



**Deutsches  
Forschungszentrum  
für Künstliche  
Intelligenz GmbH**

**Document**

D-95-08

## **DFKI-Jahresbericht 1994**

(durch BMBF-Rahmenvertrag geförderte Projekte)

**Kaiserslautern und Saarbrücken**

**März 1995**

**Deutsches Forschungszentrum für Künstliche Intelligenz  
GmbH**

Postfach 20 80  
67608 Kaiserslautern, FRG  
Tel.: + 49 (631) 205-3211  
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, Siemens and Siemens-Nixdorf. Research projects conducted at the DFKI are funded by the German Ministry of Education, Science, 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.

Dr. Dr. D. Ruland  
Director

## **DFKI-Jahresbericht 1994**

DFKI-D-95-08

Redaktionsschluß: März 1995

© Deutsches Forschungszentrum für Künstliche Intelligenz 1995

This work may not be copied or reproduced in whole or part for any commercial purpose. Permission to copy in whole or part without payment of fee is granted for nonprofit educational and research purposes provided that all such whole or partial copies include the following: a notice that such copying is by permission of the Deutsche Forschungszentrum für Künstliche Intelligenz, Kaiserslautern, Federal Republic of Germany; an acknowledgement of the authors and individual contributors to the work; all applicable portions of this copyright notice. Copying, reproducing, or republishing for any other purpose shall require a licence with payment of fee to Deutsches Forschungszentrum für Künstliche Intelligenz.  
ISSN 0946-0098

# **DFKI-Jahresbericht 1994**

(durch BMBF-Rahmenvertrag geförderte Projekte)

**Deutsches Forschungszentrum für  
Künstliche Intelligenz GmbH**

**Kaiserslautern und Saarbrücken**

**März 1995**

**DFKI-Dokument D-95-08**

---

## Inhaltsverzeichnis

<b>1</b>	<b>AKA-MOD: Modellierung kooperierender Agenten</b>	<b>4</b>
1.1	Kurzüberblick und Projektfortschritt . . . . .	4
1.2	Wissenschaftliche Ergebnisse . . . . .	5
1.2.1	Das Speditionsszenario . . . . .	5
1.2.2	Das Verladehofszenario (FORKS) . . . . .	5
1.2.3	Die Testumgebung . . . . .	7
1.2.4	MIKADO . . . . .	7
1.2.5	Neuspezifikation des Agentenmodells und der Testumgebung . . . . .	8
1.3	Publikationen . . . . .	9
1.4	Sonstige Ereignisse . . . . .	10
<b>2</b>	<b>AKA-TACOS: Taxonomien und Alltagswissen</b>	<b>11</b>
2.1	Kurzüberblick und Projektfortschritt . . . . .	11
2.2	Wissenschaftliche Ergebnisse . . . . .	11
2.2.1	Modallogische Erweiterungen von terminologischen Logiken . . . . .	11
2.2.2	Das TACOS-System . . . . .	12
2.2.3	Terminologische Logiken und Objektorientierte Modellierung . . . . .	13
2.2.4	Terminologische Logiken und Datenbanksysteme . . . . .	14
2.2.5	Terminologische Logiken und Information Extraction . . . . .	14
2.2.6	Sonstige . . . . .	15
2.3	Publikationen . . . . .	15
2.4	Sonstige Ereignisse . . . . .	18
<b>3</b>	<b>COSMA: Cooperative Schedule Management Agent</b>	<b>19</b>
3.1	Kurzüberblick und Projektfortschritt . . . . .	19
3.2	Wissenschaftliche Ergebnisse . . . . .	19
3.2.1	Systemarchitektur und Erweiterung von Systemkomponenten . . . . .	19
3.2.2	Dialogsteuerung . . . . .	20
3.2.3	Zentrale Kontrolle der Verarbeitung . . . . .	21
3.2.4	Schnittstelle zum Anwendungssystem . . . . .	21
3.2.5	Zusammenarbeit mit anderen DFKI-Projekten . . . . .	22
3.3	Publikationen . . . . .	24
3.4	Sonstige Ereignisse . . . . .	24
<b>4</b>	<b>IMCOD: Intelligent Manager for Comprehensive Design</b>	<b>25</b>
4.1	Kurzüberblick und Projektfortschritt . . . . .	25
4.2	Wissenschaftliche Ergebnisse . . . . .	25
4.3	Publikationen . . . . .	30
4.4	Sonstige Ereignisse . . . . .	31
<b>5</b>	<b>HYDRA: Hybride Werkbank zur Konstruktion von deduktiven Problemlösern in wissensbasierten Systemen: Berechnung und Deduktion mit Constraints</b>	<b>32</b>
5.1	Kurzüberblick und Projektfortschritt . . . . .	32
5.2	Wissenschaftliche Ergebnisse . . . . .	32
5.2.1	Freigabe von DFKI Oz 1.0 . . . . .	32
5.2.2	Weiterentwicklung von Oz . . . . .	33
5.2.3	Wissenschaftliche Kontakte . . . . .	34

---

5.3	Publikationen . . . . .	35
5.4	Sonstige Ereignisse . . . . .	36
<b>6</b>	<b>OMEGA: Office Mail Expert for Goal-Directed Analysis</b>	<b>37</b>
6.1	Kurzüberblick und Projektfortschritt . . . . .	37
6.2	Wissenschaftliche Ergebnisse . . . . .	37
6.3	Publikationen . . . . .	40
6.4	Sonstige Ereignisse . . . . .	41
<b>7</b>	<b>PARADICE: Parameterizable NL Discourse Core Engine</b>	<b>42</b>
7.1	Kurzüberblick und Projektfortschritt . . . . .	42
7.2	Wissenschaftliche Ergebnisse . . . . .	42
7.2.1	Architektur und Diskurs . . . . .	42
7.2.2	Linguistische Wissenbasen . . . . .	43
7.2.3	Verarbeitung . . . . .	45
7.3	Veröffentlichungen . . . . .	47
7.4	Sonstige Ergebnisse . . . . .	48
<b>8</b>	<b>PPP: Personalized Plan-Based Presenter</b>	<b>49</b>
8.1	Kurzüberblick und Projektfortschritt . . . . .	49
8.2	Wissenschaftliche Ergebnisse . . . . .	50
8.2.1	Teilprojekt Präsentationsplanung (PPD) . . . . .	50
8.2.2	Teilprojekt Wissensrepräsentation und Reasoning (KR&R) . . . . .	52
8.3	Publikationen . . . . .	52
8.4	Sonstige Ereignisse . . . . .	53
<b>9</b>	<b>RAP: Reasoning About Plans</b>	<b>54</b>
9.1	Kurzüberblick und Projektfortschritt . . . . .	54
9.2	Wissenschaftliche Ergebnisse . . . . .	54
9.2.1	Der Repräsentationsformalismus . . . . .	54
9.2.2	Die Entwicklungsumgebung . . . . .	55
9.2.3	Die Inferenzdienste . . . . .	56
9.3	Publikationen . . . . .	57
9.4	Sonstige Ereignisse . . . . .	58
<b>10</b>	<b>VEGA: Validierung und Exploration von Wissen durch Globale Analyse</b>	<b>59</b>
10.1	Kurzüberblick und Projektfortschritt . . . . .	59
10.2	Wissenschaftliche Ergebnisse . . . . .	59
10.2.1	Die Evolution von Wissensbasen . . . . .	60
10.2.2	Deklarative Wissensrepräsentation . . . . .	60
10.2.3	Erstellung einer Anwendungswissensbasis . . . . .	60
10.3	Publikationen . . . . .	61

