Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

Research
Report
RR-93-31

Abductive Speech Act Recognition, Corporate Agents and the COSMA System

Elizabeth A. Hinkelman, Stephen P. Spackman

December 1993

Deutsches Forschungszentrum für Künstliche Intelligenz
GmbH

Postfach 20 80
67608 Kaiserslautern, FRG
Tel.: (+49 631) 205-3211/13
Fax: (+49 631) 205-3210

Stuhlsatzenhausweg 3
66123 Saarbrücken, 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 and Saarbrücken is a non-profit organization which was founded in 1988. The shareholder companies are Atlas Elektronik, Daimler-Benz, Fraunhofer Gesellschaft, GMD, IBM, Insiders, Mannesmann-Kienzle, SEMA Group, and Siemens. 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
❑ Computer Linguistics
❑ Programming Systems
❑ Deduction and Multiagent Systems
❑ Document Analysis and Office Automation.

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.

Friedrich J. Wendl
Director

Abductive Speech Act Recognition, Corporate Agents and the COSMA System

Elizabeth A. Hinkelman, Stephen P. Spackman

DFKI-RR-93-31

Also appeared in
Abduction, Beliefs and Context: Studies in Computational Pragmatics
University College Press, 1994
W. J. Black, ed.

This work has been supported by a grant from The Federal Ministry for Research and Technology (FKZ ITW-9002 0).

Abductive Speech Act Recognition, Corporate Agents and the COSMA System

Elizabeth A. Hinkelman and Stephen P. Spackman

hinkelman@dfki.uni-sb.de, stephen@acm.org

DFKI, Stuhlsatzenhausweg 3, D-W-6600 Saarbrücken, Germany

Abstract

This chapter presents an overview of the DISCO project's solutions to several problems in natural language pragmatics. Its central focus is on relating utterances to intentions through speech act recognition. Subproblems include the incorporation of linguistic cues into the speech act recognition process, precise and efficient multiagent belief attribution models (*Corporate Agents*), and speech act representation and processing using Corporate Agents. These ideas are being tested within the COSMA appointment scheduling system, one application of the DISCO natural language interface. Abductive speech act processing in this environment is not far from realizing its potential for fully bidirectional implementation.

Contents

In linking text to task, it is crucial to capture not only propositional content of linguistic expressions but also the attitude or intention behind them. Speech acts, or utterances construed as actions, serve this purpose. Speech act recognition attempts to reverse-engineer an utterance to determine the speaker's goals in producing it, and as such is an abductive process. In the early portion of this paper, we will present our view of abduction and how speech acts are recognized in the DISCO system based on linguistic conventions.

The primary goal of the DISCO[1] project is processing of multiagent (that is, three or more agent) natural language dialogue. To address problems in the representation of multi-agent speec acts, we next introduce the notion of *Corporate Agents.* This allows us to build a model of reliable, N-way speech acts.

Multiagent capabilities make it an appropriate front end for autonomous cooperative agent and robotic systems, operating in groups with human participation. COSMA[2] is such a system, maintaining a database of appointments and negotiating appointment times with other agents on behalf of its user. Speech acts in this environment pertain not only to information, but also to commitment, and to the structure of discourse. This is the context in which we demonstrate and test our ideas. Viewing speech act recognition as abduction, as we shall see, leads to emphasis on the tactical and bidirectional aspects of our mechanisms.

1 Abduction

The notion of *abduction* currently popular in artificial intelligence is a form of unsound inference. It relies on the intuition that the implication $P \Rightarrow Q$, like English 'if', captures more than just the logically equivalent disjunction $\neg P \vee Q$.

One common use of implication is to express *causal* relationships. Here, when P is an antecedent of Q, observation of Q suggests the hypothesis P.

$$\frac{P \Rightarrow Q \quad Q}{P}$$

[1] DIalogue Systems for COoperating agents

[2] COoperative Schedule Management Agent

Although it is technically unsound, this inference is completely rational in those domains where things don't "just happen"; that is, we need to be able to assume that each observation has a cause that we can name. We refer to these as *framed domains,* because these are the same conditions under which the frame problem has a solution.

Given this extended, partially reversible interpretation, implication becomes an over-interpreted notation, retaining its conventional, truth-functional meaning but also expressing *tactical* information: given Q, the hypothesis P should be tried before other explanations.

In the case that an observation has more than one possible cause, or whenever it is necessary to manipulate the logical form, a more elaborate notation can prove useful, one in which preferences are made explicit as weights. For the evaluation of entire explanations, some form of numerical calculus must be defined on the weights. This calculus is itself a partial declarative description of the tactical interpreter. The weights do not necessarily correspond to probability measures, of course, since for some applications, the most probable explanation may not be yield the best tactical decision.

$$\begin{array}{l} P_1 \Rightarrow Q, w_1 \\ P_2 \Rightarrow Q, w_2 \\ P_3 \Rightarrow Q, w_3 \\ \ldots \\ \underline{P_N \Rightarrow Q, w_N \;\; Q} \\ P_J | w_J = \max_i w_i \end{array}$$

Although the term was not in widespread use at the time, early work on plan recognition used abductive methods. An agent was taken as an inference system with a planning algorithm. The planning algorithm constructed chains of actions that would, if executed, result in the achievement of the agent's goal. Plan recognition then consisted in reversing this process, taking observed actions as evidence for intentions [2]. Even the fine-grained linguistic structure of an utterance can be regarded as the result of a planning process, and hence the starting point for abductive recognition of speaker intentions [9, 8, 11].

2 Speech Act Recognition

Linguistic structures in the DISCO system are represented in the unification-based formalism of Head-driven Phrase Structure Grammar [27]. In this framework, the unifier can be used as an inference engine if the inference rules are coded properly. To see how this works, assume that P and Q are coded as complex feature values. A feature value may be a simple atomic identifier occurring paired with a feature name, or it may be a composed structure which can also stand as an independent top level structure. Feature-value pairs are traditionally written within a conjunction (square brackets), even when only one pair is present. For the moment, the only other composition operation needed is disjunction (curly brackets) of feature values. A simple implication $P \Rightarrow Q$ can be coded as:

$$(1)\quad \left\{ \begin{bmatrix} \mathrm{F}_0 & P \\ \mathrm{F}_1 & Q \end{bmatrix}, \begin{bmatrix} \mathrm{F}_0 & -P \\ \mathrm{F}_1 & \{Q, -Q\} \end{bmatrix} \right\}$$

This structure is a disjunction of two terms. The first term is a conjunction: P and Q may occur simultaneously as values of their respective features. The second term conjoins the complement of P to occur with Q, as well as with the complement of Q. There is no term that would allow P to occur with the complement of Q, and so the sense of the conditional is enforced by the unification procedure itself. Note also that negation is not itself part of HPSG, but rather a metanotation to avoid enumerating the disjuncts necessary to describe the complement of a complex feature structure.

Likewise, $P \Leftarrow Q$ is:

$$(2)\quad \left\{ \begin{bmatrix} \mathrm{F}_0 & P \\ \mathrm{F}_1 & Q \end{bmatrix}, \begin{bmatrix} \mathrm{F}_0 & \{P, -P\} \\ \mathrm{F}_1 & -Q \end{bmatrix} \right\}$$

The rule $P \Leftrightarrow Q$ is thus:

$$(3)\quad \left\{ \begin{bmatrix} \mathrm{F}_0 & P \\ \mathrm{F}_1 & Q \end{bmatrix}, \begin{bmatrix} \mathrm{F}_0 & -P \\ \mathrm{F}_1 & -Q \end{bmatrix} \right\}$$

Because the unification mechanism is symmetrical, these rules can be used deductively as well as abductively depending on whether P or Q is input. [3]

[3] Each such rule acts as an independent constraint. A set of rules describes the same set of possible worlds as its FOPC version, with the advantage that the standard implementation of unification maintains all alternatives consistent with the input.

In the remainder of this section, Q will correspond to any conventional pattern of linguistic features, which a speaker may produce to communicate some intention. P is a set of possible intentions. The set of communicative intentions we use is derived from speech act theory, and we will develop a formal characterization of them in a subsequent section. The set used in our implementation and examples includes informing and asking (informational), requests and commands (directives), acceptance and rejection of plans, and greetings and closings.

2.1 Linguistic Convention in Speech Act Recognition

The role of language conventions in speech act recognition can be shown through cross-linguistic, dialectal, and idiolectal arguments [9]. One relatively simple linguistic convention is that a declarative sentence is used primarily for informing, but secondarily for many other purposes. The figure below is the declarative mood rule as implemented in the DISCO system. The "P" side of the rule, the speech act, lies on the path SEMINF|SEM|CONTENT, while the "Q" side, the linguistic features, lie under the path cat.

A German declarative can be recognized in that it is a verb phrase (V+ N-) that is not verb-initial, contains an empty list of wh-terms, but has a topic before the verb. Likewise it is not an idiom (FIXED top). It may be either indicative or subjunctive in mood. In DISCO's HPSG syntax of German, this information is consolidated under a syntactic feature named SMOOD which also includes agreement information.

