buffer and attached to it to complete it.) The buffer also receives the parser's input words as they are read. The main feature of the parser (also very important to Berwick & Weinberg, see below) is that it is assumed to work "deterministically". This means that all the substructures built in the course of the process are permanent and cannot be undone. In contrast to most syntactic parsers, Parsifal does not carry along possible alternatives to eventually choose from (i.e. there is no parallelism), nor does it try a number of possibilities in series, backing up to a choice point when a chosen path leads to a dead end (i.e. there is no backtracking). Considering that keeping a number of possibilities active (the parallel approach) is very space-consuming and that continually revising wrong choices (the backtracking approach) is very time-consuming, the attractivity of determinism is clear. The least one can say is that it allows for efficient and fast computer parsing. Whether the claim that it is also a correct model of human parsing (a claim made by Marcus and by Berwick & Weinberg) is correct will be considered below. In order to achieve this determinism, the parser has to be allowed to look ahead (otherwise it could rarely make a correct (irrevocable) decision in natural language analysis). For instance, to analyze "Is the child eating peanuts yours?" and "Is the child eating peanuts?", a deterministic parser cannot interpret *eating* correctly (as part of a postmodifier or as a main verb respectively) until it sees *yours* or the question mark after *peanuts*. What the parser can look at is the content of the buffer (limiting the lookahead to three constituents).

Important to the discussion of the realization of C-models in P-models is a type of reasoning also used by Berwick & Weinberg, viz. that the principles/constraints of generative grammar can be shown to follow from the way the parser works, i.e. from its determinism. Since determinism in parsing is a completely theory-independent (i.e. tgg-independent) element, to show that the constraints on language "fall out" of the way the parser (P-model) works is indeed a serious argument in favor of the correctness of the C-model and its incorporation into a specific P-model. Note that, as with Bresnan, there is a healthy tendency to look for processing constraints on grammars, which -- as I see it -- is one of the only ways to stay clear of the pitfall of the closed level fallacy and find true explanations for the fact that languages are as they are. Yet, the attempt is not so successful as it may seem.

First, the psychological validity of Marcus' notion of determinism has been criticized by several researchers (see Sampson 1983a, Briscoe 1985, and references therein). As Marcus points out himself (1980, 17), the lookahead allowing for determinism has to be limited in order for the notion not to be vacuous. Yet, the buffer can contain constituents (like NPs) consisting of an unpredictable number of words, which gives Parsifal "infinite lookahead at the word level" (Briscoe 1985, 63), making "determinism" vacuous indeed. Moreover, the infinite delay of processing this implies is psychologically unacceptable considering our limited processing memory and the psycholinguistic evidence in favor of undelayed processing (see Marslen-Wilson & Tyler 1980). But what I consider to be the strongest argument against determinism is the psycholinguistic research into lexical access (discussed in 5.2.2 and 5.2.5). This research has convincingly shown that *all* meanings of a word are accessed in parallel by an automatic, uniform and exhaustive retrieval process. Now, one might say that determinism is only claimed for syntax and not for semantics, but even then the research findings cause problems for its adherents. For a sentence like "They all rose", it was found that *both* the verb and noun meaning of *rose* ("stood up" and "flower" were briefly accessed during comprehension (Tanenhaus et al. 1979); hence, multiple syntactically important distinctions (word class) are briefly active, which is not "allowed" under the determinism hypothesis. In this context, I believe that the notion of determinism can hardly be said to apply to the *process* of language understanding; at the most, we can say that context restricts multiple interpretations (alive during the nondeterministic process) so that the process converges on a single interpretation (the output of the process is "determined"). A final argument against determinism and its crucial use of lookahead comes from the consideration of a language like Dutch (in contrast to English). English is a language in which linguistic elements belonging together logically (like an auxiliary and a participle in a VP) are also found closely together in a sentence. It is subject to a number of "locality constraints", which also makes it possible to only use limited lookahead in parsing (a 3-place buffer will mostly suffice to find elements belonging together). Dutch, on the contrary, is dominated by a principle that is the opposite of a locality constraint, viz. the *pincers construction*. In 4.3.3.5 I will go into this phenomenon and the way the computer model complementing process linguistics can (easily) deal with it; here I only point out that it implies that linguistic elements belonging together (like an auxiliary + a participle/infinitive, or a stem of a compound verb + its particle) naturally tend to be wide apart in a sentence (holding other constituents in between them like the sharp edges of a pair of pincers). Hence, a small, fixed buffer will not suffice to cope with this phenomenon. In 4.3.3.5 we will see that we need much more dynamic machinery (using expectations plus feedback) to deal with this than an inflexible lookahead buffer.

In short, to derive constraints of a grammar from the deterministic way a parser works is one thing; to show that determinism really holds (especially if one claims psychological reality) is another, and should in fact be done first.

Another critical remark about the Marcus approach relating grammatical constraints and parser characteristics is that it is still not free from the closed level fallacy. Marcus' parser was strongly inspired by generative grammar itself (its rules, principles, structures). We will see below how this leads to a number of circular reasonings in Berwick & Weinberg 1984. How generative grammar inspired the mechanisms of Marcus' parser is pointed out by Sampson (1983a, 107-116). Recall that the parser uses an active node stack as one of its main data structures. Now, a "pure" stack (as a data structure used in computer science) only allows access to its top element; the rest of the stack is inaccessible to the processes manipulating it. Yet, Marcus' active node stack allows access to two of its elements (the top element, and the nearest node in the stack -- i.e. nearest to this top element -- labelled 'S'). In itself, this is not unacceptable, but the reason why a second element is accessible is inspired by generative grammar.

> "The idea that the parser can look leftwards in the stack but only to what is currently the nearest S node is connected with the principle of transformational grammar called 'cyclical application of rules' (..), i.e. that the ordered sequence of transformational rules applies separately to each clause in a structure containing nested subordinate clauses" (Sampson 1983a, 108).

Now, in order to show that the principles of generative grammar (complexNP-constraint, subjacency, tensedS-constraint, etc.) follow from the way the parser works, Marcus does not only need its determinism but also this extra accessibility of a second stack element. Hence,

"(..)it is not really true that he [Marcus] deduces the observable facts from the postulate of deterministic parsing alone; rather, he deduces them from the conjunction of that postulate with the postulate about accessibility of the 'closest S node' as well as the current node in the stack. And while the determinism principle is the sort of postulate that one might well want to adopt *a priori*, the 'closest S' principle looks much more *a posteriori*. That is, there is a hint of possible circularity here -- maybe the decision about how the active stack can be accessed was influenced by the need to reflect the observed constraints, in which case the 'explanation' of the constraints is purely *ad hoc* and unpersuasive" (Sampson 1983a, 115-116).

A final note about Marcus' parser: it would be interesting to compare the computer model in chapter 4 (WEP) to this parser because there are some interesting parallels and differences between both. In 5.3.1 a brief comparison will be made in the context of lesioning computer models in accord with findings in aphasia research. Let me just remark that Marcus' parser fits in with the autonomous component models of language processing, whereas WEP is an example of an interactive model. Both parsers attach great importance to bottom-up processes (processes triggered by the words themselves), as well as expectations. However, as far as this latter element is concerned, in Parsifal "expectations" are no more than static syntactic rule structures (be it that they also refer to the two data structures mentioned above), whereas in WEP they are dynamic meaning-driven processes (see 3.3.4 and chapter 4; see also Small 1980, 19-20).

Berwick & Weinberg 1984

This detour via Marcus brings me to the last station in the overview, viz. the Berwick and Weinberg research (1983, 1984). Like Bresnan et al. they stick to the competence hypothesis, but unlike Bresnan they keep defending transformational grammar (especially government-and-binding (gb), its latest version). The grammar is left untouched, but the mapping from C- to P-model is given more attention, as well as the parser that constitutes the P-model (29). As far as the

(29) See page 85.

mapping is concerned, Berwick & Weinberg take a very ambiguous position, comparable to that of Bresnan et al. but worked out in more detail and with stronger claims. The idea is again that a direct, isomorphic mapping is desirable but that sophisticated weaker mappings can just as well serve as support for the competence hypothesis:

> "(..)transparency (i.e. isomorphism, g.a.) is not a *necessary* property of a parsing model. If future [psycholinguistic] experiments show that this direct mapping is untenable, then researchers interested in constructing a theory of language should still be interested in a theory of linguistic competence, to the degree to which we can use this theory to constrain the class of possible parsers. (..) there is a continuum of more or less direct realizations of a grammar as a parser. There is not just an 'all or none' choice between a grammar embedded directly as a computational model (the DTC model) and a total decoupling between grammatical rules and computational rules (with the structural descriptions of the grammar computed by some totally unrelated 'heuristic strategies', the Fodor, Bever and Garrett [1974] conclusion)" (1983, 46).

A statement like this sounds slightly suspicious to me. It suggests that whatever the (psycholinguistic) evidence brought in against the competence hypothesis one can always retreat to some weaker form of realization and save the hypothesis (or rather, the theory of generative grammar).

As an example of the ambiguous attitude towards mapping types, let me discuss the way Berwick & Weinberg deal with the FBG verdict against the DTC. Since this verdict was reason enough for linguists and psychologists to abandon tgg, it is not surprising that Berwick & Weinberg discuss it in detail. But whereas one might expect that they simply reject the directness of mapping implied by the DTC (as suggested in the quotation above), they go a lot further and try to show that tgg can perfectly well accommodate the research results that convicted the DTC (implying that even an isomorphic mapping is not problematic for a tgg). How do they do this?

---

(29) Let me note right away that this parser is again purely syntactic; the arguments against this approach have repeatedly been given in the course of the discussion, so no more will be said about it here.

Recall that the DTC predicts that sentence complexity (as measured by reaction time) is directly proportional to the number of transformations applied to the sentence. Yet, the experiments showed no reaction time difference between sentences to which a different number of transformations were applied, which was considered a refutation of the DTC. The way to accommodate these results and save the transformational approach is by assuming that parsing actions involved in transformations do not happen serially, but in parallel. If this is the case, then it is perfectly plausible that analyzing transformed sentences should not take longer than untransformed ones (as was found in the experiments). If, for instance, five (micro)actions are taken by the parser when applying a transformation in reverse, serial application takes the sum of the time lengths of all five, whereas parallel application takes only the time of the longest action. In short, by assuming a limited form of parallelism in processing, the results against the DTC can be accommodated within a transformational framework.

Beside the fact that it is unclear to me how this argument about what they call cognitive capacity (limited parallel processing) adds value to the transformational approach in itself, the argument has two flaws. The first has been pointed out by Bresnan & Kaplan (1982, xxv-xxvi). As Berwick & Weinberg are well aware of themselves, the kind of parallelism they have in mind can only apply to parsing actions that are completely independent of each other (i.e. no action has to wait for a result from another). On a microlevel (i.e. parsing actions within one transformation, such as manipulating specific data structures), this parallelism can be sustained, but on a macrolevel (i.e. when more than one transformation is involved) it becomes impossible. The reason is that transformations are subject to "feeding" or "ordering" relations (e.g. Dative-Passive, Dative-Passive-ThereInsertion), which implies that the necessary input of one is created by the output of the other. Hence, parallel execution is impossible and the arguments against DTC still hold. A second flaw is one that recurs throughout the Berwick & Weinberg approach, viz. the assumption that the human being operates in the same way as a computer. Language analysis would then be equivalent to running a program (as specified in an algorithm) in one's head (cp. 1.3.2, the discussion of the computer metaphor). Expressions of the assumption can be found in statements like these:

- "(..)the linguist does not typically ponder the issues of computational implementation that must be faced squarely by online processors, be they people or machines" (1984, 41)

- "(..)we might be more confident that, no matter what particular "implementation" the brain had picked, our algorithm would still be superior. It still does not necessarily follow that the brain would pick that particular algorithm" (1984, 101)

- "Presumably, part of the job of the cognitive psychologist is to try to find out whether people use cubic or exponential time algorithms" (1984, 268 note (14))

The equation of computation and cognition (the strong AI view (30)) is simply assumed to hold; no evidence is adduced for it. Hence I cannot agree with Berwick & Weinberg when they claim that what they are doing is cognitive science. For one thing, psycholinguistic evidence is hardly allowed in; for another, related to this is a sloppy mixture of the design stance and the physical stance (cp. 1.3.3.2) in their cognitive science brand. They are dealing with the parsing process in the human mind, yet it is just "the brain" that picks algorithms. To come back to the parallelism assumed to accommodate the DTC results: the way it is realized in the processor comes down to a mere computer-hardware discussion (the hardware is changed to allow parallelism) and it remains totally unclear how this has to be interpreted for the human brain (let alone how the mind deals with parallelism) (31). In short, the DTC results still cannot be accommodated in a convincing way within the traditional transformational paradigm, in spite of all the computer-scientific sophistication brought in to back up the argument.

Another example of the unclarity of the view of mappings from C- to P-models is related to the position of rules in government-and-binding. Whereas they were important in

---

(30) Ironically enough, although Berwick (see Berwick 1983a) has no great idea of AI work in natural language processing, he finds himself committed to a strong version of AI.

(31) Cp. Kolers & Smythe 1984 for a sharp critique of the way some cognitive scientists sloppily mix levels of description and/or explanation.

earlier versions of tgg, in gb the stress is on general principles and constraints (32). Phrase-structure rules have become unimportant, and there is only one transformational rule left, "move α", with α an arbitrary phrasal category. Since an important aspect of the competence hypothesis is that the rules of the competence grammar are "followed" by the parser, the question arises how the principles are involved in a mapping. Berwick & Weinberg answer the question as follows:

> "(..)this change dramatically alters the conception of a parser that "follows" the government-binding theory. Earlier work assumed that a parser "followed" a grammatical theory if and only if it employed the same rules as that theory (though perhaps in inverse order). The explanatory shift to principles entails a new conception of what parsing comes to: A parser that satisfies these principles is a government-binding parser even if the algorithm it uses only roughly resembles a government-binding type Move α rule" (1984, 33-34).

The concrete parser Berwick & Weinberg use throughout their book is the Marcus parser; they try to show that with the necessary revisions it "follows" all the incarnations of tgg (from standard theory to government-and-binding) in some way. I focus on two aspects of all the mappings discussed. It is shown a.o. that the Marcus parser

1) is structure-isomorphic to the Extended Standard Theory of tgg (it builds the same annotated surface structure)

2) is "principle isomorphic" to the government-binding theory of tgg

Hence, they say, it is perfectly feasible to realize a tgg in a P-model.

Here again, the argument is not convincing. In the first place -- as already mentioned above, Marcus' parser was built on the basis of transformational theory itself. Hence, to show that the parser builds structures and respects principles of the theory is merely restating facts, and it leads to circular reasonings. As an example, consider the "dropping

---

(32) See especially Berwick 1983a for a good overview of the changed perspective on rules.

of traces" by the parser. Parsifal does this e.g. when an NP it expects does not occur in its "canonical" position but is dislocated; the trace is made to point to the dislocated NP. Now, Berwick & Weinberg seem to consider trace-dropping as an independently motivated parser action, and they try to rationalize it by referring to the principles of generative grammar:

> "(..)a parser's automatic dropping of a trace in a post-passive participial position may be rationalized as the expression of the constraint that only non-phonetic elements may appear in this position, a constraint in turn explained by the various subtheories of the government-binding theory" (1984, 147-148).

> "(..)there must be some bounding condition on trace insertion, *because this is a parsing decision*" (1984, 158; emphasis mine).

Yet, as Sampson (1983a, 104) also points out, "trace-dropping" is not an independently motivated parser action, but inspired by trace theory in generative grammar itself. Hence, a hint of circularity and of the closed level fallacy.

Beside this circularity, there is another aspect of the "principle isomorphism" which reduces the strength of its support of gb. Consider the statement that "a modified Marcus parser makes crucial reference to the principles of transformational grammar (analogues of the projection principle and the *theta*-criterion) in order to guarantee deterministic parsing" (1984, 143). Now, the principles involved are principles that any theory or approach to natural language contains in some form: the projection principle states that the subcategorization properties of lexical items have to be satisfied at all levels of description (e.g. if a verb subcategorizes for a direct object, then this direct object has to be present at all levels); the *theta*-criterion says that every NP in a sentence receives only one thematic role (e.g. agent) and that all thematic roles associated with a predicate must be assigned. It is not clear to me what makes these syntactico-semantic principles so typical of transformational grammar; reversely, almost any working parser would then be a government-binding parser, a predicate not everyone would appreciate.

A third element of criticism brings me back to the position of rules in tgg and the quotation above on p. 88. The quotation (from the conclusion to their chapter 1) downplays the role of the "move-α" rule in parsing; in the conclusion to chapter 5, however, we hear a totally different sound in a very strong claim:

> "By positing that Move-α exists and that it is engaged in mental computations, we can actually explain some facts about natural languages, namely that they will obey subjacency in certain situations and not in others, and derivatively some facts about human behavior" (1984, 196).

Here it is stated that move-α is "engaged in mental computations", which means that the rule is assumed to be "causally engaged" (Berwick 1983b) in language processing. This implies the (desirable) rule isomorphism denied in the quotation on p. 88. Note also that the move rule and its use are simply "posited", whereas to claim mental reality would necessitate psycholinguistic investigation of whether we actually move constituents around in our heads. Note, in the passing, that the statement also has an aspect of circularity, namely in saying that the move rule "explains" subjacency. Subjacency is a locality constraint introduced in tgg to *constrain* the movement transformation. In other words, subjacency is a theory-internal notion which has no meaning independently of another theory-internal notion, the movement transformation. To say that *a* explains *b* if *b* has to constrain *a* is giving a very strange meaning to "explanation" indeed.

Finally, just like Marcus, Berwick & Weinberg (1984, 153-173) try to derive the principles of gb theory (especially subjacency, the locality constraint on movement (33)) from the deterministic way the parser works. Here too, the arguments against the claimed psychological validity of determinism hold. Moreover, just like Marcus, Berwick & Weinberg need additional assumptions that reduce this validity even more. In order to show that subjacency "must" hold, determinism is combined with the assumption that the parser only has access to grammar symbols like S, NP, VP (the

---

(33) It is worth mentioning that subjacency is simply assumed to be an "axiom" (Berwick & Weinberg 1984, 154) of the grammar (a linguistic universal), whereas it is not at all certain that the constraint really holds for all natural languages (see Van Riemsdijk & Williams 1986, and also Chomsky 1985 for recent reconsiderations of subjacency).

results of parsing the left context at a certain point during analysis) (Berwick & Weinberg 1984, 158). Now, it is unlikely that the human being only has access to such (theory-bound) symbols. The left context has been fully analyzed at a certain point in parsing and its complete content (also the semantic/pragmatic content) is accessible in further processing.

In conclusion, I would say that Berwick & Weinberg's realization of a competence grammar in a performance model is even less convincing than Bresnan's. Their P-model is no more than a syntactic parser, and it is constructed on the basis of the theory it is supposed to support. Berwick & Weinberg's brand of cognitive science does not go beyond a combination of generative grammar and computer science (as they announce in their preface). Further claims about psychological (or even neural) validity are easily made but not supported by evidence (not to mention the lack of care about the different levels of analysis, i.e. the mixture of the design and the physical stance in research (cp. 1.3.3.2)). Whereas Bresnan et al. also stress the formal computational part of their approach (their chapter 4), they at least try to show that lfg is useful in characterizing a true performance problem, viz. the solution of structural ambiguities (their chapter 11). Nowhere do Berwick & Weinberg show how gb is useful in such real-world performance matters.

This concludes my critical overview of attempts to realize generative grammars in P-models. It will have become clear that I do not consider them convincing or succesful: even the post-FBG attempts do not go beyond syntax and a formal approach to processing (the theory of algorithms and computation) leaving considerations like semantics, pragmatics, the use of heuristic processes, etc. greatly unspecified in spite of loud claims about psychological and even neural validity of the models. This conclusion brings me back to the question of the relation between linguistics and cognitive science, a relation revisited in the last subsection of this chapter.

2.4. Linguistics and cognitive science revisited

At the end of chapter 1 I sketched the ambience surrounding linguistics and its relationship to AI, psychology, and cognitive science in general. In this chapter I have criticized generative grammar, concentrating on its psychological claims (mainly through a critique of the competence-performance distinction). It was shown that in spite of all the claims to the contrary, the gap between "competence" theories and performance remains unbridgable. The centrality of the former to the latter is mere wishful thinking to support the "correctness" of the formal theories of grammar.

The formal and non-psychological nature of generative linguistics has also been pointed out by philosophers like Katz (1981) or Soames (1984). Katz expresses it as follows:

> "(..)linguistics is not a psychological science, (..) its theories are not about states of mind, mental events, or their neurological realizations, but about sentences and languages directly in the way that we ordinarily take linguistics to be about sentences and languages (..) sentences and languages are abstract objects and thus linguistics is about abstract objects" (1981, 76).

And Soames:

> "(..)linguistic theories (i.e. in generative linguistics, g.a.) are *conceptually distinct* and *empirically divergent* from psychological theories of language acquisition and linguistic competence. In arguing that these two kinds of theories are *conceptually distinct*, I will try to show that they are concerned with different domains, make different claims, and are established by different means. In maintaining that they are *empirically divergent* I will argue that the formal structures utilized by optimal linguistic theories are not likely to be isomorphic to the internal representations posited by theories in cognitive psychology" (1984, 155).

As we have seen in 1.3.4, even generative linguists like Gazdar et al. want to stay clear of the dangers of psychologizing their formal approach.

To me the conclusion of the critique in this chapter can only be that linguistics needs to take a different view of language if it wants to achieve smooth integration with psychology and AI, i.e. if it wants to achieve "processual adequacy" and aspires to the predicate "cognitive-scientific". Rather than dealing with language as a (structured) object in itself, cognitive-scientific linguistics is concerned with the processes of language production, understanding, and learning in the individual (34). In its approach it makes crucial reference to models and experimental results in psychology and weak AI (the computer only serves as a means to simulate the psychological processes considered). For cognitive-scientific linguistics the structure of language is not so important; it is merely the input to or the output of psychological (and ultimately neural) processes. The way linguistic knowledge is represented has to be motivated by the processes that work with this knowledge, and not the other way around (as claimed by generative linguists); to the extent that descriptions from "traditional" linguistics fulfil this requirement, they are useful (and even indispensable) to cognitive-science linguistics. For instance, if -- as many psycho- and neurolinguistic experiments point out (see chapter 5) -- the organization of the mental lexicon is of great importance to language processing, a cognitive science approach to linguistics cannot afford to restrict itself to syntax.

Examples of approaches I consider to be cognitive-scientific linguistics (often only suggested or sketched) are Derwing (1973, chapter 9), Lakoff & Thompson (1975), Taylor (1976, 139-145), Kempen & Hoenkamp (1982, 1984), Schank & Riesbeck (1981), Small (1980), Cottrell (1985). In the next chapter I also present a cognitive-scientific approach to natural language understanding, process linguistics.

---

(34) Sociolinguistics and anthropological linguistics could be called cognitive science linguistics of the individual as functioning in a society.

CHAPTER 3: TOWARD PROCESS LINGUISTICS

> "The ascribing of meaning to a message comes from the invariance of the processing of the message by intelligences distributed anywhere in the universe." (Hofstadter 1979, 171)

> "(...) I must confess that I have always been more impressed with the capacity of the human brain to discriminate, characterize, and store in memory the 30-plus thousand arbitrary words in active use than with the complexity claimed to be involved in learning a few dozen syntactic algorithmic rules." (Marin 1982, 64)

3.1. Introduction

In the next three chapters of this book I develop a cognitive-scientific linguistic approach, called process linguistics. In this chapter (the "linguistic chapter") I discuss some general concepts that partly follow from the critique of generative grammar in chapter 2 and partly from work in the other sciences constituting cognitive science. Chapter 4 (the "AI chapter") gives a more technical presentation of a computer model (the Word Expert Parser) that implements some of the process-linguistic concepts. In chapter 5 (the "psycho- and neurolinguistic chapter") the AI model is critically confronted with specific psycho- and neurolinguistic research findings.

The sketch of process linguistics (for convenience, PL) that now follows should be seen as a first exploration of its assumptions and principles; as such it is certainly incomplete and may lack coherence. Further work will be necessary both in theory and, maybe more importantly, in practice (the application to concrete natural languages). Rather than mention people whose work I have drawn ideas from throughout the text, I shall enumerate most of them beforehand. For the general background: Winograd (1977, 1983); for particular aspects of the approach: psycholinguistic research (see chapter 5), especially the research into the role of context and the lexicon in comprehension; neurolinguistic research

(Butterworth 1983b, Arbib et al. 1982); AI research (especially Small 1980 and the related research mentioned there, mainly from the Schank school in AI); linguistic research (Starosta 1978, Gross 1984, Taylor 1980, Hudson 1984). Related - but often only suggested - models are "network grammars" (in Taylor 1976) and "cognitive grammars" (Lakoff & Thompson 1975) (not to be confused with Langacker's cognitive grammar (1983), with some of whose ideas, by the way, PL is compatible (1).

3.2. On the importance of processes

In chapter 2 I have already stressed the importance of the consideration of processes in linguistics: the anteriority of process thesis was put forward as a challenge to the stress on (linguistic) structure in the generative paradigm where processes are relegated to "performance" and not considered within the notion of "competence". I will now take a closer look at some aspects of the (elusive) process notion.

First, a terminological note. I have chosen the term "process (linguistics)" for a number of reasons. In the first place, there is the general anteriority of process (over structure) thesis. Second, PL is an approach to the (macro)process of natural language understanding and the (micro)processes that are involved in this overall process. (As far as the microprocesses are concerned, I will

---

(1) I simply enumerate some of Langacker's ideas compatible with those expressed here: the disbelief in the formal rule-based approach of generative grammar (e.g. in formal logic for the description of semantic structure); the view of the lexicon, morphology and syntax as a continuum instead of the belief in an autonomous grammar (syntax) as distinct from lexicon and semantics; the view of language as evoking other cognitive systems and to be described as an integral facet of overall psychological organization; the view of linguistic knowledge as a structured inventory of conventional linguistic units (broadening the dictionary conception to an encyclopedia conception); the stress on the importance of context (related to the disbelief in compositionality of semantic structure); the need to consider developments in AI and psychology (e.g. network models, Rosch' prototype notion); and, finally, a processual approach to motion (Langacker 1983, 1985).

concentrate on the expectation-feedback cycle which I propose as an important processing universal (see 3.3.4).) The cognitive processes will ultimately have to be in keeping with neural processes (the long term view); moreover, they are simulated by computational processes. Hence, the multiple meanings of "process", fitting in nicely with the importance of the notion in all sciences constituting cognitive science. (Note, by the way, that there even exists "process philosophy".) Third (related to the foregoing), "process" is a neutral term. By this I mean that it does not immediately suggest a computer realization, in contrast to the terms "computational" and "procedural", often used by cognitive scientists. (For process linguistics, computational processes are just one facet of the approach.) For the term "computational" I want to make a remark about two of its many usages. The first is Winograd's (1977): as already mentioned, he proposes the "computational paradigm" as an alternative to the Chomskyan generative paradigm for the study of natural language. I agree with Winograd's ideas (the basic set of assumptions of the paradigm are listed at the end of this subsection, and are all aspects of the anteriority of process thesis), but would prefer to call the paradigm the process paradigm. Note that in this context one can say that "traditional" linguistics represents the structure paradigm (in this broad sense, Chomskyan linguistics is also "structuralist"). The second usage of "computational" is in the term "computational semantics" (see e.g. Charniak & Wilks 1976). Very generally, it refers to an AI-based approach to natural language understanding where semantics and pragmatics are more important than syntax. Since the search for meaning is central in process linguistics (involving pragmatic, real-world knowledge as well), it is certainly compatible with computational semantics. However, it wants to be more than just AI-based in that it presents itself as a linguistic approach in the first place. For the term "procedural" (even more strongly reminiscent of computer science with its programs composed of *procedures*) I have to mention "procedural semantics". It refers to an attempt by AI researchers to develop a dynamic theory of meaning (meaning as procedures), partly as an alternative to static formal and truth-conditional approaches to semantics (meanings as structures); see Wilks 1982 for a critical discussion of Johnson-Laird's, Woods' and his own version of procedural semantics. Although a lot could be said about parallels and differences between process linguistics and procedural semantics, some general remarks have to suffice here. To the extent that procedural semantics is another example of how a process-centered approach is proposed as an alternative to a structure-centered one, the parallel is clear. However, PL is not a semantic theory but an overall approach to the understanding process. Finally, in procedural semantics the meaning(s) of words, constituents, and utterances are seen as "the procedures they invoke in or by the hearer/receiver" (Wilks 1982, 500), e.g. procedures to relate words to phenomena in the world. PL also holds a dynamic view of i.c. words, in that they are seen as active entities triggering (micro)processes in the macroprocess of understanding. The goal of these processes (which can involve linguistic *and* extralinguistic knowledge) is the construction of a semantic representation (see 4.2.2.2), but this representation is seen as a side-effect of the processes. In other words, the processes are not equated with the meaning as in procedural semantics, they only lead to its discovery.

With these terminological issues settled, I will now look at processes and their anteriority to structures from a more philosophical point of view, making use of two simplified examples from physics. The notion of context plays an important role in the discussion beside the structure and process notions.

As human beings we are inclined to perceive objects as having certain static characteristics. Take the example of *color*: we say a tomato is red, a tree is green, the sky is blue. Thus, color is believed to be one of the features that structure an object. However, when night falls, colors seem to disappear and everything looks grey or black. With this introduction of the *context* we may start to doubt whether color is really a characteristic of things: color seems to be context-dependent. Thinking still further in an attempt at explaining the phenomenon, the importance of light (a complex physical *process* of waves or particles in space) becomes clear: objects do not statically have color, but they reflect part of the light waves and absorb others so that we perceive the objects as having the color determined by the reflected waves (how this perception is brought about is still another complex process matter...). Starting from the obvious (but deceptive) perception of structural aspects we arrive ultimately at the processes that explain the phenomenon (in a way that seemed far from evident from the observation) through a consideration of the object in its context; observable structural characteristics are a side-effect of less obvious processes. The same reasoning applies to *weight* (cp. Hofstadter 1979, 171-172): at first it was thought to be an inherent characteristic of things, then it was related to the context of the Earth we live on, and finally it was explained by the force of gravity (again something dynamic, a non-perceptible process); thus, weight was no longer seen as a feature structuring an object. To put it schematically:

IMMEDIATE PERCEPTION	⟶	CONSIDERATION OF CONTEXT	⟶	DISCOVERY OF PROCESS
color as a characteristic		day/night		light
weight as a characteristic		the Earth		gravity

OBSERVATION <------------------------------- EXPLANATION

Now, a central assumption behind PL is that the same reasoning applies to language. To repeat the first motto of this chapter:

> "The ascribing of meaning to a message comes from the invariance of the processing of the message by intelligences distributed anywhere in the universe."

However, we are not that far yet that we can characterize this processing (viz. by the human being) like we can do with the processes of light or gravity. Yet, the implications should be clear, and they back up the proposal for an alternative view of the study of language discussed in 2.2.1: process is anterior to structure (viz. linguistic structure) and contains the ultimate explanation for the nature of the structure that does not exist in itself but only as a result of the way the processes work; these processes can be arrived at by considering the context in which the phenomena occur (viz. linguistic phenomena in their linguistic context and the cognitive context of the individual language user, ultimately to be broadened to the extralinguistic context). To put it, once again, in a schema, linking the notions of perceptibility, accessibility and static/dynamic nature (on a scale as indicated) to those of structure, context and process:

easily perceptible		hardly perceptible
easily accessible	-------------------->	hardly accessible
static		dynamic

STRUCTURE ⟶ CONTEXT ⟶ PROCESS

process explains
structure through
mediation of the
context

In linguistics, structure has received all of the attention up till now; in view of what has been discussed, this is only surface scratching dealing with the result of whatever processes are active during comprehension or production. PL holds that linguistic universals (as studied in generative linguistics) are merely abstract descriptive devices trying to characterize languages; they do not "explain" why language is the way it is and can at the most give indirect clues to a true explanation in terms of processing universals in the human mind, such as the way memory works or the way specific microprocesses (like expectation-feedback cycles) make use of this memory. PL will attempt a different approach to explain language, partly through its stress on a consideration of the context, or rather contextual interaction among linguistic elements (and the traces they leave in human cognition). This interaction may throw some light on the principles that make up the underlying processes, more or less abstractly characterizable in linguistics (see 4.3) and to be linked to the cognitive processes studied in psychology and simulated in AI (and eventually to the neural processes studied in the neurosciences).

A final remark to this general discussion of process: in spite of all the downplaying of the importance of structure, it remains a fact that structure is the most easily perceptible aspect of language and needs thorough analysis (be it that we would be better off if linguists dealt more with concrete languages than with the abstract entity "Language" (cp. Gross 1979)). Yet, although we know that processes escape direct observation, this does not mean that we cannot suggest hypotheses about them and test them through experimentation and simulation.

A distinction that is often made in the context of natural language understanding is that between algorithmic and heuristic processes, a distinction that is very important to PL. By itself, an algorithm is simply a fully-specified procedure (process) for solving some problem (e.g. an algorithm for multiplying two numbers, an algorithm for recognizing a context-free language, etc.). This is the standard computer-science sense of the word. However, Newell & Simon (1972, chapter 14) created some confusion by opposing algorithms to heuristics (cp. Pylyshyn 1984, 88). To them algorithms are dumb, mechanical and inflexible procedures for solving a problem, be it that a solution is guaranteed. They have little to do with true human problem solving which makes use of less stereotyped, smart, and flexible heuristics (i.e. procedures involving directed search), be it that they cannot be said to guarantee a solution. Hence, algorithms are now often seen as procedures guaranteed to solve a problem, whereas heuristics are seen as more plausible but incomplete procedures. From a "pure" computer-science perspective, the distinction is somehow confusing. If one wants heuristics "to work" on a computer, they will have to be incorporated into an algorithm (to be "translated" into some programming language). In short, on computers all procedures -- even heuristic ones -- are carried out by some algorithm (cp. Pylyshyn 1984, 88). Still, from a cognitive-scientific point of view the distinction is very useful. It is a fact that algorithms for problems like chess-playing, etc. can be made smarter (and work faster) by incorporating heuristics into them. (This mostly comes down to ways for reducing a potentially large search space by smartly "skipping" implausible possibilities.) Second, with Newell & Simon, I would claim that human beings indeed make use of directed, heuristic, intentional processes (see 1.3.2). The example of such a heuristic process that will return throughout the next chapters is the expectation-feedback cycle, a flexible and robust process at the heart of natural language understanding. Moreover, as far as human beings are concerned, we do not know exactly how these intentional processes are made to work: do they use "algorithms"? do they simply emerge from subsymbolic neural processes as suggested by the connectionists (cp 2.3.2 and 5.3.2)? All we can do is try to simulate them on computers, which means that we *have to* incorporate them into some (computer-bound) algorithm. What our simulation shows can teach us a lot, but it would be a mistake to claim cognitive validity of the computer algorithm. (We are back to the rejection of strong AI and the computer metaphor.) In this context Newell & Simon's distinction constitutes a warning for strong AI: algorithms built into programs are computer stuff, intentional/heuristic processes are characteristic of the human being. Weak AI bridges the gap between both by allowing a simulation of the latter by the former.

A consequence of this discussion concerns the type of equivalence claimed to hold between processes in the human being and the simulated processes. It will be clear that the equivalence is on the functional level of general features of both (testable in psycho- and neurolinguistic research) and not on the level of realizations on machines or in human beings. It seems safer to me to only claim weak equivalence than to assume strong equivalence and be left with untestable claims about parallels between computers and human beings (see further chapter 5).

To conclude these clarifications of the process notion I quote the basic assumptions of the process paradigm for the study of language (Winograd's "computational paradigm") that will sound pretty familiar after everything discussed so far:

> "The essential properties of language reflect the cognitive structure of the human language user, including properties of memory structure, processing strategies and limitations.
>
> The primary focus of study is on the processes which underlie the production and understanding of utterances in a linguistic and pragmatic context. The structure of the observable linguistic forms is important, but serves primarily as a clue to the structure of the processes and of the cognitive structures of the language user.
>
> Context is of primary importance, and is best formulated in

terms of the cognitive structures of the speaker and hearer rather than in terms of the linguistic text or facts about the situation in which an utterance is produced.

It is possible to study scientifically the processes involved in cognition, and in particular of language use. Some parts of these processes are specialized for language, while other parts may be common to other cognitive processes" (Winograd 1977,168).

3.3. General principles of process linguistics

3.3.1. Modality-boundness

It follows from the adherence to the anteriority of process thesis that PL is an approach to the processes involved in linguistic behavior. Moreover, it follows from the rejection of the single-representation hypothesis (one facet of the competence hypothesis in generative grammar) that PL can legitimately restrict itself to the study of only one process, viz. comprehension. Production, comprehension and acquisition are processes in their own right, involving the use of several knowledge sources in idiosyncratic ways that can even relate to the way the information is coded (see 2.2.2.2).

A further reduction is the stress on written language understanding (reading); this is partly motivated by the belief that this modality has its own principles (as opposed to spoken language understanding), and partly by the fact that the computer model that complements process linguistics is a written language understanding program. (Making a computer understand spoken language creates additional problems not considered here.) Modality-boundness (only *written* language *understanding* is studied) is a theoretically motivated abstraction (anteriority plus idiosyncrasy of process). Note that within this abstraction all levels of language are to be considered (morphological, syntactic, semantic, pragmatic); as such, the abstraction is orthogonal to the usual abstraction in linguistics considering one level (viz. syntax) and grafting the other levels (semantics, pragmatics) onto this level across all modalities, without considering the processual idiosyncrasies inherent in these modalities.

A short word about these other modalities. For models of production compatible with the general ideas behind process linguistics, I simply refer to Kempen & Hoenkamp (1982, 1984) ("incremental procedural grammar"), Lakoff & Thompson 1974 ("production grammar") and Bock 1982 (cognitive-psychological research into information processing contributions to sentence formulation). As far as the acquisition process is concerned, a number of researchers have proposed an alternative to the Chomskyan view of acquisition (Slobin (1966, 1984); Putnam 1975c; Derwing 1973). The Chomskyan view is called the "content" view of the acquisition device, holding that a child is born with the entire set of linguistic universals (plus evaluation procedures) and that he somehow uses this set as a grid through which the particular language he is exposed to is filtered (cp. Derwing 1973, 53). The alternative view is called the "process" (!) view, according to which

> ""the child is born not with a set of linguistic categories but with some sort of process mechanism -- a set of procedures and inference rules, if you will -- that he uses to process linguistic data." Under such an interpretation as this, then, any linguistic universal would be "the result of an innate cognitive competence (see 3.3.3, g.a.) rather than the content of such a competence" (Derwing 1973, 54, quoting from Slobin 1966).

Or, as Putnam sharply formulates it :

> "The theorems of mathematics, the solutions to puzzles, etc., cannot on *any* theory be *individually* 'innate'; what must be innate are heuristics, i.e. learning strategies. In the absence of any knowledge of what *general multipurpose learning strategies* might even look like, the assertion that such strategies (which absolutely must exist and be employed by all humans) cannot account for this or that learning process, that the answer or an answer schema must be 'innate', is utterly unfounded" (1975c, 115-116).

Note that the notions central to PL (*process* itself, and *heuristics*) are also central to this alternative view, which I adhere to but will not go into any further. I only point out that despite its modality-boundness process linguistics fits in nicely with approaches to production and learning.

3.3.2. Anthropocentrism

A principle that may sound trivial but has important implications for the approach is that the study of the comprehension process should be inspired by whatever we know about the *human* comprehension process (or can safely hypothesize about it); after all, the human being is the only "animal loquens", i.e. language-using being. This means that psycho- and neurolinguistic research plays a central role in attempts at giving a processual account of linguistic phenomena, as well as the AI research that also holds the anthropocentric view (see Schank & Riesbeck 1981, chapter 1 for a strong expression of the adherence to this view at Yale). The anthropocentric view also implies (once again) that it is not the computer and the formal models it uses that are at the center of an approach to parsing (as is the case in computational generative grammar). A type of reasoning that I consider fallacious in this context is the following:

A) the human being parses language efficiently

B) formal approach X allows efficient parsability on a computer

C) hence, approach X is a "correct", "realistic" approach, better than approach Y that does not allow such efficient computer parsing

It is clear that in this type of reasoning the human being is merely used to defend a certain formal approach. This reasoning falls prey to the strong AI assumption that computation equals cognition. Concretely, it assumes that the ease with which human beings understand language is comparable to efficiency norms for computer programs. This is an intuitively quite implausible assumption not supported by any evidence. To paraphrase Levelt (1974^{III}, 6), it is making computational virtue of psychological necessity. It is not just because we know more about computational complexity than about cognitive complexity that the former can be applied to the latter via metaphorical reasoning.

The question remains, of course, how *does* the human being parse sentences? The impenetrability of the human mind/brain seems to allow the wildest speculations about its organization. Considering existing psychological models and some careful introspection it seems reasonable to take the following view of the comprehension process:

1) In 2.2.2.2 I have expressed a bias for the interactive models over the autonomous component models. These models hold that the human being processes sentences on a word-by-word basis, going straight for their meaning in context. All sources of information that aid interpretation are involved in processes that interact freely without delays in availability of information. (This availability also refers to the partial interpretation reached at a certain point during comprehension.) The stress on the words as triggering these processes implies a bottom-up (data-driven) view of the comprehension process.

2) In the course of understanding, multiple meanings of linguistic elements may be active (as attested by psycholinguistic research, see 5.2.2), but ultimately context takes care of convergence on one meaning in normal language understanding (i.e. understanding that does not involve the metamode of processing competence (see 3.3.3), as opposed to (rare) cases of garden path sentences, remaining -- intended or unintended -- ambiguity, etc.). This implies a rejection of determinism in language processing if it means that there can never be more than one interpretation active during comprehension (cp. 2.3.4); if it simply refers to the convergence on one interpretation it is equal to the view expressed here. Note, by the way, that determinism allows efficient *computer* parsing (no time-consuming backtracking or space-consuming parallelism are needed), which partly explains its defense by e.g. Marcus or Berwick & Weinberg, but is contradicted by psycholinguistic research (see 5.2.2).

3) Memory mechanisms play a crucial role in this process: at least a large capacity long-term memory and a limited capacity short-term processing memory are involved. (How they are involved will be dealt with in 3.3.4 and in more detail when I present the computer model in chapter 4.)

3.3.3. Processing competence

As repeatedly announced in chapter 2, the incompatibility of "competence" in generative linguistics with the verbal behavior it is supposed to be the central component of makes a redefinition of competence necessary. As far as the comprehension process is concerned, a broad definition is the following: the ability of the human being to retrieve information (e.g. triggered by the incoming words of a sentence) and to dynamically relate it to and integrate it with other information. This may be information stored in long-term memory and/or in short-term memory (for the latter e.g. the information just acquired from the preceding linguistic context). (Note that it automatically follows from this definition that memory limitations are crucial to processing competence.) The ability itself consists mainly of intentional heuristic processes (searching for meaning). These processes are robust, flexible, automatic and not open to direct introspection. Two examples will be treated further in the text. We will see that the process of accessing information (words, idioms) is a uniform, automatic, exhaustive (i.e. multiple meanings are accessed in parallel) and independent mechanism (see 5.2 and 5.3.1). Second, the abstract process of the expectation-feedback cycle is one that captures understanding of linguistic units at all levels, receiving different instantiations (with a number of constraints) depending on the linguistic unit involved (see 3.3.4, 4.3.3.3 and 4.3.3.4). These two processes (automatic access and expectation-feedback cycles) are thus considered to be processing universals of the human mind.