# 1 AKA-MOD: Modellierung kooperierender Agenten

## 1.1 Kurzüberblick und Projektfortschritt

Ziel des Projektes *Autonome, kooperierende Agenten* (AKA) ist die Untersuchung der Grundlagen für Wissensrepräsentationsformalismen, Inferenz-, Kommunikations- und Kontrollmechanismen bei der Kooperation mehrerer autonom agierender Systeme (*Agenten*). Die Resultate werden an verschiedenen realitätsnahen Anwendungsbeispielen erprobt. Das Teilprojekt *Grundlagen, Anwendungen und Implementierung von verteilten KI-Systemen: Modellierung kooperierender Agenten* (AKA-MOD, BMBF Förderkennzeichen ITW 9104) beschäftigt sich mit den für die Modellierung von Multiagentensystemen (MAS) und deren spezifischen Aspekten, insbesondere Interaktion, Kommunikation und Kontrolle, und untersucht konkrete Szenarien und modelliert sie durch die Implementierung von Prototypen. Als konkrete Szenarien, anhand derer Fragen zum verteilten Problemlösen mit MAS untersucht wurden, waren das Beispiel Türme von Hanoi und die Blockwelt, die Simulation einer Gruppe von Spediteuren mit unterschiedlicher Spezialisierung in Transportmitteln und Transportmöglichkeiten (Speditionsszenario), eine Gruppe von Robotfahrzeugen mit Ladeflächen und Manipulatoren (Verladehofszenario) sowie das Szenario einer automatisierten verteilten Terminvereinbarung.

Laut Arbeitsplan lag der Schwerpunkt der Arbeiten im ersten Quartal 1994 auf der Vervollständigung der Planungshierarchie im Agentenmodell InteRRaP, in der Erweiterung der Möglichkeiten zur Bereitstellung von generischen Agenten und in der Bereitstellung eines Auswertungsmoduls. Darüber hinaus wurden vor allem Konsolidierungsarbeiten durchgeführt, die die Benutzeroberfläche der beiden Hauptanwendungsbeispiele Speditionsszenario und Verlade-

---

hofszenario vereinheitlichten und die bereits bestehenden prototypischen Lösungen für Detailprobleme weiter verfeinerten.

Außerdem wurde in dieser Phase der Antrag für das AKA-MOD Nachfolgeprojekt CoMMA-MAPS erstellt. Im März 1994 fiel dann die Entscheidung, das Projekt AKA-MOD um 7 Monate bis zum 30. November 1994 zu verlängern. Die Definition der Themenstellung und der Beginn der Verlängerungsphase prägten die Tätigkeiten des zweiten Quartals 1994.

Im 3. und 4. Quartal 1994 wurden die Meilensteine der Verlängerungsphase bearbeitet. Dabei wurde das im Projekt AKA-MOD entwickelte allgemeine Multiagenten-Testbett auf der Basis der Programmiersprache Oz reimplementiert, weil Oz als innovative high-level Programmiersprache die Vorteile unterschiedlicher Programmierparadigmen in sich vereint und die besonderen Anforderungen von Multiagenten-Systemen besonders gut unterstützt. Dadurch erlaubt Oz ein schnelles Prototyping, was auch im Hinblick auf zukünftige Anwendungsdomänen vorteilhaft ist. Diese Umstellung war erst in der Endphase des Projekts AKA-MOD sinnvoll, da zu diesem Zeitpunkt eine hinreichend stabile und leistungsfähige Version von Oz vorlag. Das Speditionsszenario wurde vollständig unter Oz reimplementiert. Parallel erfolgte eine Erweiterung des FORKS-Szenarios unter Verwendung von KHEPERA-Miniaturrobotern.

Das Projekt wurde zum Jahresende 1994 erfolgreich abgeschlossen. Nachdem bereits im Frühjahr 1994 die im Projekt AKA-MOD geleisteten Arbeiten im Rahmen der wissenschaftlichen Begutachtung durch den wissenschaftlichen Beirat sehr positiv beurteilt wurden, wurde dieses Urteil auch für die in der Verlängerungsphase erarbeiteten Inhalte bei der wissenschaftlichen



## 1.2 Wissenschaftliche Ergebnisse

### 1.2.1 Das Speditionsszenario

Im Rahmen der Konsolidierungsarbeiten wurde die Funktionalität der Benutzerschnittstelle im Speditionsszenario so umgestaltet, daß eine weitgehende Übereinstimmung mit der Funktionalität der Benutzerschnittstelle des Verladehofszenarios erreicht wurde. Die Benutzeroberflächen sehen jetzt jeweils eine Menüleiste vor, die Grundfunktionalitäten zum Erzeugen von Agenten (Agenten können dynamisch erzeugt werden. Ihre Konfiguration kann aus prototypischen Agententypen abgeleitet werden.), zum Bekanntgeben von Aufträgen, zur Steuerung der Aufgabenausführung und zur statistischen Auswertung von Testläufen bereitstellt. Für die statistische Auswertung der Aktivitäten von Agenten wurde ein domänenunabhängiges Modul entwickelt, das sowohl im Speditionsszenario als auch im Verladehof-Szenario verwendet wird.

Die eigentliche Stärke des Multiagentenansatzes zur Lösung der im Speditionsszenario auftretenden Probleme liegt in der Möglichkeit, die in der Praxis auftretende Dynamik behandeln zu können. Hier ist zum einen die Dynamik in der Planung gemeint, bei der Aufträge asynchron zu einem beliebigen Zeitpunkt in das System eingebracht werden und das System diese Aufträge verarbeiten können muß. Zum anderen ist aber vor allem auch die Dynamik in der Ausführung der in den einzelnen Agenten berechneten Pläne gemeint. Als Dynamik wird hier die Möglichkeit bezeichnet, daß bei der Ausführung der von den einzelnen Agenten berechneten Pläne, einzelne Aktionen nicht exakt so ausgeführt werden können, wie sie zunächst eingeplant waren. Im Speditionsszenario bedeutet das, daß die Zeitdauer der einzelnen Aktionen (fahren von einem Ort an einen anderen, be- und entladen) für ihre Ausführung benötigen nicht mit der übereinstimmt, die ursprünglich zum Zeitpunkt eingeplant wurde; zu dem der Plan erstellt wurde.

Um hier ein möglichst realistisches Modell zugrunde zu legen, wurde im Rahmen eines Seminars in Zusammenarbeit mit dem Lehrstuhl für Betriebswirtschaftslehre insbesondere Wirtschaftsinformatik an der Universität des Saarlandes (Prof. Dr. G. Schmidt) ein mathematisches Modell für die Simulation von Schwankungen in der Verkehrsdichte entwickelt. Dieses Modell erlaubt es, für jede Verbindung zwischen zwei Städten einer Karte einen Staurisikofaktor zu definieren, der zusätzlich vom Monat im Jahr, dem Tag im Monat und der aktuellen Uhrzeit abhängt. In der Praxis wird dabei davon ausgegangen, daß dieser Staurisikofaktor durch Messungen mit Hilfe von Sensoren empirisch ermittelt wird. Der Staurisikofaktor definiert einen Erwartungswert, der als Grundlage für einen statistischen Prozeß dient, mit dem die jeweils aktuelle Verkehrsdichte auf einer bestimmten Strecke in der für die Simulation verwendeten Karte berechnet wird.

### 1.2.2 Das Verladehofszenario (FORKS)

Die prototypische Simulationsumgebung für das Verladehofszenario, die im Oktober 1993 dem wissenschaftlichen Beirat vorgeführt worden war, wurde im Berichtszeitraum fertiggestellt. Die Haupterweiterung war in der Spezifikation und Implementierung der Kooperationskomponente des InterRaP-Agentenmodells für die Gabelstapler-Roboter zu sehen. Die Kooperationskomponente (CC, engl. Cooperation Component) beinhaltet die kooperativen Planungsfähigkeiten eines Agenten. Die wichtigsten Forschungsergebnisse im Berichtszeitraum waren die Implementierung einer Plansprache für Mehragentenpläne (joint plans), eines erweiterten Verhandlungsprotokolls, das mehreren Agenten erlaubt, für eine gegebene Situation einen geeigneten gemeinsamen Plan zu finden, und die Spezifikation und Implementierung eines Verfahrens zur Transformation von joint plans in synchronisierte Einagentenpläne, die von der planbasierten Komponente des

InteRRaP-Modells interpretiert werden können.

