Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

Research
Report
RR-92-20

Representing Grammar, Meaning and Knowledge

John Nerbonne

March 1992

Deutsches Forschungszentrum für Künstliche Intelligenz
GmbH

Postfach 20 80
D-6750 Kaiserslautern, FRG
Tel.: (+49 631) 205-3211/13
Fax: (+49 631) 205-3210

Stuhlsatzenhausweg 3
D-6600 Saarbrücken 11, FRG
Tel.: (+49 681) 302-5252
Fax: (+49 681) 302-5341

Deutsches Forschungszentrum für Künstliche Intelligenz

The German Research Center for Artificial Intelligence (Deutsches Forschungszentrum für Künstliche Intelligenz, DFKI) with sites in Kaiserslautern und Saarbrücken is a non-profit organization which was founded in 1988. The shareholder companies are Daimler Benz, Fraunhofer Gesellschaft, GMD, IBM, Insiders, Krupp-Atlas, Mannesmann-Kienzle, Philips, Sema Group Systems, Siemens and Siemens-Nixdorf. Research projects conducted at the DFKI are funded by the German Ministry for Research and Technology, by the shareholder companies, or by other industrial contracts.

The DFKI conducts application-oriented basic research in the field of artificial intelligence and other related subfields of computer science. The overall goal is to construct *systems with technical knowledge and common sense* which - by using AI methods - implement a problem solution for a selected application area. Currently, there are the following research areas at the DFKI:

- ❑ Intelligent Engineering Systems
- ❑ Intelligent User Interfaces
- ❑ Intelligent Communication Networks
- ❑ Intelligent Cooperative Systems.

The DFKI strives at making its research results available to the scientific community. There exist many contacts to domestic and foreign research institutions, both in academy and industry. The DFKI hosts technology transfer workshops for shareholders and other interested groups in order to inform about the current state of research.

From its beginning, the DFKI has provided an attractive working environment for AI researchers from Germany and from all over the world. The goal is to have a staff of about 100 researchers at the end of the building-up phase.

Prof. Dr. Gerhard Barth
Director

Representing Grammar, Meaning and Knowledge

John Nerbonne

DFKI-RR-92-20

This report also appears in the proceedings of **Workshop on Natural Language Processing and Knowledge Representation**, edited by Susanne Preuß and Birte Schmitz. The workshop was held on October 9-11, 1991 at the Technische Universität, Berlin.

This work has been supported by a grant from The Federal Ministry for Research and Technology (FKZ ITW-90020).

Representing Grammar, Meaning and Knowledge

John Nerbonne [*]
Deutsches Forschungszentrum für Künstliche Intelligenz
Stuhlsatzenhausweg 3
D-6600 Saarbrücken 11, Germany
nerbonne@dfki.uni-sb.de

Abstract

Among the expertises relevant for successful natural language understanding are grammar, meaning and background knowledge, all of which must be represented in order to decode messages from text (or speech). The present paper is a sketch of one cooperation of grammar and meaning representations—with some remarks about knowledge representation—which allows that the representations involved be heterogeneous even while cooperating closely. The modules cooperate in what might be called a PLURALIST fashion, with few assumptions about the representations involved. In point of fact, the proposal is compatible with state-of-the-art representations from all three areas.

The paper proceeeds from the nearly universal assumption that the grammar formalism is feature-based and insufficiently expressive for use in meaning representation. It then demonstrates how feature formalisms may be employed as a semantic metalanguage in order that semantic constraints may be expressed in a single formalism with grammatical constraints. This allows a tight coupling of syntax and semantics, the incorporation of nonsyntactic constraints (e.g., from knowledge representation) and the opportunity to underspecify meanings in novel ways—including, e.g., ways which distinguish ambiguity and underspecification (vagueness).

We retain scepticism vis-à-vis more ASSIMILATIONIST proposals for the interaction of these—i.e., proposals which foresee common formalisms for grammar, meaning and knowledge representation. While such proposals rightfully claim to allow for closer integration, they fail to account for the motivations which distinguish formalisms—elaborate expressive strength in the case of semantic representations, monotonic (and preferably decidable) computation in the case of grammar formalisms, and the characterization of taxonomic reasoning in the case of knowledge representation.

Keywords: disambiguation, natural language processing, knowledge representation, feature formalism

[*] Thanks to Rolf Backofen for discussion of the ideas presented here.

Representing Grammar, Meaning and Knowledge

Contents

1 Introduction

Subsystems and interfaces in natural language processing (NLP) systems are important but not well understood. While this paper cannot claim to shed any definitive light on the general problem, it will illustrate different strategies of integration using as example the semantics module and its interfaces, especially the interfaces to grammar and knowledge representation. We ultimately find that a PLURALIST strategy—in which modules with no shared formalism cooperate—is practically more auspicious and theoretically defensible. An important diatribe in this paper is that one should study the semantics module if one wishes to understand the genuine problems of modularity in NLP. This follows from the radically different, and expressively richer formalism which semantics forces us to employ. A second key thesis concerns the nature of interfacing expressively stronger with expressively weaker components: here we argue that, if information from the stronger component must be expressed in the weaker, then the weaker should be used as a metalinguistic specification of the stronger, and we illustrate this using the grammar/semantics interface. The only alternative appears to be a special-purpose translation procedure.

1.1 General Considerations on NLP Modularity

There are two very general views about linking NLP modules which are to cooperate closely. The "pluralist" view generally sees NLP as a field with a history of results and a stock of variously successful techniques and systems for solutions of selected problems. These are potential modules. One might point in 1992 to finite-state morphology, inheritance lexicons, feature grammars, logic-based semantics, and taxonomy-based knowledge representations as examples of these techniques, and to KIMMO, DATR, PATR-II, QLF, and KL-ONE as respective examples of systems which are more or less available for use in attacking problems of interest.[1] Since no single effort is likely to redo all of this successfully, part of the task in NLP system building must be the linking of heterogeneous systems. In this view NLP system building is not radically different from other sorts.

Opposed to pluralism is an ASSIMILATIONIST view, which argues from peculiarities of NLP to the need for closely coupled modules. This view finds justification in psycholinguistic

[1] Cf. Karttunen 1983, Evans and Gazdar 1990, Shieber et al. 1983, Alshawi and others July 1989, and Brachman and Schmolze 1985 for descriptions of these systems.

studies which show, e.g., that semantic interpretation influences (human) parsing and even phonetic decoding.[2] But parallel to cognitive motivations there are purely technical reasons as well for preferring to couple modules closely, i.e., allowing "cross talk" between modules. Importantly this allows one to resolve ambiguities flexibly: e.g., in attempting to resolve parsing ambiguities using information from knowledge representation, it can be advantageous to deploy knowledge representation sources early, so that parse hypotheses incompatible with these need not be pursued. Clearly, a loose level of integration can be attained even using radically heterogeneous modules, but the cost involved is special-purpose code which is capable of translating selected partial results of one module and shipping them to another. Preferable would be a tight coupling, i.e., one which allow arbitrarily close cooperation without special interpretation or control. It is difficult to see how this could be achieved without requiring a homogeneous formalism.

Each view of NLP modularity runs a risk: pluralism supports cross talk poorly and runs the risk of multiplying formalisms unnecessarily. To be sure, the latter is a risk one can "insure" against, especially through careful metatheoretical work, but it remains. Assimilationism, on the other hand, requires the development of an all-encompassing formalism, the reimplementation of old results in a new formalism, and finally, in seeking to overcome modular boundaries, it runs the (albeit again insurable) risk of abandoning modularity entirely.

The above are practical, engineering considerations. The scientific merits of expressing theories of language understanding incorporating various components in a unified descriptive language may ultimately devolve to questions of the maturity of the subfields involved. It is certainly not common practice in linguistics today.

1.2 Grammar, Meaning and Knowledge

The very general remarks above can be illustrated concretely in the case of grammar, meaning and knowledge representations. We proceed from standard (and state-of-the-art) assumptions in each case: that grammar is represented in a feature description language (like PATR-II or one of its descendants), meaning in a logic including a representation for generalized quantifiers (like QLF), and knowledge in a taxonomic logic (like KL-ONE). These formalisms are useful subjects because they are well-studied and the tasks they perform are well-understood. See the references above (note 1) for introductions and explanations of these.

Standard feature structure description languages may be regarded as variants of the quantifier-free predicate calculus with identity (cf. Johnson 1988 and Smolka 1988). Taxonomic logics of the KL-ONE variety add restricted types of quantification that keep them within the limits of decidability (by guaranteeing that definitions are not applied recursively—cf. Baader and Hollunder 1990). Meaning representation lanaguages are designed for the representation of natural language meanings (albeit sometimes within a restricted domain); as such, they invariably go well beyond not only decidable logics, but even first-order. Cf. Scha 1976, Schubert and Pelletier 1982, Zeevat et al. 1987 and Alshawi and others July 1989 for a representative sample of meaning representation proposals for computational semantics, all of which involve non-first-order expressive capacity.