(4)
$$\left[\begin{array}{ll} \texttt{SEMINF} & \left[\texttt{SEM}\ \left[\texttt{CONTENT}\ \left[\begin{array}{ll}\texttt{PRED} & \mathit{17}\left\{\begin{array}{c}\mathit{INFORM}\\ \top\end{array}\right\}\\ \texttt{THEME} & \mathit{17}\left\{\begin{array}{c}\boxed{1}\\ \top\end{array}\right\}\end{array}\right]\right]\right]\\ \texttt{CAT} & \left[\begin{array}{ll}\texttt{SMOOD} & \left[\begin{array}{ll}\texttt{FIXED} & \top\\ \texttt{V-INIT} & -\\ \texttt{TOPIC} & +\\ \texttt{WH-QUE} & *\mathit{END}\\ \texttt{HEAD} & \left[\begin{array}{ll}\texttt{INF} & *\mathit{NIL}\\ \texttt{FIN} & \left[\texttt{MOOD}\ \{\mathit{IND,\ CONJ}\}\right]\\ \texttt{V} & +\\ \texttt{N} & -\end{array}\right]\end{array}\right]\\ \texttt{SEM} & \mathit{17}\left\{\begin{array}{c}\boxed{1}[\ldots]\\ \top\end{array}\right\}\end{array}\right]\end{array}\right]$$

The speech act is a predicate with roles, like the semantic structure built during parsing. In this particular example, it is spread throughout a *distributed disjunction*, a notational convenience which eliminates the need to duplicate large blocks of information. Within a top-level structure, all disjunctions having the same reference number are understood as corresponding lists. Here, disjunction 17 contains speech act predicates in its first piece, their themes in the second, and propositional content in the third.

Consider the first interpretation, the inform predicate visible in the first line of the diagram. The declarative rule assigns it a theme, which in this case happens to be a coreference, or label of a shared structure. The structure in question is the semantic representation that would built under CAT|SEM, for an actual input sentence. A complete inform interpretation would also pick up an agent and patient from the discourse context, as well as this propositional content. Disjunction 17 is the structure that will be translated into the logical form language $\mathcal{NLL}$, and thence to the knowledge representation where it will pick up the kind of action definition discussed in Section Three. The speech act is thus effectively an implicit operator dominating the sentence semantics, constituting a new, nonsyntactic performative hypothesis.

Now consider the second interpretation within the distributed disjunction. It is not the preferred one, and in fact contains very little information at all. This notation is pronounced "top", and in this case is simply the top of the speech act abstraction hierarchy. Its corresponding theme is a semantic "top", and allowance is made for some other rule to relate this to sentence semantics by including a "top" there too. With the incorporation of preferences into DISCO's core formalism in 1993, it will be possible to yield the best speech act explanation first.

The observant reader will also notice at this point that we have shown only one HPSG term, of the two that we claimed were necessary to implement an implication. Since all rules apply simultaneously, the missing term is what allows structures from the complement of the declarative feature set to occur. The current implementation uses a direct procedural construction instead, but this will be changed in the near future due to the needs of generation.

2.2 A Related Example

The interrogative rule, below, is similar in spirit to the declarative rule. The principal difference is that where a declarative may express any of our eight speech act types, an interrogative may only express two.

(5)

$$\left[\begin{array}{ll}
\texttt{SEMINF} & \left[\texttt{SEM}\ \left[\texttt{CONTENT}\ \left[\begin{array}{ll}
\texttt{THEME} & \mathit{21}\left\{\begin{array}{l}\boxed{1}\\ \top\end{array}\right\}\\
\texttt{PRED} & \mathit{21}\left\{\begin{array}{l}Y-N-QUESTION\\ REQUEST\end{array}\right\}
\end{array}\right]\right]\right]\\
\texttt{CAT} & \left[\begin{array}{ll}
\texttt{SMOOD} & \left[\begin{array}{ll}
\texttt{FIXED} & \top\\
\texttt{V-INIT} & +\\
\texttt{TOPIC} & \\
\texttt{WH-QUE} & *END\\
\texttt{HEAD} & \left[\begin{array}{ll}
\texttt{INF} & *NIL\\
\texttt{FIN} & \left[\texttt{MOOD}\ \{\mathit{IND,\ CONJ}\}\right]\\
\texttt{V} & +\\
\texttt{N} &
\end{array}\right]
\end{array}\right]\\
\texttt{SEM} & \mathit{21}\left\{\begin{array}{l}\boxed{1}[\ldots]\\ \top\end{array}\right\}
\end{array}\right]
\end{array}\right]$$

An utterance like “Is the Pope Catholic?”, used to express agreement, would be handled as a fixed expression.

2.3 Fixed Expressions

The present version of DISCO handles a limited class of idioms: those that can be represented as a single lexical entry because they span a top-level input string. In true head-driven spirit, we store the appropriate speech act entry there. When the parser recognizes that it's a complete structure, it promotes the speech act to the root level of the syntactic structure. Later, the speech act recognition rule for fixed expressions identifies the source of the structure and passes it on. Fixed phrases are denoted by having no syntactic category.

$$(6)\quad \left[\texttt{CAT}\left[\texttt{SMOOD}\left[\texttt{HEAD}\begin{bmatrix}\texttt{V} *NIL\\ \texttt{N} *NIL\end{bmatrix}\right]\right]\right]$$

A typical fixed expression in German is a standard greeting which can be glossed as “very honored Mr/Ms N.”. In order to show you the entire lexical entry, we have split it into two parts. The first shows the general structure, with very empty syntax and the greeting speech act interpretation under the path CAT|SEMINF. Note the coreference under MLIST; this is the morphology, which appears in the subsequent figure.

(7)

$$\left[\begin{array}{ll}
\mathtt{CAT} & \left[\begin{array}{ll}
\mathtt{SMOOD} & \left[\begin{array}{ll}
\mathtt{FIXED} & \top \\
\mathtt{HEAD} & \boxed{1}\left[\begin{array}{l}\mathtt{V}\ *NIL \\ \mathtt{N}\ *NIL\end{array}\right] \\
\mathtt{TOPIC} & *NIL \\
\mathtt{V\text{-}INIT} & *NIL \\
\mathtt{WH\text{-}QUE} & *END
\end{array}\right] \\
\mathtt{SEM} & \top \\
\mathtt{SYN} & \left[\begin{array}{ll}
\mathtt{NON\text{-}LOC} & \left[\begin{array}{ll}
\mathtt{QUE} & *END \\
\mathtt{SLADJ} & \left[\begin{array}{ll}\mathtt{COND} & \left[\mathtt{SUB\text{-}WFFS}\ *END\right] \\ \mathtt{TYPE} & RP\end{array}\right] \\
\mathtt{SLASH} & *END
\end{array}\right] \\
\mathtt{LOCAL} & \left[\begin{array}{ll}
\mathtt{HEAD} & \left[\begin{array}{ll}\mathtt{INFL} & \left[\mathtt{GRADE}\ ST\right] \\ \mathtt{MAJ} & \boxed{1} \\ \mathtt{MIN} & \left[\mathtt{FCOMPL}\ +\right]\end{array}\right] \\
\mathtt{LPE} & - \\
\mathtt{PHRASAL} & \{+,-\} \\
\mathtt{SUBCAT} & *END \\
\mathtt{TOPIC} & *NIL \\
\mathtt{V\text{-}INIT} & *NIL
\end{array}\right]
\end{array}\right] \\
\mathtt{M\text{-}LIST} & \boxed{3} \\
\mathtt{SEMINF} & \left[\mathtt{SEM}\left[\mathtt{CONTENT}\left[\begin{array}{ll}\mathtt{PATIENT} & TICK \\ \mathtt{PRED} & BEGRUESSUNG\end{array}\right]\right]\right]
\end{array}\right] \\
\mathtt{RULE\text{-}NAME} & FIXED - EXPRESSION \\
\mathtt{SEMINF} & \left[\mathtt{SEM}\left[\mathtt{CONTENT}\ \top\right]\right]
\end{array}\right]$$

Here is the morphology:

$$
(8)\quad
\begin{bmatrix}
\texttt{HEAD}\begin{bmatrix}\texttt{CAT}\ \mathit{PARTIKEL}\end{bmatrix} \\
\texttt{STEM}\ \langle\, \mathit{SEHR}\, \rangle
\end{bmatrix}
$$

$$
\begin{bmatrix}
\texttt{HEAD}\begin{bmatrix}
\texttt{CAT} & \mathit{ADJ} \\
\texttt{COMPARATION} & \mathit{POS} \\
\texttt{INFL} & \mathit{ADJ-ER} \\
\texttt{VALUE} & \top
\end{bmatrix} \\
\texttt{STEM}\ \langle\, \mathit{GEEHRT}\, \rangle
\end{bmatrix}
$$

$$
\begin{bmatrix}
\texttt{HEAD}\begin{bmatrix}
\texttt{CAT} & \mathit{NOUN} \\
\texttt{CASE} & \begin{bmatrix}\texttt{GOV} & - \\ \texttt{OBL} & -\end{bmatrix} \\
\texttt{GENDER} & \begin{bmatrix}\texttt{FEM} & - \\ \texttt{MAS} & +\end{bmatrix} \\
\texttt{NUM} & \mathit{SG}
\end{bmatrix} \\
\texttt{STEM}\ \langle\, \mathit{HERR}\, \rangle
\end{bmatrix}
$$

$$
\begin{bmatrix}
\texttt{HEAD}\begin{bmatrix}
\texttt{CAT} & \mathit{P-NOUN} \\
\texttt{CASE} & \left\{\begin{matrix}\begin{bmatrix}\texttt{GOV} & +\end{bmatrix} \\ \begin{bmatrix}\texttt{OBL} & -\end{bmatrix}\end{matrix}\right\} \\
\texttt{GENDER} & \begin{bmatrix}\texttt{FEM} & - \\ \texttt{MAS} & +\end{bmatrix} \\
\texttt{NUM} & \mathit{SG}
\end{bmatrix} \\
\texttt{STEM}\ \langle\, \mathit{TICK}\, \rangle
\end{bmatrix}
$$

When the above structures are parsed, the speech act interpretation percolates up.

