

On the Interaction of Syntax and Semantics in a Syntactically Guided Caseframe Parser

Harald Trost, Ernst Buchberger, Wolfgang Heinz

Austrian Research Institute for Artificial Intelligence
Schottengasse 3, A-1010 Wien
Austria

Abstract:

In this paper we describe a parser for German based on semantic caseframe instantiation guided by a syntactic analyzer. Pure caseframe parsers lack the ability to capture syntactic regularities, which leads to redundancy in the lexicon and/or poor syntactic coverage. By combining caseframe matching with an explicit syntactic analysis our parser overcomes this problem. Approaches well suited for English are not easily transported to German with its rich morphology and its free constituent order at the clause level. Our parser which incorporates two different interacting parsing strategies is well adapted to the needs posed by German grammar.

we believe that the present understanding of structural differences between languages does not yet allow for a single parsing algorithm, at least if one wants both good coverage and efficiency. As a consequence we developed a parser which is specifically designed to cope with the peculiarities of the German language. Nevertheless, since our approach is based on sound linguistic principles, most of the solutions found could be applied to other languages with a similar structure as well.

In this paper we will focus on the core of the system's parsing component and neglect other features like spelling correction, treatment of anaphoric or elliptic utterances, quantifier scoping and the transformation into SQL. The overall system architecture is depicted in

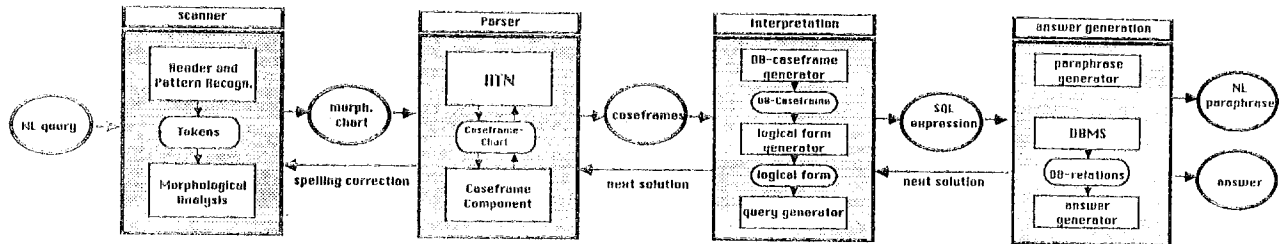


Fig.1: System Architecture

1. Introduction

DB-DIALOG is a German language interface to relational data bases. Our objectives were to design a system which has good language capabilities and which at the same time is easily portable. The system has been developed on a SYMBOLICS 3600 and up to now has been transported to a PRIME-550II, a DEC-VAX 730, and a NIXDORF TARGON-35.

DB-DIALOG translates user-queries given in the form of written German sentences into structured SQL statements. Since SQL is the de-facto standard query language for relational database systems, a wide range of database systems is accessible. The only adaptation to be done is a transformation of the structured SQL output by DB-DIALOG into the special SQL used by the specific DBMS. At the moment versions for ORACLE, REFLEX and MIMER are implemented.

In some other ways the interface is also designed to be as portable as possible. Adaptation to new domains is facilitated by keeping the linguistic coverage separate from the actual domain knowledge which rests solely in the lexicon. Independence from the modeling of the domain in the data base is achieved by distinguishing between a linguistically motivated description of the domain and a database-oriented one. There is an explicit translation step between these two parts.

Language independence is not aimed at, because

figure 1. For a description of the interface as a whole see Buchberger et al./1987/.

We have chosen to base our parser on semantic caseframe instantiation. Such an approach is well suited for a restricted domain parser, because of its efficiency (by avoiding useless parses in case of syntactic ambiguity) and its robustness in the case of ungrammaticality (see e.g. Grishman et al./1986/). On the other hand, relying solely on that method, it would be difficult to capture syntactic generalities (e.g. active-passive transformation), because syntactic as well as semantic restrictions must be specified explicitly for each slot of every caseframe. This means that for every different syntactic realization of the same statement a different caseframe has to be provided in the lexicon. There are two severe drawbacks to this kind of realization: First, general syntactic properties of the language are implicitly stated in the lexicon entries instead of explicitly in the grammar leading to a possibly inconsistent and patchy syntactic coverage. Second, the lexicon is inflated because for a single word (meaning) a number of different caseframes is needed.