There have been a number of works which try to specify the relationships between these and which suggest interfaces of various sorts. Nebel and Smolka 1990 demonstrate that feature structure description languages and taxonomic knowledge representations are extremely close conceptually. Backofen et al. 1991 provide a concrete suggestion for an interface. Bobrow 1979 provides for semantic interpretation directly in KL-ONE, and

[2] Cf. Clark and Clark 1977, 45-79 for a summary of early studies.

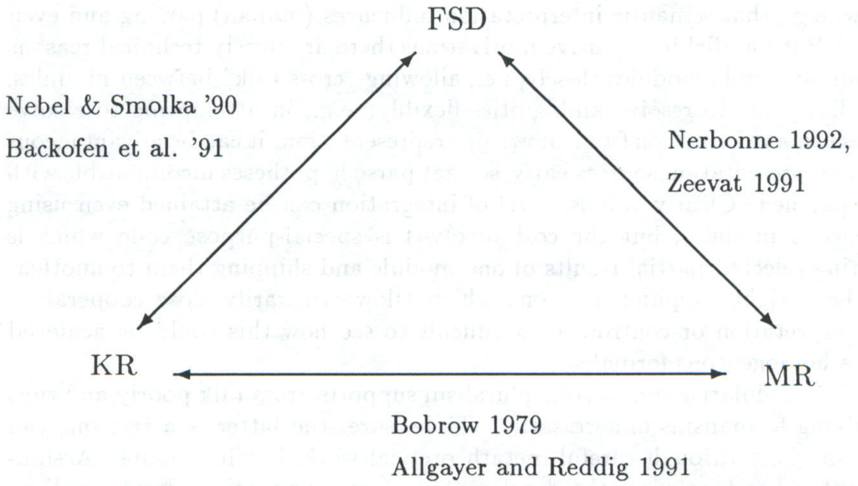


Figure 1: A triad of NLP formalisms—knowledge representations, feature structure descriptions, and meaning representations—and some works which attempt to relate them. The close affinities of feature logics and taxonomic logics was explored by Nebel and Smolka 1991, which led to a concrete proposal by Backofen et al. 1991 for an interface. Bobrow 1979 proposed using a taxonomic logic as a semantic representation language, and Allgayer and Reddig 1990 support that with proposals for radically extending the taxonomic logic to include expressive devices familiar in meaning representation. Zeevat 1991 proposes deriving semantics using higher-order feature description, and Nerbonne 1992 proposes a view in which standard feature description languages function as metalinguistic specifications on meaning representations.

Allgayer and Reddig 1990 suggest KL-ONE enhancements which make this more plausible. Zeevat 1991 considers higher-order feature structure description languages in order to represent meanings, and Nerbonne 1992b proposes using standard FSD's to express semantic constraints metalinguistically.

Given the relative expressive strengths of the logics involved, it should be clear that the most difficult tasks are those involving interfacing the expressively very rich meaning representation to the other (weaker) formalisms, and this indeed is the first conclusion I wish to argue for. Since meanings cannot be expressed in first-order logic, both PATR-II derivatives (normally involving a subset of first-order) and KL-ONE (always a subset of first order) are poor candidates for meaning representation.

Schemes which claim to provide homogeneous formalisms for NLP should be tried on this count: can they genuinely provide for meaning representations of a significantly rich sort? Those who argue for representing semantics (directly) in feature description languages or taxonomic logics should be required to demonstrate their treatment of higher-order information, such as generalized quantifiers.

We turn now to a proposal for a fairly tight interface between syntax and semantics (feature structure description languages and meaning representation languages) which exploits all of the expressive richness of the latter even while remaining within the bounds posed by the former. The key is a level of indirection.

2 Feature Structure Descriptions and Meaning Representations

Syntax/Semantics interfaces using unification-based or feature-based formalisms may be found in Shieber 1986, Pollard and Sag 1987, Fenstad et al. 1987, and Moore 1989. The

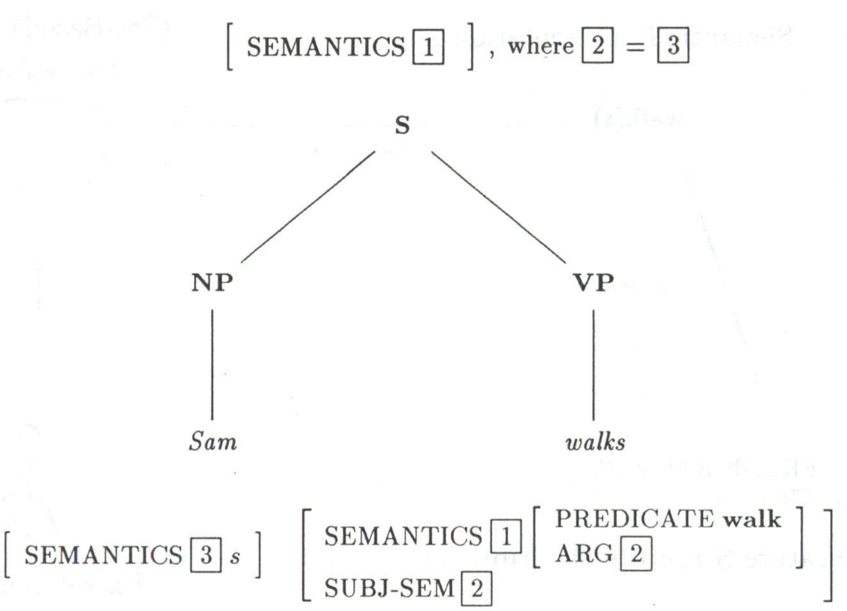

[SEMANTICS 1], where 2 = 3
S
NP
VP
Sam
walks
[SEMANTICS 3 s]
SEMANTICS 1 [PREDICATE walk ARG 2]
SUBJ-SEM 2

Figure 2: A sketch of the semantic derivation of *Sam walks*, **walk** (s) as this would proceed using unification. Unification applies to syntactic and semantic representations alike, eliminating the need to compute these in distinct processes, and unification is employed to bind variables to argument positions, eliminating the need for (a great deal of) β-reduction as used in schemes derived from Montague and the lambda calculus.

primary initial reason for investigating feature-based syntax/semantics interfaces was that they harmonize so well with the way in which syntax is normally described; this close harmony means that syntactic and semantic processing (and indeed other processing, see below) can be as tightly coupled as one wishes—indeed, there needn't be any fundamental distinction between them at all. In feature-based formalisms, the structure shared among syntactic and semantic values constitutes the interface in the only sense in which this exists.

2.1 A Simple Illustration

The fundamental idea is simply to use feature structures to represent semantics.[3].

If one wishes to compute the semantics of a sentence such as *Sam walks*, one first defines a feature SEMANTICS which must be lexically provided for in the case of *Sam* and *walks*, and which can be computed from these (together with other information) in the case of the sentence. Figure 2 provides an illustration of how this works.

In Nerbonne 1992b and Nerbonne 1992a several advantages of the unification-based view of the syntax/semantics interface over standard views are pointed out. The unification-based view sees the interface as characterized by a set of constraints to which nonsyntactic information may contribute, including phonological and pragmatic information. Let the semantics of intonation and and that of deixis serve as examples where nonsyntactic information of these two sorts crucially constrains semantic interpretation. See Nerbonne 1992a for more detailed presentation of these examples, and see Fenstad et al. 1987, pp.12-17 for a general discussion of the distinction between the constraint-based and the homomorphic views of the relation between syntax and semantics.

The nature of feature-based semantics is clarified in Nerbonne 1992b: semantic features

[3] Sections 2 and 3.1 summarize material presently in Nerbonne 1992a and Nerbonne 1992b

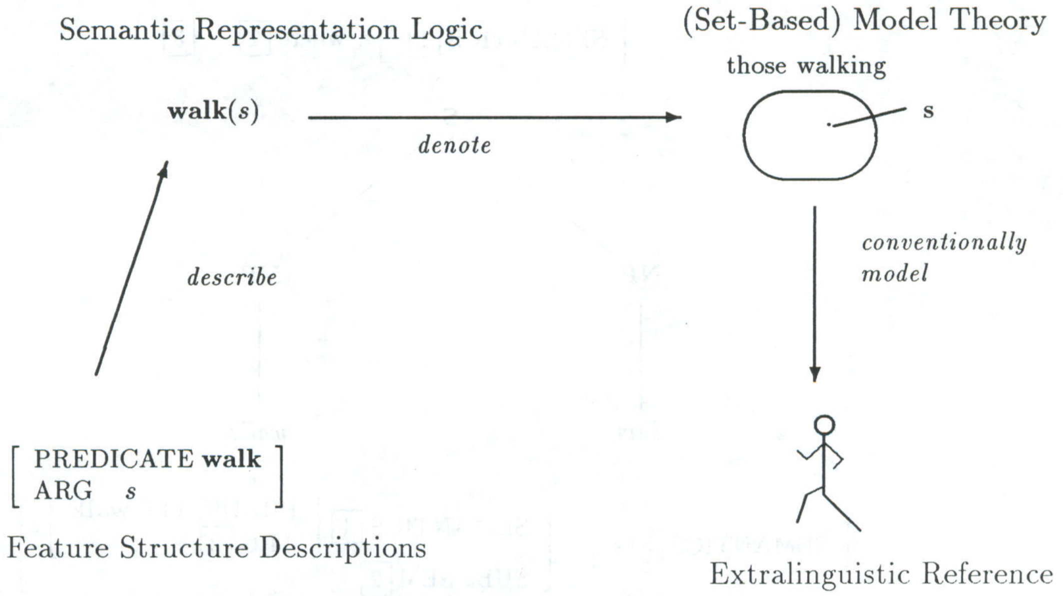


Figure 3: The relation of feature-based semantics to semantic representation logic, model theory, and the extralinguistic objects of discourse. A simpler, but ultimately unsatisfactory view, would eliminate the semantic representation logic.