(9)

$$\begin{bmatrix}
\texttt{CAT} & \begin{bmatrix}
\texttt{SMOOD} & \begin{bmatrix} \texttt{FIXED} & \top \\ \texttt{HEAD} & \boxed{1}\begin{bmatrix}\texttt{V} & *\mathit{NIL} \\ \texttt{N} & *\mathit{NIL}\end{bmatrix} \\ \texttt{TOPIC} & *\mathit{NIL} \\ \texttt{V-INIT} & *\mathit{NIL} \\ \texttt{WH-QUE} & *\mathit{END} \end{bmatrix} \\
\texttt{SEM} & \top \\
\texttt{SYN} & \begin{bmatrix} \texttt{NON-LOC} & \begin{bmatrix} \texttt{QUE} & *\mathit{END} \\ \texttt{SLADJ} & \begin{bmatrix}\texttt{COND} & \begin{bmatrix}\texttt{SUB-WFFS} & *\mathit{END}\end{bmatrix} \\ \texttt{TYPE} & \mathit{RP}\end{bmatrix} \\ \texttt{SLASH} & *\mathit{END} \end{bmatrix} \\
\texttt{LOCAL} & \begin{bmatrix} \texttt{HEAD} & \begin{bmatrix} \texttt{INFL} & \begin{bmatrix}\texttt{GRADE} & \mathit{ST}\end{bmatrix} \\ \texttt{MAJ} & \boxed{1} \\ \texttt{MIN} & \begin{bmatrix}\texttt{FCOMPL} & +\end{bmatrix}\end{bmatrix} \\ \texttt{LPE} & - \\ \texttt{PHRASAL} & \{+, -\} \\ \texttt{SUBCAT} & *\mathit{END} \\ \texttt{TOPIC} & *\mathit{NIL} \\ \texttt{V-INIT} & *\mathit{NIL} \end{bmatrix}\end{bmatrix} \\
\texttt{M-LIST} & \boxed{3} \\
\texttt{SEMINF} & \begin{bmatrix}\texttt{SEM} & \begin{bmatrix}\texttt{CONTENT} & \begin{bmatrix}\texttt{PATIENT} & \mathit{TICK} \\ \texttt{PRED} & \mathit{BEGRUESSUNG}\end{bmatrix}\end{bmatrix}\end{bmatrix}
\end{bmatrix} \\
\texttt{RULE-NAME} & \mathit{FIXED} - \mathit{EXPRESSION} \\
\texttt{SEMINF} & \begin{bmatrix}\texttt{SEM} & \begin{bmatrix}\texttt{CONTENT} & \begin{bmatrix}\texttt{PATIENT} & \mathit{TICK} \\ \texttt{PRED} & \mathit{BEGRUESSUNG}\end{bmatrix}\end{bmatrix}\end{bmatrix}
\end{bmatrix}$$

2.4 Convention Example

The fixed expression is a nearly trivial case of percolating speech act information through the parse tree. Full integration with the fundamentally lexicalist HPSG theory calls for pragmatic information in the lexical entries of particles, adverbs, modal verbs, and other items that bear on speech act interpretation. This will require some experimentation with percolation mechanisms, and a careful study of the interactions of embedded clause speech acts. They must not be allowed, for instance, to propagate through quotation, but some occasions do propagate over clause boundaries. At present, such clearly head-driven structures as modal auxiliary requests are handled by a postprocessing phase.

(10)
$$
\begin{bmatrix}
\mathtt{SEMINF} & \begin{bmatrix} \mathtt{SEM} & \begin{bmatrix} \mathtt{CONTENT} & \begin{bmatrix} \mathtt{THEME} & \mathit{37}\begin{Bmatrix} \top \\ \boxed{2} \end{Bmatrix} \\ \mathtt{PRED} & \mathit{37}\begin{Bmatrix} Y-N-QUESTION \\ REQUEST \end{Bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix} \\
\mathtt{CAT} & \begin{bmatrix}
\mathtt{SMOOD} & \begin{bmatrix}
\mathtt{FIXED} & \top \\
\mathtt{V\text{-}INIT} & + \\
\mathtt{TOPIC} & - \\
\mathtt{WH\text{-}QUE} & *END \\
\mathtt{HEAD} & \begin{bmatrix}
\mathtt{INF} & *NIL \\
\mathtt{FIN} & \begin{bmatrix}
\mathtt{AGR} & \begin{bmatrix} \mathtt{PERS} & \mathit{38}\begin{Bmatrix} 2 \\ 1 \end{Bmatrix} \\ \mathtt{NUM} & \mathit{38}\begin{Bmatrix} \top \\ PL \end{Bmatrix} \end{bmatrix} \\
\mathtt{TENSE} & PRES \\
\mathtt{MOOD} & \{IND, CONJ\}
\end{bmatrix} \\
\mathtt{V} & + \\
\mathtt{N} & -
\end{bmatrix}
\end{bmatrix} \\
\mathtt{SEM} & \mathit{37}\begin{Bmatrix}
\begin{bmatrix} \mathtt{CONTENT} & \begin{bmatrix} \mathtt{PRED} & KOENN \end{bmatrix} \end{bmatrix} \\
\begin{bmatrix} \mathtt{CONTENT} & \begin{bmatrix} \mathtt{PROPOS} & \boxed{2}[\ldots] \\ \mathtt{PRED} & KOENN \end{bmatrix} \end{bmatrix}
\end{Bmatrix}
\end{bmatrix}
\end{bmatrix}
$$

2.5 The Rest of Recognition

The work described up to this point was implemented in the DISCO system by Oliver Silvester Scherf. With the indicated refinements, it is a fully reversible abductive approach to the problem of relating speaker intentions to observable utterance structure. However, it is only the first tier of the intention recognition process. As discussed in [10, 2], subsequent abductive stages of processing are dependent upon agent beliefs, goals, and plans. Although longterm goals include a uniform base representation for the entire system, the current system is more conservative and uses a PROLOG-style inference engine behind the general knowledge representation.

3 Corporate Agents and Speech Acts

The previous section showed how abduction based on linguistic conventions can be used to generate a set of speech act interpretations. Now, we wish to use local context to filter out implausible interpretations after the manner of [10]. However, discourse among several agents is computationally complex. This consideration has led us to advances in knowledge representation for speech act processing.

The first tool that we need is a model for belief and belief attribution, since speech acts operate over these structures. *Mutual belief* is often formulated as

$$\mathbf{MB}_{\mathbf{x,y}}\mathrm{p} \triangleq \mathbf{B}_{\mathbf{x}}\mathrm{p} \wedge \mathbf{B}_{\mathbf{y}}\mathrm{p} \wedge \mathbf{B}_{\mathbf{x}}\mathbf{B}_{\mathbf{y}}\mathrm{p} \wedge \mathbf{B}_{\mathbf{y}}\mathbf{B}_{\mathbf{x}}\mathrm{p} \wedge \mathbf{B}_{\mathbf{x}}\mathbf{B}_{\mathbf{y}}\mathbf{B}_{\mathbf{x}}\mathrm{p} \wedge \ldots \qquad (1)$$

(where $\mathbf{B}_{\mathbf{x}}\mathrm{p}$ says that agent x believes proposition p). It captures the notion of full, bilateral, symmetric, introspective belief of a proposition, which is one possible outcome should x say p to y.[4]

[4] Current practice favours the use of a one-sided version of mutual belief,

$$\mathbf{BMB}_{\mathrm{x,y}}\mathrm{p} \triangleq \mathbf{B}_{\mathrm{x}}\mathrm{p} \wedge \mathbf{B}_{\mathrm{x}}\mathbf{B}_{\mathrm{y}}\mathrm{p} \wedge \mathbf{B}_{\mathrm{x}}\mathbf{B}_{\mathrm{y}}\mathbf{B}_{\mathrm{x}}\mathrm{p} \wedge \ldots$$

This may be due to a confusion between personal and omniscient perspectives on the world, for in a situated logic representing only the agent's actual beliefs, the two formulations collapse under the normal belief axioms. Use of the omniscient perspective should, we believe, be restricted to *post hoc* formal analysis of joint behaviours.

The focus of much previous work has been on the logical mechanics of creating [5, 26] and maintaining [34] such states.

There are several difficulties stemming from this approach. (1) It is implausible as a model of human cognition. (2) The superficially infinite conjunction in equation 1 may require special formal and implementational attention. (3) Even finite initial sequences of the conjunctive expansion blow up exponentially on the introduction of additional agents. None of these troubles impacts a purely *normative* system, but implementations must degrade more gracefully under pressure.

We [32] are currently exploring the alternative of *Corporate Agents*. An *agent* is a formal object, standing for any entity that makes changes in the world. A "rational" agent not only acts, but has preferences, beliefs, and a productive inference mechanism. The *Ideal Rational Agent* would have a sound, complete inference mechanism with inferential closure, but nontrivial implementable agents will necessarily lack these properties.

Within a sound and complete logic, a Corporate Agent would be merely a notational convenience, standing for a set of actual agents.[5] The idealised formal semantics of corporate agents is thus a simple powerset construction. Treating actual agents as singleton sets, the subset relation defines the partial order 'subagent'. There are then two basic modes in which an attitude (such as a belief or a goal) can be ascribed to an agent. It can be ascribed universally, holding for every subagent of the agent. This is written $\hat{\Box}_{\mathrm{a}}\mathrm{p}$, which we gloss as "p is held throughout a". An attitude can also be ascribed existentially, requiring that it hold only for *some* subagent. This is written $\hat{\Diamond}_{\mathrm{a}}\mathrm{p}$, and glossed "p is held within a". This formalisation immediately allows us to conclude that:

$$\mathrm{a} \sqsupseteq \mathrm{b} \vdash \hat{\Box}_{\mathrm{a}}\mathrm{p} \Rightarrow \hat{\Box}_{\mathrm{b}}\mathrm{p} \tag{$\sqsupseteq\hat{\Box}$}$$

$$\mathrm{a} \sqsubseteq \mathrm{b} \vdash \hat{\Diamond}_{\mathrm{a}}\mathrm{p} \Rightarrow \hat{\Diamond}_{\mathrm{b}}\mathrm{p} \tag{$\sqsubseteq\hat{\Diamond}$}$$

In particular, when a is atomic, $\hat{\Box}_{\mathrm{a}}\mathrm{p} \Leftrightarrow \hat{\Diamond}_{\mathrm{a}}\mathrm{p}$. This leaves us, for atomic agents, with a single mode of ascription, equivalent to the classical belief operator, **B**.