To illustrate the problem let's have a look at an example:

'liefern' (= to deliver)
could have the following caseframe:

```
(LIEFERN
 (AGENT (SYNTAX NP/NOM) (SEMANTICS COMPANY))
 (OBJECT (SYNTAX NP/ACC) (SEMANTICS GOODS)))
```

This would enable a parser to analyze a sentence like:

Firma XYZ liefert Kuehlschraenke.
(XYZ company delivers refrigerators.)

But there are various syntactic variations of this statement:

- passivization
Die Kuehlschraenke werden von der Firma XYZ geliefert.
(The refrigerators are delivered by the XYZ company.)
- relative clause (active and passive)
Die Kuehlschraenke, die die Firma XYZ liefert, ...
(The refrigerators, that XYZ company delivers, ...)
Die Firma XYZ, die Kuehlschraenke liefert, ... (XYZ company, that delivers refrigerators...)
- attributive (active and passive)
Die von der Firma XYZ gelieferten Kuehlschraenke...
(The refrigerators delivered by the XYZ company...)
Die Kuehlschraenke liefernde Firma XYZ...
(The XYZ company delivering refrigerators...)

As the example shows, there are six different syntactic forms which may occur with the same verb due to syntactic variations. Having six different caseframes for just one word (meaning) is ridiculous. Several improvements have been proposed which enable caseframe parsers to deal with syntax in a less ad-hoc manner, see e.g. Hayes et al./1985/ and Lytinen /1986/.

In our approach we went one step further in this direction by combining a caseframe matcher with a syntactic parser that is guiding the analysis process as a whole. Interaction with the caseframe matcher includes transformations being applied to the syntactic restrictions of the caseframes involved. That way different syntactic constructions like active, passive, attributive, relative clause and even nominalizations are handled without the need for different caseframes.

2. Language Specific Aspects

Using German as input language to our interface calls for solutions to problems which do not arise for the English language. The most prominent differences are:

- there is a rich morphology,
- constituent order at the clause level is fairly free, and
- there is the verb-second phenomenon in main clauses.

Morphology is dealt with in the morphological component of the scanner /Trost and Dorffner 1987/. This scanner passes information about case markers (as well as other syntactic features) to the parser, but - if interpreted locally - this information is usually highly ambiguous.

As for word order, there are basically two phrase types in German: noun-dependent phrases, like noun phrase (NP) and prepositional phrase (PP), with a rather rigid word order, and clause-like phrases, like sentence (S) and adjective phrase (AP), with at best a preferred ordering of constituents. For a discussion of word order in German cf. Hoehle /1982/ and, for a more

computationally oriented view Uszkoreit /1986/ and Hauenschild /1986/.

Closer inspection shows that on the one hand part of the NPs, namely APs embedded in them, exhibit free constituent order, whereas on the other hand clause-like constructions appear to have one fixed position: the head (verbal complex and adjective respectively) which always comes last. There is the only exception that in main clauses the inflected part of the verbal complex moves to second position /Haider 1984/.

In parsing a language like German one therefore needs two different (contradicting) strategies:

- one for the fixed word order of arguments inside constituents (i.e. determiner and attribute of NPs)
- one for the free constituent order of the arguments and modifiers of predicates (i.e. the constituents of S).

Our solution to this problem is the interaction of two different techniques in our parser. For processing constituents with fixed word order we chose the Augmented Transition Network (ATN) formalism /Bates 1978/, because ATNs are a well understood algorithm with very efficient implementation techniques available, and they provide for a relatively transparent notation. Since we use the ATN only for a part of the syntactic parsing which itself interacts closely with semantics, the known weaknesses inherent to ATNs do not pose a problem in the context of our parser. For free-order constituents on the other hand we use a unification-based strategy which makes heavy use of a case frame matcher. We will first describe both components in some detail and then demonstrate how they interact.