(directly) describe logical forms, not model structures. Of course, the logical forms denote in turn elements within the model structure. Figure 3 provides a graphic rendition of this view, and the Appendix of this paper is a brief illustration of the technique.

There are further advantages in the view of semantics processing as manipulation of logical form descriptions. These include (a) a characterization of semantic ambiguity (as opposed to underspecification), which in turn provides a framework in which to describe disambiguation, and (b) the opportunity to underspecify meanings in a way difficult to reconcile with other views. The former point is developed below and the latter in Nerbonne 1992b, which demonstrates more concretely how a semantics along these lines may function. A brief sketch illustrating the integration of disambiguation is provided in the next section, and some concrete details of the relation between feature system and semantic representation language are elaborated in the Appendix.

The feature-based view furthermore allows syntactic and semantic information to be bundled in complex, but useful ways. These allow the very simple statements of syntax/semantics relationships that make HPSG attractive; cf., e.g., Pollard and Sag 1987, whose SUBCATEGORIZATION PRINCIPLE (p.71) actually accounts for a good deal of semantics processing, since it effects the unification of complement semantics with an argument position within head semantics. (The computation shown in Figure 2 is a consequence of the Subcategorization Principle.) Cf. Nerbonne 1992a for further discussion.

3 Two-Level Representations, Disambiguation and Knowledge Representation

One consequence of the metalinguistic view of semantic specification is that there are two levels of semantic representation—an object language in which meanings are expressed and a metalanguage in which constraints on meanings are expressed. The first section below argues that this is a desirable bifurcation of semantic responsibility, while the second demonstrates its applicability to disambiguation and points out futher issues involved in disambiguation.

3.1 Object Language and Metalanguage

A puzzle is presented in Nerbonne 1992a which the metalinguistic view of semantics specification solves neatly, but which is quite puzzling on any view which eschews two levels of semantic representation. It concerns the distinction between vagueness and ambiguity—i.e., the difference between the noun *bank* (ambiguous between the '$-home' and 'riverside' meanings) on the one hand, and *glove* on the other. We examine the word *glove*, since, like *bank*, it can refer only to quite distinct categories of objects—left-hand gloves or right-hand gloves. This distinction is crucial in quantification and anaphora. Thus *three gloves* ignores the distinction between left- and right-hand gloves in a way that *three banks* may not (this cannot refer to a pair of financial institutions and a riverside); similarly *Sam bought a bank and Bob sold one* cannot describe a situation involving a financial institution on the one hand and a riverside on the other; i.e., it allows only a pair of the four possible combinations of the two meanings. The vague (or underspecified *glove*) is less restrictive; any of the four possible combinations of meanings can be at play in the following example:[4]

Sam lost a glove and Bob found one

If we allow that semantics makes metalinguistic specifications, then it specifies that *bank* has the meaning [PRED {$-HOME,RIVERSIDE}], using metalinguistic disjunction. Under the (quite reasonable) assumption that there is no single object-language predicate denoted by this disjunction, we can explain the quantificational and anaphoric facts. *Glove*, on the other hand, is unambiguous, even if its semantics may be equivalent to an object-language disjunction: [PRED **glove**], where the object-language disjunction holds:

$$\mathbf{glove}(x) \leftrightarrow \mathbf{left\text{-}glove}(x) \vee \mathbf{right\text{-}glove}(x)$$

The standard solution here is to postulate distinct lexical items for ambiguity, but this solution essentially denies purely semantic ambiguity, reducing it to lexical ambiguity, and it fails to generalize to the view of the lexicon as a disjunction of words, common in the feature-based theories (cf. HPSG, Pollard and Sag 1987 p.147). Under the view of the lexicon as a disjunction of words (or word descriptions), the postulation of lexical items distinct only in semantics reduces to the postulation of a single item with disjunctive semantics, since:

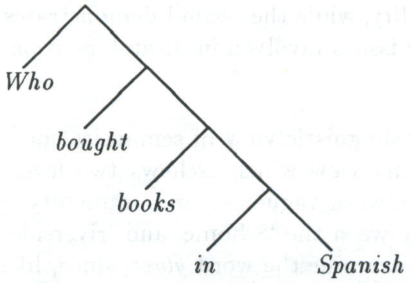
$$(p_1 \wedge \ldots \wedge p_n \wedge q) \vee (p_1 \wedge \ldots \wedge p_n \wedge r) \leftrightarrow (p_1 \wedge \ldots \wedge p_n \wedge (q \vee r))$$

The solution offered is consistent with the view that multiple lexical entries are involved, but it immediately suggests a more perspicuous representation (in analogy to the right-hand side of the biconditional); it differs only in insisting that there be two levels of semantics: a level which constrains semantic representations, and the level of semantic representations themselves. Lexical ambiguity involves underspecification at the first level; lexical vagueness is underspecification at the second.[5]

Thus, given a level of semantic representation together with a metalinguistic level at which constraints on semantic representation are expressed, a notion of semantic ambiguity as opposed to semantic underspecification may be characterized. This is further justification

[4] Cf. Zwicky and Sadock 1975 for a discussion of anaphora as a test of ambiguity.

[5] While there is not space here to anticipate all the reactions to this argument, I would like to note that the quantification facts show that this is not an issue of semantic grain, in the sense in which this is debated, e.g., in situation semantics and possible worlds semantics. Cf. Barwise and Perry 1983, Barwise 1989. Thus it will not do simply to point to logics in which there may be a relation of material equivalence, but no relation of logical equivalence between the left and right sides above. This is so because natural language quantification is insensitive to distinctions finer than material equivalence.



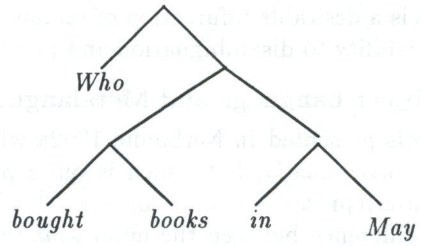


Figure 4: Disambiguation may even require the recognition of distinct constituent structures. Note that the proper names *Spanish* and *May* are not syntactically distinct, nor do that belong to distinct logical types—each denotes an entity. But they do denote objects of different SORTS, as this term is used in sortal logic, since May is a time and Spanish a language. The link to knowledge representation, especially of the kind encoded in KL-ONE, is justified by the emphasis on sorts—taxonomies, or conceptual hierarchies—which distinguishes knowledge representation schemes such as KL-ONE.

for the view that semantic processing involves the manipulation of semantic representation—which in turn further justifies the postulation of this level of representation in addition to the level at which meanings are directly modeled. The more general claim on which this argument hinges is that the characterization of ambiguity in a representation L_1 is always with respect to a second representational system L_2. The scheme proposed here distinguishes two levels of semantics and is thus capable of characterizing ambiguity; systems with a single level are not.

3.2 Disambiguation

DISAMBIGUATION is the process of determining (i) which of potentially many meanings was intended in an utterance, but also (ii), with respect to a particular application, which facet is relevant to an NL interaction. The former is a response to the ambiguity of natural language, while the latter exists even where no ambiguity does. We illustrate these in turn below. Disambiguation occupies computational linguists more than theoretical linguists, and is extremely important in applications in which there many be uncertainty about input—e.g., speech. Like parsing itself, at least some disambiguation seems to be automatic, so that untrained speakers are not aware of needing to disambiguate structures. The example below, graphed in Figure 4, suggests how unobtrusive the process is:

(1) a. Who bought books in Spanish?
b. Who bought books in May?

This sort of example is convenient because it shows how pervasive the effects of disambiguation may be—reaching even into the parsing component. It is simultaneous misleading if it suggests that genuine disambiguation tasks need to be accompanied by such striking consequences. For even if disambiguation MAY be accompanied by striking consequences in application independent ways, the need for disambiguation arises in NLP interface efforts in ways that need have no purely linguistic ramification whatsoever. In particular, NL interfaces need to be sensitive to application distinctions which do not correspond to natural language ambiguities.