[5] 'Actual' agents are the ones in the problem domain—usually people or machines, but possibly software components, personality facets or large political entities in some applications.

A practical agent can sacrifice precision by performing the same collapse of $\hat{\Box}$ and $\hat{\Diamond}$ even in the *non*-atomic case, weakening the above to (contextually or tactically controlled) default rules:

$$\mathrm{a} \sqsubseteq \mathrm{b} \vdash \mathbf{B}_{\mathrm{a}}\mathrm{p} \rightarrow \mathbf{B}_{\mathrm{b}}\mathrm{p} \qquad (\sqsupseteq\mathbf{B})$$

$$\mathrm{a} \sqsupseteq \mathrm{b} \vdash \mathbf{B}_{\mathrm{a}}\mathrm{p} \rightarrow \mathbf{B}_{\mathrm{b}}\mathrm{p} \qquad (\sqsubseteq\mathbf{B})$$

Such collapsed belief structures provide a computationally (if not logically) natural approach to the mutual belief problem, and offer hope of an alternate approach to the problems surrounding joint intentions. Because of the way that such corporate belief overgenerates ascriptions relative to sound reasoning, we need only corporate belief and the obvious generalisation of the conventional axiom of positive introspection:

$$\mathrm{a} \sqsubseteq \mathrm{b} \vdash \mathbf{B}_{\mathrm{b}}\mathrm{p} \Rightarrow \mathbf{B}_{\mathrm{a}}\mathbf{B}_{\mathrm{b}}\mathrm{p} \qquad (\mathbf{B}+)$$

to generate all the ramifications of classical mutual belief from corporate belief.

The question remains as to where corporate belief itself arises. Corporate belief is licenced by inference with respect to a *corporate charter*, a set of propositions specifying the relative roles of the members of the corporation and the corporation's domain of concern. The corporate charter controls the flow of beliefs, goals, and obligations through the corporation. Two groups may have the same members but different charters, and hence, different patterns of beliefs. In effect, the corporate charter constrains the application of the default rules to the actual domain of relevance. In the more precise model, this corresponds to listing the conditions under which conclusions stronger than just $\hat{\Diamond}_{\mathrm{a}}\mathrm{p}$ are justified.

We model conversation partners as constituting a nonce corporation. Thus, the notion of the corporate charter subsumes that of the discourse context. Furthermore, the effect of a successful Inform speech act is to raise a proposition to corporate belief from the speaking agent's beliefs. Precisely this perlocutionary effect is used by the planner in selecting an action to perform. Tentatively, we formulate Inform thus:

$$\mathsf{Inform}_{\mathrm{x,y}}\mathrm{p} \land \mathrm{x} \sqsubseteq \mathrm{z} \land \mathrm{y} \sqsubseteq \mathrm{z} \Rightarrow \mathbf{B}_{\mathrm{z}}\mathrm{p} \qquad (2)$$

General plan recognition methods allow recognition of the speaker's failed intention, achieving the illocutionary effect when required. This definition has simplicity and flexibility, but too much of the latter; the particular difficulty is in locating an appropriate z. What is needed is the speech act equivalent of the corporate charter, to discipline otherwise promiscuous inferences.

The solution to this problem can be had by taking the hearer's role in communication seriously. Consider the claim [?] that speech acts are fully joint activities; a speech act, even an Inform, is not complete until acknowledged by the hearer. In this theory, standard speech acts consist of a series of *contributions*, optional clarifications and hearer requests for clarification, and terminate in an acknowledgement that the contribution has been recognized. Because this theory provides independent justification for concluding that a speech act has been successful, we can use our new belief operator to write terse speech act definitions. Although the effects are defaults from a planning point of view, the acknowledgement greatly increases the certainty that the action was successful. Thus we have

$$\mathsf{Agree_z p} \Rightarrow \mathbf{B}_\mathrm{z}\mathrm{p} \tag{3}$$

This very general action can be achieved in several ways, such as through question/answer pairs. But for the moment we continue to discuss only the Inform speech act.

$$\mathsf{Agree_z p} \triangleq \mathsf{x} \sqsubseteq \mathsf{z} \wedge \mathsf{live\ z} \wedge \mathsf{inform_{x,z} p} \wedge \mathsf{pp\text{-}ack_z p} \vee \ldots \tag{4}$$

Here "inform" is the inform *contribution* as distinct from the Inform speech act, and must be followed by an acknowledgement. Our pp-ack is "possibly-passively acknowledge"; in many-agent discourses participants are frequently assumed to understand and agree unless they indicate otherwise. The "live" predicate covers traditional observation conditions; see below.

Should $\mathbf{B}_\mathrm{w}\neg\mathrm{p}$ be the case, for some $\mathrm{w} \sqsubseteq \mathrm{z}$, the default downward inheritance is blocked for w, who then has the option of withholding acknowledgement (possibly by cutting in). We thus allow recognition of the speaker's failed intention, as well as the proper illocutionary effect otherwise—even in the case where a proposition is accepted just for the sake of argument and is not *really* believed by all the assenting participants.

The corporate charter and observation conditions for such ordinary speech acts are conventionally established at a greeting:

$$\mathsf{greet}_{\mathsf{x}}\mathsf{y} \Rightarrow \exists_{\mathsf{z}}(\mathsf{x} \sqsubseteq \mathsf{z} \wedge \mathsf{y} \sqsubseteq \mathsf{z} \wedge \mathsf{live}\,\mathsf{z}) \tag{5}$$

For practical agents this formula must be constructive. The actual Skolem function for z includes lookup, for the case that x and y know each other in this social context. Otherwise, z must be constructed according to social conventions. It is always possible, of course, that the construction (and hence its its greeting) will fail.

The live z predicate, signifying that the discourse is active, then remains more or less constant. It is maintained by low-level dialogue processes. Thus our earlier problem of determining the social scope of each Inform is effectively solved.

Abductive speech act recognition can now be re-implemented on this firmer and more practical basis.

4 DISCO

At press time, the ideas presented in Section Two were implemented, and the ideas in Section Three planned, as part of the DISCO project. The DISCO (or DIalogue Systems for COoperating agents) project as a whole is a four-year, eight-person research effort funded by the German Ministry for Research and Technology. Its primary goal is processing of multiagent (that is, three or more agent) natural language dialogue. Multiagent capabilities make it an appropriate front end for autonomous cooperative agent and robotic systems, operating in groups with human participation.

DISCO is completing its third year. Its first task was to provide a uniform core formalism, based on unification of feature structures. The second was construction of a modular architecture orthogonal to the representation formalism, as a platform for experimentation. Third, research and construction of dialogue components is ongoing, as is (fourth) investigation of the interface between dialogue components and agent systems.

4.1 Uniform Core Formalism

The core formalism, as implemented by the unifier [3], is sufficient for the HPSG syntactic theory and includes

- atomic values
- feature-value pairs
- conjunction
- labelled (hence coreferrable) structures
- disjunction distributed across feature terms
- negation over enumerated types
- functional constraints

Planned extensions include subsumption testing, preferences, and possible generalization of negation.

The unifier is used by a multistrategy chart parser. There are several additional tools for manipulation of feature structures. The Type Definition Language permits static type checking and inheritance among feature structures, thus compacting and simplifying the debugging of grammars [14, 29]. FEGRAMED, the feature structure editor [12], permits customized display and easy editing of feature structures.

Other modules implemented using the core formalism include two morphology engines [33], reversible-grammar text generation [31, 22], and the surface speech act recognition described in Section Two.

4.2 System Architecture

The use of modern programming techniques in system integration has been crucial to DISCO's success. The system architecture was designed [24] to support the following desiderata:

- modularity of NLP components
- experimentation with flow of control
- incorporation of new modules
- building of subsystems and standalone applications
- accommodation of alternative modules with similar functionality

These goals are accomplished using object-oriented programming tools supplied by CLOS. The primary infrastructure is a 'contentless' schema, or frame system. The frame system is then given substance by integration of the independent modules and by definition of a flow of control between these modules. The principal object classes (system components) are the tool components, such as TDL or the command shell; natural language components, and the control component.

The control component directs the flow of information (Figure 1) between the other components, providing both robustness and flexibility. The main tasks of the control component are:

- to define which components constitute a desired subsystem
- to manage global memory and to call specific tools
- to collect output data from each component
- to check output data and report failures
- to determine which component should receive the new results

There is also a command shell for direct communication with the kernel. This method has met with great success. Various subsystems are indeed used for different portions of the development effort. As it happens, they largely use subsets of the flow of control shown abstractly in Figure 2. Moreover, for the COSMA application (see following section), approximately eight new common LISP modules were integrated into the system during one three week interval. We are aware of no other NL system matching these software engineering standards. A further test of the architecture will occur with the implementation of reliable speech acts. This requires that processing failures from various modules be returned to the pragmatics planner for initiation of clarification subdialogues.