3. The ATN component

Our ATN consists of the usual subnets for phrase-types (NP, AP, S, etc.). In contrast to the standard approach it works on a chart of morphological entries created by the morphological component mentioned earlier. This chart may contain ambiguities which the ATN is extended to cope with.

Since the ATN aims at the construction of functional dependencies (an argument/modifier - head structure) which is greatly eased by knowing the head /Prouidian and Pollard 1985/, we decided to use head-driven analysis in the ATN. German is basically a subject-object-verb (SOV) language, that means the head of a phrase comes last with few exceptions. These exceptions are:

- NPs may have postmodifiers (genitive NPs, PPs, relative clauses),
- in PPs the preposition comes in the first position,
- the above mentioned verb-second phenomenon in main clauses.

With a slightly different view on phrase structure all three of these exceptions disappear. Let's for the moment just assume that the head always comes in the last position. Then it proves advantageous to choose a right-to-left order for processing sentences. There are several interesting consequences of this decision:

- there is no need for a separate PP-subnet. PP and NP are collapsed into one subnet and the preposition - if found at the 'end' of

the phrase - is simply viewed as a semantic case marker.

- adjuncts to the right of a phrase head have to be parsed separately. In our case: Postmodifiers like PPs, genitive NPs and relative clauses modifying NPs are not included in the NP-subnet. Since postmodifier attachment cannot be performed well using local information only, this pairs nicely with our strategy of handling the argument/modifier attachment on the caseframe level and thereby reducing ambiguity for the ATN.
- in main clauses (where the verb-second movement /Haider 1988/ applies) this movement has to be undone to have the complete verbal complex as the head of the sentence in the last position. This has another advantage: Although word order is different in main clauses and dependent clauses on the surface, after this retransformation the same subnet can be used for all different sentence types, and the same is true for the subnet for the verbal complex.

Adopting the grammar in the way just described leads to the desired situation where for every phrase type, the head comes last.

4. Caseframes and the Case Frame Matcher

Caseframes represent both a semantic and a syntactic representation of a phrase. The semantic content is given by a 'semantic' predicate and the functional dependencies and meanings of its arguments, and further restrictions by modifiers (if any).

The very idea of representing semantic dependencies in form of case frames goes back to the work of Fillmore /1968/, whereas ideas on the additional syntactic and functional structure we use can be traced back to Chomsky's /1981/ Theta-rules and Bresnan's /1982/ functional structures and in the Artificial Intelligence paradigm to the work of Creary and Pollard /1985/.

The caseframes in DB-DIALOG consist of several parts: The head predicate, a SELF-slot for proper referencing, so-called VALENCY slots containing functional dependencies (or deep cases), a MOD slot containing modifiers, a DETERMINER slot for NPs, and SYNTAX and CATEGORY slots containing various syntactic information.

- VALENCY slots in turn consist of:
- an identifier
 - a syntactic restriction (SYN)
 - a semantic restriction (SEM)
 - a filler (VALUE)

Caseframes are instantiated from the lexicon and information is added during the analysis of subphrases. To do so there is at least one so-called "meaning" attached to the lexical entry of each verb, noun and adjective. A meaning consists of a pointer to a caseframe plus eventual modifiers to be applied to the caseframe at the time of instantiation. The instantiation process creates new edges in the chart, representing these partially filled caseframes. The Case Frame Matcher (CFM) works on that chart, which is passed on to it by the ATN. This chart consists only of those caseframes relevant to the CFM to construct the new caseframe. Other parts, like the morphological chart or already constructed

caseframes outside the scope of the phrase actually considered remain invisible to it. One or more of the caseframes in the chart passed to the CFM are marked as prospective heads, and the output of the CFM is a new caseframe (or more than one in case of ambiguity) spanning the whole chart with several slots filled.

VALENCY slots may be filled if:

- syntactic restrictions are met,
- semantic restrictions are met, and
- other restrictions stemming from the category of the head (e.g. adjacency) are met.