Consider the DISCO application, that of consulting with multiple agents who plan shipping. Here the phrase *Schmidts Ladung* 'Schmidt's freight' certainly denotes freight which stands in some relation to Schmidt. For example, we may imagine the freight contracts in

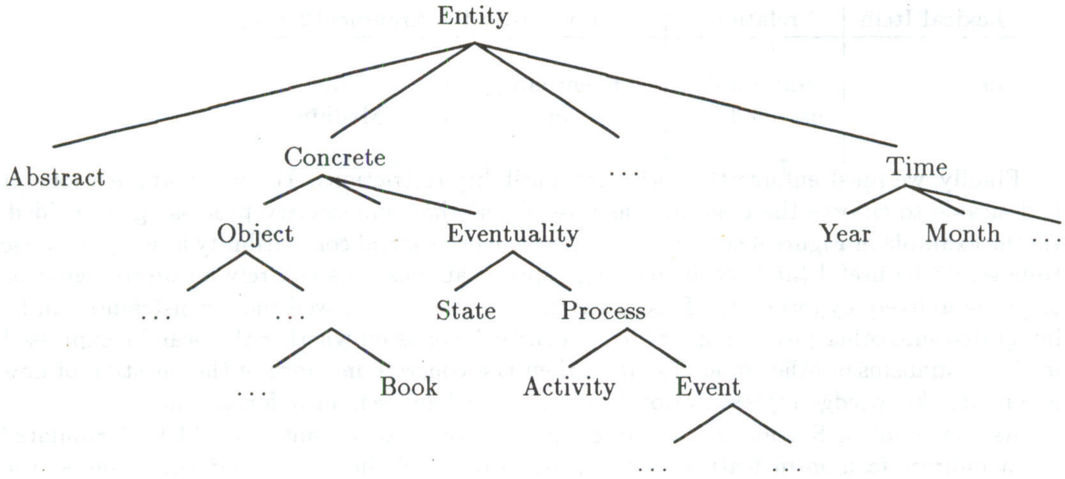


Figure 5: A sort hierarchy which distinguishes enough classes in the domain to illustrate sortal disambiguation for the sentences in (1) (in the text). The hierarchy here is better portrayed not as a tree, but as a directed graph—in which sorts inherit on more than one path back to the root. For example, states and activities might best be viewed as subsorts of both eventualities and nondiscretes, which would include physical substances (water, flour) as well. The move to non-tree-structured hierarchies does not affect the points here however, about the interaction of grammar, semantics and knowledge representations.

the application as organized into a small database, where the freight contract is the basic tuple.

Order Nr.	Contractor	Agent	Destination	?
457	Schmidt	...	...	...
574	...	Schmidt	...	...
745	...	...	Schmidt	...
475	...	...	...	Schmidt

Thus the phrase *Schmidts Ladung* could designate freight which Schmidt contracted to have shipped, freight for which he is the freight agent, freight being sent to him, and perhaps even freight which stands in yet another relation to him (as owner, inspector, as packer, etc.). Now it is unlikely that the relation expressed by the German possessive construction (genitive + $\bar{\mathrm{N}}$) is ambiguous, and it is unthinkable that the construction is ambiguous to just this degree and in just this fashion.

Taxonomic reasoning may fruitfully be applied both to the resolution of linguistic ambiguity and to resolution of application-specific distinctions. We consider these in turn. At the heart of taxonomic reasoning is the imposition of a sort hierarchy on the domain, illustrated in Figure 5 for the case where we found linguistic ambiguity.

In addition to the provision of a sort hierarchy, sortal disambiguation requires a characterization of which sorts are appropriate for which (argument positions of) relations. We would then allow that *in* translate into (at least) two relations, one temporally relating eventualities to times, and the other relating documents to media (but not to times). Schematically:

Lexical Item	relation	Argument1-Sort	Argument2-Sort
in	temporal-loc	Eventuality	Time
	use-medium	Document	Medium

Finally, we must enforce the sortal compatibility restrictions. For many applications, it is desirable to enforce these as early as possible, so that unnecessary processing is avoided. As the example in Figure 4 suggests, an enforcement of sortal compatibility as early as parse time would be useful (and recall that, e.g., speech applications will rely on disambiguation to prune unlikely hypotheses). This raises the question of how well these constraints can be integrated into other processing—which of course depends on whether they can be expressed in the formalisms of other modules. Here then is a concrete instance of the question of how one relates knowledge representation to grammatical and semantic formalisms.

As the point of Section 2 was the demonstration that semantics could be formulated in an indirect fashion in feature formalisms, so we shall show here that the same is true of knowledge representation—at least within bounds. Cf. Moens et al. 1989 for an earlier proposal along the same lines. That is, once we've taken the step to representing the semantics of *in* in a typed feature description language:

$$\begin{bmatrix} \text{PRED} & \textbf{temporal-location} \\ \text{THEME} & \\ \text{LOCATION} & \end{bmatrix}$$

Then we can also represent the sortal information, relying on unification to enforce sortal compatibility, and thus integrating sortal disambiguation with the unification used in parsing. The following feature structure description represents the ambiguous lexical item *in*:

$$\begin{bmatrix} \text{FORM} & \text{in} \\ \text{SEMANTICS} & \left\{ \begin{array}{c} \begin{bmatrix} \text{PRED} & \textbf{temporal-location} \\ \text{THEME} & [\ \textit{Eventuality}\ ] \\ \text{LOCATION} & [\ \textit{Time}\ ] \end{bmatrix} \\ \vdots \\ \begin{bmatrix} \text{PRED} & \textbf{expressive-means} \\ \text{THEME} & [\ \textit{Document}\ ] \\ \text{LOCATION} & [\ \textit{Medium}\ ] \end{bmatrix} \end{array} \right\} \end{bmatrix}$$

The representation for the word *May*, whose semantics is of the sort Month, and therefore also of the sort Time, can successfully unify with the (location argument of) the first alternative semantics for *in*, but not the last, for which an argument of the sort Medium is expected. Thus the PP *in May* seeks to attach where its first argument will be of the sort Eventuality—and this can be a VP attachment, since VP's denote eventualities, but not an NP with the head noun *books*, since this denotes objects of an incompatible sort.

Although we shall not present the details of the treatment of the resolution of application-specific distinctions, it should be clear that the same techniques apply. In the example *Schmidts Ladung*, the relation between Schmidt and the freight is potentially disambiguated by information about whether Schmidt is a shipper, a customer, or the recipient of a customer's shipment. Nor shall we attempt on the basis of this example to argue that sortal restrictions must come from the domain and NOT from the lexicon—the dilemma seems spurious, since the lexicon must in some way be accommodated to the domain for serious applications anyway. Cf. Iida et al. 1989 on the relation between lexicon and disambiguation in complex applications.

3.3 Emerging Issues in Disambiguation

Thus the feature formalism allows the integration of constraints from knowledge representation as well. There are several qualifications needed, however. First, the feature formalism cannot faithfully represent all of the sorts of richer KL-ONE-like languages, in particular not those which allow quantified sort defintions, e.g., definitions such as:

$$\text{Parent}(x) \leftrightarrow \exists\, y.\text{Child-Of}(y, x)$$

Some KL-ONE derivatives allow these without relinquishing decidability, but they are not foreseen in feature formalisms. On the other hand I am unsure of how important this sort of example is—i.e., how frequently one must appeal to sorts of this complexity.

A second qualification is that this sort of treatment will not allow the enforcement of constraints which derive from inferences based on earlier utterances—in order to accomplish this a genuine integration into the semantics representation language would be required. We have in mind the kind of inference possible when information about an individual accumulates during the course of a conversation, but which may be demonstrated even in a single sentence:

(2) Sam talked for two hours in the library and read books for one.

The interesting phrase is the *for one*, and the interesting question is how we account for its VP attachment. Of course, the sortal explanation is available—we simply postulate that *for* denotes a relation between eventualities and durations, but that it denotes no relation between books and durations. But this information cannot be available on the basis of the lexical item one—it must be inferred on the basis of previous content (and the anaphoric link).

One can, as always, attempt alternative explanations, but the ones which immediately come to mind are unconvincing. One could postulate that the choice of attachment site depends on a parallelism to the first clause, but that is not necessary (a). Or one could hypothesize that the VP atachment is strongly preferred. But the $\bar{\text{N}}$-structures of the form $\bar{\text{N}}$+ PP-for are quite possible (b):

(3) a. He spent two years at ISI. His project for one was on graphics, and he worked independently for the other.
b. Sam looked for gifts for his kids. He saw books for one and T-shirts for the other.

Thus we conclude that a proper account of disambiguation should go beyond the encoding in feature structures illustrated above, and that a more thoroughgoing integration of semantic representation and disambiguating mechanisms is ultimately required. The presentation above stills shows how a great deal of disambiguating information can be integrated into the feature systems and thus arbitrarily deep into a modern NLP system, even if it turns out to be incomplete.