4.3 Grammar

Present coverage of German in the HPSG formalism includes [21, 20]

- Nominal level:
  - determiners and numerals

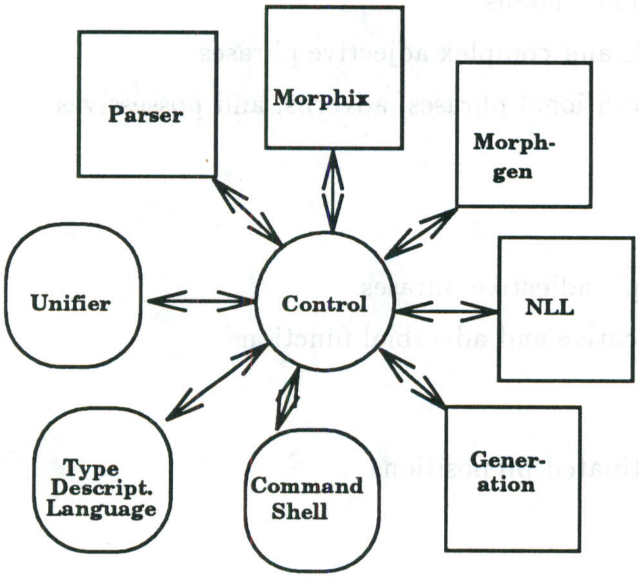

Figure 1: System Control Structure

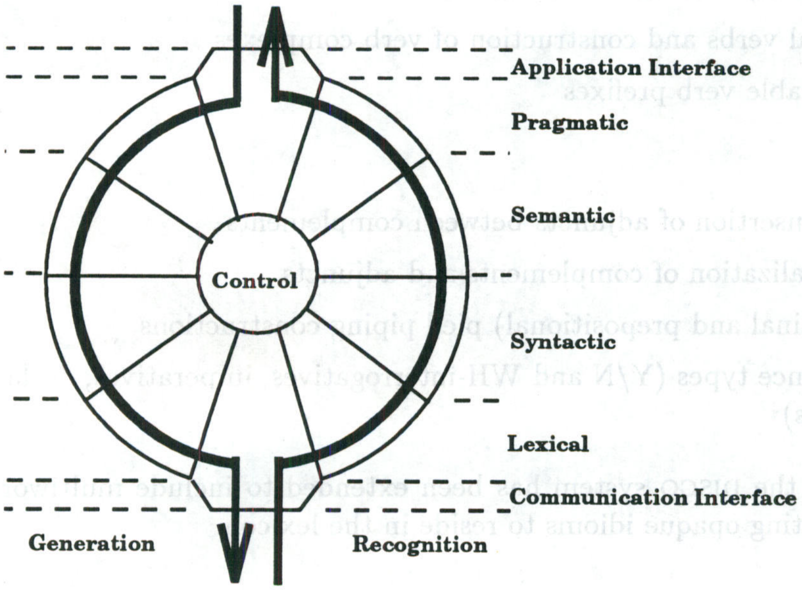

Figure 2: The DISCO System

  - bare plurals and mass nouns
  - prenominal simple and complex adjective phrases
  - postnominal prepositional phrases, adverbs, and possessives
  - nominal ellipsis

- Adjectival level:
  - complex and simple adjective phrases
  - attributive, predicative and adverbial functions

- Prepositional level:
  - simple and agglutinated prepositions
  - adverbs

- Verb level:
  - nominal, prepositional, and adjectival compliments
  - V-initial, V-second, and V-final
  - modal verbs and construction of verb complexes
  - separable verb prefixes

- Clauses:
  - free insertion of adjuncts between complements
  - topicalization of complements and adjuncts
  - (nominal and prepositional) pied piping constructions
  - sentence types (Y/N and WH-interrogatives, imperatives, declaratives)

In addition, the DISCO system has been extended to include multiword lexemes, permitting opaque idioms to reside in the lexicon.

4.4 The Logical Form

Although the core formalism has a great deal of power, the DISCO system also provides a logical form module which facilitates translation into various types of back-end systems. $\mathcal{NLL}$ is a representation of standard predicate logic, with lambda abstraction and several features that support representation of natural language constructions [17, 19, 18, 7]. These include

- Named predicate argument positions

```
ship(agt:Anterist th:shipment-47 time:15-Mar)
```

- Plural terms

Birke and Anter are shippers.

```
(exist ?x shipper(inst/i:?x) =(arg1:+{Birke, Anter} arg2:?x))
```

- Restricted parameters

A container is arriving.

```
arrive(th:(?x|container(inst/i:?x)))
```

- Location terms

Anterist is in Saarbrücken on the Saar.

```
located(th:Anterist loc:reg-X{SB, on-fn(Saar)})
```

- Generalized quantifiers
- Complex determiners

Modularity is achieved by providing several types of interfaces. $\mathcal{NLL}$ structures may be created using constructor functions, a structure parser, or a feature description language.

Truth-preserving transformations of $\mathcal{NLL}$ syntax are supported, to normalize or simplify structures into forms corresponding to client requirements.

```
(exist ?x shipper(inst/i:?x) =(arg1:+{Birke, Anter} arg2:x))
                              ⇓
             shipper(inst/i:+{Birke, Anter})
```

The goal of modifiability is realized through the implementation of $\mathcal{NLL}$ using the compiler tools, Zebu (a LISP version of YACC, [16]) and REFINE [28].

The September 1992 version of the COSMA system used $\mathcal{NLL}$ transformations and REFINE as a direct interface to the calendar management application. Future versions will use an enhanced version of this technology in the interface to discourse processing. In the next section, we will have a brief look at both.

5 COSMA

In order to prove a general-purpose facility, it is necessary to apply it in specific situations. The current testbed application of DISCO is a calendar agent named COSMA. In this section we describe its domain, our prototype implementation, and the changes at the pragmatic level that we look forward to in the coming year.

Appointment scheduling is a problem faced daily by many people and organizations, and typically solved using communication in natural language. It is therefore both a natural and a useful test of natural language systems. Other AI aspects include temporal [1, 13, 15] and spatial reasoning, and multiagent concepts such as negotiation and commitment [4]. As a testbed for evaluation of artificial intelligence techniques, it has many advantages: it is concrete, easily defined, and easily extensible, and its success can be judged by naive computer users, without recourse to specialists[6].

It is also readily compared with non-AI approaches: consider running an efficient scheduling algorithm over a centralized calendar database, spending cheap computer cycles to build optimal schedules. This does not address the issue of conflict resolution, and makes it impossible to utilize creative human input. Centralized solutions are also limited by administrative boundaries, privacy issues, and the usual concerns of reliability and failure modes. We are committed to giving individuals maximal control over their own schedules. Furthermore, we want to enable participation in the scheduling process, even for persons whose calendars are not on line. This will at times necessitate repeated rounds of communication. However, we believe it will lead to systems that are more useable and less hindrance.

[6] Our inspiration on this point was due to an elegant piece of work by Tom Mitchell [6]

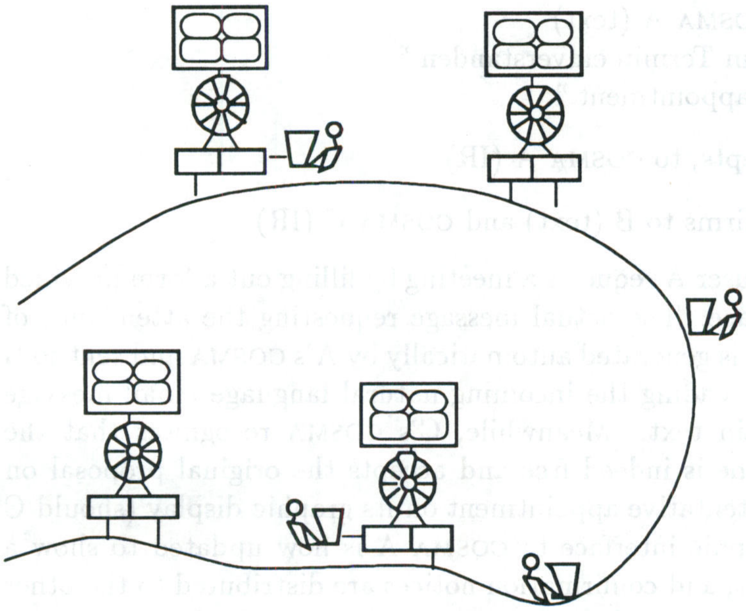

Figure 3: A COSMA Installation

The Co-Operative Schedule Management Agent, COSMA, is a secretarial assistant which maintains individuals' appointment calendars. If the user gives it permission to do so, it also communicates with other human beings and COSMA instances to satisfy appointment requests. A prototype system [23] fielded in September 1992, allowed human input through email text and through a Graphic User Interface; COSMA's generate email containing both text and their own Internal Representation language.

The following short exchanges are typical of those actually handled by this prototype. In these examples, user A communicates with a COSMA instance directly through the graphic interface. User B has no COSMA and communicates directly through electronic mail. Meanwhile, COSMA C has been instructed to operate autonomously, making arrangements for its user on its own authority.

- A enters a fully specified meeting via the graphic interface [*type:* meeting; *participants:* tick, trick, track; *time:* tuesday at 10:00; *place:* DFKI; *topic:* COSMA demo]
- A's COSMA sends Arrange to B (text) and COSMA C (IR)

- B accepts, to COSMA A (text)
  "Ich bin mit dem Termin einverstanden."
  "I agree to the appointment."

- C's COSMA accepts, to COSMA A (IR)

- A's COSMA confirms to B (text) and COSMA C (IR)

In this first example, user A requests a meeting by filling out a form provided by the graphic interface. The actual message requesting the attendance of the other participants is generated automatically by A's COSMA and sent to B and C. B, personally reading the incoming natural language email message responds, also in plain text. Meanwhile, C's COSMA recognises that the requested meeting time is indeed free and accepts the original proposal on C's behalf, posting a tentative appointment on its graphic display (should C be present). The graphic interface to COSMA A is now updated to show a confirmed engagement, and confirmation notices are distributed to the other participants; in the case that those participants (like C) also have COSMA's, their displays are once again updated.

In a second, shorter, exchange, user B (who is assumed to wield some authority) cancels the appointment:

- B cancels the original appointment with COSMA A (text)
  "Den Termin am 9. November sage ich ab."
  "I cancel the appointment of the 9th of November."

- COSMA A (after, in principle, checking for permission) propagates the cancellation to COSMA C.