Als Anwendung von kooperativer Planung im Verladehofszenario wurde der Fall der kooperativen Konfliktlösung behandelt. Zu diesem Zweck wurde als erstes eine Klassifikation von Konflikten zwischen Agenten im Verladehof erstellt. Konflikte zwischen Agenten werden dabei als alle Arten der (negativen) Beeinflussung der Erfüllung von Zielen eines Agenten durch andere Agenten definiert. In physikalischen Realwelt-Szenarien, wie beispielsweise Robotik-Applikationen, kommt zu dieser klassischen zielbasierten KI-Definition von Konflikten eine weitere Ausprägung des Konfliktbegriffes hinzu, nämlich Konflikte als physikalische Interaktion: so stellt eine drohende Kollision zwischen zwei Robotern einen physikalischen Konflikt dar, der - unabhängig

### 1.2.3 Die Testumgebung

Das MAGSY-System wurde gewartet und erweitert; hierzu zählen die Erhöhung der Stabilität von MAGSY und die Portierung auf neue Betriebssystem-Plattformen (Solaris, HP-UX, Linux). Als wesentliche Erweiterung von MAGSY wurde ein Kommunikationsanschluß an das Oz-System realisiert. Damit kann zur Programmierung der Agenten neben den Sprachen OPS5, Lisp, Prolog und C nun auch Oz verwendet werden. Dies findet bereits Anwendung im Speditionsszenario: die Agenten für Speditionen und LKW wurden in Oz reimplementiert, um die Brauchbarkeit von Oz als high-level Multiagenten-Implementierungssprache intensiv zu testen. Sie können gemischt mit den bisherigen (in OPS5 formulierten) Agenten eingesetzt werden.

Ein zweiter Schwerpunkt der Arbeit in diesem Bereich ergab sich aus der Entscheidung, eine neue Testumgebung zu entwickeln, die vollständig in Oz als Basissprache implementiert werden wird. Hierzu wurden Konzepte für die grundlegende Architektur des neuen Testbetts erarbeitet, welche die veränderten Gegebenheiten unter Oz berücksichtigen und die Möglichkeiten dieser Sprache ausnutzen. Die Konzepte orientieren sich an dem Agentenmodell InteRRaP sowie an den mit MAGSY gesammelten Erfahrungen. Zu den geleisteten Arbeiten zählen die Auswahl eines geeigneten Schemas zur Wissensrepräsentation, das Design eines situationserkennenden Regelinterpreters, sowie Entwürfe zur programmtechnischen Vereinbarkeit von reaktiven und deliberativen Verhalten.

### 1.2.4 MIKADO

Eine weitere Aktivität im Projekt AKA-MOD bestand in der Vorbereitung und Durchführung einer projektübergreifenden DFKI-Demonstration im Rahmen der CeBIT'94. Zusammen mit den BMFT geförderten Projekten ALV (Förderkennzeichen: ITW 9003 0) im Fachbereich Andreas Dengel und DISCO (Förderkennzeichen: ITW 9002 0) im Fachbereich Hans Uszkoreit und dem im Auftrag der Siemens AG entwickelten Projekt KIK-Teamware wurde am BMFT-Gemeinschaftsstand das System MIKADO (Multimedialer Intelligenter Kooperations-Assistent zur Dokumentbearbeitung) präsentiert. MIKADO stellt eine durchgängige, integrierte Lösung zur Bearbeitung eines Bürovorgangs bereit. Dazu werden Techniken der Dokumentenverarbeitung, der Sprachanalyse und der Multiagentensysteme eingesetzt.

Folgendes Szenario wurde exemplarisch vorgeführt: Ausgangspunkt ist der Eingang eines Briefes des Bundesministers an den Leiter der DFKI GmbH. In diesem Brief kündigt er seinen Besuch an und bittet den Leiter der DFKI GmbH darum, mit den Forschungsbereichsleitern einen Termin abzumachen, um über Neuheiten in der Förderpolitik zu informieren.

MIKADO unterstützt die DFKI-Chefsekretärin durchgängig bei der Bearbeitung des eingegangenen Briefes. Dazu muß der Brief lediglich von der Sekretärin eingescannt werden. In der ersten Stufe der automatischen Vorgangsbearbeitung wird der eingescannte Brief vom System ALV analysiert. Hierbei werden Layout-Struktur und logische Objekte wie beispielsweise Briefftyp, Absender, Empfänger, Betreff, Datum, Briefrumpf erkannt und zueinander in Beziehung gesetzt. ALV kann den Briefftyp als Terminvereinbarung klassifizieren und liefert in ASCII-

Das AKA-MOD/Teamware-Terminvereinbarungssystem initiiert automatisch einen Terminvereinbarungsprozeß, der zwischen den als Agenten modellierten Teilnehmern stattfindet. Den Teilnehmern zugeordnet sind "persönliche Assistenten" (Benutzeragenten), die softwaretechnisch realisiert sind und für die automatische Terminvereinbarung zuständig sind; der Benutzer kann spezifizieren, ob er diesem Assistenten voll seine Terminplanung anvertraut (wie in der Demo-Version), oder ob er jeden vereinbarten Termin noch einmal explizit bestätigen will. Die Benutzeragenten, die sich auf den jeweiligen Rechnern der Teilnehmer befinden, verhandeln über einen geeigneten Termin. Die Verhandlung erfolgt über einen Nachrichtenaustausch. Zur Verdeutlichung werden die beim Verhandlungsprozess ausgetauschten Nachrichten mittels einer Audio-Schnittstelle akustisch wiedergegeben ("sprechende" Benutzeragenten). Nach erfolgreicher Terminvereinbarung, die nach unterschiedlichen Kooperationsmethoden ablaufen kann, wird der vereinbarte Termin in die Kalender des DFKI-Leiters und der Forschungsbereichsleiter eingetragen.

In einem abschließenden Schritt kann auf Wunsch ein automatisch generiertes Antwortschreiben des DFKI-Leiters an den Bundesminister aus der vorliegenden Information erzeugt und gedruckt werden.

Neben der erforderlichen Anpassung des AKA-MOD/Teamware-Terminvereinbarungssystems

---

MOD auch die integrierende Oberfläche von MIKADO konzipiert und implementiert.

### 1.2.5 Neuspezifikation des Agentenmodells und der Testumgebung

Am 1.5.94 begann die Verlängerungsphase für das Projekt AKA-MOD. Ziel der Verlängerungsphase ist es, das Agentenmodell und die Testumgebung neu zu spezifizieren und zu implementieren.

bung übernommen werden können. Dazu wurde ein Modell entworfen, wie die Wissensrepräsentation und die Inferenzkomponenten eines Agenten in Oz modelliert werden können.

Als Kernpunkt in der Diskussion der Agentarchitekturen stellte sich die Frage heraus, wie die hierarchische Ebenenstruktur der InteRRaP-Agentenarchitektur (AKA-MOD) mit der Head-Body-Mouth-(HBM)-Architektur (CoMMA-Plat) vereint werden kann. Während bei der HBM-Architektur das Schlußfolgern des Agenten konzeptionell auf einer einzigen Ebene stattfindet, unterscheidet die InteRRaP-Agentenarchitektur Wissensrepräsentation und Schlußfolgerungen auf der verhaltensbasierten, der planbasierten und der kooperativen Ebene. Ende August stand eine abschließende Beurteilung der beiden Agentenarchitekturen fest, die auch die Neuspezifikation enthielt. Dieses Arbeitspapier bildete die Grundlage für die Reimplementierung des Agentenmodells.