A third qualification about the usefulness of feature-based disambiguation concerns a fundamental pitfall of sortal disambiguation, i.e., that it needs to distinguish between asserted and presupposed sortal information. This is quite clear in the case of application-specific distinctions, and arguably necessary for linguistic ambiguities as well. We examine the case of application-specific distinctions first. We argued above that *Schmidt's freight* might be understood on the basis of a variety of application-specific relations, including 'freight-shipper', 'freight-contractor', etc. In deciding which of these is relevant, it is legitimate to examine the sort to which Schmidt belongs (shipper, contractor, etc.). But notice that in a phrase such as *Schmidt's freight* the relation between Schmidt and the freight is presupposed, not asserted. It would clearly be wrong to apply disambiguation techniques to cases where the relation is asserted or questioned, but not presupposed, e.g., in *Is this freight Schmidt's?* or *If Schmidt sends freight, then his freight will arrive today.* (In the latter case, one can imagine blithely disambiguating *his freight* to the 'freight-recipient' relation on the basis of Schmidt's being listed only as recipient—but this would clearly lead to errors.) The case of genuine linguistic ambiguity is similar, but arguably different, in that sortal mismatches remain peculiar enough even in assertion to warrant perhaps being categorized as ill-formed. This tack would regard the following as ill-formed:

The book is for one hour.
Is the book for one hour?
The book cannot be for one hour.
The book is in May.
Is the book in May?
It would be impossible for the book to be in May.

While the examples are undoubtedly peculiar, it still seems wrong to regard them as ill-formed as opposed to unusually formulated or simply concerned with unusual circumstances. Presented with a sentence such as one of these, it would seem that the appropriate reaction would be to try to interpret it metaphorically or, if possible, to clarify it with a user. This conclusion suggests that both sorts of disambiguation—that of resoving genuine ambiguities and that of resolving application-specific distinctions—benefit from the distinction between assertion and presupposition, which therefore ought to be part of a comprehensive disambiguation scheme.

A fourth and final general comment about disambiguation concerns the need for some kind of COERCION when determining appropriate sorts. Consider the following sequence of uses of the adjunct *8 am*:

(4) a. The train departs tomorrow at *8 am*
b. The *8 am* departure
c. The regular *8 am* departure
d. The *8 am* train
e. The *8 am* passengers
f. The *8 am* crew
g. The *8 am* stops

While it is straightforward to recognize a relation between a particular departure and a particular time (a, b), we need a further level of abstraction to recognize a relation between a regularly occurring event and a time of day (c), and yet further relations to interpret (d)–(g). One could simply list these additional relations, but that would not jibe with the intuition that the latter relations are parasitic on the earlier ones, e.g., the intuition that we refer thus to the *8 am passengers* only because they travel on the train that departs (or arrives) at 8 am. Formally, this amounts to the coercion of a temporal relation (Pustejovsky 1991), and disambiguation would appear to benefit from it.

4 Summary and Conclusions

The purpose of the present paper has been the investigation of the interaction and proper division of labor between grammar, meaning and knowledge representation in natural language processing. We examined disambiguation as an area in which all three components have a potential role to play—grammar because it stands to benefit from disambiguation, and semantics and knowledge (domain) representation as the source of disambiguating information.

We formulated a proposal for encoding semantic and domain information in feature structures, arguing, e.g., that "semantic" feature structures are best conceived as constraints on logical form. This is equivalent to the view that the feature description language functions as a formalized metalanguage for an arbitrary semantic representation language. This view of the use of semantic feature structures recommends itself by the relatively few assumptions it makes about the nature of the semantic representation language itself—in particular, it need not be limited to the expressive power of the feature logic metalanguage. A further advantage for this view is the opportunity it affords for the characterization of semantic ambiguity. In an appendix we illustrate the proposal using an application to the language of generalized quantifiers. Nerbonne 1992b, using the relatively rich HPSG formalism, demonstrated a feature-based approach to the characterization of scopally underspecified formulas which eschews the level of "quasi-logical forms".

We demonstrate the applicability of this approach to problems of disambiguation, noting however, that serious gaps in the treatment nonetheless arose—primarily due to the weakness of the feature formalism. A more comprehensive treatment must await further research.

References

Allgayer, J., and C. Reddig. 1990. What KL-ONE Lookalikes Need to Cope with Natural Language—Scope and Aspect of Plural Noun PHrases. Technical Report 66, Sonderforschungsbereich 314, Universität des Saarlandes, FB 10 Informatik IV.

Alshawi, H., et al. July 1989. Research Programme in Natural Language Processing. Final report, Alvey Project No. ALV/PRJ/IKBS/105, SRI Cambridge Research Centre.

Baader, F., and B. Hollunder. 1990. KRIS: Knowledge Representation and Inference System—System Description. Technical memo, Deutsches Forschungszentrum für Künstliche Intelligenz, Kaiserslautern.

Backofen, R., H. Trost, and H. Uszkoreit. 1991. Linking Typed Feature Formalisms and Terminological Knowledge Representation Languages in Natural Language Front-Ends. In *Proceedings of the GI Congress, Knowledge-Based Systems 1991*, ed. W.Bauer. Berlin. Springer.

Barwise, J. 1989. Conditionals and Conditional Information. In *The Situation in Logic*, ed. J. Barwise, 97–136. Stanford: CSLI.

Barwise, J., and R. Cooper. 1981. Generalized Quantifiers and Natural Language. *Linguistics and Philosophy* 4(2):159–219.

Barwise, J., and J. Perry. 1983. *Situations and Attitudes*. Cambridge: MIT Press.

Bobrow, R. 1979. Semantic Interpretation in PSI-KLONE. Annual Report 4274, Bolt, Beranek and Newman, Cambridge, MA.

Brachman, R. J., and J. G. Schmolze. 1985. An overview of the KL-ONE knowledge representation system. *Cognitive Science* 9:171–216.

Carpenter, B. to appear, 1992. *The Logic of Typed Feature Structures.* Tracts in Theoretical Computer Science. Cambridge: Cambridge University Press.

Clark, H., and E. Clark. 1977. *Psychology and Language: An Introduction to Psycholinguistics.* New York: Harcourt Brace Jovanovich.

Creary, L. G., and C. J. Pollard. 1985. A Computational Semantics for Natural Language. In *Proceedings of the 25th Annual Meeting of the Association for Computational Linguistics*, 172–179.

Dalrymple, M., S. M. Shieber, and F. C.N.Periera. 1991. Ellipsis and Higher-Order Unification. *Linguistics and Philosophy* 14(4):399–452.

Dowty, D. 1991. Thematic Proto-Roles and Argument Selection. *Language* 67(3).

Evans, R., and G. Gazdar. 1990. The DATR Papers. Technical Report SCRP 139, School of Cognitive and Computing Sciences, University of Sussex.

Fenstad, J. E., P.-K. Halvorsen, T. Langholm, and J. van Benthem. 1987. *Situations, Language, and Logic.* Dordrecht: Reidel.

Iida, M., J. Nerbonne, D. Proudian, and D. Roberts. 1989. Accommodating Complex Applications. In *Proceedings of the First International Workshop on Lexical Acquisition, IJCAI 1989*, ed. U. Zernik. Menlo Park. Morgan Kaufman.

Jackendoff, R. 1972. *Semantics Interpretation in Generative Grammar.* Cambridge: MIT Press.

Johnson, M. 1988. *Attribute Value Logic and the Theory of Grammar.* Stanford: Center for the Study of Language and Information.

Karttunen, L. 1983. KIMMO: A General Morphological Processor. *Texas Linguistic Forum* 22:167–186.

Moens, M., J. Calder, E. Klein, M. Reape, and H. Zeevat. 1989. Expressing Generalizations in Unification-Based Grammar Formalisms. In *Proceedings of the 4th Meeting of the European Chapter of the Association for Computational Linguistics*, 174–181.

Moore, R. C. 1981. Problems in Logical Form. In *Proceedings of the 19th Annual Meeting of the Association for Computational Linguistics*, 117–124.

Moore, R. C. 1989. Unification-Based Semantic Interpretation. In *Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics*, 33–41.

Nebel, B., and G. Smolka. 1990. Representation and Reasoning with Attributive Descriptions. In *Sorts and Types in Artificial Intelligence*, ed. K.H.Blaesius, U.Hedstück, and C.R.Rollinger. Berlin: Springer.

Nerbonne, J. 1992a. Constraint-Based Semantics. In *Proceedings of the 8th Amsterdam Colloquium*, ed. P. Dekker and J. van der Does, xxx–yyy.

Nerbonne, J. 1992b. A Feature-Based Syntax/Semantics Interface. In *Proceedings of the Second International Conference on the Mathematics of Language*, ed. A. Manaster-Ramer and W. Zadrozny, xxx–yyy. Annals of Mathematics and Artificial Intelligence.

Pollard, C., and I. Sag. 1987. *An Information-Based Theory of Syntax and Semantics, Vol.I.* Stanford: CSLI.

Pustejovsky, J. 1991. The Generative Lexicon. *Computational Linguistics* 17(4):409–441.

Roberts, D. 1991. Linking Thematic Roles and Syntactic Arguments in HPSG. Master's thesis, Cornell University.

Scha, R. 1976. Semantic Types in PHILQA1. In *COLING 76 Preprints.* Ottawa.

Schubert, L. K., and F. J. Pelletier. 1982. From English to Logic: Context-Free Computation of Conventional Logic Translations. *Journal of the Association for Computational Linguistics* 165–176.

Shieber, S. 1986. *An Introduction to Unification-Based Approaches to Grammar.* Stanford University: Center for the Study of Language and Information.

Shieber, S., H. Uszkoreit, J. Robinson, and M. Tyson. 1983. The Formalism and Implementation of PATR-II. In *Research on Interactive Acquisition and Use of Knowledge.* Menlo Park, Cal.: AI Center, SRI International.