Although B's email is addressed to A, A's COSMA intercepts it directly.[7] It then resolves the reference in the message to a particular appointment (in this case the one set up in the first exchange). Since the message is a request for cancellation by other than the original caller of the meeting, it checks that the sender is authorised for this action, updates its display, and propagates the cancellation to the remaining participant.

In this version of the system [30], the back end consists of a calendar database, time processing functions, a finite-state protocol for arranging appointments, and an Action Memory storing the protocol state and original

[7] This is a hack.

email for each arrangement in progress. The protocol steps are based on the "abstract speech acts" of [35] (though in fact these match the state transitions of the application rather than those of unconstrained natural language). In recognition, surface speech act predicates are mapped [25] onto disjunctions of protocol steps. In generation, protocol steps are realized by mapping into text templates.

Because protocol elements are only very loosely related to surface speech acts and to text, direct mapping is a very brittle method. Indeed, this can be seen in both of the examples above. In the first exchange, for example, the final confirmation would have been unnecessary if the system had been able to recognise the need for mutual understanding of the proposal and satisfy it with a multicast exchange. Conversely, in the second, a pragmatically necessary confirmation to B is not provided, because in this case the application considers itself already to have enough information to proceed.

The planned architecture moves reasoning about actions back into the linguistic system, placing it between the propositional level and application programs. Actions of back end programs, such as calendar databases, negotiators, and email addressbooks, are modelled in just enough detail for planning and plan recognition. Appointment arrangement, which is realized using speech acts to affect other agents' beliefs and goals, will be carried out using Corporate Agents and joint speech acts as described in Section Three.

Figure 4 shows an instance of the revised system, which refers to itself as A. The corporate agent labelled "nonce{A,B}" is a current dialogue context, with attendant discourse memory, for a dialogue with agent B, and (since A and B are previously acquainted) this nonce corporation inherits from the long-term corporation "{A,B}". Beliefs can also be imported or exported from the corporation which is A's model of B.

This context is active at the moment of the snapshot because the system has recently received incoming mail from B, and it is thought by the system to be part of an ongoing exchange. The input side has by now parsed the natural language content of the message, producing three candidate speech act interpretations.

In the snapshot, only the first of the three actions has been shown to be contextually consistent. It has been recognised as part of a plan whose first step is for nonce{A,B} to come to agreement of an item held in A's own calendar database (in fact, this first step is a question/answer pair initiated

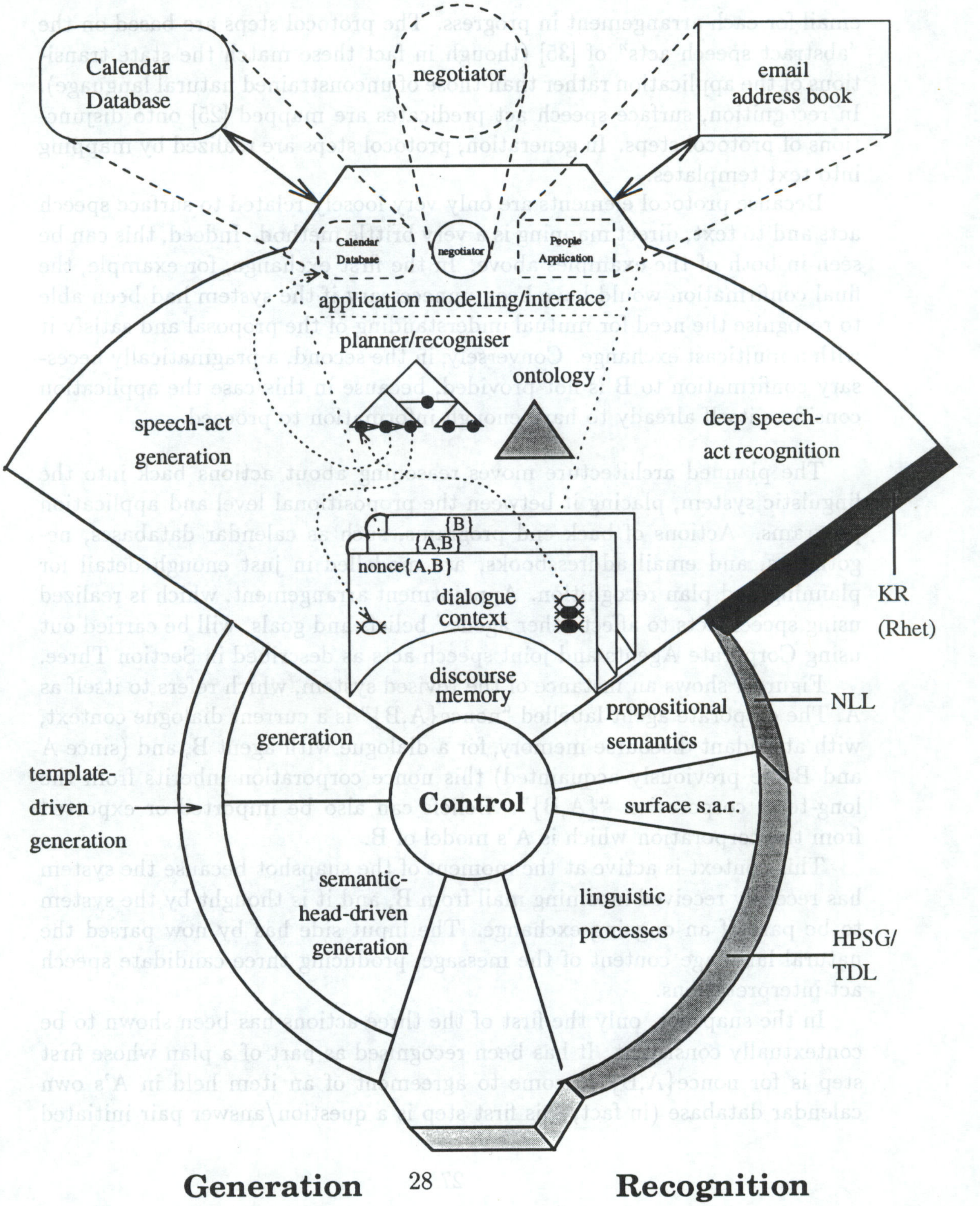

Figure 4: COSMA September 1993 (Planned)

by B). At the present moment A is partway through its contribution to satisfying this subgoal, having performed the needed local database retrieval and being in the process of generating the appropriate speech act for the reply.

(Particularly in spoken dialogue grounding acts are needed to ground communication. Since electronic mail is a "batch mode" medium grounding is more usually piggybacked on the core speech acts. In either case they are in principle modelled but are not evident at the level of abstraction of this diagram.)

In the new architecture, it will be seen that the entire discourse function has been retracted into the natural language system. The exact line of demarcation remains somewhat flexible. In particular, the figure shows a negotiation advising module implemented, for convenience, entirely within the application model; while the calendar database retains an external existence as a file and the "address book" is a separate program. What is crucial is that each of these elements has a model in the application interface and has operations which, just like speech acts, are fully present to the planner.

A point *not* visible in the picture, but crucial given our present concern with abductive reasoning, is that the system is moving steadily towards a symmetric, reversible approach. The HPSG syntactic level is fully declarative and in principle reversible, and the more general (non-template-driven) generation system is planned to use this resource directly. As we have shown, speech act processing can similarly be viewed as a deductive/abductive pair using a single reversible repertoire. Finally, the process of plan recognition and generation that binds the two sides of the linguistic system together and to the back-end applications, by its very nature interleaves deductive and abductive inference to grow high-level plans into executable sequences of actions.

6 Conclusion

In this chapter we have offered solutions to a number of subproblems in pragmatics:

- general linkage between utterances and intentions
- specific mechanisms linking speech acts to surface representations

- improved knowledge indexing through corporate agents
- a practical n-way belief attribution mechanism
- reliable, n-way speech acts

We've shown how they can or do work in the COSMA context. Note that discourse modelling is important here not only for the usual reasons, but because COSMA itself relies on input using an approximation of discourse concepts. The only architecturally feasible solution is to treat a nontrivial pragmatic layer with extensive planning facilities as an integral part of the core language engine itself.

But perhaps the most exciting thing to emerge from our survey of DISCO pragmatics as abduction is the prospect of reversible implementations rather than just reversible theories. We have become increasingly excited by the potential for a declarative system with direct reversibility of rules arising from the HPSG implementation of our syntactic front end. As our formalism group adds support for preferences in the coming year, a direct bidirectional implementation of speech acts based on abduction will become feasible. We hope that the insights gained there will help us toward reversibility of planning/plan recognition.

It is relatively rare in artificial intelligence to find a large installation of skilled researchers who also produce software that works well together. The authors would like to thank the several dozen people who have worked on projects DISCO, COSMA, and ASL, most especially Günter Neumann, who makes the DISCO wheel go 'round, and Stephan Oepen, who can make mismatched interfaces perform anyway using sheer elbow grease.

References

[1] James Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–843, November 1983.

[2] James Allen. Recognizing intentions from natural language utterances. In Michael Brady and Robert C. Berwick, editors, *Computational Models of Discourse*. MIT Press, 1983.

[3] Rolf Backofen, Lutz Euler, and Guenther Goerz. Distributed disjunktions for life. In H. Boley and M. M. Richter, editors, *Proc. of the Inter. Workshop on Processing Declarative Knowledge (PDK'91)*, pages 161–170, Berlin, 1991. Springer Veralg.

[4] Phillip R. Cohen and Hector J. Levesque. Persistence, intention, and commitment. In P. R. Cohen, J. Morgan, and M. E. Pollack, editors, *Intentions in Communication.* MIT Press, 1990.

[5] Phillip R. Cohen and Hector J. Levesque. Rational interaction as the basis for communication. In P. R. Cohen, J. Morgan, and M. E. Pollack, editors, *Intentions in Communication.* MIT Press, 1990.