Performance then is simply this processing competence at work in actual behavior. Linguistic knowledge is just one of the many sources appealed to in the course of understanding. When we take a closer look at performance, we can say that there are two kinds: one is "unconscious performance" by which I refer to normal undisturbed understanding; the other is "conscious performance" and refers to those types of understanding that require conscious awareness of linguistic knowledge (e.g. to give grammaticality judgments about sentences, to solve intended ambiguities or word play, to deal with garden path sentences, etc.). These two types run parallel to two modes in which processing competence can operate: in the normal mode the heuristics are at work and there is no awareness of linguistic knowledge being accessed (unconscious performance); in the other mode (less often at work), which I call the *metamode*, awareness of the linguistic knowledge accessed is required. Whereas the normal mode of processing competence involves the (innate) processing universals (heuristics), the metamode depends on acquired skills and knowledge, e.g. a capacity for word play, education in language(s), knowledge of linguistic theories. Note that in the metamode (as in the normal mode) there is no awareness of the processes themselves, but merely of structural characteristics of their output, language. This awareness can take the form of rules about structure, rules that can sometimes help solve problems occurring in normal mode but that are not in any way "used" in this normal mode. Hence, normal and metamode can interact, but metamode is secondary and less often appealed to. This view fits in nicely with the characterization of rules as "fall-back procedures" (in metamode) not active in the normal processing mode (see Butterworth 1982b and the discussion of his view -- proposed in the context of morphology -- in 5.2.6).

Linguistic intuitions (central in generative grammar) are relegated to the metamode of processing competence and have no important role to play in process linguistics. On the other hand, phenomena like hesitations, errors, shifts of attention (performance matters of no importance to Chomskyan competence) are considered as indications of processing mechanisms at work (or in trouble) and may be very important in an attempt at studying the underlying competence (both its normal and its metamode). (For instance, many "uh's" of spoken language are examples of hesitations in the normal mode (we are not aware of all the "uh's"); on the other hand, elaborate selfcorrections after completing a sentence are instances of metamode processing interrupting (c.q. monitoring) normal processing.)

If we compare the two interpretations of the competence-performance distinction in the Chomskyan tradition to the (re)interpretation given here, the following remarks can be made. What remains is that competence is related to an *idealized* model of performance, bracketing a number of factors that disturb the understanding process, such as fatigue, absent-mindedness, etc. (in the hearer), bad articulation or formulation, errors, etc. (in the speaker), noise (on the channel), etc. For the rest, there are few parallels. The content of competence comes closer to the *intended* content of Chomskyan competence in the early interpretation, viz. competence as an *ability*, as a "model of idealized performance". This seems to be the only interpretation that allows a smooth transition from competence to performance, guaranteed if competence is processing competence but not if it is merely static linguistic knowledge as in generative grammar. However, processing competence is not by itself a "model of idealized performance" (as the early Chomskyan interpretation suggested). Here the relationship between processing competence and performance comes closer to the type of relationship suggested by the canonical interpretation of competence as a "central component" to performance. Yet, here again, processing competence is not a static component consisting of linguistic knowledge, but it is the dynamic driving force of performance, a set of processing mechanisms. What process linguistics wants to do is to offer a model of this driving force and how it makes use of linguistic (and, for that matter, extralinguistic) knowledge. The king's part in the language play is no longer taken by the linguistic knowledge, but by the processing mechanisms; linguistic knowledge merely plays a servant's role.

In the context of linguistic intuitions, I want to point out that the formal notion of grammaticality of sentences (see the discussion in 2.2.2.2) has no place within process linguistics. For an approach to natural language understanding it is understandability (cp. the Chomskyan performance notion of acceptability) that counts. Grammaticality is in the first place determined by the way the (theory of) grammar is built. In so far as intuitions about e.g. grammaticality are said to be the data of the theory, an appeal is made to the metamode of processing ("Is this sentence grammatical?"). The necessity of considering understandability is a consequence of the process-linguistic notions of anteriority of process, modality-boundness, anthropocentrism and processing competence. Understandability can both be considered in the metamode *and* (more importantly) in the normal mode of processing. Dealing with understandability-in-metamode means (as with grammaticality) that one asks the question "Is this sentence understandable/acceptable?" Yet, this is a fairly uninteresting question because of the subjectivity of the judgments (cp. 2.2.2.2) and because it tells us nothing about the process of understanding. To overcome these drawbacks it is necessary to test understandability in normal mode, i.e. with techniques from cognitive psychology. Tests of whether a sentence is understandable can be done by checking a user's response to a sentence (giving an answer to a question, paraphrasing, etc.). An important element in these tests could be the *time* needed to reach an interpretation for a sentence (and respond to it) (cp. Clark & Haviland 1973); variations in measured time might be an indication of the degree of understandability. For reading, the study of eye-movements may also throw some light on the understanding process, with e.g. the frequency of backward fixations giving an indication of the degree of understandability (cp. 5.2.1). But more important than the theoretical considerations about understandability are its practical implications. It implies that the input range a computer model can handle should include more than merely theory-defined grammatical sentences. Systems that only include this subset of language (and this means most systems in the generative linguistic tradition (Marcus' parser, Berwick & Weinberg's revision of it, Bresnan & Kaplan's parser, etc.) cannot say that they model the human being at all, because this human being deals with ungrammatical (but understandable!) input without apparent difficulty (as well as with fragmentary input that is perfectly comprehensible in context). The model proposed in chapter 4 is e.g. capable of coming up with an analysis for sentences like "The man eat a peach", or "Man eats peach" showing that the relevant concepts were correctly determined (see 4.2.2.2); ellipses ("In the morning?") do not offer problems either. (A parser with its input range limited to "grammatical" sentences cannot handle any of these.) We will even see in chapter 5 that the model still exhibits human-like understanding behavior when it is "lesioned" in accordance with our knowledge about the disruption of language abilities in aphasics. All this follows from the fact that in keeping with the definition of competence as processing competence the destruction of linguistic knowledge used by the robust and flexible processes does not affect the processes themselves so that they can still operate when the knowledge they access is limited or disrupted. In a model focussing on the representation of static linguistic knowledge in the form of rules and leaving the processing to an inflexible parsing algorithm (as developed for formal language recognition), removing a single rule from that linguistic knowledge automatically makes interpretation of a large subset of the language the model analyses impossible.

3.3.4. The centrality of a dynamic lexicon

It is amazing how linguists, psycholinguists, neurolinguists and AI researchers have shown a growing interest in the lexicon over the last decade (see e.g. Testen et al. 1984, Hoppenbrouwers et al. 1985, Small et al. forthcoming). It looks like the field of natural language research is unified under cognitive science through the growing consensus about the importance of the lexicon. For process linguistics too the lexicon is in the focus of attention, be it in a way that will look more familiar to psychologists and AI researchers than to linguists. But let me first give a short overview of how the lexicon is central to a wide range of (generative and non-generative) linguistic approaches (see Taylor 1980 or Hoekstra 1980 for more historically oriented overviews).

Recent approaches in generative linguistics that attribute a more important role to lexical phenomena than used to be the case include lexical-functional grammar (Bresnan 1978, 1982), government-and-binding (Chomsky 1981, Berwick & Weinberg 1984), lexical-generative grammar (Diehl 1981) and head-driven phrase structure grammar (Proudian & Pollard 1985).

As already discussed in 2.3.4, lexical-functional grammar grew out of traditional transformational grammar (Chomsky 1965); it reduces the transformational component of such a grammar drastically and greatly enlarges the (neglected) lexicon and the semantic component. Thus, nontransformational rules -- lexical and interpretive -- play a great role in the model. Also, for the assignment of a functional syntactic structure to a sentence (an important element in the theory), the specification of the words in the lexicon is of crucial importance. In government-and-binding the importance of the lexicon is embodied in the "projection principle": linguistic representations have to obey the subcategorization properties of lexical items (and a number of constraints that take the place of phrase structure and transformational rules) at all levels of linguistic description. Diehl's lexical-generative grammar also rejects the rule-system conception of grammar, but instead of introducing constraints Diehl sees grammar itself as consisting of (a) a lexicon of fully specified lexical entries whose specification also contains information about combinability with other lexical forms and (b) redundancy rules (the only rules retained). Finally, head-driven phrase structure grammar is a refined and extended version of the closely related generalized phrase structure grammar (Gazdar et al. 1985) in which we have a similar massive relocation of linguistic information from i.c. phrase structure rules into the lexicon (subcategorization of lexical items is no longer handled by those rules, whose only remaining function is to map lexical entries to surface constituent order).

Beside these generative approaches, there are a number of non-generative ones (all of them actually reactions against generative theories) that equally stress the role of the lexicon. Gross' "lexicon grammar" (Gross 1979, 1984) does not consider words as basic syntactic units to which grammatical information is attached, but uses a huge number of simple sentences as dictionary entries that capture the idiosyncratic distributional and transformational properties of linguistic elements. Starosta's "lexicase grammar" (Starosta 1978) resembles Diehl's approach in that it also sees a fully specified lexicon (idiosyncratic lexical entries with their distributional properties + rules) as an adequate grammar of a language; syntax is seen as derivative: it falls out of the generalities found in the lexicon. Hudson's "word grammar" (Hudson 1984) takes a radically different approach to language, rejecting the dichotomy of syntax and lexicon altogether (cp. Langacker 1983) and viewing language as a network of linguistic entities (with the word as upper boundary) related by propositions.

To the extent that static information of morphological, syntactic, semantic (and possibly pragmatic (2)) nature usually attached to words in a lexicon is also involved in the characterization of words in process linguistics, the convergence of a number of linguistic approaches on the centrality of the lexicon is a development to be followed closely. Yet, all the approaches mentioned above still consider the lexicon as a static repository of information, however much this repository has grown. If a processing model completes the linguistic one, it still uses an extraneous interpreting mechanism that plucks lexical information from its lexicon as needed (as in Bresnan 1982, Marcus 1980, or Berwick & Weinberg 1984). Process linguistics takes a radically different view of "the lexicon" following from the importance attached to the words themselves for the comprehension process. Words

---

(2) See e.g. Haiman 1980 or Langacker 1983 for a discussion of whether a lexicon should contain pragmatic ("real-world") information (and come closer to an encyclopedia than to a simple dictionary).

are not seen as static containers of information to be drawn on by other, "more important" components (morphological, syntactic, semantic), but as triggers of dynamic processes that use the static information (which is of morphological, syntactic, semantic and pragmatic nature) in search of a meaning of the textual fragment they occur in (cp. the interactive model of language understanding (3)). Words are not servants, they are the masters of the comprehension process. In chapter 4 a detailed formal characterization of these processes will be given; let me concentrate here on their general nature. They can be subsumed under the general term "interaction": words interact with each other within the sentence they occur in (horizontal lexical-contextual interaction) and within the lexicon (vertical intralexical interaction) (4). The first kind of interaction makes ample use of short-term memory (containing pieces of built-up meaning); the second kind is more a matter of long-term memory. This latter kind of interaction also implies that the lexicon cannot be some kind of dictionary of unconnected words, but that it has to be organized in a way that densely connects linguistic elements in a way that involves more than just redundancy rules (i.e. rules that link (morphologically, syntactically or semantically) related lexical entries). Let me add right away that I will not go into vertical intralexical interaction and the internal organization of the lexicon in this book; horizontal lexical-contextual interaction has been the focus of attention. I am fully aware that the approach needs to be complemented by a motivated view of the internal organization of the lexicon. Certain aspects of this general issue will briefly be touched upon in sections to come, especially in chapter 5 where a lot of recent psycho- and neurolinguistic research into the (mental) lexicon and the way its information is processed will be discusssed. I refer the

(3) See also Hörmann (1982, 1983), Smith 1978, Miller & Johnson-Laird 1976, Miller 1978 for discussions that stress the need of a more process-oriented view of the lexicon with an eye on verbal behavior.

(4) The distinction between horizontal lexical-contextual interaction and vertical intralexical interaction runs parallel to the distinction between syntagmatic and paradigmatic ("associative") relationships (cp. Saussure 1949,170-175) among linguistic elements, (again) with the difference that Saussure's distinction fits in with a static structuralist approach and mine with a dynamic processual one.

reader to Smith 1978, Marslen-Wilson & Tyler 1978, Cuyckens 1982, Hörmann 1983, or Schreuder & Levelt 1978 for overviews of models of "semantic memory" (the internal lexicon) in psychology and/or AI; see also Katz & Fodor 1963 -- often referred to in the overviews -- for an early linguistic attempt at a feature-based semantic theory in which the lexicon gets full attention. It is also noteworthy that the (semantic) organization of the lexicon used to be the object of (prechomskyan) linguistic research into semantic fields, research that could be revived in the context of the growing interest in the internal organization of the lexicon.

Before I go into lexical-contextual interaction, I want to stress a few important implications of what has been said so far. First, PL does not consider the lexicon as a highly unspecified or unimportant checklist to be used during the application of general rules from other components (syntactic, semantic, etc.) of a system, but as the driving force of the system, with the words highly specified for their phonological, morphological, syntactic and semantic characteristics to be used by the processes of lexical-contextual interaction during meaning determination. This, in turn, implies the stress on idiosyncrasy instead of generality: there is, for instance, no top-down application of general rules imposing structure on a sentence (often to be revised when the concrete words are considered), but the words themselves gradually build up this structure through complex interactions with each other. (A conviction lying behind this is that any system not trying to deal with the idiosyncratic nature of linguistic entities right away will eventually crash on these idiosyncrasies (cp. Small 1980, Starosta 1978).) Related to this is the assumption of breadth-wise complexity and depth-wise simplicity of language. In many approaches using (context-free) phrase structure rules, the application of these rules often introduces unnecessary structure and depth, as a consequence of which many relations between linguistic elements are lost (cp. the problems of "long-distance dependencies" between elements wide apart in the trees used in generative grammar) (5). In the approach

---

(5) See e.g. Hudson 1984 or Van Langendonck 1985 for a defense of a dependency approach over a constituency approach. A dependency approach rejects the rigid tree structures used by a constituency approach in favor of an approach incorporating more directly the relevant relationships among linguistic elements (a dependency structure is a less res-

taken here, language is considered more as a "flat" phenomenon, with interactions spanning the distances between related elements more easily in a dynamic way (the short-term memory taking care of holding the pieces of semantic structure gradually built up and possibly needed later on as the analysis proceeds). This also shows the difference between a static approach to language as a timeless phenomenon and a dynamic approach considering its time-bound nature: a static approach imposes structure "postfactum" onto completed sentences, unable of accounting in a natural way for relations between separated constituents, whereas a dynamic approach solves these dependencies as it moves to and fro in the sentence. This view also fits in nicely with the limitations of short-term memory: it cannot contain many pieces of structure (explaining, for instance, the difficulty of the human being to deal with center-embedding, see also 2.2.2), but the processes working with and on these pieces are not bound by the limited capacity of this memory; there are general mechanisms of feedforward (expectation) and feedback that apply freely and at great repetition across pieces of uninterpreted structure and newly incoming information (words).

With this last remark we are back to lexical-contextual interaction. In essence, it consists of expectation-feedback cycles (EFC's for convenience) crucially involving the use of memory (cp Schank & Birnbaum 1984 for a comparable discussion of the importance of expectations and memory in understanding). As already mentioned a number of times, the EFC is a general processing mechanism, intentional and heuristic in nature (meaning-searching) and taking care of several aspects of the understanding process. Microprocesses dealing with subparts of the overall process are concrete instantiations of the EFC. The subparts focussed on here are:

1) word sense disambiguation in context (especially of content words)

2) dynamic buildup of constituent meaning (featuring especially the function words like articles and prepositions)

3) dynamic buildup of overall sentence meaning (featuring especially the verb)

---

tricted graph structure than a tree). Process linguistics interprets dependency in a dynamic way (see further in the text).

Before I deal with these subparts in turn, the EFC needs some more elaboration. That expectations play a role in understanding is intuitively obvious. However, they are often seen just as states of mind (i.e. as intentional states, states about something meaningful in the world) and not as dynamic processes: expectations can be *fulfilled*, *thwarted* (6) or they can simply *die away*. This is where the F of EFC comes in: expectations can get positive, negative or no feedback respectively. Hence, they do not stand on themselves but are crucially linked to the response that answers them. Expectations project things in time at specific moments during processing (feedforward), and at later moments there is some kind of response to them (feedback). This is what makes the EFC a crucially time-bound process and not a mere static state of "waiting". Chapter 4 gives a detailed formal characterization of how EFC's can be simulated in a computer model; here I just want to introduce an abstract notion that helps in understanding how they "work", viz. the *time triangle*. It is a spatial representation of EFC's. Although it is hard to be aware of the dynamic (temporal) nature of a phenomenon if it is represented spatially, there seems to be no other way of rendering it on paper, so I stress that time triangles capture a *process* over time in spite of their static appearance (cp. 4.3.3.3 and the notion "dynamic caseframe" in 4.3.3.4). Figure I contains two types of time triangles. (The differences between both are not important right now). The vertical axis represents the time course of the EFC process (t1, t2, etc.; time elapses from the top of the Figure down). The horizontal axis needs a little more explanation. Recall that words are seen as active entities involved a.o. in the creation of expectations upon their arrival in the hearer. W1, w2 and w3 represent such words; the *a*-index indicates their arrival in the hearer, upon which they immediately trigger whatever processes associated with them. The *m*-index indicates that they have entered the memory of the hearer. Now, if a word triggers some expectation (e.g. an article triggers an expectation for an "entity", a semantic term for "noun"), it then remains in a sleeping state until feedback

(6) It is interesting to note here (as Schank & Birnbaum (1984, 240) point out) that expectation failures could be the basis of generalization and learning processes. Hence, one more link between the approach to understanding and a possible complementary approach to learning. I will reconsider this issue in a little more detail in 4.2.3.

reaches it with the awaited concept. Hence, the horizontal axis represents the words in their possibly multiple states of activity (triggering an expectation ("exp") at t1 and receiving feedback at t5 (in the left part of Figure I), upon which another microprocess may be triggered). Now, let w1 = *the*, w2 = *big*, w3 = *car* (7). At t1 *the* enters the understanding process, and triggers an expectation for a concept of type entity; it then goes to sleep. At t2 *big* enters the process and also triggers such an expectation. At t3 *car* arrives, a concept of type entity is created, and feedback occurs to w2 and w1 (which can continue their business now that their awaited information has arrived). In the left triangle, it is suggested that feedback to w2 and w1 occurs at different points in time (t4 and t5), which essentially means that feedback does not happen in parallel (a theoretical possibility). In the right triangle (also a theoretical possibility), creation of the w3-concept and the feedback to w2 and w1 all happen at t3, hence in parallel. Note the different shapes of the two triangles, especially the rectangular character of the right one. In Appendix 3 a figure is shown containing the "history" of all the EFC's that occurred during processing of a sentence; such a figure consists of a number of time triangles in one another with subtriangles showing processing of constituents, and the overall triangle (driven by the verb) showing overall sentence processing (Figure II abstractly represents such a "triangle forest").

(7) See 4.3.3.3 for a more detailed discussion of the processing of a simple noun phrase.

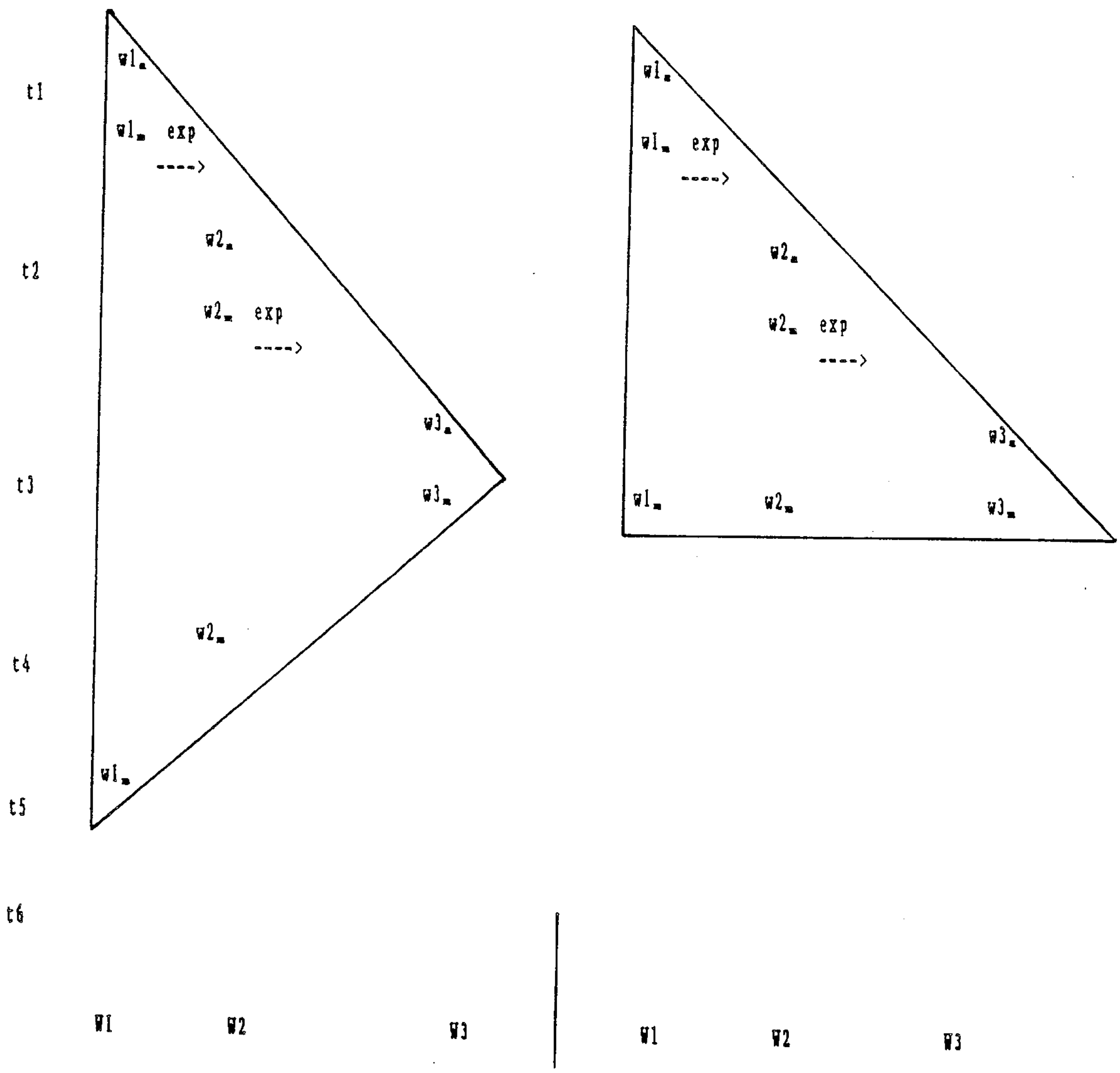


Figure I. Time triangles representing EFC's
(left triangle: serial feedback; right triangle: parallel feedback).

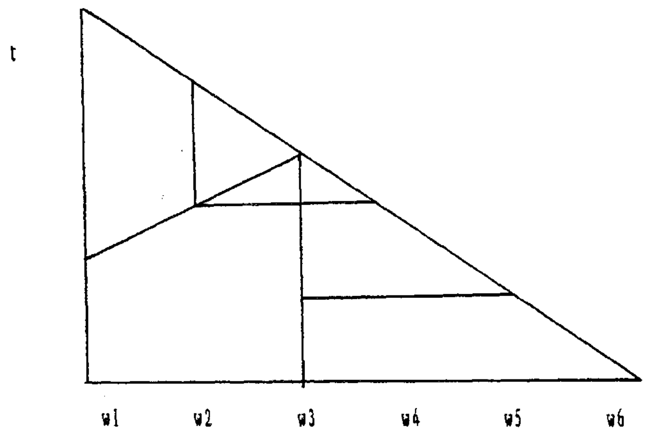


Figure II. Triangle forest representing sentence processing.

Back now to the subparts of the overall understanding process focussed on in WEP. The first subpart is word sense disambiguation. The importance of disambiguation follows partly from the fact that recent psycholinguistic research has shown that upon access of a word *all* its meanings are briefly accessed (see 5.2.2), with postaccess processes (the ones I concentrate on) taking care of determining the contextually appropriate meaning. A remark to be made here about the place of ambiguity in PL is the following: whereas it is acknowledged that ambiguity is present during the process of meaning determination, it is important to see that the stress is on *dis*ambiguation and the convergence on one meaning (cp. 3.3.2). When sentence processing is finished, ambiguity has gone (conform to the experience that ambiguity hardly plays an overt role in verbal behavior, suggesting also that much of the disambiguation process happens very fast and/or automatically, i.e. without the language user's explicitly being aware of it -- cp. the normal mode of processing competence). This implies a criticism of the way ambiguity is often seen as a static phenomenon in linguistics. For instance, the "multiple ambiguity" of the well-known sentence "I saw a man on a hill with a telescope" can be considered as an artifact of a static (syntax-centered) approach. In context, there is ultimately no ambiguity; lexical-contextual interaction dissolves it dynamically as sentence processing goes on. Examples of how lexical-contextual interaction disambiguates polysemous and/or homonymous words (by probing the context as present in memory) are given in chapter 4; it is even so that the multiple meanings of a word form the "skeleton" around which the disambiguation processes are built.

For the second and third subpart of the understanding process, the distinction between content and function words needs to be looked at. It is a distinction often turning up in linguistic literature (see Carlson & Tanenhaus 1984 for a short historical overview) and a hot topic in psycho- and neurolinguistic research trying to determine whether the intuitive and descriptive differences between the classes have psychological reality and if so, how (see chapter 5).

Since it cannot be denied that most words have a content as well as a (syntactic) function (cp. Fronek 1982), the terms "closed class" and "open class" (for function and content words respectively) capture the distinction more precisely since they refer to an empirically correct reality (articles, prepositions and conjunctions are not subject to productivity phenomena, whereas the other classes can still be extended); yet, I will continue to use the function versus content word distinction as it seems to be the most commonly used. In Figure III I try to show that the distinction is more a matter of relative weight of both "ingredients" than of an either-or distinction (8):

---

(8) The careful reader may have noticed that the pronouns are not on the scale of function/content words; although it is not so important for the train of thought developed here (different types of words will have different structural and especially processual aspects) I left them out because of their special status in language (they cover a wide variety of subclasses that have specific relations to the other word classes and to larger linguistic constituents; see also Blanche-Benveniste et al. 1984). Another problem with the scale is that bound morphemes should have a place in it too (with inflectional morphemes on the "function > content" side and derivational ones on the other side).

```
FUNCTION > CONTENT                        CONTENT > FUNCTION

---------->--------------->-------------->------------------>

     art    prep     conj      verb    adv     adj      noun

     CLOSED       CLASSES     |      OPEN           CLASSES
                              |
                              |
interaction type:  mainly     |  interaction type: mainly
                   syntactic  |                   semantic
interaction scope: local      |  interaction scope: global
```

Figure III. Function and content words.

In general, we can say that function words trigger processes of pattern expectation (syntactically speaking) or concept expectation (semantically speaking) and initialization; the articles can be seen as the prototypical function words. They project the semantic expectation (and initialization) of a concept, realized syntactically by the possible sequence of adjective(s)+noun. If the article is indefinite, there is no feedback to concepts processed earlier and present in memory; if the article is definite, however, feedback is triggered and a search through memory may eventually link the concept at hand to one already introduced before. As we move along the scale from left to right, interaction becomes less local, and more semantic in nature. A preposition, for instance, also expects a concept, but it will have to interact with the noun phrase/concept following it, or even with the other sentential elements in order to determine the function of the prepositional constituent (e.g. in "in the summertime", "in" carries a.o. a mild expectation of a time constituent, projects this expectation to the concept that follows, the latter feeding back an affirmative answer to the expectation of the former through its semantic characteristics). Content words mainly bring in semantic information, then, often feeding this information back to the expectations set up by the function words. The prototypical content word (at the end of the scale) is the noun since it seems to be the linguistic element with the richest semantic content. (As to its (syntactic) function: it shares its function as (right-located) head of a larger constituent (viz. a noun phrase) with adverbs and adjectives.) The most important class (at the center of the scale) is formed by the verb that brings in a rich semantic content and at the same time carries the responsibility for the interactions that lead to the correct assignment of (semantic) cases to the concepts (to be) processed in the sentence. In short, the lexicon with its two types of words (function & content) naturally reconciles syntax and semantics: function words take care of "low-level" (constituent) syntax, content words (especially nouns) bring in meaning, and the verb takes care of sentential syntax. In 4.3.3.3 a detailed example will be given of noun phrase processing (subpart 2 of the comprehension process), and in 4.3.3.4 the important notion of "dynamic caseframe" (a processual encoding of case-searching by the verb) is explained in detail (subpart 3 of the overall process).

A word-based approach as advocated by PL cannot avoid issues of morphology and the status of idioms. I postpone the discussion of these two matters till chapter 5, since the PL view of morphology and idioms is closely linked to the psycholinguistic research into the way morphologically complex words and idioms are processed by the human being. Let me only note here that bound morphemes are -- just like free morphemes -- considered as active entities interacting with the stem they are attached to in an attempt to find out their meaning/function in context. The English *-s* morpheme, for instance, interacts with its stem to determine whether it is a verb-singular morpheme or a noun-plural morpheme. (Lexical-contextual interaction also pervades the morphological level of analysis.) As for idioms: they can be considered as strings of words with lexical interaction among them reduced to an absolute minimum, which gives them the status of separate words by themselves.

Finally, returning to the matter of language acquisition touched upon in 3.3.1, it is interesting to note that the stress on the lexicon (or rather, on lexical-contextual interaction) can make the process view of learning more concrete. The process is seen essentially as *learning words in context* (cp. Bolinger 1965, 570), i.e. extracting their static and dynamic characteristics from linguistic input, something a human being does mainly as a child but also as an adult (vocabulary extension is a lifelong process).

3.3.5. Summary: generative grammar versus PL

Figure IV opposes the most important principles and assumptions of generative grammar (plus its computational realization) to those of PL by way of recapitulation of the issues discussed so far in chapters 2 and 3.

generative grammar	process linguistics
- anteriority of linguistic structure: study of timeless, static structures (spatial & vertical: depth-wise complexity)	- anteriority of cognitive process: study of time-bound, dynamic processes (temporal & horizontal: breadth-wise complexity)
- roots in formal language theory	- roots in psychology (anthropocentrism) and AI
- competence-performance: (structure) (process) unbridgable gap	- competence-performance: (process) (process) smooth transition
- single-representation hypothesis	- modality-boundness
- study of metalinguistic judgment, intuition (derived behavior)	- study of comprehension (primary behavior)
- stress on syntax and generality	- stress on lexicon and idiosyncrasy
- grammaticality	- understandability
- language universals content view of acquisition	- processing universals process view of acquisition
- rules as descriptive/explanatory devices	- cognitive processes as descriptive/explanatory devices
- view of performance:	- view of performance:
+ rules are used	+ rules are fall-back procedures (metamode)
+ autonomous component model (syntax first)	+ interactive model
+ algorithmic process	+ heuristic processes

Figure IV. Generative grammar versus process linguistics.

Note, finally, that PL as a cognitive-scientific linguistic approach uses a lot of concepts from the linguistic tradition, and as such is certainly related to it. The traditional word classes are used, syntactic constituents, notions of dependency, distribution and syncategorematicity (i.e. the phenomenon that a word's meaning depends on the other words it is used with (9); cp. "a good husband", "a good meal", "a good movie", with *good* receiving a different meaning depending on the concept that follows it), etc. However, the stress here is on the dynamic way these notions are "at work" during sentence comprehension, and not on the (traditional) static way they are used in postfactum sentence analysis. Referring to Katz & Fodor (1963), it is also interesting to see how the "projection rules" of their semantic theory get a dynamic interpretation in PL. Projection rules are rules that must complement a fully specified dictionary and whose function it is to "select the appropriate sense of each lexical item in a sentence in order to provide the correct readings for each distinct grammatical structure of that sentence" (1963, 183), or, alternatively, to "take account of semantic relations between morphemes and of the interaction between meaning and syntactic structure in determining the correct semantic interpretation for any of the infinitely many sentences which the grammar generates" (ibid.) (10). In PL the processes of lexical-contextual interaction take care of this: static rules are replaced by dynamic processes. Still, it remains very important that detailed descriptions be made, e.g. of the distribution of articles or of the idiosyncratic nature of the way certain verbs can be passivized (see Gross 1979, 1984), since it is this information that the processes of retrieval and integration through feedforward and feedback work with.

---

(9) See also Anderson & Ortony 1975 or Miller 1978 for a treatment of syncategorematicity.

(10) Note that this quotation shows that Katz & Fodor's theory was developed to complement the syntactic component of a (transformational) generative grammar: semantic interpretation comes after syntactic generation.

3.4. The bridge to practice: some reflections

As said in the introduction to this chapter, I have only sketched the main principles of process linguistics. Many aspects (e.g. the study of the lexicon, or the notion of understandability) need further elaboration within the general process-oriented framework. In spite of the sketchiness I believe that this framework offers a valuable alternative to generative linguistics, fitting in nicely with a lot of research in AI and cognitive psychology.

As far as the AI research is concerned: chapter 4 will deal with a computer model that works in accordance with some of the process-linguistic principles. Before I present it, it is useful to look at what aspects of PL the model does (and especially, does not) incorporate. The system is an interactive model (see 2.2.2.2) that analyzes written language without explicit use of rules; hence, we recognize the principles of anteriority of process and modality-boundness (the study of comprehension). Futher, the system stresses the importance of the (dynamic) lexicon + idiosyncrasy over syntax + generality. It also wants to be a model of human language understanding (the principle of anthropocentrism), in which the abstract heuristic process of the EFC plays a crucial role. As far as modeling expectations is concerned (the E of EFC), the stress is on bottom-up (word-bound) expectations, and not so much on top-down ones (those suggested by e.g. the general subject of a text, the knowledge of the reader about that subject, etc.) (11). This limitation is linked to the stress on horizontal lexical-contextual interaction in the computer model (vertical intralexical interaction and other mechanisms involving long-term memory -- like top-down expectations -- are not in focus), as well as to the consideration of isolated sentences in the first place. We will see that the model makes it possible to deal with larger fragments of text than just sentences, but in this early stage of application of the model to other languages than English (the one it was designed for in the first place), studying isolated sentences was a necessary limitation. I also stress that the way I chose to work with the model was to try and apply it concretely to Dutch "from

---

(11) Work in the Yale tradition in AI has concentrated partly on expectations of this kind (see e.g. Schank & Riesbeck 1981, Schank & Birnbaum 1984).

scratch" (and not to elaborate it further for application to English by e.g. working out the details of a fully developed long-term semantic memory). Finally, the model concentrates on simulating the normal mode of processing, with possibilities to incorporate metamode processing provided but not implemented (see 4.2.3 for further discussion).

As far as research in cognitive psychology (including psycho- and neurolinguistic research) is concerned, it will be dealt with in chapter 5. The presentation will take the form of a confrontation of some of the specific characteristics of the computer simulation with related research in cognitive psychology. A lot more could be said about the relation between process linguistics and cognitive psychology, but here again limitations were necessary. An aspect that will hardly be touched upon, for instance, is the important *time* issue in processing. More concretely, of a process like the EFC the static part of "having expectations" has received some attention in cognitive-psychological research (see e.g. 5.2.3), but the dynamic feedback part including how long a language user is capable of or willing to wait for it has hardly received any. Especially for spoken language understanding the duration of pauses (and possibly the conscious manipulation of time during conversation based on the knowledge that the listener is waiting for specific elements) deserve closer study.

To conclude chapter 3, and by way of transition to the next, an informal example now follows of the (micro)processes at work in the language user during sentence comprehension. Only bottom-up processes triggered by the words themselves as they enter the mind of the reader/listener (and give rise to the formation of concepts) are rendered here. If the sentence occurred in context, non-word-bound expectations (triggered by the preceding context, for instance) would have to be added. In chapter 4 some subprocesses will be dealt with in more detail, and Appendix 3 contains a trace of a complete analysis of a sentence as simulated by the computer program.

A...

initialization and
expectation of a
concept of type "entity"
(a noun(phrase))

...good...

meaning addition
to the concept being built up
("something positive");
expectation of the main element
of the concept still active
and strengthened

...school...

strong meaning addition
to the concept by the arrival
of the noun;
feedback to "a" and to "good"
for further meaning concretization of
the concept (= dynamic interpretation of
syncategorematicity);
expectations by "a" and "good" dissolved

...should...

initialization of a verb concept;
meaning addition of modality;
strong expectation of a verb

...offer...

verb meaning addition
with feedback to "should";
the verb triggers a complex
process of searching for
caseframe-fillers;
the concept "a good school"
is assigned the agent role
(see 4.3.3.4);
expectation of semantic

object concept

...a...

see above;

...lot...

concept formation and
concretization;
fulfilment of object
expectation through
interaction with "offer"
(feedback);
expectation of a
postmodifying phrase

. <period>

last
expectation thwarted;
sentence processing
completed.

CHAPTER 4: PROCESS LINGUISTICS, WORD EXPERT PARSING, AND DUTCH

"Something is happening in the *course of* the processing of information, not only as a result of this process. Is it happening *inside* the lexicon?"
(Hörmann 1983, 9-10).

4.1. Introduction

In this chapter, process linguistics comes alive. I will describe how an existing computer model of natural language understanding, the Word Expert Parser (*WEP* for convenience, Small 1980) works for a small subset of English, and how it was revised and extended in an application to a larger subset of Dutch. The discussion will focus on how words are turned into processes, what these processes look like (the WEP representation language), how they implement important process-linguistic notions (especially the EFC at the heart of lexical-contextual interaction), and how they handle some typical characteristics of Dutch.

Before I start discussing WEP, a metascientific note is in order. As the discussion goes on, it will become clear that some of the principles that lie at the basis of WEP are also principles of process linguistics, or that principles of process linguistics inspired revisions of the model. (Just like psycho- and neurolinguistic research inspired/inspires revisions, see chapter 5.) A reader might ask the question then of what came first, the linguistic framework, the computer model, or the psycho- and neurolinguistic considerations? Referring to the cognitive science framework, I would say that the question is irrelevant. There has been a continuous interaction among the insights from the different sciences involved, so there is no "firstness" to any of the perspectives on natural language understanding (linguistic, AI, psychological). It is merely because I am a linguist by training and because I consider it important to draw the attention of linguists to the processual nature of language that the linguistic part has come first (i.e. in space; chapters 2 and 3) and is the most extensive part. Note that the important idea of interaction (the interactive model of

NLU, lexical-contextual interaction in PL and WEP) also pervades the metascientific (cognitive-scientific) level. On this level too, it is the interactions among the subdisciplines that are important and that lead to a truly cognitive-scientific approach integrating all perspectives in a (hopefully) coherent whole. Maybe this way of doing science is also an antidote against building e.g. linguistic theories *first* and without consideration of other perspectives, and trying to find justifications (in psychology, AI) for them *after*, with the closed level fallacy and unbridgable gaps between the approaches as a negative result.

To give an example of how the interactions among the different perspectives can take strange forms: the importance of the EFC in language processing had not been so clear to me until a toy extension to the WEP program was made to visualize the way control passes from word-expert to word-expert (see below for the details). The "history diagram" (shown in Appendix 3 for an example WEP run) neatly showed the time-triangles and led to the EFC abstraction.

4.2. Word Expert Parsing

4.2.1. General principles

As Small (1980, 26) points out, the WEP approach was inspired by an anthropocentric AI view of natural language understanding:

> "The WEP approach was originally motivated by observations about human language processing on one hand and computational efforts to engineer and/or model the process on the other. Certain phenomena of human language use have particularly influenced this perspective, such as the relative ease with which people understand idioms and collocations, organize and select appropriate word senses, and perform reference. An important computational influence has been the difficulty of incorporating such mechanisms into computer programs organized along traditional (rule-based) lines. Viewing language comprehension from the perspective of individual words demystifies many classical semantic complexities and suggests an entirely different set of language analysis mechanisms based on distributed lexi-

cal control".

Before WEP and with the exception of Wilks' preference semantics (Wilks 1973 and passim), most existing parsing systems had largely ignored the idiosyncratic riches of individual words and the ease of the human being to effortlessly "pick" the right contextual meaning of a word. The common approach to NLU was (is?) to put down generalities in syntactic and/or semantic rules, and to treat words as tokens that simply participate in comprehension by virtue of their inclusion in these rules (cp. Rieger & Small 1981). WEP somehow played *enfant terrible* in the parsing community and proposed a totally different model organization aimed in the first place at solving the sense-selection problem (1). Instead of having a number of components (morphological, syntactic, semantic,...) consisting of static rule structures spanning sentence constituents or complete sentences, with some central interpreter taking care of the application of these rules to the input, the words themselves are considered as active agents (word-experts) triggering processes that idiosyncratically control the whole parsing process. This process involves continuous interaction of a word with a number of knowledge sources in memory: the words in its immediate context, the concepts processed so far or expected locally, knowledge of the overall process state, of the discourse, and real-world knowledge. These knowledge sources are not invoked uniformly by a general interpreter, but are accessible at all times by the word-expert processes throughout the overall process of sense discrimination in order to enable the experts to eventually agree on a context-specific semantic interpretation of a fragment of text. It will be clear from this description that parsing is not just the assignment of a syntactic structure to a sentence, but full understanding of language in context. In 4.2.2.2 we will see what kind of semantic/conceptual structure WEP builds as a side-effect of the overall understanding process.

As we have seen in chapters 2 and 3, variations upon this

(1) We will see in the application to Dutch that the model does not only deal successfully with the sense selection problem, but that its mechanisms are powerful enough to deal with matters of syntax and overall sentence semantics (see 4.3). These aspects were not focussed on by Small, which is understandable considering the view of language understanding as word sense discrimination.

view have since been taken by other researchers, be it independently of WEP (e.g. Marslen-Wilson & Tyler's or Just & Carpenter's interactive models) or influenced by it (Cottrell 1985, Hirst 1983), and computer models are being developed in accordance with this interactive view.

Beside viewing parsing as a word-by-word (data-driven), highly interactive process, Small also stresses that parsing is deterministic (1980; 12, 20). For this feature, and how it is combined with a wait-and-see strategy to make its implementation possible (the use of expectations) he refers to Marcus (1980) (taking his distance from the syntax-first view of Marcus' approach though). I will take up the matter of expectations and their role in WEP in 4.2.3; since I criticized the idea of determinism in chapters 2 and 3 a comment is in order here. When Small says that WEP is deterministic, he explicitates the statement by saying that it uses no backtracking (in contrast to ATNs, for instance). Since it is a fact that WEP never backtracks (i.e. never takes a decision back), I have no objection against calling the program deterministic in this sense. Yet, Marcus (and, for that matter, Berwick & Weinberg also) consider parallelism as a non-deterministic element, and this is where WEP cannot be said to be "deterministic" (or even, where determinism becomes an irrelevant notion). In WEP, parallelism enters the picture in several ways: subprocesses of word-experts were designed to be run in parallel, all word senses (even across syntactic categories) are accessed in parallel, and word-bound expectations (though not binding or linked explicitly to the control structure, see 4.2.3) cover a range of several possibilities kept active during parsing. (All these aspects of WEP will become clear in the course of the discussion.) Hence, I would say that WEP is not really deterministic, but at the same time that the notion of parallelism (presumably very important and pervasive in human cognitive/neural functioning) renders the notion irrelevant: why insist on determinism (absence of parallelism) if cognitive functioning largely happens in parallel?