Smolka, G. 1988. A Feature Logic with Subsorts. Technical Report 33, WT LILOG–IBM Germany.

Wechsler, S. M. 1991. *Argument Structure and Linking.* PhD thesis, Stanford University.

Zeevat, H. W. 1991. *Aspects of Discourse Semantics and Unification Grammar.* PhD thesis, University of Amsterdam.

Zeevat, H., E. Klein, and J. Calder. 1987. Unification Categorial Grammar. Research Paper EUCCS/RP-21, Centre for Cognitive Science, University of Edinburgh.

Zwicky, A. M., and J. Sadock. 1975. Ambiguity Tests and How to Fail Them. In *Syntax and Semantics, Vol. IV*, ed. J. Kimball, 1–36. New York: Academic Press.

Appendix: An Illustration

This appendix repeats (and simplifies) material from Nerbonne 1992b and is provided here only for the sake of concrete illustration.

We illustrate the view that semantics involves the accumulation of constraints expressed about a semantic representation language, by applying the view to a well-known semantic representation language, the language of generalized quantifiers; second, by developing its metalanguage within the feature description language used in HPSG; and third by demonstrating its application to the problem of scope ambiguity.

Logic—A Generalized Quantifier Language

To illustrate the approach we shall first need a target semantic representation language. Here we deliberately use a popular semantic representation scheme—a version of the language of generalized quantifiers; this is a kind of *lingua franca* among theoretically oriented computational linguists. We emphasize that the overall scheme—that of employing feature structure descriptions as a formalized metalanguage for a semantic representation logic—is general, and could easily be applied to other logics, e.g., first order logic, higher-order or intensional logics, discourse representation structures, or a language of situation theory. It could even be applied to nonlogical representations, but that would be harder to motivate.

A BNF for a Language of Generalized Quantifiers—LGQ

$$\langle\text{var}\rangle ::= x_0 \mid x_1 \mid \ldots$$

$$\langle\text{const}\rangle ::= c_0 \mid c_1 \mid \ldots$$

$$\langle\text{pred}\rangle ::= P_0 \mid P_1 \mid \ldots$$

$$\langle\text{term}\rangle ::= \langle\text{var}\rangle \mid \langle\text{const}\rangle$$

$$\langle\text{atomic-wff}\rangle ::= (\langle\text{pred}\rangle\langle\text{term}\rangle^*)$$

$$\langle\text{conn}\rangle ::= \wedge \mid \vee \mid \leftrightarrow$$

$$\langle\text{det}\rangle ::= \forall \mid \exists \mid \mathit{MOST} \mid \ldots$$

$$\langle\text{g-quant}\rangle ::= \langle\text{det}\rangle\langle\text{var}\rangle\langle\text{wff}\rangle$$

$$\begin{aligned}\langle\text{wff}\rangle ::= \;& \langle\text{atomic-wff}\rangle\\ & \mid (\langle\text{wff}\rangle\langle\text{conn}\rangle\langle\text{wff}\rangle)\\ & \mid (\neg\langle\text{wff}\rangle)\\ & \mid (\langle\text{wff}\rangle \;\rightarrow\; \langle\text{wff}\rangle)\\ & \mid (\langle\text{g-quant}\rangle\langle\text{wff}\rangle)\end{aligned}$$

There is one detail about this syntax definition which may seem peculiar to non-computational semanticists. The definition foresees quantified formulas not in the Barwise-Cooper notation (cf. Barwise and Cooper 1981) where the quantified formula normally had λ-terms in the restrictor and scope, but rather in the notation more frequently used in computational linguistics, in which these positions are filled by open formulas. The formula below highlights the difference:

Barwise and Cooper Notation	**PresentNotation**
$\forall x(\lambda y.\mathbf{man}(y),\ \lambda z.\mathbf{mortal}(z))$	$(\forall x\ \mathbf{man}(x)\,\mathbf{mortal}(x))$

This is generally preferred in order to keep the visual complexity of formulas (the number of λ's) at a minimum. Cf. Moore 1981 for an early use; and Dalrymple et al. 1991, pp.414-17 for model-theoretic definitions of the alternative forms.

We turn now to the description of expressions in this language using a typed feature description language of the sort common in grammar processing.

Metalogic—AVM Specifications for LGQ

In this section we employ typed feature logic (often referred to as ATTRIBUTE-VALUE MATRICES or AVMs) to provide a set of type definitions for expressions in the logical language just presented. We shall not present the typed feature logic formally, relying on

(Carpenter to appear 1992) for definitions. The initial specifications are limited to very vanilla-flavored uses of the feature description scheme, and we shall warn when more particular assumptions are made (Section 5).

$$\langle\text{var}\rangle ::= x_0 \mid x_1 \mid \ldots \qquad \left[\begin{array}{l} \textit{var} \\ \text{INDEX} \end{array}\right]$$

$$\langle\text{atomic-wff}\rangle ::= (\langle\text{pred}\rangle\langle\text{term}\rangle^*) \qquad \left[\begin{array}{l} \textit{atomic-wff} \\ \text{PRED } [\textit{pred}] \\ \text{FIRST } [\textit{term}] \\ \vdots \\ \text{N-TH } [\textit{term}] \end{array}\right]$$

$$\langle\text{g-quant}\rangle ::= \langle\text{det}\rangle\langle\text{var}\rangle\langle\text{wff}\rangle \qquad \left[\begin{array}{l} \textit{gen-quant} \\ \text{DET } [\textit{det}] \\ \text{VAR } [\textit{var}] \\ \text{REST } [\textit{wff}] \end{array}\right]$$

$$\begin{array}{l} \langle\text{wff}\rangle ::= \;\; \ldots \\ \qquad\quad \mid (\langle\text{wff}\rangle\langle\text{conn}\rangle\langle\text{wff}\rangle) \end{array} \qquad \left[\begin{array}{l} \textit{connective-wff} \\ \text{CONN } [\textit{conn}] \\ \text{WFF1 } [\textit{wff}] \\ \text{WFF2 } [\textit{wff}] \end{array}\right]$$

$$\ldots \mid (\neg\langle\text{wff}\rangle) \qquad \left[\begin{array}{l} \textit{negation} \\ \text{SCOPE-WFF } [\textit{wff}] \end{array}\right]$$

$$\ldots \mid (\langle\text{wff}\rangle \rightarrow \langle\text{wff}\rangle) \qquad \left[\begin{array}{l} \textit{implication} \\ \text{ANTE } [\textit{wff}] \\ \text{CONSEQ } [\textit{wff}] \end{array}\right]$$

$$\ldots \mid (\langle\text{g-quant}\rangle\langle\text{wff}\rangle) \qquad \left[\begin{array}{l} \textit{q-wff} \\ \text{QUANT } [\textit{gen-quant}] \\ \text{SCOPE } [\textit{wff}] \end{array}\right]$$

We shall make use of TYPE predicates, e.g., *var*(x), which holds iff x is of type *var*. It is easy to note the absence of several expression types from the set of feature-structure types defined; for example, terms, predicate and individual constants have not been defined here. That is because we rely on TYPE information for distinctions which are not realized in one or more distinct attributes. Figure 6 illustrates the type hierarchy we assume. Of course, we can express the type hierarchy in the language of typed feature descriptions—and in this way obtain a specification fully equivalent to the BNF. For example, in Carpenter to appear 1992 we can specify that

$$\textit{term}(x) \leftrightarrow \textit{var}(x) \vee \textit{const}(x)$$

For the purposes of this paper, we may rely on the informal presentation in Figure 3.

Use of Semantic Descriptions—Examples

Some simple examples of the kinds of metalinguistic specifications allowed are illustrated in Figure 7. These would be compiled in ways suggested by Figure 2; i.e., we imagine that construction principles (or grammatical rules) may have a semantic correlate which constrains