The syntactic restrictions are met if the features of the SYN-slot and SYNTAX of the filler caseframe can be unified. The restrictions given are usually on category, case, proposition, etc. But they need not be given explicitly in all cases. One can make use of a number of structural cases like SUBJ (subject) and DOBJ (direct object). Transformations can apply to these cases under certain circumstances and e.g. transform DOBJ into SUBJ in case of passive. The realization of the structural case is evaluated at the time of slot filling, depending on the category of the head.

Only if a restriction is stated explicitly it is taken as it stands. But structural cases like e.g. SUBJ get different interpretations: for an S (sentence) a nominative NP with number agreement with the head is sought, for an AP SUBJ has to be the head of the governing NP, agreeing in case, gender and number, and for an NP SUBJ is realized as a genitive NP or a PP with the preposition 'von'.

This way great flexibility is gained and it is possible to reduce the lexicon and the meanings stored therein to the essentials. It is even possible to process nominalizations using the meaning of the corresponding verb.

The semantic restrictions to be met are given by a hierarchy of predicates. SEM and the predicate of the filler caseframe must be compatible to allow slot filling. Similar considerations apply to the construction of modifiers: syntactic and semantic compatibility must be given.

5. Interaction

Generally speaking, the topological regularities of phrases are handled by the ATN, whereas free word order constituents are being taken care of by the unification process. This unification process works on a local chart created by the ATN, comprising only those parts of the sentence relevant to it. Thus various island phenomena fall out from the conception of the parser.

Flow of control between the ATN and the other components is organized in a way proposed by Boguraev /1979/. The ATN starts processing a sentence in the usual way. After recognizing a phrase boundary by reaching a POP arc control is given either directly to the CFM or the unification process. The process evoked serves as a test for the POP arc, i.e. in case of failure the ATN has to backtrack.

In constituents (with strict word order) the CFM is invoked directly and tries to build up a caseframe (or more than one in case of ambiguity). The result is returned to the ATN which makes use of it during further processing.

In structures with free constituent order (clauses) the ATN acts solely as a collector

INTPRON	NOUN	VERB	PREP	ART	NOUN	PPP	NOUN	PREP	NOUN	PREP	ART	NOUN
RELPRON				RELPRON							RELPRON	

Welche .. Wien bezieht fuer die Produktion benoetigte Stoffe von Firmen aus dem Ausland ?

Fig.2: Morphological chart

of constituents. Constituent caseframes are merely stored in a local chart and attachment is postponed. The only constituent recognized topologically is the head which always comes in the last position. This chart of constituents is then given to the unification process when the POP arc is reached. In addition to relying heavily on the CFM, the unificator also has various strategies at its disposal in order to take into consideration restrictions of adjacency and category dependent of the category of the phrase processed. This way possible syntactic ambiguity is reduced and almost no backtracking is needed inside the ATN.

Generally, information passed to the CFM is collected while traversing the subnet: head caseframes are instantiated, arguments and modifiers are collected by pushing the appropriate subnets and morphological and/or syntactic clues trigger various informations on the caseframes.

As an example we mention the passive transformation: if evidence for passive is gathered while analyzing the verbal complex (for S) or a participle (for APs), this information is passed on to the CFM. The CFM then applies the passive transformation to the relevant slots of the head caseframe before the slot filling takes place. These transformations are one way to take general syntactic information away from the lexicon (the caseframes) to reduce redundancy /Hayes et al.1985/.

6. An Annotated Example

To demonstrate how the system works, we will conclude the paper by giving an annotated example of a parse. For the sake of clarity some of the details shall be simplified, but all of the essentials will be properly described.

We have chosen the following example sentence:

"Welche von unseren Abteilungen in Wien bezieht fuer die Produktion benoetigte Stoffe von Firmen aus dem Ausland?" ("Which of our Viennese departments gets materials necessary for production purposes from abroad?")

Please note that the free translation does not capture the grammatical subtleties involved in the original sentence; especially the adjective phrase "fuer die Produktion benoetigte Stoffe" includes a passivization that is usually not expressed this way in English.

The words are first processed morphologically and a chart is returned, rendering a canonical form for each of the words together with word class and syntactic information (e.g. case markers). At this level, some ambiguities arise, e.g. that of "welche" which might be an interrogative pronoun or a relative one, and "die" which may be an article or a relative pronoun.

