Language Networks: the new Word Grammar

Richard Hudson

Contents

Preface

This book is a collection of ideas about language – about how language is structured at every level, about the overall architecture of the whole system, and about how it fits into a larger framework of ideas about human cognition. The broad cognitive context is just as important as the detail about language structure precisely because my argument is that all the detail derives from this context. Language is not 'sui generis', a unique system which can, and should, be studied without reference to any other system; this may have been a healthy methodological antidote to the psychology of the early twentieth century, but the intellectual world has changed. Our intellectual neighbours have grown up into the healthy sciences of cognitive psychology and psycholinguistics, but intellectual isolationism is still strong on both sides. However well informed we may be about the neighbours' comings and goings, neither side really allows these developments to influence theoretical work on their side. (Just to give a small example, phonological theories ignore the popular psychological theory that working memory includes a 'phonological loop' (e.g. Baddeley and Logie 1999), which in turn evolved without any significant input from phonological theory.)

The structuralist tradition still dominates linguistics through the view that we can discover the structure of language just by applying the traditional methods of linguistics. This was especially true in the traditional Chomskyan approach, which presented the isolation of language not merely as a methodological assumption but as a matter of fact: language really is unique, so, as a matter of fact, there are no similarities to other cognitive abilities. But even Chomsky now questions this view (Fitch, Hauser and Chomsky 2006; Hauser, Chomsky and Fitch 2002), and the past decade has seen a great increase in the theoretical trend called 'cognitive linguistics' which explicitly rejects it, so maybe we are now moving towards what I believe will be a much more healthy period for linguistics (and maybe for psychology too). In this new order, linguists will allow psychological theories and findings to influence their theories of how language is organized.

To my mind, the most important example of this will involve the notion of **spreading activation**, a very basic notion in cognitive psychology which plays absolutely no part in most theories of language structure. It is true that this spreading of activation is a process, so it belongs clearly in a theory of performance rather than competence; but where it takes place is a structure, and that structure is what we all mean by competence – the permanent knowledge of language. Moreover, psychologists also agree that spreading activation interacts with longer-term activation levels that are sensitive to frequency and recency, so that frequent and recent items are relatively easy to access. Most linguists know these facts from psychology, but very few allow them to influence their thinking about language structure. This resistance may be based in part on the old idea that 'the lexicon' is different from 'the core' of language; so even if spreading activation is obvious in the lexicon, it may not be relevant to the core. This defence is undermined by the evidence for 'structural priming' which shows that even syntactic patterns activate each other (Branigan, Pickering, Liversedge, Stewart and Urbach 1995, Bock and Griffin 2000); but in any case the distinction between lexicon and core is itself very unclear and controversial even among linguists. Whatever the reason, it is a great pity that linguists have ignored spreading activation in this way, because it provides a crucial constraint on any theory of language structure: it must model language as a **network**. This conclusion is inescapable if the supporting model of language processing includes spreading activation and if activation can only spread in a

network; but it has been ignored by most linguists, with a very few exceptions (notably Lamb 1966, Lamb 1998).

Another important idea which is well established in cognitive science (especially in Artificial Intelligence) is **default inheritance**, the logic of ordinary reasoning which allows us to assume that something has its expected ('default') properties unless we have evidence to the contrary (e.g. Luger and Stubblefield 1993:387-9). This idea is simply common-sense and underlies every traditional grammar which contains not only general rules but also their exceptions; but its implications deserve far more attention than they normally get from theoretical linguists. After all, if the mechanism of default inheritance is available in ordinary reasoning, then (by default) we expect it to be available in all kinds of reasoning including language. And if it is available in language, it is at least a promising candidate for handling all sorts of contrasts that linguists have tended to handle in terms of very different mechanisms, from the 'elsewhere' condition of phonology (where 'elsewhere' defines the default) to transformations which change the default structure into a special one (Hudson 2003c). In this case the idea has certainly had some impact on general theories of language structure, but outside cognitive linguistics this impact is mostly found in theories of the lexicon (e.g. Pollard and Sag 1994:36). But what if the lexicon is just the most specific part of the general 'lexico-grammar' (Halliday 2002)? In that case default inheritance can also apply to general schematic constructions (as in Sag 1997).

Theoretical linguists could use the same defence against default inheritance as I suggested for spreading activation, namely that this is a matter of language use (performance), not competence. But once again the defence has the same fundamental weakness: the procedure of default inheritance has to apply to a structure in which there are 'inheritance hierarchies' (hierarchies of more and less general concepts, where less general concepts inherit from the more general ones above them). This being so, any theory of competence has to ensure that the structure of language includes the necessary hierarchies for inheriting, and to make them available not only in 'the lexicon' but also in 'the core'. Every theory includes some way of classifying elements in terms of both general and specific categories (often called 'features'), but not many theories provide the kind of consistent hierarchical classification which is needed to make default inheritance work smoothly.

Both spreading activation and default inheritance are widely accepted and used outside linguistics, but (rather surprisingly, to my mind) they are rarely combined in the same theory. This is especially surprising since default inheritance is a rather obvious solution to a widely recognized problem in network theories. One of the issues in the connectionist tradition of network modelling is precisely how to use a network to express generalizations and rules, and some researchers have identified this as a fundamental weakness of all networks (Browne and Sun 2001). The problem is not generalization as such; this can often be arranged as an automatic product of connectionist systems. Rather, it has two main sources. One is that most network theories have no mechanism for expressing properties that have variable reference – properties such as 'X has wings' (different birds have different wings) or (harder still) 'X suckles X's young'. The other weakness is the lack of any way of accommodating exceptions to general properties – in other words, of applying default inheritance. All that is needed, therefore, is a system that combines the virtues of network architecture with spreading activation and default inheritance.

Unfortunately, this is easier said than done, and my colleagues and I have spent a good part of the last decade trying to work out the details. Default inheritance

may be elementary common-sense, but the details are definitely where the devil is. Default inheritance is notorious among logicians for being messy and difficult, especially if the aim is an algorithm that is so clear that even a computer can understand it and mimic common-sense reasoning. After all, if any generalization may be overridden, then no inference is safe until the entire database of knowledge has been checked for potential exceptions – a recipe for disaster, especially in the real world of humans where speed is more important than absolute reliability. What is needed for survival in the real world is an efficient logic which gives the right answer first time – and better still, one which only provides relevant information. After all, given that both time and mental energy are limited, there's not much point in inheriting dozens of irrelevant facts along with the one or two relevant ones. This is a serious challenge for any theory of human reasoning. However this book offers a simple solution based in part on spreading activation and in part on the distinction between types and tokens. In a nutshell (which is expanded in section 0), default inheritance only applies to tokens, and only inherits active facts.

To return to the main point: if spreading activation and default inheritance apply to language, then any theory of language structure must accommodate them; and yet very few do. But the problem does not stop with these two phenomena. Elementary psychological theory also has a great deal to say about other parts of cognition, such as categorization and the structure of memory, which are highly relevant to linguistic theory. The logic is very simple: if language is a type of cognition, and we know that general cognition has property X, then we must assume that language also has property X unless we have good reasons for denying it. Of course there may in fact be good reasons to deny it, but the evidence had better be strong. In this book I argue to the contrary, that language is indeed just like other kinds of cognition.

Moreover, reversing the logic gives a useful heuristic: if language has property X, then it is worth looking for property X outside language too. After all, we probably know more about the structure of language than about the structure of any other human faculty, so it makes good sense to treat language as a 'window on the human mind'. Some properties of language are, of course, unique to language; for example, it is only in language that we find words or topicalization. But many of these unique characteristics are either true by definition (words are surely part of language by definition) or can be explained in terms of the functions for which we use language (topicalization is useful for communicating); and a surprising number of the remaining elements of language can in fact be matched quite easily outside language. (Even in syntax, it is easy to find non-linguistic analogues of word order, dependency and agreement.)

This book explores these very general ideas about language and cognition and tries to follow through their consequences for the theory of language structure. I am a linguist, not a psychologist, so language structure is my focus; and language use (and learning) and other areas of cognition are neighbouring territory – interesting and relevant, but ultimately not what I want to talk about. However, even within language structure we find the same tendency towards intellectual fragmentation, with each of the traditional levels of analysis (phonology, syntax and so on) attracting its own structural theories which might be based on completely different principles from neighbouring levels. The problems of this tendency are obvious, not least that sooner or later the levels will have to meet up. Here too, I have tried to develop a general theory which integrates all the levels into a seamless whole.

As far as language structure is concerned, most of my ideas – especially the good ones – come from other people. My contribution has been to select them and fit them together. For example, at the start of my career as an academic linguist I chose Halliday's ideas about sentence structure and Chomsky's on competence and on generative grammar. Since then, ideas have come from people as diverse as (in alphabetical order) Anderson, Bresnan, Bybee, Deacon, Fillmore, Huddleston, Jackendoff, Labov, Lakoff, Langacker, Levin, Levinson, McCawley, Pollard, Sadock, Sag, Slobin, Tomasello and Winograd. To some, this list will look like an intellectual mess, a recipe for chaos; but to me, it is a reservoir of brilliant insights which, I believe, belong in any theory of language.

One of my long-standing interests has been the interface between linguistics and education (Brookes and Hudson 1982; Hudson 1992; Hudson 2001c; Hudson 1981a; Hudson 2002; Hudson 2004b; Hudson and Walmsley 2005; Hudson 2001b). In my attempts to build bridges between linguistics and schools in the UK, I have tried hard to promote a general-purpose, theory-lite version of linguistics without bias towards any own theoretical preferences (and perhaps especially not towards my own). And conversely, I have never tried to defend Word Grammar in terms of its benefits for education; if it's true, this will emerge from the evidence, and if not, it's no use for teachers. This book will say nothing about school teaching, but I do believe that some of the issues I discuss here are crucial to education. In particular, education needs to know whether language is an innate faculty which simply needs to be 'triggered' or whether it needs to be learned from experience; and whether it is a list of vocabulary and rules, or a network (Hudson 2007b). I hope the book will make a small contribution to building the bridge that some of us have been working on for some time; but the bridge deserves a separate book all to itself.

One problem I haven't worried much about is the name of the theory. Is this really the same theory as the ones I described in 1984 and 1990, both called 'Word Grammar'? I don't know, just as I don't know whether I speak the 'same language' as the one Chaucer spoke. But I'm sure it isn't the same as the first theory I learned and worked on, 'Systemic Grammar' (Hudson 1971) – Halliday surely has the right to that name since he invented it. Nor can I call it 'Daughter Dependency Grammar', which is what I called the first theory I developed on my own (Hudson 1976a); after all, I no longer believe in grammatical 'daughters'. But since I started to use the name 'Word Grammar' in the early 1980s the package of ideas has changed at least as much as it did in the previous decade, so it is probably time for a slightly new name. Hence the startlingly original name in the title of this book: 'the new Word Grammar'. Like the contents, the label is half old and half new.

These changes would probably not have happened without the lively debates that occasionally erupt on the Word Grammar email list, so I want to thank the other participants in these (and other) discussions, and especially the following: And Rosta, Chet Creider, Eva Eppler, Geoff Williams, Haitao Liu, Jasper Holmes, Joe Hilferty, Mark P. Line, Matthias Trautner Kromann, Nik Gisborne, Sean Wallis and So Hiranuma. And, Mark, Haitao and Eva also gave me extensive and penetrating comments on all or parts of an earlier draft of the book, all of which have been acted on. I should also like to thank John Davey at OUP for his encouragement and patience during all those years when the book was 'just six months away'. If only!

Introduction

Conceptual Networks

Word Grammar (henceforward: **WG**) is a theory of language which touches on almost all aspects of synchronic linguistics and unifies them all through a single very general claim (Hudson 1984: 1):

(1) **The Network Postulate**: Language is a conceptual network.

This claim is not unique to WG and could even be described as a commonplace of modern linguistics. After all, we all see ourselves as successors to the early structuralists who saw language as 'a system of interdependent terms in which the value of each term results solely from the simultaneous presence of the others' (Saussure 1959). Any system of interconnected entities is a network under the normal everyday meaning of this word, so the structuralist legacy can be interpreted as the view of language as a network - a view that every modern linguist would surely accept, at least in contrast with the idea that a language is merely a collection of otherwise disconnected units. However, I suggest below that the network idea is actually quite controversial when taken seriously.

The modifier *conceptual* is not much more controversial. It is obvious that language is conceptual in the sense that it exists in the minds of individual people; this is what we mean by 'knowing a language'. Some linguists have emphasized that language has a social mode of existence in addition to this conceptual mode - as a 'social fact' (Saussure 1959), as a 'social phenomenon' (Sapir 1921) or as 'social semiotic' (Halliday 1978). This is equally obvious; after all, language is the foundation for society as we know it, and it is from others in our society that we learn our language. Indeed, I shall argue in section **Error! Reference source not found.** that it is impossible to separate language from the social relations between speakers and those with whom they interact, so I have considerable sympathy with the view that language is a social fact.

However, I also believe that social facts are relevant only to the extent that they are conceptual - i.e. only to the extent that they are known by individual people. In contrast, an extreme version of the social view is that 'our primary object of interest [is] the speech community', and 'the individual does not exist as a linguistic object' (Labov 2001: 34). This must surely be wrong – linguists often study the language of one individual to produce very successful descriptions. The only problem they face is in not being able to generalize from that individual to a whole 'community', but the notion of speech community is in any case highly contentious (Hudson 1996: 24-29). Moreover, the only way to study the language of a community is by first studying individual members, so the individual must be the primary object of study. If there are social patterns as well, they can be studied, but this research must build on the study of individuals. In short, I agree that our primary object of study should be 'I-language ... where 'I' is understood to suggest 'internal', 'individual' and 'intensional'.' (Chomsky 1995b:6)

The conclusions so far, then, are more or less uncontroversial:

• Language is a system of interconnected elements.

• Language is conceptual in the sense that it is 'in the mind', even if there is also a sense in which it is 'in society'.

But however bland it may seem at first sight, the idea of language as a conceptual network actually leads to new questions and highly controversial conclusions. The

words *network* and *conceptual* are both contentious. We start with the notion of language as a network. In WG, the point of this claim is that language is **nothing but** a network - there are no rules, principles or parameters to complement the network. Everything in language can be described formally in terms of nodes and their relations. This is also accepted as one of the main tenets of cognitive linguistics (Langacker 2000; Goldberg 1995; Lamb 1998), so WG fits very comfortably in this new tradition which has developed in parallel with WG. In WG, the whole of language has a uniform structure, and consists of abstract patterns which all share the same basic formal characteristics (though some are much more general than others). The same is true of the other theories in the cognitive linguistics tradition (Cognitive Grammar, Construction Grammar and Stratificational Grammar), and also of Systemic Functional Grammar, the theory from which WG ultimately derives (Hudson 1971; Halliday 1985).

For example, in WG the generalization which combines any finite verb with its subject is analyzed and described in the same way as the one which combines the verb *hit* with its object, though the former is much more general than the latter. This claim is very different from the view that rules belong in the grammar - the 'computational system' - while idiosyncratic facts belong in the lexicon, which is merely 'a set of lexical elements' (Chomsky 1995b:130). This radical split between rules and the lexicon is central to a lot of work in modern linguistics - for another example, consider Pinker's claim that in morphology regularities are handled by rules while irregular and semi-regular exceptions are handled by a lexical network (Pinker 1998). But it seems to have more to do with the traditional division of linguistic facts between grammar books and dictionaries than with any reality that can be observed in language. It creates an artificial boundary between 'general' and 'specific' where there is actually a continuous gradation, and generates more analytical problems than solutions. We shall consider one such example below.

The claim that language is a network therefore conflicts with the claim that information is divided between the grammar and the lexicon. In a network analysis, the same network includes the most general facts ('the grammar') and the least general ('the lexicon'), but there is no division between the two. Indeed, we shall see in section 0 that the network includes even more specific facts than the lexicon, namely unique uttered (or written) tokens of words (or other items of ongoing experience). I shall use the terms '**token**' and '**type**' with their established meanings, so types are stored and tokens are not; this contrast will play an important part in the theory. The idea that the network includes one-off tokens as well as permanently stored types is even more controversial, but it will turn out to be really helpful in explaining both how we process experience and how we learn from it. To summarize, therefore: There is no clear boundary between the network of 'the lexicon' and the rules of 'the grammar', nor between stored knowledge of types and temporary tokens.

Turning to the 'conceptual' part of the claim, this means simply that language is in the mind. It says nothing about how language gets there, and on this question too the WG answer is controversial: very little of language is innate, so almost all is learned from experience. This is the standard answer in cognitive linguistics, where language is assumed to be 'usage-based' (Barlow and Kemmer 2000), and it also attracts strong support in computational linguistics (Bod 1998); but it is diametrically opposed to the 'nativist' idea that most of language ('universal grammar') is innate. The debate is in part about learning mechanisms and other psychological questions which may be outside the scope of linguistics; but it also has major implications for purely linguistic theory.

For example, if the basic structures of language are already in place at birth, or develop automatically soon after, all the learner has to do is to set parameters and fill in a lot of lexical items according to a standard template. The result will be free of redundancy because the general patterns exist before the details are registered and need not be stored twice. In some sense, language will be 'perfect'. The usage-based view is very different. If language is induced from experience, all the details are stored before the general patterns become apparent so there is no way to avoid redundancy. The resulting knowledge will be very rich, very redundant and very 'messy'. This is not to deny the general patterns or their clarity, so for example a usage-based grammar will still include the very clear rule of English that requires finite verbs to have subjects. Such generalizations are an important part of language; but so are the myriad idiosyncratic and sometimes irregular details about particular words and constructions. For example, the English passive is normally realized as a passive participle, but exceptionally it may be realized as a present participle just in case it is the complement of a verb that means 'need' (i.e. NEED, REQUIRE, WANT):

(2) This pot needs cleaning.

A theory of language structure must accommodate such messy details as well as the broad generalizations. In short, language is mostly learned (rather than innate), and the learning process combines massive storage of examples with induction of generalizations. Consequently, the end state contains a great deal of redundant detail as well as high-level generalizations.

Networks turn out to be convenient for modelling this spectrum of information which ranges from fine detail to broad generalization precisely because there is no clear dividing line between the two. For instance the pattern of *needs cleaning* in (2) is idiosyncratic, but it also goes well beyond any one lexical item so is it a rule or a lexical fact? In the absence of general principles, most of us would prefer not to choose at all. In contrast, a uniform network analysis accommodates general and particular facts in the same way, so it forces no choice.

I hope to have shown that the conceptual-network idea is not merely a matter of our choice of metaphors for thinking about language or what kinds of diagram we draw. It also has important consequences for the theory of language structure, such as the supposed split between the grammar and the lexicon. However its importance goes well beyond questions about the internal architecture of language, because it raises even more basic questions. We shall consider five:

Question 1. Is language different from other kinds of cognition?
Question 2. Is language separate from other kinds of cognition?
Question 3. Is there a specialized short-term memory system for language processing?
Question 4. If language is a network, what kind of network is it?
Question 5. Is the network of language distributed or local?

The point of the discussion is not so much that we can already answer these questions satisfactorily, but rather that we can ask them and find relevant evidence.

Question 1. Is language different from other kinds of cognition? How does the language network fit into general cognition? It is a commonplace of cognitive psychology that long-term memory is a network (Reisberg 1997:257), though it is a matter of dispute whether this network is symbolic, with one node per concept, or distributed, with each node represented by a particular setting of connection weights across the entire network (ibid: 292). (I shall return briefly to this question below.)

Similarly, computer models of general knowledge often analyze it as a 'semantic network' (Luger and Stubblefield 1993:35). If language is a network, then we can compare its network with the networks that are found in other areas of knowledge in order to decide whether it is basically the same or different. But if language contains structures of a type that is only found in language, obviously this question does not even arise.

The question is not whether the language network can be distinguished from the network for our knowledge of people, places and so on. It is different by definition: what we mean by 'the language network' is, put simply, our knowledge of words and their properties. (This is why the theory is called 'Word Grammar'.) Rather the question is whether the networks for words are different kinds of networks from those that we use for storing our knowledge of people, places, experiences and so on. The null hypothesis is presumably that there are no differences, so what we need to look for are potential differences. Are there any features of language which are unique to language? For example, are there any general link-types which are only found in language? Does a language network have architectural characteristics which are special to language? Ultimately, of course, these are questions for those who know about other kinds of knowledge but in the meantime it is possible for a linguist to assemble informal evidence, and my tentative answer is that every apparent peculiarity of language turns out to have a close analogue outside language. However, the main point is not the answer, but the fact that the question can even be asked. One of the purposes of this book is to highlight the similarities between language and other kinds of knowledge, as I did in earlier work (Hudson 1984:37-39; Hudson 1990:53-83). My tentative conclusion, therefore, is that knowledge of language is very similar to other areas of knowledge in terms of its organization and even in some of its general analytical categories.

Question 2. Is language separate from other kinds of cognition? In other words, is language a distinct 'module' of the mind? This question is closely related to the previous one, since the clearest evidence for a separate module would be distinctive organizing principles; but even if (as I have just argued) language has the same organizing principles as other kinds of cognition, it might still be stored separately, giving a version of **modularity**. For example, when Fodor argues that language-perception is a distinct mental module, his main evidence is not its internal organization but rather what he calls 'information encapsulation' (Fodor 1983) – the property of not being influenced at all by information which is available elsewhere in the mind. It is very clear that some areas of experience are encapsulated in this way, and immune to general knowledge; a clear example from everyday experience is the effect on us of a stationary escalator (a staircase which normally moves): however clearly we can see that it is stationary, we still stumble awkwardly when we get on because we 'expect' it to be moving. On the other hand it is equally clear that our perception of language is heavily influenced by higher-level factors such as the phonological contrasts of our language (Harley 1995:222), and that interpretation at higher levels is driven by contextual information as well as by bottom-up perceptual input (ibid:225).

One conclusion is that 'it may be that we have to rethink the concept of module and allow for a kind of continuum, from peripheral perceptual systems, which are rigidly encapsulated (not diverted from registering what is out there), through a hierarchy of conceptual modules, with the property of encapsulation diminishing progressively at each level as the interconnections among domain-specific processors

increase.' (Carston 1997:20) Another possible conclusion is that it is wrong to think in terms of modules, and that instead we should be looking for a network model of cognition in which some defaults are much harder to override than others – for example, in the case of immobile escalators maybe we cannot override the default motor-programme that we associate with escalators.

Another kind of supposed evidence for modularity comes from neuro-psychology, where it is often suggested that some areas of the brain are dedicated exclusively to language. If this were the case, then these areas would define the language module physically. However, the neurological evidence in fact seems to suggest the opposite: 'The traditional theory equating the brain bases of language with Broca's and Wernicke's neocortical areas is wrong. Neural circuits linking activity in anatomically segregated populations of neurons in subcortical structures and the neocortex throughout the human brain regulate complex behaviors such as walking, talking, and comprehending the meaning of sentences. When we hear or read a word, neural structures involved in the perception or real-world associations of the word are activated as well as posterior cortical regions adjacent to Wernicke's area. Many areas of the neocortex and subcortical structures support the cortical-striatal-cortical circuits that confer complex syntactic ability, speech production, and a large vocabulary. However, many of these structures also form part of the neural circuits regulating other aspects of behaviour.' (Lieberman 2002:36)

Modularity creates far more theoretical problems than it solves. As Tomasello puts it: 'The major problem for modularity theories has always been: What are the modules and how might we go about identifying them?' (Tomasello 1999:203) For example, should we think in terms of separate modules for the traditional 'levels' of phonology, morphology, syntax and meaning, or in terms of one module for the lexicon (which contains information from all the levels) and another for the general rules of grammar? It is true that research has revealed strong tendencies for particular kinds of information to cluster together in the brain or to be injured together in pathology, but this is exactly as predicted in a non-modular, network-based account: nodes that are neighbours are more likely to be located near each other and to be affected by the same traumas than nodes that are distantly related. But these tendencies are a far cry from the absolute module-wide patterns that we should expect if modules are like boxes which are located and affected in their entirety. My conclusion, therefore, is that the language network does not occupy a distinct part of the human mind or brain, but is intimately embedded in the general cognitive network.