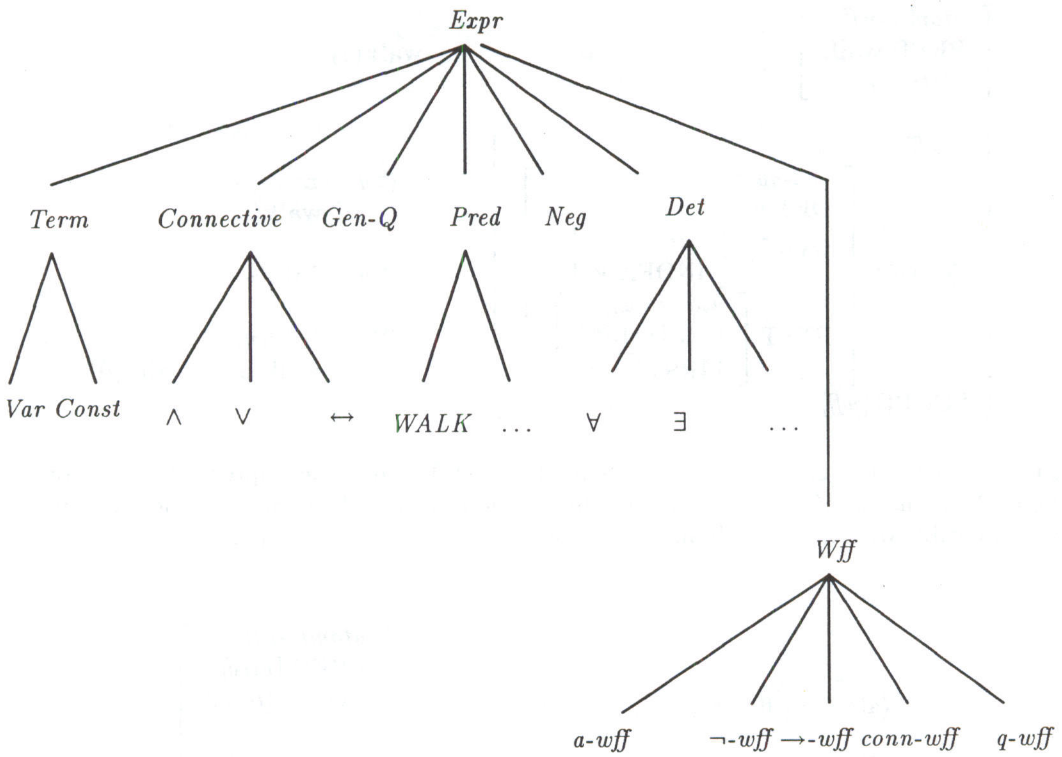


Figure 6: A type hierarchy for the domain of expressions in the language of generalized quantifiers as defined in the text. Carpenter 1992 provides the theory of typed feature structures within which a hierarchy such as this functions.

the semantic representation which is the meaning of the construction. Of course, as we noted above, there is no reason that only syntax should have the privilege of constraining meaning.

The use of feature structures as a metalinguistic level of semantic representation allows greater freedom in semantic explanations. This may be illustrated with respect to the representation of argument positions in atomic formulas and their specifications in the current scheme. The semantic representation language LGQ uses order-coding to represent which arguments are bound to which argument positions, while the metalanguage (feature structure descriptions) identifies this using features (FIRST, etc.). The first bit of freedom we might exercise concerns the identification of argument positions. Nothing would stand in the way of using more contentful-sounding role names to pick these out. For example, we might alter the feature specifications in such a way as to allow the following:

$$\begin{bmatrix} \textit{atomic-wff} \\ \text{PRED } \textbf{send} \\ \text{SOURCE } x \\ \text{THEME } y \\ \text{GOAL } z \end{bmatrix}$$

In this case the simplest generalization would appear to be that allowing any name to designate a semantic role. We would again represent roles as features, postponing any more detailed representation until it is motivated:

$$\begin{bmatrix} \textit{atomic-wff} \\ \text{PRED } \textbf{walk} \\ \text{FIRST } j \end{bmatrix} \qquad \textbf{walk}(j)$$

$$\begin{bmatrix} \textit{q-wff} & \\ \text{QUANT} & \begin{bmatrix} \textit{gen-quant} & \\ \text{DET } \forall & \\ \text{VAR } \boxed{1} & \begin{bmatrix} \textit{var} \\ \text{INDEX y} \end{bmatrix} \\ \text{REST} & \begin{bmatrix} \textit{atomic-wff} \\ \text{PRED } \textbf{child} \\ \text{FIRST } \boxed{1} \end{bmatrix} \end{bmatrix} \\ \text{SCOPE } [\textit{wff}] & \end{bmatrix}$$

$$(\forall y \quad \textbf{child}(y) \quad \textbf{walk}(y))$$

$$(\forall y \quad \textbf{child}(y) \quad \phi)$$

$$(\forall y \quad \textbf{child}(y) \quad \textbf{walk}(y) \wedge \textbf{talk}(y))$$

Figure 7: Feature structure descriptions used as metalinguistic descriptors of expressions in LGQ. Note the use of underspecification in the last example. The underspecified description is compatible with many, including semantically contradictory formulas.

$$\langle\text{atomic-wff}\rangle ::= (\langle\text{pred}\rangle\langle\text{pair}\rangle^{*}) \qquad \begin{bmatrix} \textit{atomic-wff} \\ \text{PRED } [\textit{pred}] \\ \text{ROLE}_1 \ [\textit{term}] \\ \vdots \\ \text{ROLE}_n \ [\textit{term}] \end{bmatrix}$$

Although we will not make use of this representation below, it illustrates a degree of freedom allowed by the present semantic representation scheme but absent from simpler ones. It would moreover appear to suffice for all the syntactic purposes for which so-called thematic roles are deployed (cf. Jackendoff 1972, 29-46; Roberts 1991; Dowty 1991; Wechsler 1991).

On the other hand, there are purely semantic grounds for preferring keyword-coding to order-coding as a way of identifying argument positions. Two such reasons often adduced are (i) that the ORDER of arguments in relations is never used semantically, and thus that every alternative ordering leads to a perfectly equivalent logic, so that the keyword-coding suffices; and (ii) that the use of keyword coding allows us to make sense of anadic predication (i.e., using the same predicate with a variable number of arguments). On the latter point, see Creary and Pollard 1985.

An extension of the illustration provided here may be found in Nerbonne 1992b, which furthermore demonstrates how, using more expressive mechanisms of the HPSG feature formalism (Pollard and Sag 1987), scope ambiguity may be characterized at the level of feature structures.

Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

DFKI
-Bibliothek-
PF 2080
D-6750 Kaiserslautern
FRG

DFKI Publikationen

Die folgenden DFKI Veröffentlichungen sowie die aktuelle Liste von allen bisher erschienenen Publikationen können von der oben angegebenen Adresse bezogen werden.
Die Berichte werden, wenn nicht anders gekennzeichnet, kostenlos abgegeben.

DFKI Publications

The following DFKI publications or the list of all publisched papers so far can be ordered from the above address.
The reports are distributed free of charge except if otherwise indicated.

DFKI Research Reports

RR-91-08
Wolfgang Wahlster, Elisabeth André, Som Bandyopadhyay, Winfried Graf, Thomas Rist:
WIP: The Coordinated Generation of Multimodal Presentations from a Common Representation
23 pages

RR-91-09
Hans-Jürgen Bürckert, Jürgen Müller, Achim Schupeta: RATMAN and its Relation to Other Multi-Agent Testbeds
31 pages

RR-91-10
Franz Baader, Philipp Hanschke: A Scheme for Integrating Concrete Domains into Concept Languages
31 pages

RR-91-11
Bernhard Nebel: Belief Revision and Default Reasoning: Syntax-Based Approaches
37 pages

RR-91-12
J.Mark Gawron, John Nerbonne, Stanley Peters:
The Absorption Principle and E-Type Anaphora
33 pages

RR-91-13
Gert Smolka: Residuation and Guarded Rules for Constraint Logic Programming
17 pages

RR-91-14
Peter Breuer, Jürgen Müller: A Two Level Representation for Spatial Relations, Part I
27 pages

RR-91-15
Bernhard Nebel, Gert Smolka:
Attributive Description Formalisms ... and the Rest of the World
20 pages

RR-91-16
Stephan Busemann: Using Pattern-Action Rules for the Generation of GPSG Structures from Separate Semantic Representations
18 pages

RR-91-17
Andreas Dengel, Nelson M. Mattos:
The Use of Abstraction Concepts for Representing and Structuring Documents
17 pages

RR-91-18
John Nerbonne, Klaus Netter, Abdel Kader Diagne, Ludwig Dickmann, Judith Klein:
A Diagnostic Tool for German Syntax
20 pages

RR-91-19
Munindar P. Singh: On the Commitments and Precommitments of Limited Agents
15 pages

RR-91-20
Christoph Klauck, Ansgar Bernardi, Ralf Legleitner
FEAT-Rep: Representing Features in CAD/CAM
48 pages

RR-91-21
Klaus Netter: Clause Union and Verb Raising Phenomena in German
38 pages

RR-91-22
Andreas Dengel: Self-Adapting Structuring and Representation of Space
27 pages

RR-91-23
Michael Richter, Ansgar Bernardi, Christoph Klauck, Ralf Legleitner: Akquisition und Repräsentation von technischem Wissen für Planungsaufgaben im Bereich der Fertigungstechnik
24 Seiten

RR-91-24
Jochen Heinsohn: A Hybrid Approach for Modeling Uncertainty in Terminological Logics
22 pages

RR-91-25
Karin Harbusch, Wolfgang Finkler, Anne Schauder: Incremental Syntax Generation with Tree Adjoining Grammars
16 pages

RR-91-26
M. Bauer, S. Biundo, D. Dengler, M. Hecking, J. Koehler, G. Merziger:
Integrated Plan Generation and Recognition
- A Logic-Based Approach -
17 pages

RR-91-27
A. Bernardi, H. Boley, Ph. Hanschke, K. Hinkelmann, Ch. Klauck, O. Kühn, R. Legleitner, M. Meyer, M. M. Richter, F. Schmalhofer, G. Schmidt, W. Sommer:
ARC-TEC: Acquisition, Representation and Compilation of Technical Knowledge
18 pages