[6] Lisa Dent, Jesus Boticario, John McDermott, Tom Mitchell, and David Zabowski. A personal learning apprentice. In *Proceedings of the National Conference on Artificial Intelligence*, 1992.

[7] Abdel Kader Diagne and John Nerbonne. Flexible semantics communication in integrated speech/language architectures. In Günter Göerz, editor, *Tagungsband KONVENZ.* Springer, 1992.

[8] Robert P. Goldman and Eugene Charniak. A probabilistic approach to plan recognition and text understanding. In *Proc. Plan Recognition Workshop*, 1989.

[9] Elizabeth Hinkelman. *Linguisic and Pragmatic Constraints on Utterance Interpretation.* PhD thesis, University of Rochester, 1990.

[10] Elizabeth A. Hinkelman and James F. Allen. Two constraints on speech act ambiguity. In *Proceedings ACL-89*, pages 212–219, 1989.

[11] Jerry Hobbs, Mark Stickel, Paul Martin, and Douglas Edwards. Interpretation as abduction. In *Proceedings ACL-88*, pages 95–103, 1988.

[12] Bernd Kiefer and Stephan Diehl. *FEGRAMED: An Interactive Graphics Editor for Feature Structures.* DISCO, 1992.

[13] Johannes A.G.M. Koomen. *Reasoning About Recurrence.* PhD thesis, University of Rochester, 1989.

[14] Hans-Ulrich Krieger and John Nerbonne. Feature-based inheritance networks for computational lexicons. In Ted Briscoe, Anne Copestake, and Valerie de Paiva, editors, *Default Inheritance within Unification-Based Approaches to the Lexicon*. Cambridge University Press, Cambridge, 1992. also DFKI Research Report RR-91-31.

[15] Leslie Lamport. The synchronization of independent processes. *Acta Informatica*, 7:15–34, 1977.

[16] Joachim Laubsch. Zebu: A tool for specifying reversible lalr(1) parsers. Technical report, Hewlett-Packard Laboratories, Palo Alto, CA, July 1992.

[17] Joachim Laubsch and John Nerbonne. An overview of \↕↕. Technical report, Hewlett-Packard Laboratories, Palo Alto, July 1991.

[18] John Nerbonne. $\mathcal{NLL}$ models. Research report, DFKI, Saarbrücken, 1992.

[19] John Nerbonne. Constraint-based semantics. In Paul Dekker and Martin Stokhof, editors, *Proceedings of the 8th Amsterdam Colloquium*, pages 425–444. Institute for Logic, Language and Computation, 1992. also DFKI RR-92-18.

[20] Klaus Netter. Morphosyntax of german noun phrases. In John Nerbonne, Klaus Netter, and Carl Pollard, editors, *German Grammar in HPSG*, pages xxx–yyy, Stanford, 1993. CSLI.

[21] Klaus Netter. Structure and coverage of the disco grammar. In Stephan Busemann and Karin Harbusch, editors, *Proc. DFKI Workshop on Natural Language Systems: Modularity and Re-usability*, Saarbrücken, Germany, 1993.

[22] Günter Neumann. A bidirectional model for natural language processing. Technical report, Computerlinguistik an der Universitt des Saarlandes, Saarbrücken, Germany, 1991.

[23] Günter Neumann, Stephan Oepen, and Stephen P. Spackman. Design and implementation of the COSMA system. Technical report, Deutsches

Forschungszentrum für Künstliche Intelligenz, Saarbrücken, Germany, 1993.

[24] Günter Neumann and Gertjan van Noord. Reversibility and self-monitoring in natural language generation. In Tomek Strzalkowski, editor, *Reversible Grammars and Natural Language Processing*. Kluwer, 1992. to appear.

[25] Stephan Oepen. Mapping from $\mathcal{NLL}$ to cosma-ir. In *EACL 93*, 1993. submitted to EACL student session.

[26] C. R. Perrault. An application of default logic to speech act theory. In P. R. Cohen, J. Morgan, and M. E. Pollack, editors, *Intentions in Communication*. MIT Press, 1990.

[27] Carl Pollard and Ivan Sag. *An Information-Based Theory of Syntax and Semantics, Vol.I.* CSLI, Stanford, 1987.

[28] Reasoning Systems Inc., Palo Alto. REFINE *User's Guide*, May 1990.

[29] Ulrich Schäfer and Hans-Ulrich Krieger. TDL *extra-light User's Guide: Franz Allegro Common LISP Version.* DISCO, 1992.

[30] Achim W. Schupeta. Cosma: Ein verteilter terminplaner als fallstudie der verteilten ki. In Juergen Mueller and Donald Steiner, editors, *Kooperierende Agenten*, Saarbrücken, Germany, 1993.

[31] Stuart M. Shieber, Gertjan van Noord, Robert C. Moore, and Fernando C. N. Pereira. A semantic-head-driven generation algorithm for unification-based formalisms. *Computational Linguistics*, 16(1):30–42, 1990.

[32] Stephen P. Spackman and Elizabeth A. Hinkelman. Corporate agents. In *subm. AAAI-93*, 1993.

[33] Harald Trost. The application of two-level morphology to non-concatenative german morphology. In *Proceedings of the 13th International Conference on Computational Linguistics (COLING)*, 1990.

[34] Yorick Wilks. Belief ascription and translation. PLUS-ABC-92, Workshop on Computational Pragmatics, Alghero, 14-18 Sept. 1992. Invited Talk.

[35] Terry Winograd and Fernando Flores. *Understanding Computers and Cognition.* Ablex, Norwood, NJ, 1986.

Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

DFKI
-Bibliothek-
PF 2080
67608 Kaiserslautern
FRG

DFKI Publikationen

Die folgenden DFKI Veröffentlichungen sowie die aktuelle Liste von allen bisher erschienenen Publikationen können von der oben angegebenen Adresse oder per anonymem ftp von ftp.dfki.uni-kl.de (131.246.241.100) unter pub/Publications bezogen werden.

Die Berichte werden, wenn nicht anders gekennzeichnet, kostenlos abgegeben.

DFKI Publications

The following DFKI publications or the list of all published papers so far are obtainable from the above address or per anonymous ftp from ftp.dfki.uni-kl.de (131.246.241.100) under pub/Publications.

The reports are distributed free of charge except if otherwise indicated.

DFKI Research Reports

RR-92-48
Bernhard Nebel, Jana Koehler:
Plan Modifications versus Plan Generation: A Complexity-Theoretic Perspective
15 pages

RR-92-49
Christoph Klauck, Ralf Legleitner, Ansgar Bernardi:
Heuristic Classification for Automated CAPP
15 pages

RR-92-50
Stephan Busemann:
Generierung natürlicher Sprache
61 Seiten

RR-92-51
Hans-Jürgen Bürckert, Werner Nutt:
On Abduction and Answer Generation through Constrained Resolution
20 pages

RR-92-52
Mathias Bauer, Susanne Biundo, Dietmar Dengler, Jana Koehler, Gabriele Paul: PHI - A Logic-Based Tool for Intelligent Help Systems
14 pages

RR-92-53
Werner Stephan, Susanne Biundo:
A New Logical Framework for Deductive Planning
15 pages

RR-92-54
Harold Boley: A Direkt Semantic Characterization of RELFUN
30 pages

RR-92-55
John Nerbonne, Joachim Laubsch, Abdel Kader Diagne, Stephan Oepen: Natural Language Semantics and Compiler Technology
17 pages

RR-92-56
Armin Laux: Integrating a Modal Logic of Knowledge into Terminological Logics
34 pages

RR-92-58
Franz Baader, Bernhard Hollunder:
How to Prefer More Specific Defaults in Terminological Default Logic
31 pages

RR-92-59
Karl Schlechta and David Makinson: On Principles and Problems of Defeasible Inheritance
13 pages

RR-92-60
Karl Schlechta: Defaults, Preorder Semantics and Circumscription
19 pages

RR-93-02
Wolfgang Wahlster, Elisabeth André, Wolfgang Finkler, Hans-Jürgen Profitlich, Thomas Rist:
Plan-based Integration of Natural Language and Graphics Generation
50 pages

RR-93-03
Franz Baader, Berhard Hollunder, Bernhard Nebel, Hans-Jürgen Profitlich, Enrico Franconi:
An Empirical Analysis of Optimization Techniques for Terminological Representation Systems
28 pages

RR-93-04
Christoph Klauck, Johannes Schwagereit:
GGD: Graph Grammar Developer for features in CAD/CAM
13 pages

RR-93-05
Franz Baader, Klaus Schulz: Combination Techniques and Decision Problems for Disunification
29 pages

RR-93-06
Hans-Jürgen Bürckert, Bernhard Hollunder, Armin Laux: On Skolemization in Constrained Logics
40 pages

RR-93-07
Hans-Jürgen Bürckert, Bernhard Hollunder, Armin Laux: Concept Logics with Function Symbols
36 pages

RR-93-08
Harold Boley, Philipp Hanschke, Knut Hinkelmann, Manfred Meyer: COLAB : A Hybrid Knowledge Representation and Compilation Laboratory
64 pages

RR-93-09
Philipp Hanschke, Jörg Würtz:
Satisfiability of the Smallest Binary Program
8 Seiten

RR-93-10
Martin Buchheit, Francesco M. Donini, Andrea Schaerf: Decidable Reasoning in Terminological Knowledge Representation Systems
35 pages

RR-93-11
Bernhard Nebel, Hans-Juergen Buerckert:
Reasoning about Temporal Relations:
A Maximal Tractable Subclass of Allen's Interval Algebra
28 pages

RR-93-12
Pierre Sablayrolles: A Two-Level Semantics for French Expressions of Motion
51 pages