Question 3. Is there a specialized short-term memory system for language processing? Another basic question about conceptual networks concerns the theory of memory. A traditional view, which has had some influence in linguistics, is that our minds contain two separate kinds of memory, long-term and short-term. Long-term memory is our linguistic competence, and has one kind of structure, be it rules and lists, a network or whatever. Short-term memory, on the other hand, is a kind of work-bench with a very different structure, onto which we copy material from long-term memory in order to combine it with incoming data such as a sentence that we are currently trying to understand. It is our short-term memory, for example, that we use to hold arbitrary lists of numbers and which has a limited capacity (the famous 7 ± 2 of Miller 1956). It is a short step from this idea to the idea that we have distinct short-term memories for different tasks, including a special one for processing language; so a great deal of psycholinguistic work has been devoted to exploring the structure of

this supposed area of the mind in terms of syntactic parsers and phonological buffers with very specific characteristics. For example, the syntactic parser might be unable to cope with more than two constituents under the same syntactic relation (e.g. subject within subject) (Lewis 1996), and the phonological buffer might only be able to hold up to two seconds worth of speech (Baddeley and Logie 1999). A recent version of this general approach is that ‘linguistic working memory’ is a ‘workbench’ or ‘blackboard’ with three separate divisions for handling phonology, syntax and semantics (Jackendoff 2002:200).

On the other hand, more recent work on memory has suggested that 'working memory' (the preferred term for short-term memory) ‘consists of a set of processes and mechanisms and is not a fixed 'place' or 'box' in the cognitive architecture. ... its contents consist primarily of currently activated LTM representations ...’ (Miyake and Shah 1999:450). This idea has been promoted by some leading psychologists (Cowan 1997; Cowan 1999; Ericsson and Kintsch 1995; Ericsson and Delaney 1999). In other words, maybe there really is only a single memory, the network of long-term memory, and working memory is just the 'working' part of this network. (As I mentioned earlier, this network actually includes not only permanent long-term nodes but also some temporary short-term nodes.) If this is so, then all our models of the mechanism for processing language need to be rethought to the extent that they depend on specific work-bench structures. I shall return to this question in section 0, where I shall present the outlines of a WG theory of processing. Meanwhile the tentative conclusion is that there may not be a separate 'work-space' for processing language (or anything else).

Question 4. If language is a network, what kind of network is it? Do all nodes have the same status? Are the links differentiated from one another in any way? If so, what kinds of links are there? We shall consider all these questions, and come to the conclusion that language is a network in which all concepts, including relations, are richly classified. This is probably the most distinctive claim of WG, but the question to which it is an answer simply does not arise in a more conventional approach to language. Even more interestingly, we can try to fit language networks into the typology of networks that has recently been discovered in graph theory (see Barabási 2003 for graph theory, and Chipere 2003:28-31 for its relevance to language). For example, are links distributed randomly among the nodes, or are there 'hub' nodes which have far more links than others? In technical terms, is language a ‘random’ network or a ‘scale-free’ network? This is a quantitative question which can only be answered by counting the number of links per node; in a random network the distribution of links shows a bell-shaped curve in contrast with the power-law distribution found in scale-free networks. Recent work on existing databases has suggested that language is scale-free (Ferrer i Cancho, Solé and Köhler 2004; Ferrer i Cancho and Solé 2001; Solé 2005), but we need studies on more theoretically sophisticated databases before we can be sure of the conclusion.

Another quantitative question is whether nodes are linked more or less evenly across the entire network, or whether there are sub-networks where the links among the members are denser than those to non-members (so-called 'hierarchical modularity'). At present we have nothing approaching a full network grammar of a language so we cannot even start to answer these questions, but we can at least look forward to the day when we shall be able to. Meanwhile we may be able to learn from small-scale experimental computer network models of processes such as vocabulary attrition (Meara 2002).

Question 5. Is the network of language distributed or local? This is the question about connectionism which I touched on earlier. Is knowledge represented locally, with a separate node for each concept, or globally, with each concept distributed across all the nodes. In other words, is it a symbolic network or a connectionist network? On this question, my impression is that linguists answer with one voice (Bybee 1995; Bybee 1998; Corbett and Fraser 1993; Croft and Cruse 2004; Culicover 1999; Givón 1998; Goldberg 1995; Lamb 1998; Langacker 2000); see, for example, the trenchant criticisms in Lamb 1998:2. We all agree that, if language is a network, the network is symbolic rather than distributed; in other words, one node represents the word *dog*, another node represents each sound, and so on. Indeed, it is hard to imagine doing linguistics (as we know it) without this assumption. A symbolic network allows us to explore the structure of the network and challenges us to think clearly about the details; in short, it is a good tool for research on linguistic structure. But a distributed network has none of these attractions; it may be able to learn simple correspondences such as between verb bases and their past tense forms (Rumelhart and McClelland 1986), but what is learned is just a table of numbers - no help at all in understanding the structure of this part of English grammar.

To summarize this section, WG is based on the Network Postulate that language is a conceptual network, a system of interconnected elements in the mind without any clear boundary between the network of 'the lexicon' and the rules of 'the grammar'. It is mostly learned (rather than innate), and the learning process combines massive storage of examples with the induction of generalizations, with the result that the end state contains a great deal of redundant detail as well as high-level generalizations. The network for knowledge of language may be very similar to other areas of knowledge in terms of its organization and even shares some of the same general analytical categories; and there may not be a separate 'work-space' for processing language (or anything else). In terms of its formal properties, the network is symbolic rather than distributed and all nodes and links are classified. All these claims will be developed in later sections.

Classification and the Isa relation

What then can we say about the language network? What kind of network is it? We can now start to enter into questions of detail. We already know a great deal of detail because most of what any linguist knows about language can be translated quite easily into network notation. Later chapters will give network analyses for significant areas of morphology, syntax and semantics, and will include a lot of the well-known facts about language that have emerged over two thousand years of study. What we know about the structure of language is far more detailed and highly structured than our research-based knowledge in any other area of human cognition, so we can treat language as a particularly clear window into human cognition.

One very clear conclusion is that links are of different 'types' according to the kind of relation that they represent: some links show class-membership, others show part-whole relations, and so on. In other words, we are not dealing with mere associative networks in which all links have the same status and the same meaning. For example, the significance of a class-member relation is quite different from that of a part-whole relation, and a word's sense is different from its grammatical subject and from its morphological realization. Moreover, links are all directional, so that their significance varies according to which end of the link is under consideration: for example, in *John snores*, *John* is the subject of *snores* but not vice versa. To a linguist

most such distinctions are obvious and completely uncontroversial, and what we miss in the distributed connectionist networks mentioned above.

What kinds of links are there? This is not a matter of logic or philosophy, but of linguistics and ultimately of psychology: what kinds of links does a working linguist need in order to analyze linguistic competence? The following generalizations are based on my own experience of descriptive analysis, and are fundamental to WG theory, but of course they are as tentative as any other theoretical generalization.

One relation stands out from all the others as particularly fundamental: the **Isa relation** used in classification, as in 'Dick isa Linguist' or 'Penguin isa Bird'. This relation and its name are familiar from the 'semantic networks' of early Artificial Intelligence (Reisberg 1997:280), but of course it is also one of the ordinary meanings of the verb *be* (as in *Dick is a linguist*) and it underlies any thesaurus or ontology. It is hardly necessary to stress the importance of this relation. As the basis for all classification, it is also fundamental to all generalization. For example, anything we know about Bird generalizes to anything which isa Bird - in other words, to any particular bird or type of bird. This process of generalization is '**inheritance**', which I discuss below. Inheritance plays such a fundamental part in all conceptual networks that I shall call these networks **'inheritance networks'**. In short, these networks allow generalizations thanks to the links which are labelled 'isa'.

I now have three brief comments on terminology and notation.

• The usual name for this relation is 'isa', which works well in simple cases such as '*and* isa Conjunction', but raises grammatical problems when used in sentences where ordinary grammar demands a form other than *is*. I have tried a number of alternative solutions (such as 'is-a', 'are-a', 'be-a'), but the most popular one seems to be to use *isa* even where other forms such as *are, was* or *be* would be expected; so with regret I shall write such things as 'Penguin and Sparrow isa Bird' and 'the subject must isa noun'.

• My practice in naming concepts in the text is to give the name a capital letter, as in ordinary English. Thus Penguin is the name for the category 'penguin', and Noun means 'the category noun'. (I make an exception for words, where the usual italics signify the name; so *penguin* means 'the word *penguin*'.) When I use category names as common nouns, of course, I do not give them capital letters: 'a noun is a word'. I shall argue below that relations themselves are also concepts, so I shall follow the same practice in naming them; thus the Isa relation will be called 'Isa'. Diagrams are obviously different from the sentences of this text, so capital letters are unnecessary.

• Isa has a standard notation in WG diagrams: a small triangle whose base is next to the super-category and whose apex is connected to the sub-categories. (The triangle is iconic: the base is larger than the apex, as the super-category is larger than the sub-categories.) The line may point in any direction, so all three diagrams in Figure 0.1 are equivalent.

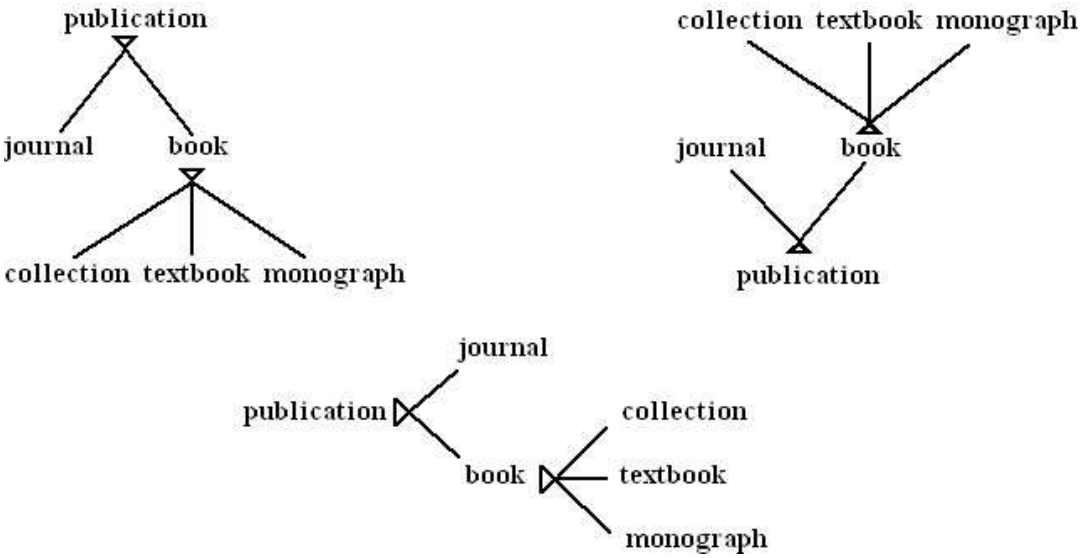


Figure 0.1: Three equivalent Isa diagrams

A WG network is built round a 'skeleton' of Isa relations because every node is involved in at least one such relationship. Most nodes, of course, isa some 'higher' node (taking 'higher' in its metaphorical sense rather than literally in terms of the diagrams; as we have just seen, a super-ordinate node may appear below its subordinates in a diagram). And similarly, most nodes are super-categories for other nodes. Of course Isa hierarchies must have a top node, but it is possible that every hierarchy leads to the same super-node, the node shown as a dot in Figure 0.3 below; this is merely speculation given the present state of research. However what does seem clear is that every other node is classified by at least one Isa link to a super-category. This claim follows, in fact, from the WG theory of processing and learning (see sections 0 and 0), since the very existence of a node presupposes some classification. The only possible exceptions are nodes which are innate, but if there are innate nodes, most of them must surely play an important role in classification.

The Isa skeleton is much more complex than a mere hierarchy because one node may isa more than one other node. This multiple membership is part of everyday life; for example, Dog isa Pet as well as Mammal, and each of us isa many different super-categories. For example, I myself isa Man, Brit, Linguist, Cyclist and Londoner. Multiple Isa relations are also commonplace in language; for example, the lexeme *attempt* isa Verb, English word and Formal word, and the inflected word *attempts* isa this lexeme and Present singular. In general, these separate super-categories carry orthogonal (i.e. independent) properties, but they can conflict and when they do the conflict cannot be resolved except by fiat; this, I suggest, is why we cannot say **I amn't* although we know perfectly well what it would be if we could say it (Hudson 2000a). Figure 0.2 shows WG diagrams for the examples just quoted:

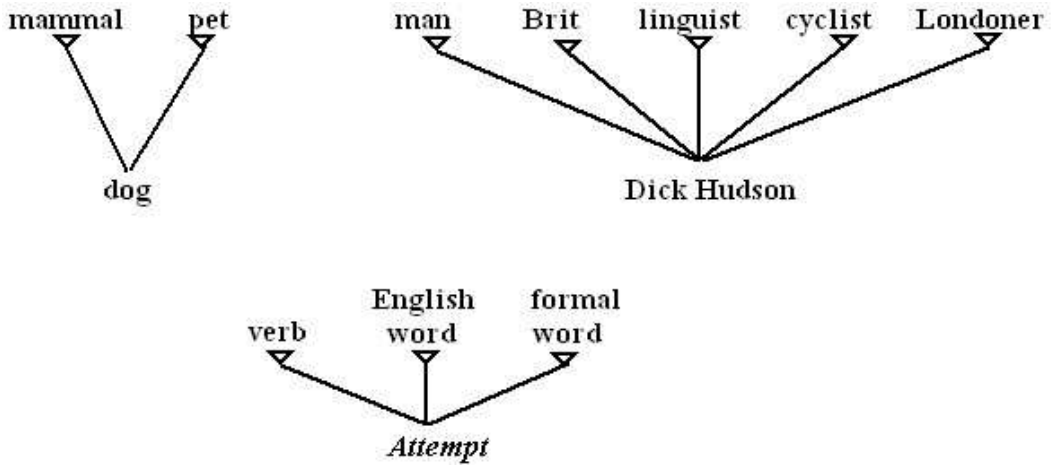


Figure 0.2: Isa hierarchies showing multiple membership

So far, then, we have identified just one basic relation: Isa. This relation has its own notation (the triangle) in WG diagrams, and its own logic (default inheritance, to be discussed in section 0 below). I shall introduce four other similarly basic relations in later sections: Argument and Value (later in this section), Quantity (section 0), and Identity (section 0). All the other links are treated in a different way from these primitive relations. In WG, these relations are themselves concepts, whereas the primitive relations are probably not; for example, they might be manifested neurologically by distinct neuron types rather than by distinct relations to other concepts. The WG ontology (i.e. its classification of concepts) probably includes something like the hierarchy in Figure 0.3 in which Relation is contrasted with Entity at the top level (Hudson 1990:76). (As expected, it is difficult to find natural-language names for some nodes at this level, so the top node has no name; but this does not matter because we shall see below that names are not important.)

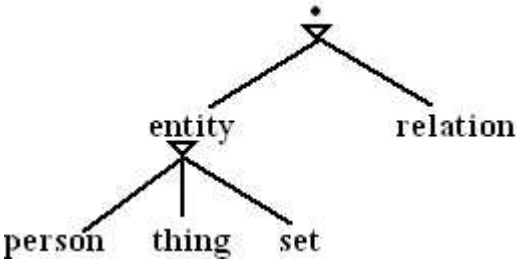


Figure 0.3: The top of the ontology

In addition to the basic Isa relation, then, we also recognize a multiplicity of more specific relations ranging from very general (e.g. Part) to very specific (e.g. Beak). Figure 0.4 shows two specific relations which link the typical bird to its beak and its tail. The number 1 is explained in the next section, but in a nutshell this diagram shows that a typical bird has a beak and a tail. It is surprisingly hard to find distinctive terminology for relations because nouns in English (and perhaps in all languages) tend to refer to entities rather than to relations, and to do this even if the entity is defined by its relation to some other entity. Take the word *father*, for example, a clear example of a relational noun: a father is a person, not a relation, although the particular person is picked out by their relation to someone else. Strictly speaking, therefore, Father isa Person, not Relation; and yet 'father' is the obvious

name for the relation. Even more confusingly, non-relational nouns such as *hand* or *car* are often used relationally as in *my hand* or *Mary's car*; each one picks out a particular object on the basis of its relation to some other person or object. Once again the hand or car is an object and not a relation, but the relation needs a name and it is tempting to lend it the object's name. For example, this naming system would give the label 'car' to the link from Mary to her car. Similarly, the relation between a bird and its beak is called 'beak', which of course is different from the label 'Beak' on the node for the general category of beaks. This potential ambiguity of labels between relations and entities is harmless because networks use arcs for one and nodes for the other, but in any case I shall explain shortly that the terminology is simply a matter of convenience, so nothing theoretical hangs on it at all.

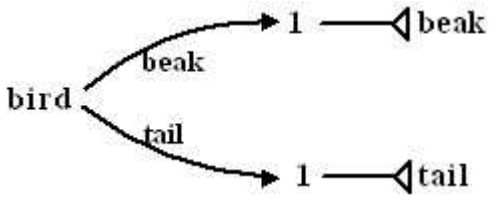


Figure 0.4: A bird has one beak and one tail

It is easy to see that relations themselves must also be concepts because we sometimes have ordinary non-technical names for them such as *friendship*, *distance*, and, of course, the word *relation* itself. If the sense of a word is a concept, then these relations must be concepts. Of course, a relation is fundamentally different from the other kind of concept, Entity, in that it must relate two entities, but there are also important similarities.

One of these is that, like entities, relations can be classified; so in everyday life we recognize a variety of relations between people - family relations, work relations, personal relations and so on - as well as spatial, temporal and causal relations. Similarly, grammarians have for a long time recognized a hierarchy of syntactic relations in which, for example, Complement subsumes Object and Predicative. The natural conclusion is that relations, just like entities, must also be organized in isa hierarchies. The importance of this conclusion cannot be exaggerated, because it solves the well-known analytical problem of relations: 'If each type of relation is represented by a specific type of associative link, then we risk losing the simplicity of the network idea and thereby render the whole proposal less attractive.' (Reisberg 1997:280).

The usual approach to this problem is to assume that the list of possible link-types is given in advance, and that the list is finite and hopefully quite short. But this hope is dashed as soon as we start considering even simple examples such as a bird's parts. The fact is that a bird's relation to its beak is quite different from its relation to its tail; each part has a distinct location within the bird, and even more importantly it has a distinct use. We cannot simply talk about parts, but must refer more specifically to the bird's beak and its tail in order to formalize even simple statements like (3) in which *a bird's beak* and *its head* invoke different relations.

(3) A bird's beak is attached to its head.

But if this is so, there is little hope of finding a limited, or even finite, set of pre-determined relations.

The solution is based on the fact that precisely the same problem arises with entity concepts: where does the list ‘come from’? It is generally accepted that at least most of these concepts are not drawn from a pre-defined list, but are learned from experience. Given the diversity of human experience, we predict an open-ended variety of entity concepts which are held together conceptually by isa hierarchies (and other links). The solution to the problem of relations is to apply the same kind of treatment to the non-primitive relations that link entities, and the result is an open-ended hierarchy of relations. Similar suggestions have been made before for limited domains such as semantic cases (Charniak 1981) and grammatical functions (Bresnan 2001:97, Hudson 1990:189-218), but so far as I know the idea that all relations are classified is unique to current WG. It is clearly controversial, and if true it is important. In short, our fundamental Isa relation applies not only to nodes, but also to non-primitive links. Figure 0.5 shows how this claim applies to the earlier example of bird-parts by expanding Figure 0.4 to show that the relations Beak and Tail both isa Part.

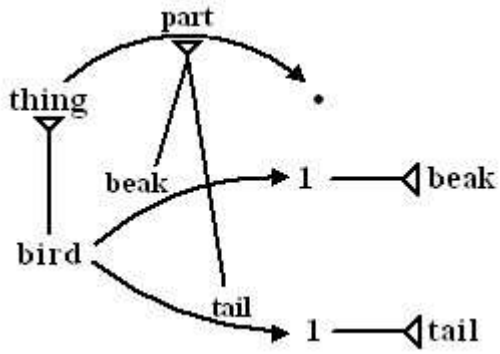


Figure 0.5: Beak and Tail isa Part

Classified relations appear to put the networks with which we are dealing onto a higher formal plane than the networks that are usually discussed in the literature. We might call them 'second-order' networks because the links are themselves interrelated in a separate network of Isa relations. This is a major change in the logical and formal status of networks which makes the whole network idea less attractive. After all, if we can now show that cognition is a second-order network, maybe next year we shall find evidence for third-order networks and so on and on ad infinitum? Every extra order that we discover implies more computing power in the mind, and one thing that is certain is that computing power is limited, so theories about higher-order networks require careful consideration.

There is in fact an alternative theory of relations which assumes a less obvious answer to the question whether relations are network links or nodes. The obvious answer is that they are links; this is what I have assumed in the discussion so far, and indeed I shall pretend to assume it in the rest of this book. But the alternative treats relations as nodes, just like entities; for example,the relations Part, Tail and Beak are not represented by arrows, as in Figure 0.5, but by nodes just like those for the entities Bird, Tail and Beak.

One advantage of this analysis is to explain the similarities between relations and entities that I discussed above, and in particular to explain how it is that relations can be classified and learned just like entities. Furthermore, this analysis is more consistent with the diagram of the top of the ontology (Figure 0.3) in which Relation

and Entity, as sisters, have the same status. Given this analysis, the Isa hierarchy of relations is of the same order as that for entities, so there is no need to worry about second-order networks.

But, of course, the price paid for these benefits is that we have no links (except Isa links). For example, if the relation Part is a node rather than a link, then it obviously cannot link entity nodes to each other. And yet we know that something links entities, and indeed that their distinctiveness depends entirely on the distinctiveness of their links. The solution is to introduce yet more links, but this time primitive links (like Isa). As primitives, they have properties that are 'built in' rather than inherited via an Isa hierarchy, and these properties are exactly the same for every link. Moreover, like Isa they are directed (so 'A isa B' is different from 'B isa A'), and they control the logic of inheritance. The obvious names for these new links are **Argument** and **Value**, so the two facts 'Part Argument Thing' and 'Part Value X' combine to express the fact that a thing's part is X which, in the previous system, would have been expressed by a single fact: 'Thing Part X'. In other words, the solution to the problem of relations is that there are, in fact, very few true relations: just a handful of primitives (Isa, Argument and Value, plus two others to be introduced below). However there is also a large collection of relational nodes (e.g. Part and Beak) which function as 'pseudo-relations' thanks to their Argument and Value links to entity nodes and which can grow without limit thanks to the ordinary processes which are responsible for the learning of entities.

Decomposing every non-primitive relation into a node plus two primitive relations may provide a satisfying theory, but it multiplies the problems of diagramming. Even if we use obvious abbreviations for 'Argument' and 'Value' ('of' for Argument and '=' for Value), quite simple networks become unreadable. For example, it would be hard to expand Figure 0.6, which just shows how the entities Bird and Beak are related via the relational node Beak. The diagramming complications come from the Isa, Argument and Value links to the relation nodes, so the rest of this book will ignore these links in most diagrams and pretend that each relation corresponds to a single arrow whose classification is shown just by the label attached to it.

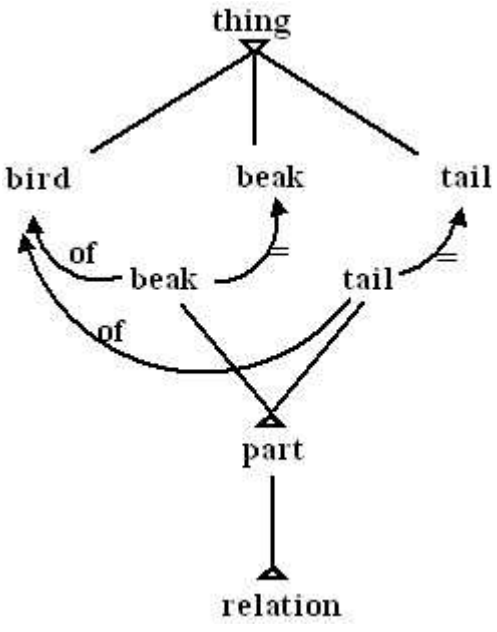


Figure 0.6: Birds, beaks and tails with relations as nodes

The discussion in this section raises important questions about the mental resources that a mind needs in order to handle a cognitive network. In a simple associative network, the basic unit of thought consists of two nodes (A and B) connected by a simple relationship, R: 'A R B'. This constitutes a 'fact', so manipulating a fact in this network would involve just three cognitive units and holding this fact in working memory would take just three units of mental resources. In an inheritance network as defined here, the relation R itself has an Isa link to a super-relation R+, so this figure rises to at least five: two nodes (A and B), two relations (R and R+) and one Isa link (between R and R+). (The figure could be higher, given the possibility of multiple Isa links between any one of the nodes and super-categories.) The idea that cognition is an inheritance hierarchy may raise fundamental questions for comparative psychology; for example, are non-human animals capable of creating inheritance hierarchies? If our uniqueness lies in our ability to conceptualize symbols (Deacon 1997), is this because only we are able to learn relation-types (such as the relation Meaning, which I discuss in more detail in **Error! Reference source not found.**)? Section **Error! Reference source not found.** considers these questions in more detail.