In short, in WEP parsing is seen as a data-driven (word-by-word), highly interactive process using no backtracking, some forms of parallelism, and an important wait-and-see strategy; the whole parsing process is not coordinated by a rule-applying central interpreter but by the individual words that are viewed as active knowledge sources (word-experts). I will now take a closer look at matters of representation and implementation that turn these principles into a working program.

4.2.2. Representation

4.2.2.1. Sense Discrimination Language

As already said in 4.2.1, the sense selection problem was the point of departure for the development of word-experts; this explains their "raw" representation. Informally, word-experts) can be viewed as sense discrimination networks consisting of nodes of context-probing questions and arcs corresponding to the range of possible answers; each of the leaves of the network represents a specific contextual meaning of the word in question reached after network traversal during sentence processing. Figure I shows such an informal network for the highly ambiguous word *deep*. The left half of the net represents its adjectival usages ((1) through (4)), the right half its nominal ones ((5) through (7)). Meaning (1) would be arrived at in a context like "The deep philosopher likes Levinas", (2) in a context like "He throws the ring into the deep pit"; meaning (5) would be chosen in the context "The giant squid still lives in the deep", etc. for the other usages.

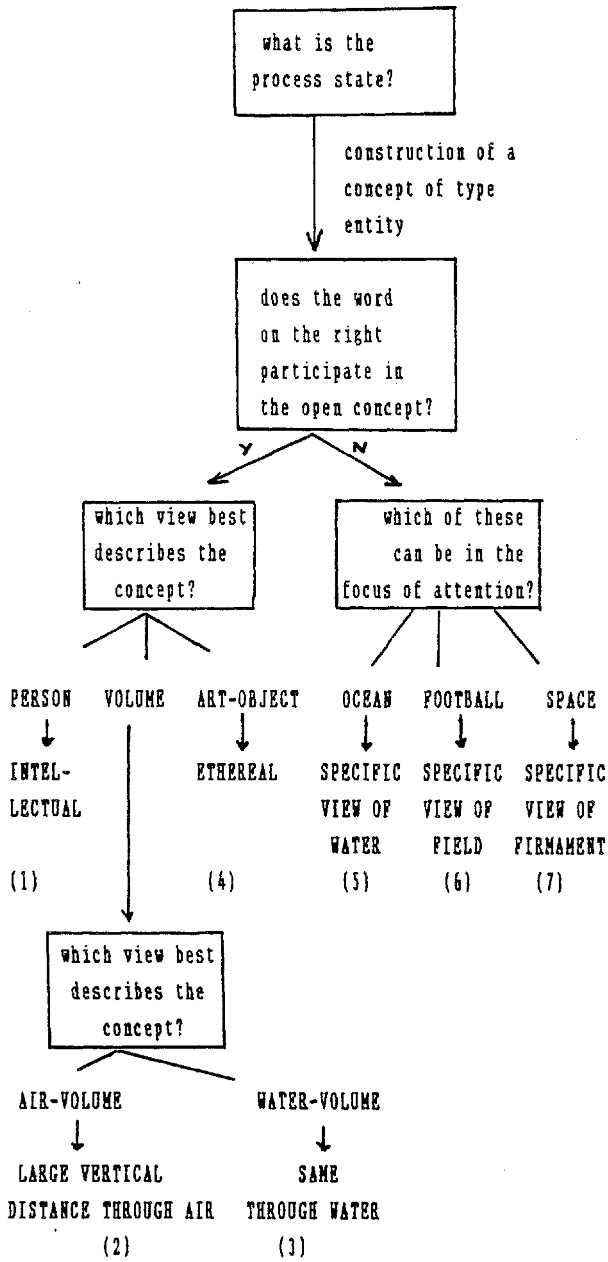


Figure I. An informal sense-discrimination net
(adapted from Small 1980, 48).

Linguistically speaking, the difference between such a net and a dictionary entry is worth pointing out (though there are similarities too, of course). Whereas dictionary entries will only contain the leaves of the net (in some orderly way), the stress here is more on reconstructing the contexts the words were extracted from. These contexts are not so much static linguistic contexts, but rather the possible processing contexts of the words. To some extent the branching in the net corresponds to static linguistic distinctions (e.g. the word-class membership of *deep* comes as an answer to the concept-processing question about the word to its right), in other cases branching happens after questioning the mental lexicon in memory (the VIEW question, further explained below). The informal nets can be of great use in the design and development of word-experts, but have to be translated into the formal declarative representation (to be used procedurally by the WEP process). Such a representation is a graph composed of designated subgraphs without cycles. Each subgraph (an "entry point") consists of question nodes that probe the multiple knowledge sources mentioned above and action nodes that build and refine concepts, keep track of their lexical sequences, etc. These questions and actions constitute the formal representation language for word-experts; the constant refinement and revision of this language forms one of the most important aspects of the WEP research. Figure II shows the skeleton of the word-experts and Figure III shows a completely specified word-expert that implements part of the network of Figure I; Appendix 1 contains the completely specified syntax of the WEP representation language in Backus-Naur form.

```
<word-expert  <name>
     <entry0
           (node0: <node type>  <node body>)
           (node1: <node type>  <node body>)
                    ...
           (noden: <node type>  <node body>)
     >
     <entry1
           (node0: <node type>  <node body>)
     >              ...


     <entrym
           (node0: <node type>  <node body>)
                    ...
     >
>
```

Figure II. Word-expert structure
(Note: <node type> = 'question' or 'action').

```
<word-expert   deep
   <entry0     (node0:question  signal s0
                      (*entity-construction* node1)
                      (* node2) ); "*" is a catchall
                                       matching anything
               (node1:action  (declareg)
                              (continue entry1) )
               (node2:action  (openg *entity-construction*)
                              (declareg)
                              (continue entry1) )
   >
  ; from here onwards 'node' is abbreviated to 'n', 'question' to 'q'
  ; and 'action' to 'a'
   <entry1     (n0:a  (buildc concept0 entity
                               (oneof PERSON ARTISTIC-OBJECT VOLUME ANYTHING))
                      (await concept entity
                              (filter concept0)
                              (bindconcept concept1)
                              (report here)
                              (wait group 1)
                              (continue entry2)
                              (else entry3)) )
   >
```

```
<entry2     (n0:q  view concept1
                   (ANYTHING n4) (PERSON n1) (ARTISTIC-OBJECT n2) (VOLUME n3))
            (n1:a  (refinec concept1 INTELLECTUAL)
                   (report concept1) )
            (n2:a  (refinec concept1 ETHEREAL)
                   (report concept1) )
            (n3:a  (refinec concept1 VERTICALLY-SPACIOUS)
                   (report concept1) )
            (n4:a  (refinec concept1 DEEP)
                   (report concept1) )
 >
 <entry3    (n0:a  (buildc concept2 entity)
                   (refinec concept2 DEEP-ENTITY)
                   (link concept2)
                   (closeg *complete-entity*)
                   (report concept2) )
 >
>
```

Figure III. Word-expert for "deep" implementing part of Figure I (adapted from Small 1980, 238).

I will briefly discuss the questions and actions that were grouped as "Sense Discrimination Language" (SDL) in Small 1980; in 4.2.3 the actions grouped as "Lexical Interaction Language" (LIL) will be discussed, since they are more closely related to implementation matters. For the interested reader, Appendix 1 contains the complete specification of the syntax of the WEP representation language.

Figure IV contains the SDL questions and actions with a short description of their semantics, explained further in the text (2).

(2) Two actions have been left out in Figure IV: BINDC IMMEDIATE and BINDC ASPECT. BINDC IMMEDIATE is only a matter of giving a concept another name locally; BINDC ASPECT is hardly ever used and not further discussed here (see Small & Lucas 1983, 43-44 for a description of its purpose).

QUESTIONS

?SIGNAL : probing incoming control signals (which are either the process
state or idiosyncratic signals from other experts)

?VIEW : probing the proximity of two concepts
?BOUND : checking success or failure of an attempt to find a piece of
information in memory (see the BINDC action below)

?LITERAL: probing the word of an expert process
?IDIOM : probing the lexical sequence of a concept to see if it is
an idiomatic expression or not

ACTIONS
----

a) Bookkeeping actions keeping track of linguistic sequences

OPENG : start a lexical sequence
DECLAREG: participate in the current sequence
CLOSEG : terminate the current sequence

BREAKG : signal the end of a sentence

b) Actions dealing with semantic concept structures

CREATEC: create a simple concept
BUILDC : build a complex concept
REFINEC: refine a concept constructed earlier
STOREC : store a concept in active memory
LINK : incorporate the current lexical sequence
into a concept structure

ROLEC : specify the role of a concept (e.g. concept1 is object to concept2)
ASPECTC: specify the slots of a concept (to be filled by other concepts that
fit in its frame; e.g. concept2 takes concept1 as an object)

c) Expert-internal control flow

NEXT : branch within an entry point to another node
CONTINUE: execute another entry point concurrently
PAUSE : CONTINUE after giving other experts a chance to catch

information just become available first

ALIAS : create an entry point variable

d) Memory probing actions

BINDC MEMORY ACTIVE/EXPECT : try to find a specific concept in active memory or expected by it

BINDC DISCOURSE FOCUS/EXPECT : ask memory whether a specific concept is in focus or expected by the discourse

BINDC REAL-WORLD PLAUSIBLE/BELIEF: probe specific kinds of real-world knowledge, e.g. whether a concept refinement is plausible in context

Figure IV. Sense Discrimination Language.

Questions

Many word-experts start with the SIGNAL question, trying to determine at what point they enter the overall comprehension process. The word-expert for *deep* contains an example of this. If the incoming signal is *entity-construction* (meaning that we are in the middle of the construction of an entity -- a semantic term for a noun/noun phrase, as in *a very deep pit*) *deep* simply participates in the current lexical sequence and goes on to entry1; if not, it first opens a sequence (as in *deep pits*). Another use of the SIGNAL question is the determination of the nominal or verbal usage of a word: if the signal is *entity-construction*, we have a nominal use, if it is *action-construction* (a verbal group is being built), we choose the subtree with the verbal usages. The LITERAL and IDIOM questions look for particular lexical elements. The *throw* expert, for instance, looks at the word to its right (with the LITERAL question) and takes appropriate actions if it happens to be one of the possible particles it can take (*away*, *up*, *in*, or *out*); supposing it was *in*, and a concept was processed after "throw in", *throw* probes the lexical sequence of this concept (with the IDIOM question) to see if it is e.g. "the towel", which leads to a concept refinement "give up". The BOUND and VIEW questions are very important, and they are closely related to the memory probing actions. A BINDC action -- checking the presence of a concept in some memory mechanism, see below -- is usually followed by the BOUND question checking whether the binding attempt succeeded or failed, with different actions taken accordingly (e.g. if the binding attempt fails -- the concept is not in memory -- a common action is to wait for that concept (see 4.2.3)). An example of the VIEW question can again be found in the *deep* expert. In entry1 it posts an expectation of possible entities it can be used with (specified in the ONEOF-slot of concept0), and when a candidate concept arrives, VIEW selects the most reasonable characterization of that concept from the multiple choice of possibilities (entry2). For instance, if the entity concept was "pit" (created by the word-expert for *pit*), a match would be effected between concept1 and the "volume" possibility of concept0. Thus, VIEW tries to determine the conceptual closeness (proximity) of two memory objects; it is a form of best-fit pattern matching, necessary because an expert can never anticipate the exact content of a concept (the number of different contexts a word occurs in is potentially infinite). This also explains why VIEW has an ANYTHING choice for cases when the concept is completely unexpected, as can be the case in metaphorical language use. (I will come back to how expectations relate to contextual unpredictability in 4.2.3.)

Actions

The first group of actions keep track of linguistic units (used by the system for specific purposes, see 4.2.3). OPENG, DECLAREG and CLOSEG take care of delimiting the lexical sequences in syntactic constituents (NPs -- as in the *deep* expert, PPs, VPs); BREAKG is an action signalling the end of a sentence (invoked by the experts for the punctuation marks).

The first subgroup under b) are actions that speak for themselves: simple and complex concepts are constructed, gradually refined and finally stored when their exact contextual meaning is found. This subgroup is closely related to the actions under a) in that concept and sequence building go hand in hand. The former dominates the latter, though: lexical sequences are not structures in their own right, but form part of the concept they belong to (see Figure V below for a complete specification of what a concept looks like in WEP). The incorporation of a lexical sequence into a concept structure happens through the LINK action often taken by the main element of a concept (in syntactic terms: the head of a phrase). In "the deep pit", for instance, *the* opens a group (OPENG) and participates in it (DECLAREG), *deep* participates in it too, *pit* also, and moreover it closes the group (CLOSEG) and LINKs it to the concept it created. I will come back to the relation between lexical sequences and concepts when I discuss the revisions of WEP (see 4.3.3.3).

Subgroup 2 under b) contains the actions that are important for the interpretation of verbs: they fill in the contextually appropriate caseframe of the verb. When in the sentence "The man loves his wife" *loves* (call it concept0) receives the concept corresponding to "the man" (concept1) it takes the following actions, incorporating "the man" as agent in its caseframe, and specifying the role of the "the man" concept as agent too:

```
(ASPECTC C0 (AGENT C1))
(ROLEC   C1 (AGENT TO C0)).
```

These actions complement each other, but note that an aspect is mostly unique in a concept whereas a concept can fulfil more than one role in different other concepts. In "The man loves his wife", for instance, "the man" is agent to *loves* and at the same time it fulfils the role of "first term" in the concept "relationship" (call it concept2) created by *his*, in which "wife" is the second term:

```
(ASPECTC C2 (TERM1 C1))
(ROLEC   C1 (TERM1 TO C2)).
```

With the actions relating to concepts described, it is interesting to look at the complete specification of concepts in WEP (these concepts form also the final output of the parsing process). They are objects with a number of slots to be filled and used in the course of processing:

CONCEPTS

SLOTS:

TYPE : the type of the concept (e.g. "entity" for a noun or noun phrase, "action" for a verb or verb phrase (although "action" does not cover all verbs really))

VALUE : a list of value specifications of a concept (see 4.2.2.2 for an example)

[ONEOF : a list of specifications of at least one of which the concept is an instance

NONEOF : a list of specifications of which the given concept is definitely not an instance]

ASPECTS : a list of pairs indicating the slots of a concept and the concepts that fill those slots

ROLES : the counterpart of the ASPECTS list, indicating the roles played by the concept in the slots (aspects) of other concepts.

LEXICAL : the surface lexical sequence corresponding to the concept (if there is one)

Figure V. The slots of a concept structure.

In 4.2.2.2 a simple example output structure built after processing a sentence is shown with the slots of all the concepts involved filled. The slots that do not appear in the output structure are the ones that did not receive a value (e.g. not all concepts have a lexical sequence associated with them, cp. the "relationship" concept built by *his* in "the man loves his wife"), and also the ONEOF and NONEOF slots. These are mainly used during processing as long as the VALUE of the concept is not yet exactly determined through contextual interaction (e.g. by invoking the VIEW question). In the final output structure the VALUE slot contains the contextually appropriate specification from the ONEOF list, and there is no need for the NONEOF list any longer either then (it has played its role as helpful element in determining the nature of the concept).

The next group of actions are the expert-internal control flow actions. NEXT is a branching action within an entry to a next node, whereas CONTINUE is a branching action to another entry. PAUSE is like CONTINUE, but branching is not executed immediately in order to give other experts (not running currently but awaiting information, see 4.2.3) time to react to a piece of information (e.g. a signal) just become available after an action of the now PAUSE-ing expert. When these other experts have taken their actions triggered by the piece of information, the PAUSE-ing expert continues at the specified entry point. The ALIAS action is not an easy one to understand, but it is a necessary evil if one works with discrimination nets (see Figure VI). If *sequence1* is a line of processing in an expert that consists of the three subsequences A, X and B (in that order) and *sequence2* is a line of processing that consists of C, X and D, subsequence X is shared by the two paths through the expert process. If the representation language does not provide an action to make sharing of subprocesses possible, these would have to be duplicated. ALIAS solves this problem in the following way: before sequence1 and sequence2 enter their shared subsequence X, they specify that after the execution of X they want to continue with B and D respectively. The continuation from X goes in different directions then, depending on whether sequence1 entered it or sequence2. Figure VI shows how ALIAS takes care of this.

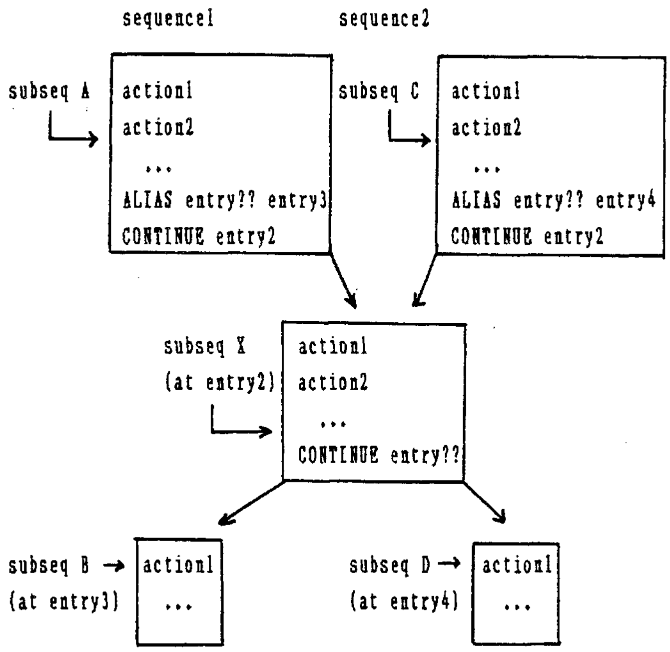


Figure VI. The use of ALIAS illustrated.

In the application to Dutch this action is not used any more, since beside its advantage (no duplication of information) it has the great(er) disadvantage of making the experts less readable: if one looks at subsequences B or D when reading an expert, there is no information present *there* that says in what sequence of processing or tree traversal they belong. An alternative approach to the problem of shared subprocesses in discrimination trees (used in the application to Dutch, but not completely satisfactory either) is the following: there are no ALIASes; at the end of subsequence X, subsequences B and D are brought together. The correct branching (now within X) is obtained by performing an action or asking a question that allows the expert to recover what it had done so far in *either* subsequence A or subsequence B. Thus, there is some duplication of information (i.e. through the attempt at retracing the execution steps taken thus far), but the

---

(3) One of the questions that was added to the representation language for the application to Dutch to avoid using ALIASes by making possible the recovery of information specified earlier is QFEATURE that asks if a concept has received

advantage is the greater perspicuity of the experts (3).

The last group in Figure IV are the memory probing actions; they allow experts to interact with several knowledge sources present in short-term or long-term memory when they are trying to find or refine concepts. The ACTIVE knowledge source contains the concepts processed so far in the sentence (during the processing of "The man eats a peach", for instance, it contains the concept "the man" when *eats* enters the process); the EXPECT knowledge source contains possible concepts waited for. In "He eats a lot", *eats* binds *he* into its agent role (like "the man" above), and creates an expectation for an object; this expectation enters the EXPECT region of memory. When *a lot* executes, it probes this region to find out if an object is waited for (if so, it signals to the verb that it is a candidate for that role). (In "He loves her a lot" there is no object expected when *a lot* executes -- *her* fulfils that role -- and *a lot* then determines in context that it is an adjunct indicating intensity; this is a good example of how the sense disambiguation process can work and it is explained in detail for the Dutch equivalent of "a lot" - *veel* - in 4.3.3.2.) Experts can also probe the discourse situation (what activity or concept is in focus? what is expected given the nature of the discourse so far?); this often happens when the VIEW question is incapable of discriminating well enough among its possible concepts. Finally, specific real-world knowledge can also be probed when pragmatic matters of plausibility or belief have to be probed for correct sense discrimination in context (4).

---

a certain specification in the course of the process. Depending on the answer, branching can continue. (QFEATURE is not dealt with further in 4.3 since its introduction had no further linguistic motivation.)

(4) A minor change in the representation language for the application to Dutch concerns the BINDC actions and the BOUND question usually following them. In Small 1980 a typical sequence of nodes with these elements would be the following:

```
(n1:action (BINDC c1 MEMORY ACTIVE c2)
           (NEXT n4))
(n2:...)
(n3:...)
(n4:question BOUND c1
         [bound n5]
         [unbound n6])
```

Since this completes our overview of the sense discrimination part of the WEP representation language, a word is in order about the implementation of all the memory mechanisms discussed. The answers to the real-world, discourse and view probes (BINDC REAL-WORLD, BINDC DISCOURSE, VIEW) are currently provided by interaction with the WEP user. This is mainly a matter of clean design: the inter-expert interactions have been the focus of attention; access mechanisms to the knowledge sources in memory are provided, but rather than hacking e.g. a complicated pattern-matcher or a simple network into the system to handle the subset of the language parsed, the existence of a fully developed central semantic network scheme is assumed (cp. the stress on horizontal lexical-contextual interaction in process linguistics, with vertical intralexical interaction not in focus). Since the focus of attention of the application to Dutch has not been the development of such a network either, I have retained the user-interaction, except in cases where simple pattern matching can do (e.g. "Can concept1 (refined as a person) be VIEWed as a person?" is handled by the system itself instead of the user). A high priority issue for future research is plugging a fully specified network into WEP.

```
(n5:...)
(n6:...)
```

Two objections to this have led to a revision of BINDC and BOUND. The first is that two elements of the representation language always occurring together are spread out over several nodes, with reduced readability as a result (i.e. node4 does not contain any information about its link with node1); the second is that the BINDCs all *query* some memory mechanism, and as such are better viewed as question nodes. An elegant solution to these objections was obtained by melting the BINDC action and the BOUND question into one question, with the following syntax:

```
(n0:question (BINDC <...>)
             [BOUND n1]
             [UNBOUND n2]).
```

(The experts in Appendix 2 do not contain this new question yet, but it has been implemented in the most recent version of WEP.)

4.2.2.2. Output concept structure

Although the focus of WEP research is on the processes of lexical-contextual interaction rather than on processed structures, the active memory of the system contains a representation of the concepts processed and the relations among them; this structure can be seen as the ultimate side-effect of the comprehension process. Figure VII shows the contents of the active memory of the system after the simple sentence "The case was thrown out by federal court" has been parsed (it is a kind of dependency structure). (See also Appendix 3 for the concept structure of a more complex Dutch sentence.)

The content of the ROLES, ASPECTS and TYPE slots was described in Figure V. A remark about the TYPE slot: in the example sentence there are no concepts of type "setting", which is the semantic term for a prepositional phrase (see Appendix 3 for examples). The LEXICAL slot contains the word sequence corresponding to the concept. The reader may wonder why the "by federal court" concept does not have "by" in its LEXICAL slot (though it has been refined correctly as the agent of the action) and why it is not of the "setting" type; this is a matter of the relative importance of concepts and their lexical sequences in Small (1980), which is discussed in detail in 4.3.3.3 (with the revisions for Dutch). The reason why "was thrown" has "was -en throw" as its sequence will become clear in 4.3.2 when I discuss the order of execution of the experts in morphologically complex words (in which affixes are considered as experts in their own right) as realized in Small (1980) and in the application to Dutch. Finally, the VALUE slot contains the (semantic) concept refinements that were made in the course of the comprehension process; these refinements are given as a list, with the head element containing the last refinement. The last element of the list shows the default value given to a concept upon its creation: if it is of type "entity" its value is ANYTHING, if it is of type "action" it is ANYACTION (5). Note that the

(5) Whereas "entity" seems to cover the semantic nature of a noun (phrase) fairly well, the reader will have noticed that it is not so easy to find a semantic term for a verb (phrase) or a prepositional phrase. "Action" covers some verbs, but not all; "setting" also covers some prepositional phrases semantically, but not all; maybe "adjunct" covers them all, but it does not say much about the content of the

refinement history (to be read from right to left in the VALUE list) shows that *case*, *court* and *throw out* were disambiguated correctly.

```
************active memory************

   CONCEPT44

ROLES  : AGENT TO CONCEPT19
LEXICAL: FEDERAL COURT
VALUE  : (FEDERAL-GOVERNMENT-AGENCY JUDICIAL-COURT COURT ANYTHING)
TYPE   : ENTITY

   CONCEPT19

ASPECTS: AGENT  CONCEPT44
         OBJECT CONCEPT9
LEXICAL: WAS EN THROW
VALUE  : (ORGANIZATION-THROW THROW-OUT-OF-COURT DISCHARGE-INTENSELY
         THROWOUT THROW ANYACTION)
TYPE   : ACTION

   CONCEPT9

ROLES  : OBJECT TO CONCEPT19
LEXICAL: THE CASE
VALUE  : (COURT-CASE CASE ANYTHING)
TYPE   : ENTITY
```

Figure VII. Concept structure in memory after parsing "The case was thrown out by federal court" (adapted from Small 1980, 253).

---

concepts any more. I only signal the difficulty here, without attaching too much importance to these terminological issues.

4.2.3. Implementation

So far I have discussed the general principles behind WEP, and the representation issues following from those principles. In this last subsection I will discuss the overall system implementation: how is it that the experts can communicate with each other throughout the disambiguation process to finally agree on overall meaning of a fragment of text?

The decentralized representation of parsing knowledge in word-experts leads to an overall model organization to support exchange of information and distributed decision-making (i.e. agreement on overall meaning). Every word-expert is implemented as a coroutine, i.e. a process that runs for a while (coordinating the entire parsing process when it does), suspends itself when it needs a piece of information (letting another coroutine take over control), runs again when this information arrives, etc. until it stops executing. Thus, the lexical interaction among the expert coroutines consists of (a) providing information and (b) waiting for needed information. Figure VIII contains the WEP actions that take care of this (called "Lexical Interaction Language" in Small 1980).

a) Providing information

SIGNAL: make a message about the process state
available for use by other experts
REPORT: make a concept structure available
for use by other experts

b) Awaiting information

AWAIT SIGNAL: counterpart of the SIGNAL action
AWAIT CONCEPT: counterpart of the REPORT action
(AWAIT WORD : wait for a specific word to arrive)

c) Lookahead

PEEKW: look at the word corresponding to the next
expert on the execution list
READW: get the next expert and let it start executing

Figure VIII. Lexical Interaction Language.

The provision of information happens through the WEP actions REPORT and SIGNAL, making a concept or signal (the two types of information sent and received by the word-experts) available for use by other experts respectively. Awaiting and receiving information requires a more complicated protocol: it implies suspension of execution while waiting for a piece of information from another expert and resumption of execution when that information becomes available. This basic aspect of the distributed control is indirectly taken care of by the WEP action AWAIT. It specifies the nature of the awaited information and the point at which to continue execution upon arrival of the awaited data. When it is executed in the course of a word-expert process, the expert suspends itself (6) and creates two important internal data structures that will take care of resumption of execution. The most important of the two is the *restart demon* (7). A restart demon checks every reported concept (if the demon was created by an AWAIT *CONCEPT* action) or every sent signal (if the demon was created by an AWAIT *SIGNAL* action) to see if it matches the concept/signal specified in the AWAIT action. Of course, this restart demon cannot try to effect a match forever. It has to be constrained by limiting its lifespan (consider also the fact that in some cases there is no certainty about arrival of what it expects). This processing constraint is taken care of by the second internal data structure created indirectly by the AWAIT action, the *timeout demon*. A look at a fully specified AWAIT action will make clear what information it uses and what it does to the overall control:

---

(6) It should be noted here that execution of the AWAIT action does not necessarily imply *complete* suspension of a word-expert since the different entry points of an expert are designed to be executed in parallel; as such, one part of the process may temporarily be suspended, but other parts can go on, even initiating other restart demons in turn (several outstanding AWAITs are possible).

(7) What this data structure looks like exactly is a technical implementation matter I will not go into (see Small 1980, 65-69); only its function is described here. As far as its uncanny name is concerned: think of a demon as someone spying continually on specific computational events and jumping out of his hideout to take over control when one of those events occurs (cp. "jack-in-the-box", or rather its Dutch equivalent "little-demon-out-of-the-box"...).

```
1      (AWAIT CONCEPT ENTITY
2                 (FILTER CONCEPT1)
3                 (BINDCONCEPT CONCEPT2)
4                 (WAIT BREAK 1)
5                 (CONTINUE ENTRY7)
6                 (ELSE ENTRY10))
```

This AWAIT is part of the process associated with the verb *eat*. The typical sequence of actions by a verb is to try and find (bind) certain concepts in memory first (concepts that could fulfil roles in its caseframe). Thus, *eat* builds a concept (call it CONCEPT1) of type "entity", refines it as "something edible" and "potential object" and looks for a matching concept in memory. If no such concept is present, it creates the expectation formalized above (referring to CONCEPT1 created earlier on in the process). Lines 1, 2 and 5 of the action contain information used by the restart demon (line 3 is not important here). The demon has to keep track of concepts (not signals) reported further on in the overall WEP process, and these concepts have to be of type "entity" (line 1). They have to match CONCEPT1 created earlier (i.e. CONCEPT1 is the filter for the match; line 2), and if a successful match occurs, entry7 is the continuation (resumption) point for the expert that created this expectation (line 5). Lines 4 and 6 contain information for the accompanying timeout demon constraining the restart one. The attempt to find a matching concept should not be continued longer than the next sentence break (i.e. when an expert sends the end-of-sentence signal) (line 4). If this break occurs before a successful match (i.e. if the restart demon *times out*), the continuation point is entry10 instead of entry7 (line 6). (By the way, since the expectation above is part of the verb *eat* waiting for an object, entry10 will refine *eat* as having an implicit object since no object arrived in time.) As far as the units of measurement for timeouts are concerned: in the current system they are based on certain model events, including

(a) the number of syntactic groups created
(b) the number of words read
(c) the number of sentence breaks encountered.

It is my intention to change these units to semantically more interesting ones in future WEP research; however, since such a revision may require serious changes to the system itself (because the timeout demons need the counters), I will not deal with it in 4.3. One possibility would be to have experts wait for the partial or complete processing of concepts of specific types (e.g. "wait for one concept of type action to be processed completely before timing out").

The reader may have noticed that what I have described here is actually an implementation of the process-linguistic concept of the expectation-feedback cycle. Modeling words as coroutines and having AWAIT actions leading to the creation of restart + timeout demons is exactly what is needed to simulate expectation and feedback. If an expectation is fulfilled, the restart demon takes care of further processing, if it is thwarted (no match) nothing happens, and if it dies out (times out) the timeout demon takes care of further processing.

Because I am relating process linguistics and WEP to each other, it is also interesting to point out here that WEP only models the normal mode of processing by human beings (cp. 3.4). The metamode (appealed to in cases of word play, garden-paths, etc.) is not simulated; WEP would have to be extended with a metamode processing component to deal with those (exceptional) cases of natural language processing.

In the context of both expectations and what mode of processing WEP simulates, an interesting extension suggests itself. Consider the *deep* expert once again (Figure III). In entry1 it posts an expectation for the concept it hopes to find to its right, anticipating this concept to be "oneof PERSON ARTISTIC-OBJECT VOLUME ANYTHING". When the concept arrives, *deep* checks (in entry2) with the VIEW question which of these possibilities applies. What I want to look at here is the option ANYTHING in the *oneof* slot of the concept. As mentioned in 4.2.2.1 it is a necessity because of the potential infinity of contextual usages of words in combination with one another. ANYTHING will always match the concept under inspection, so the process can continue. In fact, in cases where the VIEW question has to be answered with the ANYTHING option, we have an *unexpected* concept. (Note that this is not the same as what I call a thwarted expectation here, which is simply a no-match or negative-feedback matter, e.g. when the types of two compared concepts do not correspond.) As already mentioned in 3.3.4 (note (6)) -- with reference to Schank & Birnbaum 1984 -- these cases of unexpected (but acceptable) concepts could be very important in learning. Learning might then be considered as an important specific instance of the metamode of processing, triggered a.o. by unknown (unexpected) combinations of words or concepts. Hence, the extension of WEP I have in mind here is that in cases where the VIEW question can only be answered with the ANYTHING option WEP might enter the learning mode as an instance of the metamode of processing. What could be done then is an automatic extension of the expert, in this case simply adding the specification of the unexpected concept to the ONEOF list and extending the multiple choice of the VIEW question. Note that relating the EFC to the learning process enhances its importance beyond language understanding; when an unexpected concept arrives, metamode is entered and acquisition is triggered (cp. Small 1980, 211-213).

At this point in the discussion I should point out a difference of view on expectations between Small and myself. Throughout chapter 3 and also in the discussion of the WEP implementation I have stressed the importance of the EFC, and hence of expectations. In Small (1980), though, expectations are not considered so important. It is acknowledged that they help understanding (1980, 4), but the stress in WEP is on what the words themselves bring about. Expectations are not binding, i.e. they are decoupled from the WEP control structure (1980, 20). What it means that expectations *are* coupled to a control structure becomes clear from Small's comparison of WEP to Riesbeck's parser (Riesbeck 1974). Both parsers have a lot in common (see the discussion in Small), but in Riesbeck's parser it is the expectations that drive the overall parsing process. If an expectation (which is a hypothesis about the conceptual content of text to come) is thwarted, the parser backs up and tries another expectation; expectation and control (backup) are indissolubly linked. It is certainly true that in WEP there is not such a direct link between expectations and control. Yet, the indirect link is strong enough to say that expectations are of crucial importance to WEP as well. As discussed above, it is the AWAIT action that leads to the creation of restart and timeout demons, the mechanisms at the heart of the WEP coroutine regime. Without AWAITS (expectations) there would be no restart/timeout demons, and hence no distributed control structure. Moreover, the restart demons always use information about what is expected when they try to effect a match of concepts or signals. Still, it is true that there is no attempt to match expected structures within a strict backtracking control regime, and that a timeout demon discards the content of the expectation (all that counts for it is the timeout condition). I hasten to add here that I believe the WEP implementation of expectations as crucially important but not control-dominating is a feature of the system that correctly models human understanding. During comprehension, we expect a lot on several levels of processing (ranging from simple syntactico-semantic expectations of a noun (phrase) / entity concept triggered by a determiner to broad expectations about the content of the discourse), but it is always the actually occurring words that "prune the expectation forest" and lead to convergence on one specific contextually appropriate meaning. In short, understanding remains in essence a bottom-up process driven by the words themselves rather than a process driven by hypotheses (viz. top-down expectations).

Finally, I have to go back to Figure VIII for the last actions of the lexical interaction language to be looked at. They are the lookahead actions PEEKW and READW. Sometimes an expert needs to know the identity of an adjacent word in order to make the right choice in its sense discrimination process. A good example from English is the verb-particle sequence. *Throw*, for instance, PEEKs at the word to its right, checks -- with the LITERAL question -- whether it is one of the possible particles it can pair up with (*up*, *away*, *in*, *out*), and READs this word if it is one of them; PEEKW does not lead to execution of the word peeked at, whereas READW does imply execution of the word read. Of course, if the word is not one of the particles, *throw* leaves it alone and continues its sense discrimination at a different entry point. Figure IX shows how this sequence of PEEKW, LITERAL and READW (already mentioned in 4.2.2.1 as well) happens in the *throw* expert. (Remember that a simple "*" is used as a catchall matching anything.)

```
(entryX (node0:action (PEEKW nextexpert)
                      (NEXT node1))
        (node1:question LITERAL nextexpert
                      (away node2)
                      (up   node3)
                      (in   node4)
                      (out  node5)
                      (*    node6))
        (node2:action (READW nextexpert)
                      (SIGNAL *action-construction*
                            (to nextexpert)))
        (node3:...)

          ...
```

Figure IX. PEEKW and READW in the *throw* expert
(adapted from Small 1980, 219-224).

I mention PEEKW and READW for completeness' sake here, but I have made little use of them in the Dutch experts because they go against the principle of lexical-contextual interaction. Although it may speed up processing, the experts should not peek at each other but talk to each other if there is a processing difficulty.

To conclude subsection 4.2.3, Figure X playfully summarizes the overall WEP control flow resulting from the coroutine environment. It shows that this flow of control can be viewed as the movement of a window across the input stream. Inside the window, control passes from word-expert to word-expert and back again, but the reading of new words and the termination of old experts causes the overall processing window to move ahead, expanding its right side and contracting its left side respectively. Eventually, the window includes the last word of the input text, and the process terminates.

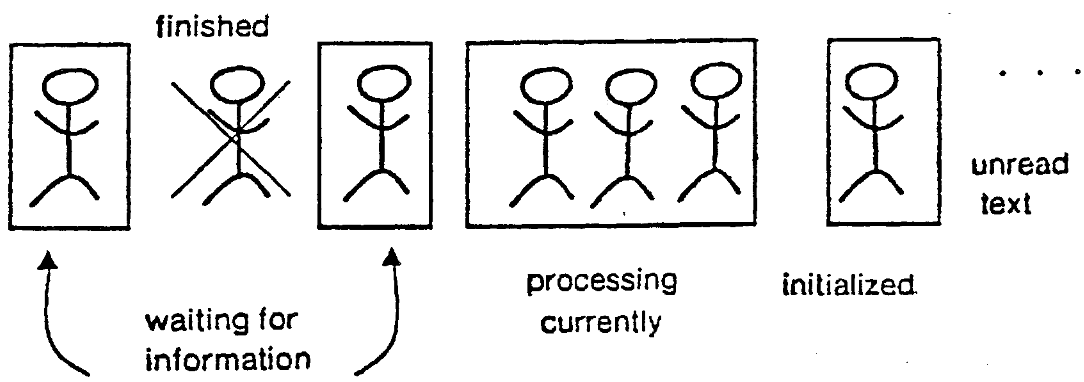


Figure X. Word Expert Parser control flow
(Small & Lucas 1983, 12).

4.2.4. WEP scope for English

In Small 1980 the following word-experts were implemented:

- the, a, such
- in, by, out
- -s, -en (-ed), -ing
- was, has, growl, eat, throw
- man, philosopher, river, peach, pit,
  towel, court, case, tiger, party, spaghetti
- #period#

Though this may look like a small vocabulary, it should not be forgotten that sense discrimination was the focus of attention; a word like *throw*, for instance, is a 6-page process disambiguating the many senses of the verb (throw up, in, out, away; throw a party; throw something in something, etc.). Some example sentences the Word Expert Parser can analyze correctly (i.e. lexically disambiguate and assign a semantic concept structure) are:

The man eating peaches throws out a pit.
The man eating tiger growls.
The man eating spaghetti growls.
The deep philosopher throws a peach pit in the deep pit.
The man has thrown in the towel.
The case was thrown out by federal court.

For a description and discussion of all the interesting phenomena and difficulties these sentences contain I refer to Small 1980. In the next section I will consider some aspects of the way the analysis is done in greater detail, with their extensions and revisions for the application to Dutch.

4.3. WEP applied to Dutch

4.3.1. Introduction

For the application of WEP to Dutch the overall system implementation (the coroutine regime) did not require any changes at all; on the contrary, as will be discussed below, it proved very handy for parsing the numerous discontinuities of linguistic elements in Dutch. The changes that have been made are of the following kinds:

1) changes in the representation language: new questions and actions have been added (cp. QFEATURE mentioned above), whereas others have been removed (cp. ALIAS above).

2) changes to the processes associated with verbs to enhance the scope of syntactic structures dealt with.

3) changes in the way specific linguistic phenomena are handled.

To make it easier to follow the discussion, I give the experts implemented for Dutch beforehand. It should be noted that not all possible meanings of all words have been considered; the English translations indicate which ones have. In Appendix 2 the complete process associated with each of the experts (29 in all) is given, and Appendix 3 contains an example of a WEP run for a full sentence.

de	- the (article)
	- (Latin preposition in *de facto, de iure*)
een	- a (article)
	- one (numeral)
van	- of, from, out of,... (preposition)
door	- through, by (agent to passive) (preposition and particle)
in	- in (preposition and particle)
op	- up (preposition and particle)
eet- (verb stem)	- eat (with *op*: eat up)
bel- (verb stem)	- ring
	- call (up) (also with *op*)
houd- (verb stem)	- hold, keep
	- love, like (with preposition *van*)
word- (verb stem)	- turn, become, get
	- auxiliary of the passive
man	- man
	- husband
vrouw	- woman
	- wife

other nouns implemented: hand (hand), zomer (summer), maandag (Monday), appel (apple/roll-call), Geert, Hilde

rood	- red (only predicative use implemented)
veel	- a lot of, many
	- a lot, much
	(both usages in "Veel mannen houden veel van vrouwen" "A lot of men love women a lot")
zijn	- his (it is also infinitive "to be" and verb form "are", but these have not been implemented yet)

haar	- her (possessive and personal pronoun; only the former implemented) - hair (noun)
-s	- plural noun morpheme
-en	- plural noun morpheme - plural verb morpheme
-t	- singular verb morpheme (2nd and 3rd person singular present tense; only 3rd person used)
ge-	- past participle morpheme (-en/-ed) (see 4.3.2 for a more precise description of the past participle in Dutch)

punctuation marks: #punt# (period), #uitroepteken# (exclamation m.), #vraagteken# (question m.)
(Their role is limited to augmenting the sentence counter and to refining sentence structure as "elliptic" if they find no verb among the concepts processed and present in memory)

4.3.2. Morphology

In WEP (as in most natural language analysis systems) a morphological analysis subprogram is invoked at an early stage of analysis (i.e. as soon as a word is read from the input); it consists of affix-stripping rules, and the algorithm for their application roughly runs like this:

```
(1) look for the word in the lexicon
    if present: task completed
    if absent (2) try to find base and affixes using
                  affix-stripping rules (implying the
                  presence of base forms and possibly
                  affixes in the lexicon)
                  if success: task completed
                  if failure: signal trouble
```

In both the application to English and to Dutch, this algorithm looks for the presence of a word in the list of monomorphemic experts (*eat*, *up*; *man*, *in*) or in the list of irregular experts; in the latter case, the component elements (listed with the word) are returned as a result of the analysis (e.g. *thrown* is listed as *throw + -en).* Otherwise, rules are applied to the word to find its component parts. The exact nature of the relationship between stems and affixes is determined by lexical interaction among them. In the application to English, a side-effect of the morphological analysis is that the order of stem and suffix is reversed and execution happens in that order. *Thrown*, for instance, is processed as a sequence of *-en* and *throw* (remember that affixes are modeled as word-expert processes too). The reason why this is done is that it makes analysis easier: when a suffix runs, it knows that the word to its right is its stem, which makes interaction less complicated. *-En* signals that it starts the construction of a verb group, which allows *throw* (receiving this signal) to choose immediately between its verb and noun usages. If *throw* ran first, it would not always be able to disambiguate itself without looking at the next expert. PEEKW might help, but in many cases it would be an unnecessary action; moreover, if the next expert was allowed to run unconditionally to "help" *throw* disambiguate itself, and if it had nothing to do with *throw*, interaction could be fouled up because *throw* might get unexpected and useless feedback ("noise"). (This, by the way, shows the complexities of intricate processes whose course is not predictable.) In spite of these potential problems with disturbed interaction, the order of execution of stem and affixes is their order of occurrence in the application to Dutch, and not the reversed order. There were two reasons for changing this. The first is that WEP also wants to be a model of human language understanding; it seems implausible that the order of stem and affixes is reversed in this process: when we hear/read a word, we get the stem first (if there are no prefixes, of course) and we start building concepts from this stem before we get the suffix. The second reason is that problems arise when a suffix is ambiguous (e.g. *-s* in English (verb singular and noun plural) or *-en* in Dutch (verb and noun plural)); more complex interaction between suffix and stem is needed whatever the order they are executed in. Problems also arise when prefixes are considered as well. For Dutch especially, past participles of compound verbs, for instance, contain their particle and the past participle marking before the stem (cp. English *eaten up* with Dutch *opgegeten*); the latter marking can even be considered as a discontinuous affix (English *worked* (-ed), Dutch *gewerkt* (ge-...-t). Thus, to avoid complications the order of stem and affixes is retained. *Opgegeten*, for instance, is listed as *op* + *ge* + *eet*- and its component parts start executing in that order. (Note that the discontinuous affix is reduced to *ge*- in the execution; the same applies to *gebeld*, which is executed as *ge*- + *bel*-. *Ge*- contains enough information to interpret the sequence correctly as a past participle.) As mentioned above, though, and as experienced in early attempts at parsing Dutch words this way, interaction can be fouled up or become unnecessarily complicated since a stem does not "know" whether what follows it is its affix or not (remember that each expert coordinates the parsing process in turn). The solution to these problems was the introduction of a new question in the representation language (QPARTOFWORD). An expert can ask this question to the morphological analysis processor whenever it needs to know whether it is part of a morphologically complex word or not. If it is, it can start interacting with the next expert with the certainty that it belongs together with it; if it is not, it leaves the next expert alone. This allows for a more flexible and decentralized use of the morphological information.