RR-91-28
Rolf Backofen, Harald Trost, Hans Uszkoreit:
Linking Typed Feature Formalisms and Terminological Knowledge Representation Languages in Natural Language Front-Ends
11 pages

RR-91-29
Hans Uszkoreit: Strategies for Adding Control Information to Declarative Grammars
17 pages

RR-91-30
Dan Flickinger, John Nerbonne:
Inheritance and Complementation: A Case Study of *Easy* Adjectives and Related Nouns
39 pages

RR-91-31
H.-U. Krieger, J. Nerbonne:
Feature-Based Inheritance Networks for Computational Lexicons
11 pages

RR-91-32
Rolf Backofen, Lutz Euler, Günther Görz:
Towards the Integration of Functions, Relations and Types in an AI Programming Language
14 pages

RR-91-33
Franz Baader, Klaus Schulz:
Unification in the Union of Disjoint Equational Theories: Combining Decision Procedures
33 pages

RR-91-34
Bernhard Nebel, Christer Bäckström:
On the Computational Complexity of Temporal Projection and some related Problems
35 pages

RR-91-35
Winfried Graf, Wolfgang Maaß: Constraint-basierte Verarbeitung graphischen Wissens
14 Seiten

RR-92-01
Werner Nutt: Unification in Monoidal Theories is Solving Linear Equations over Semirings
57 pages

RR-92-02
Andreas Dengel, Rainer Bleisinger, Rainer Hoch, Frank Hönes, Frank Fein, Michael Malburg:
Π_{ODA}: The Paper Interface to ODA
53 pages

RR-92-03
Harold Boley:
Extended Logic-plus-Functional Programming
28 pages

RR-92-04
John Nerbonne: Feature-Based Lexicons: An Example and a Comparison to DATR
15 pages

RR-92-05
Ansgar Bernardi, Christoph Klauck, Ralf Legleitner, Michael Schulte, Rainer Stark:
Feature based Integration of CAD and CAPP
19 pages

RR-92-07
Michael Beetz:
Decision-theoretic Transformational Planning
22 pages

RR-92-08
Gabriele Merziger: Approaches to Abductive Reasoning - An Overview -
46 pages

RR-92-09
Winfried Graf, Markus A. Thies:
Perspektiven zur Kombination von automatischem Animationsdesign und planbasierter Hilfe
15 Seiten

RR-92-11
Susane Biundo, Dietmar Dengler, Jana Koehler: Deductive Planning and Plan Reuse in a Command Language Environment
13 pages

RR-92-13
Markus A. Thies, Frank Berger: Planbasierte graphische Hilfe in objektorientierten Benutzungsoberflächen
13 Seiten

RR-92-14
Intelligent User Support in Graphical User Interfaces:

1. InCome: A System to Navigate through Interactions and Plans
   Thomas Fehrle, Markus A. Thies
2. Plan-Based Graphical Help in Object-Oriented User Interfaces
   Markus A. Thies, Frank Berger

22 pages

RR-92-15
Winfried Graf: Constraint-Based Graphical Layout of Multimodal Presentations
23 pages

RR-92-17
Hassan Aït-Kaci, Andreas Podelski, Gert Smolka: A Feature-based Constraint System for Logic Programming with Entailment
23 pages

RR-92-18
John Nerbonne: Constraint-Based Semantics
21 pages

RR-92-19
Ralf Legleitner, Ansgar Bernardi, Christoph Klauck
PIM: Planning In Manufacturing using Skeletal Plans and Features
17 pages

RR-92-20
John Nerbonne: Representing Grammar, Meaning and Knowledge
18 pages

DFKI Technical Memos

TM-91-05
Jay C. Weber, Andreas Dengel , Rainer Bleisinger: Theoretical Consideration of Goal Recognition Aspects for Understanding Information in Business Letters
10 pages

TM-91-06
Johannes Stein: Aspects of Cooperating Agents
22 pages

TM-91-08
Munindar P. Singh: Social and Psychological Commitments in Multiagent Systems
11 pages

TM-91-09
Munindar P. Singh: On the Semantics of Protocols Among Distributed Intelligent Agents
18 pages

TM-91-10
Béla Buschauer, Peter Poller, Anne Schauder, Karin Harbusch: Tree Adjoining Grammars mit Unifikation
149 pages

TM-91-11
Peter Wazinski: Generating Spatial Descriptions for Cross-modal References
21 pages

TM-91-12
Klaus Becker, Christoph Klauck, Johannes Schwagereit: FEAT-PATR: Eine Erweiterung des D-PATR zur Feature-Erkennung in CAD/CAM
33 Seiten

TM-91-13
Knut Hinkelmann:
Forward Logic Evaluation: Developing a Compiler from a Partially Evaluated Meta Interpreter
16 pages

TM-91-14
Rainer Bleisinger, Rainer Hoch, Andreas Dengel: ODA-based modeling for document analysis
14 pages

TM-91-15
Stefan Bussmann: Prototypical Concept Formation
An Alternative Approach to Knowledge Representation
28 pages

TM-92-01
Lijuan Zhang:
Entwurf und Implementierung eines Compilers zur Transformation von Werkstückrepräsentationen
34 Seiten

DFKI Documents

D-91-06
Gerd Kamp: Entwurf, vergleichende Beschreibung und Integration eines Arbeitsplanerstellungssystems für Drehteile
130 Seiten

D-91-07
Ansgar Bernardi, Christoph Klauck, Ralf Legleitner
TEC-REP: Repräsentation von Geometrie- und Technologieinformationen
70 Seiten

D-91-08
Thomas Krause: Globale Datenflußanalyse und horizontale Compilation der relational-funktionalen Sprache RELFUN
137 Seiten

D-91-09
David Powers, Lary Reeker (Eds.):
Proceedings MLNLO'91 - Machine Learning of Natural Language and Ontology
211 pages
Note: This document is available only for a nominal charge of 25 DM (or 15 US-$).

D-91-10
Donald R. Steiner, Jürgen Müller (Eds.):
MAAMAW'91: Pre-Proceedings of the 3rd European Workshop on „Modeling Autonomous Agents and Multi-Agent Worlds"
246 pages
Note: This document is available only for a nominal charge of 25 DM (or 15 US-$).

D-91-11
*Thilo C. Horstmann:*Distributed Truth Maintenance
61 pages

D-91-12
Bernd Bachmann:
$\mathrm{H}^{\mathrm{iera}}\mathrm{C}_{\mathrm{on}}$ - a Knowledge Representation System with Typed Hierarchies and Constraints
75 pages

D-91-13
International Workshop on Terminological Logics
Organizers: Bernhard Nebel, Christof Peltason, Kai von Luck
131 pages

D-91-14
Erich Achilles, Bernhard Hollunder, Armin Laux, Jörg-Peter Mohren: KRIS : Knowledge Representation and Inference System
- Benutzerhandbuch -
28 Seiten

D-91-15
Harold Boley, Philipp Hanschke, Martin Harm, Knut Hinkelmann, Thomas Labisch, Manfred Meyer, Jörg Müller, Thomas Oltzen, Michael Sintek, Werner Stein, Frank Steinle:
μCAD2NC: A Declarative Lathe-Worplanning Model Transforming CAD-like Geometries into Abstract NC Programs
100 pages

D-91-16
Jörg Thoben, Franz Schmalhofer, Thomas Reinartz:
Wiederholungs-, Varianten- und Neuplanung bei der Fertigung rotationssymmetrischer Drehteile
134 Seiten

D-91-17
Andreas Becker:
Analyse der Planungsverfahren der KI im Hinblick auf ihre Eignung für die Abeitsplanung
86 Seiten

D-91-18
Thomas Reinartz: Definition von Problemklassen im Maschinenbau als eine Begriffsbildungsaufgabe
107 Seiten

D-91-19
Peter Wazinski: Objektlokalisation in graphischen Darstellungen
110 Seiten

D-92-01
Stefan Bussmann: Simulation Environment for Multi-Agent Worlds - Benutzeranleitung
50 Seiten

D-92-02
Wolfgang Maaß: Constraint-basierte Plazierung in multimodalen Dokumenten am Beispiel des Layout-Managers in WIP
111 Seiten

D-92-03
Wolfgan Maaß, Thomas Schiffmann, Dudung Soetopo, Winfried Graf: LAYLAB: Ein System zur automatischen Plazierung von Text-Bild-Kombinationen in multimodalen Dokumenten
41 Seiten

D-92-06
Hans Werner Höper: Systematik zur Beschreibung von Werkstücken in der Terminologie der Featuresprache
392 Seiten

D-92-08
Jochen Heinsohn, Bernhard Hollunder (Eds.):
DFKI Workshop on Taxonomic Reasoning Proceedings
56 pages

D-92-09
Gernod P. Laufkötter: Implementierungsmöglichkeiten der integrativen Wissensakquisitionsmethode des ARC-TEC-Projektes
86 Seiten

D-92-21
Anne Schauder: Incremental Syntactic Generation of Natural Language with Tree Adjoining Grammars
57 pages

Representing Grammar, Meaning and Knowledge

John Nerbonne

RR-92-20

Research Report