Another important consequence of accepting inheritance hierarchies is that a network consists of **nothing but** nodes and links; the labels that we put on either nodes or links are simply mnemonics for our own purposes, and have no theoretical status whatsoever (Lamb 1966; Lamb 1998:59). For example, Bird is uniquely defined by its relations to nodes such as Beak and Wing, so the label 'Bird' is redundant; and likewise for every other label, provided the network is firmly 'anchored' to external units such as perceptual categories. Indeed, both Figure 0.4 and Figure 0.5 contained nodes that had no label (except a dot or a 1) which illustrate the point well. For example, the dot in Figure 0.5 is defined as the typical part by its Part relation to the super-general category Thing, so the label 'part' would have been redundant; and similarly the two nodes labelled '1' are uniquely defined by their relations as the typical wing and tail. From a theoretical point of view, then, we could in principle remove all the labels for relations and rely entirely on the isa hierarchies

that relate them to one another, though the practical value of such diagrams would be close to nill.

To summarize this section, I have argued that language networks, and more generally human conceptual networks, consist of nodes and links. The links are all of three primitive types: Isa, Argument and Value (with two more to be introduced later), and the nodes include relations as well as entities. (But, to simplify the diagrams and the discussion, I shall reduce a relational node plus its Argument and Value arrows to a single arrow.) Every node (except one) isa at least one other node, and every entity node is the argument or value of at least one relation node.

Quantity, optionality and 'variables'

The only primitive relation that will figure in further discussion is Isa. Another relation that early AI workers also considered very basic is what they called 'hasa', as in 'Book hasa Title' or 'Bird hasa Beak' (Reisberg 1997), but this is actually very different from Isa. Any 'hasa' statement is really just a way of counting relata (whatever is picked out by the relation). For example, if we say that a bird has a beak, we are asserting the existence of one beak per bird; if we deny it, we are asserting that the relevant number is 0; and if we say it has two wings, our claim is that there is one two-member set. In contrast, Isa is not dependent on any other relation and does not involve either an existential claim or a numerical one; it is simply about class-membership. In other words, 'has-a', unlike Isa, combines two separate bits of information: a relation (e.g. 'beak-of', 'part-of'), and a quantity, which may be a number (1 or 0) or a range (>0, ≤1, any number). In an earlier version of WG (Hudson 1990:16), I did treat 'has' as a basic relation which always combined with a 'quantifier' such as 'a' (= 1) or 'ano' (= a or no, i.e. either 1 or 0). However, current WG dispenses with the 'has' relation altogether. In diagrams, it sometimes (but not always) uses the quantity as a label for the node concerned; for example, Figure 0.4 shows that the number of beaks for a bird is 1. A simple dot shows that the number is unconstrained - i.e. either that the relatum is optional and may be multiple, or that the quantity is inherited from elsewhere.

But however convenient this notation may be when drawing networks, it is no more than a notational trick. If labels are basically mere mnemonics, as I claimed at the end of section 0, then it is wrong to pack so much information into labels. To be theoretically pure, these numerical labels should be replaced by separate relations, so we must consider this underlying reality.

Take the example of a bird's beak. The network must show not only that this node isa beak, but also that it is an obligatory part of the bird's anatomy – in other words, every bird must have precisely one beak. The WG solution is to recognize numerical quantities as entities with the same properties as other entities. Thus '1' is an entity which is related, inter alia, to the value of the 'beak-of' relation (i.e. x in Figure 0.7). The relation between a numerical quantity and another entity is '**quantity**', represented in diagrams (when needed) by an arrow labelled '#', and implied by a number (0 or 1) standing in for the node itself. This is another primitive relation, like Isa, Argument and Value.

The discussion in this section raises important questions about the mental resources that a mind needs in order to handle a cognitive network. In a simple associative network, the basic unit of thought consists of two nodes (A and B)

connected by a simple relationship, R: ‘A R B’. This constitutes a ‘fact’, so manipulating a fact in this network would involve just three cognitive units and holding this fact in working memory would take just three units of mental resources. In an inheritance network as defined here, the relation R itself has an Isa link to a super-relation R+, so this figure rises to at least five: two nodes (A and B), two relations (R and R+) and one Isa link (between R and R+). (The figure could be higher, given the possibility of multiple Isa links between any one of the nodes and super-categories.) The idea that cognition is an inheritance hierarchy may raise fundamental questions for comparative psychology; for example, are non-human animals capable of creating inheritance hierarchies? If our uniqueness lies in our ability to conceptualize symbols (Deacon 1997), is this because only we are able to learn relation-types (such as the relation Meaning, which I discuss in more detail in **Error! Reference source not found.**)? Section **Error! Reference source not found.** considers these questions in more detail.

Another important consequence of accepting inheritance hierarchies is that a network consists of **nothing but** nodes and links; the labels that we put on either nodes or links are simply mnemonics for our own purposes, and have no theoretical status whatsoever (Lamb 1966; Lamb 1998:59). For example, Bird is uniquely defined by its relations to nodes such as Beak and Wing, so the label ‘Bird’ is redundant; and likewise for every other label, provided the network is firmly 'anchored' to external units such as perceptual categories. Indeed, both Figure 0.4 and Figure 0.5 contained nodes that had no label (except a dot or a 1) which illustrate the point well. For example, the dot in Figure 0.5 is defined as the typical part by its Part relation to the super-general category Thing, so the label ‘part’ would have been redundant; and similarly the two nodes labelled ‘1’ are uniquely defined by their relations as the typical wing and tail. From a theoretical point of view, then, we could in principle remove all the labels for relations and rely entirely on the isa hierarchies that relate them to one another, though the practical value of such diagrams would be close to nill.

To summarize this section, I have argued that language networks, and more generally human conceptual networks, consist of nodes and links. The links are all of three primitive types: Isa, Argument and Value (with two more to be introduced later), and the nodes include relations as well as entities. (But, to simplify the diagrams and the discussion, I shall reduce a relational node plus its Argument and Value arrows to a single arrow.) Every node (except one) isa at least one other node, and every entity node is the argument or value of at least one relation node.

Quantity, optionality and ‘variables’

The only primitive relation that will figure in further discussion is Isa. Another relation that early AI workers also considered very basic is what they called 'hasa', as in 'Book hasa Title' or 'Bird hasa Beak' (Reisberg 1997), but this is actually very different from Isa. Any 'hasa' statement is really just a way of counting relata (whatever is picked out by the relation). For example, if we say that a bird has a beak, we are asserting the existence of one beak per bird; if we deny it, we are asserting that the relevant number is 0; and if we say it has two wings, our claim is that there is one two-member set. In contrast, Isa is not dependent on any other relation and does not involve either an existential claim or a numerical one; it is simply about class-membership. In other words, 'has-a', unlike Isa, combines two separate bits of information: a relation (e.g. ‘beak-of’, ‘part-of’), and a quantity, which may be a number (1 or 0) or a range (>0, ≤1, any number). In an earlier version of WG (Hudson

1990:16), I did treat 'has' as a basic relation which always combined with a 'quantifier' such as 'a' (= 1) or 'ano' (= a or no, i.e. either 1 or 0). However, current WG dispenses with the 'has' relation altogether. In diagrams, it sometimes (but not always) uses the quantity as a label for the node concerned; for example, Figure 0.4 shows that the number of beaks for a bird is 1. A simple dot shows that the number is unconstrained - i.e. either that the relatum is optional and may be multiple, or that the quantity is inherited from elsewhere.

But however convenient this notation may be when drawing networks, it is no more than a notational trick. If labels are basically mere mnemonics, as I claimed at the end of section 0, then it is wrong to pack so much information into labels. To be theoretically pure, these numerical labels should be replaced by separate relations, so we must consider this underlying reality.

Take the example of a bird's beak. The network must show not only that this node isa beak, but also that it is an obligatory part of the bird's anatomy – in other words, every bird must have precisely one beak. The WG solution is to recognize numerical quantities as entities with the same properties as other entities. Thus '1' is an entity which is related, inter alia, to the value of the 'beak-of' relation (i.e. x in Figure 0.7). The relation between a numerical quantity and another entity is '**quantity**', represented in diagrams (when needed) by an arrow labelled '#', and implied by a number (0 or 1) standing in for the node itself. This is another primitive relation, like Isa, Argument and Value.

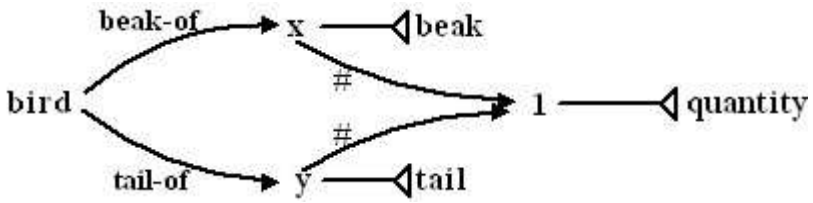


Figure 0.7: 'Quantity' as a separate relation

The effect of treating quantity in this way is to separate it from other properties, which is clearly right. For example, the physical and functional properties of beaks are quite separate from the questions of which creatures have them and how many they have. Similarly, throughout language structure a revealing analysis must separate quantity from other properties such as word class and position. For example, syntactic subjects have a large number of properties that typically converge, such as word class, position and semantic role, but (as we shall see in section **Error! Reference source not found.**) these other properties are independent of whether or not the subject has a 'realization' (i.e. an audible or visible form); so we know that an imperative verb normally has an unrealized subject, but we also know what (and where) the subject would have been if it had been realized.

Quantities are important for processing because they determine what we expect to meet in experience. For example, if we see a cat we expect four legs and if we hear the verb *give* we expect a subject, a direct object and possibly an indirect one – one subject, one direct object, and either one or no indirect object. Every token of experience, by definition, has a quantity of one, so when we process experience we have to match the expected quantity against this observed quantity. Even if the three-legged cat's overall properties match well those of a cat, the missing leg is registered as an exceptional feature; and of course ungrammatical combinations of words are commonplace in everyday speech. If by default every concept's quantity is 1, this can

be overridden in the usual way by other quantities (including 0, ≤ 1 and 0 ≥, meaning respectively 'impossible', 'optional up to 1' and 'any number'). However it is also possible that quantities reflect frequency, so that a commonplace experience is stored with quantity 1 but a rare one is stored with a lower figure, while an impossible one (such as unicorns or Father Christmas) has zero quantity. A system like this would allow us to distinguish commonplace experiences from unusual or even astonishing ones, but it also raises serious research questions that I cannot even try to answer here, such as how to make it sufficiently context-dependent.

Nodes that have a quantity specified in this way are naturally those that have no specific referent such as a particular individual or a general concept – nodes with meanings such as 'the father of a typical person' or 'the subject of a typical verb' or even 'the subject of the verb *go*', bearing in mind that *go* has different subjects in different sentences. Since their reference varies with the situation we might call them 'variables' (as indeed I did in Hudson 2007a), but this would be misleading because they are different from the variables of predicate logic. For one thing, they are never completely empty of content because they always have an Isa link to some other concept, so they are more like Jackendoff's 'typed variables' (Jackendoff 2002:42). This being so, there is only a difference of degree between them and 'constants' such as the concepts for John or the typical man: 'variables' are relatively poorly specified and constants are relatively richly specified, but in between we find a continuum of richness. Another difficulty in applying the constant/variable contrast to the concepts of WG is the principle introduced in section 0 that all the information in a network resides in the relations rather than in node labels. If we cannot use the labels to distinguish constants and variables, how can we distinguish them at all?

The conclusion must therefore be that WG has no variables as such; but one survey of inference in network models claims that, in a 'localist' network, variables are essential for generalization (Browne and Sun 2001). This claim may be true of other systems, but fails for WG because although the networks are localist (rather than distributed – i.e. each concept is represented by a single node) and have no variables, they certainly do allow generalization. The discrepancy can be explained if we note that none of the networks in Browne and Sun's survey allows default inheritance, the mechanism for generalization in WG. For example, WG allows us to refer to a bird's beak, or a verb's subject, even though different birds have different beaks and different verbs have different subjects. As in a logic-based system, the beak or subject is represented by a node, but this is allowed to have variable referents because default inheritance creates a new token node for each inference. This will be explained more fully in section 0, and section 0 will show how WG expresses the distinctions of predicate logic such as quantification. Furthermore, since the notion of 'variable' is closely linked in classical logic with the notion of 'binding', I shall also explain in the discussion of binding (section 0) how WG shows which nodes need to be bound.

Multiple default inheritance

Default inheritance is the logic of the Isa relation. By definition, if (say) Penguin isa Bird, then facts about Bird generalize to Penguin (and to all other sub-classes of Bird). This is what Isa means, and no other relation type has this meaning. The technical term for this downward spreading of facts is 'inheritance', so Penguin is said to inherit facts from Bird. In other words, the facts listed in the network directly for Penguin are supplemented by those which are listed for any other concept that Penguin isa, which in turn are supplemented by their super-categories and so on. This is not only an efficient way of storing predictable information, but it is also an important way of

supplementing our existing knowledge. For example, we may not know from personal experience whether or not penguins have hearts, but if we don't know we can easily inherit this information from a higher concept.

Apart from the technical term, this logic is merely a matter of common sense. Given that we are trying to model how humans actually store information, it is obvious that some kind of inheritance system must be available because everyday experience confirms that this is the logic we live by - we see something, we guess what super-category it isa, and then we assume that it has all the unobservable properties of the super-category. For example, if we guess that something isa Cat, we assume it likes to be stroked because this is one of the facts that are stored in our knowledge of Cat. There is very little doubt that inheritance of properties plays an important part not only in ordinary human reasoning, but also in our knowledge of language. For example, as soon as we learn a new word and assign it to a word-class, we can infer a great deal of unobservable information from that word-class.

What is less clear is the extent to which we (as learners rather than as analysts) actually exploit inheritance in order to minimize storage. Are inheritable facts ever stored, or do we always avoid storing them because they are redundant? For example, given that Peacock isa Bird, and that Bird has feathers, we certainly don't need to store the fact that Peacock has feathers; but do we in fact store it? The experimental evidence suggests that we do store it for Peacock, though maybe not for other birds with less memorable feathers (Reisberg 1997:270). In any case redundancy is not a major issue given the vast storage capacity of our long-term memories, so we may assume that some facts which could be inherited are in fact stored directly.

This raises a serious problem if we are trying to model human competence: how can we know for sure which facts are stored and which are inherited? For example, in a detailed analysis of inflectional morphology can we assume that regular inflections are always inherited? Evidence from experiments and from language change suggest that we cannot (Bybee 1999; Ellis and Schmidt 1998; Harley 1995:161): at least some regular forms are in fact stored, and especially so if they are used frequently. Indeed, it is hard to see how it could be otherwise if generalizations are induced from observed 'usage' – i.e. from a collection of memorized instances – as I shall argue in section 0. Once a generalization has been made on the basis of stored instances, those instances may be redundant but there is no mechanism for deleting them from memory, so we must assume that at least these stored cases persist; and if these redundant memories can coexist with the generalization from which they could be inherited, why not other memories too? What this means for linguists is probably that we cannot claim to model actual knowledge; all we can model is an idealized knowledge with minimum redundancy. This will define the minimum of stored knowledge, while recognizing that actual speakers may add vast amounts of redundant links.

Returning to the general principle of inheritance, its psychological reality is surely uncontroversial. It is also relatively easy to combine with a network model of knowledge such as WG, provided that this network includes Isa relations. Inheritance can be represented schematically as the relation between the two networks shown at the top of Figure 0.8. In this figure,

• the dotted line shows '**transitive-isa**', i.e. a chain of one or more Isa relations, so if X isa Y and Y isa Z, then X transitive-isa both Y and Z (and so on up the Isa hierarchy).

• the double-headed arrow means a relation pointing in either direction.

• the broad horizontal arrows show that the network on the left can be expanded by inheritance into the one on the right.

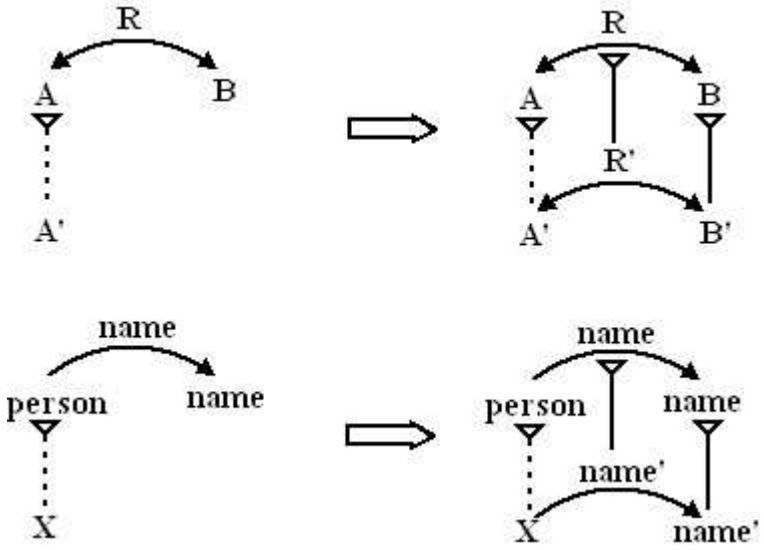


Figure 0.8: Inheriting defaults in a network, e.g. a person has a name

It might be thought that inheritance would apply the source fact (A R B) to the inheriting node (A′) in the simplest possible way, by providing an extra link from A′ to B; in this approach, a person X would inherit Name from the default person in the form of a direct link from X to Name. However this would lead to serious logical problems because it would imply that Name (the typical name) belonged not only to Person but also to person X, so anyone else inheriting Name would also inherit this link to X. To avoid this logical problem, inheritance works by creating a new and distinct copy of the inherited fact: A′ R′ B′.

The general idea of inheriting information from general to particular is uncontroversial in cognitive psychology (though it is noticeably absent from most network models of knowledge). Much more controversial is the idea that we only inherit information 'by default' - hence '**default** inheritance'. Once again this can be seen merely as a matter of common sense. Our stored information defines the typical bird, penguin or whatever, but we can also cope with non-typical examples such as plucked birds (which have no feathers), albino penguins and so on. We happily classify something as a cat even if one of its legs is missing, and in language we accept non-typical features such as irregular morphology and even spelling mistakes. The same is true of stored concepts; for example we recognize that Ostrich isa Bird even though it doesn't fly. In other words we allow its 'walking' to **override** the default 'flying' as the typical means of locomotion (and similarly for its size). In case common sense needs experimental support, this is available in abundance from work on 'prototype effects' (Reisberg 1997:311-329). Categories have relatively ‘good’ (i.e. typical) or ‘bad’ members (e.g. robins are better birds than ostriches are), and they may have borderline members (e.g. what counts as a piece of furniture – how about TV sets and ashtrays?). These effects are exactly as expected if categorization allows exceptions: good members inherit all the default properties, worse members override some of them, and borderline members override so many that it is debatable whether they are members at all (Hudson 1990:45, Jackendoff 2002:185).

Default inheritance is clearly a fact of ordinary life, and it can be modelled in a network. The two pairs of networks in Figure 0.9 show how an existing proposition blocks the inheritance of any competing proposition (i.e. one with the same relation and entity). For example, given that mushrooms are plants (rather than animals) but that their colour is grey, we do not try to inherit the default green.

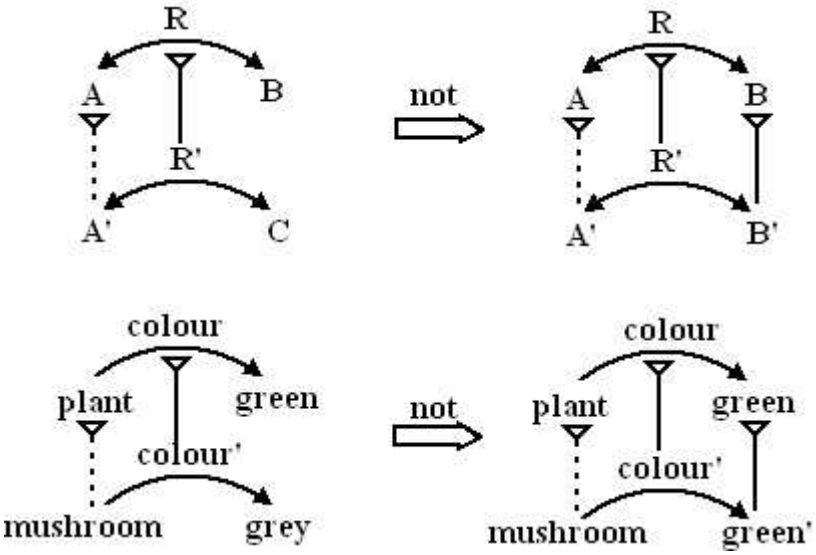


Figure 0.9: Overriding a default – the colour of mushrooms

Nevertheless, default inheritance is controversial in the research world of AI because it 'compromizes the nature of definitions themselves. ... If we define a penguin as a bird that does not fly, what is to prevent us from asserting that a block of wood is a bird that does not fly, does not have feathers, and does not lay eggs?' (Luger and Stubblefield 1993:388). The answer is surely that no human mind would make this classification because it would be unlearnable, given the principles of processing and of learning that I shall outline in sections 0 and 0. The stored classification is based on the classification of some token of experience, which in turn is based on the 'best fit' principle of choosing the classification which provides the best global fit between the token's observed properties and the existing network. How could a block of wood qualify as a bird in this scenario?

Another standard objection to default inheritance is that it is very hard to implement in a working computer model (Shieber 1986). The problem is that this logic is assumed to be 'non-monotonic', but I shall show shortly that in WG this assumption is false. Inference is said to be monotonic if it is simply cumulative, so that later inferences never overturn earlier ones; in non-monotonic inference, on the other hand, any conclusion which is drawn may turn out later to be wrong. For example, default inheritance would be non-monotonic if the default was inherited and then later abandoned because of exceptions. Non-monotonicity makes every inference provisional because there is no way to know in advance which will be overridden, so no firm conclusions can be drawn until every inference has not only been drawn, but also checked for possible overriders. If these assumptions are true, it is easy to understand why those working in logic and AI are uncomfortable with non-monotonic inference.

In spite of these widespread anxieties about default inheritance, I believe they have been exaggerated and the problem has an easy solution: **default inheritance**

only applies to tokens. In other words, tokens can inherit from stored types, but types cannot inherit from each other.) To start with a non-linguistic example, suppose I have a stored concept Cat and I want to apply it to a particular token of experience X which I have classified as a cat; so all I know is that X isa Cat. I can apply default inheritance to X, so if I know that Cat (the typical cat) enjoys being stroked, I can assume that X does too. On the other hand, because inheritance only applies to tokens, I cannot apply it to Cat in order to find out whether Cat has skin; but if I want to know whether X has skin, I can inherit this fact from any concept that X transitive-isa (e.g. Animal) because it is transitive-isa, rather than plain isa, the allows inheritance.

One of the advantages of restricting inheritance to tokens is that it explains why default inheritance does not clog the network with redundant properties. As mentioned earlier, there is in fact a great deal of redundant information, but it is fair to assume that most of this information results from direct experience rather than from inheritance. Putting this assumption in functional terms, there is very little point in enriching a stored node by inheritance, because inheritance itself already makes the added information so easily available; but it is absolutely essential to apply inheritance to a token node because that is the only way to enrich it beyond the properties which are directly observable.