There is a simple global control structure which works on this morphological chart. Its main task is to transfer control to ATN networks for phrase-like constituents and to the unificator for clause-like constituents. The control structure starts by transferring control to the PP/NP-ATN. The chart entry for "Ausland" is treated first (remember the right-to-left direction of processing). It is found to be a noun, and the next edge, DET, is processed. The third word, "aus", finishes the PP/NP. Control is transferred to the caseframe matcher (CFM). The caseframe for the head, "Ausland", becomes instantiated, and the features of the other components are unified with it, especially the feature of dative, which is derived from the determiner.

After completion of this caseframe, control is transferred back to the PP/NP net which processes "von Firmen" in a similar way. The CFM is called again, constructing another caseframe. According to our strategy, PP attachment will not be performed at this step, instead all the constituents will be collected first.

The PP/NP ATN gets its next chance. It treats the chart entry for "Stoffe" which makes a perfectly suitable head for a more complex constituent. We start to anticipate this when the next word, "benoetigte" ("necessary" - albeit not an adjective, but a PPP in German), is processed. In general, inflected PPPs trigger a PUSH AP, so does this one. (Uninflected PPPs form part of the verb complex). Next, a PUSH PP/NP is performed which will lead to a constituent embedded in the AP. But let's see this in detail. The PP is processed similar to the others before, the head "Produktion" becoming instantiated and the caseframe filled after the entry for "fuer" has been processed. This finishes the AP, since the verb, "bezieht", definitely cannot be part of an AP. As you may remember, APs trigger the unification component which in turn calls the CFM to handle the simpler tasks. Thus, the head of the AP, "benoetigte", becomes instantiated. The associated caseframe is presented below:

```
(BENOETIG
(SYN SUBJ)          (SEM ORGANIZATIONAL_UNIT)
(SYN DOBJ)          (SEM MATERIAL)
(SYN PPOBJ (FUER)) (SEM PURPOSE))
```

Before the caseframe will be filled, a passive transformation is applied, due to the fact that the example sentence contained the verb "benoetigen" in its PPP form. This transformation simply changes SUBJ to

		PP/NP3								PP/NP2		PP/NP1	
		AP											
		PP/NP											
INTPRON		NOUN	VERB	PREP	ART	NOUN	PPP	NOUN	PREP	NOUN	PREP	ART	NOUN
RELPRON					RELPRON							RELPRON	

Fig.3: After processing of PP/NP3

POBJ PASS and DOBJ to SUBJ. The consequences of this transformation will not turn out in this step, but in the next one, when the PP/NP with the head "Stoffe" will have been finished. But let's stick to the correct order. The caseframe of "benoetigen" has been instantiated and transformed, and it is about to be filled. Normally, the unificator will now come into its own, having to decide for proper attachments. In this case, there is only one constituent left at this level, namely "fuer die Produktion". Since no conflicting attachment alternatives arise, the CFM is called directly by the unificator. "Fuer die Produktion" fits nicely into the PURPOSE slot, so it is placed there. The AP now has been finished, and POP PP/NP is the next edge to be taken.

needs of the domain modelled. In our domain, this reading would not be appropriate, so we did not provide a caseframe for it, thus excluding a theoretical ambiguity where in the practical application there is none. As the slot filling fails, the AP-ATN will backtrack. We get an AP consisting of just one single word ("gefaerbte") filling a slot in "Stoffe", making up for one PP/NP and another PP/NP, namely "fuer die Produktion". These two PP/NPs will be collected at this stage of processing and only attached when all of the sentence will have been parsed.