In der zweiten Hälfte der Verlängerungsphase konzentrierten sich die Arbeiten auf diese Reimplementierungsarbeiten. Dabei wurde das im Projekt AKA-MOD entwickelte allgemeinen Multiagenten-Testbetts auf der Basis der Programmiersprache Oz reimplementiert. Neben der Reimplementierung der Agentenarchitektur wurde das Speditionsszenario als erste Anwendung für das neue Testbett vollständig reimplementiert.

### 1.3 Publikationen

- [Fis94] K. FISCHER: *The Design of an Intelligent Manufacturing System*. In: *Proceedings of the 2nd International Working Conference on Cooperating Knowledge-based Systems (CKBS'94)*, Keele, England, 1994.
- [FKM94b] K. FISCHER, N. KUHN und J. P. MÜLLER: *Distributed, Knowledge-Based, Reactive Scheduling in the Transportation Domain*. In: *Proceedings of the Tenth IEEE Conference on Artificial Intelligence and Applications*, San Antonio, Texas, 1994.
- [FMPJ94] K. FISCHER, J. P. MÜLLER, M. PISCHEL und E. JACOB: *Modeling Traffic Jams in a Logistics Simulation Environment*. In: *Proceedings of the European Simulation Multiconference (ESM'94)*, Barcelona, 1994.
- [LS94] A. LUX und A. SCHUPETA: *PASHA - Personal Assistant for Scheduling Appointments*. In: *Working Notes of: Workshop on Coordinated Design and Planning at Conference on Applied Artificial Intelligence (CAIA-94)*, 1994.
- [MP94] J. P. MÜLLER und M. PISCHEL: *An Architecture for Dynamically Interacting Agents*. In: *Proceedings of the 2nd International Conference on Cooperative Information Systems (CoopIS-94)*, University of Toronto Press, 1994.
- [Mül94] J. P. MÜLLER: *A Conceptual Model of Agent Interaction*. In: *Proceedings of CKBS'94*, University of Keele, England, 1994.
- [OF94] C. ODENDAHL und K. FISCHER: *Using Genetic Algorithms to Optimise the Behaviour of Agents*. In: *Draft Proceedings of the 2nd International Working Conference on Cooperating Knowledge-Based Systems (CKBS'94)*, Keele, England, 1994.
- [Sch94] A. SCHUPETA: *Interactive Decisions in COSMA-Negotiations*. In: *Working Notes of the AAAI 1994 Spring Symposium Series*, Decision-Theoretic Planning. AAAI Press, Menlo Park, 1994.

#### **1.4 Sonstige Ereignisse**

In Kooperation mit der Daimler-Benz AG wurde das Speditionsszenario reimplementiert. Bereits 1992 war zusammen mit der Daimler-Benz AG eine Spezifikation der Agenten im Speditionsszenario erstellt worden, die nun in einer Neuimplementierung in einem Forschungsprototypen umgesetzt wurde.

## 2 AKA-TACOS: Taxonomien und Alltagswissen

### 2.1 Kurzüberblick und Projektfortschritt

Das Projekt TACOS (BMBF Förderkennzeichen ITW 9201) hat die für 1994 angestrebten Ziele, logik-basierte Intergration von terminologischen Formalismen mit Formalismen zur Repräsentation von nicht-monotoner Information, Zeitwissen, unsicherer Information, und Wissen über Wissen, erreicht. Die Ergebnisse (Meilensteine) sind in DFKI-Reports und Konferenz-Beiträgen dokumentiert (siehe Abschnitt 3). Pilot-Implementierungen der entwickelten Algorithmen wurden erstellt. Ein Demonstrator-System für menu-gesteuerte Auftragsakquisition im Speditionsszenario wurde auf der Basis des Systems KRIS implementiert und beim Treffen des wissenschaftlichen Beirats der DFKI GmbH im Frühjahr 1994 vorgestellt. Danach wurde das System erweitert und seine breitere Einsatzmöglichkeit für weitere Anwendungsdomänen demonstriert. Entsprechend den Empfehlungen des Wissenschaftlichen Beirats wurden im zweiten Halbjahr 1994 verstärkt mögliche Anwendungen der bisherigen Ergebnisse evaluiert: Zusammenhänge zu objekt-orientierten Modellierungsformalismen und Einsatzmöglichkeiten im Bereich *Information Extraction*.

### 2.2 Wissenschaftliche Ergebnisse

#### 2.2.1 Modallogische Erweiterungen von terminologischen Logiken

Die Untersuchungen zur Kombination von Modallogik und terminologischer Logik wurden weitergeführt. In den bisher entwickelten Ansätzen zur Repräsentation von Wissen und Glauben wurden Modaloperatoren nur direkt vor assertionalen und terminologischen Axiomen erlaubt; weiterhin wurden keine Interaktionen zwischen diesen Operatoren berücksichtigt. Zur Beschreibung der Interaktionen von Wissen und Glauben wurde nun ein Ansatz von Sarit Kraus und Daniel Lehmann so adaptiert, daß mehrere Interaktionsmodelle verwendet werden können. Ein einfaches Beispiel hierfür ist etwa das Axiom „wenn ein Agent etwas weiß, dann glaubt er es auch“. Für diese Kombinationen konnten vollständige und korrekte Schlußfolgerungsalgorithmen entwickelt werden.

In einigen Anwendungen ist die Verwendung von Modaloperatoren nicht nur vor assertionalen und terminologischen Axiomen, sondern auch vor Konzepten notwendig. Es wurde daher eine erweiterte terminologische Sprache entwickelt, in der beide Plazierungen der Modaloperatoren möglich sind. Im Gegensatz zu einem von Franz Baader und Hans-Jürgen Ohlbach vorgeschlagenen Ansatz wird dabei die Objektdimension (also die terminologische Komponente) nicht als eine spezialisierte modallogische Dimension betrachtet. Diese Vorgehensweise hat den Vorteil, daß die zusätzlichen Dimensionen, etwa Zeit, nicht derart von den Objekten abhängig sind, daß jedes Objekt eine eigene Zeitdimension hat. Darüberhinaus hat der entwickelte Ansatz den Vorteil, daß terminologische Axiome, d.h. Konzeptdefinitionen, nicht in allen möglichen Welten gelten, sondern mittels Modaloperatoren quantifiziert werden können. Eine wichtige Frage bei dieser Art der modallogischen Erweiterung ist der Zusammenhang der Domänen in den einzelnen möglichen Welten. Diese Domänen können voneinander völlig unabhängig (*varying domain*), konstant (*constant domain*), monoton wachsend (*increasing domain*) oder monoton fallend (*decreasing domain*) sein. Die am häufigsten vorkommenden Interaktionen, *constant* und *increasing domain*, wurden algorithmisch untersucht. Während für *increasing domain* ein korrekter und vollständiger Schlußfolgerungsalgorithmus entwickelt werden konnte, stellte sich heraus, daß der *constant domain* Ansatz mit einem tableaux-basierten Erfüllbarkeitsalgorithmus nicht vernünftig zu kombinieren ist.

### 2.2.2 Das TACOS-System

Zur Modellierung der Auftragsannahme im Speditionsszenario wurde das System TACOS (Task Acquisition and COnfirmation System) entwickelt. Durch eine geeignete Menu-Führung wird die Interaktion zwischen einem Auftraggeber und dem Sachbearbeiter einer Spedition bei der Erfassung der Daten für Transportaufträge unterstützt, d.h. die Vorentscheidung des Sachbearbeiters über Annahme/Ablehnung des Auftrags aufgrund der speditionseigenen Transportmöglichkeiten und -kapazitäten simuliert. Damit soll demonstriert werden, daß ein terminologisches Wissensrepräsentationssystem, in unserem Fall das in unserer Gruppe entwickelt und implementierte KRIS-System, dazu benutzt werden kann, ein Menu-Interface intelligent zu steuern. Grundlage ist eine Taxonomie der Anwendungsdomäne, in unserem Anwendungsszenario also eine Reihe verschiedener Transport- und Umzugsarten. Der Benutzer spezifiziert seine Wünsche im Verlauf einer Programmsitzung immer genauer, wobei er vom System geführt wird. Dazu stehen ihm zwei Möglichkeiten zur Verfügung. Zum einen kann er seine Wünsche entlang der (Transport- und Umzugs-) Hierarchie spezialisieren. Zum anderen kann er Attribute der Begriffe während der Interaktion mit dem TACOS-System eingeben und verfeinern.