It could of course be objected that we can in fact draw inferences about stored concepts; for example, we can infer that birds in general – i.e. Bird, the typical bird - have a heart because we know that they are animals and that animals have hearts. However, it is easy to accommodate this kind of inference by assuming that what we are actually doing is setting up a hypothetical token and inferring to that. This explains why we can use ordinary anaphoric pronouns to refer to such tokens as in the following exchange (which I owe to Mark P. Line):

(4) A. Can a bird fly?
B. Yes.
A. What if it's a penguin?

The normal rules of interpretation give the pronoun *it* the same referent as its antecedent, which in this case is *a bird* in the first line; but this is only possible if this referent is a token which is distinct from both Bird and Penguin (the senses of *bird* and *penguin*), because its super-class must be able to shift from Bird in the first line to Penguin in the third.

However, if inheritance only applies to tokens, another crucial characteristic follows: **inheritance works bottom-up**, i.e. starting with the lowest node in the Isa hierarchy, and then working up from there. This is again a very natural assumption in terms of network structure – what could be more natural than to enrich a token node from the nearest node first? It is also very easy to design a recursive algorithm for inheriting first from node A, then from A's super-category B, then from B's super-category C, and so on. But most important of all, this solves the problem of non-monotonic inheritance, because default inheritance will, in fact, be monotonic. No inherited property will ever be overridden, because more specific properties will always be inherited before more general ones and the first property always wins.

This is an important conclusion because it explains why inheritance is so fast and so trouble-free. All the processor has to do is to visit a clearly defined series of nodes, and for each one inherit onto the token any relations for which it does not yet have a value. (We shall see in section 0 that spreading activation may make inheritance even easier than this if inheritance only applies to relations which are already active.) In particular there is no question of searching the total database for potential overriding properties.

Even more controversial is the use of **multiple** default inheritance, which follows automatically in WG from multiple isa, i.e. the fact that one node may isa several other nodes. The classic discussion of the problems of multiple inheritance is Touretzky 1986, which illustrates them with the so-called 'Nixon diamond' in Figure 0.10. This refers to the historical fact that the American president Richard Nixon was both a Republican and a Quaker. These two reference groups typically hold opposing views on warfare (represented crudely in the diagram by the relation 'war?'), with the consequence that Nixon could inherit contradictory views.

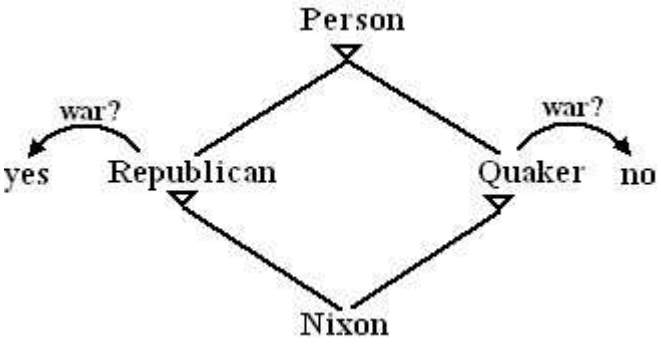


Figure 0.10: The Nixon diamond

This is generally presented as an argument against multiple inheritance on the assumption that a logic should not lead to contradictory conclusions; but in my opinion it actually shows the rightness of multiple inheritance. After all, the ultimate test of a logic is whether its conclusions are correct, and in this case the conclusion is in fact correct: Nixon's situation was contradictory. For consistency he should have renounced one of his reference-groups, but in fact he resolved the conflict by fiat - by deciding in favour of the Republican value. A more accurate representation of the situation would therefore be as in Figure 0.11, where Nixon's preferred (and stipulated) value correctly overrides that for Quaker.

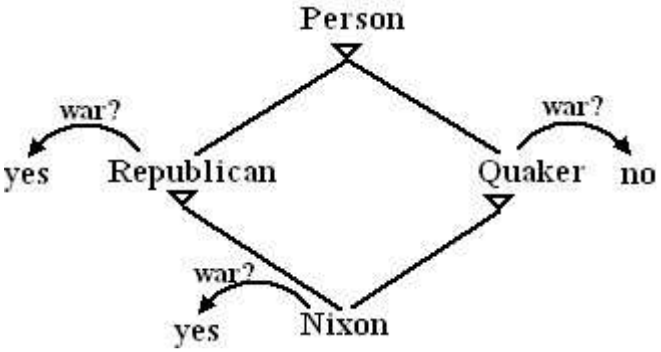


Figure 0.11: The Nixon diamond resolved

However controversial it may be, multiple inheritance seems exactly right for language structure. It is extremely common for linguistic categories to intersect, the obvious example being inflectional morphology where an inflectional category such as Plural-noun intersects with some lexical item such as DOG to define a combined category 'DOG:plural' - the plural of DOG. Later chapters include a general discussion of mixed categories in syntax (**Error! Reference source not found.**) as well as an extended discussion of multiple inheritance showing how gerunds inherit

both from Noun and from Verb (chapter **Error! Reference source not found.**). I have also used multiple default inheritance to explain the rather odd gap where we expect **I amn't* (or **I aren't*) as in (5) Hudson 2000a:

(5) a He's tired.
b He isn't tired.
c You're tired.
d You aren't tired.
e I'm tired.
f *I amn't tired.

In a nutshell, the missing form inherits from both Negative (like *aren't*) and First-person (like *am*), but since neither of these categories transitive-isa the other, the conflict cannot be resolved.

In the last few sections I have presented a general theory of classification and inheritance that may strike the reader as simply a matter of common sense – as a formalization of existing practices, or perhaps as a notation for recording information about how concepts are classified and defined (in terms of links to other concepts). However, the theoretical package which contains inheritance networks and default inheritance makes a considerable difference to the analysis itself. Here are some very general consequences of adopting this theory:

• Sub-classification may distinguish just one sub-class. It is tempting to think of classification as the division of a larger class into at least two smaller ones, but this is wrong in an inheritance network. A sub-class contrasts not with other sub-classes, but with the super-class. For example, a person might recognize just one sub-class of rose, e.g. dog-rose, without necessarily lumping all other roses together as non-dog-roses; rather, the rest would simply be ordinary (i.e. default) roses. In linguistics, this is how we recognize 'markedness'. The unmarked member of a pair is the superclass, while the marked member is the sub-class – for example, singular nouns are simply default nouns, the unmarked member, with Plural-noun as the exceptional sub-class.

• Features are independent of classification. Most linguistic theories assume that classification is done in terms of contrastive features (also widely known as 'attributes') such as gender, number and tense. This approach is probably most fully developed in the theory that I learned first, Systemic Grammar (now called Systemic Functional Grammar), in which these features are organized in contrasting sets called 'systems' and systems are interrelated in a 'system network' (Halliday 1985, Hudson 1971). An earlier version of WG assumed that features were part of the classification system; for example, I suggested (Hudson 1990:93) that English verbs were divided by the feature finiteness into finite and non-finite, with mood dividing finite into imperative and tensed. However I now think features are merely a particular kind of relation; for example, the 'feature' Gender is a relation between a noun and one of the values Masculine, Feminine, or Neuter, just as Meaning is a relation between a word and its meaning. Where features are needed – in section **Error! Reference source not found.** we shall consider some situations where they are important for syntax – they can be recognized, but they are predictable from classes, rather than providing the foundation for these classes. To take the example mentioned in the previous bullet point, we can recognize Number as a feature of nouns, which relates them to the abstract values Singular or Plural, while also distinguishing singular and plural nouns through the Isa hierarchy (where singular nouns are in fact just default nouns, with plural nouns as exceptions). In this analysis, the default value for Number is Singular, but exceptionally plural nouns have the value Plural (Hudson 1999).

• Sub-classes and members are not distinct. Standard set theory makes a fundamental distinction between sub-sets, which are sets, and members, which are individuals. This distinction has no place in WG because categories are all more or less abstract and schematic 'types' rather than sets. (In fact, Set is one special type which we shall exploit in sections 0 and **Error! Reference source not found.**.) The category Dog must be the same kind of thing, logically speaking, as the particular dog Fido, because otherwise Fido could not inherit the characteristics of Dog; and in particular, Dog is not a kind of set. (Dog is the typical dog which has a tail and barks, but sets don't have tails or bark; conversely, sets have members and numerical sizes, which dogs do not have.) In WG, types, sub-types and individuals have just the same status and are mixed up together in the network; for example, under Noun we might find both Proper (a sub-class) and DOG (a member). Indeed, even individual tokens of experience, such as a particular cat or a particular instantiation of the word DOG, are part of the same inheritance hierarchy as the more general categories, and have just the same logical status (apart from being tokens rather than stored types). This merging of individuals and general types seems psychologically sound; for example, we can recognize exceptional and dated cases of an individual (e.g. John when unwell, or John when he was a small child) just as we can with general types (e.g. person when unwell or small children). Moreover, according to the WG theory of learning (section 0), general types are induced from more specific types, which in turn are learned as tokens; if this theory is right, individuals and sub-classes must have a very similar cognitive status and compatible formal properties.
All these principles follow from the general properties of inheritance networks, and they all affect the way we analyze knowledge in general, and language in particular.

In conclusion, then, the logic of WG is multiple default inheritance, defined by the following facts about a concept A which isa B:
• **Inheritance**: Normally A inherits all the characteristics of B and any other nodes on the isa chain leading up from B (i.e. any node which A transitive-isa).
• **Default inheritance**: But it does not inherit values for relations which already have a value.
• **Multiple inheritance**: If A transitive-isa any other concept, it inherits from this in the same way as from B.
It is this inheritance system that lies behind all classification and all generalization, so it is a very important part of any conceptual network - hence my description of such networks as 'inheritance networks'. We shall return in section 0 to the details of how it may be implemented in a model of processing.

Logic

It may be helpful at this point to compare the expressive power of a WG network with that of the predicate calculus. This is an important comparison for readers who are already familiar with the predicate calculus and who may be wondering to what extent a mere network of nodes can achieve the same effects. I shall try to show that WG has a similar expressive power, though of course the two systems can never be exactly equivalent because they are based on contradictory assumptions. Classical logic allows no exceptions, but exceptions are part of everyday reasoning so WG does allow them (through default inheritance). Thus given the axioms 'If something is a bird, then it flies' and 'A penguin is a bird', in classical logic it follows unavoidably that a penguin flies; whereas in WG this conclusion may be blocked by the exceptional axiom: 'A penguin does not fly'. However I have already explained in

section 0 that WG can achieve this effect while maintaining monotonicity so that although the general case is not always true, it will only be applied when it is true.

We start with **universal** quantification. In the predicate calculus, axioms are defined by propositions which consist of a predicate and its arguments expressed as variables which are bound by a quantifier; for example, the axiom that people die would be expressed by the predicate Die, the variable x and the universal quantifier ∀:

(6) $\forall x$, Person(x) → Die(x)

(For all x, if x is a person then x dies.) In WG this axiom is defined by the network in Figure 0.12: the typical person is the die-er in one instance of Die (i.e. dying). The effect of the universal quantifier is achieved by assigning the property to the general category Person; default inheritance applies it universally (subject to possible overriding). In contrast with predicate logic, WG makes no distinction between predicates and variables for the reasons I explained in section 0. As far as the underlying network is concerned, Die and 1 are both just nodes, distinguished by their relations to other nodes but not by their labels.

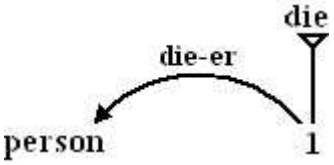


Figure 0.12: Everybody dies in WG

Both systems allow predicates to have more than one argument, but they do this in different ways. In the predicate calculus a predicate may have any number of arguments (including none at all); and it distinguishes these arguments only by their order. For example, the proposition that people give their children presents on their birthdays might be expressed with a four-argument predicate: Give (w, x, y, z), combined with quantifiers and propositions that classify w as a person, x as w's child, and so on:

(7) ∀(w), Person (w), ∀(x), Child (x, w), ∀(y), Birthday (y, x), ∃ (z), Present (z), Give (w, x, y, z)

(For every person w, for every child x of w, for every birthday y of x, there is a present z which w gives to x on birthday y.) In WG, in contrast, every relation is necessarily binary – a link between two nodes in the network – so four-argument relations cannot be expressed directly. Nor is it possible in WG to rely on the order of arguments to distinguish them, because there is no left-right order in a network. Instead, each proposition is represented by a single node for its predicate, and the arguments are linked to it by binary relations whose classification distinguishes their roles. The proposition about birthday presents is therefore expressed by the network in Figure 0.13. This network claims that on every birthday of every child of every person there is an act of giving whose giver is the person, whose receiver is the child and whose time is the birthday; in this act of giving, the gift is some present.

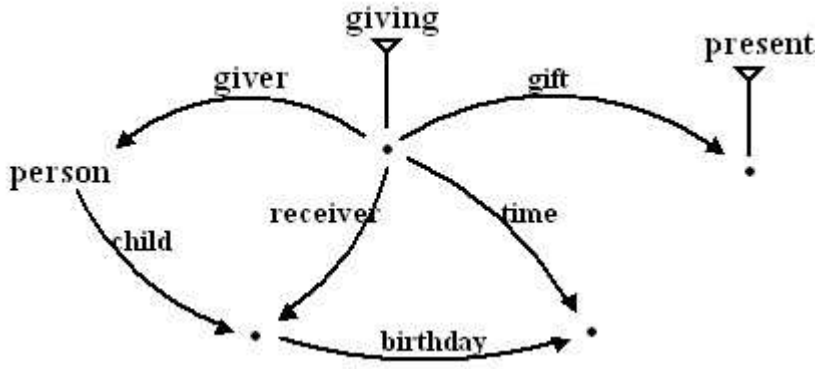


Figure 0.13: Everybody gives birthday presents in WG

It is difficult to evaluate these different ways of handling propositions and arguments, which each arise out of an established tradition in semantics – the logical tradition for the predicate calculus, and the traditions of AI and lexical semantics for the WG approach. However the main attraction of the WG approach is that it allows generalizations which are not possible at least in the classical version of first-order predicate calculus. On the one hand, we can generalize across predicates by relating them in isa hierarchies – man isa person, lend isa give, and so on. In terms of the predicate calculus, this is equivalent to treating predicates as variables; for example, if lending is a special case of giving then 'Give (Lend)', an illegal formula. On the other hand, we can also generalize across argument-roles because these are stipulated categories which can be related in isa hierarchies; for example, we might generalize about givers or children (or, in linguistics, about agents or subjects).

Returning to quantification, I have already explained how WG expresses universal quantification but we can now consider **existential** quantification. Figure 0.13 contains two examples, both shown by unlabelled nodes. Consider the node which isa Present. This means 'some present', not 'every present', because not every present is given to a child on its birthday. (More precisely, this node means 'some present which is given to some child by that child's parent on that child's birthday'; but the main point is that it is distinct from the Present node which means 'every present'.) Similarly, the node which isa Giving means 'some act of giving', not 'every act of giving', because not every act of giving involves a child's birthday present. However, it is important to stress that the notation achieves this effect because of the way in which default inheritance works, not because of the difference between labelled nodes and unlabelled dots; as I stressed earlier labels in themselves have no theoretical status. If something is classified as a present, it inherits all the properties of Present, but not of specific sub-cases of Present – i.e. properties are inherited down the Isa hierarchy, but never up it. Consequently, the properties of the sub-case of Present which is shown in Figure 0.13 cannot be inherited from Present, so they are not 'universally quantified'. The same principle explains why inheritance works as shown in Figure 0.8 in section 0: if A' inherits from A a relation to B, this relation must not relate it to B itself, because it would then be available for inheritance to any sub-case of B. Instead, A' inherits a relation to B', a sub-case of B.

In short, any node X always means 'every X', regardless of whether it has a distinctive label (e.g. Person, Present) or a mere dot or number, and regardless of whether it has a generic or an individual reference. What this means is that any other node X* which isa X must inherit X's properties. But if X isa Y, then it is merely 'some Y', so its properties are not inherited by other instances of Y. Even more

briefly, nodes are universally quantified, but their sub-cases are existentially quantified.

The **logical operators** ($\wedge$, $\vee$, $\neg$, $\rightarrow$) of classic predicate logic can also be expressed in a network, though again the means of expression are very different. We start with the 'and' and 'or' operators ($\wedge$, $\vee$), which are expressed in terms of sets. As I mentioned briefly in 0, WG treats a set as a particular kind of individual which has properties such as size and members. This approach works well in semantics; for example, the referent of a plural noun is a set whose typical member isa the lexeme's sense; so *dogs* refers to a set whose typical member isa dog (Hudson 1990:139-45), and a word like *families* or *sets* refers to a set of sets. Sets are also important for the grammar and semantics of coordination, where the meaning is a set (ibid: 410-11); for example, the meaning of (8) is a two-member set whose members are the events of John shopping and Mary cooking.

(8) John shops and Mary cooks.

The semantic structure is shown in Figure 0.14.

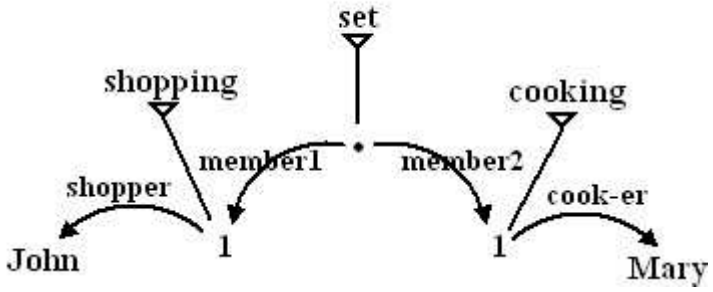


Figure 0.14: John shops and Mary cooks

One advantage of this approach compared with the logical operator $\wedge$ is its ability to handle combinations of things other than propositions. Thus we can recognize exactly the same structure in terms of sets in the meaning of conjoined nouns as in *John and Mary bought a house*. The semantic structure for the collective interpretation of this sentence (where they bought it jointly) is shown in Figure 0.15. The crucial part of this diagram is that the buyer is the set consisting of John and Mary.

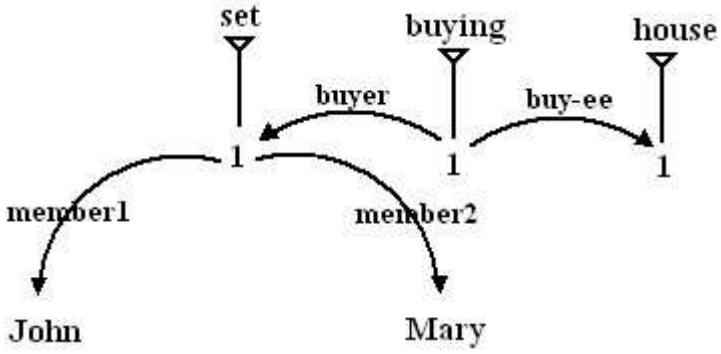
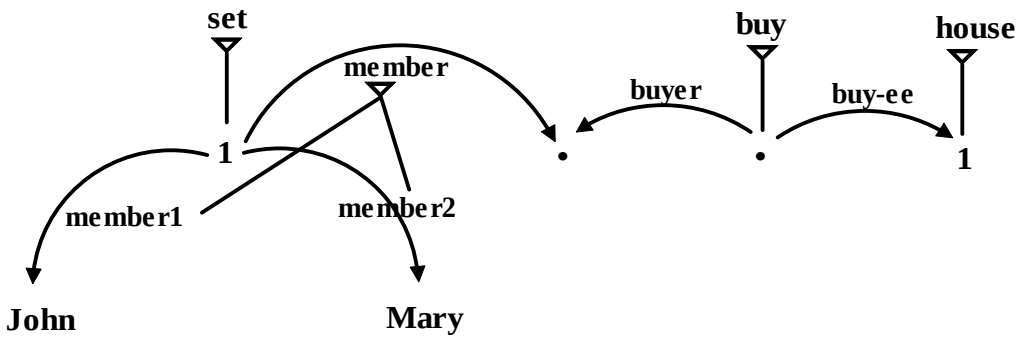


Figure 0.15: John and Mary bought a house (collective).

The sentence's other interpretation is the distributed one in which they each bought a house. This is a little more complex, but builds on the same set structure. As can be seen in Figure 0.16, the buyer is the typical member of the set of John and Mary, so by inheritance it generalizes to every member. Another difference is that the number of events is no longer restricted to one; in a more complete analysis (as explained in section **Error! Reference source not found.**), the number of events would be tied to the number of members of the set.

Figure 0.16: John and Mary bought a house (distributed).



How then can we distinguish 'and' from 'or' in the analysis of sets? In 1990 I offered a rather unsatisfactory analysis involving two elements '&' and '/' whose status was undefined, but I can now do better. As we might expect, the 'and' meaning is the simpler of the two, and in fact requires no further structure. The last three figures all force a universal interpretation in which both events exist (Figure 0.14) and both John and Mary are involved in the house buying, either collectively (Figure 0.15) or singly (Figure 0.16). The effect of changing *and* to *or* is much the same as that of changing universal to existential quantification because we change from 'every member' to 'some member'. As with existential quantification, we can achieve the desired effect by referring to an arbitrary sub-case (here, an arbitrary member) rather than the entire set. The semantic structure for this sentence is shown in Figure 0.17, where the third member (labelled 'm3') is the arbitrary member which is to be bound to one of the others (i.e. to John or Mary). The meaning of the sentence could thus be paraphrased as 'some member of the set consisting of John and Mary bought a house'. As can be seen, this approach has the attraction of keeping the syntax and semantics closely in step, so that phrasal or word coordination can both be represented as sets of individuals. This strikes me as much better than the predicate calculus, where

disjunction is always a relation between entire propositions so that coordinated words have to be interpreted as though they were coordinated clauses.

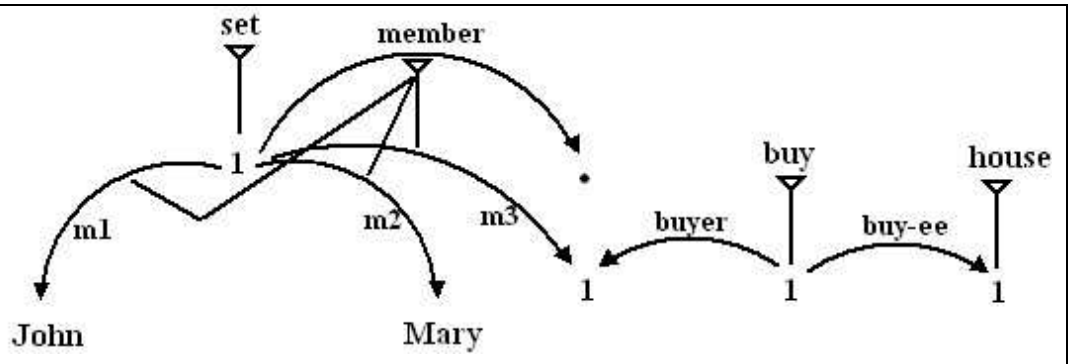


Figure 0.17: John or Mary bought a house.

The third operator is ¬, meaning 'not'. Negation is very easy because we already have exactly the right apparatus: the relation 'quantity' which I introduced in section 0. Negation is shown in semantic structure by the value 0. For example, *It is not raining* has exactly the same semantic structure as *It is raining*, except that the quantity of the event is 0 rather than 1. Similarly, *no student* has just the same semantics as *a student* except that its quantity is 0; and likewise for its plural, *no students*, though of course in this case it is a set rather than an individual that has zero quantity.

The last logical operator → has roughly the same meaning as *if*, but it has a more precise meaning which can be defined truth-functionally: 'P → Q' is false if P is true and Q is false; otherwise it is true (or irrelevant). In logical form, 'P → Q' means the same as '(P ∧ Q) ∨ ¬P'. Since we already know that WG can express the other three operators (∧, ∨, ¬) we can also be sure that it can also express this particular combination of them.

In conclusion, a WG network has all the strengths of first-order predicate logic without (so far as I know) any of its weaknesses.

Spreading activation

One of the many attractions of the network view of language structure is that it provides a strong bridge to current work in psycholinguistics and cognitive psychology, where network models are also popular. Linguists and psycholinguists are studying the same object – language – so their theories must eventually converge on one which is supported by both linguistic and psycholinguistic evidence. The psycholinguistic evidence for networks is overwhelming. The crucial difference between a network and a collection of rules is that only the former defines the notion of '**topological distance**', i.e. the distance between nodes, which in turn supports the notion of **spreading activation**, whereby activation spreads blindly from one node to its '**neighbours**' (a notion that makes no sense outside a network).