The difficulties with interaction of stems and affixes, taken together with the many idiosyncrasies of derivational morphology (see e.g. Taylor 1980) and the absence of proof in the psycholinguistic literature that words are morphologically analyzed on-line (see chapter 5) suggest that it may be better to fully list complex words and let them have their own process. Fully listing words even becomes a necessity with multiply segmentable words such as *kwartslagen* that means "quarter beats" if segmented as *kwart+slagen* and "quartz layers" if segmented as *kwarts+lagen*. Correct segmentation is only possible by considering the context; yet, in WEP the words are the only active processes, and no interaction between the morphological analysis subprocess on the one hand and the context on the other ("over the heads of the experts") is possible. Thus, the segmentation process cannot decide which experts will have to be initialized. The only possibility left is to have *kwartslagen* as a word in its own right in the dictionary; the process associated with it can then easily disambiguate the word through interaction with the context. The main disadvantage of fully listing words is the enormous duplication of (processual) information in the lexicon; trying to avoid this implies the necessity of closely examining the internal structure of the lexicon (grouping together words that are morphologically related and trying to have them share information).

Since WEP does not (yet) have a fully developed semantic network relating words phonologically, morphologically and semantically, little can be done about the organization of such a network now. However, the problems mentioned above suggest a non-uniform treatment of morphology, compatible with the stress on idiosyncrasy over generality in natural language advocated both in process linguistics and in the WEP research. The segmentation-rule component can be removed, and all morphemes (words and affixes) are "listed", be it in different ways. Stems and affixes trigger their processes as they do now; in cases where lexical interaction can handle the relationship between stem and affixes of complex words (disambiguating multiple possibilities if necessary), these words are listed with their component parts (i.e. morphology is present in the lexicon); these parts are executed in the order they occur in. However, in cases where segmentation is problematic, or where lexical interaction between stem and affixes may look like "processual overkill", the words are listed without internal morphemic marking, and they have their own process associated with them. Which words are to be listed without morphemic marking and which with marking is a matter of further application of WEP to a larger subset of Dutch (and also of results of research in psycholinguistics dealing with on-line morphological analysis).

4.3.3. Syntax and semantics

4.3.3.1. Introduction

In 4.3.3.2 I will start by describing an interesting case of how disambiguation of a highly ambiguous Dutch word (*veel*, "a lot") is modeled in the word-expert *veel*. After all, sense discrimination is the phenomenon that started the whole WEP research going. In 4.3.3.3 I will look at "low level" syntax and semantics, viz. the relationship between lexical sequences (constituents) and the concepts they correspond to. The important point made in that subsection will be that concepts dominate lexical sequences in a way that the latter "fall out" of the way the former are processed. Next, as can be seen from the examples given in 4.2.4, WEP does not handle a wide variety of syntactic structures for English. All the sentences are of the NP(Subject)-VP-(NP) (PP) variety; no questions, imperatives or declaratives starting with a PP are analyzed. The most interesting syntactic phenomenon that WEP can handle nicely are passive sentences, in which the lexical encoding of processual information in the *-ed* morpheme and the interaction of this morpheme with the rest of the verb group leads to correct interpretation of the sentence (see Small 1980 for a more detailed description). Thus, the challenge was to try to enhance the scope of syntactic structures parsed without giving up the view of language analysis as a lexically-based decentralized process. Considering the more varied word/constituents order in Dutch (compared to English) the challenge became even more interesting: can the words themselves take care of the correct analysis of declaratives (with or without inversion, see 4.3.3.4), questions and imperatives without the imposition of "extraneous" rules? The answer to this challenge was the development of the important notion of "dynamic caseframe", a notion that will be discussed in 4.3.3.4. Finally, in 4.3.3.5 a very typical feature of Dutch (the occurrence of discontinuous constituents in the "pincers construction") will be dealt with. It will be shown that specific cases of this phenomenon can easily be dealt with in a highly interactive, context-bound, expectation-based system like WEP.

4.3.3.2. Disambiguating *veel*

In 4.2.2.1 it was shown how the English word-expert *deep* was built from an informal sense discrimination net. I will now describe how the expert for the polysemous Dutch word *veel* was built from such a net in order to enable it to disambiguate itself in context.

A look into the Dutch dictionary *Van Dale* suggests the informal net in Figure XI, ordered in the first place according to the different word classes *veel* can belong to, and further according to its syntactic and semantic usages.

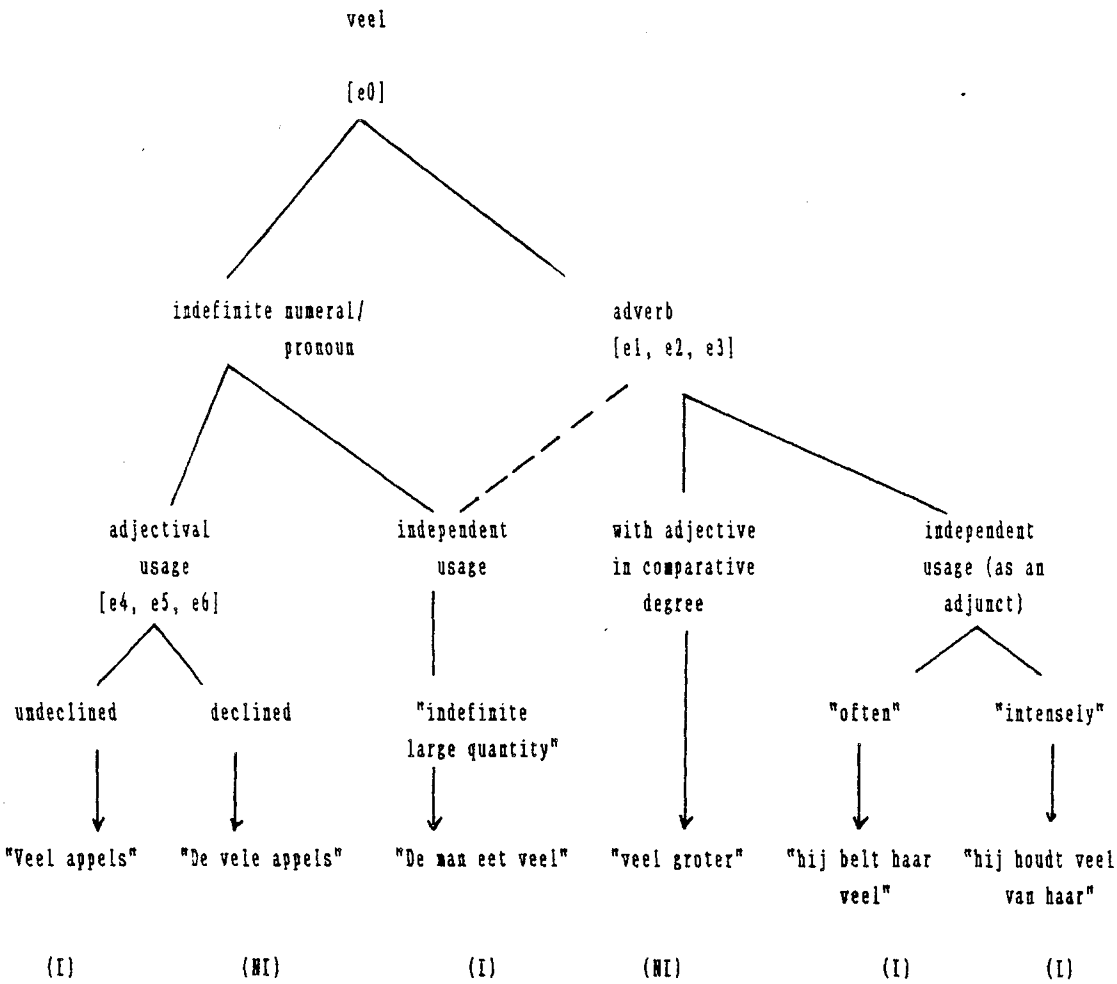


Figure XI. The sense discrimination net of *veel* ("a lot").

The "I" and "NI" stand for "implemented" and "not implemented" respectively. Four of the possible usages have been implemented; the other two would have required that the bound morphemes *-e* (adjectival declension) and *-er* (comparative degree) had been implemented too (with morphological interaction helping in the disambiguation of these usages of *veel*), which is not the case. As can be seen, I have also added the entries (e0, e1, etc.) of the expert (given below in Figure XII) that belong to the important subparts of the net. Entry0 will take care of a first choice, actually between the adjectival usage of *veel* and its independent usages. This means that the independent usage of the numeral/pronoun is moved to the right subtree (see the dotted line; cp. *Van Dale*[xxx], p. 3106 *sub veel* II,2 where it is also suggested that the independent numeral usage and the adverbial usages come close to one another). This slight reorganization of the tree is a consequence of the fact that it is easier to find out in context whether a word is used independently or adjectivally than to find out whether it is a numeral or an adverb. (Moreover, as already said, this distinction is blurred for the independent usages of *veel*.) Entries 4, 5 and 6 take care of the adjectival usage then, whereas entries 1, 2 and 3 deal with the different independent usages. Let us have a look now at how this happens exactly in the expert (the questions and actions involved).

```
[word-expert veel

[e0  (n0:a (await signal
                   (filter setting break)
                   (wait word 1)
                   (continue e1)
                   (else e4)))
]
[e1  (n0:a (buildc c1 entity)
                   (role object))
           (bindc c2 memory active c1)
           (next n1))
     (n1:q bound c2
              [bound n2]
              [unbound n3])
     (n2:a (createc c3 setting)
           (addlex c3 x0)
           (refinec c3 =c#dikwijls)
           (rolec c3 bijw-bep)
           (report c3))
     (n3:a (buildc c4 setting
                  (role vzvw))
           (bindc c5 memory expect c4)
           (next n4))
     (n4:q bound c5
              [bound n5]
              [unbound n6])
     (n5:a (createc c3 setting)
           (addlex c3 x0)
           (continue e3))
]
[e2  (n0:q view c3
```

```
                [=c#dikwijls n1]
                [=c#intens  n2])
      (n1:a (refinec c3 =c#dikwijls)
            (rolec c3 bijw-bep)
            (report c3))
      (n2:a (refinec c3 =c#intens)
            (rolec c3 bijw-bep)
            (report c3))
]
[e3  (n0:a (bindc c2 memory expect c1)
            (next n1))
      (n1:q bound c2
                [bound n2]
                [unbound n3]])
      (n2:a (addlex c2 x0)
            (refinec c2 =c#grote-hoev)
            (report c2))
      (n3:a (createc c3 setting)
            (addlex c3 x0)
            (continue e2))
]
[e4  (n0:a (openg entity-construction)
            (declareg)
            (await concept entity
                      (bindconcept c1)
                      (report here)
                      (wait group 1)
                      (continue e5)
                      (else e6)))
]
[e5  (n0:a (refinec c1 =c#1#groot-aantal)
            (link c1)
            (closeg complete-entity)
            (report c1))
]
[e6   ; metanode
]
]
```

Figure XII. Word-expert *veel* implementing part of Figure XI.

To find out whether a word is used adjectivally or independently, the easiest thing to do is to wait for information from the word following it. If this word signals the beginning of a different constituent/sentence or sends other information that clearly shows its non-involvement with *veel*, we can assume *veel* is independent. Alternatively, we might wait for a signal saying that the following word is part of an entity (noun phrase), but this is not so easy as it may seem. If *veel* is part of a noun phrase, the next word can be from a variety of word classes (adverb, adjective, noun) that by themselves cannot immediately signal that they are participating in a noun phrase (there are more possibilities). Hence, a clear bordering signal is the best *veel* can hope for, in the other cases it assumes it is part of an entity/noun phrase. In entry0 (e0) in Figure XII the *setting* (another adjunct starts) or *break* (end-of-sentence) signals will cause *veel* to branch to entry1 (e1), where the independent-usage subtree starts. In all other cases, it will branch to entry4 (e4), the entry point for the adjectival-usage subtree. For the subset of Dutch WEP handles, the *setting* and *break* signals suffice as filter to lead to correct analysis of *veel*; when the scope of WEP is enhanced it is possible that more signals will have to be considered. What counts here is the general way a WEP action leads to branching within a discrimination net. The easiest part of the expert is the way it deals with the adjectival usage of *veel* (entries 4 and 5). Just like the other "entity starters" (articles, possibly adverbs or adjectives, etc.) it opens a lexical group and waits for a concept to be reported to it. When (if) it arrives, it refines the concept as "large-amount of" and takes a number of bookkeeping actions as described in 4.2.2.1. Note that a timeout demon accompanies the restart demon in the AWAIT action of entry4, making the expert continue at entry6 if no concept arrives in time. What entry6 might do then is left open because if the expert continues execution at that point, it means that something went wrong with the sentence and normal processing cannot deal with it any further. Entry6 might be specified as a "metamode entry" in future extensions of WEP. A simple possibility of what the entry might look like is that it could make the expert process continue at entry1 after all (i.e. there might be a form of backtracking in the metamode of processing). In entry1 then *veel* starts dealing with its usages as an object indicating an indefinite large quantity of something, an adjunct of time (meaning "often"), or an adjunct of manner indicating intensity (see the tree in Figure XI). To discriminate among these usages, the *veel* process runs in a way that nicely shows the strength of the memory binding mechanisms of WEP.

How it is done can be made clear by looking at the order of constituents in complete sentences. Consider:

1) De man belt zijn vrouw veel op.
2) Geert houdt veel van Hilde.
3) Hilde denkt veel aan Geert.
4) Geert eet veel.
[5) Geert eet veel appels.]

In sentence 1) we see that *veel* comes after the direct object of the sentence (also when the sentence is in interrogative or inverted form), and that it means "often". There seem to be semantic restrictions on the co-occurrence of *veel* with a direct object: when occurring with such an object it can only mean "often". Extensive research into the occurrence of *veel* in specific sentences should make clear whether this is indeed a correct assumption (see "Conclusions and further research" for a suggestion of how this context research could be done by using computational tools). Anyway, these observations explain the processing sequence eln0, eln1, eln2. Eln0 checks whether memory already contains an object; if so (eln1: c2 is bound), a setting/adjunct concept is created, its lexical and role slots are filled, it is refined as meaning "often" and reported to memory (eln2). If no object is present (eln: c2 remained unbound), we go to eln3. As can be seen, sentences 2) and 3) contain a prepositional object, which is *preceded* by *veel*. It plays the role of adjunct of intensity in 2) and adjunct of time in 3); when it is combined with a prepositional object, the meaning of *veel* is not so easy to determine. Translated in processual terms: since *veel* occurs before the prepositional object (but after the verb), it checks whether such an object is expected (i.e. by the verb) (eln3). If so, we go to eln5 (via eln4). A setting/adjunct concept is created and e2 is entered. The VIEW question is used to determine whether *veel* means "often" or "intensely" in combination with the verb and its prepositional object. Note that if we wanted to be able to make an immediate decision leading to the correct refinements in sentences 2) and 3), it would be necessary to incorporate information into the verb process about the nature of the adjuncts it can be combined with. *Veel* in itself cannot decide from the signals it gets or from the content of memory what it means in cases like 2) versus 3). It is in fact the idiosyncratic nature of the verb that determines what *veel* can mean.

Maybe future research will show that the verb not only has to "catch" (see 4.3.3.4) its agent, patient, etc. but also its adjuncts of time, place, manner, etc. As the experts are written now, adjuncts (e.g. prepositional phrases) try to find out through interactions of the participating words what they mean without consulting the verb, or without the verb controlling their interactions. Here again, research into the distribution (and not just the syntactic distribution) of constituents and their combinability is badly needed.

In order to refine *veel* correctly as an object indicating an unspecified quantity (sentence 4)), the *veel* expert goes to e3 (that is where it got to when no prepositional object was expected); there, it checks if a direct object is expected. If so, it refines the concept it created as this object (e3n2). Here again -- as with the adjectival usage, if no object is expected, either something went wrong (i.e. the sentence is incomprehensible) or we might try to reenter the expert at an earlier point (in the metamode of processing). E3n3 shows the specification of such a metamode entry sending the expert back to e2.

Finally, I added sentence 5) to show that waiting for information is necessary and leads to correct sense refinement. Looking at language as a phenomenon evolving *in time* (the process view) and not in space (the structure view), sentences 4) and 5) are the same at say t_x during processing:

		t_x	t_y
4) Geert eet	veel		.
5) Geert eet	veel		appels.

The *veel* expert does not decide anything, however, until at t_y something happens that can lead to feedback of information to the expert to enable it to interpret *veel* correctly. In this case, in 4) the *break* signal (sent by the period) leads to branching to e1, whereas in 5) the *entity-construction* signal (sent by *appel*) leads to correct branching to e4 (the adjectival usage of *veel*).

Though I am sure that further research into the contexts *veel* can occur in will show that the process is at least incomplete, I hope the example has made clear a number of things. First, detailed linguistic description of the semantic co-occurrence of words is badly needed. As far as I know, it does not exist -- not even for English. Second, NLU by computers (i.e. translating some of the descriptive results from linguistics in processual terms) is a complex problem, whatever approach one takes. If this second remark makes the need for detailed description of linguistic elements and their distributional behavior clear to researchers, a lot will have been reached already. Referring to chapter 2: generative linguistics has led to beautiful abstractions but they are sterile and worthless when it comes down to the urgent problem of concrete analysis of a concrete language. It is high time for descriptive linguistics to step out of the shade of generative linguistics and start again where it left the scene 30 years ago (cp. Gross' critique of generative grammar: "Linguistics has vanished" (1979, 879)).

4.3.3.3. Lexical groups and concepts

As mentioned in 4.2.2.1 the processing of (semantic) concepts and of (syntactic) constituents (lexical sequences) goes hand in hand, with the former dominating the latter. CREATEC/BUILDC, REFINEC and REPORT/STOREC are the concept building actions; OPENG, DECLAREG and CLOSEG build lexical groups; LINK and ADDLEX (see 4.3.3.5) take care of the incorporation of a lexical sequence into a concept.

In Small 1980 little importance is attached to the lexical sequences "corresponding to" the concepts. The concepts receive the correct refinements by the words contributing to them without these words necessarily showing up in the LEXICAL slot (see 4.2.2.2). On the whole, the lexical groups fulfil two functions in the system. The first and most important function is their use as units of measurement for the timeout demons at the heart of the coroutine regime (see 4.2.3); as such, they are indispensable. Their second function is a matter of clarity in the VIEW question (or other question dealing with the comparison of concepts): when the user is asked if a concept just processed (call it concept1) can be viewed as one of a number of possibilities, it is easier to answer the question if the lexical realization of concept1 is given along with its semantic refinements so far. An example of the VIEW question is given below. It is asked by the *court* expert during processing of "The case was thrown out by federal court". More precisely, it is asked by the *court* expert when it first runs after the following sequence of execution of the experts in the sentence: the - case - the - was - *en* - throw - out - throw - out - *en* - by - federal. *Court* has just created a concept (concept1) and tries to refine it; *federal* is waiting for this concept.

```
q> concept1    VALUE    (COURT ANYTHING)
q>             LEXICAL (federal court)
q>
q> possible views: KINGDOM-ASPECT
q>                 SPORTS-ARENA
q>                 GOVERNMENT-ORGANIZATION
q>
q> which views apply (best first)?
```

In this case it would be impossible to answer the question without the lexical sequence: the VALUE slot does not contain enough refinements yet to answer it correctly. This also shows that lexical sequence building and concept building are asynchronous activities: the former happens in a strict left-to-right order (i.e. the words are put in a lexical sequence as they enter the comprehension process), whereas the latter is a matter of interactions among experts (interactions whose nature is not always predictable, i.e. the order of execution of the experts is not fixed). To summarize: lexical sequences are not indispensable (though useful) for correct concept refinement, but they seem indispensable for control reasons (the timeout demons).

During the development of the Dutch experts, I was in two minds about the sequences. On the one hand, the OPENG, DECLAREG and CLOSEG actions proved a nuisance when writing experts: it is not always clear where to put them in the expert process, the more so as they have no function in the main WEP process of sense discrimination. Moreover, leaving out the DECLAREG and CLOSEG actions altogether and retaining only the OPENG action to make sure the word-group counter is set correctly (for the timeout demons) led to no problems for the analysis process (except in the questions exemplified above). On the other hand, however, the cinderella treatment of lexical sequences is linguistically unsatisfactory. One would expect a prepositional phrase to have its preposition included in the LEXICAL slot of the concept it contributes to (such as *by* in "by federal court", see 4.2.2.2) or at least that a preposition starts a new lexical sequence (which it does not in the English experts) since its function as constituent boundary cannot be ignored. Thus, two opposite revisions of the system suggested themselves: one, try to do away with lexical sequence building altogether and two, try to introduce *more* lexical sequencing than in the original system. Both revisions have been designed, and the second one has been implemented as well; I will discuss them both, starting with the second.

In the original WEP system there could only be one "active" lexical sequence at any point in the process, which made it impossible to have sequences within other sequences. For a prepositional phrase like "in the morning" this means that if *in* is made to start a prepositional phrase sequence, and then *the* starts a noun phrase sequence, the pp sequence is lost (overwritten). The following changes made it possible to have several active sequences:

1) the introduction of a special data structure, a "chartlike stack".

2) the convention that a word that starts a lexical sequence (with OPENG) also closes the sequence it started (with CLOSEG).

Whereas a "pure" stack only allows access of its top element (cp. the gadget in a car where one keeps coins for the parking meter), a chartlike stack allows simultaneous access of all its elements. For the stack of lexical sequences in the making, this means that a new input word is added to *all* active lexical sequences simultaneously when it triggers the DECLAREG action (the example below will make this clear). The second change was needed to make sure that all sequences pushed on the stack would eventually be removed from it ("popped"): if a word *ending* more than one active sequence were to have a CLOSEG action in it, it would be unclear which sequence or sequences to remove from the stack. Note also that the second change makes lexical sequencing more dependent on concept building. As I said above, lexical sequencing normally happens in a strict left-to-right order, whereas concept building is a matter of moving to and fro between experts (interaction); with the convention in 2) the strict left-to-right order for lexical sequencing is given up since the *first* word of the sequence closes it and not the last. That the first word can do this is a consequence of the feedback of a concept (e.g. created by a noun) to the words waiting for it (e.g. an article and/or a preposition) during semantic processing. An additional advantage of the convention in 2) is that it is easier to put the OPENG and CLOSEG actions in the right place: if a word has an OPENG action, it must also have a CLOSEG action. Let us look at a concrete example now to clarify how lexical sequencing works. Figure XIII shows what happens to the chartlike stack in the course of execution of the word-experts in the prepositional phrase "van de erg lekkere appels" ("of the very tasty apples") (8). For correct understanding of the execution trace I repeat that words enter the comprehension process in a left-to-right order, but the processes associated with them can be active at several points in the overall understanding process, depending on whether they suspend themselves temporarily to wait for information. In Figure XIII the stress is on the OPENG, DECLAREG and CLOSEG actions (with their effects on the chartlike stack -- represented as a list of lexical sequences), and not so much on the AWAIT actions (discussed further below).

executing word	effect on the chartlike stack
van : enters the process OPENG/DECLAREG suspends itself to wait for an "entity" concept	((van)) (one sequence started)
de : enters the process OPENG/DECLAREG suspends itself just like *van*	((de) (van de)) (new sequence started at top of stack; *de* added to all active sequences)
erg : enters the process OPENG/DECLAREG suspends itself to wait for a concept of type modifier	((erg) (de erg) (van de erg)) (see *de* above)

(8) As Figure XIII shows, the morphology of the adjective has not yet been implemented; hence, *lekkere* is one expert, though it could be two experts if *-e* were implemented (the same applies to *rode* in Figure XIV). This does not change anything to the discussion here, though (cp. the -s morpheme in *appels*).

----

lekkere: enters the process | ((erg lekkere) (de erg lekkere) (van de erg lekkere)) (*lekkere* added to all sequences)

DECLAREG

create a concept
of type modifier
(feedback to *erg*)

wait in turn for
an entity concept

----

erg : receives the concept created by *lekkere* | ((de erg lekkere) (van de erg lekkere)) (top sequence removed)

CLOSEG

----

appel : enters the process | ((de erg lekkere appel) (van de erg lekkere appel)) (*appel* added to all sequences)

DECLAREG

create a concept
of type entity
(triple feedback,
to *lekkere*,
de and *van*)

-s : enters the process | ((de erg lekkere appel -s) (van de erg lekkere appel -s)) (*-s* added)

DECLAREG

----

lekkere: receives the "appel -s" concept | (nothing changes)

----

de : also receives it | ((van de erg lekkere appel -s)) (top sequence removed)

CLOSEG

----

van : also receives it | () (empty, top (last) sequence removed)

CLOSEG

Figure XIII. Lexical sequence building.

A question that arises from this approach to lexical sequences is how they relate to their possible concept counterparts: do all the sequences correspond to concepts? if so, do all these concepts have to be included in the output structure ("erg lekkere", "de erg lekkere appels", "van de erg lekkere appels")? I will not answer this question in a definitive way here, but only point out that the creation of lexical sequences is non-committal as far as concept building is concerned: if no LINK action is taken by some expert, a sequence will be removed from the stack without incorporation into a concept. Concept building still dominates lexical sequencing, but the revision suggested here offers a more powerful mechanism for the user of the WEP system. It is interesting to note that the problem of intermediate concepts (and how many there are) is similar to the problem of intermediate syntactic categories (and their number) in X-bar theory in generative linguistics (cp. Radford 1981, 91-108). X-bar theory was introduced as a competitor of phrase structure theory, a.o. because the latter is too restricted in the number of category types it permits (it has only lexical categories, e.g. noun or verb, and phrasal categories, e.g. noun phrase and verb phrase). X-bar theory posits the existence of intermediate categories between these two. In phrase structure theory, "this very tall woman" is a noun phrase corresponding to the lexical noun category, whereas in X-bar "very tall woman" would at least be one intermediate category (called a "bar-projection" of the noun) between both. Once intermediate categories are accepted, however, the question arises how many bar-projections there are; see Radford 1981, 100-109 for a discussion. Beside this parallel problem, it is also interesting to note the importance of the lexical categories in both the X-bar approach to constituent structure and the WEP approach to concept building: in X-bar all categories are projections of a lexical category (the "head" element); in WEP it is these head elements that are responsible for the creation of concepts. In the above example *van* creates a setting concept (parallel to a prepositional phrase with a preposition as its head), *lekkere* creates a modifier concept (parallel to an adjectival phrase with an adjective as head), and *appel* creates an entity concept (parallel to a noun phrase with a noun as head). As we will see in the discussion of the second revision for lexical sequencing, the importance of a word as the head of a phrase (in structural terms) has its processual parallel in the resolution of one or more AWAITs upon creation/reporting of a concept by the expert corresponding to that word (i.e. heads cause busy lexical interaction).

The other possible revision of lexical sequencing is an attempt at doing away with it altogether. As I mentioned earlier, the fact that semantic understanding does not seem to need the sequences suggests that they are at least less important and possibly even derivable from the semantic comprehension process. Note that in the first revision discussed, the sequences were already made more dependent on the interaction process, with CLOSEG being done by the word-expert awaiting a concept upon arrival of this concept. And indeed, a closer look at the AWAIT actions taken by the word-experts shows that lexical sequences (constituents) can be retrieved as a side-effect of the treatment of these AWAITs by the WEP process. Figure XIV shows this for the lexical sequence "de erg lekkere rode appel" ("the very tasty red apple"). In this second revision I assume a minimum of lexical sequences, with the example being only one sequence. In Figure XIV the stress is on the AWAIT actions taken by the words and the resolution of these AWAITs when concepts are created and reported. The point is that no lexical sequence is started as long as there are AWAITs on the small stack of expectations representing part of the WEP process state; in other words, a word starts a lexical sequence when the AWAIT stack is empty. This means that lexical sequences are derived from an important aspect (expectations) of the semantic comprehension process; "low level" constituent structure can be seen as a side-effect of the local expectations posted by the words.

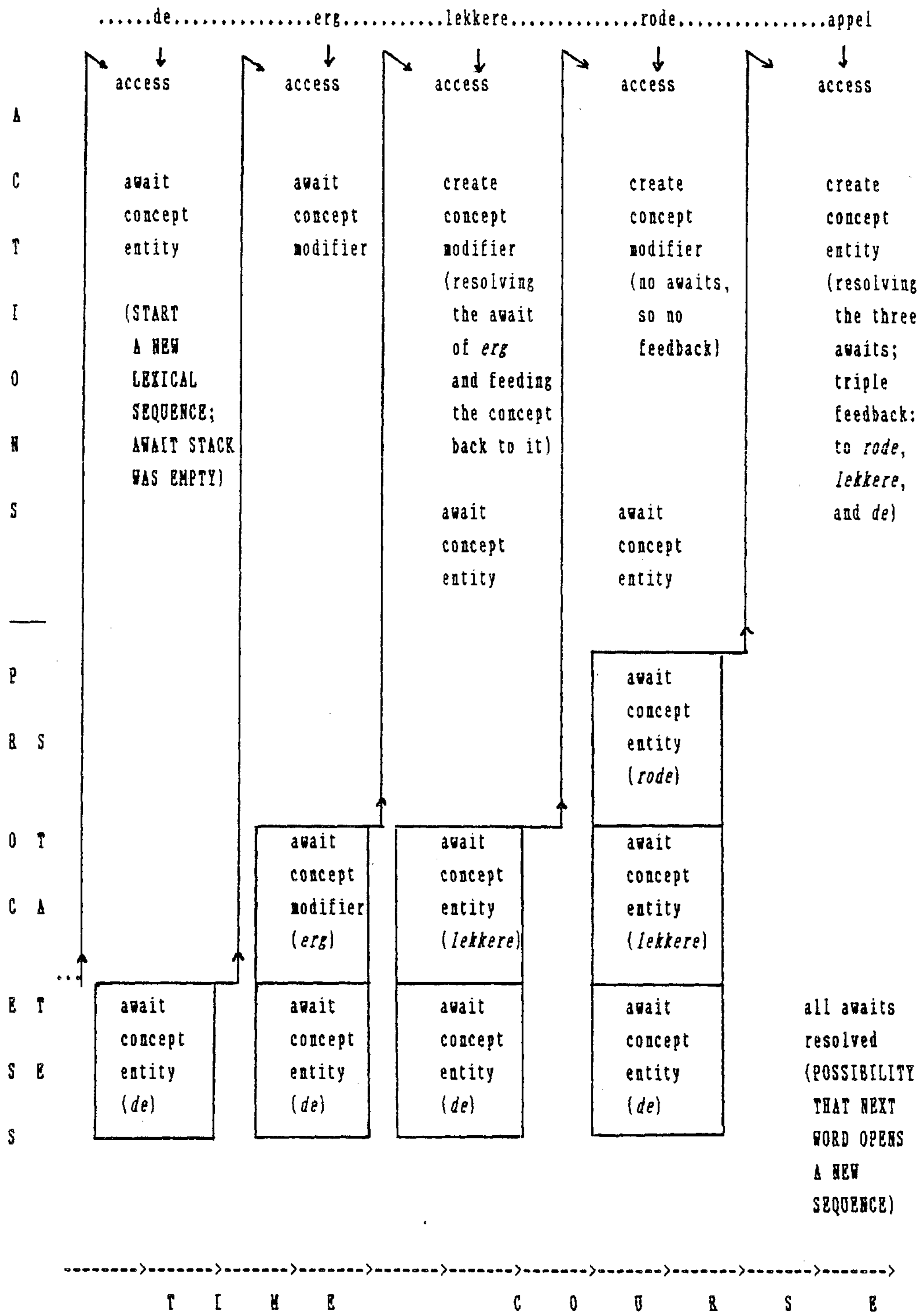


Figure XIV. Noun phrase processing through local expectations.

Because as a static device Figure XIV falls short of its task to model a dynamic (time-bound) process, I will explain how the small stack representing part of the process state grows and shrinks as the words execute and suspend themselves. When the word *de* enters the process, its processual content is accessed, and two actions are taken. First, a probe of the process state (what was done before I arrived?) tells *de* that there are no outstanding local expectations ("awaits") (see 4.3.3.4 for the difference with global awaits), which implies that it starts a new lexical sequence. Second, *de* projects a local expectation that is constrained in two ways (only the first of which shows in Figure XIV): 1) the expectation concerns a specific concept, viz. one of type entity (in syntactic terms: a simple or compound noun or other word(s) used nominally) possibly preceded by concepts of type modifier (adjectives and/or adverbs), and 2) it has a time constraint saying that *de* should only wait as long for its concept as no new lexical sequence is started (possibly by another article, a preposition, a verb, etc.), i.e. as long as there are local expectations in the stack (9). The expectation of *de* is binding, i.e. that if it is thwarted (no noun group arrives in time), the process is fouled up. This reflects the incomprehensibility of e.g. "the comes tomorrow" or "the very is here". (Note that what is called "ungrammaticality" in most linguistic approaches is seen as incomprehensibility as a consequence of unfulfilled expectations in PL, cp. 3.3.3.) The expectation projected by *de* becomes the current process state. *Erg* then posts an expectation for a concept of type modifier along the same lines as the *de* expectation. This adds a second await to the process state stack. When *lekkere* enters the process, a modifier is created; this resolves the expectation of *erg* that receives the "lekkere" ("tasty") concept and refines it as "tasty to a high degree". As a consequence of this resolution, the await for the modifier is removed from the process state; *lekkere* adds another await then, viz. for a concept of type entity (just like *de*). *Rode* adds a third entity concept expectation to the process state stack, and finally *appel* resolves the three awaits by creating a concept that is fed back to *rode*, *(erg) lekkere* and *de* (which can all refine the "apple" concept then; for *de* this refinement also implies feedback to a

(9) Note that a sequence like *de in de zomer genomen beslissing* ("the decision taken in summer") is also analyzed correctly: *in* does not start a new lexical sequence because *de* created a local expectation (so the stack is not empty).

possible "apple" concept introduced earlier in the discourse). With all the local awaits resolved, the next word can start a new lexical sequence. Note also that the importance of a noun as the head of a noun phrase (in structural terms) has its processual parallel in the resolution of multiple awaits upon entry of the noun in the process.

A question that arises when one looks at Figure XIV is of course: what if one assumes that there are more lexical sequences? Do the sequences still "fall out" of the local AWAITs? The answer is yes, provided the local AWAITs for concepts of specific types are kept separate. In the example, this would mean that *de* starts a sequence (no AWAITs for concepts of type entity) and *erg* starts one (no AWAITs for concepts of type modifier), with "erg lekkere" being a sequence within the "de erg lekkere rode appel" sequence. Note that the problem of how many sequences to have arises again (is "erg lekkere" a sequence by virtue of the presence of "erg"? if not, why is "rode" not a one-word sequence?). A more serious problem is that the typing of the AWAITs does not distinguish between prepositional phrases and noun phrases: both prepositions and determiners wait for the same entity concept, which would not allow an NP sequence within a PP (the "await entity" stack is not empty when the determiner starts executing). An easy solution would be to change the type of a concept waited for by a preposition, but this would mean that for the sequences "in summer" and "the summer" "summer" is a different concept depending on whether *in* awaits it or *the*; the combination of the words certainly gives rise to a different concept, but it would be "cheating" to put information that can only be obtained by interaction in the expectations beforehand. Another solution would be to reduce the importance of the AWAITs as determiners of the lexical sequences and simply have certain words start them unconditionally. The advantage of this approach is that it is in accord with the intuitive psychological reality of constituent boundaries (cp. the discussion of the Fodor, Bever & Garrett results in 2.3.4, though). Further research will have to show what the best solutions are for the problems with the relationship between lexical sequencing and concept building.

4.3.3.4. Dynamic caseframes

In a lexically-based system like WEP it is the verb that is responsible for overall sentence understanding. Dynamic caseframes are a processual encoding of the attempt of a verb to "catch" concepts processed before it (if any) and to be processed after it into possible frames (10) and to assign them a (semantic) role. Here again, memory binding mechanisms and expectations (waiting for specific concepts) play a crucial role. Dynamic caseframes are one more example of the processual view of language (no fixed "places" for cases, i.e. for the concepts fulfilling roles in a caseframe) and of the expectation-feedback cycle. For constituent processing (4.3.3.3), we had an instantiation of the EFC with a limited scope (local expectations and feedback); here, we have an instantiation of the EFC with a broader scope (global expectations involving completely processed concepts; see further point 3) below). Together these two EFC types lead to sentence understanding (remember the time-triangle forest described in 3.3.4).

As an example I take the Dutch verb *eten*. As can be seen in the sentences below, Dutch constituent order is more varied than in English (11), which is the reason why dynamic caseframes were developed for Dutch in the first place. It is less predictable in Dutch where -- or rather, when -- a constituent with a specific role will be found; hence the verb process is more complex than in English. Sentences (1) through (5) contain the possible orderings of constituents in simple Dutch sentences; they are referred to by number in Figure XV.

---

(10) Though "dynamic" and "frame" are more or less oxymoronic (a frame suggests a fixed structure), I use "frame" because in the context of cases related to a verb it is a very common term.

(11) Beside the order of agent and main verb in the example sentences, there is also the fact that in Dutch an object can easily be separated from the verb (by adjuncts), whereas in English this is hardly possible (see also 4.3.3.5).

(1) De man eet een appel. (The man eats an apple)	Dutch : agent+main verb (declarative without inversion) English: agent+main verb
(2) 's Morgens eet de man een appel. (In the morning the man eats an apple)	Dutch : main verb+agent (declarative with inversion) English: agent+main verb
(3) Eet de man een appel? (Does the man eat an apple?)	Dutch : main verb+agent (yes-no-question) English: agent+main verb
(4) Waarom eet de man een appel? (Why does the man eat an apple?)	Dutch : main verb+agent (question-word-question) English: agent+main verb
(5) Eet een appel ! (Eat an apple)	(imperative)

Informally, when a verb enters the understanding process, case searching runs like this (see Appendix 2 for examples of formally specified verb processes):

```
i) [if    no concepts have yet been processed ((3) and (5))
   [then  refine sentence structure as question or imperative
   [else  if    the concept is an interrogative word (group) ((4))
          then  refine sentence structure as question
          else  refine structure as declarative

ii) try to find a concept of type entity in active memory
    (short-term processing memory)
    (viz. an animate entity) that can fill the agent role

   [if    found  ((1))
   [then  a)  refine sentence structure as declarative without inversion
   [
   [      b)  try to find a concept of type entity in active memory
   [          that can fit the object role (viz. something edible)
   [
   [          [if    found
   [          [then  refine sentence structure as imperative ((5))
   [          [             unless it was refined as declarative
   [          [                      with inversion earlier ((2))
   [          [
   [          [else  wait for such a concept (global await)
   [
   [                 if    it arrives then OK
   [                 else  (if it does not arrive before the end of
   [                        the sentence) refine
   [                        eten as having an
   [                        implicit object (e.g. "De man eet")
   [
   [else (i.e. agent not found ((2) (3) (4) (5))
         wait for a possible agent (global await)

         - if    agent arrives in time ((2) (3) (4))
           then  refine sentence structure as decarative-with-inversion
                 unless it was refined as a question earlier (which it remains then)

         - goto b) above
```

Figure XV. Dynamic caseframe process of a verb.

Before I go into some interesting linguistic aspects of this process, there are two remarks about the computational realization of dynamic caseframes I would like to make.

First, as can be seen in the full specification of the verbs in Appendix 2 (*eet-*, *houd-*, *bel-*, *word-*), no new questions or actions were needed to implement dynamic caseframes (not used in the application to English). The BINDC and AWAIT actions proved powerful enough to do this. Second, the careful reader may have noticed that case searching by verbs as rendered in Figure XV is an example of a part of a word-expert where ALIASes would have been used in the original WEP version; the actions under b), for instance, are executed from three different points in the process (once immediately after a) and twice from points further down ("goto b"). Each of the question/action subsequences before b) is entered is distinct and the refinements that happen after b) is entered are different too. (Tracing the process for each of the sentences will show the different paths followed.) Thus, ALIASes could have been used here; in Figure XV, a consequence of not doing this (beside the hopefully greater perspicuity) are the unless-clauses that check whether earlier in the process sentence structure had been refined in a specific way or not; depending on the result of the check the refinement is left as it was, or a further refinement results. This shows how ALIASes are avoided through retrieval of traces of the path followed so far.

Some interesting linguistic aspects of the verb process then are:

1) It is semantic in nature: the search triggered by the verb looks or waits for semantically specified agents, objects or complements. As yet, this specification is not very precise, which is a consequence of the stress on horizontal lexical-contextual interaction. As already said, vertical intralexical interaction that implies a fully specified internal lexicon (semantic memory) is not in the focus of attention, but is certainly a high priority issue for further research. Note that syntactic structure specification falls out of discrete points in the semantic search process. It can of course not be denied that observation of linguistic structure (sentence types and their frequencies) has influenced the ordering of the search (i.e. an agent is looked for first), but these structures do not guide the search process; on the contrary, they can be retrieved as a side-effect of the process.

2) Related to 1): dynamic caseframes do not assume structurally fixed positions (as put down in rules, for instance) for the different cases revolving around the verb; the verb is seen as an active "case catcher" that finds its cases at specific points in the process (i.e. time-bound processes dominate over space-bound structures). This also implies that the verb is considered as the word class triggering the most important aspect of the understanding process (specifically in isolated sentences, as opposed to e.g. verbless ellipses in a discourse).