Die generelle Idee dabei ist, daß der Benutzer nur solche Information auswählen kann, die ihm das Menu-System anbietet. Dies bedeutet, daß er auch nur solche Information zu sehen bekommt, die in der aktuellen Situation sinnvoll ist, d.h. die konsistent ist mit dem aktuellen Wissen der TBox und ABox. Darüberhinaus ist das TACOS-System in der Lage, alle relevanten Konsequenzen, die aus der aktuellen Situation folgen, auszurechnen und anzuzeigen. Das System erkennt daher interessante Situationen durch Klassifikation der aktuellen Information und reagiert gemäß der Spezifikation durch die Modellierung, in unserem Falle durch die Verwendung von geeigneten terminologischen Regeln. Der wesentliche Vorteil unseres Ansatzes liegt darin, daß das TACOS-System extrem flexibel ist. Es ist parametrisiert durch das Modell der Anwendungsdomäne. Im Prinzip ist es daher für beliebige Domänen einsetzbar, die durch Hierarchien und ABoxen modelliert werden können. Um diese Flexibilität von TACOS zu demonstrieren, haben wir die von AT&T entworfene Wissensbasis über Weine für unser System adaptiert. Nun unterstützt das TACOS-System die Zusammenstellung von Mahlzeiten und den dazu passenden Weinen (*electronic waiter*). Die Wissensbasis enthält ungefähr 160 Klassen, 15 Attribute, 90 Objekte sowie 160 Regeln.

Realisiert wurde das TACOS-System als Client-Server Architektur. Das Interface, welches den Client darstellt, kommuniziert über einen „Remote-Procedure-Call“-Mechanismus mittels UNIX Stream Sockets mit dem KRIS-System als Server. Das Interface wurde in Oz implementiert, einer Programmiersprache, die bei der DFKI GmbH im Projekt Hydra (BMBF Förderkennzeichen ITW 9105) des Forschungsbereichs Programmiersysteme entwickelt wurde. KRIS ist in Allegro COMMON LISP implementiert. Eine erste Version (5000 Zeilen Oz-Code) wurde auf dem Treffen des Wissenschaftlichen Beirats und auf der französischen Konferenz über Wissensakquisition (JAVA'94) in Straßburg im März vorgestellt. Inzwischen ist eine zweite, erweiterte Version (7500 Zeilen Oz-Code) fertiggestellt.

Um den zusätzlichen Anforderungen aus der TACOS-Anwendung gerecht zu werden, wurden die Arbeiten am KRIS-System weitergeführt. Zu nennen ist hier (i) die Erweiterung und Optimierung der funktionalen Schnittstelle, (ii) die Integration von zusätzlichen Sprachkonstrukten zur Domänenmodellierung, (iii) die Einbindung einer Regelkomponente und (iv) die Entwicklung einer frame-artigen Modellierungssprache. Damit die vom KRIS-System berechneten Schlußfolgerungen effizient vom TACOS-System verarbeitet und somit dem Benutzer geeignet präsentiert werden können, war es notwendig, weitere Schnittstellenfunktionen für das KRIS-System zu implementieren. Es stellte sich heraus, daß die so erweiterte Funktionalität auch für



weitere Anwendungen (z.B. „kompaktes“ Informationsretrieval oder Testzwecke) sinnvoll eingesetzt werden kann. Neben der Implementierung der zusätzlichen Schnittstellenfunktionen haben wir weitere Sprachkonstrukte in die Eingabesprache von KRIS integriert. Bei der Domänenmodellierung können nun zum Beispiel Aufzählungstypen verwendet werden. Es sollte erwähnt werden, daß diese Erweiterung jedoch – aus Effizienzgründen – algorithmisch unvollständig behandelt wird. Werden die Aufzählungstypen aber eingeschränkt verwendet (wie es auch für die Domänenmodellierung des Speditionsszenarios ausreichend ist), so ist der in KRIS implementierte Algorithmus im logischen Sinne vollständig.

Es stellte sich heraus, daß die im Projekt ARC-TEC implementierte, und bereits prototypisch in das KRIS-System integrierte Regelkomponente für unsere Zwecke einige nicht-behebbar Defizite aufweist. Aus diesem Grund haben wir eine neue Regelkomponente implementiert, die mit der von KRIS angebotenen Schnittstelle optimal zusammenarbeitet. Anfangs arbeitete diese Komponente nach der sogenannten naiven Strategie, d.h. die Anwendbarkeit von Regeln wird immer gegen die gesamte aktuelle Wissensbasis abgetestet. Optimierungen dieser Strategie, sowie eine Implementierung der sogenannten semi-naiven Strategie wurden inzwischen im Rahmen einer Diplomarbeit entwickelt.

### 2.2.3 Terminologische Logiken und Objektorientierte Modellierung

Modellierungsformalisen wie sie zum Entwurf von Datenbankschemata oder zur Domänenmo-

kanonisch aus der Semantik, die terminologischen Sprachen zugrundeliegt, abzuleiten. Es ist jedoch nicht direkt möglich, die von uns entwickelten Schlußfolgerungsalgorithmen zu übertragen, da Modellierungsformalismen i.a. ausdrucksstärker sind als terminologische Wissensrepräsentationssprachen. Wir sehen daher die beiden folgenden Vorgehensweisen: (1) die Einschränkung auf eine interessante Menge von solchen Sprachkonstrukten, für die vollständige Schlußfolgerungsmechanismen möglich sind, oder (2) das Verarbeiten des vollen Sprachumfangs unter Verwendung unvollständiger Algorithmen.

#### 2.2.4 Terminologische Logiken und Datenbanksysteme

Die von uns in Zusammenarbeit mit der Gruppe von Prof. Matthias Jarke (RWTH Aachen) entwickelten Techniken zur Optimierung von Anfragen an objekt-orientierte, deduktive Datenbanken wurden auf der „International Conference on Extending Database Technology, EDBT'94“, in Cambridge, Großbritannien vorgestellt und mit Wissenschaftlern aus dem Datenbankbereich diskutiert. Daraus ergaben sich Kontakte zu Wissenschaftlern aus diesem Bereich, die durch Einladungen an die DFKI GmbH intensiviert wurden. Der Vorschlag einer neuen Architektur für terminologische Systeme, der aus den Arbeiten im Datenbankbereich und der weiter unten beschriebenen Anwendungsstudie resultiert, wurde auf Workshops und bei eingeladenen Vorträgen vorgestellt und stieß auf großes Interesse. Der Vorschlag sieht vor, zwischen einer Schema und einer View-Komponente in einem terminologischen System zu unterscheiden. Für Klassenbeschreibungen in den beiden Komponenten sind unterschiedliche Sprachkonstrukte und Semantiken vorgesehen. Der Ansatz wurde in der Zwischenzeit dahin verfeinert, daß Bedingungen an Schemata identifiziert wurden, die eine für praktische Fälle ausreichende Ausdrucksstärke erlauben, aber dennoch effiziente Inferenzen zulassen.

Weiterhin wurde der mögliche Einsatz terminologischer Logiken im Kontext von relationalen Datenbanken untersucht. An einigen kleinen Fallbeispielen haben wir gezeigt, daß mit Hilfe einer universellen terminologischen Logik tatsächlich nichttriviale Anfragen an relationale Datenbanken formuliert werden können. Ein solches Fallbeispiel betrifft die in der Künstlichen Intelligenz häufig verwendeten UND-ODER-Graphen und insbesondere deren wohlfundierte Lösungen. Zudem haben wir einen polynomialen Algorithmus zur effizienten Auswertung solcher Anfragen entwickelt, der nicht nur rekursive Konzeptdefinitionen, sondern auch rekursive Rollendefinitionen behandeln kann.