The psycholinguistic evidence for spreading activation comes from two sources:

• speech errors, in which a target word is replaced by a different one which is almost always one of its neighbours in the permanent network, as well as often being one of

its neighbours in the network of the current utterance. For example, when Dr Spooner told a student that he had 'tasted the whole worm', the word *tasted* showed the influence not only of its permanent neighbour *wasted* but also of its utterance neighbour *term*.

• priming experiments, in which a preceding word 'primes' a later word by making it more accessible so that an experimental subject can retrieve it more quickly. Not surprisingly, it turns out that words prime their network neighbours. For example, experimental subjects take slightly less time to decide that *doctor* is an English word if it follows *nurse* than if it follows an unrelated word such as *lorry*. Both semantic and formal (phonological or spelling) similarities are relevant to priming, though semantic priming lasts much longer than formal priming (Harley 1995:146, 149). Every experiment which shows that one word primes another is evidence that these words are near to one another in the network. For example, the words *nurse* and *doctor* might be separated by as few as four links, as in Figure 0.18. Once again, all that counts is the number of links, and not their classification or direction.

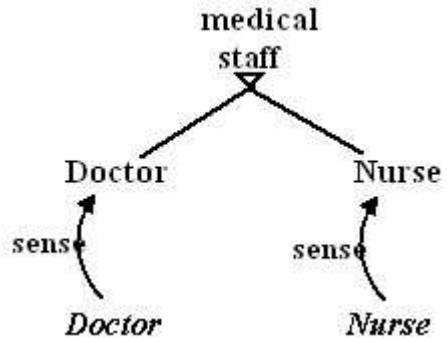


Figure 0.18: Links needed to explain the priming of *doctor* by *nurse*.

There is very little doubt about the reality of spreading activation. Moreover, it is important to stress that errors and priming are found at every linguistic level, including those which are often thought of as the domains of 'rules' rather than network activity. Starting with errors, the interfering items may be neighbours of the target at the following levels, and at some levels they may also be near to each other in the utterance ('utterance neighbours'):

• Phonology:

(9) There were lots of little orgasms (for: organizms) floating in the water. (Aitchison 1994:20)

(10) utterance neighbours: the mirst (for: first) of May (Harley 1995:352)

• Morphology:

(11) the chung (for: young, children) of today (Harley 1995:352)

(12) utterance neighbours: slicely thinned (for: thinly sliced) (Levelt, Roelofs and Meyer 1999a)

• Syntax:

(13) I'm making the kettle on. (for: making some tea + putting the kettle on (Harley 1995:355)

• Meaning:

(14) Get me a fork (for: spoon). (Harley 1995:352)

• The environment of the utterance:

(15) (Addressee is sitting at a computer.) You haven't got a computer (screwdriver) have you? (Harley 1990)

Examples such as the last one are particularly interesting because they reveal the intimate connection between the language network and the rest of cognition. The computer in this example has nothing whatever to do with language as such; it is simply part of the physical context which the speaker is processing non-linguistically. And yet it interferes with the choice of words in just the same way as it might have done if the discussion had been about computers, which shows that activation spreads as easily from 'general cognition' to 'language' as it spreads within the language network. The example is not isolated; Harley lists hundreds of attested examples.

The evidence from priming experiments leads to the same conclusion. Once again, we find that spreading activation can affect elements at all levels, including some general 'syntactic' patterns which might be associated with 'rules' rather than networks.

• Phonology:
verse primes *nurse* (Brooks and Macwhinney 2000; James and Burke 2000)
• Morphology:
hedges primes *hedge* in a way that can be distinguished from phonological priming (Bauer 2003:287)
• Syntax:
Vlad brought a book to Boris primes other sentences containing Verb + Direct Object + Prepositional Phrase (Harley 1995:356; Bock and Griffin 2000; Chang, Dell, Bock and Griffin 2000)
• Semantics:
bread primes *butter* (Harley 1995:17)

The most significant category in this list is the syntactic priming. It is relatively easy to accept that lexical items are interrelated in a network, but syntactic patterns are widely believed to be stored in a different way, as separate rules or schemas. The existence of priming effects suggests strongly that they too are stored as items in a network. I shall explain in chapter **Error! Reference source not found.** how syntactic patterns can be stored in a network as properties of general word types.

How exactly does spreading activation work? How does such a crude, unguided process help us to achieve our cognitive goals, rather than leave us drifting aimlessly round our mental networks? It is very unclear exactly how it works in mathematical terms, but the WG hypothesis is that a single formula controls activation throughout the network. (As I admit in section 0, this hypothesis can only be tested, of course, in a computer model.) What is clear, however, is that processing is goal-directed; for example, when we hear a word we (normally) look for its meaning and are frustrated if we cannot find it. In some activation-based models the directionality is 'hard-wired'; thus a model of production will lay down a series of steps through which the processor must pass in order to achieve the predefined goal (Levelt, Roelofs and Meyer 1999a, Jackendoff 2002:198). This is not how WG handles directionality. Instead, it assumes that goals are defined by current interests and goals, which in turn are expressed as spreading activation.

For example, when I hear a word, it is the context which decides whether I am most interested in its meaning, its syntax, its etymology or even its pronunciation. (The latter situations are familiar to any practising linguist or phonetician.) No single hard-wired model of speech perception will accommodate all these interests, but they are easy to explain if we assume that each interest involves a different kind of property (such as meaning, etymology and so on). These are defined in WG in terms of relations (as explained in 0) – i.e. classified links from one node to another. Each relation link is in effect a concept, so it may receive activation and pass it on to other

related links. In this way, spreading activation applies not only to nodes, but also to the links between nodes. Consequently, when Meaning is active, a word may have an active link to its meaning but not to its etymology; and more generally, activation spreading through the relation hierarchy activates links differentially.

How, then, does this activation of links help to direct processing? Imagine a situation where I have heard a word in the course of ordinary conversation. When I hear the word, my mind is already oriented towards meanings by virtue of the activation in the (general) Meaning link that is left over from the previous words. There is no need for hard-wired 'extrinsic ordering' of processes leading from sound to meaning because the word's sound and meaning already provide focuses of activity from which activation spreads . These active nodes define the goal of the processing: to find the best 'path' from the (known) form to the (unknown) meaning by selectively activating intervening nodes which receive activation from both directions and damping down the activation on all other nodes. In other words, the node which stands for the unknown meaning defines the target by 'pulling' the activation towards it. This process is illustrated schematically in Figure 0.19, where R is the relation which is currently active (e.g. Meaning). Its activation selects one target node, which is poorly defined ('empty') but active and in turn spreads activation to neighbouring nodes. Nodes which receive activation from this source as well as from the highly active 'known' node stay active while other nodes lose activation quickly, and these active intervening nodes provide properties which enrich the empty node. Exactly how this happens is the topic of the next section.

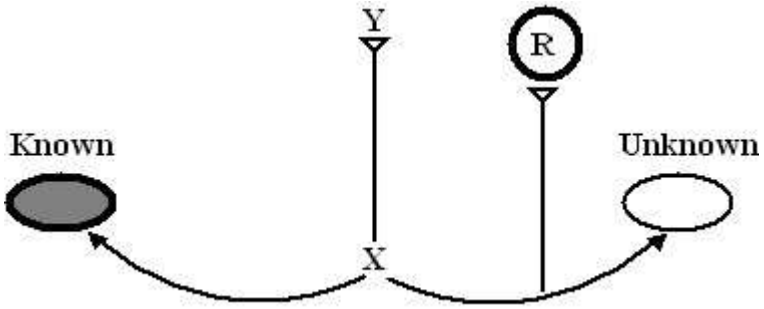


Figure 0.19: How activation from an active function defines the processing target.

Although further details must wait until the next section, there is one processing issue which we can address immediately: the nature and limitations of **working memory**. Activation involves physical resources (energy and time) which are limited. This limitation has tended to be discussed in terms of the number of items of information which we can hold in short-term memory (Miller 1956) but it is widely accepted nowadays that 'working memory' is simply the active part of permanent, long-term, memory:

> 'What is working memory? ... Working memory is those mechanisms or processes that are involved in the control, regulation and active maintenance of task-relevant information in the service of complex cognition, including novel as well as familiar, skilled tasks. It consists of a set of processes and mechanisms and is **not a fixed 'place' or 'box' in the cognitive architecture.** It is not a completely unitary system in the sense that it involves multiple

> representational codes and/or different subsystems. Its capacity limits reflect multiple factors and may even be an emergent property of the multiple processes and mechanisms involved. Working memory is closely linked to LTM, and **its contents consist primarily of currently activated LTM representations**, but can also extend to LTM memory representations that are closely linked to activated retrieval cues and, hence, can be quickly reactivated.' (Miyake and Shah 1999:450, emphasis added)

A reasonable hypothesis is that the 'capacity' of working memory is simply the amount of available activation. If this quantity is limited, then only a limited number of nodes and links can be highly active at a given moment so (for example) it will not be possible to keep more than a few unrelated items of information active – hence the famous limit of about seven to the number of arbitrary digits we can hold in memory.

What I hope to have established in this section is that spreading activation is massively supported by psychological experiment as well as by observation of spontaneous speech errors, and that it in turn gives overwhelming support for the Network Postulate in section 0, the claim that the whole of language is best modelled as a network. I have also shown that activation need not be directed along a predefined path provided that it spreads not only from the current 'known' node but also from an 'empty' target node. This section thus provides a bridge between the earlier discussion of how we store information and the following sections which deal with how we use this information in processing and how we learn it. We shall see that spreading activation plays a crucial role in processing.

Processing

The central claim of WG is that language, i.e. knowledge of language, is a network. In itself, this claim says nothing about processing, but one of its attractions has always been (since the early days of Stratificational Grammar – Lamb 1971) that a model of processing is relatively easy to add, and in principle 'the theories of competence and performance should line up' (Jackendoff 2002:30). Spreading activation is not unique to WG, of course, and has been a common element in computer models of speech and language processing. In psycholinguistics, spreading activation models have been constructed or proposed in:

• letter and word recognition (McClelland and Rumelhart 1988),
• word sense disambiguation (Quillian 1968, Anderson 1983, Hirst 1988),
• morphology (Marslen-Wilson 1984,Bybee 1995, Roelofs 1997),
• parsing and syntactic disambiguation (McClelland and Rumelhart 1988, Macdonald, Pearlmutter and Seidenberg 1994; McRae, Spivey-Knowlton and Tanenhaus 1998; Roland 2001; Sturt, Pickering, Scheepers and Crocker 2001; Vosse and Kempen 2000, Rushton 2004),
• information retrieval (Crestani 1997).

However, 'activation of words alone is not sufficient to account for understanding of sentences' (Jackendoff 2002:58) so the WG theory of processing rests on a distinctive combination of other assumptions:

• As explained in section 0, the network is **symbolic** rather than distributed, so each node or link corresponds to an identifiable concept.
• Processing is highly **interactive** rather than modular. A single very general-purpose mechanism (outlined below) is responsible for all processing of symbolic structures, whether inside language or outside, whether in production or perception, and across all 'levels' of language. There are several other interactive models for sentence-comprehension (e.g. McClelland and Rumelhart 1988, Macdonald, Pearlmutter and

Seidenberg 1994; McRae, Spivey-Knowlton and Tanenhaus 1998; Roland 2001; Sturt, Pickering, Scheepers and Crocker 2001; Vosse and Kempen 2000), but many of these models divide processing into a series of stages which apply in a fixed order (Levelt, Roelofs and Meyer 1999a). One of the attractions of highly interactive models is the possibility of using contextual information (which in these models is part of the same network as the grammar) to guide language processing, for example by resolving ambiguities. Again, there are other models of general and linguistic knowledge which explain how these interact, notably ACT-R and SOAR (Anderson and Lebiere 1998, Laird, Newell and Rosenbloom 1987), but both these large-scale systems combine a network architecture with procedures which trigger specific actions, which makes them very different from the purely declarative networks of WG.

• Following the principles outlined in the previous section, processing takes place in '**long-term working memory**'Ericsson and Kintsch 1995, rather than in a separate part of the mind called 'short-term memory'. The processor adds new temporary 'token' nodes to the permanent network, rather than simply tracing paths through the existing network. These token nodes, for transient items of experience, form a constantly changing fringe on the edge of the permanent network. When first created they are highly active, but their activity dissipates rapidly and most of them soon vanish from the network (or at least become unusable).

• There is a **typology of links** rather than an undifferentiated set of 'associations', and the processor treats different types of link in different ways. Isa links allow **multiple default inheritance**, while the argument and value links of section 0 allow relation nodes to classify other links and to pass spreading activation, via the Isa hierarchy, directly from relation to relation, rather than only via entity nodes.

• There is also a **typology of entities** which distinguishes stored **types** from **tokens**. As with links, the typology affects the way in which the processor treats nodes. In particular, the procedure of binding only applies to tokens.

The following account of WG processing will develop these claims. The leading idea in all the psycholinguistic research cited earlier is that the network is not just a static collection of nodes and links, but a highly active organizm in which the nodes and links may be 'active' in some metaphorical sense which ultimately translates into chemical and physical activity in neurons. A good comparison would be a circuit-board in a computer, which allows electrical charges to pass from node to node; but it is actually much more dynamic than that because the 'wiring' is constantly changing in a way that will become clear below.

Consider a very simple non-linguistic example: what I do when I see a fly in the air. The main task is simply to recognize it as a fly, so my network has to establish a connection between it and my general concept Fly. In terms of network activity, this requires the following operations:

• First create a node for the perceived object; call this node E (for 'Experience'). E is linked to its observed properties (size, colour, movements, noise and so on), which are stored as links to the relevant permanent concepts, so E is in the centre of a sub-network. Since we can't react to any experience until we have classified it, the top priority is to find a 'type', a permanently stored concept, of which E is an example; we can call this node T. At this stage, all I know (or at least hope) is that I shall be able to classify E, so I provisionally introduce a node for T and add an Isa link from E to T.

• Then let activation spread from the observed properties and converge on the node Fly, as the only node which combines them all. Bind T provisionally to this node, thus classifying E as a fly.
• Then apply default inheritance to inherit as much inheritable information as possible from Fly to E. (This inheritance may prioritize information which is already active and therefore relevant to the immediate context; for example, if I try to swat the fly, its movements are more relevant than its colour.)

This example has nothing to do with language and yet it contains all the ingredients of language processing:
• **Node** creation **and definition** - creating two new nodes for the current word token: E, for the experience itself, and T, for its 'type' (which may already be classified as a word, so T isa Word). The procedure is basically the same whether E is the word currently being perceived or the target of speech planning; but in one case it is the form that is already known whereas in the other it is the meaning. These new nodes are the current focus of attention, so they receive a great deal of activation which will continue until E is classified and otherwise 'dealt with'. E is, of course, linked to all its observed properties.
• **Spreading activation** which leads to **binding,** binding T to the stored node S which matches these attributes best. Spreading activation guarantees that S satisfies the **Best Fit Principle**, i.e. it ensures that S is a better model for E than any other stored concept is.
• **Default inheritance** - selectively inheriting other attributes from S to E.

We shall now consider these processes in more detail

1. Node creation and definition.
A word token is distinct from the corresponding type, so WG gives them distinct names (Hudson 1984: 24). For example, in the sentence *I speak English*, the word token *speak* might be called 'word 2' (or 'w2') but it isa the word type *Speak:present*. Most linguistic theories do not make this distinction explicit in their notation because they use ordinary orthography for labelling both the token and its type, but this is highly misleading because the two things have quite different properties - e.g. the token has a specific speaker/writer and time or place, but the type does not. Indeed, their properties can even conflict, as when the token is in some sense defective - e.g. mispronounced or mis-spelt. The conceptual distinction between the two is very clear and hardly a matter of dispute. Consequently, the first step in processing a word token is to assign a new conceptual node to it.

One of the most controversial claims in WG is that utterance tokens are 'part of the grammar'. (This is what I meant when I said that processing is done in 'long-term working memory', rather than in a separate 'short-term memory'.) Consequently, the conceptual node which represents a word token (or a token of any other unit) is linked to nodes in the permanent network. Ultimately it will isa some word type such as *Speak:present*, but even at the first stage the token must be connected to the network in order for the activation invested in the node for w2 to spread through its attributes to the permanent network. As mentioned earlier, tokens of experience can be thought of as a constantly changing 'fringe' attached to the permanent network, with the possibility that some of them may stabilize and become permanent. (This is the basis for learning as I shall explain in section 0.)

The same principles apply whether we are producing or perceiving (speaking or listening, writing or reading). In perception the token stands for an observed word, and the aim is to enrich it so as to discover its unobservable characteristics such as its

meaning. In production, the token is the target word and this time the enrichment will provide its pronunciation or spelling. Either way round, the token needs its own node, and this node will be enriched by integration into the network. As I explained earlier, the most active relation produces the most enrichment, so when listening, we devote most resources to meanings, and when speaking, to pronunciations. For example, when we perceive the word *speak* it is pronunciation or spelling that is observable, as well as a sentential environment consisting of the words already processed and a contextual environment consisting of a speaker, an addressee and so on; and the target is a meaning node which is waiting to be enriched under the guidance of activation from all these sources.

On the other hand, in production our starting state is some kind of meaning, plus the sentential environment so far and everything we know about the situation, the audience and so on. If we choose and say *I speak* it is because we are aiming at some word whose sense is Talking and which is compatible with *I* as its subject - in other words, our target is a finite verb. I am ignoring important questions about timing - no doubt *speak* has already been selected by the time the word *I* is uttered, but the point is simply that the words chosen have to be put together into a grammatical sentence structure. The meaning may not fully determine the choice of word - for example, the verb *talk* would have done equally well in other contexts. Just as in perception, therefore, production starts with a rich but incomplete definition of a word token, and the aim is to enrich it by consulting the grammar.

Another similarity between perception and production is that in both cases the token word is important to the user, so it receives considerable activation which spreads through the little network that defines it; and thanks to spreading activation, the nodes in this network (e.g. the constituent phoneme tokens) share their activation with nodes in the permanent network. As we shall see in the next step, this is what allows the target word type to be selected.

2. Spreading activation and binding.

Suppose a listener or speaker has identified some word token w2 and built a mental network for it as described above. In the processor's mind, w2 is linked to permanent concepts for its constituent sounds (in hearing) or for its meaning (in speaking), but the classification of w2 consists so far of nothing but the provisional token T, standing for some 'type'. The next step is to enrich T by binding it to at least one permanent word-type. This binding process is the basis for classifying new tokens of experience, but the same process in fact plays an even larger role in processing because it goes well beyond mere classification and applies to all **bound tokens** – tokens which need to be bound to some other entity. For example, if I hear an example of *dog*, I can inherit for it a referent node and a syntactic parent node, which must be a determiner. Each of these nodes needs to be bound to some other node for enrichment, so I need a dog for the referent and a determiner for the parent. In other words, it is binding that is responsible for reference-assignment in pragmatics and also for finding grammatical relations ('parsing') in syntax. According to WG, all these processes – classification, reference-assignment and parsing, and perhaps other processes as well – are handled by a single mechanism. In the following paragraphs I shall present binding in relation to classification, but it should be borne in mind that the mechanism has a much wider application.

Binding applies, then, to impoverished tokens, and the aim of processing is to '**enrich**' these tokens by binding them to one or more other node. (The notion of enrichment is taken from Relevance Theory - Sperber and Wilson 1995.) We can

illustrate this by supposing that I hear the sounds [spi:k]. By step 1 I have represented this experience by the node E, with suitable links to the nodes for the constituent sounds. I also know that E must isa some word type, which is also represented by a node. The state of play is shown schematically in Figure 0.20, where the unknown category is shown provisionally as a question mark – a notation which I replace in the next paragraphs. All being well, I will decide that I have just heard someone say the word *speak*.

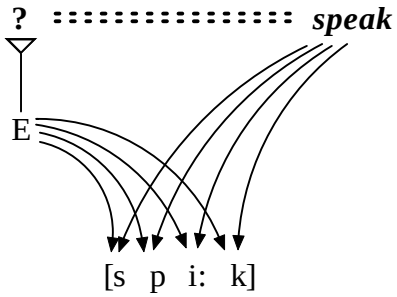


Figure 0.20: Binding an example of [spi:k] to the stored word *speak*

How does the processor know which nodes to bind? Binding applies only to token nodes, so the processor can ignore all stored nodes; but it only applies to a sub-set even of the tokens. For example, contrast the referents of definite and indefinite noun phrases such as *the dog* and *a dog*. The choice of *the* is a signal that the referent should be bound to some pre-existing 'dog' node. In contrast, *a dog* refers to a newly created node which needs no binding. This distinction can be made in the network by a property which is inherited by definite referents but not by indefinites. The property concerned involves **Identity**, but it is directional because it links a 'known' (which inherits it) to an 'unknown' (which will be found by the Binding procedure). For example, take this little story-opening:

(16) A man had a dog. The dog barked all night.

Both *a dog* and *the dog* have referents, but the referent of *the dog* inherits the property of being identical to some other node. This much is inherited by the word tokens concerned, but Binding will then establish an identity link from the referent of *the dog* to that of *a dog*. Using the obvious notation for identity (an extended '=' with a head to show directionality), the network after Binding will include the links in Figure 0.21. Identity joins Isa, Argument, Value, and Quantity on the list of primitive and unclassified relations; as far as I know it completes the list.

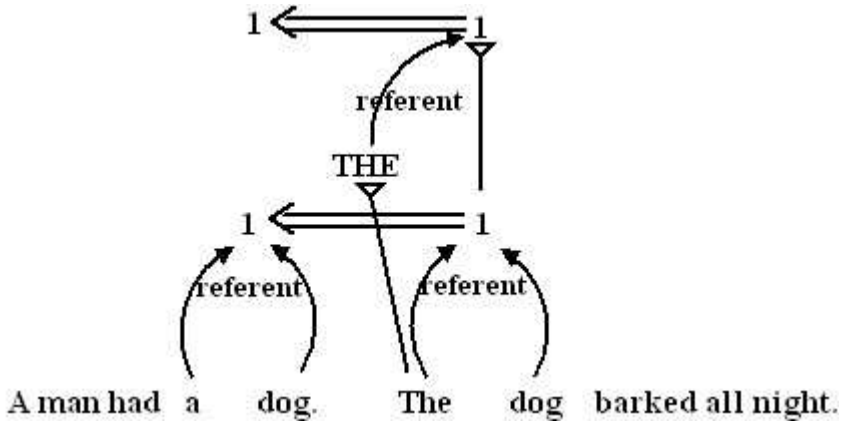


Figure 0.21: The anaphora from *the dog* to *a dog*

Given the relation Identity, therefore, the binding system 'knows' that a token needs to be bound if it inherits an Identity to link to one other node. This will be true for all classification, for all syntactic valents (i.e. expected dependents and parents) and for definite referents. The next question is how the processor chooses among the many thousands of available constants. We can start by noting that the search is usually limited to some general category of things; for example, in the case of word-recognition, the target must be a word because this is what the processor is expecting in the current context. Of course there are occasions where we hear a sound which we think is a word but which turns out to have been something else, such as a mere cough, but in general classification is a matter of refining an initial guess rather than starting completely from scratch. The task, therefore, is to select the best **eligible** candidate, where nodes qualify as eligible if they have an Isa link to the relevant super-category (such as 'word').

This choice is in turn guided by the **Best-Fit Principle**, which is familiar in AI (Winograd 1976, Luger and Stubblefield 1993:117) and fundamental to WG (Hudson 1984:20). The main feature of this strategy is to prefer the match which makes the best **global** fit, even if some of the individual attributes are 'wrong'. The mechanism behind the Best-Fit Principle struck me at one time as entirely mysterious (Hudson 1990:46) but I now believe it too can be explained in terms of spreading activation. The principle is very simple: The winner is **the most active eligible node**. For example, when we read the letters *speak* we take the known properties of our token w2, and look for a stored type which best fits everything we know about w2 at the point of processing the utterance *I speak* .... In this case the Best-Fit Principle works smoothly and without conflict, but it would have given the same decision even if the input had been the deviant *I speaks* because globally *speaks* matches *Speak:present* better than it matches any other stored word, and only deviates in one minor respect. The Best-Fit Principle seems psychologically plausible because it recognizes that we can classify deviant tokens while still noticing the deviations; and it is worth reiterating that this is only possible if w2 is a distinct node from its stored model, so that its properties can be different.