RR-93-13
Franz Baader, Karl Schlechta:
A Semantics for Open Normal Defaults via a Modified Preferential Approach
25 pages

RR-93-14
Joachim Niehren, Andreas Podelski,Ralf Treinen:
Equational and Membership Constraints for Infinite Trees
33 pages

RR-93-15
Frank Berger, Thomas Fehrle, Kristof Klöckner, Volker Schölles, Markus A. Thies, Wolfgang Wahlster: PLUS - Plan-based User Support Final Project Report
33 pages

RR-93-16
Gert Smolka, Martin Henz, Jörg Würtz: Object-Oriented Concurrent Constraint Programming in Oz
17 pages

RR-93-17
Rolf Backofen:
Regular Path Expressions in Feature Logic
37 pages

RR-93-18
Klaus Schild: Terminological Cycles and the Propositional μ-Calculus
32 pages

RR-93-20
Franz Baader, Bernhard Hollunder:
Embedding Defaults into Terminological Knowledge Representation Formalisms
34 pages

RR-93-22
Manfred Meyer, Jörg Müller:
Weak Looking-Ahead and its Application in Computer-Aided Process Planning
17 pages

RR-93-23
Andreas Dengel, Ottmar Lutzy:
Comparative Study of Connectionist Simulators
20 pages

RR-93-24
Rainer Hoch, Andreas Dengel:
Document Highlighting —
Message Classification in Printed Business Letters
17 pages

RR-93-25
Klaus Fischer, Norbert Kuhn: A DAI Approach to Modeling the Transportation Domain
93 pages

RR-93-26
Jörg P. Müller, Markus Pischel: The Agent Architecture InteRRaP: Concept and Application
99 pages

RR-93-27
Hans-Ulrich Krieger:
Derivation Without Lexical Rules
33 pages

RR-93-28
Hans-Ulrich Krieger, John Nerbonne, Hannes Pirker: Feature-Based Allomorphy
8 pages

RR-93-29
Armin Laux: Representing Belief in Multi-Agent Worlds viaTerminological Logics
35 pages

RR-93-30
Stephen P. Spackman, Elizabeth A. Hinkelman:
Corporate Agents
14 pages

RR-93-31
Elizabeth A. Hinkelman, Stephen P. Spackman:
Abductive Speech Act Recognition, Corporate Agents and the COSMA System
34 pages

RR-93-32
David R. Traum, Elizabeth A. Hinkelman:
Conversation Acts in Task-Oriented Spoken Dialogue
28 pages

RR-93-33
Bernhard Nebel, Jana Koehler:
Plan Reuse versus Plan Generation: A Theoretical and Empirical Analysis
33 pages

RR-93-34
Wolfgang Wahlster:
Verbmobil Translation of Face-To-Face Dialogs
10 pages

RR-93-35
Harold Boley, François Bry, Ulrich Geske (Eds.):
Neuere Entwicklungen der deklarativen KI-Programmierung — *Proceedings*
150 Seiten
Note: This document is available only for a nominal charge of 25 DM (or 15 US-$).

RR-93-36
Michael M. Richter, Bernd Bachmann, Ansgar Bernardi, Christoph Klauck, Ralf Legleitner, Gabriele Schmidt: Von IDA bis IMCOD: Expertensysteme im CIM-Umfeld
13 Seiten

RR-93-38
Stephan Baumann: Document Recognition of Printed Scores and Transformation into MIDI
24 pages

RR-93-40
Francesco M. Donini, Maurizio Lenzerini, Daniele Nardi, Werner Nutt, Andrea Schaerf:
Queries, Rules and Definitions as Epistemic Statements in Concept Languages
23 pages

RR-93-41
Winfried H. Graf: LAYLAB: A Constraint-Based Layout Manager for Multimedia Presentations
9 pages

RR-93-42
Hubert Comon, Ralf Treinen:
The First-Order Theory of Lexicographic Path Orderings is Undecidable
9 pages

RR-93-44
Martin Buchheit, Manfred A. Jeusfeld, Werner Nutt, Martin Staudt: Subsumption between Queries to Object-Oriented Databases
36 pages

RR-93-45
Rainer Hoch: On Virtual Partitioning of Large Dictionaries for Contextual Post-Processing to Improve Character Recognition
21 pages

RR-93-46
Philipp Hanschke: A Declarative Integration of Terminological, Constraint-based, Data-driven, and Goal-directed Reasoning
81 pages

DFKI Technical Memos

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

TM-92-02
Achim Schupeta: Organizing Communication and Introspection in a Multi-Agent Blocksworld
32 pages

TM-92-03
Mona Singh:
A Cognitiv Analysis of Event Structure
21 pages

TM-92-04
Jürgen Müller, Jörg Müller, Markus Pischel, Ralf Scheidhauer:
On the Representation of Temporal Knowledge
61 pages

TM-92-05
Franz Schmalhofer, Christoph Globig, Jörg Thoben:
The refitting of plans by a human expert
10 pages

TM-92-06
Otto Kühn, Franz Schmalhofer: Hierarchical skeletal plan refinement: Task- and inference structures
14 pages

TM-92-08
Anne Kilger: Realization of Tree Adjoining Grammars with Unification
27 pages

TM-93-01
Otto Kühn, Andreas Birk: Reconstructive Integrated Explanation of Lathe Production Plans
20 pages

TM-93-02
Pierre Sablayrolles, Achim Schupeta:
Conlfict Resolving Negotiation for COoperative Schedule Management
21 pages

TM-93-03
Harold Boley, Ulrich Buhrmann, Christof Kremer:
Konzeption einer deklarativen Wissensbasis über recyclingrelevante Materialien
11 pages

TM-93-04
Hans-Günther Hein: Propagation Techniques in WAM-based Architectures — The FIDO-III Approach
105 pages

TM-93-05
Michael Sintek: Indexing PROLOG Procedures into DAGs by Heuristic Classification
64 pages

DFKI Documents

D-92-28
Klaus-Peter Gores, Rainer Bleisinger: Ein Modell zur Repräsentation von Nachrichtentypen
56 Seiten

D-93-01
Philipp Hanschke, Thom Frühwirth: Terminological Reasoning with Constraint Handling Rules
12 pages

D-93-02
Gabriele Schmidt, Frank Peters, Gernod Laufkötter: User Manual of COKAM+
23 pages

D-93-03
Stephan Busemann, Karin Harbusch(Eds.): DFKI Workshop on Natural Language Systems: Reusability and Modularity - Proceedings
74 pages

D-93-04
DFKI Wissenschaftlich-Technischer Jahresbericht 1992
194 Seiten

D-93-05
Elisabeth André, Winfried Graf, Jochen Heinsohn, Bernhard Nebel, Hans-Jürgen Profitlich, Thomas Rist, Wolfgang Wahlster: PPP: Personalized Plan-Based Presenter
70 pages

D-93-06
Jürgen Müller (Hrsg.): Beiträge zum Gründungsworkshop der Fachgruppe Verteilte Künstliche Intelligenz, Saarbrücken, 29. - 30. April 1993
235 Seiten
Note: This document is available only for a nominal charge of 25 DM (or 15 US-$).

D-93-07
Klaus-Peter Gores, Rainer Bleisinger: Ein erwartungsgesteuerter Koordinator zur partiellen Textanalyse
53 Seiten

D-93-08
Thomas Kieninger, Rainer Hoch: Ein Generator mit Anfragesystem für strukturierte Wörterbücher zur Unterstützung von Texterkennung und Textanalyse
125 Seiten

D-93-09
Hans-Ulrich Krieger, Ulrich Schäfer: TDL ExtraLight User's Guide
35 pages

D-93-10
Elizabeth Hinkelman, Markus Vonerden,Christoph Jung: Natural Language Software Registry (Second Edition)
174 pages

D-93-11
Knut Hinkelmann, Armin Laux (Eds.): DFKI Workshop on Knowledge Representation Techniques — Proceedings
88 pages

D-93-12
Harold Boley, Klaus Elsbernd, Michael Herfert, Michael Sintek, Werner Stein: RELFUN Guide: Programming with Relations and Functions Made Easy
86 pages

D-93-14
Manfred Meyer (Ed.): Constraint Processing – Proceedings of the International Workshop at CSAM'93, July 20-21, 1993
264 pages
Note: This document is available only for a nominal charge of 25 DM (or 15 US-$).

D-93-15
Robert Laux: Untersuchung maschineller Lernverfahren und heuristischer Methoden im Hinblick auf deren Kombination zur Unterstützung eines Chart-Parsers
86 Seiten

D-93-16
Bernd Bachmann, Ansgar Bernardi, Christoph Klauck, Gabriele Schmidt: Design & KI
74 Seiten

D-93-20
Bernhard Herbig: Eine homogene Implementierungsebene für einen hybriden Wissensrepräsentationsformalismus
97 Seiten

D-93-21
Dennis Drollinger: Intelligentes Backtracking in Inferenzsystemen am Beispiel Terminologischer Logiken
53 Seiten

D-93-22
Andreas Abecker: Implementierung graphischer Benutzungsoberflächen mit Tcl/Tk und Common Lisp
44 Seiten

D-93-24
Brigitte Krenn, Martin Volk: DiTo-Datenbank: Datendokumentation zu Funktionsverbgefügen und Relativsätzen
66 Seiten

D-93-25
Hans-Jürgen Bürckert, Werner Nutt (Eds.): Modeling Epistemic Propositions
118 pages
Note: This document is available only for a nominal charge of 25 DM (or 15 US-$).

D-93-26
Frank Peters: Unterstützung des Experten bei der Formalisierung von Textwissen
INFOCOM - Eine interaktive Formalisierungskomponente
58 Seiten

Abductive Speech Act Recognition, Corporate Agents and the COSMA System

Elizabeth A. Hinkelman, Stephen P. Spackman

RR-93-31

Research Report