#### 2.2.5 Terminologische Logiken und Information Extraction

Eine der ursprünglichen Motivationen für die Entwicklung terminologischer Wissensrepräsentationsformalismen war die Modellierung von Domänenwissen in natürlichsprachlichen Systemen. In Zusammenarbeit mit dem DFKI-Forschungsbereich Computerlinguistik wird derzeit untersucht, inwieweit die bisher entwickelten Formalismen und Systeme dafür eingesetzt werden können. Dabei geht es insbesondere um Techniken aus den Bereichen *Information Retrieval*, *Information Filtering*, *Information Classification* und *Message Extraction*. Die ersten beiden Techniken zielen im wesentlichen darauf ab, den aktuellen Informationsbedarf eines einzelnen Benutzers (bzw. periodisch den vordefinierten Informationsbedarf mehrerer Benutzer) zu befriedigen. Für diesen Typus von Anwendungen wurden bisher statistische Verfahren (etwa Häufigkeit des Auftretens bestimmter Schlüsselwörter) angewendet, die im praktischen Einsatz aber schnell an Grenzen stoßen.

Durch die Verwendung deklarativer Wissensrepräsentationsformalismen besteht die Möglichkeit, sowohl Robustheit wie auch Genauigkeit solcher Systeme zu verbessern. Bei Information

Classification geht es im wesentlichen darum, Texte in eine vorgegebene Hierarchie einzusortieren. Hier gibt es daher einen starken Bezug zu terminologischen Klassifikationsverfahren.

Bei der Message Extraction können (terminologische) Wissensrepräsentationsformalisten sogar eine noch stärkere Rolle spielen: Ziel bei diesem Ansatz ist es, die in einem Text vorhandene Information zu extrahieren, d.h. zu erkennen und in einem geeigneten vordefinierten Format abzuspeichern. Um die relevante von der irrelevanten Information trennen zu können, muß das System Wissen über eine oder mehrere konkrete Problemomänen besitzen. Wie und in welchem Umfang solches Wissen repräsentiert werden kann, soll in Zusammenarbeit mit Wissenschaftlern des Bereiches Computerlinguistik erarbeitet werden. Ein erster Schritt in diese Richtung war die gemeinsame Entwicklung eines Message Extraction Demonstrators für ein Verkehrsszenario. Dazu wurde dazu Wissen über Verkehrsunfälle mit Hilfe des in unserer Gruppe entwickelten KRIS-Systems strukturiert und abgespeichert, sowie eine prototypische Benutzeroberfläche implementiert. Aufbauend auf dieser Wissensbasis können einfache Texte in vorgegebene Konzepte einklassifiziert werden.

### 2.2.6 Sonstige

In Zusammenarbeit mit dem vom BMBF geförderten Projekt AKA-MOD (BMBF Förderkennzeichen ITW 9104) haben wir an der Entwicklung einer objektorientierten Wissensrepräsentationskomponente für die Multiagentenszenarien Spedition und Loading Dock gearbeitet. Diese Komponente umfaßt sowohl Elemente terminologischer Sprachen wie Klassen-, Relationen- und Attributdeklarationen, Hierarchien, Vererbung und Defaults, als auch Elemente vorwärtsschließender Regelformalismen, die an die Programmiersprache OPS5 angelehnt sind. Die Komponente wird in Oz implementiert werden und als Basis für die Reimplementierungen der Szenarien in Oz dienen.

Die Arbeit an der im letzten Jahr begonnenen Studie über Anwendungen terminologischer Wissensrepräsentationssysteme wurde fortgesetzt. Die Literatursammlung wurde vervollständigt. Auf dem „International Workshop on Description Logic“, der im Mai in Bonn stattfand, wurde das weitere Vorgehen mit der amerikanischen Mitautorin der Studie besprochen. Außerdem wurde die Gelegenheit genutzt, um den anwesenden Anwendern terminologischer Wissensrepräsentationssysteme einen Fragebogen zu ihren Erfahrungen mit den verwendeten Systemen auszuhändigen. Aus der Auswertung dieser Fragebogen erwarten wir weitere Aufschlüsse über den Einsatz terminologischer Systeme. Ein Anwender aus dem Bereich „Natural Language Processing“, Dr. Enrico Franconi vom IRST Trento in Italien, war im Anschluß an den Description Logic Workshop für zwei Tage zu Gast bei der DFKI GmbH.

## 2.3 Publikationen

- [AJM<sup>+</sup>94] K. APT, M. JARKE, E. MARCHIORI, W. NUTT, D. DE SCHREYE und F. TURINI (Herausgeber): *Esprit Basic Research Action Compulog II: Deliverables for Year 2*. CWI, Amsterdam, 1994.
- [BB94] W. BUNTINE und H.-J. BÜRCKERT: *On Solving Equations and Disequations*. Journal of the ACM, Vol. 41 (4), 1994.
- [BBH94] F. BAADER, M. BUCHHEIT und B. HOLLUNDER: *Cardinality restrictions on concepts*. In: *Proc. of the German Conference on Artificial Intelligence (KI-94)*, Lecture Notes in Artificial Intelligence, Springer-Verlag, Berlin, 1994.

- [BBJN94] F. BAADER, M. BUCHHEIT, M. A. JEUSFELD und NUTT, W. (Herausgeber): *Working Notes of the KI'94 Workshop: KRDB-94: Reasoning about Structured Objects - Knowledge Representation Meets Databases*, Saarbrücken, Germany, 1994.
- [BDNS94a] M. BUCHHEIT, F. M. DONINI, W. NUTT und SCHAERF, A.: *Refining the Structure of Terminological Systems: Terminology = Schema + Views*. In: *Proc. of the 12th National Conference on Artificial Intelligence*, Seattle, USA, 1994.
- [BDNS94b] M. BUCHHEIT, F. M. DONINI, W. NUTT und SCHAERF, A.: *Terminological Systems Revisited: Terminology = Schema + Views*. In: [BBJN94].
- [BDNS94c] M. BUCHHEIT, F. M. DONINI, W. NUTT und SCHAERF, A.: *Terminological Systems Revisited: Terminology = Schema + Views*. In: J. KUNZE und H. STOYAN (Herausgeber): *Workshop Proc. of the 18th German Annual Conference on Artificial Intelligence, KI-94*, Saarbrücken, Germany, 1994.
- [BDNS94d] M. BUCHHEIT, F. DONINI, W. NUTT und SCHAERF, A.: *Refining the Structure of Terminological Systems*. In: [BLNPS94].
- [BFH<sup>+</sup>94] F. BAADER, E. FRANCONI, R. HOLLUNDER, R. NEBEL und H.-J. PROFITLICH

---

*An Empirical Analysis of Optimization Techniques for Terminological Knowledge Representation Systems, or: Making KRIS get a Move on.* Applied Intelligence, Special Issue on Knowledge Base Systems, Vol. 4, 1994.