In production, Best Fit itself may be responsible for **speech errors**, which illuminate the activation which underlies speech production. These show that the most active node may not in fact be the 'correct' target. This can arise when a word is closely enough related to the target word to share some of its activation but also receives activation from some other source. For example, consider the attested

example (17), in which the target word was (presumably) *corporal* but the word actually selected was *capital* (Aitchison 1994:19):

(17) Corporal punishment is a last resort. It is difficult to use *capital* punishment in any institution. A beating is very valuable: it shows people you have come to the end of your tether.

Why did this mistake happen? We can only guess, of course, and whatever explanation we offer has to deal with the fact that the word *corporal* was correctly used in the previous sentence. A plausible explanation is that the phrase *capital punishment* is more frequent than *corporal punishment*; this is confirmed by a search on Google, and is probably true of everyday experience. However, both phrases are stored in memory and both are similar not only in meaning but also in pronunciation, as shown in Figure 0.22 (which, ideally, would also show relative activity levels based on frequency). Consequently, activity on one phrase automatically spreads to the other, so they are always in direct competition. In the first sentence the choice was made correctly for reasons that we can only guess at, but after this choice both nodes remain highly active and the higher frequency of *capital punishment* proved decisive.

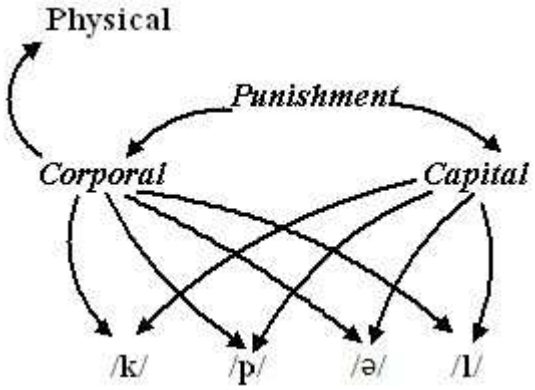


Figure 0.22: links needed to explain the use of *capital* instead of *corporal*

Examples such as this support the idea that Best Fit favours the most active candidate. One of the most interesting consequences of this principle is that it always favours the most specific candidate, because this is the one which collects activation from the most sources simply because it is the most highly specified candidate. For example, any word token could be classified simply as a word, but this would ignore the activation coming from its pronunciation and other information which distinguishes it from other words, so its link to some specific lexeme will always be stronger. By the same token, multiple class-membership will always be preferred to single class-membership because each class contributes more activation, so if a word can be classified in terms of inflectional categories as well as a lexeme, it will be. The intersection classes (e.g. *speak:present*) need not be available as stored nodes; the processor will seek all available super-categories in order to maximize activation, so it will find both the lexeme and the inflectional category as long as they are mutually compatible. We can therefore draw two general conclusions about classification: the search for a super-category will favour lower (more specific) nodes over higher nodes, and it will favour larger numbers of nodes over smaller numbers.

Returning to our example of *I speak English*, the output of the previous process was a network in which w2 was linked via all its observed properties to the corresponding properties stored in the permanent network. Activation spreads out

from w2 and affects all these stored nodes; for example, it goes from w2 to the phonemes /s/, /p/ and so on. Thence it spreads to all the word-forms which contain these phonemes, but the activation is spread so thinly that it dissipates very fast; so the winner is the form which receives activation from all the phonemes - the form *speak* - and all the others de-activate almost immediately. As I pointed out above, activation spreads via other routes as well, but in a simple case like this it all converges on the same answer, so by Best Fit, *speak* (or more precisely, *Speak:present*, since the pronoun *I* selects a finite verb) is the winner.

3. Default inheritance.

Both the previous steps are merely a preparation for this one, which provides the functional motivation for them all. Merely classifying a piece of experience is not in itself a useful activity; the benefit comes from all the enriching information which derives from this classification and which cannot be known otherwise. This is the result of default inheritance. In the case of speech perception this provides information about unobservables such as meaning and syntax; in speech production it provides non-semantic information such as morphology, pronunciation and (again) syntax.

Default inheritance is a process that takes time. In a classic experiment, (Quillian and Collins 1969) subjects were given English sentences such as 'A cat has fur', 'A cat has a heart' and 'A cat has wings', and their task was to decide whether each sentence was true or false. The dependent variable was the time taken to make this decision, and Collins and Quillian found that sentences like 'A cat has a heart' took longer to judge than did sentences like 'A cat has fur'. The obvious explanation for this difference is that the property of having a heart is stored at a higher level in the Isa hierarchy, maybe at the level of Animal, whereas fur is a memorable (and remembered) property of cats; so we retrieve fur simply by finding it ready-made among the characteristics of cats, but we have to infer hearts by inheritance. This experiment showed that inheritance takes time, but of course it does not prove that all inheritable properties are in fact inherited rather than retrieved directly. On the contrary, it is experiments like these that provide the evidence that I noted earlier which showed that properties which could be inherited may in fact be stored redundantly (Reisberg 1997: 269). For example, to judge by reaction times, we seem to store the property of having feathers not only with the super-category Bird, but also with particular species whose feathers are memorable, such as robins or peacocks. The main point of these experiments is that if properties are inherited, then this process takes a measurable amount of time.

There is also evidence that we take time to deal with exceptions which override defaults. This time the evidence comes from language, where regular and irregular morphology offer an ideal testing ground. Here we can be sure that an irregular past-tense form such as *took* is stored whereas a relatively rare regular one such as *extrapolated* is not. Given the results of Collins and Quillians's experiment, we might perhaps expect *took* to be easier to produce than *extrapolated*, because the latter involves inheritance rather than direct retrieval. However, the experimental results are actually the reverse of this expectation: irregular forms like *took* are slower than regulars like *extrapolated*. Moreover, the brain area activated for *took* is larger than (and almost includes) that for *extrapolated* (Jaeger, Lockwood, Kemmerer, Van Valin, Murphy and Khalek 1996). In short, the advantage of being stored directly is outweighed by the disadvantage of being irregular because irregularity involves

reconciling a competition between two forms: the stored irregular and the inherited regular form.

This finding is important for a theory of processing, because it excludes what at first sight might be an attractive theory of default inheritance. According to this theory, when we are searching for a past-tense form, we start at the bottom, with the most specific information we can find (e.g. the entry for the particular verb in question), and move up the Isa hierarchy until we find a form; and then we stop searching, so that we never in fact access the regular form. The extra time taken by irregulars shows that this must be wrong: we must retrieve both forms and choose between them by applying the logic of default inheritance outlined in section 0. In short, default inheritance has at least two separate components, each of which takes a measurable amount of time: climbing the Isa hierarchy in search of relevant properties, and choosing between any competing properties that may result from this search.

All the examples given so far have involved single words, but the same principles will in fact allow us to explain syntactic processing. Naturally this explanation presupposes the WG theory of syntax which is the topic of later chapters, but the most relevant fact about this theory is that words relate directly to one another via dependency links. To take a very simple example, the syntactic structure of (18) is as shown in Figure 0.23.

(18) I actually live in London.

Each word type has a dependency structure which is inherited by its tokens; for example, *live* needs a subject and a complement, *in* needs a complement and a parent, *actually* needs a word (such as a verb) to depend on. As I explained in section 0, inheritance automatically creates a new token for each inherited property, so each inherited dependency links the observed word token to an unknown one which is waiting to be bound. For example, the token *live* inherits not only the subject relation, but also the fact that the word concerned needs to bound to some other word token. What parsing does is to apply Best Fit to all these tokens which need binding to some other word in the sentence. Once all these identifications are done, the syntactic structure is complete.

Figure 0.23: Syntactic dependency structure of *I actually live in London.*

In this section I have reviewed the outlines of a very general procedure for applying any kind of network to any kind of 'experience'. The procedure applies equally to non-linguistic or linguistic behaviour, and to the understanding of other people's behaviour or to the planning of our own. The steps that have to be taken are as follows, using E as the name for the piece of experience in focus:

• **Node creation** and **node identification**, to produce a representation of E which includes all the information currently available - the perceived information about the incoming experience, or the partial description of the planned experience.

• **Spreading activation** activates some part of the network and **Best Fit Binding** selects the most active node (A) in the network.

• **Default inheritance** enriches the description of E (or any other token) once E isa some node F by creating a new copy of every property of F, and especially of properties which are highly active.

Learning

Learning raises two kinds of question:
• How do children learn the general properties of words, and in particular that words combine a sound with a meaning (and, eventually, that they have morphological structure, belong to word classes and have a syntactic valency)?
• How do they learn the specifics of individual words?
The first question is clearly more fundamental, and harder, than the second, and to some it seems too hard to explain in terms of learning. It also raises even more fundamental questions about why other primates cannot understand how words work. However, I do believe that learning is possible even for such abstract properties and in section **Error! Reference source not found.**. I shall offer a brief explanation of how infants may be cognitively prepared, unlike other primates, to learn one of the essential word properties, that words have a meaning. In this section I shall focus on the easier question of how children (or for that matter adults) learn new words and their properties, and in the process I shall try to put together a coherent theory of learning to accommodate this particular kind of learning.

The main thrust of the last section was that conceptual networks, including the language network, are dynamic. New links and new nodes are continually being established, and activation levels throughout the network are affected by spreading activation. There is a great deal of evidence that these effects of experience are not ephemeral but, at least in some cases, very long-lasting indeed. The most obvious examples of this are the 'recency effect' and the 'frequency effect', which show that words are more accessible if they have been used recently or frequently (Reisberg 1997:51). If we explain these effects in terms of spreading activation, it seems that activating a node has a more or less long-term affect on it, making it more easily activated on future occasions. This is the variable called **'entrenchment'** in Cognitive Grammar (Langacker 2000), and once again one advantage of a network model of language is the possibility of at least debating this important variable, and possibly even finding a suitable theoretical basis for it. Unfortunately this is an area of WG which has not yet been developed except in relation to sociolinguistic data which will be discussed briefly in section **Error! Reference source not found.**.

This dynamic interaction between the network and experience is the basis for **learning** - i.e. for permanent extensions to the network. I am impressed by the evidence for massively 'usage based' learning, in which the learner stores large numbers of very specific experiences - specific utterances of words or word-groups - and then uses this data-base as a source of inductive generalizations which constitute the grammar/lexicon (Bybee 1998; Langacker 2000; Ellis 2002; Tomasello 2003; Bod 1998). Inductive generalization is particularly evident in morphology, where it is generally agreed that children first store all observed forms, whether regular or irregular, and do not recognize general rules until they have stored a significant collection of regular forms. (As noted earlier, one consequence of this inductive approach is that at least some regular patterns must be stored, because generalization presupposes stored examples; see Jackendoff 1997:122.) I should like to show now how this kind of learning may be a by-product of the processes described in the previous section.

Suppose a child hears a new word in a sentence such as this:

(19) Let's get rid of those nasty germs.

The new word is *germs*. Segmentation is easy if the child already knows all the other words in the sentence, so let's assume that its sounds have been identified as a word-segment. At this point the child has created a new node w7 to represent this word, and knows already that w7 isa Word, which allows all the general characteristics of words (e.g. having a speaker and a referent) to be inherited; the child can even supply specific values for some of the inherited variables: the word's pronunciation, its speaker and its time. Spreading activation from the earlier words (especially the preceding determiner *those* and the adjective *nasty*) has already strongly activated the Common-noun node and the Plural-noun node; so Best Fit adds new Isa links from w7 to Common-noun and Plural-noun. So far, then, w7 isa Common noun, Plural noun and Word.

Once again default inheritance applies, giving *germs* the morphological and semantic structures of a typical plural noun. The morphological structure consists of a base and the suffix *s*; the details of this structure will be explained in chapter 2. This allows the child to identify the morpheme *germ* as the base of w7. The Plural-noun node also provides a schematic semantic structure, showing that the word refers to a set each of whose members isa the word's sense, so the child 'knows' that *germs* refers to a set of things each of which is called a germ. That is the end state after processing, unless the child can make a guess (right or wrong) about what kind of thing a germ might be.

What happens to w7 after this? One possibility is that it weakens (in some sense) to the point where it no longer counts as a part of the network. This is presumably the fate of the vast majority of word tokens, at least to the extent that they become inaccessible to any kind of retrieval system. A great deal of psychological research has shown that to the extent that we can remember sentences we remember them in terms of their content, not their exact wording (Harley 1995:313). Another possibility, however, is that, because of its novelty, w7 is sufficiently salient to receive a great deal of activation, and that this activation is sufficient to keep it accessible until the next time the child encounters the same word - in short, a token node turns into a type node simply by persisting in memory.

It could be objected that this is logically impossible because types and tokens have completely different statuses; after all, in most theories types belong to competence whereas tokens belong to performance. But I have already explained that this is not so in WG: although token nodes and type nodes can be distinguished, they both have the same formal status in the network. Admittedly, types and tokens have different psychological statuses since one comes from memory while the other comes from perception or planning; but even this contrast is blurred by those word-tokens that we can recall from memory, of which we all have a large stock. We can all recall individual datable tokens of particular words – tokens which stand out in memory for some reason such as being our first encounter with them. In WG these tokens are permanently stored as examples of their respective types, from which most of them are distinguished only by the fact that they have specific values for the deictic categories of time, place, speaker, addressee and so on. For example, our imaginary child may remember the word *germs* for some time, together with some of the details of who used it and when; and the same may even be true of the new type *germ* which *germs* isa.

It could also be argued that a node that carries specific deictic details which tie it to a specific situation cannot be a type because types are by definition general; so

even if we remember a token, it is still just a token, not a type. It only becomes a type by losing its specificity, and only then can it be used as the super-category for another token. However this argument ignores the effects of Best Fit and Default Inheritance. Even if the child's memory of the first token of *germs* is tied to a particular time and speaker, another token of the same word will strongly activate this memory and this activation will be enough for Best Fit to choose it as the new word's super-category. The deictic contradictions between them do not prevent this identification because Default Inheritance allows defaults to be overridden. In any case, it seems likely that most deictic details will fade into oblivion through the normal processes of memory loss, so most tokens will automatically become more abstract the longer they are stored.

My proposal, then, is that our first encounter with a word produces a new node for that token, which is attached to everything we know about it – who said it, when they said it and who they said it to, as well as its observable form (whether pronunciation or spelling) and (probably) a word class and (possibly) a meaning. Since it is a new word, we don't assign it to an existing lexeme, so we register it as new and therefore interesting and important. Its novelty has the effect of distinguishing it from tokens of familiar words, and prolonging its life so that it may act as a super-category for the next token of the same word. If the next token enriches the description of the word (e.g. by providing a richer or different meaning), it too will survive, so gradually the stored information about this word becomes more and more rich and informative. Best Fit guarantees that the richest node will always be selected so this is the one which will prosper, while its poorer relatives fade away and become less and less accessible. In other words, the richest concept becomes the 'official' representation for the item concerned.

Here then are the elements of my account of how we learn a new word W:

- We hear a token of W and create a node for it, called E.
- We store all the known characteristics of E, including:
  - its observable form (spoken or written),
  - its deictic characteristics (time, place, speaker, addressee, etc.),
  - its high-level classification as a word,
  - any general characteristics that can be inherited from Word, including a new node for its meaning.
- We apply Best Fit to find as informative a super-category for E as possible, given the currently active nodes; these reflect the morphology, the grammatical context and the conceptual context, so they may produce more high-level classifications in terms of syntactic and semantic categories (e.g. Plural-noun for *germs* and Set, Nasty and Invisible for its meaning).
- We add these inferred characteristics to the store of known characteristics of E and its meaning, giving:
  - its word class(es)
  - its rough meaning
- All the preceding steps are parts of normal processing, but unlike most word tokens E does not fade from memory; because of its novelty it remains accessible to future processing. In other words, this token node E turns into a type node simply by staying active and ‘alive’.
- The next token of W isa E by the usual classification procedure. If its characteristics add to those of E, it survives and replaces E as the provisional representation of W. This process repeats until the internalized representation of W stops changing because there is nothing more to learn.

This theory has the attraction of explaining how we can learn a new word (or any other kind of concept) after meeting it just once, while also allowing subsequent experiences to enrich and correct the first attempt.

Individual lexical items are the 'basic-level categories' of language (Rosch 1976) - the most informative categories, which combine the largest number of non-inherited characteristics, and provide the best fit between form and function. Outside language they are fundamental to learning; for example, we presumably learn Chair and Table before we learn the higher-level category Furniture and lower-level distinctions between types of chairs and tables. Similarly in language: we learn individual lexical items before word classes, so the above account of how this happens is fundamental to a theory of language learning. However this theory also needs an explanation for how grammar goes beyond the individual lexeme in two directions: in terms of size and in terms of generality. The first produces syntax, and the second produces rules and generalizations.

Syntax is already implicit in the account of how we learn lexemes if, as in WG, syntax consists of nothing but pair-wise links among words. (This is the main theme of the later chapters on syntax; see chapter **Error! Reference source not found.** for a summary.) The accompanying words are highly salient characteristics of a word token, so if a child hears the utterance *Dogs bark*, it can store the fact that they occurred next to each other in this order along with all the other information stored about each word separately. Stored word-sequences are the basis for learning dependencies because most of the time adjacent words are in fact linked by a dependency; in English, for example, estimates of the number of words that depend on an immediately adjacent word range from 63% (Pake 1998) to 78% (Eppler 2004:156-8) for conversation and one estimate for written English is 74% (Collins 1996). In other words, most words have a syntactically relevant link to the preceding word (either as dependent or as parent). Moreover, words that are not adjacent are much less likely to have a significant relation, so a child benefits greatly from having a limited span of only two words since this helps to filter out irrelevant links (Elman 1993).

This strong tendency for adjacency to favour syntactic links means that a strategy based on nothing but adjacency will provide a very useful database of word pairs for future learning; but of course actual language learners learn meanings alongside words, so they can in fact distinguish the semantically relevant links from the irrelevant ones. For example, *buy cherry yogurt* contains two adjacent pairs, one of which does show a semantically relevant link (*cherry yogurt*) while the other does not (*buy cherry*). Presumably adjacent pairs are more likely to be stored for future reference if they are also related semantically on the principle that rich links attract more activation. As the dependency system becomes more sophisticated the learner can rise more and more above mere adjacency, but adjacency is a very good starting point.

The other direction for growth is towards increasing **generality**. According to the 'usage-based' approach described in section 0, learning is based on experience and generalizations are built by induction from stored examples of experience. Inductive generalizations produce the stuff of grammar - word classes, constructions, dependency types, word order rules and so on. The benefits of higher-level categories are very clear, and especially so in learning; for example, in our earlier example the child could infer that *germs* was a plural noun because it already knew the categories Plural noun and Noun, and this in turn allowed the form *germs* to be segmented into a base and a suffix. However it is less clear exactly how induction works in a network

as described so far, and it may well involve psychological processes that go beyond those that I have assumed so far. The following remarks are pure speculation even as a model of mind, let alone of its underlying neurology.

Somehow the learner's mind 'spots' a similarity among a range of nodes (which we can call A, B, C) and creates a new node D such that:

• A, B, and C all isa D

• D has the characteristics which A, B, and C share.

One possible explanation is that we have a special 'induction mechanism' which randomly activates nodes during slack periods (e.g. during sleep) in search of correlations - bundles of two or more characteristics that tend to occur on the same nodes. Suppose the characteristics that A, B, and C share are their links to two other nodes, X and Y. In that case, activity on both X and Y will make A, B and C more active than any other nodes, which indicates a correlation between their links to X and Y. Following Hebb's principle that 'nodes that fire together wire together' (Hebb 1949), the induction mechanism creates an explicit link among A, B, and C ('wires them together') by building an Isa link from each one to a new super-category D. Once this super-category exists, it will inevitably attract all other nodes that have similar links to X and Y and presumably its properties can also become richer in the same way as I suggested above for new lexeme-type nodes.

Whatever the mechanism, it is clear that inductive generalization is a life-long process. For example, a detailed study of irregular past tenses such as *kept* and *told* showed that speakers are more likely to recognize them as a distinct sub-class of verbs as they become older (Guy and Boyd 1990). The speakers (in Philadelphia, USA) sometimes 'drop' the final t/d from these words by a process called t/d deletion which applies more frequently in mono-morphemic words such as *apt* than in regular bi-morphemes like *walked*. Young children never use the suffix t/d in irregular verbs like *kept* and *told*, but at some point in later life everyone uses it at the same rate as in mono-morphemes, which shows that they have not yet recognized the possibility of a morpheme boundary. However some adults later reduce their 'dropping' rate in these irregular verbs to that of bi-morphemes, from which we may conclude that they have recognized that these words form a distinct group which contains a semi-regular suffix alongside an irregular base – a clear example of late learning based on induction.

If new concept nodes can be created by induction, the same must be true of relation-types. The discussion in section 0 (around Figure 0.6) led to the conclusion that non-primitive relations are also concepts (even if my simplified diagrams do not show them as nodes.), so these must be generalizable by the same inductive processes as entity concepts. Once again the induction mechanism looks for correlations which are revealed by random activation, but this time it is on relations rather than entities that the activation converges. Imagine two relations R1 and R2 whose tendency to link the same pairs of nodes is revealed by random activation; the result is the creation of a super-node which is 'defined' in terms of R1 and R2. This is the kind of process that explains abstract relations, including those of syntax. Take the Subject relation, for example. This is famous for bringing together a bundle of disparate characteristics from word order to semantics (Keenan 1976), each of which is a simpler relation such as Before (word order) or Agent (meaning). The induction mechanism just sketched explains how the correlations among these simpler relations can lead to the creation of a super-relation which has the simpler ones as its inheritable characteristics. Each of the relations concerned helps to define the others and to make them available, by inheritance, during processing.

To summarize this discussion, I have made the following rather tentative suggestions about how basic-level lexical items are extended in terms of both length and generality. Two-word syntactic constructions (dependencies) can be induced from stored pairs of adjacent words, most of which are in fact linked by syntactic dependency and some semantic relations; these dependencies can be stored as facts about the words concerned, and recurrent patterns will reinforce each other. More general categories are built, by induction, out of the basic-level lexemes. It is possible that inductive generalizations are spotted by a random activation-generator which discovers correlated link-patterns, and which then records the correlation by creating shared super-category nodes. The same process applies to the relations among lexemes, so that increasingly general and abstract syntactic (and other) relations can be induced. In short, 'rules' are learned and stored as facts about general categories which are (therefore) inherited by their members.

Evaluating the theory

The theory that I present in this book is primarily intended to specify the nature of language structure, but a background assumption is that this cannot be done in isolation. This theory must meet up sooner or later with theories of how the structure is used and learned, and of how other kinds of knowledge are structured, used and learned. It would be very easy to build a theory which failed at the last post because it failed to mesh with established psychology, so my view is that the integration should happen sooner rather than later: better to build some elementary psychology into the theory from the start than simply to hope for the best and leave it till later. In short, a theory of language structure can and should aim at the 'psychological reality' that has been on the agenda for some decades now (Chomsky 1965, Lamb 1971, Bresnan 1978). Moreover, just the same arguments apply to the relations among analyses at different levels of language: sooner or later they must meet up, so the sooner the better. The aim of this theory, therefore, is to integrate the structures at one level with those at the other levels as well as with more general conceptual structures.

This rather ambitious aim makes evaluation problematic. The standard criteria for any linguistic theory still apply, so the theory must allow accurate and revealing solutions to well-known descriptive problems. This bread-and-butter work has taken up all my working life, and I include in this book a number of examples to show that at least some problems are soluble within the WG framework. The main show-piece is an extended discussion of gerunds in English (chapter **Error! Reference source not found.**), but the book also outlines descriptions of other complex phenomena, of which the following are just a sample:

- Latin verb morphology (section **Error! Reference source not found.**)
- Slovene noun morphology (section **Error! Reference source not found.**)
- Beja clitics (section **Error! Reference source not found.**)
- Serbo-Croatian clitics (section **Error! Reference source not found.**)
- German Partial VP Fronting (section **Error! Reference source not found.**)
- Zapotec prepositional pied-piping (section **Error! Reference source not found.**)
- Icelandic case agreement (section **Error! Reference source not found.**).