3) As with noun phrase processing, expectations play a central role in dynamic caseframes: the whole process is one of either catching concepts in memory or else waiting for them. However, as already mentioned, the type of the expectations is different from that of the ones discussed in 4.3.3.3; we have a different instantiation of the EFC here. Verbs wait for completely processed concepts, whereas determiners (or prepositions) take care of "pre"-processing these concepts through local expectation resolution. Thus, the scope of the two kinds of expectations is different: expectations attached to determiners or prepositions are local, whereas those projected by verbs have a wider scope and are more global. Both types have the same kinds of constraints, but for the local expectations the concept type constraints are stronger than the time constraints; for verbs the time-course of the (non)arrival of the concepts is more important because it determines sentence type.

4) Note also that possible different caseframes are related to each other; for *eten* the transitive frame and the implicit-object frame are related through the expectation for the object: if it does not arrive in time, the implicit-object frame is automatically chosen; the application of the latter frame depends on the failure of application of the former.

5) Finally, the verb only tries to catch its agent and object(s); adjuncts of time, place, manner, etc. are not caught by the verb (unless they obligatorily occur with it), but report themselves as independent concepts in the concept structure of the sentence, which is the ultimate side-effect of the understanding process (see 4.2.2.2). Of course, this does not exclude the possibility of

interaction between the verb (group) and such adjuncts (or between different adjuncts) if the (idiosyncratic) nature of a verb should require such interaction.

An interesting verb to look at in the context of dynamic caseframes is *worden* (its stem *word-*, see Appendix 2), the Dutch verb for some forms of the passive. It uses a dynamic caseframe that is similar to that of active verbs (note the general applicability of the dynamic caseframe notion), but that looks for different cases at different points during analysis. Instead of waiting for an agent first, it looks/waits for an "object-or-affected" first, and then for a possible by-agent ("door-bepaling"). The "object-or-affected" expectation is a consequence of the ambiguity of *worden* (see 4.3.1). In "De appel wordt door de man opgegeten" (The apple is eaten by the man), "de appel" is finally refined as object after lexical interaction between *wordt* and *opgegeten* (see entries 4 and 41 in Appendix 2); in "Een appel wordt rood in de zomer" (An apple turns red in summer), on the contrary, "een appel" is refined as affected. As can be seen in entries 8 and 9 the latter refinement is a matter of waiting (by *word-*) for the signal *predicative* which (in "Een appel wordt rood etc.") will be sent by *rood* when it finds out in context that it is used predicatively and not attributively; when *word-* catches this signal, it refines "een appel" as affected.

It is interesting to note that both in the application to English and to Dutch understanding passives is mainly a matter of lexical interaction of the verbal elements in the sentence. For English this interaction happens immediately (locally) within the verb group (usually a continuous lexical sequence); it is the *-ed/-en* morpheme that carries most of the responsibility for correct interpretation of the concepts in the sentence (cp. Small 1980, 64-65). For Dutch the interaction takes longer in the overall course of the process (it spans a wider distance in the sentence) because of the discontinuity of the verb group; it is the verb *word-* that carries the responsibility for correct concept interpretation ("case catching"). WEP is powerful and flexible enough to accommodate these language-bound processing differences.

Finally, as announced in 2.3.4, a short note about the resemblance between dynamic caseframes and the lexical structure of verbs in lfg. In lfg the lexical-functional structure of a verb includes the subject of the sentence and not merely its object(s), as in strict subcategorization (i.e. the way verbs are specified in a lexicon in the Chomskyan tradition of generative grammar). Moreover, the idea of fixed canonical positions for arguments in the subcategorization frame of a deep structure verb phrase is rejected; a lexical element may reflect differences in surface constituent order, such as *give*(<subj> <obj> <to-dative>) versus *give*(<subj> <dative> <obj>), related by a lexical redundancy rule (cp. Kaplan & Bresnan 1982, Ford et al. 1982, 772-775). The resemblance between the treatment of verbs in lfg and WEP then is that verbs are given "access" to *all* the important grammatical functions in a sentence. However, in lfg the syntactic specification of the verb (in terms of grammatical functions) is much more important than the semantic specification (in terms of cases), and phrase-structure rules play a crucial role in determining the surface location of the subject, object, etc. In WEP no phrase-structure rules are involved; the verb looks dynamically for its semantic cases and derives syntactic information from the time-course of the semantic analysis process. As to the lexical redundancy rules: they can be derived (as an interesting observation) by looking at the verb-expert processes -- different entries take care of the possible word order differences through memory binding and expectations, but play no further role in WEP (nor process linguistics, for that matter). When a new verb has to be implemented, it is more important to look for a similar verb already implemented, and see to what extent the idiosyncratic nature of the new one allows copying of the process of the old one. The overall (idiosyncratic) process is more important than the (general) rules that can be read off from it.

4.3.3.5. Discontinuous constituents

Sentences 1) and 2) contain examples of discontinuous constituents (the parts belonging together are underlined):

1) De appel *wordt* op maandag door Hilde *opgegeten*.
(The apple is eaten by Hilde on Monday)

2) Geert *belt* Hilde op maandag *op*.
(Geert calls up Hilde on Monday)

1) contains a discontinuous verb group and 2) a discontinuous compound verb (*opbellen*, "to call (up)"). The occurrence of these discontinuous constituents is an essential characteristic of Dutch; it has been dubbed the "pincers

construction" (*tangconstructie*) because the discontinuous elements are like the sharp edges of a pair of pincers holding other constituents in between them. To get an idea of how it constrains linguistic structure in Dutch, compare the following English and Dutch sentences, which seem to have some strange complementarity as far as grammaticality is concerned:

Josh calls up his wife on Monday	*Josh belt op zijn vrouw op maandag
Josh calls his wife up on Monday	Josh belt zijn vrouw op op maandag
*Josh calls his wife on Monday up	Josh belt zijn vrouw op maandag op

In English, there is some locality constraint (*up* cannot move too far away from its verb), whereas in Dutch the pincers construction pushes the particle away from the verb.

Discontinuous linguistic elements offer serious problems for linguistic theories that deal with syntax through context-free rules. Earlier versions of transformational grammar, for instance, did not allow discontinuous constituents in deep structures because they are generated by context-free rules; transformations were introduced to take care of "moving" constituents to their surface position (see also Radford 1981, 81-83). Thus, a parsing system based on these theories, or on context-free rules in general needs extra machinery to interpret sentences like 1) or 2) correctly. In Small 1980 and 1983 it is claimed that WEP should have no trouble analyzing those sentences because of its powerful wait-and-see mechanism and its stress on contextual interaction; Small (1983, 256-258) gives an account of how WEP could analyze "Joanie washes the hundred patterned dishes up" correctly, but this analysis was not implemented. The Dutch pincers construction was an ideal testcase to see if these claims were correct. It proved very easy indeed to deal with discontinuous constituents by having the words of such a sequence interact through the sending and awaiting/receiving of signals. For the verb *opbellen*, for instance, *bel-* (the stem) contains an entry that waits for a possible *particle* signal from a particle, and *op* sends this signal when it has found out in the course of its process that it is not a preposition but a particle. I will explain in a little more detail how this happens.

Note first that the WEP lexicon does not contain *opbellen* as an infinitive, but *bel-* and *op* that can pair up through interaction. Of course, when the scope of WEP is enhanced, the infinitive or the non-discontinuous verb forms (in sub-clauses, e.g. "Ik wil dat je haar *opbelt*", badly translated "I want that you call her up") have to be included in the lexicon; I will not go into the nature of the processual information that will be associated with those forms, or the way they will be related to *op* and *bel-* here; see also 4.3.2. The expectation posted by the verb is simple; it can be found in entries 11 and 12 of the *bel-* expert in Appendix 2. Note that the AWAIT in entry11 does not contain a timeout condition (i.e. there is no *else* part); this simply means that nothing happens if no *particle* signal is sent (as in "De man eet veel"): the expectation dies. It is interesting to mention here that entry12 -- entered if the *particle* signal arrives -- contains a new action (ADDLEX) that was introduced to bring the lexical elements in discontinuous sequences together; in this case, the particle ("w1" in entry12) is added to the slot for the lexical sequence of the verb concept ("c1" in entry12). How the *op* expert determines whether it is a preposition or a particle can be found in its entries 1 and 5: *op* starts by assuming that it is a preposition and waits for the concept that should follow it in that case; if this concept does not arrive in time (i.e. before a sentence break), it knows that it is a particle and signals this to the WEP system (entry5), where the restart demon of the verb (created in the AWAIT for the signal) catches it and makes the verb process run again (see entries 11 and 12 of *bel-* -- or *eet-*, for that matter). This simple process leads to correct interpretation of a number of variations on the *opbel* sentence:

1) Geert *belt* Hilde *op*.
2) Geert *belt* Hilde op maandag *op*.
3) Geert *belt* Hilde op maandag. (no particle)
4) Geert *belt* Hilde *op* op maandag.

1) is straightforward. In 2) the first *op* finds a concept ("maandag") before the end of the sentence and integrates it into a prepositional time adjunct; the second *op* is a particle like in 1). In 3) the AWAIT for the *particle* signal posted by *bel-* simply times out. 4) is the most interesting case. The first *op* starts waiting for a concept; then the

second *op* starts waiting for one in turn, and gets the "maandag" concept (i.e. the last AWAIT has priority over the first; AWAITs are handled by a last-in-first-out stack, cp. 4.3.3.3); this concept is incorporated into the time adjunct and becomes inaccessible to the first *op*, whose AWAIT times out at the sentence break so that it sends the *particle* signal correctly. To deal with cases like 4) containing a sequence of two equal words an approach leading to more rapid understanding would be to peek at the word to the right of the first occurrence of e.g. *op*; when it is the same, the interpretation of the first occurrence as a particle is more immediate than when the correct interpretation is delayed until the end of the sentence. Though this approach seems ad hoc (the more so as natural languages avoid consecutive occurrences of equal linguistic elements, which makes such cases rare), it has the advantage of greater psychological plausibility: in a sentence like 4) we do not have to wait for the end of the sentence to interpret the first *op* as a particle. The occurrence of a second *op* (or any other preposition) immediately blocks the preposition interpretation of the first one. I leave this matter open, and only make the remark that a more radical approach stressing idiosyncrasy over generality might choose to make "ad hoc-ness" the general "rule" and general rules ad hoc (favoring the latter approach to prepositions and particles).

The discontinuity of verb groups is handled in the same way: a potential auxiliary (such as *wordt* in "De appel wordt door Hilde opgegeten") waits for a signal from a verbal element (viz. a past participle) arriving later in the comprehension process. This participle figures out in the context whether it is used as an adjective or not. In the latter case, it sends the signal *complete-action* (meaning that a verbal element was found in context), the auxiliary catches this signal, and both parts of the group pair up nicely; in the former case no such signal is sent and the expectation of the potential auxiliary times out. (Remember, however, that a word can wait for several signals and/or concepts at the same time, depending on its usages; *wordt* waits for the signals *complete-action* and *predicative* -- sent by *rood* in "De appel wordt rood" -- simultaneously.)

A final remark about syntactic discontinuities: beside the fairly simple cases of split constituents discussed here and handled easily by WEP, one could consider "filler-gap sentences" (long-distance dependencies) like "Who <= filler> do you think I saw <gap>?" or "He wondered who <= filler> his sister had been seeing <gap>" as complex cases of syntactic discontinuities. Though no attempts have yet been made to make WEP analyze such sentences, I believe that the wait-and-see strategy and the signal/concept communication of WEP can handle these cases without great difficulties (and certainly without the necessity of introducing complex additional formal machinery as in lexical-functional grammar and parsers based on it -- Bresnan's "bounded domination metavariables" (Bresnan 1982, 231-263)). An analysis of "Wie denkt hij dat je gezien heb?" ("Who does he think you have seen?") in WEP would not be a matter of trying to fill gaps (a spatial or structural view of the phenomenon) but of the dynamic caseframes of the verbs trying to catch concepts in memory at specific points in the process. The dynamic frame of *denk-* could be written in such a way that it does not immediately assume that a potential agent concept processed earlier (viz. "Wie") is its agent, but that it checks the concept following it first; if this is also an agent candidate (as in the example, *hij*), *Wie* will be left uninterpreted by the *denk-* expert. When *ge-* and *zie-* run later on and *zie-* tries to bind a concept into the object role, *Wie* is still available and is correctly caught by it. This interpretation process is a matter of concepts being rejected for certain roles and remaining available for others; as such, in the sentence "Wie denkt hij dat je gezien hebt?" *Wie* is not seen as a constituent "moved" from a position further down in the sentence, but as a concept left uninterpreted by the *denk*-caseframe running before the *zie*-caseframe. In structural terms this means that "...denkt hij dat..." is seen as an interposed clause even though it is the main clause syntactically; I believe that "Wie heb je gezien?" is the more important part of the requested information. Now, if we consider this last question as a sentence in its own right with a specific word order (no "movement" of constituents whatsoever), all that happens in the sentence "Wie denkt hij dat je gezien hebt?" is the enlargement of the distance between "Wie" and "gezien hebt" (in spatial terms) or the temporary uninterpretability of "Wie" (in temporal terms), as reflected in the suggested parsing by WEP (see also 5.2.3 for a discussion of the use of filler-gap constructions in lexical expectation research in psycholinguistics).

4.4. WEP scope for Dutch

In 4.3.1 I have listed the experts implemented for Dutch; by way of light note to end the discussion of the WEP revision and application to Dutch, I list some nice sentences that WEP can analyze correctly (without further comments). In Appendix 3 a complete annotated parsing trace of one of them is shown.

In de zomer houdt de man een appel in de hand.
De man houdt (veel) van zijn vrouw.
Houdt de man (veel) van zijn vrouw?
Op maandag belt Geert Hilde op.
Geert belt Hilde op op maandag.
Geert belt Hilde op maandag op.
Een appel?
In de zomer.
Eet op !
Belt de man een vrouw op?
Belt de man zijn vrouw op?
Eet een appel !
De appel wordt (door de man) opgegeten.
De appel wordt opgegeten door de man.
In de zomer wordt een appel rood.
De man houdt veel van vrouwen.
De man houdt van veel vrouwen.
Veel mannen houden veel van veel vrouwen.
Hilde houdt van zijn haar.
Hilde houdt van haar haar.
Geert eet.
Geert eet veel.
Geert eet veel appels.
Geert eet veel appels op.
Op maandagen bellen vrouwen mannen veel op.
De man wordt veel door een vrouw opgebeld.

CHAPTER 5: WEP CONFRONTED WITH PSYCHO- AND NEUROLINGUISTIC RESEARCH.

"There is no substitute for human intelligence."
(Fodor, Bever & Garrett 1974, very last sentence)

5.1. Introduction

After looking at natural language understanding from the linguistic and the AI points of view, we will now come full circle by taking the psycholinguistic (and neurolinguistic) perspective. More precisely, the way WEP works (and through it, process linguistics) will be confronted with recent research findings in psycho- and neurolinguistics dealing with the mental lexicon and the way its information is processed in comprehension. In this way the merits and flaws of the computer model as a model of human language understanding (what it wants to be) can be pointed out, as well as predictions the model makes about human natural language understanding. This implies that the purpose of this chapter is twofold: one, it wants to give a justification of WEP characteristics in the light of what is known about human language processing, and two, it wants to suggest that some of WEP's characteristics can inspire further psycho- and neurolinguistic research.

At this point it is important to reiterate that WEP fits in with a cognitive-science framework incorporating *weak* AI (see 1.3.2): WEP only *simulates* human behavior on a computer, and no further parallels are drawn between the human being and the computer. Computer simulations can show us the incompleteness or impreciseness of our ideas about cognitive processes or of our linguistic descriptions but they should not serve as a metaphor for human cognitive functioning. In accordance with the adherence to weak AI, the type of equivalence claimed to exist between simulated and simulating processes is also only a weak one (cp 3.2). This means that comparisons between WEP functioning on the one hand and psycho- and neurolinguistic research on the other stay on the (higher) functional level; nothing is said about possible (impossible) parallels between the realization of cognitive processes in the human brain and the realization of a simulation of these processes on computers (be it on the software or hardware level). To give an example: on the one hand, we have the importance of expectations + feedback in WEP; on the other, we have research into the role of word-bound (lexical) expectations. If the research shows that dynamic expectations are indeed real and important in understanding, then WEP can be said to model (simulate) an important aspect of the understanding process. Yet, nothing is said about Lisp coroutines, the implementation of Lisp, and their "parallels" with brain functioning. In the other direction (from WEP to psycholinguistics), WEP suggests that research should also look at how and when feedback to expectations fed forward plays a role. Is multiple feedback (as modeled in WEP) real in a sense that it causes processing difficulties, i.e. are there signs of slower reaction times when multiple feedback has to take place (from a concept to words expecting it)? If not, does this feedback happen in parallel, or is the whole idea "unreal"? Although to some scientists (cp. Pylyshyn 1984), the unwillingness to look for or claim strong equivalence may seem unattractive, I repeat that it seems safer to me to only claim weak equivalence than to assume strong equivalence and be left with untestable claims about parallels between computers and human beings. To restate the computational paradox: if computers can teach us something about cognitive functioning, then it is in the first place that this functioning is very much *unlike* their own functioning. But considering the as yet unsurmountable problems with directly penetrating cognitive processes, any hint as to how they work should be welcomed, also if it comes from machine functioning.

Finally, it should be mentioned here that the first short exploration of the relationship between WEP and cognitive psychology was undertaken by Small and Lucas (1983, 48-60). Their paper has been the point of departure of 5.2 (WEP and psycholinguistics). I have incorporated and updated their findings about lexical access, idiom processing and function words in sections 5.2.2, 5.2.4 and 5.2.5. The other topics dealt with in this chapter (lexical expectation, 5.2.3; morphology, 5.2.6; model lesioning, 5.3.1; parallelism, 5.3.2) add new material to Small and Lucas' exploration.

5.2. WEP and psycholinguistics

5.2.1. Introduction

When I criticized the autonomy of syntax thesis in 2.2.2.2 I also introduced the two general psycholinguistic models of language processing that one currently finds in the literature. One of these models (the autonomous component model) fits in nicely with generative grammar, whereas the other (the interactive model) fits in with process linguistics. It need hardly be repeated that in WEP comprehension is also viewed as a highly interactive process, and that the program then simulates the interactive model of language processing. Understanding happens on a word-by-word basis, all knowledge can be brought to bear right away (word-experts have access to knowledge on all levels, from morphology to pragmatics), context plays an important role (lexical-contextual interactions), and information that becomes available (such as a just created concept) can immediately be accessed by the experts waiting for it.

As discussed in 2.2.2.2, one of the interactive models was developed for spoken language understanding (Marslen-Wilson & Tyler 1980), whereas another was developed for written language understanding (Just & Carpenter 1980). Since WEP is meant as an analysis program for written text (cp. the modality-boundness of process linguistics), it comes closer to Just & Carpenter's model. Beside the compatibility of the general characteristics of the model with WEP, it is worth pointing out that Just & Carpenter's research has shown the unmistakable reality of the word as reading unit:

> "There is a common misconception that readers do not fixate every word, but only some small proportion of the text, perhaps one out of every two or three words. However, the data (...) show that during ordinary reading, almost all content words are fixated. This applies not only to scientific text but also to narratives written for adult readers" (1980, 329-330).

Hence, associating processes at work during reading comprehension with the words themselves (as is done in WEP) finds support in psycholinguistic research. Although Just & Carpenter attach great importance to the duration of word fixations in reading (it is assumed to reflect processing time directly -- the "eye-mind assumption" (1980, 330)), it is a pity that they did not take backward fixations into account. In other words, they did not study to what extent readers go back to words read earlier when they are trying to understand text (only consecutive fixations on a word were studied in their experiments). The reason why I mention this is that it might have been interesting to see whether there are any parallels between backward fixations and the way control is passed back and forth between the experts as they execute and suspend within the coroutine regime (1). Of course, this coroutine regime is meant in the first place to simulate a mental reality (it is the processes in the mind that communicate with one another), but it might be the case that words executing repeatedly in the WEP program (like verbs) are also words whose processes are "refreshed" from time to time by fixating them again. It would be interesting to test this prediction to see whether in reading re-fixating words occurs as a way to assist reading comprehension. If it occurs, it could maybe be explained as a way of making up for the lack of the clues one has in spoken language understanding (intonation, extra pauses, stress, gestures, etc.). Written language is a derived and impoverished medium, but a reader is capable of making up for this by controlling the input rate and possibly re-fixating important words. Note that in spoken language this type of control is hardly possible (unless the listener asks the speaker explicitly to repeat things), and no externally stimulated re-fixations of words as they occurred literally (e.g. by "looking" at them) is possible: spoken text cannot be "rewound" (unless it is on tape, of course), whereas a written text does not disappear as it is read. I will come back to Just & Carpenter's research results in 5.2.5 when I discuss research into the distinction between content and function words, one of the specific topics addressed in the subsections to come.

---

(1) I stress that this comparison does not imply a strong type of equivalence. It is merely the case that the obvious analogy between control in a coroutine regime and backward fixations in reading *suggests* that the latter may have an important function in reading.

5.2.2. Lexical access of ambiguous words

A question that has been the subject of a lot of psycholinguistic research in the last decade is the following: does context restrict lexical access so that only the contextually appropriate meaning of a word is accessed (the Prior Decision Hypothesis) or do we access all meanings temporarily, with the context aiding selection of the appropriate meaning after access (the Post Decision Hypothesis)? Whereas early research yielded mixed results (2), recent work has produced results that support the post decision hypothesis (3): for noun-noun ambiguities (e.g. *bug* = insect or microphone) as well as for noun-verb ambiguities (e.g. *rose* = "flower" or "stood up") it was found that *all* meanings were accessed, including the contextually inappropriate one(s). Cases in which prior decision showed up in the results (Seidenberg et al. 1982) -- viz. with noun-noun ambiguities in a highly constraining context -- were more readily explained by automatic intralexical network priming (i.e. a word occurring before the target was strongly semantically related to one reading of the ambiguous word) instead of by contextual influence. Moreover, attempts to induce priming based on other types of contextual information (syntactic or pragmatic) have failed, which strengthens the idea that the (exceptional) prior decision cases were a matter of vertical intralexical interaction and not of horizontal contextual interaction.

WEP operates in accordance with the post decision hypothesis: the interactive disambiguation process starts after retrieval of all the word's meanings. It is interesting to note that an earlier attempt at constraining word-experts by *prior* pruning (Rieger & Small 1981) was abandoned; in fact, that prior pruning proved very hard to do helps us understand why human beings access all meanings of a word: exhaustive access with pruning after is less resource-consuming than the use of information to restrict access. However, WEP cannot account for intralexical priming effects, since the experts are not connected into a semantic network through which these effects could be spread; the introduction of such a network to account for vertical intralexical interaction is a high priority issue in future research.

---

(2) See e.g. Lackner & Garrett 1972, Foss & Jenkins 1973, Conrad 1974, Swinney & Hakes 1976, Holmes 1977.

(3) See Swinney 1979, Tanenhaus et al. 1979, Seidenberg et al. 1982 and 1984.

5.2.3. Lexical expectation

Since word-bound expectations play a central role in the WEP understanding process (through the AWAIT action), we will take a look at the psycholinguistic research into this topic.

It has repeatedly been suggested that during understanding people use lexical information about the possible syntactic or thematic frames of words (especially verbs): anticipation of verb complements can guide understanding of words and phrases in a sentence (Fodor & Garrett 1967, 1968; Clark & Clark 1977). Fodor & Garrett (1968) have shown that the existence of possible alternative frames of a verb influences the ease with which complex sentences are paraphrased or sentence anagrams are solved. Lexical expectation has also been related to the understanding of filler-gap constructions (e.g. "Who <=filler> do you think I like <gap> ?"): Fodor has suggested that the relative ranking of subcategorization options for verbs plays an important role in understanding filler-gap constructions; Clifton et al. (1984) have shown that grammaticality judgements about certain filler-gap sentences were quicker when sentence structure matched the preferred verb frame (viz. transitive or intransitive), which was interpreted as possible evidence for the use of verb frame information to guide gap postulation; finally, Stowe (1984) and Clifton et al. (1984) have suggested that for gap-finding and -filling pragmatic information about the plausibility of the filler as an object for a (transitive) verb is also important. Ford et al. (1982) have further demonstrated that preference for specific verb frames shows up in a task like sentence meaning paraphrasing: the ambiguous sentence "The woman wanted the dress on that rack", for instance, was usually interpreted as subject-verb-objectNP and not as subject-verb-objectNP-complementPP.

It will be clear that none of the above findings are the result of experiments with on-line comprehension and as such can only give indirect evidence for on-line use of lexical expectations. Luckily, experiments testing lexical expectations (especially the transitive/intransitive distinction) during comprehension are starting to emerge (Clifton et al. 1984, Tanenhaus et al. 1985); since they are not numerous, I will take a closer look at them, especially at Clifton et al. 1984.

Clifton et al. had subjects read sentences of the types illustrated in Figure I. The idea of the experiment was the following: verbs like *read* have a preferred transitive frame, whereas verbs like *sing* have a preferred intransitive one; at the same time they can also be used intransitively and transitively respectively, but this is the less common reading of the verb (4).

verb preference	sentence form	sentence
(1) transitive	transitive	The babysitter read the @ story to the sick child.
(2) transitive	intransitive	The babysitter read to @ the sick child.
(3) intransitive	intransitive	The babysitter sang to @ the sick child.
(4) intransitive	transitive	The babysitter sang the @ story to the sick child.

Note: sentence presentation interrupted at @ for lexical decision task.

Figure I. Sentences used by Clifton et al. (1984) (Figure adapted from their Table I p. 698).

(4) Both Clifton et al. (1984) and Tanenhaus et al. (1985) used the norms for verb frame preferences determined by Connine et al. (1984), who asked subjects to write sentences about specified topics with a number of verbs and counted the frequency of each of a variety of syntactic constructions. These constructions were defined in terms of the syntactic categories of the complements of the verb (i.e. NP, PP, NP PP, inf-S, etc.; cp. the strict subcategorization frames in the Chomskyan tradition).

A prediction that follows from this is that sentences with transitive verbs used transitively (case (1) in Figure I) should be easier to understand than when the verbs are used intransitively (case (2)) (the same reasoning applies for intransitives (cases (3) and (4)). To test this prediction, the subjects were given a lexical decision task (5) at point @ (see Figure I) during the word-by-word presentation of the sentence; in the "easy" sentences ((1) and (3)) reaction times should then be shorter for the secondary task than in the "difficult" ones ((2) and (4)), reflecting that in the former cases sentence processing was easy enough for the subjects to be able to divert their attention to the secondary task, whereas in the latter cases sentence processing difficulties resulted in slower reactions to the additional task. The results confirmed the hypothesis: subjects were faster on the lexical decision task when the phrase following the verb matched its preferred frame than when it did not. The final interpretation by Clifton et al. is that "lexical information can have an effect at a stage of sentence processing prior to combining the meanings of the words in a sentence into a coherent representation" (1984, 699). They motivate the "prior to" by observing that the secondary task was presented before the semantic content of the NP or PP following the verb was available to the readers; thus, the readers could only determine that a (syntactic) NP or PP would follow.

Tanenhaus et al. (1985) tested lexical expectations with transitive/intransitive verbs in filler-gap constructions, looking at the same time for the possible influence of pragmatic information (viz. the plausibility of the filler). Their results confirmed those of Clifton et al. in that lexical expectation (viz. about (in)transitivity) was found to control gap detection and gap filling; plausibility effects also showed up in the results, but more clearly with transi-

---

(5) In a lexical decision task subjects are asked to decide as quickly as possible whether a string presented to them is a word or not (they push a button or key on a terminal board to indicate their decision). The task is usually given to subjects in the course of another task (the target task of the experiment; for Clifton et al. this is reading and understanding sentences), with the idea that reaction times in the lexical decision task reflect ease/difficulty in dealing with the target task.

tive verbs than with intransitive ones.

Now, how does the way WEP models lexical expectations (viz. of verbs) relate to these research findings? It will be clear that the adherence to the interactive model of language understanding in general, and the use of dynamic caseframes in particular imply a critique of Clifton et al.'s interpretation of their results. Recall that dynamic caseframes are processes that look/wait for semantically specified role-fillers, whose syntactic realization as an NP or PP is derivative; moreover, once again, they are not like strict subcategorization frames assuming fixed canonical positions for verb objects or complements, but they deal with case-catching in a time-bound dynamic way. Clifton et al. try to make their results fit their bias for a syntax-first comprehension process (the autonomous component model) in a way that is not convincing at all. They take their results to show that lexical expectation for transitive/intransitive verbs is a matter of expecting syntactic constituents (NPs or PPs), supporting this claim by the observation that the lexical decision task came *before* the arrival of the semantically important noun (phrase). However, it is clear that this noun (phrase) is not needed for lexical expectation to be of a semantic kind: it is the verb itself that may easily project a semantically determined object expectation, with the syntactic realization as an NP, pronoun or whatever of secondary importance. In the experiment, the verb had been processed before even the arrival of the determiner/preposition following it, so its possible semantic expectations were active before the arrival of the noun (phrase) later on. In short, their results can just as well be interpreted as showing *semantic* lexical expectations, as modeled in WEP. The distinction implied in a syntax-first approach between "prestored preferences for linguistic structures" (influencing comprehension above and before all) and "preferences for analyses computed on line" (Connine et al. 1984, 318) looks like an artificial one from the standpoint of the interactive model and the dynamic caseframe notion in WEP: understanding goes straight for the meaning in a dynamic way.

Another item of (indirect) criticism of the lexical expectation research also follows from the dynamic caseframe notion with its rejection of structurally fixed positions for the cases revolving around the verb. As said above, the research into lexical expectations often makes use of filler-gap sentences (e.g. "Who <=filler> do you think I despise <gap>?"). Now, those sentences are never seen as sentences in their own right, but are always related to some other (normative) structure (the deep structure from transformational grammar (Fodor 1978) or another fixed subcategorization frame that assumes an object to follow its verb). The filler-gap view of those sentences rests on an assumption of movement of constituents from some position where they "ought to be". The critical comment on this view is partly based on the difficulty of positing a "gap" in Dutch sentences of the type above. Whereas in English verbal groups are always found nicely together -- with objects following the verb group, part of the typical structure of Dutch is that the verb group is often split with objects and complements occurring obligatorily *between* e.g. an auxiliary and the past participle (the *pincers construction*, cp. 4.3.3.5). Thus, when a present perfect is used, for instance, an object can *never* follow the meaning-bearing past participle, but is always somewhere in front of it (between the auxiliary -- with many possible functions -- and the participle, or in front of the auxiliary even. Compare:

(1) Ik heb een man gezien
("I have a man seen", I have seen a man)

(2) Ik heb gisteren een man gezien
("I have yesterday a man seen", Yesterday I saw a man)

(3) Gisteren heb ik een man gezien
("Yesterday have I a man seen", Yesterday I saw a man)

With these sentences in mind, it is hard to see where "gaps" for objects are to be located in sentences like "Wie heb je <gap?> gisteren <gap?> gezien <gap?>?" (Who did you see yesterday?). Moreover, Dutch does not have the "stranded preposition" construction used for the intransitive cases in the filler-gap experiments. The preposition occurs together with the "moved" NP: "Met wie was je <gap?> aan het praten <gap?>?" (Who were you talking to? *Met = to*). Since going into these differences here would lead me too far, I only stress once again the point about the non-fixation of objects in specific positions: finding objects is something that happens in the *time* course of the verb process and is not necessarily related to assumed positions in sentence structure.

Of course, these comments are easier made than tested, but they suggest at least that using strict (syntactic) subcategorization frames to test lexical expectations, or else as a complementary factor the (pragmatic) plausibility of the filler in filler-gap constructions (the Tanenhaus et al. 1985 research) may be insufficient to find lexical preferences for verb frames, certainly if one favors an interactive model in which meaning determination is the driving force.

5.2.4. <u>The processing of idioms</u>

Idiomatic expressions have long been the object of study by linguists (see e.g. Fraser 1976, Makkai 1972, Fernando & Flavell 1981, Gross 1984): to what extent are idioms (still) flexible, i.e. what syntactic operations do they allow ? how are they used in word play, e.g. through a subtle interplay of their possible literal interpretation and their (more or less) frozen idiomatic interpretation ? etc. (This last question implies that certain lexical sequences can be ambiguous between a literal meaning (composed of the meanings of its parts) and a (more or less) frozen idiomatic one.)

In psycholinguistics too, idioms have been studied, be it from a different perspective. The question is whether idioms are processed in a manner similar to other lexical sequences or instead require a special processing mode. Here again, as with lexical access, two hypotheses are put forward, with one of the two getting most of the support from experimental research. The Idiom List Hypothesis (Bobrow & Bell 1973) holds that idioms are stored in (and accessed from) a special list which is not part of the normal lexicon, and that a literal analysis is always attempted on a word string before an idiom mode of processing is undertaken. The Lexical Representation Hypothesis (supported by most of the results, see Swinney & Cutler 1979, Estill & Kemper 1982, Glass 1983), on the contrary, holds that idioms are stored in and retrieved from the lexicon in the same manner as any other word, and that both the literal and idiomatic meanings are computed upon occurrence of the first word (with contextual disambiguation choosing the appropriate one, cp. the Post Decision Hypothesis in 5.2.2). However, some results by Ortony et al. (1978) and Gibbs (1980) showed that idiomatic interpretations are reached more quickly than literal ones; although Estill & Kemper (1982) failed to replicate these results and attribute them to post-retrieval processes (contextual integration and postcomprehension paraphrasing abilities), they suggest that there may be a bias towards idiomatic interpretation in cases where a literal one seems highly improbable or rare (e.g. "let the cat out of the bag").

As an instance of the stress on idiosyncrasy in language, one of the linguistic ideas behind WEP was that all fragments of language are *more or less* idiomatic (cp. Bolinger 1979; Small 1983, 248-250), disputing the sharpness of the idiom notion. As a consequence, WEP portrays idiom processing as an instance of normal comprehension where the degree of idiomaticity is reflected in the relative importance of idiosyncratic interactions (e.g. the LITERAL question) that dominate for highly idiomatic expressions on the one hand, and more global interactions (e.g. the BINDC question) that play a more important role for less idiomatic linguistic units. Thus, WEP does not assume any type of special mode of access and further processing of idioms; they are treated by the same overall disambiguation process as any other input but the types of interactions involved during comprehension/integration with the context may vary. This is in accordance with the Lexical Representation Hypothesis. However, assuming that it exists, a bias towards the idiomatic interpretation of expressions is not incorporated in the model. Since future research will try to incorporate frequency of occurrence of particular meanings of linguistic elements over others (as observed in large corpora or obtained by giving subjects specific tasks such as constructing sentences with certain verbs, for instance, to check their preferred caseframes (cp. Connine et al. 1984)) into the disambiguation process, it will also be considered to give priority to idiomatic interactions over literal ones in specific cases.

5.2.5. Function and content words

In 3.3.4 the distinction between function and content words was discussed in the context of the importance of the lexicon in process linguistics, with the stress on the processual differences between the two types (rather than on the descriptive differences usually made in linguistics). In psycholinguistics (and neurolinguistics, see 5.3.1) the question has been whether there are differences in the way function and content words are stored and processed during language comprehension. Hence, an interesting topic of research whose results are worth comparing to the way WEP models content and function word processes.

From studies by Garrett (1976) there is some evidence that function and content words are processed differently: analysis of speech error data showed that the two classes do not typically interact in the production of errors (e.g., an exchange of words would involve two content words, not a content and a function word) and that they seem to have their own kind of errors. Content words are usually involved in exchange errors (e.g. "we have to *gap* the *bridge*"), whereas function words (especially bound morphemes) do not exchange but are shifted to another word (e.g. "I'd forgot about*en* that"). Since these data pertain specifically to speech production, and WEP is a model of (written) language comprehension (cp. the modality-boundness of process linguistics), I will not go into them any further, concentrating instead on lexical access and eye-fixation research.

Bradley (1978) looked at frequency effects in recognizing function versus content words. While the usual finding in psycholinguistic research is that higher frequency words are recognized faster, Bradley only found this frequency effect with content words, but not with function words. Her conclusion that caused much excitement in the psycholinguistic community was that there is a separate non-frequency-sensitive accessing mechanism for function words -- stored outside the general lexicon -- in linguistic processing. However, recent research (Gordon & Caramazza 1982, Garnsey & Chapman 1985) has failed to replicate Bradley's results, which implies that there are no differences between function and content words in initial access (but rather in post-access processing). It is interesting to note that these results, taken together with the lexical access (5.2.2) and idiom processing (5.2.4) ones seem to converge on a view of human information accessing as a uniform, automatic, exhaustive and independent processing mechanism (see also 5.3.1).

Here again, WEP is in accord with the research findings: function and content words are stored and accessed in the same way, but post-access processing differs for the two classes (cp. 3.3.4). Content words require significant processing of the sense discrimination variety (building and refining concepts), whereas function words (and inflectional or derivational morphemes) aid correct functional interpretation of those concepts through their "concept catching" actions and through the sending and receiving of control signals. I will come back to this when I deal with aphasia and lesioning of computational models (5.3.1).

Just and Carpenter (1980) have looked at eye-fixations during reading of scientific text (in the context of their interactive model of reading comprehension). They found that almost all content words are fixated (with considerable variations in fixation duration), whereas (short) function words (*of*, *the*, *a*) are not always fixated. Within their model of written language understanding they propose that gaze durations reflect the time to execute comprehension processes (longer fixations implying longer processing). Though they extensively discuss processing associated with content words (e.g. also disambiguation), they do not talk about the strange result that function words are hardly fixated. Within their framework this probably implies that function words are processed very fast and do not strain the comprehension mechanisms, which are more concerned with semantic processing of the content words. WEP cannot account for these results since the model has nothing to say about the relationship between the time course of its operations and that of real-time processing; moreover, traces of sentence parses show that the executions of function words and inflectional or derivational morphemes constitute a large part of the process (see e.g. Appendix 3). This may be a flaw in the model, but I also suggest that not fixating a word does not necessarily mean that it plays no significant role in processing, which implies an extension of Just & Carpenter's model.

Let me briefly stress here that the fact that WEP has nothing to say about the relationship between the time-course of its (computer) operations and that of on-line cognitive processes is not a shortcoming of the model but rather an inevitable consequence of the totally different nature of computer processing and cognitive processing. For one thing, computational processes may take a certain amount of time due to computer-bound (or computer-language bound) phenomena (6); for another, we do not have a very clear idea of the time-course of cognitive processing either. Once again, a phenomenon that inspires a retreat to a position in which only weak equivalence between human and computer processing

---

(6) For instance, process (over)load, the difference in speed of execution of Lisp on a VAX versus on a Lisp-machine, the way Lisp functions ar written (efficiently or inefficiently), etc., all things that have nothing to do with the time-course of cognitive processing whatsoever.

can be claimed (cp. 1.3.2 or 5.3.2).

5.2.6. Morphology

In the discussion of how WEP deals with morphology (4.3.2) I described how the system works now, and how it could be revised in order to better deal with the many idiosyncrasies of (derivational) morphology. (I will repeat the essence of the revision below). Beside being based on the linguistic observation of idiosyncrasy in morphology, this revision was also suggested by psycholinguistic research into on-line morphological analysis by the human being and into the nature of representations in the mental lexicon. I present this research in support of the view on morphology in both WEP and process linguistics.

Butterworth (1983b) gives an overview of the literature (deploring, by the way, the lack of convincing research in the area of lexical representation and on-line morphological analysis). Two of the topics he discusses are of particular interest in the context of WEP and morphology. The first topic is the rule/list controversy: are inflected and derived forms listed explicitly in the lexicon (the Full Listing Hypothesis (FLH)) or are only the base forms listed, and all other forms derived or interpreted by rules (the classical linguistic approach) ? No solid evidence for either hypothesis seems to emerge from research into these matters. As with function and content words, errors in speech production of normals (and aphasics) suggest that stems and bound morphemes lead separate lives in the production process, in that the bound morphemes are selected independently of their stems and added to them at a relatively late stage in production (1983b, 266-269). Butterworth challenges this "evidence" for the base form + rule hypothesis by suggesting that a supporter of the FLH (like himself) can easily accommodate these results in that he is not forced to deny that affixing rules are known to language users. The assumption that completes the FLH is that those rules are not routinely used by people: anything not available via a rule *must* be separately listed, but availability via a rule does not necessarily imply that there is no other way of finding some morphologically complex form (viz. in a list). This is related to his view of rules as "fall-back procedures", a view that has also influenced the discussion of processing competence in 3.3.3. Rules are probably known to language users to varying degrees, but not used in verbal behavior, where much information is routinely accessed. (This is one of the many expressions of dissatisfaction with a direct mapping of rule-based approaches to processing models (cp. 2.3.2, Marin 1982, Rumelhart 1984)). Evidence from the other modalities (hearing/reading) hardly yields any support for either hypothesis, and Butterworth concludes (rather prematurely) that the lack of evidence for the reigning view of base form + rule application leaves us with the "weaker" alternative of the FLH, provided it is supplemented with some grouping together of morphologically related forms (not necessarily with a base form or other abstract representation as a heading).

This proviso brings us to the second issue discussed by Butterworth: what is the type of unit proposed for lexical representations (words without internal morphemic structure ? words with morphemic boundaries marked ? abstract underlying representations ? morphemes ? etc.). I will not go into the many possibilities proposed by (psycho)linguists, since there seem to be as many proposals as researchers (with not much evidence for any of them). To complete the discussion of Butterworth, however, I simply mention his conclusion about these matters, which is similar to the one above: Butterworth supplements his bias towards FLH with a view of full words as unit type. For the problematic data with this view -- people sometimes produce errors that are morphologically well-formed but non-existing words -- he suggests rules as fall-back procedures, with full words being the input to those rules.

As discussed in 4.3.2, WEP does not operate in accordance with the FLH, but uses a "segmentation first, interpretation after" approach as discussed above. If we assume this approach to be correct, the following view of on-line morphological analysis suggests itself: segmentation happens automatically (like lexical access, see 5.2.2) with post-segmentation processes (viz. lexical interaction) interpreting morphemes in context. To make the parallel with the lexical access research complete: in cases of ambiguity, all possible segmentations would have to be made, with the context choosing the correct one. As we saw in 4.3.2, WEP does not (and cannot) operate in accord with the latter part of this view. Also, problematic segmentation cases seem to imply the necessity of fully listing certain words, with the presence of these words in the lexicon blocking the segmentation process. Thus, this view has its problems and may be incorrect. Listing everything, on the other hand, may not have these problems, but it creates others: it is unclear how the lexicon has to be organized internally, and there is a danger of massive duplication of information. As a consequence of all this, I suggested a different view in 4.3.2 (trying to combine the best aspects of all the views discussed): the segmentation process is done away with, and complex words are fully listed, some with and some without morphemic marking (7). In both cases, lexical-contextual interaction remains the main process bound *and* free morphemes engage in.

5.3. WEP and neurolinguistics

In this last section, I will first discuss some results in studies of aphasia and their implications for models of language understanding. Next, I will briefly go into the issue of parallelism in the WEP model, looking at its relationship to models of brain functioning. This will lead to a brief discussion of what I consider one of the hardest problems for cognitive science, i.e. the mappings between models developed in the different disciplines that constitute the new multidisciplinary field. As announced in 1.3.3.2 Dennett's distinction beteen the design/intentional stance on the one hand, and the physical stance on the other will prove useful in this context.