- [BGL94] H.-J. BÜRCKERT, A. GRÄBER und A. LAUX: *Terminological Reasoning with Knowledge and Belief*. In: J. KUNZE und H. STOYAN (Herausgeber): *Workshop Proc. of the 18th German Annual Conference on Artificial Intelligence, KI-94*, Saarbrücken, Germany, 1994.
- [BH94] F. BAADER und B. HOLLUNDER: *Computing extensions of terminological default theories*. In: G. LAKEMEYER und B. NEBEL (Herausgeber): *Foundations of Knowledge Representation and Reasoning*, Lecture Notes in Artificial Intelligence 810, Springer-Verlag, Berlin, 1994.
- [BHL94] H.-J. BÜRCKERT, B. HOLLUNDER und A. LAUX: *Concept logics with function symbols*. In: A. COHN (Herausgeber): *Proc. of the 11th European Conference on Artificial Intelligence*, Amsterdam, 1994.
- [BJNS94a] M. BUCHHEIT, M. JEUSFELD, W. NUTT und STAUDT, M.: *Subsumption between Queries to Object-Oriented Databases*. In: M. JARKE, J. BUBENKO und JEFFREY, K. (Herausgeber): *Proc. of the 4th International Conference on Extending Database Technology, EDBT'94*, Lecture Notes in Computer Science 779. Springer-Verlag, Berlin, 1994.
- [BJNS94b] M. BUCHHEIT, M. JEUSFELD, W. NUTT und STAUDT, M.: *Subsumption between Queries to Object-Oriented Databases*. Information Systems, Vol. 19 (1), 1994.
- [BKN94] M. BUCHHEIT, R. KLEIN und W. NUTT: *Configuration as Model Construction: The Constructive Problem Solving Approach*. In: *Proc. of the 3rd International*

- [BLNPS94] F. BAADER, M. LENZERINI, W. NUTT und PATEL-SCHNEIDER, P.-F. (Herausgeber): *Proc. of the International Workshop on Description Logics*, Saarbrücken, Germany, 1994.
- [Bür94a] H.-J. BÜRCKERT: *A Resolution Principle for Constrained Logics*. Artificial Intelligence, 66, 1994.
- [Bür94b] H.-J. BÜRCKERT: *Terminologies and Rules*. In: K. v LUCK und H. MARBURGER (Herausgeber): *Management and Processing of Complex Data Structures, 3rd Workshop on Information Systems and Artificial Intelligence*, Lecture Notes in Computer Science 777, Springer-Verlag, Berlin, 1994.
- [DLN<sup>+</sup>94] F. DONINI, M. LENZERINI, D. NARDI, W. NUTT und A. SCHAERF: *Queries, Rules and Definitions as Epistemic Statements in Concept Languages*. In: G. LAKEMEYER und B. NEBEL (Herausgeber): *The Theoretical Foundations of Knowledge Representation and Reasoning*, Lecture Notes in Artificial Intelligence 810. Springer-Verlag, Berlin, 1994.
- [Hol94b] B. HOLLUNDER: *An Alternative Proof Method for Possibilistic Logic and its Application to Terminological Logics*. In: *Proc. of the 10th Conference on Uncertainty in Artificial Intelligence*, Seattle, USA, 1994.
- [Lau94a] A. LAUX: *Beliefs in Multi-Agent Worlds: A Terminological Logics Approach*. In: A. COHN (Herausgeber): *Proc. of the 11th European Conference on Artificial Intelligence*, Amsterdam, 1994.
- [Lau94b] A. LAUX: *Integrating a Modal Logic of Knowledge into Terminological Logics*. In: H. GEFFNER (Herausgeber): *Proc. of the 4th Iberoamerican Congress on Artificial Intelligence*, Caracas, Venezuela, 1994.
- [NB94] B. NEBEL und H.-J. BÜRCKERT: *Managing Qualitative Temporal Information: Expressiveness vs. Complexity*. In: K. v LUCK und H. MARBURGER (Herausgeber): *Management and Processing of Complex Data Structures, 3rd Workshop on Information Systems and Artificial Intelligence*, Lecture Notes in Computer Science 777. Springer-Verlag, Berlin, 1994.
- [NPS94] W. NUTT und P. PATEL-SCHNEIDER: *Useful Defaults in Description Logics*. In: [BLNPS94].
- [Sch94a] K. SCHILD: *Terminological Cycles and the Propositional  $\mu$ -calculus*. In: *Proc. of the 4th International Conference on Principles of Knowledge Representation and Reasoning, KR'94*, Bonn, Germany, 1994.
- [Sch94b] K. SCHILD: *Tractable Reasoning in a Universal Description Logic*. In: [BLNPS94].
- [Sch94c] K. SCHILD: *Tractable Reasoning in a Universal Description Logic*. In: [BBJN94].
- [Sch94d] K. SCHILD: *A Universal Description Logic as a Tractable Query Language for Databases*. In: J. KUNZE und H. STOYAN (Herausgeber): *Workshop Proc. of the 18th German Annual Conference on Artificial Intelligence, KI-94*, Saarbrücken, Germany, 1994.

## 2.4 Sonstige Ereignisse

In Zusammenarbeit mit Prof. Franz Baader (Aachen), Prof. Maurizio Lenzerini (Rom) und Dr. Peter F. Patel-Schneider (Bell-Labs, USA) organisierten TACOS-Mitarbeiter den internationalen Workshop on Description Logics (DL-94) in Bonn. Auf diesem Workshop trafen sich die wichtigsten Forscher auf dem Gebiet terminologischer Logiken. Schwerpunkte der Diskussion waren Anwendungen in Datenbanken, in der Sprachverarbeitung, Expertensystemen und Tutor-systemen, sowie Spracherweiterungen und Systemarchitektur. Es kristallisiert sich eine Reihe von Gebieten heraus, in denen terminologische Systeme erfolgreich angewandt werden. Dabei fällt auf, daß ein Durchbruch in große industrielle Anwendungen bislang erst in den USA gelungen ist.

In Zusammenarbeit mit dem Forschungsbereich Programmiersysteme an der DFKI GmbH und der Daimler-Benz AG haben wir ein System zur Konfiguration implementiert. Die Zusammenarbeit baut auf einem Ansatz auf, Konfigurationsprobleme in einer taxonomischen Sprache zu repräsentieren, der früher von Mitarbeitern aus TACOS und der Daimler-Benz AG erarbeitet wurde. Es wird daran gearbeitet, Spezifikationen in dieser Sprache so auf die Programmiersprache Oz abzubilden, daß Konfigurationsprobleme effizient gelöst werden können. Dabei wird vor allem die Fähigkeit des Oz-Systems zur flexiblen Abarbeitung von Constraints ausgenutzt.

Zusammen mit Mitarbeitern des Siemens-Telekooperationszentrums an der DFKI GmbH haben wir das Problem der Modellierung und Konfiguration von Telekooperationssystemen untersucht. Dieses Problem ist essentiell für die Markteinführung solcher Systeme, da für jeden Kunden, entsprechend seinen speziellen Erfordernissen, eine spezielle Variante eines Telekooperations-systems installiert werden muß. Grundlage für die Zusammenarbeit sind unsere Erfahrungen mit terminologischen Sprachen, die wie oben beschrieben eng mit Formalismen zur Software-Modellierung verwandt sind, sowie unsere Vorarbeiten über die Repräsentation und das Lösen

## 3 COSMA: Cooperative Schedule Management Agent

### 3.1 Kurzüberblick und Projektfortschritt

Im Projekt COSMA (BMBF-Förderkennzeichen ITW 9402) wird ein natürlichsprachliches Dialogsystem entwickelt, das für maschinelle Terminplanung zwischen mehreren Teilnehmern verwendet wird. COSMA kann als automatischer Sekretariatsassistent zusammen mit maschinellen Terminplanungsagenten Termine selbständig vereinbaren und verwalten. Die Terminabsprachen zwischen maschinellen und menschlichen Dialogpartnern erfolgen mithilfe elektronischer Post (e-mail) in natürlicher Sprache (deutsch).

Die Projektziele wurden gemäß den Auflagen des BMBF teilweise revidiert und in verschiedenen Punkten konkretisiert. Die Ergebnisse wurden dem BMBF als Ergänzungen zur Projektbewilligung schriftlich vorgelegt.

Die im ersten Meilenstein definierte Version eines natürlichsprachlichen Dialogservers wurde konzipiert und aufbauend auf Software aus Vorgängerprojekten bzw. dem Partnerprojekt PARADICE teilweise implementiert. Neue Komponenten zur Dialogsteuerung, zur Kontrolle und zur Generierung sowie eine allgemeine Schnittstelle zu Anwendungssystemen wurden spezifiziert.