Each of these discussions supports some part of the general theory, but this also rests on a great many other descriptive analyses which I mention in passing. These briefer discussions go well beyond the core areas of morphology and syntax into semantics and sociolinguistics. The obvious gap remains phonology, both segmental and prosodic.

Any linguist can evaluate these analyses in relation to the facts and to other analyses of the same data expressed in terms of other theories. However the most important fact about them is that whether they involve morphology, syntax, semantics or sociolinguistics, they all assume the same theory. A theory of language structure which integrates separate sub-theories for morphology, syntax and so on is more comprehensive and therefore more explanatory than a library of unintegrated theories for different levels; so given a straight choice between the single theory and the library of theories, the single theory must always win.

The problems of evaluation multiply when we look beyond language. At this point, of course, the best judges are psychologists. When I make claims about spreading activation and its effects in priming and speech errors, I think I am simply repeating what can be found in virtually any textbook of psychology (e.g. Reisberg 1997). The idea that spreading activation implies a network is both obvious and widely accepted among psychologists, though I recognize that some psychologists are uneasy about the idea of using nothing but networks to model knowledge:

> There is surely widespread agreement that memory does draw on associative processes and spreading activation. There is likewise no doubt that network theorizing can encompass an enormous range of memory data. But there is considerable uncertainty about whether network theorizing, either in a traditional version or in PDP [Parallel Distributed Procesing], can explain all of mental functioning. This is still 'work-in-progress' on an immensely complex and subtle topic – merely the task of describing All of Knowledge. Moreover, we can take considerable comfort from the fact that, unsolved mysteries or no, we have at least a part of the puzzle under control. (Reisberg 1997:303)

However, I believe that a linguist may have an important contribution to make in this debate about psychological theory because the structure of language is so much better understand than any other area of knowledge. What I am offering is a theory of networks which accommodates all the complexity that linguists know about; and in particular, which includes a theory of how relations are classified (which is one of the main weaknesses in associative theories). So far as I know, psychologists have never considered a network of this type, so all the evaluation remains to be done.

Another characteristic of WG networks is the procedure for enriching token nodes through default inheritance. This idea belongs to Artificial Intelligence rather than to psychology, though it also explains the prototype effects that psychologists find in categorization (section 0). However, default logic is very controversial in those parts of the AI world (and of logic) which prefer 'clean' solutions; after all, a logic which allows earlier conclusions to be overridden later is a potential disaster not only in terms of logic but also in terms of computer programming (Touretzky 1986). After a survey of the problems, one textbook concludes:

> Unfortunately, most commercially available inheritance software does not provide a clean enough implementation of inheritance to avoid these problems. This is because many of these problems have not yet been solved or the solutions that are available are either too new or too inefficient to affect the design of current programs. (Luger and Stubblefield 1993:389)

However I believe that the approach to default inheritance that I describe in section 0 avoids most of the problems described in this literature by restricting inheritance to tokens. As with the design of networks, I believe this is an innovation so it remains to be evaluated.

Since the research tool of AI is computer modelling rather than experimentation, it may be that the only way to evaluate this area of WG is to build computer models and to match their performance against observed human performance, warts and all. A computer model would fail if it performed differently from the typical human being, regardless of whether its behaviour was worse or better; for example, it should make some errors (so long as these were like the errors that humans make), and it should take longer to retrieve a rare word than a common one. This approach to theory evaluation is already quite familiar in psycholinguistics (e.g. Levelt, Roelofs and Meyer 1999a), and it would certainly be a good way to evaluate WG. If WG is right, it should be possible to apply a single 'inference engine' equally successfully to networks for any area of language or for other kinds of knowledge such as kinship systems and social behaviour. Once again, this research has not yet been done, though a start has been made on a general-purpose network simulator (called Babbage) which can be adapted to different network models, including WG. (Interested readers should consult the Babbage website at www.babbagenet.org; the software is being developed by Mark P. Line.)

In conclusion, therefore, this book offers a single unified theory for language as well as for other kinds of knowledge, but its various parts need to be evaluated in different ways. I feel relatively confident about the strictly linguistic claims to the extent that I have tested them in my own research (though of course I know that there are plenty of phenomena that I haven't even tried to deal with). But in the areas of overlap with psychology and AI, I am merely offering a new theory. Ideally I would have offered new research evidence to support the new parts of this theory, but I hope the theory already has enough support to justify further evaluation.

References

Abeillé, A. (2004). *Building and using Parsed Corpora.* Dordrecht: Kluwer.

Abney, S. (1987). *The English noun phrase in its sentential aspect.* PhD. MIT.

Aitchison, J. (1994). *Words in the Mind. An Introduction to the Mental Lexicon. Second Edition.* Oxford: Blackwell.

Allerton, D. (1994). 'Valency and valency grammar', in Asher, R.(ed.), *Encyclopedia of Language and Linguistics*. Oxford: Pergamon. 4878-4886.

Anderson, J. R. (1983). A spreading activation theory of memory. Journal of Verbal Learning and Verbal Behavior 22: 261-295.

Anderson, J. R. and Lebiere, C. (1998). *The Atomic Components of Thought.* Hillsdale, NJ: Erlbaum.

Anderson, J. (1977). *On case grammar: Prolegomena to a theory of grammatical relations.* London: Croom Helm.

Anderson, S. (1992). *A-morphous morphology*. Cambridge: Cambridge University Press.

Anderson, S. (1996). How to put your clitics in their place, or why the best account of second-position phenomena may be something like the optimal one. The Linguistic Review 13: 165-191.

Andrews, A. (1982). 'The representation of case in Modern Icelandic", in Bresnan, J.(ed.), *The Mental Representation of Grammatical Relations*. Cambridge, Mass.: MIT Press. 427-503.

Aronoff, M. (1976). *Word Formation in Generative Grammar.* Cambridge, MA: MIT Press.

Aronoff, M. (1994). *Morphology by Itself. Stems and inflectional classes.* Cambridge, MA: MIT Press.

Baddeley, A. and Logie, R. (1999). 'Working memory: The multiple-component model', in Miyake, A. & Shah, P.(eds.), *Models of Working Memory. Mechanisms of Active Maintenance and Executive Control.* Cambridge: Cambridge University Press. 28-61.

Baker, M. (1985). Syntactic affixation and English gerunds. Proceedings of the West Coast Conference on Formal Linguistics 4: 1-11.

Barabási, A.-L. (2003). *Linked: How everything is connected to everything else and what it means for business, science and everyday life.* London: Penguin.

Barlow, M. and Kemmer, S. (2000). *Usage Based Models of Language*. Stanford: CSLI.

Bates, E. (1998). Construction grammar and its implications for child language research. Journal of Child Language 25: 462-466.

Bauer, L. (2003). *Introducing Linguistic Morphology (Second edition)*. Edinburgh: Edinburgh University Press.

Beard, R. (1994). 'Lexeme-morpheme base morphology', in Asher, R.(ed.), *Encyclopedia of Language and Linguistics*. Oxford: Pergamon. 2137-2140.

Berko Gleason, J. (1958). The child's learning of English morphology. Word 14: 150-177.

Bharati, A., Chaitanya, V., and Sangal, R. (1995). *Natural Language Processing. A Paninian Perspective.* New Delhi: Prentice-Hall of India Private Ltd.

Biber, D., Johansson, S., Leech, G., Conrad, S., and Finegan, E. (1999). *Longman Grammar of Spoken and Written English*. London: Longman.

Blake, B. (1990). *Relational Grammar*. London: Croom Helm.

Blevins, J. P. (2001). Paradigmatic derivation. Transactions of the Philological Society 99: 211-222.

Blevins, J. P. (2003). Stems and paradigms. Language 79: 737-767.

Bloomfield, L. (1933). *Language*. New York: Holt, Rinehart and Winston.

Bock, K. and Griffin, Z. (2000). The persistence of structural priming: Transient activation or implicit learning? Journal of Experimental Psychology-General 129: 177-192.

Bod, R. (1998). *Beyond Grammar : An Experience-Based Theory of Language*. Stanford: CSLI.

Borer, H. (1992). 'Clitics: pronominal clitics', in Bright, W.(ed.), *International Encyclopedia of Linguistics*. Oxford: Oxford University Press. 270-271.

Branigan, H. P., Pickering, M. J., Liversedge, S. P., Stewart, A. J., and Urbach, T. P. (1995). Syntactic priming: investigating the mental representation of language. Journal of Psycholinguistic Research 24(6): 489-506.

Bresnan, J. (1978). 'A realistic transformational grammar', in Halle, M., Bresnan, J., & Miller, G.(eds.), *Linguistic Theory and Psychological Reality*. Cambridge, MA: MIT Press. 1-59.

Bresnan, J. (1994). Locative Inversion and Universal Grammar. Language 70: 72-131.

Bresnan, J. (1997). 'Mixed categories as head sharing constructions.', in Butt, M. & Holloway King, T.(eds.), *Proceedings of the LFG97 Conference.* Stanford: CSLI Publications.

Bresnan, J. (2001). *Lexical-Functional Syntax*. Oxford: Blackwell.

Broadwell, G. (1999). Broadwell, George A. (1999). Focus alignment and optimal order in Zapotec. Proceedings of
the 35th Chicago Linguistic Society. Proceedings of the Chicago Linguistic Society 35:

Bröker, N. (1998). 'A projection architecture for dependency grammar and how it compares to LFG.', in Butt, M. & Holloway King, T.(eds.), *Proceedings of the LFG 98 Conference*. Stanford: CSLI.

Bröker, N. (2000). Unordered and non-projective dependency grammars. Traitement Automatique Des Langues 41: 245-272.

Bröker, N. (2001). 'Formal foundations of dependency grammar', in Ágel, V.(ed.), *Dependency and Valency. An International Handbook of Contemporary Research.* Berlin: Walter de Gruyter.

Brookes, A. and Hudson, R. (1982). 'Do linguists have anything to say to teachers?', in Carter, R.(ed.), *Linguistics and the Teacher*. London: Routledge and Kegan Paul. 52-74.

Brooks, P. and Macwhinney, B. (2000). Phonological priming in children's picture naming. Journal of Child Language 27: 335-366.

Brown, D., Corbett, G., Fraser, N., Hippisley, A., and Timberlake, A. (1996). Russian noun stress and network morphology. Linguistics 34: 53-107.

Brown, P. and Levinson, S. (1987). *Politeness. Some universals in language usage. 2nd edition*. Cambridge: Cambridge University Press.

Browne, A. and Sun, R. (2001). Connectionist inference models. Neural Networks 14: 1331-1355.

Bybee, J. (1995). Regular Morphology and the Lexicon. Language and Cognitive Processes 10: 425-455.

Bybee, J. (1998). The emergent lexicon. Proceedings of the Chicago Linguistics Society 34: 421-435.

Bybee, J. (1999). Use impacts morphological representation. Behavioral and Brain Sciences 22: 1016-+.

Bybee, J. and Moder, C. (1983). Morphological classes as natural categories. Language 59: 251-270.

Camdzic, A. and Hudson, R. (2007). Serbo-Croat Clitics and Word Grammar. Research in Language (University of Lodz) 4:

Carroll, J., Minnen, G., and Briscoe, T. (2004). 'Parser Evaluation. Using a Grammatical Relation Annotation Scheme', in Abeille, A.(ed.), *Building and Using Parsed Corpora.* Dordrecht: Kluwer.

Carstairs-McCarthy, A. (1992). *Current Morphology*. London: Routledge.

Carstairs-McCarthy, A. (1998). 'Paradigmatic structure: Inflectional paradigms and morphological classes', in Spencer, A. & Zwicky, A.(eds.), *The Handbook of Morphology*. Oxford: Blackwell. 322-334.

Carston, R. (1997). Relevance-theoretic pragmatics and modularity. UCL Working Papers in Linguistics 9:

Chametzky, R. (2003). 'Phrase structure', in Hendrick, R.(ed.), *Minimalist Syntax*. Oxford: Blackwell. 192-235.

Chang, F., Dell, G. S., Bock, K., and Griffin, Z. (2000). Structural priming as implicit learning: A comparison of models of sentence production. Journal of Psycholinguistic Research 29: 217-229.

Charniak, E. (1981). The case-slot identity theory. Cognitive Science 5: 285-292.

Chipere, N. (2003). *Understanding Complex Sentences: Native Speaker Variation in Syntactic Competence*. London: Palgrave Macmillan.

Chomsky, N. (1957). *Syntactic Structures*. The Hague: Mouton.

Chomsky, N. (1965). *Aspects of the Theory of Syntax.* Cambridge, MA: MIT Press.

Chomsky, N. (1970). 'Remarks on nominalizations', in Jacobs, R. & Rosenbaum, P.(eds.), *Readings in Transformational Grammar*. Waltham, MA: Ginn. 184-221.

Chomsky, N. (1995a). 'Categories and Transformations. ', in Chomsky, N.(ed.), *The Minimalist Program.* Cambridge, Mass.: MIT Press. 219-394.

Chomsky, N. (1995b). *The Minimalist Program*. Cambridge, MA: MIT Press.

Clark, E. (1993). *The Lexicon in Acquisition*. Cambridge: Cambridge University Press.

Collins, M. (1996). A new statistical parser based on bigram lexical dependencies. Proceedings of the Association for Computational Linguistics 34: 184-191.

Corbett, G. and Fraser, N. (1993). Network morphology: a DATR account of Russian nominal inflection. Journal of Linguistics 29: 113-142.

Covington, M. (1984). *Syntactic Theory in the high middle ages.* Cambridge: Cambridge University Press.

Cowan, N. (1997). *Attention and Memory: An integrated framework.* New York: Oxford University Press.

Cowan, N. (1999). 'An embedded-processes model of working memory.', in Miyake, A. & Shah, P.(eds.), *Models of Working Memory. Mechanisms of Active Maintenance and Executive Control.* Cambridge: Cambridge University Press. 62-101.

Creider, C. (2002). 'Swahili verbal inflection in theoretical perspective.', in Sugayama, K.(ed.), *Studies in Word Grammar*. Kobe: Research Institute of Foreign Studies, Kobe City University of Foreign Studies. 33-46.

Creider, C. and Hudson, R. (2006b). 'Case Agreement in Ancient Greek: implications for a theory of covert elements', in Sugayama, K.(ed.), *Kensei's Next Book.*

Creider, C. and Hudson, R. (2006a). 'Case Agreement in Ancient Greek: implications for a theory of covert elements', in Sugayama, K.(ed.), *Kensei's Next Book.*

Creider, C. and Hudson, R. (1999). Inflectional Morphology in Word Grammar. Lingua 107: 163-187.

Creider, C. and Hudson, R. (2006c). 'Case agreement in Ancient Greek: Implications for a theory of covert elements.', in Sugayama, K. & Hudson, R.(eds.), *Word Grammar. New Perspectives on a Theory of Language Structure*. London: Continuum. 35-53.

Crestani, F. (1997). Application of Spreading Activation Techniques in Information Retrieval. Artificial Intelligence Review 11: 453-482.

Croft, W. (1998). 'The structure of events and the structure of language.', in Tomasello, M.(ed.), *The New Psychology*. London: Lawrence Erlbaum. 67-92.

Croft, W. (2001). *Radical Construction Grammar - Syntactic Theory in Typological Perspective*. Oxford: Oxford University Press.

Croft, W. and Cruse, A. (2004). *Cognitive Linguistics*. Cambridge University Press.

Culicover, P. (1999). *Syntactic nuts: Hard cases, syntactic theory and language acquisition.* Oxford: Oxford University Press.

Deacon, T. (1997). *The Symbolic Species. The co-evolution of language and the human brain.* London: Penguin.

Denison, D. (1993). *English Historical Syntax.* London: Longman.

Denison, D. (1998). 'Syntax', in Romaine, S.(ed.), *The Cambridge History of the English Language*
Volume IV, 1776-1997. Cambridge: Cambridge University Press. 92-329.

Dik, S. (1991). 'Functional Grammar', in Droste, F. & Joseph, J.(eds.), *Linguistic Theory and Grammatical Description*. Amsterdam: Benjamins. 247-274.

Donner, M. (1986). The gerund in Middle English. English Studies 67: 394-400.

Dowty, D. (1991). Thematic proto-roles and argument selection. Language 67: 547-619.

Dowty, D. (2000). The Dual Analysis of Adjuncts/Complements in Categorial Grammar

David Dowty. ZAS Papers in Linguistics (Zentrum Für Allgemeine Sprachwissenschaft, Berlin) 17:

Durrell, M. (1996). *Hammer's German Grammar and Usage. 3rd edition.* London: Arnold.

Ellis, N. (2002). Reflections on frequency effects in language processing. Studies in Second Language Acquisition 24: 297-339.

Ellis, N. and Schmidt, R. (1998). Rules or associations in the acquisition of morphology? The frequency by regularity interaction in human and PDP learning of morphosyntax. Language and Cognitive Processes 13: 307-336.

Elman, J. (1993). Learning and development in neural networks - the importance of starting small. Cognition 48: 71-99.

Eppler, E. (2004). *The syntax of German-English code-switching.* PhD. UCL.

Ericsson, K. A. and Delaney, P. (1999). 'Long-term working memory as an alternative to capacity models of working memory in everyday skilled performance.', in Miyake, A. & Shah, P.(eds.), *Models of Working Memory. Mechanisms of Active Maintenance and Executive Control.* Cambridge: Cambridge University Press. 257-297.

Ericsson, K. A. and Kintsch, W. (1995). Long-term working-memory. Psychological Review 102: 211-245.

Evans, N., Brown, D., and Corbett, G. (2001). 'Dalabon pronominal prefixes and the typology of syncretism: a Network Morphology analysis. ', in Booij, G. & Marle, J. v.(eds.), *Yearbook of Morphology 2000*. Dordrecht: Kluwer. 187-231.

Fanego, T. (1996a). The development of gerunds as objects of subject-control verbs in English (1400-1700). Diachronica 13: 29-62.

Fanego, T. (1996b). The gerund in early modern English: Evidence from the Helsinki corpus. Folia Linguistica Historica 17: 97-152.

Ferrer i Cancho, R. (2004). Euclidean distance between syntactically linked words . Physical Review E 70: 056135

Ferrer i Cancho, R. and Solé, R. (2001). The small world of human language. Proceedings of the Royal Society Series B 268: 2261-2265.

Ferrer i Cancho, R., Solé, R., and Köhler, R. (2003). Patterns in syntactic dependency networks. Physical Review E 69: 1-8.

Ferrer i Cancho, R., Solé, R., and Köhler, R. (2004). Patterns in syntactic dependency networks. Physical Review E 69: 1-8.

Fillmore, C. (1968). 'The case for case', in Bach, E. & Harms, R.(eds.), *Universals in Linguistic Theory*. New York: Holt, Rinehart and Winston. 1-90.

Fillmore, C., Kay, P., and O'Connor, M. (1988). Regularity and idiomaticity in grammatical constructions: the case of let alone. Language 64: 501-538.

Fitch, W. T., Hauser, M., and Chomsky, N. (2006). The Evolution of the Language Faculty: Clarifications and Implications. Cognition

Fodor, J. (1983). *The Modularity of the Mind*. Cambridge, MA: MIT Press.

Fraser, N. and Corbett, G. (1996). Gender assignment in Arapesh: a Network Morphology analysis. Lingua

Frazier, L. (1985). 'Syntactic complexity', in Dowty, D. R., Karttunen, L., & Zwicky, A. M.(eds.), *Natural Language Parsing. Psychological, Computational and Theoretical Perspectives.* Cambridge: Cambridge University Press. 129-189.

Gaifman, H. (1965). Dependency systems and phrase-structure systems. Information and Control 8: 304-337.

Gentner, T., Fenn, K., Margoliash, D., and Nusbaum, H. (2006). Recursive syntactic pattern learning by songbirds. Nature 440: 1204-1207.

Gibson, E. (1998). Linguistic complexity: locality of syntactic dependencies. Cognition 68: 1-76.

Gibson, E. (2002). The influence of referential processing on sentence complexity. Cognition 85 : 79-112.

Giles, H. and Powesland, P. (1975). *Speech Style and Social Evaluation*. London: Academic Press.

Gisborne, N. (2006). 'Factoring out the subject dependency', in Sugayama, K. & Hudson, R.(eds.), *Not Known*. London: Continuum.

Gisborne, N. (1996). *English Perception Verbs*. PhD. UCL, London.

Givón, T. (1998). 'The functional approach to grammar', in Tomasello, M.(ed.), *The New Psychology of Language*. London: Lawrence Erlbaum. 41-66.

Goldberg, A. (1995). *Constructions. A Construction Grammar Approach to Argument Structure.* Chicago: University of Chicago Press.

Greenbaum, S. (1996). *The Oxford English Grammar*. Oxford: Oxford University Press.

Griffin, R. (1991). *Cambridge Latin Grammar*. Cambridge: Cambridge University Press.

Gruber, J. (1965). *Studies in Lexical Relations*. MIT.

Guy, G. (1994). The phonology of variation. Chicago Linguistics Society Parasession 30: 133-149.

Guy, G. and Boyd, S. (1990). The development of a morphological class. Language Variation and Change 2: 1-18.

Haider, H. (1990). 'Topicalization and other puzzles of German syntax. ', in Grewendorf, G. & Sternefeld, W.(eds.), *Scrambling and Barriers*. Amsterdam: Benjamins. 93-112.

Halle, M. and Marantz, A. (1993). 'Distributed morphology and the pieces of inflection.', in Hale, K. & Keyser, S.(eds.), *The View From Building 20: Essays in Linguistics in Honor of Sylvain Bromberger*. Cambridge, MA: MIT Press. 111-176.

Halliday, M. (1970). 'Language structure and language function', in Lyons, J.(ed.), *New Horizons in Linguistics*. Harmondsworth: Penguin. 140-165.

Halliday, M. (1977). 'Text as Semantic Choice in Social Contexts. 176-225', in Van Dijk, T. & Petofi, J.(eds.), *Grammars and Descriptions (Studies in Text Theory and Text Analysis)*. New York: Walter de Gruyter. 176-225.

Halliday, M. (1978). *Language as Social Semiotic*. London: Arnold.

Halliday, M. (1985). *An Introduction to Functional Grammar*. London: Arnold.

Halliday, M. (2002). *On Grammar*. New York: Continuum.

Harley, T. (1990). Environmental Contamination of Normal Speech. Applied Psycholinguistics 11: 45-72.

Harley, T. (1995). *The Psychology of Language*. Hove: Psychology Press.

Harris, Z. (1951). *Structural Linguistics*. Chicago: University of Chicago Press.

Haspelmath, M. (2002). *Understanding Morphology*. London: Arnold.

Hauser, M., Chomsky, N., and Fitch, W. T. (2002). The faculty of language: What is it, who has it, and how did it evolve? Science 298: 1569-1579.

Hawkins, J. A. (2001). Why are categories adjacent? Journal of Linguistics 37: 1-34.

Hebb, D. (1949). *The Organization of Behaviour*. New York: Wiley.

Heringer, H.-J., Strecker, B., and Wimmer, R. (1980). *Syntax. Fragen - Lösungen - Alternativen*. Munich: Wilhelm Fink Verlag.

Hiranuma, S. (1999). Syntactic Difficulty in English and Japanese: a Textual Study. UCL Working Papers in Linguistics 11: 309-322.

Hirst, G. (1988). 'Resolving lexical ambiguity computationally with spreading activation and Polaroid Words.', in Small, S., Cottrell, G., & Tanenhaus, M.(eds.), *Lexical Ambiguity Resolution: Perspectives From Psycholinguistics, Neuropsychology, and Artificial Intelligence*. San Mateo, CA: Morgan Kaufmann. 73-107.

Holmes, J. (2005). *Lexical Properties of English Verbs*. PhD. UCL, London.

Holmes, J. and Hudson, R. (2006). 'Constructions in Word Grammar', in Östman, J.-O. & Fried, M.(eds.), *Construction Grammar(s): Cognitive Dimensions*. Amsterdam: Benjamins.

Houston, A. (1989). 'The English gerund: syntactic change and discourse function.', in Fasold, R. & Schiffrin, D.(eds.), *Language Change and Variation*. Amsterdam: Benjamins. 173-195.

Huddleston, R. (1988). *English Grammar. An Outline*. Cambridge: Cambridge University Press.

Hudson, R. (2004a). Are determiners heads? Functions of Language 11: 7-43.