5.3.1. Aphasia and model lesioning

In recent literature (Gigley 1982, Cottrell & Small 1983) it has been suggested that models of language understanding should be "lesionable" in order to be realistic, i.e. it should be possible to disrupt (part of) the workings of computer models (without substantial reprogramming) in a way that does not lead to a complete breakdown of the system. This requirement of lesionability is partly inspired by the observation that aphasics (people with damaged linguistic abilities) are still capable of producing or understanding

---

(7) I note here that such a revision of the system is a very easy one to "implement"; it merely means that the expert "dictionary" is extended so that it contains the result of what the segmentation process does now.

language to various degrees.

The type of aphasia that has received most of the attention in this context is Broca's aphasia. Traditionally, it has been characterized as a severe disturbance of speech production with relatively well retained comprehension. Speech is effortful, hesitant and scant (nonfluent) with phonemic distortions and articulatory difficulties. Of special interest to us is its agrammatic quality: Broca's aphasics tend to omit function words and certain bound morphemes, whereas their production of content words (uninflected nouns, verbs, and adjectives) is relatively well preserved. Research by Caramazza & Zurif (1976) has further demonstrated that the comprehension of Broca's aphasics is asyntactic (i.e. when sentence comprehension required syntactic analysis with correct interpretation of its function words, performance of Broca's was very poor). Thus, Broca's aphasics demonstrate both agrammatic production and asyntactic comprehension. Caramazza et al. (1981) present further evidence hereof, and interpret it as support for the hypothesis that the syndrome of Broca's aphasia results from an impairment to the syntactic component of the language processing system.

In Bradley, Garrett & Zurif (1980) (further abbreviated to BGZ) this syntactic processing component is characterized more precisely. As discussed in 5.2.5, Bradley found a frequency effect in the recognition of content words by normals, but not in the recognition of function words, from which she concluded that there are separate access mechanisms for both classes of words (a frequency-sensitive route for content words and a non-frequency-sensitive one for function words). In contrast to normals, however, Broca's aphasics recognize function *and* content words in a way that shows frequency effects for both classes (see BGZ). This leads BGZ to the conclusion that function words are doubly represented and accessed (once via their special access route, used by normals to find the structure-building aspects of these words, and once via the frequency-sensitive one that also "contacts" the content words); in Broca's aphasics the former route would be disrupted but the latter intact, allowing for recognition but not for correct interpretation of function words. This is the more precise characterization of the impairment of the syntactic processing component.

One reason why I have given so much attention to these findings is that they have formed the background of an attempt to lesion Marcus' Parsifal parser (Marcus 1982) discussed below. Another is that there is serious doubt about the correctness of the conclusions drawn by BGZ, since the research mentioned in 5.2.5 showed that there is *no* difference between function and content words in initial access, which -- together with results from related research -- led to the general view of human information accessing as one uniform, automatic, exhaustive and independent mechanism. In short, there are no different access routes for specific words. This also undermines BGZ's componential view of language processing in which the syntactic component would contain a specific mechanism for retrieving function words for their structure-building relevance (moreover, both BGZ and Caramazza et al. 1981 are very vague about the precise content of this processing component).

Before I discuss lesioning WEP, a number of caveats with such an enterprise are in order:

1) As pointed out in Arbib et al. (1982, 171): however fascinating language disorders in aphasic patients are, it is not obvious to what extent research findings can be extrapolated to the linguistic behavior of normals. The brain may have reorganized itself after injury, or aphasics may use more "metalinguistic" means (cp. the metamode of processing competence) to understand or speak rather than the on-line processes of normals (witness e.g. the slowness in speech performance of Broca's). As a consequence, the fact that a computational model can be used to account for abnormal performance does not justify its adoption as a theory of normal function.

2) Related to 1): since lesioning is an attempt at mimicking behavior of *brain* damaged language users, one cannot avoid the issue of the neural plausibility of one's model, even if one assumes a weak form of equivalence between computer and human processing. The question then is whether there is a relationship (and if so, how far it goes) between the lesion inflicted upon the computational model and neural models of language processing in normals and aphasics (see 5.3.2). Otherwise, lesioning could be seen as a gratuitous way to show the "correctness" of a computational model.

3) Compared to psycholinguistic research with normals, neurolinguistic research with aphasics has additional difficulties: the number of patients tested is small, the syndromes may vary widely from patient to patient, and --

especially for the comprehension modality we are interested in here -- testing on-line performance is very hard.

Nevertheless, as Marcus (1982) points out, when a working computational model is lesioned in some way, it may exhibit interesting behavior that implies predictions to be put to the test of experimental confirmation or refutation (implying correctness of the model, and incorrectness plus need for revision respectively).

As a background for the lesioning of WEP, two of the ideas discussed so far are worth repeating:

1) There is a single access mechanism for all words, intact in both normals and aphasics (cp. their word recognizing ability mentioned above); whatever goes wrong is a matter of postaccess processes. For WEP, this means that (part of) the processes associated with words are erased or disrupted.

2) There are no separate syntactic or semantic components applied serially to the input; use of syntactic and semantic information happens in a way idiosyncratic to the words by interaction with other words in the context or concepts in short-term memory (see BGZ or Caramazza et al. (1981) for a discussion of the intactness of short-term processing memory in Broca's aphasics) and long-term memory (the fully specified semantic network, see also below).

Assuming then, that Broca's aphasics are incapable of using the processing information attached to function words and bound morphemes, lesioning of WEP is straightforward. For the function words implemented (i.e. articles and prepositions), a crude way of lesioning would be to simply erase the complete process associated with them. WEP still recognizes the word then, but no action is taken and the overall process is handed to the next input word. Before looking at what happens exactly in such a case, there is an interesting prediction to be made from a more "refined" way of lesioning articles and prepositions. In WEP, their "syntactic" function dominates: the processes for both word classes start by awaiting a concept of type entity. Now, if we assume that only this syntactic function is lesioned, it follows that whatever semantic actions would have been taken upon arrival of the concept (e.g. a refinement of a concept as definite for *the* or an attempt at interpreting the prepositional phrase for *in*) are also disrupted, exactly because the await mechanism does not work. In other words: lesioning the syntactic process of a function word leads automatically to the disruption of the ability to semantically interpret it. If aphasics are capable of this semantic interpretation, WEP incorrectly models the processes associated with the function words implemented; if they are not, either of the two ways of lesioning (crude or refined) may be correct. Though this last statement in itself is not a very strong one ("anything could have happened"), taken together with the assumed intactness of content words it has implications for a non-componential interactive view of language processing triggered by the words themselves. In content words, semantic processing (i.e. the creation and refinement of concepts) dominates, but especially the verbs are modeled as having implications for syntax as well through their dynamic caseframes (see 4.3.3.4): semantically driven search processes first try to bind concepts in active memory into their caseframe; if this binding fails, the verb waits for those concepts to arrive. Elementary sentence syntax falls out of this process: roughly, if binding succeeds, we have an NP-V-... declarative sentence; if it fails, we have a V-... imperative or question. Thus, we have the opposite of the function words: if semantics is intact, syntax should be no problem since it falls out of semantics. Here again, if aphasics *do* have trouble with elementary syntax, WEP is wrong; if they don't, modeling may be correct, with the implication that syntax and semantics are not to be found in separate components but are associated with the words themselves and interact differently depending on the (type of) word.

Before I discuss the effect of function word disruption, a word about the bound morphemes. Assuming that the segmentation process (see 5.2.6) is intact -- just like the access process -- a similar treatment as that of the function words suggests itself. However, it is harder to make predictions here: the correct interpretation of inflected verbs or nouns happens through signal passing during lexical interaction, i.e. the bound morphemes (*-s*, *-en*, *-ing*, etc.) probe the incoming signal (coming from the noun or verb they are attached to) and take semantic disambiguation actions accordingly (e.g. for *-s*: if the signal is *entity-construction* the entity concept (created by a noun) is refined as plural, if it is *action-construction* the action (created by a verb) is refined as a 3rd person singular). Since it is hard to selectively lesion the signal-passing mechanism (and, if we lesion it completely, it is unclear whether any comprehension is still possible at all), we seem forced to accept the crude lesioning solution: the complete process associated with bound morphemes is disrupted and beside recognition no comprehension of these morphemes whatsoever would have to be the outcome.

As to overall sentence processing, the following observations and predictions can be made: articles and prepositions take care of "low level syntax" (i.e. indicating boundaries between constituents or "catching" concepts to refine them locally). If they do not work, a number of concept chunks will be reported to memory as a result of the comprehension process. For instance, "the man eats a peach in the morning" will (informally represented) on the level of unrelated constituents lead to "man: human-adult-male", "eat: ingest food", "peach: type of fruit", "morning: first part of the day", without refinements of the concepts as defined/undefined, or without their interpretation within prepositional phrases. As to the bound morphemes: many have only local morphosemantic functions, but the *-en* morpheme also takes care of the "high level" syntactic phenomenon of passive sentences (see 4.3.1 and 4.3.3.5). Thus, with this morpheme disrupted, WEP correctly predicts the Broca's aphasic's inability to interpret reversible passives like "The girl was chased by the boy" (reversible meaning that if we only consider "girl", "boy" and "chase" either can be chasing the other) as reported by Caramazza et al. (1981). (For non-reversible passives like "The bone was chewed by the dog" the WEP semantic refinements of "dog", "bone" and "chew" would allow correct interpretation without necessity of syntactic analysis, as also observed by Caramazza et al.) So much for function words and bound morphemes. As mentioned above, the high level syntactic phenomenon of correct interpretation of constituent order in a semantic caseframe is taken care of in WEP as a side-effect of dynamic caseframes encoded in the verb. Assuming that short-term processing memory is intact, WEP would be able to correctly interpret "man eat peach morning" with "man" as agent and "peach" as object. An interesting effect of the non-interpretation of "in the" to the left of "morning" is that an attempt is made to interpret "peach morning" as a complex noun phrase, with semantics excluding this possibility; this leaves "morning" as a dangling uninterpreted noun, as one would expect. (Thus, interesting unexpected process interactions occur when the model is lesioned.) The prediction from the intactness of the verb processes is that aphasics should be able to understand simple sentences of the NP-V-NP kind. However, sentences with inversion, such as "In the morning the man eats a peach" should lead to more problems, since "Morning man eat peach" is what is left of the sentence, with some extra difficulty of finding the agent due to interference of "morning". Yet, even here correct understanding (with again a dangling "morning") is predicted. To my knowledge, no research has been conducted with simple sentences, so that the correctness of these predictions remains to be seen. We also intend to look at what happens with center-embedded sentences in WEP: Caramazza & Zurif (1976) found that aphasics had difficulty interpreting "The boy that the girl is chasing is tall" correctly (who is chasing whom?). As the system works now, "boy girl is chasing is tall" will probably be intepreted correctly ("girl" as the last reported concept would be found as the subject of "is chasing" and "boy" would then be interpreted as the subject of "is tall"), so that lesioning the function words and bound morphemes seems to leave WEP too good a patient.

A question that remains to be answered is why function words and bound morphemes are disrupted, whereas content words are not. A different access mechanism is excluded as a possibility, but I suggest a different hypothesis, i.e. a disruption in the internal organization of the lexicon. If we assume that all words are somehow related in a huge network, the following tentative view suggests itself. The "referring" content words (adjectives, nouns, verbs) are very tightly woven together through their numerous associative semantic links, which make up part of their content. Thus, there will always be some path to reach the processual information associated with these words. The "non-referring" function words and bound morphemes, on the contrary, seem more marginal to this tightly woven net, with hardly any semantic links allowing reachability of their processual aspects. Now, if we view lesioning the internal lexicon as destroying the links that keep the net together (with the link between a word and its (processual) content as one of the weaker ones in the net), it is the function words and the bound morphemes that will have no alternative paths left to reach their content, whereas the content word subnet is strong enough to have alternative paths left after disruption for retrieval of information. Even though this hypothesis may be a little vague and hard to test, I hope it will inspire research into the exact structure of the internal lexicon.

To conclude this subsection about model lesioning, I will take a brief critical look at Marcus' attempt at lesioning his Parsifal parser (Marcus 1982). Recall that Parsifal is a rule-based syntactic analysis program of the autonomous component model variety (i.e. the output of the syntactic process is the input to a semantic component) built in the first place to show that parsing can be done deterministically (see 2.3.4). In lesioning his model, Marcus mainly looks at how blocking the syntactic interpretation of closed class words (this blocking is called *Hypothetical Deficit I* (HD-I)) leads to fragmentation of the input in unrelated constituents (passed on to the semantic processor, together with the unanalyzed function words). A first critical remark concerns the necessity of modification of Parsifal before it can be lesioned. Even if we accept the fact of this modification (Parsifal was not built as a full comprehension model), the way this is done shows the difficulty of a rule-based model to deal with lexical idiosyncrasy: Marcus has to introduce a separate rule for each closed class word (as well as for a number of other lexical phenomena), which leaves him with some very divergent types of "rules" (very general ones for syntax, and very specific ones for idiosyncratic lexical phenomena). A system like WEP, which takes idiosyncrasy as its point of departure, has no problems with specific words, and needs no extensions to deal with them. An extra problem for Marcus' lesioned Parsifal is formed by morphology, since Parsifal does not perform morphological analysis of its input. The ad hoc solution Marcus suggests (morphological analysis of verbs inserts morphemes for tense and aspect into the buffer of the parser) leads to a further fragmentation of the input by the parser so that "no full sentence could be fully analysed by an HD-I parser" (1982, 131). As we saw above, bound morphemes cause no such problems in WEP, and leave integrated comprehension of the content words intact.

5.3.2. Parallelism and the mapping problem

In the course of this book (and especially in this chapter) I have taken the design/intentional research stance (see 1.3.3.2) combined with a weak (i.e. functional) equivalence view of human and computer functioning. In the context of neurolinguistic research and neural models of language processing I want to point out a few things that to me further justify this position.

Several researchers have stressed that a serious problem for the neural plausibility of computational models is the completely different nature of computer and brain functioning. Referring to Crick, Cottrell & Small (1983, 91) summarize the main differences in the following Figure:

	computer	brain
speed	fast	slow
order	serial	parallel
component reliability	reliable	unreliable
faults	fatal	no degradation
signals	precise, symbolic	imprecise, terse

Figure II. Computers versus brains.

In short, the brain exhibits a completely different structure than a conventional computer: it consists of comparatively simple and slow neural computing elements with complex parallel connections (cp. also Feldman & Ballard 1982). It has repeatedly been said in the course of this book that to me these fundamental differences are reason enough to retreat to a weak AI plus weak equivalence position, and not to take the physical stance in research into natural language understanding (especially if one is interested in computer models). I point out that this does not mean it is impossible to relate research into aphasia (a matter of brain damage) to computer (mal)functioning, as was done in 5.3.1. Issues like the disruption of function words, or articulatory difficulties are instances of lesioning at the functional level, and nothing is said about the exact nature of the damage to the brain in terms of destroyed neurons or neuron functions (there are at the most at the macrolevel certain brain regions that can be pointed out as damaged).

Still, instead of retreating to a skeptical position, some researchers have taken a more audacious step: if the computer is so different from a brain, then let us make it look and act like a brain. As mentioned in 1.3.2, these "second generation cognitive scientists" have radically abandoned the computer metaphor of human cognitive functioning and are trying out the human-cognition metaphor of computer functioning. This has led to the connectionist approach discussed in 2.3.2. An assumption of this approach is that in order to explain the wealth of psychological data on (low-level) language processing, the correspondence between computational models and processing strategies of people has to be considered on the level of the mechanisms (i.e. brain functioning) involved in carrying out cognitive functions, and not at the (higher) functional level (i.e. a correspondence between the functions the program performs and those the human being performs) (see Cottrell & Small 1983). Hence they seem to collapse the design/functional stance and the physical stance in research. However, I do not believe that there is an obvious way 1) in which such complex functions as human language understanding emerge from the underlying structure of the brain (cp. Marin 1982) and 2) in which this structure can be simulated on existing computers. Moreover, our knowledge of both language understanding and brain functioning -- however rapidly growing -- is too incomplete for an attempt at mapping the two onto each other. Though functioning of the brain will some day form the ultimate touchstone for whatever theories of cognitive processing we have, I do not consider it right to force a direct mapping between the two at this stage of scientific research (cp. Kolers & Smythe 1984 for a sharp attack of the sloppy mixing of levels of description/explanation in some cognitive science approaches). Note that this issue is just one instance of a serious problem the new field of cognitive science has to face, i.e. how do models developed in the subsciences map onto one another? More concretely, how do 1) abstract formal models (as developed in generative linguistics, for instance), 2) psychological process models of high-level cognitive processes, and 3) neural models of brain functioning fit in with one another? (In a broad perspective, the long standing discussion about the competence-performance distinction in (psycho)linguistics is an example of the difficulty to map 1) on 2); discussions about "mind" versus "brain" reflect mapping problems between 2) and 3).) And where does the computer fit in? I only point out the problems here and offer no ready solutions for them. Suffice it to say that to me they inspire carefulness about especially very direct mapping attempts; I also believe that the vitality of the new multidisciplinary field will partly depend on the success (or failure) of finding satisfactory ways to map the contents of different scientific disciplines onto one another.

How does all this relate to WEP now? One aspect of the system I have not spent too much time on is that WEP was designed to be a distributed system not only by the use of coroutines to model word processes, but also by having the entries of the expert processes (by themselves subprocesses) execute in parallel. Implementation of this proved very hard because of the complexities of the symbol-passing interactions and the sequential ordering of discrimination networks. Partly because of these problems and partly because of the wish to model brain functioning (with its massive parallelism), the original WEP approach was abandoned in favor of a connectionist one. I refer the reader to Cottrell & Small (1983) and Cottrell (1985) for a full account of this completely different research that grew out of WEP. I do not go into it here because of my disagreement with the fundamental premise of a direct mapping from brain to cognitive functioning and also because the research is still in an early stage. Cottrell (1985) proposes a *design* of a connectionist analyzer without a working implementation. Also, whereas his model seems capable of disambiguating words correctly and in detailed accordance with psycholinguistic research into lexical access (see 5.2.2), he points out (p. 122) that matters of syntax were not the main concern and as such are hardly dealt with (cp. the early WEP model). In contrast, the revision of WEP for Dutch seems to have shown that the system can handle syntactic matters in an interesting way; whether the connectionist WEP offspring can do the same remains to be seen.

Having continued research in the "old" WEP style, then, implied relaxation of the parallelism claims, and further attempts to map the way it works to psycho- and neurolinguistic research on the functional level (and not on the brain level). It must also not be forgotten that not everything in language processing happens in parallel: for one thing, the input arrives in a sequential fashion and the processing mechanisms have to deal with its time-bound arrival; for another, as suggested by WEP, processes of different kinds (morphological, syntactic, semantic,...) may prove to be so closely intertwined and dependent on each other in unpredictable ways, that it is unclear how a (uniformly) parallel processing mechanism could deal with them. A combination of sequential and parallel processing is probably the ideal way to model natural language understanding. In the reflections and suggestions for further research which now follow, I will briefly come back to parallelism in WEP.

CONCLUSION AND FURTHER RESEARCH

I have started this book by situating its research in a cognitive science perspective, stressing the need of a multidisciplinary approach to the complex problem of natural language understanding (chapter 1). As to the ingredients of this multidisciplinary approach, I have tried to point out the importance of cognitive psychology and weak AI to cognitive-scientific linguistic research. Concretely, the main linguistic paradigm to date (generative linguistics) has been criticized as hardly integratable with the other disciplines when natural language understanding is the object of study (chapter 2), and a different approach has been developed (chapter 3). For its linguistic notions it stresses the importance of the lexicon as the natural basis of morphological, syntactic, semantic and pragmatic knowledge. The crucial notion that turns this "lexicon" into a highly dynamic entity is the process notion, leading to a view of language as an intrinsically time-bound phenomenon rather than a timeless, static object. Process linguistics can thus be seen as a dynamic reinterpretation of existing linguistic notions of semantic fields (hardly touched upon here, though), contextual distribution, syncategorematicity, interdependency and case assignment. It is especially for its process notion that the approach draws ideas from AI (computational processes) and psycho/neurolinguistics (cognitive processes, interactive processing models). These disciplines are the focus of attention in chapters 4 and 5 respectively. Chapter 4 shows that process linguistics can stand the test of its computational realizability through the successful revision and application of an existing AI program (WEP) to Dutch; chapter 5 shows that this computer model can stand the test of its psychological plausibility through a confrontation with psycho- and neurolinguistic research. In short, the whole of the approach can claim the label of cognitive-scientific linguistic research.

Yet, nothing ends with this conclusion. Once the viability of the approach is shown, it has to be filled in further, both in theory and in practice.

As far as the theory is concerned, it must be stressed that the metascientific cognitive science framework is not (yet) an established paradigm; we are only starting to explore its foundations and assumptions (cp. Pylyshyn 1984, Gardner 1985). Problems to be further addressed in this context are the feasibility of multidisciplinarity involving paradigms working along different lines and developing different models involving the same phenomenon (cp. the conflicting research poles (1.3.3.5) and the mapping problem (5.3.2)), a badly needed theoretical approach to the crucial but elusive process notion (a process theory encompassing both human and computer processes, which could maybe draw on insights from cybernetics and quantum mechanics (cp. the motto of the book)). Surely not an easy job for philosophers of science.

When we step down one level, there is the theory of process linguistics. Issues to be addressed here are:

- The search for more processing universals (like the expectation-feedback cycle or the exhaustive memory access mechanism) and research into the way they can be linked to structural aspects of language (i.e. the way language structure can be shown to fall out of the processing universals).

- A processual account of linguistic facts. Concretely, in the context of the interactive models of language processing, research into and description of *how*, *where*, and *when* knowledge from different sources interacts during comprehension is badly needed. Instead of ignoring the time dimension of language and describing it at the traditional levels (even carrying this description through to the parsing problem), with artificial ambiguity and problem postponement as negative effects (1), linguists should be more concerned with the knowledge (of all kinds) that is needed during time-bound comprehension with complete

(1) With artificial ambiguity and problem postponement I mean e.g. the ambiguity of the sentence "I saw a man on the hill with a telescope"; if we only look at it from the syntactic point of view (within an autonomous component model, leaving semantic/pragmatic analysis "for later"), it is multiply ambiguous merely *because* we look at it from that limited point of view. Methodologically it may seem interesting to look at sentences from this limited perspective, but it may be wrong from the perspective of online comprehension in reality, where the *resolution* of ambiguity is important, and not its creation.

understanding as a result. Questions to be addressed in this context are: how is meaning built up dynamically during comprehension (not: how can I describe the meaning of complete(d) sentences postfactum)? what are the systematicities of the interactions of knowledge of different kinds? what is the relative contribution of the different knowledge sources?

- The closer study of the neglected half of the interaction (the vertical intralexical interaction). It implies the need to reconsider semantic field theory, as well as a study of how (static) models in linguistics can be combined with the dynamic models developed in psychology and AI (semantic networks, spreading activation).

- Beside these three crucial issues, a minor one is the further development of a view on language learning (and language production) that can complete the process-linguistic account of comprehension.

Stressing the need of a processual account of language is not merely a theoretical issue, but of course also one of practice, i.e. of description. Description of the distribution of linguistic elements (not merely syntactic), of their combinability (syncategorematicity) is indispensable as the static basis for the study of (dynamic) lexical-contextual interaction. Note that the computer can be of great help here, in that corpus research (not very popular to linguists blinded by the beauty of formal theories) can be done very efficiently with computational tools (see below).

Mentioning the computer brings me to the issues in the further development of the computer model, the Word Expert Parser. All the issues pertain to aspects of one general question, viz. "How far can lexical-contextual interaction go?".

- Through corpus research it should be possible to find out how much left and right context is needed in order for specific words to be understandable (disambiguatable); especially the words considered to be global sentence modifiers (such as negating elements) are worth looking at: can WEP interaction correctly interpret words that seem to need the complete sentence they occur in as context (i.e. by making use of its wait-and-see strategy)? Useful tools for this kind of research are the QUERY and

PARSPAT programs developed at the University of Amsterdam for linguistic research with precoded corpora. A simple test in this context could be to have these programs give lists of specific words (from sentences in the corpora) with one, two, three, etc. words as left and/or right context. The degree of understandability depending on the broadness of the context could be an indication of how to build the word-expert processes.

- Of course, the implementation of more words is a research issue that will help the further development and refinement of the representation language and that will further show the strong and weak sides of the system. Experts under development are *die* ("that/who"; relative and demonstrative pronoun), *het* ("the" (article) or "it" (pronoun)), *niet* ("not"), and *en* ("and"). The addition of metamode entries to the existing experts is also a development that could enhance the robustness and scope of the system (allowing the experts to back up when the choice of one of their paths proved wrong, or making them learn from half-fulfilled expectations). An interesting extension could also be to make the experts sensitive to the relative frequency of the different word meanings they deal with. Branching to entry points could be made to depend on weights given to the subprocesses involved (here too, corpus research into usage frequencies could be very helpful).

- Two important and related issues (also mentioned by Small (1980, 196-202)) are the avoidance of duplication of information and the necessity of a central semantic network in memory. As to the duplication of information, it comes in two forms. The first is that every expert has its own process, with no sharing of identical subparts across experts. Once enough experts of the same word class (e.g. verbs) are tested, it will become clear what subprocesses can be considered as general enough to be entered into a "library" or collection of entry points easily integratable in new experts (and possibly shared during processing by the experts involved). The second type of duplication is strongly related to the absence of a central memory network. As Small points out:

> "When an expert asks the multiple choice VIEW question, for example, the possible answers to the question are listed explicitly as part of the question node. Such duplication of pieces of the memory within experts is not a satisfactory implementation. The central knowledge base should contain all conceptual information and the known relationships among its parts. Word experts should access this central store of knowledge, rather than duplicating needed substructures from within it" (1980, 199-200).

Plugging such a memory network into WEP could also eliminate the need of user interaction to answer the VIEW and other memory binding questions.

- Finally, two issues pertaining to the computational aspects of the system are worth mentioning. WEP now runs in Franz Lisp on a VAX750 at the computer science department of Leuven; an attempt will be made to develop a Prolog version of the system. At the same time, research will be conducted into the possibilities and limitations of coroutines and other modest forms of (simulated) parallelism. Parallelism was one of the important design issues in WEP (the concurrent executability of entry points within one expert coroutine and/or within several expert coroutines). As discussed, it proved a hard problem to deal with due a.o. to the complexity of expert interactions (symbol-passing) and the sequential ordering of discrimination networks. A closer examination of the experts implemented may show that parts of their process (possibly across the entries as defined now) are independent of one another and might then be executable in parallel.

To conclude the list of issues for further research, the psycho- and neurolinguistic aspects of the approach must not be forgotten either. Though further research in this area will not be my major concern in the near future, WEP has shown that more experiments are needed to test online morphological processing, eye-movement and -fixations during reading (what happens to the function words?), and the status of specific subclasses of function and/or content words. The predictions made about Broca's aphasics also deserve testing to see how the behavior of a lesioned WEP version relates to aphasic verbal behavior.

In short, it is my hope that the different facets of process linguistics will inspire a wide range of cognitive science research into natural language understanding. Maybe process linguistics can then help unveil the non-apparent complexity of verbal behavior, so easily performed by the human being yet so poorly understood by scientists and so poorly imitated by computers.

APPENDIX 1: WORD-EXPERT REPRESENTATION LANGUAGE

Note: this appendix is almost identical to Small's Appendix A (1980, 214-217); hence, the credits for writing down the whole representation language syntax in Backus-Naur form go to Small.
The differences are a matter of

1) a different Lisp version the original experts and later experts (including the Dutch ones) were written in (Maryland versus Franz Lisp)
2) omitted and added actions/questions for the Dutch experts
3) some forgotten non-terminal expansions (the <concept actions>).

Nonterminals are between "<" and ">", terminals between double quotes. The nonterminals <string> and <integer> are the classical alphanumerical and numerical sequences; <string> is used very generally, and can be any kind of constant (a signal constant, a word constant, a concept constant, a feature constant, etc.) depending on the context of usage; see the examples in appendix 2)

```
<entry variable>        ::= "e"<integer>
<node variable>         ::= "n"<integer>
<signal variable>       ::= "s"<integer>
<concept variable>      ::= "c"<integer>
<word variable>         ::= "w"<integer>
<expert variable>       ::= "x"<integer>

<word-expert>           ::= "[" "word-expert"  <string> <entry points> "]"
                            (the string is the expert's word)

<entry-points>          ::= <entry-point>  |  <entry-point> <entry-points>

<entry-point>           ::= "[" <entry variabele> <nodes> "]"

<nodes>                 ::= <node>  |  <node> <nodes>

<node>                  ::= "(" <node variable> ":" <node type> ")"

<node type>             ::= <question node>  |  <action node>

<question node>         ::= "q" <question>
```

```
<question>                 ::= <signal question> | <idiom question> | <literal question> | <view question>
                               | <feature question> | <bound question> | <partofword question>

<signal question>          ::= "signal" <signal variable> <string-node pairs>

<idiom question>           ::= "idiom" <concept variable> <stringlist-node pairs>

<literal question>         ::= "literal" <word variable> <string-node pairs>

<view question>            ::= "view" <concept variable> <string-node pairs>

<feature question>         ::= "qfeature" <concept variable> <string-node pairs>

<string-node pairs>        ::= <string-node pair> | <string-node pair> <string-node pairs>

<stringlist-node pairs>    ::= <stringlist-node pair> | <stringlist-node pair> <stringlist-node pairs>

<string-node pair>         ::= "[" <string> <node variable> "]" | <else pair>

<else pair>                ::= "[" "*" <node variable> "]"

<stringlist-node pair>     ::= "[" <string list> <node variable> "]"

<bound question>           ::= "bound" <concept variable>
                               "[" "bound"   <node variable> "]"
                               "[" "unbound" <node variable> "]"

<qpartofword question>     ::= "qpartofword" <expert variable>
                               "[" "yes" <node variable> "]"
                               "[" "no"  <node variable> "]"

<action node>              ::= "a" <actions>

<actions>                  ::= <action> | <action> <actions>

<action>                   ::= <control action> | <structure building action> | <binding action>

<control action>           ::= <internal control action> | <EFC action> | <lookahead action>

<internal control action>  ::= <branch to node> | <branch to entry> | <pause and branch>

<branch to node>           ::= "(" "next" <node variable> ")"

<branch to entry>          ::= "(" "continue" <entry variable> ")"
```

```
<pause and branch>       ::= "(" "pause" <entry variable> ")"

<EFC action>             ::= <await action> | <send action>

<await action>           ::= <await concept> | <await signal> | <await word>

<await concept>          ::= "(" "await" "concept" <concept type>
                                                   <concept specs>
                                                   <bindconcept>
                                                   <bindsignal>
                                                   <bindsender>
                                                   <report constraint>
                                                   <timeout condition>
                                                   <branch to entry>
                                                   <timeout continue> ")"

<await signal>           ::= "(" "await" "signal"  <signal specs>
                                                   <bindsignal>
                                                   <bindconcept>
                                                   <bindsender>
                                                   <report constraint>
                                                   <timeout condition>
                                                   <branch to entry>
                                                   <timeout continue> ")"

<await word>             ::= "("  "await"  "word"  <string>
                                                   <bindsender>
                                                   <timeout condition>
                                                   <branch to entry>
                                                   <timeout continue> ")"

<concept specs>          ::= see below

<signal specs>           ::= "(" "filter" <string list> ")"

<bindconcept>            ::= "(" "bindconcept" <concept variable> ")"

<bindsignal>             ::= "(" "bindsignal" <signal variable> ")"

<bindsender>             ::= "(" "bindsender" <expert variable> ")"

<report constraint>      ::= "(" "report" <report destination> ")"

<report destination>     ::= "here" | "normal"
```

```
<timeout condition>          ::= "(" "wait" <wait duration> ")"

<wait duration>              ::= "word" <integer> ¦ "group" <integer> ¦ "break" <integer>

<timeout continue>           ::= "(" "else" <entry variable> ")"

<send action>                ::= <report action> ¦ <signal action>

<report action>              ::= "(" "report" <concept variable> <report addenda> ")"

<report addenda>             ::= <signal accompaniment> <recipient constraint>

<signal accompaniment>       ::= <signal action>

<signal action>              ::= "(" "signal" <string> <signal addenda> ")"

<recipient constraint>       ::= "(" "to" <expert variable> ")"

<signal addenda>             ::= <concept accompaniment> <recipient constraint>

<concept accompaniment>      ::= "(" "concept" <concept variable> ")"

<lookahead action>           ::= <peek action> ¦ <read action>

<peek action>                ::= "(" "peekw" <word variable> ")"

<read action>                ::= "(" "readw" <expert variable> ")"

<structure building action> ::= <grouping action> ¦ <concept action>

<grouping action>            ::= <open group> ¦ <participate in group> ¦ <close group>
                                 ¦ <addword to concept> ¦ <sentence break>

<open group>                 ::= "(" "openg" <signal constant> ")"

<participate in group>       ::= "(" "declareg" ")"

<close group>                ::= "(" "closeg" ")"

<addword to concept>         ::= "(" "addlex" <concept variable> <word variable> ")"

<sentence break>             ::= "(" "breakg" ")"

<concept action>             ::= <create concept> ¦ <build concept> ¦ <refine concept>
                                 ¦ <specify concept role> ¦ <specify concept aspect>
```

```
                              ¦ <store concept> ¦ <link-to-group concept>

<create concept>            ::= "(" "createc" <concept variable> <concept type> ")"

<build concept>             ::= "(" "buildc" <concept variable> <concept type> <concept specs> ")"

<concept specs>             ::= <concept spec> ¦ <concept spec> <concept specs>

<concept spec>              ::= <filter spec> ¦ <value spec> ¦ <choice spec> ¦ <role spec>
                              ¦ <aspects spec> ¦ <lexical spec>

<filter spec>               ::= "(" "filter" <concept variable> ")"

<value spec>                ::= "(" "value" <string> ")"

<choice spec>               ::= <oneof spec> ¦ <noneof spec>

<oneof spec>                ::= "(" "oneof" <string list> ")"

<noneof spec>               ::= "(" "noneof" <string list> ")"

<string list>               ::= <string> ¦ <string> <string list>

<aspects spec>              ::= "(" "aspects" <aspect list> ")"

<aspect list>               ::= <aspect> ¦ <aspect> <aspect list>

<aspect>                    ::= "(" <string> <concept variable> ")"

<lexical spec>              ::= "(" "lexical" <string list> ")"

<refine concept>            ::= "(" "refinec" <concept variable> <string> ")"

<specify concept role>      ::= "(" "rolec" <concept variable> <string> ")"

<specify concept aspects>   ::= "(" "aspectc" <concept variable> <aspects> ")"

<store concept>             ::= "(" "storec" <concept variable> ")"

<link-to-group concept>     ::= "(" "link" <concept variable> ")"

<binding action>            ::= "(" "bindc" <concept variable> <binding region> <concept variable> ")"
                                              [local name]                      [filter]

<binding region>            ::= "memory" <memory region> ¦ "discourse" <discourse mechanism>
```

```
                         ¦ "immediate" ¦ "real-world" <pragmatic spec>

<memory region>          ::= "active" ¦ "expect"

<discourse mechanism>    ::= "focus" ¦ "expect"

<pragmatic spec>         ::= "plausible" ¦ "belief"
```