We will stop our digression here and come back to the original example. Remember, the AP has just been finished and the PP/NP with the head "Stoffe" is Popped. This means a transfer of control to the CFM (in PP/NPs the CFM is called directly, whereas in an AP or S the unificator is called first in order to find correct attachments. Afterwards, the unificator in turn calls the CFM to realize the selected attachments). The AP is integrated into the PP/NP caseframe as a modifier predicate in the MOD slot. The SUBJ slot of the subordinated caseframe (the one of "benoetigen") is still unfilled. For syntactic reasons, its filler must be the head of the superordinated PP/NP "Stoffe". The semantic restriction of the SUBJ slot is MATERIAL which is compatible with the noun "Stoffe", so the slot may be filled (note that SUBJ is the transformed syntactic restriction which had been DOBJ before the passive transformation had taken place). Thus, a third constituent has been added to the pool of collected constituents.

Self	3	
Type	NP	
Syntax	Nom/Acc Plural	
Head	Stoff	
Mod	Self	...
	Head	benoetig
	Pobj-Pass Subj	Organizational_Unit
	Subj	Doj Material
	PPobj (fuer)	Purpose
	Self	...
	Head	Produktion

Fig.4: Caseframe for PP/NP3

Let us take a little digression. Suppose the PP/NP "fuer die Produktion" would not have fit into a slot of the PPP. If we had taken "gefaerbte" ("dyed") instead of "benoetigte" this would do. In this case we would not get the reading "materials dyed for production purposes" but instead two components, dyed materials" and "for production purposes". The sophisticated reader could argue that the first reading might also be correct. The argument here is that the caseframes in our systems are constructed in a way to fit the

The global control structure continues by processing the next entry, the representation of the word "bezieht", which is a finite verb and has to be at the second position according to German grammar. It is set aside for later processing and a special state is entered, knowing that exactly one constituent has been left over. The PP/NP "in Wien" is processed, and a corresponding caseframe is created.

Similarly, a caseframe for "welche von unseren Abteilungen" is created and "in Wien" is attached to it when the unificator applies its knowledge that there cannot be more than one constituent in this position. This way, possible ambiguities e.g. trying to fill "in Wien" into a slot at sentence level, are avoided.

PP/NP4		PP/NP3								PP/NP2		PP/NP1	
PP/NP		PP/NP		AP									
		PP/NP											
INTPRON		NOUN	VERB	PREP	ART	NOUN	PPP	NOUN	PREP	NOUN	PREP	ART	NOUN
RELPRON					RELPRON							RELPRON	

Fig.5: Before unification at sentence level

By this time we have finished our way from right to left through the morphological chart and have collected many components (PP/NPs and the predicate) at the sentence level. The global control structure passes control to the unificator which has to find correct attachment and to perform the slot filling at the sentence level. Caseframe instantiation takes place, building a frame for the verb "beziehen".

(BEZIEH
 (SYN SUBJ) (SEM ORGANIZAT.UNIT)
 (SYN DOBJ) (SEM MATERIAL)
 (SYN PPOBJ (FUER)) (SEM PURPOSE)
 (SYN PPOBJ (VON BEI)) (SEM ORGANIZAT.UNIT))

Next, all possible attachments are sought. Two conditions have to hold for them: adjacency and semantic compatibility. PP/NP4 e.g. cannot be attached to any other constituent, because it is adjacent only to the main verb. Therefore, this constituent has to fill a slot in "beziehen". For the remaining PP/NPs there exist different possibilities. Let us denote subordination by the hyphen character. From the adjacency point of view, the possibilities are:

- 1) PP/NP3 , PP/NP2, PP/NP1
 (three constituents at sentence level)
- 2) PP/NP3 - PP/NP2, PP/NP1
- 3) PP/NP3, PP/NP2 - PP/NP1
- 4) PP/NP3 - (PP/NP2 - PP/NP1)

1 and 2 are excluded, because there is no slot in the "beziehen" caseframe which matches the syntax of PP/NP1 (preposition "aus"), nor would there be semantic compatibility. 3 is the reading we prefer. As for 4, its acceptability depends on whether we allow a slot in the caseframe for "Stoffe" which could hold an ORGANIZATIONAL UNIT. If we do, we will get an ambiguity. In that case, the system will offer both solutions, using a heuristic which of the solutions to present first. The heuristic implemented prefers flat syntactic structures.

As for the preferred reading, the CFM realizes it by filling PP/NP3 into the DOBJ slot and (PP/NP2 - PP/NP1) into the PPOBJ slot of the caseframe for "beziehen". PP/NP4 has already been filled in the SUBJ slot, so the parse of the sentence has been completed.