Hudson, R. (2006a). 'Buying and selling in Word Grammar', in Andor, J. & Pelyvás, P.(eds.), *Empirical, Cognitive-Based Studies In The Semantics-Pragmatics Interface*. Oxford: Elsevier Science.

Hudson, R. (1964). *A Grammatical Study of Beja*. PhD. University of London.

Hudson, R. (1971). *English Complex Sentences. An introduction to systemic grammar*. Amsterdam: North Holland.

Hudson, R. (1973). An 'item-and-paradigm' approach to Beja syntax and morphology. Foundations of Language 9: 504-548.

Hudson, R. (1974). A structural sketch of Beja. African Language Studies 15: 111-142.

Hudson, R. (1976a). *Arguments for a Non-transformational Grammar*. Chicago: Chicago University Press.

Hudson, R. (1976b). 'Beja', in Bender, M. L.(ed.), *The Non-Semitic Languages of Ethiopia*. East Lansing: African Studies Center, Michigan State University. 97-132.

Hudson, R. (1981a). Some issues on which linguists can agree. Journal of Linguistics 17: 333-344.

Hudson, R. (1981b). Wanna and the Lexicon. The Nottingham Linguistic Circular 10: 132-154.

Hudson, R. (1984). *Word Grammar*. Oxford: Blackwell.

Hudson, R. (1990). *English Word Grammar*. Oxford: Blackwell.

Hudson, R. (1992). *Teaching Grammar. A guide for the National Curriculum.* Oxford: Blackwell.

Hudson, R. (1995). Does English really have case? Journal of Linguistics 31: 375-392.

Hudson, R. (1996). *Sociolinguistics, 2nd edition*. Cambridge: Cambridge University Press.

Hudson, R. (1998). *English Grammar.* London: Routledge.

Hudson, R. (1999). Subject-verb agreement in English. English Language and Linguistics 3 : 173-207.

Hudson, R. (2000a). *I amn't. Language 76: 297-323.

Hudson, R. (2000b). Discontinuity. Traitement Automatique Des Langues. 41: 15-56.

Hudson, R. (2000c). 'Grammar Without Functional Categories.', in Borsley, R.(ed.), *The Nature and Function of Syntactic Categories*. New York: Academic Press. 7-35.

Hudson, R. (2001a). Clitics in Word Grammar. UCL Working Papers in Linguistics 13: 243-294.

Hudson, R. (2001b). Grammar teaching and writing skills: the research evidence. Syntax in the Schools 17: 1-6.

Hudson, R. (2001c). The educational context. Franco-British Studies (Journal of the British Institute in Paris)

Hudson, R. (2002). 'Richard Hudson', in Brown, K. & Law, V.(eds.), *Linguistics in Britain: Personal Histories*. Oxford: Blackwell. 127-138.

Hudson, R. (2003a). Case-agreement, PRO and structure sharing. Research in Language 1: 7-33.

Hudson, R. (2003b). Gerunds without phrase structure. Natural Language & Linguistic Theory 21: 579-615.

Hudson, R. (2003c). 'Mismatches in Default Inheritance', in Francis, E. & Michaelis, L.(eds.), *Mismatch: Form-Function Incongruity and the Architecture of Grammar*. Stanford: CSLI. 269-317.

Hudson, R. (2003d). Trouble on the left periphery. Lingua 113: 607-642.

Hudson, R. (2004b). Why education needs linguistics (and vice versa). Journal of Linguistics 40: 105-130.

Hudson, R. (2006b). Wanna revisited. Language 82:

Hudson, R. (2007a). 'Word Grammar', in Cuyckens, H. & Geeraerts, D.(eds.), *Handbook of Cognitive Linguistics*. Oxford: Oxford University Press.

Hudson, R. (2007b). 'Word Grammar, cognitive linguistics and second-language learning and teaching', in Robinson, P. & Ellis, N.(eds.), *Handbook of Cognitive Linguistics and Second Language Acquisition*. Lawrence Erlbaum.

Hudson, R. and Holmes, J. (2000). 'Re-cycling in the Encyclopedia. .', in Peeters, B.(ed.), *The Lexicon/Encyclopedia Interface.* Amsterdam: Elsevier. 259-290.

Hudson, R., Rosta, A., Holmes, J., and Gisborne, N. (1996). Synonyms and Syntax. Journal of Linguistics 32: 439-446.

Hudson, R. and Walmsley, J. (2005). The English Patient: English grammar and teaching in the twentieth century. Journal of Linguistics 41: 593-622.

Jack, G. (1988). The origins of the English gerund. Nowele 12: 15-75.

Jackendoff, R. (1977). *X-bar Syntax: A Study of Phrase Structure.* Cambridge, MA: MIT Press.

Jackendoff, R. (1990). *Semantic Structures*. Cambridge, MA: MIT Press.

Jackendoff, R. (1997). *The Architecture of the Language Faculty*. Cambridge, MA: MIT Press.

Jackendoff, R. (2002). *Foundations of Language*. Oxford: Oxford University Press.

Jaeger, J., Lockwood, D., Kemmerer, R., Van Valin, R., Murphy, B., and Khalek, H. (1996). A positron emission tomographic study of regular and irregular verb morphology in English. Language 72: 451-498.

James, L. and Burke, D. (2000). Phonological Priming Effects on Word Retrieval and Tip-of-the-Tongue Experiences in Young and Older Adults. Jornal of Experimental Psychology -Learning, Memory and Cognition 26: 1378-1391.

Jorgensen, E. (1981). Gerund and to-infinitive after it-is-(of)-no-use, it-is-no-good, and it-is-useless. English Studies 62: 156-163.

Joshi, A. and Rambow, O. (2003). 'A Formalism for Dependency Grammar Based on Tree Adjoining Grammar.', in Kahane, S. & Nasr, A.(eds.), *Proceedings of the First International Conference on Meaning-Text Theory*. Paris: Ecole Normale Supérieure.

Kaiser, L. (1997). 'CPR for Korean Type III Nominalizations', in Kaiser, L.(ed.), *Yale A-Morphous Linguistics Essays*. New Haven: Yale University, Linguistics Dept. 89-99.

Kaiser, L. (1999). *The Morphosyntax of Clausal Nominalization Constructions*. PhD. Yale.

Karlsson, F. (1995). *Constraint grammar :a language-independent system for parsing unrestricted text*. Berlin: Mouton de Gruyter.

Karmiloff-Smith, A. (1992). *Beyond Modularity. A developmental perspective on cognitive science.* Cambridge, MA: MIT Press.

Kay, P. (2002). An Informal Sketch of a Formal Architecture for Construction Grammar. Grammars 5: 1-19.

Kay, P. and Fillmore, C. (1999). Grammatical constructions and linguistic generalizations: The what's X doing Y? Construction. Language 75: 1-33.

Keenan, E. (1976). 'Towards a universal definition of 'subject'.', in Li, C.(ed.), *Subject and Topic*. New York: Academic Press. 303-333.

Kempson, R. and Quirk, R. (1971). Controlled activation of latent contrast. Language 47: 548-572.

Klein, D. and Manning, C. (2004). Corpus-Based Induction of Syntactic Structure: Models of Dependency and Constituency. Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL 2004) 42:

Kreps, C. (1997). *Extraction, Movement and Dependency Theory*. PhD. UCL.

Kuzar, R. (1998). Constructions: A construction grammar approach to argument structure. Journal of Pragmatics 29: 359-362.

Labov, W. (1972). *Sociolinguistic Patterns*. Oxford: Blackwell.

Labov, W. (1989). The child as linguistic historian. Language Variation and Change 1: 85-97.

Labov, W. (2001). *Principles of Linguistic Change, Volume 2: Social Factors*. Oxford: Blackwell.

Laird, J., Newell, A., and Rosenbloom, P. (1987). Soar: An architecture for general intelligence. Artificial Intelligence 33: 1-64.

Lamb, S. (1966). *Outline of Stratificational Grammar*. Washington, DC: Georgetown University Press.

Lamb, S. (1971). 'The crooked path of progress in cognitive linguistics.', in O'Brien, R. J.(ed.), *Linguistics: Developments of the Sixties - Viewpoint for the Seventies.* Washington, DC: Georgetown University Press.

Lamb, S. (1998). *Pathways of the Brain. The neurocognitive basis of language.* Amsterdam: Benjamins.

Langacker, R. (1998). 'Conceptualization, symbolization and grammar.', in Tomasello, M.(ed.), *The New Psychology of Language: Cognitive and Functional Approaches to Language Structure.* Mahwah, NJ: Erlbaum. 1-39.

Langacker, R. (2000). 'A dynamic usage-based model.', in Barlow, M. & Kemmer, S.(eds.), *Usage-Based Models of Language*. Stanford: CSLI. 1-63.

Langenhove, G. C. v. (1925). *On the origin of the gerund in English*. Grand: van Rysselberghe & Rombaut.

Lapointe, S. (1993). 'Dual lexical categories and the syntax of mixed category phrases.', in Kathol, A. & Bernstein, M.(eds.), *Proceedings of the Eastern States Conference of Linguistics.* 199-210.

Lecarme, J. (1978). *Aspects Syntaxiques des Complétives du Grec.* PhD. University of Montreal.

Levelt, W. J. M., Roelofs, A., and Meyer, A. S. (1999a). A theory of lexical access in speech production. Behavioral And Brain Sciences 22, 1-45.

Levelt, W. J. M., Roelofs, A., and Meyer, A. S. (1999b). A theory of lexical access in speech production. Behavioral And Brain Sciences 22: 1-+.

Levin, B. (1993). *English Verb Classes and Alternations. A preliminary investigation.* Chicago: University of Chicago Press.

Levin, B. and Rappaport Hovav, M. (1991). 'Wiping the slate clean: a lexical semantic exploration', in Levin, B. & Pinker, S.(eds.), *Lexical and Conceptual Semantics*. Oxford: Blackwell. 123-151.

Lewis, R. (1996). Interference in short-term memory: The magical number two (or three) in sentence processing. Journal of Psycholinguistic Research 25: 93-115.

Lieberman, P. (2002). On the nature and evolution of the neural bases of human language. American Journal of Physical Anthropology Supplement: Yearbook of Physical Anthropology 119: 36-62.

Lin, D. (2004). 'Dependency-based evaluation of Minipar', in Abeille, A.(ed.), *Building and Using Parsed Corpora.* Dordrecht: Kluwer.

Liu, H. and Hudson, R. (2006). Measuring dependency distance based on a Chinese treebank.Anon.

Luger, G. and Stubblefield, W. (1993). *Artificial Intelligence. Structures and strategies for complex problem solving*. New York: Benjamin Cummings.

Macdonald, M. C., Pearlmutter, N. J., and Seidenberg, M. S. (1994). Lexical nature of syntactic ambiguity resolution. Psychological Review 101: 676-703.

Macwhinney, B. (1989). 'Competition and teachability.', in Rice, M. & Schiefelbusch, R.(eds.), *The Teachability of Language*. Baltimore: Brookes. 63-104.

Malouf, R. (1998). *Mixed Categories in the Hierarchical Lexicon*. PhD. Stanford University.

Malouf, R. (2000). *Mixed categories in the hierachical lexicon.* Stanford: CSLI Publications.

Marslen-Wilson, W. (1984). 'Function and structure in spoken word recognition.', in Bouma, H. & Bouwhuis, D.(eds.), *Attention and Performance. Vol X: Control of Language Processes.* Hillsdale, NJ: Lawrence Erlbaum.

McClelland, J. and Rumelhart, D. (1988). *Explorations in parallel distributed processing: a handbook of models, programs, and exercises.* Cambridge, MA: MIT Press.

McRae, K., Spivey-Knowlton, M., and Tanenhaus, M. (1998). Modeling the influence of thematic fit (and other constraints) in on-line sentence comprehension. Journal of Memory and Language 38: 283-312.

Meara, P. (2002). 'Modelling attrition in vocabularies', in Hauksdóttir, A., Arnbjörnsdóttir, B., Gardharsdóttir, M., & þorvaldsdóttir, S.(eds.), *Forskning i Nordiske Sprog Som Andet- Og Fremmedsprog.* Reykjavík: Háskóli Islands. 153-175.

Meidner, O. M. (1994). 'Emotive meaning', in Asher, R.(ed.), *Encyclopedia of Language and Linguistics*. Oxford: Pergamon. 1111-1111.

Mel'cuk, I. (1997). *Vers une Linguistique Sens-Texte.* Paris: Collège de France: Chaire Internationale.

Michaelis, L. and Lambrecht, K. (1996). Toward a construction-based theory of language function: The case of nominal extraposition. Language 72: 215-247.

Miller, G. (1956). The magical number seven plus or minus two: some limits on our capacity for processing information. Psychological Review 63: 81-97.

Milroy, L. (1980). *Language and social networks*.

Miyake, A. and Shah, P. (1999). 'Toward unified theories of working memory. Emerging general consensus, unresolved theoretical issues and future research directions.', in Miyake, A. & Shah, P.(eds.), *Models of Working Memory. Mechanisms of Active Maintenance and Executive Control.* Cambridge: Cambridge University Press. 442-481.

Mollá, D., Schneider, G., Schwitter, R., and Hess, M. (2000). Answer extraction using a dependency grammar in Extrans. Traitement Automatique Des Langues 41: 145-178.

Ninio, A. (1994). Predicting the order of acquisition of three-word constructions by the complexity of their dependency structure. First Language 14: 119-152.

Ninio, A. (1996). 'A proposal for the adoption of dependency grammar as the framework for the study of language acquisition.', in Ben Shakhar, G. & Lieblich, A.(eds.), *Volume in Honor of Shlomo Kugelmass*. Jerusalem: Magnes. 85-103.

Ninio, A. (1998). 'Acquiring a dependency grammar: The first three stages in the acquisition of multiword combinations in Hebrew-speaking children. In G. Makiello-Jarza, J. Kaiser & M. Smolczynska (Eds.),', in Makiello-Jarza, G., Kaiser, J., & Smolczynska, M.(eds.), *Language Acquisition and Developmental Psychology. Cracow: Universitas.* Cracow: Universitas.

Nivre, J. (2004). Incrementality in Deterministic Dependency Parsing.Anon.

Nivre, J., Hall, J., and Nilsson, J. (2004). 'Memory-Based Dependency Parsing. In Ng, H. T. and Riloff, E. (eds.)', in Ng, H. T. & Riloff, E.(eds.), *Proceedings of the Eighth Conference on Computational Natural Language Learning (CoNLL), May 6-7, 2004*. Boston, MA: 49-56.

Owens, J. (1988). *The Foundations of Grammar: an Introduction to Mediaeval Arabic Grammatical Theory.* Amsterdam: Benjamins.

Pake, J. (1998). *The Marker Hypothesis. A constructivist theory of language acquisition.* PhD. Edinburgh.

Percival, K. (1990). Reflections on the History of Dependency Notions in Linguistics. Historiographia Linguistica. 17: 29-47.

Pickering, M. and Barry, G. (1991). Sentence processing without empty categories. Language and Cognitive Processes 6: 259

Pinker, S. (1994). *The Language Instinct*. London: Penguin.

Pinker, S. (1998). Words and rules. Lingua 106: 219-242.

Pollard, C. and Sag, I. (1994). *Head-Driven Phrase Structure Grammar*. Chicago: Chicago University Press.

Postal, P. (1966). On so-called 'pronouns' in English. Monographs on Languages and Linguistics 19: 177-206.

Poutsma, H. (1923). *The infinitive, the gerund and the participles of the English verb*. Groningen: P. Noordhoff.

Pullum, G. (1991). English nominal gerund phrases as noun phrases with verb-phrase heads. Linguistics 29: 763-799.

Quillian, M. R. (1968). 'Semantic memory.', in Minsky, M.(ed.), *Semantic Information Processing.* Cambridge, MA: MIT Press.

Quillian, M. R. and Collins, A. M. (1969). Retrieval time from semantic memory. Journal of Verbal Learning and Verbal Behavior 8: 240-247.

Quirk, R., Greenbaum, S., Leech, G., and Svartvik, J. (1985). *A Comprehensive Grammar of the English Language.* London: Longman.

Rambow, O. and Joshi, A. (1994). 'A Formal Look at Dependency Grammars and Phrase-StructureGrammars, with Special Consideration of Word-Order

Phenomena.', in Waner, L.(ed.), *Current Issues in Meaning-Text Theory.* London: Pinter.

Reisberg, D. (1997). *Cognition. Exploring the Science of the Mind.* New York: Norton.

Richards, N. (2004). Against bans on lowering. Linguistic Inquiry 35: 453-464.

Robins, R. H. (2001). 'In Defence of WP' (Reprinted from TPHS, 1959). Transactions of the Philological Society 99: 114-144.

Robinson, J. (1970). Dependency structure and transformational rules. Language 46: 259-285.

Robinson, P. (1986). Constituency or Dependency in the Units of Language Acquisition? An Approach to Describing the Learner's Analysis of Formulae. Linguisticae Investigationes 10: 417-437.

Roelofs, A. (1997). The WEAVER model of word-form encoding in speech production. Cognition 64: 249-284.

Roland, D. (2001). *Verb sense and verb subcategorization probabilities*. PhD. University of Colorado.

Rosch, E. (1976). 'Classification of real-world objects: origins and representations in cognition. (Reprinted in P. Johnson-Laird and P. C. Wason (eds.) (1977) Thinking: Readings in Cognitive Science. Cambridge: Cambridge University Press. 212-222.)', in Ehrlich, S. & Tulving, E.(eds.), *La Mémoire Sémantique.* Paris: Bulletin de Psychologie.

Rosta, A. (1997). *English Syntax and Word Grammar Theory*. PhD. UCL, London.

Rosta, A. (2005). 'Structural and distributional heads', in Sugayama, K. & Hudson, R.(eds.), *Word Grammar: New Perspectives on a Theory of Language Structure*. London: Continuum. 171-203.

Rosta, A. (2006). 'Structural and distributional heads', in Sugayama, K. & Hudson, R.(eds.), *Word Grammar: New Perspectives on a Theory of Language Structure*. London: Continuum.

Rumelhart, D. and McClelland, J. (1986). 'On learning the past tenses of English verbs.', in McClelland, J. & Rumelhart, D.(eds.), *Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Vol 2: Psychological and Biological Models.* Cambridge, MA: MIT Press.

Rushton, J. N. (2004). 'Natural Language Parsing Using Simple Neural Networks.', in Anon., *Proceedings of the 2003 International Conference on Artificial Intelligence.*

Rusteberg, F. G. A. (1874). *Historical development of the gerund in the English language* . Göttingen: Druck der Dieterichischen univ-buchdruckerei.

Sadock, J. (1991). *Autolexical Syntax: A theory of parallel grammatical representations.* Chicago: University of Chicago Press.

Sag, I. (1997). English relative clause constructions. Journal of Linguistics 33: 431-483.

Sapir, E. (1921). *Language*. New York: Harcourt, Brace and World.

Saussure, F. d. (1959). *Course in General Linguistics (translated by W. Baskin; French edition 1916)*. Lausanne: Payot.

Seyfarth, R., Cheney, D., and Bergman, T. (2005). Primate social cognition and the origins of language. Trends in Cognitive Sciences 9: 264-266.

Shieber, S. (1986). *An Introduction to Unification-based Approaches to Grammar*. Stanford: CSLI Publications.

Siewierska, A. (1991). *Functional Grammar*. London: Routledge.

Smith, N. V. (1999). *Chomsky. Ideas and ideals*. Cambridge: Cambridge University Press.

Solé, R. (2005). Syntax for free? Nature

Somers, H. (1984). On the validity of the complement-adjunct distinction in valency grammar. Linguistics 22: 507-531.

Sperber, D. and Wilson, D. (1995). *Relevance. Communication and cognition.* Oxford: Blackwell.

Sproat, R. (1988). 'Bracketing paradoxes, cliticization and other topics: the mapping between syntactic and phonological structure.', in Everaert, M., Evers, A., Huybregts, R., & Trommelen, M.(eds.), *Morphology and Modularity: in Honour of Henk Schultink*. Dordrecht: Foris. 339-360.

Steedman, M. (2000). *The Syntactic Process*. London: MIT Press.

Stump, G. (1993). On rules of referral. Language 69: 449-479.

Sturt, P., Pickering, M., Scheepers, C., and Crocker, M. (2001). The preservation of structure in language comprehension: Is reanalysis the last resort? Journal of Memory and Language 45: 283-301.

Sutcliffe, R., Koch, H.-D., and McElligott, A. e. (1996). *Industrial Parsing of Software Manuals*. Amsterdam: Rodopi.

Swan, M. (1995). *Practical English Usage.* Oxford: Oxford University Press.

Tajima, M. (1985). *The Syntactic Development of the Gerund in Middle English.* Tokyo: Nanun-do.

Talmy, L. (1988). Force dynamics in language and cognition. Cognitive Science 12: 49-100.

Tesnière, L. (1959). *Éléments de syntaxe structurale*. Paris: Klincksieck.

The Meaning of the Sentence in its Semantic and Pragmatic Aspects (1986). *Sgall, Petr; Hajicova,Eva;Panevova,J.* Prague: Academia.

Tomasello, M. (1998). Constructions: A construction grammar approach to argument structure. Journal of Child Language 25: 431-442.

Tomasello, M. (1999). *The Cultural Origins of Human Cognition*. London: Harvard University Press.

Tomasello, M. (2000). The item-based nature of children's early syntactic development. Trends in Cognitive Sciences 4: 156-163.

Tomasello, M. (2003). *Constructing a language: a usage-based theory of language acquisition.* Harvard University Press.

Touretzky, D. (1986). *The Mathematics of Inheritance Systems*. Los Altos, CA: Morgan Kaufmann.

Tzanidaki, D. (1996). *The Syntax and Pragmatics of Subject and Object Position in Modern Greek.* PhD. UCL.

Tzanidaki, D. (1998). 'Clause structure and word order in Modern Greek', in Joseph, B., Horrocks, G., & Philippaki-Warburton, I.(eds.), *Themes in Greek Linguistics 2.* Amsterdam: Benjamins. 229-254.

Van Langendonck, W. (1987). Word Grammar and child grammar. Belgian Journal of Linguistics 2: 109-132.

Van Valin, R. (1993). *Advances in Role and Reference Grammar*. Amsterdam: Benjamins.

Vosse, T. and Kempen, G. (2000). Syntactic structure assembly in human parsing: a computational model based on competitive inhibition and a lexicalist grammar. Cognition 75, 105-143.

Warneken, F. and Tomasello, M. (2006). Altruistic Helping in Human Infants and Young Chimpanzees. Science 311: 1301-1303.

Wells, R. (1947). Immediate constituents. Language 23: 81-117.

Wescoat, M. (1994). Phrase structure, lexical sharing, partial ordering and the English gerund. Proceedings of the Berkeley Linguistics Society 20: 587-598.

Wierzbicka, A. (1996). *Semantics: Primes and universals*. Oxford: Oxford University Press.

Wik, B. (1973). *English nominalizations in -ing.Synchronic and diachronic aspects*. Uppsala: Stockholm: Almqvist & Wiksell.

Williams, E. (1984). Grammatical relations. Linguistic Inquiry 15: 639-674.

Williams, S., Savage-Rumbaugh, S., and Rumbaugh, D. M. (1994). 'Apes and language', in Asher, R.(ed.), *Encyclopedia of Language and Linguistics*. Oxford: Pergamon. 139-146.

Winograd, T. (1976). Towards a procedural understanding of semantics. Revue Internationale De Philosophie 30: 260-303.

Wurff, W. v. d. (1993). 'Gerunds and their objects in the Modern English period.', in Marle, J. v.(ed.), *Historical Linguistics 1991*. Amsterdam: Benjamins. 363-375.

Wurff, W. v. d. (1997). Gerunds in the Modern English period: structure and change. The History of English 3: 163-196.

Yoon, J. (1996). Nominal gerund phrases in English as phrasal zero derivations. Linguistics 34: 329-356.

Zwicky, A. (1977). *On clitics*. Bloomington, Indiana: Indiana University Linguistics Club.

Zwicky, A. (1992a). 'Clitics: an overview.', in Bright, W.(ed.), *International Encyclopedia of Linguistics*. Oxford: Oxford University Press. 269-270.

Zwicky, A. (1992b). 'Morphology: Morphology and syntax', in Bright, W.(ed.), *International Encyclopedia of Linguistics*. Oxford: Oxford University Press. 10-12.