APPENDIX 2: DUTCH WORD-EXPERTS

The word-experts are presented in the following order: first, the bound morphemes are listed (*-en, -s, -t, ge-*), then the punctuation marks (#punt# (period), #vraagteken# (question mark), #uitroepteken# (exclamation mark)), and finally the free morphemes (words) are listed in alphabetical order (from *appel* (apple) to *zomer* (summer)).

```
(word-expert -en

(e0  (n0 :q signal s0
              [action-construction n1]
              [entity-construction n2])
     (n1 :a (bindc c1 immediate c0)
            (continue e2))
     (n2 :a (bindc c1 immediate c0)
            (continue e1))
)
(e1  (n0 :a (declareg)
            (refinec c1 =c#1#meerdere)
            (report c1)
            (next n1))
     (n1 :q feature c1
              [c#onbepaald n2])
     (n2 :a (closeg complete-entity))
)
(e2  (n0 :a (declareg)
            (refinec c1 =c#1#actie-meervoud)
            (closeg complete-action)
            (link c1)
            (report c1))
)
)


(word-expert -s

(e0  (n0 :q signal s0
              [entity-construction n1]
              [* n2])
     (n1 :a (bindc c1 immediate c0)
```

```
        (continue e1))
    (n2 :a)
]
[e1  (n0 :a (declareg)
        (refinec c1 =c#1#meerdere)
        (link c1)
        (report c1)
        (next n1))
    (n1 :q feature c1
          [c#onbepaald n2])
    (n2 :a (closeg complete-entity))
]
]

[word-expert -t

[e0  (n0 :q signal s0
          [action-construction n1]
          [* n2])
    (n1 :a (bindc c1 immediate c0)
        (continue e1))
    (n2 :a)
]
[e1  (n0 :a (declareg)
        (refinec c1 =c#1#actie-enkelvoud)
        (closeg complete-action)
        (report c1))
]
]

[word-expert ge-

[e0  (n0 :q signal s0
          [entity-construction n1]
          [action-construction n2])
    (n1 :a (declareg))
    (n2 :a (declareg)
        (continue e1))
]
[e1  (n0 :a (buildc c2 action
                (allof =c#worden))
        (bindc c3 memory active c2)
        (next n1))
    (n1 :q bound c3
          [bound n2]
```

```
              [unbound n3])
      (n2 :a (readw w1)
             (signal passive
                     (to w1)
                     (concept c3)))
      (n3 :a (buildc c2 action
                     (value =c#hebben))
             (bindc c3 memory active c2)
             (next n4))
      (n4 :q bound c3
              [bound n5]
              [unbound n6])
      (n5 :a (readw w1)
             (signal vtt (to w1))
      (n6 :a)
]
]

[word-expert #punt#

[e0   (n0 :a (breakg)
             (buildc c0 action)
             (bindc c1 memory active c0)
             (next n1))
      (n1 :q bound c1
              [bound n2]
              [unbound n3])
      (n2 :a)
      (n3 :a (createc c0 zinstype)
             (refinec c0 =c#mededeling)
             (refinec c0 =c#ellipsis)
             (report c0))
]
]

[word-expert #vraagteken#

[e0   (n0 :a (breakg)
             (buildc c0 action)
             (bindc c1 memory active c0]
             (next n1))
      (n1 :q bound c1
              [bound n2]
              [unbound n3])
```

```
      (n2 :a)
      (n3 :a (createc c0 zinstype)
             (refinec c0 =c#vraag)
             (refinec c0 =c#ellipsis)
             (report c0))
]
]

[word-expert #uitroepteken#

[e0   (n0 :a (breakg)
             (buildc c0 action)
             (bindc c1 memory active c0)
             (next n1))
      (n1 :q bound c1
                [bound n2]
                [unbound n3])
      (n2 :a)
      (n3 :a (createc c0 zinstype)
             (refinec c0 =c#bevel)
             (refinec c0 =c#ellipsis)
             (report c0))
]
]

[word-expert  appel

[e0   (n0 : a (createc c1 entity)
              (next n1))
      (n1 : q signal s0
                [entity-construction n2]
                [entity-construction? n2]
                [* n5])
      (n2 :q literal w0
                [het n3[
                [de n4]
                [*  n4])
      (n3 :a)
      (n4 :a (pause e1))
      (n5 :a (pause e4))
]
[e1   (n0 :a (declareg)
             (refinec c1 =c#vrucht)
             (refinec c1 =c#1#appel)
             (link c1)
```

```
            (next n1))
      (n1 :q partofword x0
                [y n2]
                [n n5])
      (n2 :a (readw w1)
            (signal entity-construction
                            (to w1)
                            (concept c1)))
      (n5 :a (continue e3))
]
[e3   (n0 :a (refinec c1 =c#enkelvoud)
            (report c1))
]
[e4   (n0 :a (openg entity-construction)
            (refinec c1 =c#onbepaald)
            (continue e1))
]
]

[word-expert bel

[e0   (n0 :q signal s0
                [break n1]
                [passive n2]
                [* n3])
      (n1 :a (buildc c0 zinstype
                      (oneof =c#vraag =c#bevel)
                      (noneof =c#mededeling))
            (report c0)
            (continue e1))
      (n2 :a (bindc c1 immediate c0)
            (refinec c1 =c#passief)
            (refinec c1 =c#gegeten-worden)
            (declareg)
            (link c1)
            (closeg complete-action))
      (n3 :a (createc c0 zinstype)
            (refinec c0 =c#mededeling)
            (report c0)
            (continue e1))
]
[e1   (n0 :a (openg action-construction)
            (declareg)
            (createc c1 action)
            (refinec c1 =c#bellen)
```

```
        (link c1)
        (next n1))
    (n1 :q partofword x0
           [y n2]
           [n n3])
    (n2 :a (readw w1)
        (signal action-construction
               (to w1)
               (concept c1))
        (pause e2))
    (n3 :a)
]
[e2 (n0 :a (buildc c2 entity
               (role agent)
               (oneof =c#persoon  =c#groepering))
        (bindc c3 memory active c2)
        (continue e11)
        (next n1))
    (n1 :q bound c3
           [unbound n2[
           [bound n3])
    (n2 :a (await concept entity
               (filter c2)
               (bindconcept c3)
               (wait group 2)
               (continue e5)
               (else e3)))
    (n3 :a (refinec c0 =c#meded-rechte-orde)
        (continue e4))
]
[e3 (n0 :a (buildc c4 entity
               (role object)
               (oneof =c#persoon =c#groepering))
        (bindc c5 memory active c4)
        (next n1))
    (n1 :q bound c5
           [unbound n2]
           [bound n3])
    (n2 :a (await concept entity
               (filter c4)
               (bindconcept c5)
               (wait break 1)
               (continue e7)))
    (n3 :a (buildc c8 zinstype
               (allof =c#mededeling =c#inversie))
```

```
                (bindc c9 memory active c8)
                (next n4))
          (n4 :q bound c9
                 [unbound n5]
                 [bound n6])
          (n5 :a (refinec c0 =c#bevel)
                (report c0)
                (continue e7))
          (n6 :a)
]
[e4   (n0 :a (aspectc c1 (agent c3))
             (rolec c3 agent)
             (report c1)
             (continue e3))
]
[e5    (n0 :a (buildc c8 zinstype
                         (value =c#mededeling))
              (bindc c9 memory active c8)
              (next n1))
       (n1 :q bound c9
                [bound n2]
                [unbound n3])
       (n2 :a (refinec c9 =c#meded-inversie)
              (continue e4))
       (n3 :a (refinec c0 =c#vraag)
              (report c0)
              (continue e4))
]
[e6
]
[e7   (n0 :a (aspectc c1 (object c5))
             (rolec c5 object)
             (refinec c1 =c#telefoneren-naar)
             (report c1))
]
[e11   (n0 :a (await signal
                         (filter particle)
                         (bindsender w1)
                         (report here)
                         (wait break 1)
                         (continue e12)))
]
[e12   (n0 :a (addlex c1 w1))
]
]
```

```
[word-expert  de

[e0   (n0 :a (peekw w1)
             (next n1))
      (n1 :q literal w1
                [facto n4]
                [jure  n3]
                [*     n2])
       (n2 :a (openg entity-construction)
              (declareg)
              (continue e1))
       (n3 :a)
       (n4 :a)
]
[e1   (n0 :a (await concept entity
                        (bindconcept c1)
                        (report here)
                        (wait group 1)
                        (continue e2)))
]
[e2   (n0 :a (closeg complete-entity)
             (bindc c2 memory focus c1)
             (next n1))
      (n1 :q bound c2
                [unbound n2]
                [bound n3])
      (n2 :a (refinec c1 =c#nieuw-bepaald)
             (report c1))
      (n3 :a (refinec c2 =#oud-bepaald)
             (report c2))
]
]


[word-expert door

[e0   (n0 :q partofword x0
                [y n1]
                [n n5])
      (n1 :a (declareg)
             (next n2))
      (n2 :q signal s0
                [entity-construction n3]
                [* n4])
      (n3 :a)
```

```
     (n4 :a (continue e11))
     (n5 :a (continue e1))
]
[e11 (n0 :a (readw w1)
           (signal action-construction
                              (to w1)))
]
[e1  (n0 :a (openg setting)
           (declareg)
           (createc c2 setting)
           (await concept entity
                    (bindconcept c1)
                    (report here)
                    (wait group 2)
                    (continue e3)
                    (else e5)))
]
[e3  (n0 :q view c1
             [=c#tijd  n1]
             [=c#plaats n2]
             [=c#agens n3]))
     (n1 :a (link c2)
           (aspectc c2 (oblique c1))
           (rolec c1 oblique)
           (rolec c2 bwbep-tijd-periode)
           (report c2)
           (closeg complete-setting)
           (storec c1))
     (n2 :a (link c2)
           (aspectc c2 (oblique c1))
           (rolec c1 oblique)
           (rolec c2 bwbep-doorheen-ruimte)
           (report c2)
           (closeg complete-setting)
           (storec c1))
     (n3 :a (link c2)
           (aspectc c2 (oblique c1))
           (rolec c1 oblique)
           (rolec c2 door-bep-agens)
           (report c2)
           (closeg complete-setting)
           (storec c1))

]
[e4  (n0 :a (signal particle))
```

```
]
[e5
]
]

[word-expert een

[e0  (n0 :q signal s0
          [er n3]
          [entity-construction n2]
          [ * n1])
     (n1 :a (openg entity-construction)
           (declareg)
           (continue e1))
     (n2 :a (declareg)
           (createc c1 entity)
           (refinec c1 =c#zelfst-eenheid)
           (link c1)
           (closeg complete-entity))
     (n3 :a)
]
[e1  (n0 :a (await concept entity
                    (bindconcept c2)
                    (report here)
                    (wait group 1)
                    (continue e2)))
]
[e2  (n0 :a (refinec c2 =c#onbepaald)
          (closeg complete-entity)
          (report c2))
]
]


[word-expert eet-

[e0  (n0 :q signal s0
             [break n1]
             [entity-construction n2]
             [passive n3]
             [vtt n4]
             [action-construction n5]
             [* n6])
      (n1 :a (buildc c0 zinstype
                    (oneof =c#vraag =c#bevel)
```

```
                  (noneof =c#mededeling))
            (report c0)
            (continue e1))
      (n2 :a (declareg))
      (n3 :a (bindc c1 immediate c0)
            (refinec c1 =c#passief)
            (refinec c1 =c#gegeten-worden)
            (declareg)
            (link c1)
            (closeg complete-action))
      (n4 :a (continue e14))
      (n5 :a)
      (n6 :a (createc c0 zinstype)
            (refinec c0 =c#mededeling)
            (report c0)
            (continue e1))
]
[e1   (n0 :a (openg action-construction)
            (declareg)
            (createc c1 action)
            (refinec c1 =c#zich-voeden)
            (link c1)
            (next n1))
      (n1 :q partofword x0
              [y n2]
              [n n3])
      (n2 :a (readw w1)
            (signal action-construction
                     (to w1)
                     (concept c1))
            (pause e2))
      (n3 :a)
]
[e2   (n0 :a (buildc c2 entity
                     (role agent)
                     (oneof =c#persoon =c#dier))
            (bindc c3 memory active c2)
            (continue e11)
            (next n1))
      (n1 :q bound c3
              [unbound n2]
              [bound n3])
      (n2 :a (await concept entity
                     (filter c2)
                     (bindconcept c3)
```

```
                    (wait group 2)
                    (continue e5)
                    (else e3)))
        (n3 :a (refinec c0 =c#meded-rechte-orde)
               (continue e4))
]
[e3     (n0 :a (buildc c4 entity
                          (role object)
                          (allof =c#iets-eetbaars =c#ding))
               (bindc c5 memory active c4)
               (next n1))
        (n1 :q bound c5
               [unbound n2]
               [bound n3])
        (n2 :a (await concept entity
                    (filter c4)
                    (bindconcept c5)
                    (wait break 1)
                    (continue e7)
                    (else e10)))
        (n3 :a (buildc c8 zinstype
                    (allof =c#mededeling =c#inversie))
               (bindc c9 memory active c8)
               (next n4))
        (n4 :q bound c9
               [unbound n5]
               [bound n6])
        (n5 :a (refinec c0 =c#bevel)
               (report c0)
               (continue e7))
        (n6 :a)
]
[e4   (n0 :a (aspectc c1 (agent c3))
             (rolec c3 agent)
             (report c1)
             (continue e3))
]
[e5     (n0 :a (buildc c8 zinstype
                    (value =c#mededeling))
               (bindc c9 memory active c8)
               (next n1))
        (n1 :q bound c9
               [bound n2]
               [unbound n3])
        (n2 :a (refinec c9 =c#meded-inversie)
```

```
            (continue e4))
      (n3 :a (refinec c0 =c#vraag)
             (report c0)
             (continue e4))
]
[e7  (n0 :a (aspectc c1 (object c5))
            (rolec c5 object)
            (refinec c1 =c#voedsel-innemen)
            (report c1))
]
[e8  (n0 :a (buildc c0 zinstype
                       (value =c#bevel))
            (report c0)
            (bindc c5 immediate c4)
            (continue e7))
]
[e9  (n0 :a (await concept entity
                       (filter c4)
                       (report here)
                       (wait break 1)
                       (continue e7)
                       (else e10))
            (bindc c5 immediate c4))
]
[e10  (n0 :a (refinec c1 =c#eten-impliciet-object)
             (report c1))
]
[e11  (n0 :a (await signal
                       (filter particle)
                       (bindsender w1)
                       (report here)
                       (wait break 1)
                       (continue e12)))
]
[e12  (n0 :a (refinec c1 =c#perfectief-opeten)
             (addlex c1 w1))
]
]

[word-expert geert

[e0  (n0 :a (openg entity-construction)
            (declareg)
            (continue e1))
]
```

```
[e1   (n0 :a (createc c1 entity)
             (refinec c1 =c#persoon)
             (refinec c1 =c#mannelijk)
             (refinec c1 =c#gek)
             (refinec c1 =c#braaf)
             (refinec c1 =c#ijdel)
             (link c1)
             (report c1)
             (closeg complete-entity))
]
]


[word-expert haar

[e0   (n0 :q signal s0
               [entity-construction n1]
               [* n2])
      (n1 :a (declareg)
             (createc c1 entity)
             (refinec c1 =c#1#haar)
             (refinec c1 =c#1#huidbedekking)
             (link c1)
             (report c1))
      (n2 :a (openg entity-construction)
             (declareg)
             (buildc c2 entity
                        (oneof =c#vrouwelijk-persoon =c#abstractum))
             (bindc c3 memory active c2)
             (next n3))
      (n3 :q bound c3
               [bound n4]
               [unbound n7])
      (n4 :q view c3
               [=c#vrouwelijk-persoon n5]
               [=c#abstractum n6])
      (n5 :a (await concept entity
                        (bindconcept c4)
                        (wait group 1)
                        (continue e1)))
      (n6 :a (await concept entity
                        (bindconcept c4)
                        (wait group 1)
                        (continue e2)))
      (n7 :a (continue e3))
```

```
]
[e1   (n0 :a (createc c5 relatie)
            (refinec c5 =c#1#relatie-via-bez-vn)
            (aspectc c5 (term1 c3) (term2 c4))
            (storec c5))
]
[e2
]
[e3   (n0 :a (bindc c3 discourse focus c2)
            (next n1))
      (n1 :q bound c2
              [bound n2]
              [unbound n3])
      (n2 :a)
      (n3 :a)
]
]

[word-expert hand

[e0   (n0 :a (createc c1 entity)
            (next n1))
      (n1 :q signal s0
              [entity-construction n2]
              [entity-construction? n2]
              [* n3])
      (n2 :a (pause e1))
      (n3 :a (pause e4))
]
[e1   (n0 :a (declareg)
            (refinec c1 =c#lichaamsdeel)
            (refinec c1 =c#armuiteinde)
            (link c1)
            (next n1))
      (n1 :q partofword x0
              [y n2]
              [n n5])
      (n2 :a (readw w1)
            (signal entity-construction
                           (to w1)
                           (concept c1)))
      (n5 :a (continue e3))
]
[e3   (n0 :a (refinec c1 =c#enkelvoud)
            (report c1))
```

```
]
[e4   (n0 :a (openg entity-construction)
            (refinec c1 =c#onbepaald)
            (continue e1))
]
]

[word-expert hilde

[e0   (n0 :a (openg entity-construction)
            (declareg)
            (continue e1))
]
[e1   (n0 :a (createc c1 entity)
            (refinec c1 =c#persoon)
            (refinec c1 =c#vrouwelijk)
            (refinec c1 =c#lief)
            (refinec c1 =c#mooi)
            (refinec c1 =c#bruinoog)
            (refinec c1 =c#wipneus)
            (link c1)
            (report c1)
            (closeg complete-entity))
]
]

[word-expert houd

[e0   (n0 :q signal s0
              [break n1]
              [passive n2]
              [* n3])
       (n1 :a (buildc c0 zinstype
                     (oneof =c#vraag =c#bevel)
                     (noneof =c#mededeling))
             (report c0)
             (continue e1))
       (n2 :a (bindc c1 immediate c0)
             (refinec c1 =c#passief)
             (refinec c1 =c#gegeten-worden)
             (declareg)
             (link c1)
             (closeg complete-action))
       (n3 :a (createc c0 zinstype)
             (refinec c0 =c#mededeling)
```

```
            (report c0)
            (continue e1))
]
[e1   (n0 :a (openg action-construction)
            (declareg)
            (createc c1 action)
            (refinec c1 =c#houden)
            (link c1)
            (next n1))
      (n1 :q partofword x0
              [y n2]
              [n n3])
      (n2 :a (readw w1)
            (signal action-construction
                     (to w1)
                     (concept c1))
            (pause e2))
      (n3 :a)
]
[e2   (n0 :a (buildc c2 entity
                     (role agent)
                     (oneof =c#persoon =c#ding =c#groep))
            (bindc c3 memory active c2)
            (continue e11)
            (continue e31)
            (next n1))
      (n1 :q bound c3
              [unbound n2]
              [bound n3])
      (n2 :a (await concept entity
                     (filter c2)
                     (bindconcept c3)
                     (wait group 2)
                     (continue e5)
                     (else e3)))
       (n3 :a (refinec c0 =c#meded-rechte-orde)
             (continue e4))
]
[e3    (n0 :a (buildc c4 entity
                     (role object)
                     (oneof =c#dier =c#ding =c#persoon))
             (bindc c5 memory active c4)
             (next n1))
        (n1 :q bound c5
             [unbound n2]
```

```
          [bound n3])
    (n2 :a (await concept entity
                 (filter c4)
                 (bindconcept c5)
                 (wait break 1)
                 (continue e7)))
    (n3 :a (buildc c8 zinstype
                 (allof =c#mededeling =c#inversie))
           (bindc c9 memory active c8)
           (next n4))
    (n4 :q bound c9
           [unbound n5]
           [bound n6])
    (n5 :a (refinec c0 =c#bevel)
           (report c0)
           (continue e7))
    (n6 :a)
]
[e31  (n0 :a (buildc c22 setting
                 (role vzvw))
          (await concept setting
                 (filter c22)
                 (bindconcept c20)
                 (report here)
                 (wait break 1)
                 (continue e32)))
]
[e32  (n0 :a (aspectc c1 (vzvw c20))
          (rolec c20 vzvw)
          (report c20)
          (refinec c1 =c#graag-mogen)
          (report c1))
]
[e4  (n0 :a (aspectc c1 (agent c3))
          (rolec c3 agent)
          (report c1)
          (continue e3))
]
[e5   (n0 :a (buildc c8 zinstype
                 (value =c#mededeling))
          (bindc c9 memory active c8)
          (next n1))
      (n1 :q bound c9
          [bound n2]
          [unbound n3])
```

```
      (n2 :a (refinec c9 =c#meded-inversie)
             (continue e4))
      (n3 :a (refinec c0 =c#vraag)
             (report c0)
             (continue e4))
]
[e6
]
[e7   (n0 :a (aspectc c1 (object c5))
            (rolec c5 object)
            (refinec c1 =c#iets-houden)
            (report c1))
]
[e11   (n0 :a (await signal
                      (filter particle)
                      (bindsender w1)
                      (report here)
                      (wait break 1)
                      (continue e12)))
]
[e12   (n0 :q literal w1
                [in n1]
                [op n2]
                [aan n3])
       (n1 :a)
       (n2 :a (addlex c1 w1)
              (refinec c1 =c#ophouden)
              (report c1))
]
]


[word-expert in

[e0   (n0 :q partofword x0
               [y n1]
               [n n5])
      (n1 :a (declareg)
             (next n2))
      (n2 :q signal s0
               [entity-construction n3]
               [* n4])
      (n3 :a)
      (n4 :a (continue e11))
      (n5 :a (continue e1))
```

```
]
[e11  (n0 :a (readw w1)
             (signal action-construction
                                 (to w1)))
]

[e1   (n0 :a (openg setting)
             (declareg)
             (createc c2 setting)
             (await concept entity
                        (bindconcept c1)
                        (report here)
                        (wait group 2)
                        (continue e3)
                        (else e5)))
]
[e3   (n0 :q view c1
                [=c#tijd  n1]
                [=c#plaats n2]
                [=c#iets-anders n3])
      (n1 :a (link c2)
             (aspectc c2 (oblique c1))
             (rolec c1 oblique)
             (rolec c2 bwbep-tijd)
             (report c2)
             (storec c1)
             (closeg complete-setting))
      (n2 :a (link c2)
             (aspectc c2 (oblique c1))
             (rolec c1 oblique)
             (rolec c2 plaats)
             (report c2)
             (closeg complete-setting)
             (storec c1))
      (n3 :a (buildc c3 action)
             (bindc c4 memory active c3)
             (next n4))
      (n4 :q bound c4
                [bound n5]
                [unbound n6])
      (n5 :a (signal vzvw (concept c1)))
      (n6 :a (await concept action
                       (bindconcept c4)
                       (report here)
                       (wait break 1)
```

```
                    (continue e4)))
]
[e4   (n0 :a (signal vzvw (concept c1)))
]
[e5   (n0 :a (signal particle))
]
[e6
]
]

[word-expert  maandag

[e0   (n0 :a (createc c1 entity)
             (next n1))
      (n1 :q signal s0
               [entity-construction n2]
               [entity-construction? n2]
               [setting n4]
               [* n3])
      (n2 :a (continue e1))
      (n3 :a (continue e4))
      (n4 :a (continue e4))
]
[e1   (n0 :a (declareg)
             (refinec c1 =c#tijdstip)
             (refinec c1 =c#1#eerste-weekdag)
             (link c1)
             (next n1))
      (n1 :q partofword x0
               [y n2]
               [n n5])
      (n2 :a (readw w1)
             (signal entity-construction
                          (to w1)
                          (concept c1)))
      (n5 :a (continue e3))
]
[e3   (n0 :q signal s0
               [setting n1]
               [* n2])
      (n1 :a (refinec c1 =c#enkelvoud)
             (closeg complete-setting)
             (report c1))
      (n2 :a (refinec c1 =c#enkelvoud)
             (report c1))
```

```
]
[e4   (n0 :a (openg setting)
             (refinec c1 =c#onbepaald)
             (continue e1))
]
]


[word-expert  man

[e0   (n0 : a (createc c1 entity)
              (next n1))
      (n1 : q signal s0
                [entity-construction n2]
                [entity-construction? n2]
                [* n5])
      (n2 :q literal w0
                [mijn n3]
                [haar n3]
                [*  n4])
      (n3 :a (continue e4))
      (n4 :a (continue e1))
      (n5 :a (continue e5))
]
[e1   (n0 :a (declareg)
             (refinec c1 =c#persoon)
             (refinec c1 =c#volwassen-mannelijk)
             (link c1)
             (next n1))
      (n1 :q partofword x0
               [y n2]
               [n n5])
      (n2 :a (readw w1)
             (signal entity-construction
                              (to w1)
                              (concept c1)))
      (n5 :a (continue e3))
]
[e3   (n0 :a (refinec c1 =c#enkelvoud)
             (report c1))
]
[e4   (n0 :a (declareg)
             (refinec c1 =c#persoon)
             (refinec c1 =c#echtgenoot)
             (link c1)
```

```
            (report c1))
]
[e5   (n0 :a (openg entity-construction)
            (refinec c1 =c#onbepaald)
            (continue e1))
]
]

[word-expert  op

[e0   (n0 :q partofword x0
               [y n1]
               [n n5])
      (n1 :q signal s0
               [entity-construction n3]
               [* n4])
      (n3 :a)
      (n4 :a (openg action-construction)
            (declareg)
            (continue e11))
      (n5 :a (continue e1))
]
[e11  (n0 :a (readw w1)
            (signal action-construction
                              (to w1)))
]
[e1   (n0 :a (openg setting)
            (declareg)
            (createc c2 setting)
            (await concept entity
                      (bindconcept c1)
                      (report here)
                      (wait break 1)
                      (continue e3)
                      (else e5)))
]
[e3   (n0 :q view c1
               [=c#tijd  n1]
               [=c#plaats n2]
               [=c#iets-anders n3])
      (n1 :a (link c2)
            (aspectc c2 (oblique c1))
            (rolec c1 oblique)
            (rolec c2 bwbep-tijdstip)
            (report c2)
```

```
          (closeg complete-setting)
          (storec c1))
     (n2 :a (link c2)
          (aspectc c2 (oblique c1))
          (rolec c1 oblique)
          (rolec c2 bwbep-plaats)
          (report c2)
          (closeg complete-setting)
          (storec c1))
     (n3 :a (buildc c3 action)
          (bindc c4 memory active c3)
          (next n4))
     (n4 :q bound c4
              [bound n5]
              [unbound n6])
     (n5 :a (signal vzvw (concept c1)))
     (n6 :a (await concept action
                   (bindconcept c4)
                   (report here)
                   (wait break 1)
                   (continue e4)))
]
[e4  (n0 :a (signal vzvw (concept c1)))
]
[e5  (n0 :a (signal particle))
]
]


[word-expert  rood

[e0  (n0 :q signal s0
              [* n1])
     (n1 :a (continue e1))
]
[e1  (n0 :a (openg concept-construction)
          (createc c1 eigenschap)
          (refinec c1 =c#kleur)
          (refinec c1 =c#rood)
          (declareg)
          (link c1)
          (closeg concept-construction)
          (refinec c1 =c#predicatief-adj)
          (signal predicatief)
          (report c1))
```

```
]
]

[word-expert van

[e0   (n0 :q signal s0
              [* n1])
      (n1 :a (continue e1))
]
[e1   (n0 :a (openg setting)
             (declareg)
             (createc c2 setting)
             (await concept entity
                    (bindconcept c1)
                    (report here)
                    (wait group 2)
                    (continue e3)
                    (else e5)))
]
[e3   (n0 :q view c1
              [=c#beginpunt  n1]
              [=c#plaats-herkomst n2]
              [=c#bezitter n3]
              [=c#materie n4]
              [=c#last-hinder n5]
              [=c#oorzaak n5]
              [=c#geen-van-vorige n5])
      (n1 :a (link c2)
             (aspectc c2 (oblique c1))
             (rolec c1 oblique)
             (rolec c2 bwbep-plaats)
             (report c2)
             (closeg complete-setting)
             (storec c1))
      (n2 :a (link c2)
             (aspectc c2 (oblique c1))
             (rolec c1 oblique)
             (rolec c2 bwbep-herkomst)
             (report c2)
             (closeg complete-setting)
             (storec c1))
      (n3 :a)
      (n4 :a)
      (n5 :a (buildc c3 action)
             (bindc c4 memory active c3)
```

```
            (next n6))
        (n6 :q bound c4
                [bound n7]
                [unbound n8])
        (n7 :a (continue e4))
        (n8 :a (await concept action
                        (bindconcept c4)
                        (report here)
                        (wait break 1)
                        (continue e4)))
]
[e4   (n0 :a (link c2)
            (aspectc c2 (oblique c1))
            (rolec c1 oblique)
            (rolec c2 vzvw)
            (report c2)
            (closeg complete-setting)
            (storec c1))
]
[e5   (n0 :a (signal particle))
]
[e6
]
]


[word-expert veel

[e0  (n0:a (await signal
                        (filter setting break)
                        (wait word 1)
                        (continue e1)
                        (else e4)))
]
[e1  (n0:a (buildc c1 entity)
                        (role object))
          (bindc c2 memory active c1)
          (next n1))
     (n1:q bound c2
                [bound n2]
                [unbound n3])
     (n2:a (createc c3 setting)
          (addlex c3 x0)
          (refinec c3 =c#dikwijls)
          (rolec c3 bijw-bep)
```

```
          (report c3))
     (n3:a (buildc c4 setting
                    (role vzvw))
          (bindc c5 memory expect c4)
          (next n4))
     (n4:q bound c5
              [bound n5]
              [unbound n6])
     (n5:a (createc c3 setting)
          (addlex c3 x0)
          (continue e3))
]
[e2  (n0:q view c3
              [=c#dikwijls n1]
              [=c#intens   n2])
     (n1:a (refinec c3 =c#dikwijls)
          (rolec c3 bijw-bep)
          (report c3))
     (n2:a (refinec c3 =c#intens)
          (rolec c3 bijw-bep)
          (report c3))
]
[e3  (n0:a (bindc c2 memory expect c1)
          (next n1))
     (n1:q bound c2
              [bound n2]
              [unbound n3])
     (n2:a (addlex c2 x0)
          (refinec c2 =c#grote-hoev)
          (report c2))
     (n3:a (createc c3 setting)
          (addlex c3 x0)
          (continue e2))
]
[e4  (n0:a (openg entity-construction)
          (declareg)
          (await concept entity
                     (bindconcept c1)
                     (report here)
                     (wait group 1)
                     (continue e5)
                     (else e6)))
]
[e5  (n0:a (refinec c1 =c#1#groot-aantal)
          (link c1)
```

```
        (closeg complete-entity)
        (report c1))
]
[e6  ; metanode
]
]

(word-expert  vrouw

[e0  (n0 : a (createc c1 entity)
          (next n1))
     (n1 : q signal s0
            [entity-construction n2]
            [entity-construction? n2]
            [* n5])

     (n2 :q literal w0
            [mijn n3]
            [zijn n3]
            [*  n4])
     (n3 :a (continue e4))
     (n4 :a (continue e1))
     (n5 :a (continue e5))
]
[e1  (n0 :a (declareg)
        (refinec c1 =c#persoon)
        (refinec c1 =c#volwassen-vrouwelijk)
        (link c1)
        (next n1))
     (n1 :q partofword x0
           [y n2]
           [n n5])
     (n2 :a (readw w1)
        (signal entity-construction
                          (to w1)
                          (concept c1)))
     (n5 :a (continue e3))
]
[e3  (n0 :a (refinec c1 =c#enkelvoud)
        (report c1))
]
[e4  (n0 :a (declareg)
        (refinec c1 =c#persoon)
        (refinec c1 =c#echtgenote)
        (link c1)
```

```
            (report c1))
]
[e5   (n0 :a (openg entity-construction)
            (refinec c1 =c#onbepaald)
            (continue e1))
]
]


[word-expert word

[e0   (n0 :q signal s0
              [break n1]
              [* n2])
       (n1 :a (buildc c0 zinstype
                        (oneof =c#vraag =c#bevel)
                        (noneof =c#mededeling))
             (continue e1))
       (n2 :a (createc c0 zinstype)
             (refinec c0 =c#mededeling)
             (report c0)
             (continue e1))
]
[e1   (n0 :a (openg action-construction)
            (declareg)
            (createc c1 action)
            (refinec c1 =c#worden)
            (link c1)
            (next n1))
      (n1 :q partofword x0
              [y n2]
              [n n3])
      (n2 :a (readw w1)
            (signal action-construction
                        (to w1)
                        (concept c1))
            (pause e2))
      (n3 :a)
      (n4 :a)
]
[e2   (n0 :a (buildc c2 entity
                        (role object-or-affected))
            (bindc c3 memory active c2)
            (continue e8)
            (next n1))
```

```
      (n1 :q bound c3
              [unbound n2]
              [bound n3])
      (n2 :a (await concept entity
                     (filter c2)
                     (bindconcept c3)
                     (wait group 2)
                     (continue e5)
                     (else e3)))
       (n3 :a (refinec c0 =c#meded-rechte-orde)
             (continue e4))
]
[e3    (n0 :a (buildc c4 setting
                     (role door-bepaling))
             (bindc c5 memory active c4)
             (next n1))
        (n1 :q bound c5
              [unbound n2]
              [bound n3])
        (n2 :a (await concept setting
                     (filter c4)
                     (bindconcept c5)
                     (wait break 1)
                     (continue e7)))
        (n3 :a (buildc c8 zinstype
                     (allof =c#mededeling =c#inversie))
              (bindc c9 memory active c8)
              (next n4))
        (n4 :q bound c9
              [unbound n5]
              [bound n6])
        (n5 :a (refinec c0 =c#bevel)
              (report c0)
              (continue e7))   ; overbodig : geen bevel mogelijk
        (n6 :a)
]
[e4   (n0 :a (await signal
                     (filter complete-action)
                     (wait break 1)
                     (continue e41)))
]
[e41   (n0 :a (aspectc c1 (object c3))
             (rolec c3 object))
]
[e5    (n0 :a (buildc c8 zinstype
```

```
                        (value =c#mededeling))
               (bindc c9 memory active c8)
               (next n1))
          (n1 :q bound c9
                  [bound n2]
                  [unbound n3]))
          (n2 :a (refinec c9 =c#meded-inversie)
                 (continue e4))
          (n3 :a (refinec c0 =c#vraag)
                 (report c1)
                 (continue e4))
]
[e7  (n0 :a (aspectc c1 (door-bepaling c5))
            (rolec c5 door-bepaling)
            (aspectc c1 (object-van-passief c3))
            (rolec c3 object-van-passief))
]
[e8  (n0 :a (await signal
                        (filter predicatief)
                        (wait break 1)
                        (continue e9)))
]
[e9  (n0 :a (aspectc c1 (affected c3))
            (rolec c3 affected))
]
]

(word-expert  zijn

[e0  (n0 :a (openg entity-construction)
            (declareg)
            (pause e1))
]
[e1  (n0 :a (buildc c1 entity
                        (oneof =c#mannelijk-persoon =c#ding))
            (bindc c2 memory active c1)
            (next n1))
     (n1 :q bound c2
              [bound n2]
              [unbound n5])
     (n2 :q view c2
              [=c#mannelijk-persoon n3]
              [=c#ding n4])
     (n3 :a (await concept entity
                        (bindconcept c3)
```

```
                        (wait group 1)
                        (continue e2)))
     (n4 :a (await concept entity
                        (bindconcept c3)
                        (wait group 1)
                        (continue e3))))
     (n5 :a (continue e5))
]
[e2  (n0 :a (createc c4 relatie)
            (refinec c4 =c#1#relatie-via-bez-vn)
            (aspectc c4 (term1 c2) (term2 c3))
            (storec c4)
            (refinec c3 =c#bepaald-door-bezvn)
            (closeg complete-entity))
]
[e3
]
[e4
]
[e5  (n0 :a (bindc c2 memory focus c1)
            (next n1))
     (n1 :q bound c2
              [bound n2]
              [unbound n3])
     (n2 :a)
     (n3 :a (await concept entity
                        (bindconcept c4)
                        (wait group 1)
                        (continue e6))))
]
[e6  (n0 :a (refinec c4 =c#bepaald)
            (refinec c4 =c#onopgeloste-anafoor)
            (closeg complete-entity))
]
]

[word-expert zomer

[e0  (n0 :a (createc c1 entity)
            (next n1))
     (n1 :q signal s0
              [entity-construction n2]
              [entity-construction? n2[
              [* n3]])
     (n2 :a (pause e1))
```

```
        (n3 :a (pause e4))
]
[e1   (n0 :a (declareg)
              (refinec c1 =c#tijdstip)
              (refinec c1 =c#tweede-seizoen)
              (link c1)
              (next n1))
        (n1 :q partofword x0
                 [y n2]
                 [n n5])
        (n2 :a (readw w1)
              (signal entity-construction
                                (to w1)
                                (concept c1)))
        (n5 :a (continue e3))
]
[e3   (n0 :a (refinec c1 =c#enkelvoud)
              (report c1))
]
[e4   (n0 :a (openg entity-construction)
              (refinec c1 =c#onbepaald)
              (continue e1))
]
]
```

APPENDIX 3: EXAMPLE PARSING TRACE

"BELT DE VROUW HAAR MAN VEEL OP OP MAANDAG ?"
("DOES THE WOMAN CALL UP HER HUSBAND A LOT ON MONDAY ?")

For clarity's sake a note about the slots in the concepts is in order. In the output structure discussed in 4.2.2.2. concepts have a ROLES and/or ASPECTS slot, a LEXICAL slot, a VALUE slot and a TYPE slot. Here the VALUE slot is called "ALLOF", and concepts can also contain a "ONEOF" slot (indicating that they are one of a number of possibilities) and a "NONEOF" slot (indicating that they are not one of a number of other possibilities). The "ONEOF" and "NONEOF" slots are used in the questions asked in the course of the process. See also the comments on the contents of active memory given at the end of this trace below.

```
-------------------------------------------------------
-> (wIwep)

                    Word Expert Parser
                       Version 8.0
                    University of Rochester (Steve Small)
                    University of Leuven (Geert Adriaens)

[wep] script voorbeeld

[WEP 8.0 Recording Initialized Thu Nov 28 14:17:44 1985]

[wep] trace 25
[wep] parse

          Word Expert Parser

=> belt de vrouw haar man veel op op maandag ?
=> _

----------------------------------------
```

This parsing trace lists the entries

and nodes chosen during expert tree
traversal; it echoes most of the
questions asked and actions taken.
For the *bel* expert it is
easy to see what happens if one also
looks at the process for *bel-*
in Appendix 2.

```
------------------------------------------


                                reading: belt
                                initializing: bel/e00085
                                initializing: -t/e00086
                                queueing: e00085/initentry

**************************************************************

           e00085/bel

=========================================
entry/initentry * expert/nil
concept/nil * signal/break
=========================================

*entry: initentry
                                tick: word/1
*entry: e0
**node: n0/q
        qsignal: s0/break
**node: n1/a
        createc: c00090/zinstype
                 building concept message
        report: c00090
                                storec: c00090
*entry: e1
**node: n0/a
                                tick: group/1
                                control state: action-construction
        signal: action-construction
        createc: c00098/action
        refinec: c00098/c#bellen
**node: n1/q
        qpartofword: e00085/bel
**node: n2/a
        signal: action-construction
```

```
                                signal demon fires: d00105/e00086/-t
                                queueing: e00086/initentry
                                exiting: e00085
------------------------------------------
```

Example of morphological interaction:
morphological analysis has segmented
belt as *bel* + *-t*,
and both experts have been initialized
automatically. *bel* asks the
qpartofword question, finds out that it
is part of a complex word and sends the
action-construction signal to its right
neighbour.

The next expert (*-t*) catches the
signal ("signal demon fires") and executes
refining "bel" as a singular action.

```
------------------------------------------


**************************************************************
          e00086/-t

=====================================
entry/initentry * expert/e00085
concept/c00098 * signal/action-construction
=====================================

*entry: initentry
                                tick: word/2
*entry: e0
**node: n0/q
        qsignal: s0/action-construction
**node: n1/a
        bindc: immediate: c00098/succeeds
*entry: e1
**node: n0/a
        refinec: c00098/c#1#actie-enkelvoud
                                control state: complete-action
        signal: complete-action
        report: c00098
                                storec: c00098
                                exiting: e00086

                                signal demon fires: d00108/e00085/bel
```

```
                                queueing: e00085/e2

**************************************************************

-----------------------------------------
Bel resumes after having paused
to let -t refine it first;
it starts the search for concepts that
can fulfil roles in its dynamic
caseframe.
-----------------------------------------

            e00085/bel

=========================================
entry/e2 * expert/wep
concept/nil * signal/pause
=========================================

*entry: e2
**node: n0/a
        createc: c00120/entity
                 building concept message
        bindc: active/fails
**node: n1/q
        qbound: c3/unbound
**node: n2/a
        posting restart demon: d00127
        **await: concept/e5
        **else: group/3/e3
*entry: e11
**node: n0/a
        posting restart demon: d00130
        **await: signal/e12
        **else: break/1/timeout
                                exiting: e00085

                                reading: de
                                initializing: de/e00131
                                queueing: e00131/initentry

**************************************************************

-----------------------------------------
```

The *de* expert first checks if
it is part of the Latin expression
"de facto" or "de iure"; this is not
the case, so it starts waiting for
a concept (which is the essence of the
article in language understanding).

```
----------------------------------------

          e00131/de

=======================================
entry/initentry * expert/e00086
concept/nil * signal/complete-action
=======================================

*entry: initentry
                               tick: word/3
*entry: e0
**node: n0/a
**node: n1/q
        qliteral: e00134/vrouw
**node: n2/a
                               tick: group/2
                               control state: entity-construction
        signal: entity-construction
*entry: e1
**node: n0/a
        posting restart demon: d00144
        **await: concept/e2
        **else: group/3/timeout
                               exiting: e00131

                               queueing: e00134/initentry

***********************************************************

----------------------------------------
```

Vrouw (woman/wife) creates a
concept, checks whether it could mean
"wife" by looking at the word that
started the lexical sequence it is part
of (if it were *zijn* ("his"),
vrouw would be refined as "wife";
here it is *de*, which leads to
the refinement "adult-female-person".

```
-----------------------------------------

          e00134/vrouw

=======================================
entry/initentry * expert/e00131
concept/nil * signal/entity-construction
=======================================

*entry: initentry
                              tick: word/4
*entry: e0
**node: n0/a
        createc: c00147/entity
**node: n1/q
        qsignal: s0/entity-construction
**node: n2/q
        qliteral: e00131/de
**node: n4/a
*entry: e1
**node: n0/a
        refinec: c00147/c#persoon
        refinec: c00147/c#volwassen-vrouwelijk
**node: n1/q
        qpartofword: e00134/vrouw
**node: n5/a
*entry: e3
**node: n0/a
        refinec: c00147/c#enkelvoud
        report: c00147
                              concept demon fires: d00144/e00131/de

                              queueing: e00131/e2
                              exiting: e00134


************************************************************

-----------------------------------------
```

De catches the "vrouw" concept
just reported, checks whether it was
introduced earlier in the discourse,
and -- finding out this is not the
case -- refines it as newly introduced

definite concept.

```
            e00131/de

=======================================
entry/e2 * expert/e00134
concept/c00147 * signal/nil
=======================================

*entry: e2
**node: n0/a
                                 control state: complete-entity
        signal: complete-entity

q> **********==Discourse Focus==**********
q>
q> c00147
q> **   allof: c#enkelvoud c#volwassen-vrouwelijk c#persoon
               c#anything

q> **lexical: (de vrouw)
q>
q> Is such a concept in focus?  no
        bindc: focus/fails
**node: n1/q
        qbound: c2/unbound
**node: n2/a
        refinec: c00147/c#nieuw-bepaald
        report: c00147
```

The demon created by the agent expectation
of *bel* checks the "de vrouw"
concept (c00147) to see if it matches
the expectation filter (c00120); since
it does, *bel* executes next to
incorporate c00147 into its caseframe.

```
        **Testing concept demon d00127 for filter match**

q> **********==Multiple Perspective==**********
q>
q> c00147
```

```
q> **   allof: c#nieuw-bepaald c#enkelvoud c#volwassen-vrouwelijk
             c#persoon c#anything

q> **lexical: (de vrouw)
q>
q> c00120
q> **   allof: c#anything
q> **   oneof: c#groepering c#persoon
q> **  role: agent
q>
q> Assuming that former can be viewed as the latter.

                              concept demon fires: d00127/e00085/bel
                              queueing: e00085/e5
                              storec: c00147
                              exiting: e00131


**********************************************************
```

Bel further refines sentence
structure (see 4.3.3.4): if it had been
refined as "mededeling" (declarative)
earlier, it would now be refined as
declarative-with-inversion; however,
it was refined as "either a question
or an imperative" (*Bel* opens
the sentence), and can now be further
refined as a question since the agent
was just found.
Bel now also starts waiting
for an object concept.

```
          e00085/bel

======================================
entry/e5 * expert/e00131
concept/c00147 * signal/nil
======================================

*entry: e5
**node: n0/a
```

```
        createc: c00176/zinstype
                 building concept message

q> **********==Multiple Perspective==**********
q>
q> c00090
q> **   allof: c#medvrbev
q> **   oneof: c#bevel c#vraag
q> **  noneof: c#mededeling

q>
q> c00176
q> **   allof: c#mededeling

q>
q> Assuming that former cannot be viewed as the latter.

        bindc: act                   storec: c00090
*entry: e4
**node: n0/a
        rolec: c00147/agent
        report: c00098
                               storec: c00098
*entry: e3
**node: n0/a
        createc: c00191/entity
                 building concept message
        bindc: active/fails
**node: n1/q
        qbound: c5/unbound
**node: n2/a
        posting restart demon: d00197
        **await: concept/e7
        **else: break/1/timeout
                               exiting: e00085

                               reading: haar
                               initializing: haar/e00198
                               queueing: e00198/initentry

************************************************************
```

Haar finds out that it is a
possessive pronoun (it is not part of

an NP started earlier) and immediately looks for a concept in memory that could be the "possessor" of the concept that is expected to follow *haar* (see below). In this case "de vrouw" is found as a matching candidate; if this had not been the case, the discours would have been probed to see if a candidate "possessor" concept had been introduced earlier on in the fragment of text.

```
----------------------------------------

           e00198/haar

======================================
entry/initentry * expert/e00131
concept/nil * signal/complete-entity
======================================

*entry: initentry
                              tick: word/5
*entry: e0
**node: n0/q
       qsignal: s0/complete-entity

**node: n2/a
                              tick: group/3
                              control state: entity-construction
       signal: entity-construction
       createc: c00205/entity
                building concept message

q> **********==Multiple Perspective==**********
q>
q> c00147
q> **   allof: c#nieuw-bepaald c#enkelvoud c#volwassen-vrouwelijk
              c#persoon c#anything
q> **  role: agent
q> **lexical: (de vrouw)
q>
q> c00205
q> **   allof: c#anything
q> **   oneof: c#abstractum c#vrouwelijk-persoon
```

```
q>
q>
q> Can the former be viewed as the latter? yes
       bindc: active/succeeds
**node: n3/q
       qbound: c3/c00147
**node: n4/q

q> ***********==Conceptual Proximity==**********
q>
q> c00147
q> **   allof: c#nieuw-bepaald c#enkelvoud c#volwassen-vrouwelijk
              c#persoon c#anything

q> **  role: agent
q> **lexical: (de vrouw)
q>
q> view concepts:
   c#vrouwelijk-persoon c#abstractum
q> Which views apply (best first) 1
q> Applicable views: c#vrouwelijk-persoon

       qview: c00147/c#vrouwelijk-persoon
```

Haar now creates an expectation
for a concept to its right (just like
de).

```
**node: n5/a
       posting restart demon: d00213
       **await: concept/el
       **else: group/4/timeout
                              exiting: e00198

                              reading: man
                              initializing: man/e00214
                              queueing: e00214/initentry

***********************************************************
```

Man creates an entity concept
and disambiguates itself as "echtgenoot"

("husband") by looking at the word that
started the lexical sequence it is part
of (i.e. *haar*, the possessive
pronoun; cp. *vrouw* above).
It reports this concept, which is awaited
by two other experts (*haar* and
bel, the latter waiting for an
object). *haar* gets it first,
and will execute next.

```
---------------------------------------

          e00214/man


=====================================
entry/initentry * expert/e00198
concept/nil * signal/entity-construction
=====================================

*entry: initentry
                              tick: word/6
*entry: e0
**node: n0/a
       createc: c00217/entity
**node: n1/q
       qsignal: s0/entity-construction
**node: n2/q
        qliteral: e00198/haar
**node: n3/a
*entry: e4
**node: n0/a
       refinec: c00217/c#persoon
       refinec: c00217/c#echtgenoot
       report: c00217
                              concept demon fires: d00213/e00198/haar
                              queueing: e00198/el

      **Testing concept demon d00197 for filter match**

q> **********==Multiple Perspective==**********
q>
q> c00217
q> **   allof: c#echtgenoot c#persoon c#anything
q>
q>
```

```
q> c00191
q> **   allof: c#anything
q> **   oneof: c#groepering c#persoon
q> **  role: object
q>
q> Assuming that former can be viewed as the latter.
```

The success of the match between "haar man" (c00217) and the filter of the object expectation of *bel* puts *bel* on the execution list after *haar*.

```
                              concept demon fires: d00197/e00085/bel
                              queueing: e00085/e7
                              storec: c00217
                              exiting: e00214
```

****

Haar catches the "man" concept and creates a concept "relationship", reflecting the anaphora-like link between "de vrouw" and "haar man"; these two concepts become term1 and term2 in the relationship concept respectively.

```
           e00198/haar

=======================================
entry/e1 * expert/e00214
concept/c00217 * signal/nil
=======================================

*entry: e1
**node: n0/a
        createc: c00231/relatie
        refinec: c00231/c#1#relatie-via-bez-vn
```

```
                              storec: c00231
                              exiting: e00198

************************************************************

----------------------------------------
Bel incorporates "haar man"
into its caseframe.
----------------------------------------

           e00085/bel

======================================
entry/e7 * expert/e00214
concept/c00217 * signal/nil
*entry: e7
**node: n0/a
        rolec: c00217/object
        refinec: c00098/c#telefoneren-naar
        report: c00098
                              storec: c00098
                              exiting: e00085

                              reading: veel
                              initializing: veel/e00240
                              queueing: e00240/initentry

************************************************************

----------------------------------------
Veel tentatively assumes that
it starts a concept/lexical sequence,
and waits for a signal from the next
expert to confirm or disconfirm its
assumption.
----------------------------------------

           e00240/veel

======================================
entry/initentry * expert/e00198
concept/nil * signal/entity-construction
======================================
```

*entry: initentry

tick: word/7

*entry: e0

**node: n0/a

control state: entity-construction?

signal: entity-construction?

posting restart demon: d00244

**await: signal/e1

**else: word/8/e2

exiting: e00240

reading: op

initializing: op/e00245

queueing: e00245/initentry

****

Op starts waiting for a concept
to its right (see 4.3.3.5), and also
provides a signal to *veel* that
allows it to reject its assumption
that it is a concept-starter.