7. Conclusion

In this paper we have described a parser which is able to deal with a wide variety of German sentences in an efficient and robust way. This is made possible by two special properties of the parser:

First, syntactic analysis uses two different strategies, namely an ATN and unification to deal with fixed word order at the constituent level and free constituent order at the clause level respectively. Such an approach is well suited for German, but also for other SOV languages like Dutch, Swedish, Danish and Japanese.

Second, the interaction between caseframe instantiation and a syntactic parser leads to the possibility of

- clear and precise formulation of syntactic and semantic rules and regularities,
- reduction of entries in the lexicon,
- flexibility and better maintainability.

Thus problems posed by various aspects of the structure of the German language have led to general solutions which may be applied to other languages as well.

Acknowledgement

Work on DB-DIALOG is done jointly with Software Management GmbH, A-1140 Vienna, Austria. It has also been sponsored by the Austrian Government within the "Schwerpunkt S7 (Artificial Intelligence) des Mikroelektronikfoeroderungsprogrammas der Oesterreichischen Bundesregierung".

References:

- Bates, M. (1978) 'The Theory and Practice of Augmented Transition Network Grammars.' In: Bolc L.(ed.): Natural Language Communication with Computers. Springer, Berlin.
- Boguraev, B.K. (1979) 'Automatic Resolution of Linguistic Ambiguities.' University of Cambridge, Comp. Laboratory, TR-11, Cambridge.
- Bresnan, J.(ed.) (1982) 'The Mental Representation of Grammatical Relations.' MIT Press, Cambridge MA.
- Buchberger E., Heinz W., Matiassek J., Trost H.(1987) 'The German Language Database Interface DIALOG.' Proc.8th International Congress Data Processing in Europe, Vienna.
- Chomsky, N.(1981) 'Lectures on Government and Binding.' Foris, Dordrecht.
- Creary L., Pollard C.(1985) 'A Computational Semantics for Natural Language.' In Proc.23rd Meeting of the ACL, Chicago.
- Fillmore, C.J.(1968) 'The Case for Case.' In: Bach E., Harms R.T.(eds.): Universals in Linguistic Theory, Holt, Rinehart and Winston Inc., New York.
- Grishman R., Hirschman L., Nhan N.T.(1986) 'Discovery Procedures for Sublanguage Selectional Patterns: Initial Experiments.' Computational Linguistics,3(12).
- Haider, H.(1984) 'Topic, Focus and Verb-second.' Groninger Arbeiten zur Germanistischen Linguistik 25, pp72-120.
- Haider, H.(1988) 'Verb-second in German.' In: Haider H., Prinzhorn Th.(eds.): Verb-second, Foris, Dordrecht.
- Hauenschild, Ch.(1988) 'GPSC and German Word Order.' In: Reyle U., Rohrer Ch.(eds.): Natural Language Parsing and Linguistic Theory, Reidel, Dordrecht.
- Hayes Ph., Andersen P., Safir S.(1985) 'Semantic Caseframe Parsing and Syntactic Generality.' In: Proc.23rd Annual Meeting of the ACL, Chicago.
- Hoehle, T.N.(1982) 'Explication fuer normale Betonung und normale Wortstellung.' In: Abraham W.(ed.): Satzglieder im Deutschen, Narr, Tuebingen.
- Lytinen, S.(1986) 'Dynamically Combining Syntax and Semantics in Natural Language Processing.' Proc.AAAI-86, Philadelphia PA.
- Proudian, D., Pollard, C.(1985) 'Parsing Head-Driven Phrase Structure Grammar.' Proc.23rd Annual Meeting of the ACL, Chicago.
- Trost, H., Dorffner, G.(1987) 'A System for Morphological Analysis and Synthesis of German Texts.' In: Hainline D.(ed.): New Developments in Computer-Assisted Language Learning, Croom Helm, London.
- Uszkoreit, H.(1986) 'Constraints on Order' Stanford Univ., CSLI-86-46, Stanford, CA.