----

e00245/op

==

entry/initentry * expert/e00240
concept/nil * signal/entity-construction?

==

*entry: initentry

tick: word/8

*entry: e0

**node: n0/q

qpartofword: e00245/op

**node: n5/a

*entry: e1

**node: n0/a

tick: group/4

control state: setting

signal: setting

signal demon fires: d00244/e00240/veel

```
                        queueing: e00240/e1
        createc: c00254/setting
        posting restart demon: d00256
        **await: concept/e3
        **else: break/1/e5
                        exiting: e00245
```

****

```
----------------------------------------
```

Veel now checks whether it could be the object to the verb by looking at the concepts already in memory; it finds out that an object is already present ("haar man") and refines itself as a one-word adjunct of time ("often"). (If this object had not been present, a complex interaction with memory would have been started, as discussed in 4.3.3.2.

```
----------------------------------------

          e00240/veel

========================================
entry/e1 * expert/e00245
concept/nil * signal/setting
========================================

*entry: e1
**node: n0/a
        createc: c00259/setting
**node: n1/a
        createc: c00264/entity
                building concept message
        bindc: active/succeeds
**node: n2/q
        qbound: c3/c00217
**node: n3/a
        refinec: c00259/c#dikwijls
                        storec: c00259
                        exiting: e00240
```

```
                                reading: op
                                initializing: op/e00271
                                queueing: e00271/initentry

**************************************************************

----------------------------------------
The second op also starts
waiting for a concept. (Both
op's are kept distinct by
the unique numbers assigned to them
by WEP (e00245 versus e00271)).
----------------------------------------

              e00271/op

=========================================

entry/initentry * expert/e00245
concept/nil * signal/setting
=========================================

*entry: initentry
                                tick: word/9
*entry: e0
**node: n0/q
        qpartofword: e00271/op
**node: n5/a
*entry: e1
**node: n0/a
                                tick: group/5
                                control state: setting
        signal: setting
        createc: c00280/setting
        posting restart demon: d00282
        **await: concept/e3
        **else: break/1/e5
                                exiting: e00271

                                reading: maandag
                                initializing: maandag/e00283
                                queueing: e00283/initentry

**************************************************************
```

```
---------------------------------------
Maandag creates, refines and
reports a time concept.
---------------------------------------

           e00283/maandag

=======================================
entry/initentry * expert/e00271
concept/nil * signal/setting

*entry: initentry
                              tick: word/10
*entry: e0
**node: n0/a
       createc: c00286/entity
**node: n1/q
       qsignal: s0/setting
**node: n4/a
*entry: e4
**node: n0/a
                              tick: group/6
                              control state: setting
       signal: setting
       refinec: c00286/c#onbepaald
*entry: e1
**node: n0/a
       refinec: c00286/c#tijdstip
       refinec: c00286/c#1#eerste-weekdag
**node: n1/q
       qpartofword: e00283/maandag
**node: n5/a
*entry: e3
**node: n0/q
       qsignal: s0/setting
**node: n1/a
       refinec: c00286/c#enkelvoud
                              control state: complete-setting
       signal: complete-setting
       report: c00286
                              concept demon fires: d00282/e00271/op
                              queueing: e00271/e3
                              exiting: e00283
```

```
************************************************************
---------------------------------------
The second op catches the concept
and incorporates it into a time adjunct.
---------------------------------------

           e00271/op

=======================================
entry/e3 * expert/e00283
concept/c00286 * signal/nil
=======================================

*entry: e3
**node: n0/q

q> **********==Conceptual Proximity==**********
q>
q> c00286
q> **   allof: c#enkelvoud c#1#eerste-weekdag c#tijdstip
             c#onbepaald c#anything

q> **lexical: (maandag)
q>
q> view concepts:
   c#tijd c#plaats c#iets-anders
q> Which views apply (best first) 1
q> Applicable views: c#tijd

        qview: c00286/c#tijd
**node: n1/a
        rolec: c00286/oblique
        rolec: c00280/bwbep-tijdstip
        report: c00280
                                storec: c00280
                                control state: complete-setting
        signal: complete-setting
                                storec: c00286
                                exiting: e00271

                                reading: #vraagteken#
                                initializing: #vraagteken#/e00319
                                queueing: e00319/initentry
```

```
***********************************************************

----------------------------------------
The question mark signals a sentence
break and checks memory for the
presence of an action (a verb); there
is one in memory, and no further
refinements are added to sentence
structure.  If no verb had been present,
sentence structure would have been
refined as elliptic.
----------------------------------------

           e00319/#vraagteken#

========================================
entry/initentry * expert/e00271
concept/nil * signal/complete-setting
========================================

*entry: initentry
                                tick: word/11
*entry: e0
**node: n0/a
                                tick: group/7
                                tick: break/1
                                control state: break
        signal: break
        createc: c00323/action
                 building concept message

q> **********==Multiple Perspective==**********
q>
q> c00098
q> **   allof: c#telefoneren-naar c#1#actie-enkelvoud
               c#bellen c#anyaction

q> **lexical: (bel -t)
q>
q> c00323
q> **   allof: c#anyaction

q>
q> Assuming that former can be viewed as the latter.
```

```
        bindc: active/succeeds
**node: n1/q
        qbound: c1/c00098
**node: n2/a
                                  exiting: e00319

                                  demon timeout: d00256/e00245/break
                                  queueing: e00245/e5

**********************************************************
```

The expectation of the first *op*
times out and the particle signal is
sent.

```
          e00245/op

======================================
entry/e5 * expert/wep
concept/nil * signal/else
======================================

*entry: e5
**node: n0/a
                                  control state: particle
        signal: particle
                                  signal demon fires: d00130/e00085/bel
                                  queueing: e00085/e12
                                  exiting: e00245

**********************************************************
```

Bel catches the particle
signal and incorporates its sender
(*op*) into its lexical sequence.

```
          e00085/bel

======================================
entry/e12 * expert/e00245
```

concept/nil * signal/particle
======================================

*entry: e12
**node: n0/a

exiting: e00085

Word Expert Parser

=> _

[wep] history

The history diagram shows which expert was executing at what moment during overall sentence processing (cp. Figure XIII in 4.3.3.3, pp. 172-173). The vertical axis contains the moments in time (starting at t1 and ending with t20); the vertical bars on the same level as the t's indicate which expert is executing at t*x* (the experts are in the sentence on top of the diagram).
Looking at the diagram from left to right one can see 1) how control is passed from expert to expert as it gradually moves over the sentence and 2) how many times an expert executed. (*De*, for instance, executed twice (at t4 and at t6).) As far as this second element depicted in the history diagram is concerned, it is interesting to note that the verb (*bel*) executes more often than any of the other experts (5 times, viz. at t1, t3, t7, t11 and t20), which nicely reflects the central role of the verb in sentence understanding.

h> h

		bel	-t	de	vrouw	haar	man	veel	op	op	maandag	#vraagteken#
T	t1	¦										
	t2		¦									
I	t3	¦										
	t4			¦								
M	t5				¦							
	t6			¦								
E	t7	¦										
	t8					¦						
	t9						¦					
C	t10					¦						
	t11	¦										
O	t12							¦				
	t13								¦			
U	t14							¦				
	t15									¦		
R	t16										¦	
	t17									¦		
S	t18											¦
	t19								¦			
E	t20	¦										

```
h> q
[wep] active
```

The ALLOF slot contains the refinements that were made to the concept (called VALUE slot in 4.2.2.2). I admit that the list contains a variety of heterogeneous refinements (syntactic, semantic), but as mentioned in 4.2.2.2, structures are relatively unimportant and only play a role as indirect control of the correct processing course (i.e. the refinements indicate that branching happened correctly within the expert processes).
Note the presence of two concepts without lexical sequence: the "relationship" concept created by *haar* (c00231) and the sentence structure concept created by *bel* (c00090). The latter concept shows that the process correctly discovered that it was dealing with a question.

```
          *** active memory ***


concept: c00286
  roles: ((oblique . c00280))
  lexical: ((maandag))
  allof: (c#enkelvoud c#1#eerste-weekdag c#tijdstip
          c#onbepaald c#anything)
  type: entity

<Type a <CR> to continue>
concept: c00280
  lexical: ((op maandag))
  role: bwbep-tijdstip
  aspects: ((oblique . c00286))
  type: setting

<Type a <CR> to continue>
concept: c00259
  role: implicit
  allof: (c#dikwijls c#adjunct)
  lexical: (veel)
  type: setting

<Type a <CR> to continue>
concept: c00231
  role: implicit
  aspects: ((term2 . c00217) (term1 . c00147))
```

```
  allof: (c#l#relatie-via-bez-vn c#anyrelation)
  type: relatie

<Type a <CR> to continue>
concept: c00217
  roles: ((object . c00098) (term2 . c00231))
  allof: (c#echtgenoot c#persoon c#anything)
  type: entity

<Type a <CR> to continue>
concept: c00147
  roles: ((term1 . c00231) (agent . c00098))
  lexical: ((de vrouw))
  allof: (c#nieuw-bepaald c#enkelvoud c#volwassen-vrouwelijk
         c#persoon c#anything)
  type: entity

<Type a <CR> to continue>
concept: c00098
  aspects: ((object . c00217) (agent . c00147))
  lexical: (op (bel -t))
  allof: (c#telefoneren-naar c#l#actie-enkelvoud c#bellen c#anyaction)
  type: action

<Type a <CR> to continue>
concept: c00090
  allof: (c#vraag c#medvrbev)
  type: zinstype

<Type a <CR> to continue>

[wep] q

[WEP 8.0 Recording Terminated Thu Nov 28 14:27:19 1985]
-> (exit)
```

BIBLIOGRAPHY

Abbreviations of periodicals:

AI Artificial Intelligence
BBS The Behavioral and Brain Sciences
Co Cognition
CoPs Cognitive Psychology
CoSc Cognitive Science
JPsR Journal of Psycholinguistic Research
JVLVB Journal of Verbal Learning and Verbal Behavior
Lang Language
Ling Linguistics
MemCo Memory and Cognition
PsRev Psychological Review

AARONSON, D. & R.W. RIEBER (eds) (1979) - Psycholinguistic Research: Implications and Applications. Lawrence Erlbaum, Hillsdale New Jersey.

ADRIAENS, G. (1985) - Process linguistics: missing link between linguistics, psychology and artificial intelligence (AI) ? University of Rochester Cognitive Science TR-28.

id. (1986a) - Word Expert Parsing: a natural language analysis program revised and applied to Dutch. In *Leuvense Bijdragen* 75 (1), 73-154.

id. (1986b) - WEP (Word Expert Parsing) revised and applied to Dutch. To appear in the Proceedings of the 7th European Conference on Artificial Intelligence (Brighton 21-25 July 1986).

id. & S.L. SMALL (forthcoming) - Word Expert Parsing: lexical ambiguity resolution and its implications for cognitive science. To appear in Small, Cottrell & Tanenhaus (eds) "Lexical Ambiguity Resolution in Human Language Understanding".

ANDERSON, J.R. (1983) - The Architecture of Cognition. Cognitive Science Series, 5. Harvard University Press, Cambridge Mass.

id. & G.H. BOWER (1973) - Human Associative Memory. Winston, Washington.

ANDERSON, R.C. & A. ORTONY (1975) - On Putting Apples into Bottles. A Problem of Polysemy. In *CoPs* 7, 167-180.

ARBIB, M.A. et al. (eds) (1982) - Neural Models of Language Processes. Academic Press, London.

BARCLAY, J.R. et al. (1974) - Comprehension and Semantic Flexibility. In *JVLVB* 13, 471-481.

BERWICK, R.C. (1983a) - Transformational Grammar and Artificial Intelligence: A Contemporary View. In *Cognition and Brain Theory* 6(4), 383-416.

id. (1983b) - Using what you know: A computer-science perspective. Comments on Stabler 1983. In *BBS* 6, 402-403.

id. (1985) - The Psychological Relevance of Transformational Grammar: A Reply to Stabler. In *Co* 19, 193-204.

BERWICK, R.C. & A.S. WEINBERG (1983) - The role of grammars in models of language use. In *Co* 13, 1-61.

id. (1984) - The Grammatical Basis of Linguistic Performance: Language Use and Acquisition. MIT Press, Cambridge Mass.

BEVER, T.G. et al. (eds) (1984) - Talking Minds: The Study of Language in Cognitive Science. MIT Press, Cambridge Mass.

BLANCHE-BIENVENISTE, C. et al. (1984) - Pronom et syntaxe. L'approche pronominale et son application au francais. Selaf, Paris.

BOBROW, D.G. & A. COLLINS (eds) (1975) - Representation and Understanding. Studies in Cognitive Science. Academic Press, London.

BOBROW, S. & S. BELL (1973) - On catching on to idiomatic expressions. In *MemCo* 1, 343-346.

BOCK, J.K. (1982) - Toward a Cognitive Psychology of Syntax: Information Processing Contributions to Sentence Formulation. In *PsRev* 89 (1), 1-47.

BOLINGER, D. (1965) - The Atomization of Meaning. In *Lang* 41, 555-573.

id. (1979) - Meaning and Memory. In Haydu 1979, 95-111.

BRADLEY, D.C. (1978) - Computational Distinctions of Vocabulary Type. MIT Phd.

BRADLEY, D.C. et al. (1980) - Syntactic Deficits in Broca's Aphasia. In Caplan 1980, 269-286.

BRANSFORD, J.D. & N.S. McCARRELL (1974) - A Sketch of a Cognitive Approach to Comprehension: Some Thoughts about Understanding What it Means to Comprehend. In Weimer & Palermo 1974, 189-229.

BRESNAN, J. (1978) - A Realistic Transformational Grammar. In Halle et al. 1978, 1-59.

id. (ed) (1982) - The Mental Representation of Grammatical Relations. MIT Press, Cambridge Mass.

id. & R.M. KAPLAN (1982/1984) - Grammars as Mental Representations of Language. Introduction of Bresnan 1982, xvii-lii. Reprinted in Kintsch et al. 1984, 103-135.

BRISCOE, E.J. (1985) - Determinism and its Implementation in PARSIFAL. In Sparck-Jones & Wilks, 61-68.

BUTTERWORTH, B. (ed) (1983a) - Language Production Volume 2. Development, Writing and Other Language Processes. Academic Press, London.

id. (1983b) - Lexical Representation. Chapter 7 in Butterworth 1983a, 257-294.

CAIRNS, H.S. et al. (1981) - Effects of Prior Context upon the Integration of Lexical Information during Sentence Processing. In *JVLVB* 20, 445-453.

CAPLAN, D. (ed) (1980) - Biological Studies of Mental Processes. MIT Press, Cambridge Mass.

CARAMAZZA, A. & E.B. ZURIF (1976) - Dissociation of algorithmic and heuristic processes in language comprehension: evidence from aphasia. In *Brain and Language* 3, 572-582.

CARAMAZZA, A. et al (1981) - Syntactic Processing Deficits in Aphasia. In *Cortex* 17, 333-348.

CARLSON, G. & M.K. TANENHAUS (1984) - Lexical Meanings, Structural Meanings, and Concepts. In Testen et al. 1984, 39-52.

CHARNIAK, E. (1983) - Passing Markers: A Theory of Contextual Influence in Language Comprehension. In *CoSc* 7, 171-190.

id. & D. McDERMOTT (1985) - Introduction to Artificial Intelligence. Addison-Wesley, Reading Mass.

& Y. WILKS (1976) - Computational Semantics. An Introduction to Artificial Intelligence and Natural Language Comprehension. (Fundamental Studies in Computer Science 4) North-Holland, Amsterdam.

CHOMSKY, N. (1957) - Syntactic Structures. Mouton, The Hague.

id. (1963) - Formal Properties of Grammars. In Luce et al. 323-418.

id. (1965) - Aspects of the Theory of Syntax. MIT Press, Cambridge Mass.

id. (1966) - Topics in the Theory of Generative Grammar. In Sebeok, 1-60.

id. (1969) - Linguistics and Philosophy. In Hook, 51-94.

id. (1976) - Conditions on Rules of Grammar. In *Linguistic Analysis* 2, 4.

id. (1980a) - Rules and Representations. In *BBS* 3, 1-61.

id. (1980b) - Rules and Representations. Columbia University Press, New York.

id. (1981) - Lectures on Government and Binding. Foris, Dordrecht.

id. (1982) - Some Concepts and Consequences of the Theory of Government and Binding. MIT Press, Cambridge Mass.

id. (1985) - Barriers. Unpublished Manuscript.

CLARK, H.H. & E.V. CLARK (1977) - Psychology and Language. An Introduction to Psycholinguistics. Harcourt Brace Jovanovich, New York.

CLARK, H.H. & S.E. HAVILAND (1974) - Psychological Processes as Linguistic Explanation. In Cohen 1974, 91-124.

CLIFTON, C. et al. (1984) - Lexical expectations in sentence comprehension. In *JVLVB* 23, 696-708.

COGEN, C. et al. (eds) (1975) - Proceedings of the First Annual Meeting of the Berkeley Linguistic Society. Berkeley, California.

COHEN, D. (ed) (1974) - Explaining Linguistic Phenomena. Hemisphere, Washington.

COLLINS, A. (1977) - Why Cognitive Science ? Editorial of *CoSc* 1(1), 1-2.

COLLINS, A.M. & E.F. LOFTUS (1975) - A Spreading-Activation Theory of Semantic Processing. In *PsRev* 82(6), 407-428.

CONNINE, C. et al. (1984) - Verb frame preferences: descriptive norms. In *JPsR* 13, 307-319.

CONRAD, C. (1974) - Context Effects in Sentence Comprehension: A Study of the Subjective Lexicon. In *MemCo* 2(1a), 130-138.

COOPER, W.E. & E.C.T. WALKER (eds) (1979) - Sentence Processing: Psycholinguistic Studies Presented to Merrill Garrett. Lawrence Erlbaum, Hillsdale New Jersey.

COTTRELL, G.W. (1985) - A Connectionist Approach to Word Sense Disambiguation. University of Rochester Computer Science PhD (TR-154). Rochester, NY

id. & S.L. SMALL (1983) - A Connectionist Scheme for Modelling Word Sense Disambiguation. In *Cognition and Brain Theory* 6(1), 89-120.

CRAIK, F.I.M. & R.S. LOCKHART (1972) - Levels of Processing: A Framework for Memory Research. In *JVLVB* 11, 671-684.

CUYCKENS, H. (1982) - Componentiële analyse versus niet-componentiële analyse. In *Interdisciplinair Tijdschrift voor Taal- en Tekstwetenschap* 2(3), 169-198.

DAMASIO, A.R. & N. GESCHWIND (1984) - The Neural Basis of Language. In *Annual Review of Neuroscience* 7, 127-174.

DENNETT, D.C. (1978) - Brainstorms: Philosophical Essays on Mind and Psychology. Bradford Books, MIT Press, Cambridge Mass.

DERWING, B.L. (1973) - Transformational Grammar as a Theory of Language Acquisition. A Study of the Empirical, Conceptual and Methodological Foundations of Contemporary Linguistics. Cambridge University Press, Cambridge.

DIEHL, L.G. (1981) - Lexical-generative grammar: toward a lexical conception of linguistic structure. Indiana University PhD. UMI, London.

DIK, S.C. (1978) - Functional Grammar. North-Holland, Amsterdam.

DRESHER, B.E. & N. HORNSTEIN (1976) - On Some Supposed Contributions of Artificial Intelligence to the Scientific Study of Language. In *Co* 4, 321-398.

id. (1977a) - Reply to Schank and Wilensky. In *Co* 5, 147-149.

id. (1977b) - Reply to Winograd. In *Co* 5, 379-392.

ESTES, W.K. (ed) (1978) - Handbook of Learning and Cognitive

Processes: Volume 6 Linguistic Functions in Cognitive Theory. Lawrence Erlbaum, Hillsdale New Jersey.

ESTILL, R.B. & S. KEMPER (1982) - Interpreting Idioms. In *JPsR* 11(6), 559-568.

FAHLMAN, S.E. (1979) - NETL: A System for Representing and Using Real-World Knowledge. MIT Press, Cambridge Mass.

id. (1982) - Three flavors of parallelism. In Proceedings of the Canadian Society for Computational Studies of Intelligence 1982, 230-235.

FELDMAN, J.A. & D.H. BALLARD (1982) - Connectionist Models and Their Properties. In *CogSci* 6, 205-254.

FERNANDO, C. & R. FLAVELL (1981) - On Idiom: critical views and perspectives. University of Exeter, Exeter.

FLANAGAN, O. J. Jr. (1984) - The Science of the Mind. Bradford Books, MIT Press, Cambridge Mass.

FODOR, J. & M. GARRET (1966) - Some Reflections on Competence and Performance. In Lyons & Wales, 135-154.

FODOR, J. et al. (1974) - The Psychology of Language. An Introduction to Psycholinguistics and Generative Grammar. McGraw-Hill, New York.

FODOR, J.A. (1981) - Representations: Philosophical Essays on the Foundations of Cognitive Science. MIT Press, Cambridge Mass.

id. (1983) - The Modularity of Mind. Bradford Books, MIT Press, Cambridge Mass.

FODOR, J.D. (1978) - Parsing strategies and constraints on transformations. In *Linguistic Inquiry*, 9, 427-474.

FORD, M. et al. (1982) - A Competence-Based Theory of Syntactic Closure. Chapter 11 in Bresnan 1982, 727-796.

FORSTER, K.I. (1979) - Levels of Processing and the Structure of the Language Processor. In Cooper & Walker 1979, 27-85.

FOSS, D. & C. JENKINS (1973) - Some Effects of Context on the Comprehension of Ambiguous Sentences. In *JVLVB* 12, 577-589.

FRANKS, J.J. (1974) - Toward Understanding Understanding. In Weimer & Palermo 1974, 231-261.

FRASER, B. (1976) - The Verb-Particle Combination in English. Academic Press, London.

FRAZIER, L. & J.D. FODOR (1978) - The Sausage Machine: A new two stage parsing model. In *Co* 6, 291-325.

FRONEK, J. (1982) - *Thing* as a function word. In *Ling* 20, 633-654.

GARDNER, H. (1983) - Frames of Mind: The Theory of Multiple Intelligences. Basic Books, New York.

id. (1985) - The Mind's New Science. A History of the Cognitive Revolution. Basic Books, New York.

GARNSEY, S.M. & R.M. CHAPMAN (1985) - Function and Content Word Reaction Times and Evoked Potentials During Lexical Decision. Unpublished manuscript.

GARRETT, M.F. (1976) - Syntactic Processes in Sentence Production. In Wales & Walker 1976, 231-256.

GAZDAR, G. et al. (1985) - Generalized Phrase Structure Grammar. Harvard University Press, Cambridge Mass.

GESCHWIND, N. (1981) - Neurological Knowledge and Complex Behaviors. In Norman 1981a, 27-35.

GIBBS, R.W. Jr. (1980) - Spilling the beans on understanding and memory for idioms in conversation. In *MemCo* 8, 149-156.

GIGLEY, H.M. (1982) - Neurolinguistically Constrained Simulation of Sentence Comprehension: Integrating Artificial Intelligence and Brain Theory. University of Massachussetts Phd.

GIVÓN, T. (1984) - Deductive vs. Pragmatic Processing in

Natural Language. In Kintsch et al. 1984, 137-190.

GLASS, A.L. (1983) - The Comprehension of Idioms. In *JPsR* 12(4), 429-442.

GORDON, B. & A. CARAMAZZA (1982) - Lexical decision for open- and closed-class words: failure to replicate differential frequency sensitivity. In *Brain and Language* 15, 143-160.

GREENE, J. (1972) - Psycholinguistics. Penguin, Harmondsworth UK.

GROSS. M. (1979) - On the Failure of Generative Grammar. In *Lang* 55(4), 859-885.

id. (1984) - Lexicon-grammar and the Syntactic Analysis of French. Proceedings COLING 1984, 275-282.

HAIMAN, J. (1980) - Dictionaries and Encyclopedias. In *Lingua* 50, 329-357.

HALLE, M. et al. (eds) (1978) - Linguistic Theory and Psychological Reality. MIT Press, Cambridge Mass.

HARMAN, G. (1967) - Psychological Aspects of the Theory of Syntax. In *Journal of Philosophy* 64, 75-87.

HAUGELAND, J. (1984) - First Among Equals. In Kintsch et al. 1984, 85-99.

HAYDU, G.G (ed) (1979) - Experience Forms. Their Cultural and Individual Place and Function. Mouton, The Hague 1979.

HINTON, G.E. & J.A. ANDERSON (1981) (eds) - Parallel Models of Associative Memory. Lawrence Erlbaum, Hillsdale New Jersey.

HIRST, G. (1983) - A Foundation for Semantic Interpretation. In Proceedings ACL 21 (Cambridge, Mass), 64-73.

HOEKSTRA, T. et al. (eds) (1980) - Lexical Grammar. (Publications in Language Sciences 3. Glot 2(3/4).) Foris, Dordrecht.

HOFSTADTER, D.R. (1979) - Gödel, Escher, Bach: An Eternal Golden Braid. A Metaphorical Fugue on Minds and Machines in the Spirit of Lewis Carroll. Penguin, Harmondsworth.

id. & D.C. DENNETT (eds) (1981) - The Mind's I. Fantasies and Reflections on Self and Soul. The Harvester Press, Brighton.

HOLMES, V.M. (1977) - Prior Context and the Perception of Lexically Ambiguous Sentences. In *MemCo* 5, 103-110.

HOOK, S. (ed) (1960) - Dimensions of Mind. New York University Press, New York.

id. (ed) (1969) - Language and Philosophy. New York University Press, New York.

HOPPENBROUWERS, G.A.J. et al. (eds) (1985) - Meaning and the Lexicon. Foris, Dordrecht.

HöRMANN, H. (1982) - Hidden Determinants of Understanding. In Le Ny & Kintsch 1982, 251-260.

id. (1983) - On the Difficulties of Using the Concept of a Dictionary - and the Impossibility of Not Using it. In Rickheit & Bock 1983, 3-16.

HUDSON, R. (1984) - Word Grammar. Basil Blackwell, New York.

HYMES, D.H. (1972) - On Communicative Competence. In Pride & Holmes, 269-293.

JACKENDOFF, R.S. (1972) - Semantic Interpretation in Generative Grammar. MIT Press, Cambridge Mass.

JASTRZEMBSKI, J.E. (1981) - Multiple Meanings, Number of Related Meanings, Frequency of Occurrence, and the Lexicon. In *CoPs* 13, 278-305.

JOHNSON-LAIRD, P.N. (1977) - Procedural Semantics. In *Co* 5, 189-214.

id. (1981) - Mental Models in Cognitive Science. In Norman 1981a, 147-188.

JUST, M.A. & P.A. CARPENTER (1980) - A Theory of Reading: From Eye Fixations to Comprehension. In *PsRev* 87(4), 329-354.

KAPLAN, R.M. & J. BRESNAN (1982) - Lexical-Functional Grammar: A Formal System for Grammatical Representation. In Bresnan 1982, 173-281.

KATZ, J.J. (1981) - Language and Other Abstract Objects. Basil Black- well, Oxford.

id. & J.A. FODOR (1963) - The structure of a semantic theory. In *Lang* 39, 170-210.

id. & P.M. POSTAL (1964) - An Integrated Theory of Linguistic Descriptions. MIT Press, Cambridge Mass.

KEMPEN, G. & E. HOENKAMP (1982) - Incremental Sentence Generation: Implications for the Structure of a Syntactic Processor. In Proceedings COLING 1982, 151-156.

id. (1984) - An Incremental Procedural Grammar for Sentence Formulation. Unpublished research paper. Department of Psychology, Nijmegen.

KIMBALL, J. (1973) - Seven principles of surface structure parsing in natural language. In *Co* 2, 15-47.

KING, M. (ed) (1983) - Parsing Natural Language. Academic Press, London.

KINTSCH, W. (1984) - Approaches to the Study of the Psychology of Language. In Bever et al., 111-145.

id. et al. (eds) (1984) - Methods and Tactics in Cognitive Science. Lawrence Erlbaum, Hillsdale New Jersey.

KOLERS, P.A. & W.E. SMYTHE (1984) - Symbol Manipulation: Alternatives to the Computational View of Mind. In *JVLVB* 23, 289-314.

KUHN, T.S. (1962) - The structure of scientific revolutions. University of Chicago Press, Chicago.

LABOV, W. (1972) - Sociolinguistic Patterns. University of

Pennsylvania Press, Philadelphia PA.

LACKNER, J.R. & M.F. GARRETT (1972) - Resolving Ambiguity: Effects of Biasing Context in the Unattended Ear. In *Co* 1, 359-372.

LAKOFF, G. (1978) - Some Remarks on AI and Linguistics. In *CoSc* 2, 267-275.

id. & H. THOMPSON (1975) - Introducing Cognitive Grammar. In Cogen 1975, 295-313.

LANGACKER, R.W. (1983) - Foundations of Cognitive Grammar. Bloomington, Indiana.

id. (1985) - Topics in Cognitive Grammar. (Handouts to a series of lectures held at UFSAL, Brussels March 1985).

LANGENDOEN, D.T. (1979) - The Role of Grammar in the Use of Language. In Aaronson & Rieber 1979, 229-240.

LEHNERT, W.G. & M.H. RINGLE (eds) (1982) - Strategies for Natural Language Processing. Lawrence Erlbaum, Hillsdale New Jersey.

LE NY, J.-F. & W. KINTSCH (eds) (1982) - Language and Comprehension. (Advances in Psychology 9.) North-Holland, Amsterdam.

LEVELT, W.J.M. (1974) - Formal Grammars in Linguistics and Psycholinguistics. Vol I : An Introduction to the Theory of Formal Languages and Automata. Vol II : Applications in Linguistic Theory. Vol III: Psycholinguistic Applications. (Janua Linguarum Series Minor 192/1-3.) Mouton, The Hague.

LEWI, J. et al. (1982) - A Programming Methodology in Compiler Construction. Part 2: Implementation. North-Holland, Amsterdam.

LUCE, R. et al. (eds) (1966) - Handbook of Mathematical Psychology Vol II. Wiley, New York.

LYONS, J. & R.J. WALES (eds) (1966) - Psycholinguistics

Papers. (Proceedings of the 1966 Edinburgh Conference.) University Press, Edinburgh.

MAKKAI, A. (1972) - Idiom Structure in English. Mouton, The Hague.

MANDLER, G. (1984) - Cohabitation in the Cognitive Sciences. In Kintsch et al. 1984, 305-315.

MARCUS, M.P. (1980) - A Theory of Syntactic Recognition for Natural Language. MIT Press, Cambridge Mass.

id. (1982) - Consequences of Functional Deficits in a Parsing Model: Implications for Broca's Aphasia. In Arbib et al. 1982, 115-133.

id. (1984) - Some inadequate theories of human language processing. In Bever et al., 253-278.

MARIN, O.S.M. (1982) - Brain and Language: The Rules of the Game. In Arbib et al. 1982, 45-69.

MARR, D. (1982) - Vision: A Computational Investigation into the Human Representation and Processing of Visual Information. Freeman, San Francisco.

MARSLEN-WILSON, W.D. & A. WELSH (1978) - Processing Interactions and Lexical Access during Word Recognition in Continuous Speech. In *CoPs* 10, 29-63.

MARSLEN-WILSON, W.D. & L.K. TYLER (1980) - The Temporal Structure of Spoken Language Understanding. In *Co* 8, 1-71.

McCLELLAND, J.L. & D.E. RUMELHART (1981) - An Interactive Activation Model of Context Effects in Letter Perception: Part 1. An Account of Basic Findings. In *PsRev* 88(5), 375-407.

McCLELLAND, J.L. & D.E. RUMELHART (1982) - An Interactive Activation Model of Context Effects in Letter Perception: Part 2. The Contextual Enhancement Effect and Some Tests and Extensions of the Model. In *PsRev* 89(1), 60-94.

MILLER, G.A. (1978) - Semantic Relations among Words. In Halle et al. 1978, 60-118.

id. & N. CHOMSKY (1963) - Finitary Models of Language Users. In Luce et al. 419-491.

MILLER, J.R. et al (1984) - Problems of Methodology in Cognitive Science. In Kintsch et al. 1984, 1-18.

MINSKY, M. (ed) (1968) - Semantic Information Processing. MIT Press, Cambridge Mass.

MOORE, T. & C. CARLING (1982) - Language Understanding. Towards a Post-Chomskyan Linguistics. St. Martin's Press, New York.

MORTON, J. (1970) - A Functional Model for Memory. In Norman 1970, 203-254.

MYERS, T. et al. (eds) (1981) - The Cognitive Representation of Speech. North-Holland, Amsterdam.

NEWELL, A. (1981) - Physical Symbol Systems. In Norman 1981a, 37-85.

id. & H.A. SIMON (1972) - Human Problem Solving. Prentice-Hall, Englewood Cliffs New Jersey.

NORMAN, D.A. (ed) (1970) - Models of Human Memory. Academic Press, London.

id. (ed) (1981a) - Perspectives on Cognitive Science. Ablex, New Jersey.

id. (1981b) - What is Cognitive Science ? In Norman 1981a, 1-11.

ORTONY, A. et al (1978) - Interpreting metaphors and idioms. Some effects of context on comprehension. In *JVLVB* 17, 465-477.

PARRET, H. (1974) - Over de waarde van teoretische constructies: de competentie-performantie dichotomie. In *Leuvense Bijdragen* 63, 311-332.

PEREIRA, F.C.N. & D.H.D. WARREN (1980) - Definite Clause Grammar for Language Analysis. A Survey of the Formalism and a Comparison with Augmented Transition Grammars. In

AI 13, 231-278.

POLLACK, J. & D. WALTZ (1985) - Massively parallel parsing: A strongly interactive model of natural language interpretation. In *CogSci* 9, 51-74.

POLSON, P.G. et al. (1984) - Methods and Tactics Reconsidered. In Kintsch et al. 1984, 277-296.

PRIDE, J.B. & J. HOLMES (eds) (1972) - Sociolinguistics. Selected Readings. Harmondsworth UK.

PROUDIAN, D. & C. POLLARD (1985) - Parsing Head-driven Phrase Structure Grammar. In Proceedings of the 23rd Annual Meeting of the ACL. Chicago Ill, 167-171.

PUTNAM, H. (1975a) - Mind, Language and Reality: Philosophical Papers, vol. 2. Cambridge University Press, Cambridge Mass.

id. (1975b) - Minds and Machines. In Putnam 1975a, 362-385.

id. (1975c) - The 'innateness hypothesis' and explanatory models in linguistics. In Putnam 1975a, 107-116.

PYLYSHYN, Z.W. (1980) - Computation and Cognition: issues in the foundations of cognitive science. In *BBS* 3, 111-169.

id. (1984) - Computation and Cognition. Toward a Foundation for Cognitive Science. Bradford Books, MIT Press, Cambridge Mass.

RADFORD, A. (1981) - Transformational Syntax. A Student's Guide to Chomsky's Extended Standard Theory. Cambridge University Press, Cambridge.

RICKHEIT, G. & M. BOCK (eds) (1983) - Psycholinguistic Studies in Language Processing. Walter de Gruyter, Berlin.

RIEGER, C.J. & S.L. SMALL (1981) - Toward a Theory of Distributed Word Expert Natural Language Parsing. In *IEEE Transactions on Systems, Man, and Cybernetics* 11(1), 43-51.

RIESBECK, C.K. (1978) - An Expectation-Driven Production

System for Natural Language Understanding. In Waterman & Hayes-Roth 1978, 399-431.

RUMELHART, D.E. (1984) - The Emergence of Cognitive Phenomena from Sub-symbolic Processes. In Proceedings of the 6th Annual Conference of the Cognitive Science Society, Boulder Colorado 1984, 59-62.

SAMPSON, G.R. (1983a) - Deterministic Parsing. In King, 91-116.

id. (1983b) - Context-free Parsing and the Adequacy of Context-free Grammars. In King, 151-170.

SAUSSURE, F. DE (1949) - Cours de linguistique générale. (Publié par C. Bally & A. Sechehaye). Payot, Paris.

SCHANK, R.C. (1972) - Conceptual Dependency: A Theory of Natural Language Understanding. In *CoPs* 3(4), 552-631.

id. & L. BIRNBAUM (1984) - Memory, Meaning, and Syntax. In Bever et al. 1984, 209-251.

id. & K.M. COLBY (eds) (1973) - Computer Models of Thought and Language. Freeman, San Francisco.

id. & R. WILENSKY (1977) - Response to Dresher and Hornstein. In *Co* 5, 133-145.

id. & C.K. RIESBECK (eds) (1981) - Inside Computer Understanding. Five Programs Plus Miniatures. Lawrence Erlbaum, Hillsdale New Jersey.

SCHREUDER, R. & W.J.M. LEVELT (1978) - Psychologische theorieën over het lexicon. In *Forum der Letteren* 19(1), 40-58.

SEARLE, J.R. (1981) - Minds, Brains and Programs. In Hofstadter & Dennett 1981, 353-382.

SEBEOK, T.A. (ed) (1966) - Current Trends in Linguistics, Vol III. Theoretical Foundations. Mouton, The Hague.

SEIDENBERG, M.S. et al. (1982) - Automatic Access of the Meanings of Ambiguous Words in Context: Some Limitations of Knowledge-based Processing. In *CoPs* 14, 489-537.

id. (1984) - Pre- and Postlexical Loci of Contextual Effects on Word Recognition. In *MemCo* 12(4), 315-328.

SIMON, H.A. & K.A. ERICSSON (1984) - Protocol Analysis. Bradford Books, MIT Press, Cambridge Mass.

SLOBIN, D.I. (1966) - Comments on "developmental psycholinguistics". In Smith & Miller, 85-91.

id. (1984) - The cronlinguistic study of language acquisition. Lawrence Erlbaum, Hillsdale New Jersey.

SMALL, S.L. (1980) - Word Expert Parsing: A Theory of Distributed Word-Based Natural Language Understanding. Computer Science Technical Report Series. University of Maryland PhD.

id. (1981) - Demon timeouts: limiting the life span of spontaneous computations in cognitive models. In Proceedings of the 3rd Annual Conference of the Cognitive Science Society, Berkeley CA.

id. (1983) - Parsing as Cooperative Distributed Inference: Understanding through Memory Interactions. Chapter 12 in King 1983, 247-276.

id. & M.M. LUCAS (1983) - Word Expert Parsing: A Computer Model of Sentence Comprehension. University of Rochester Cognitive Science TR-1. Rochester NY.

id. & C.J. RIEGER (1982) - Parsing and Comprehending with Word Experts (A Theory and its Realization). In Lehnert & Ringle 1982, 89-147.

SMITH, F. & G.A. MILLER (1966) - The Genesis of Language: A Psycholinguistic Approach. MIT Press, Cambridge Mass.

SMITH, E.E. (1978) - Theories of Semantic Memory. In Estes 1978, 1-56.

SOAMES, S. (1984) - Linguistics and Psychology. In *Linguistics and Philosophy* 7, 155-179.

SPARCK JONES, K. & Y. WILKS (eds) (1985) - Automatic Natural Language Parsing. Ellis Horwood, Chichester.

STABLER, E.P. Jr. (1981) - How are grammars represented ? In *BBS* 6, 391-403.

id. (1984) - Berwick and Weinberg on Linguistics and Computational Psychology. In *Co* 17, 155-179.

STAROSTA, S. (1978) - Lexicase II: Lexicon. Unpublished Manuscript, University of Hawaii.

SWINNEY, D.A. (1979) - Lexical Access during Sentence Comprehension: (Re)Consideration of Context Effects. In *JVLVB* 18, 645-659.

id. (1981) - Lexical Processing during Sentence Comprehension: Effects of Higher Order Constraints and Implications for Representation. In Myers et al. 1981, 201-209.

id. & D.T. HAKES (1976) - Effects of Prior Context upon Lexical Access during Sentence Comprehension. In *JVLVB* 15, 681-689.

id. & A. CUTLER (1979) - The Access and Processing of Idiomatic Expressions. In *JVLVB* 18, 523-534.

TANENHAUS, M.K. et al. (1979) - Evidence for Multiple Stages in the Processing of Ambiguous Words in Syntactic Contexts. In *JVLVB* 18, 427-440.

TANENHAUS, M.K. et al. (1985) - The Interaction of Lexical Expectation and Pragmatics in Parsing Filler-Gap Constructions. In Proceedings of the 7th Annual Conference of the Cognitive Science Society. Irvine CA 1985, 361-365.

TAYLOR, I. (1976) - Introduction to Psycholinguistics. Holt, Rinehart and Winston, New York.

TAYLOR, S.M.H. (1980) - Lexical Idiosyncrasy in English: An Argument for a Lexically Based Grammar. University of Oregon PhD, UMI, London.

TESTEN, D. et al. (eds) (1984) - Papers from the Parasession on Lexical Semantics. Chicago Linguistic Society.

THOMPSON, H. (1985) - Natural Language Processing: a critical analysis of the structure of the field, with some implications for parsing. In Sparck Jones & Wilks 1985, 22-38.

TURKINGTON, C. (1985) - Imaging. Computer unlocks secrets in folds, functions of brain. In *Monitor of the APA*, September 1985, 12-13.

TYLER, L.K. (1980) - Serial and Interactive Parallel Theories of Syntactic Processing. MIT Center for Cognitive Studies Occasional paper No. 8.

VAN LANGENDONCK, W. (1985) - De bepaling van gesteldheid in een dependentienetwerk. In Linguistics in Belgium 6 (Antwerp Papers in Linguistics), 176-191. UIA Antwerp.

VANLEHN, K. (1984) - A critique of the connectionist hypothesis that recognition uses templates, and not rules. In Proceedings of the Sixth Annual Conference of the Cognitive Science Society (Boulder, Colorado), 74-80.

VAN RIEMSDIJK, H.C. & E. WILLIAMS (1986) - Introduction to the Theory of Grammar. (Current Studies in Linguistics Series 12.) MIT Press, Cambridge Mass.

WALES, R.J. & E. WALKER (eds) (1976) - New Approaches to Language Mechanisms. North-Holland, Amsterdam.

WATERMAN, D.A. & F. HAYES-ROTH (eds) (1978) - Pattern-directed Inference Systems. Academic Press, London.

WEIMER, W.B. & D.S. PALERMO (eds) (1974) - Cognition and the Symbolic Processes. Lawrence Erlbaum, Hillsdale New Jersey.

WILKS, Y. (1973) - An Artificial Intelligence Approach to Machine Translation. In Schank & Colby 1973, 114-151.

id. (1982) - Some Thoughts on Procedural Semantics. In Lehnert & Ringle 1982, 495-516.

WINOGRAD, T. (1975) - Frame Representations and the Declarative/Procedural Controversy. In Bobrow & Collins

1975, 185-210.

id. (1977) - On Some Contested Suppositions of Generative Linguistics about the Scientific Study of Language. A Response to Dresher and Hornstein's "On Some Supposed Contributions of Artificial Intelligence to the Scientific Study of Language." In *Co* 5, 151-179.

id. (1983) - Language as a Cognitive Process. Volume I: Syntax. Addison-Wesley, Reading Mass.