COSMA arbeitet mit den Projekten CoMMA-Plat (Auftrag der Siemens AG) und dem vom BMBF geförderten Projekt AKA-Mod (Förderkennzeichen ITW 9104) eng zusammen. Die Integration menschlicher Kommunikationspartner in ein Netz autonomer Software-Agenten für die Terminplanungs-Anwendung wurde definiert.

### 3.2 Wissenschaftliche Ergebnisse

Die Projektziele wurden gemäß Auflagen des BMBF teilweise revidiert und in verschiedenen Punkten konkretisiert. Die Ergebnisse wurden dem BMBF als Ergänzungen zur Projektbewilligung im Juni 1994 schriftlich vorgelegt. Notwendig wurden die Änderungen vor allem infolge der um drei Monate verkürzten Projektlaufzeit. Die Änderungen erforderten eine Neuabstimmung mit dem vom BMBF geförderten Projekt PARADICE (Förderkennzeichen ITW 9402), die im

Berichtszeitraum durchgeführt wurde. Die Empfehlungen des wissenschaftlichen Beirats wurden dabei berücksichtigt.

#### 3.2.1 Systemarchitektur und Erweiterung von Systemkomponenten

Die Anwendungsdomäne Terminplanung wird ebenfalls in den DFKI-Projekten CoMMA-Plat (Auftrag der Siemens AG) und dem vom BMBF geförderten Projekt AKA-Mod verwendet. In diesen Projekten kooperieren maschinelle Agenten bei der Vereinbarung von Terminen für ihre jeweiligen (menschlichen) Benutzer. Ausführliche Gespräche ergaben eine ausreichende gemeinsame Grundlage auf theoretischer Ebene (hinsichtlich der Verwendung von Sprechakten, Planungsverfahren und deklarativen Repräsentationen), so daß eine intensive Zusammenarbeit vereinbart wurde. Gemeinsames Ziel dieser Zusammenarbeit ist die Ausstattung der maschinellen Agenten mit natürlichsprachlichen Dialogfähigkeiten. Dies erfordert eine geeignete Kopplung der in den jeweiligen Projekten zu entwickelnden Systeme. Im Berichtszeitraum wurden verschiedene Modelle für eine gemeinsame Architektur diskutiert. Dabei steht die Sichtweise ei-

des Anwendungssystems abgedeckt werden. Das COSMA-System kann somit ohne Verlust an Funktionalität einfacher konzipiert werden. So braucht das bisher verwendete Wissensrepräsentationssystem Rhet nicht in das zum ersten Meilenstein fertigzustellende System integriert zu werden.

Die Integration menschlicher Kommunikationspartner in ein Netz autonomer Software-Agenten bei der verteilten Terminplanung stellt eine potentielle Fehlerquelle in einem ansonsten theoretisch fehlerfreien Kommunikationsablauf dar. Aufgabe eines natürlichsprachlichen Servers

ist daher in dieser Anwendung, neben den normalen Dialoginteraktionen in natürlicher Sprache auch die Zuverlässigkeit von Dialogschritten gegenüber den Client-Systemen zu garantieren, d.h. mögliche fehlerhafte oder inkonsistente Eingaben mittels Klärungsdialogen mit dem menschlichen Dialogpartner zu korrigieren.

Klärungsdialoge sollen bei Auftreten unterschiedlicher Fehlschläge geführt werden können. Daher sind Kontrollinteraktionen zwischen einzelnen Komponenten erforderlich, so daß strikt auf die Verwendung einer einheitlichen Schnittstellensprache zwischen den Komponenten geachtet wurde. Hierfür werden getypte Merkmalsstrukturen verwendet, die bereits in vielen vorhandenen Komponenten benutzt werden. Zu ihrer Verarbeitung wird der im Vorgängerprojekt DISCO entwickelte Unifikator UDiNe eingesetzt. Für Komponenten, die intern nicht mit getypten Merkmalsstrukturen arbeiten, mußten Schnittstellen zu ihren internen Repräsentationen definiert werden.

Verschiedene wiederverwendbare Komponenten von Vorgängersystemen wurden bereits auf ihren Einsatz in COSMA hin erweitert. Das neue (reduzierte) Pragmatikmodul hat eine Schnittstelle zu getypten Merkmalsstrukturen, wie sie in der Grammatikverarbeitung benutzt werden. Somit werden einige Übersetzungsschritte zwischen verschiedenen Formalismen vermieden, die sich im Vorgängerprojekt DISCO als problematisch erwiesen haben. Das Diskursgedächtnis und die Komponente zur Resolution anaphorischer Bezüge wurden an einige Anforderungen von parallel zu führenden Mehrteilnehmer-Dialogen angepaßt. So werden Diskursobjekte klassifiziert nach den Sätzen, in denen sie erwähnt wurden, nach dem Sprecher und nach dem Diskurs, in dem die Sätze geäußert wurden. Funktional wurde die Komponente um Referenzauflösung auf der Basis von Oberbegriffen erweitert (z.B. kann "der Tag" auf einen zuvor erwähnten "Freitag" bezogen werden).

### 3.2.2 Dialogsteuerung

Serversysteme, die natürlichsprachliche Dialogleistung über verschiedene Anwendungsbereiche anbieten, müssen anwendungsunabhängige Dialogstrategien und Dialogsteuerungsmodelle enthalten. In COSMA wurde daher ein Computermodell des in der Literatur über Diskursanalyse als *dialogue grounding* bekannten Kommunikationsprozesses erarbeitet. In diesem Prozeß geben die Dialogteilnehmer einander aktiv Feedback, so daß gemeinsame Überzeugungen hinsichtlich der Erfolge und Fehlschläge einzelner Dialogschritte entstehen. Wie notwendig dieser Prozeß für erfolgreiche Kommunikation ist, läßt sich an folgendem negativen Beispiel mit einem fehlerhaften Eingabewort (z.B. "Donstag") veranschaulichen:



*Benutzer:* Lösche den Termin am Donstag!

*a) System:* Wie bitte?

*b) System:* Donstag?

Anstatt nun eine Menge domänenspezifischer Heuristiken zu definieren, die solche Probleme behandeln würden, legen wir die linguistischen Grundlagen für eine weitaus allgemeinere Behandlung, die auch in anderen Anwendungsbereichen gültig bleibt. Das Modell des Grounding-Prozesses, das von D. Traum und E. Hinkelman vorgeschlagen wurde, beruht auf endlichen Automaten, deren Übergänge als Dialog-Kontroll-Aktionen definiert sind (z.B. Klärungsanforderung, Klärung, Bestätigung, Beendigung), die als Grundlage für klassische Sprechakte (mitteilen, anfordern, fragen) dienen.

Neben diesen theoretischen Grundlagen wurden im Berichtszeitraum die folgenden praktischen Ergebnisse erzielt:

- Entwicklung einer Repräsentation endlicher Automaten in getypten Merkmalsformalimen (basierend auf einer Arbeit von H.-U. Krieger);
- darauf aufbauend Kodierung eines Automaten für Dialogkontrolle nach Arbeiten von Traum;
- Spezifikation der Schnittstelle zwischen dem Automaten und Anwendungsprogrammen als getypte Merkmalsstrukturen:

- Die Anwendung sollte das natürlichsprachliche System als Server benutzen können, anstatt eng auf Eigenschaften seiner Software oder Hardware abgestimmt zu sein.
- Die Anwendung sollte nicht mit Einzelheiten der linguistischen Kontrollstrategien oder der Modularisierung der Software in Berührung kommen.
- Die Auswirkungen von Upgrades des Servers auf die Anwendung sollten unbedingt minimiert werden.
- Die Form der Schnittstelle sollte so einfach wie möglich gehalten werden und nicht von den benötigten Dienstleistungen abhängen.
- Daten sollten in einem neutralen Format kodiert werden und nicht auf eine bestimmte Anwendung oder eine bestimmte Repräsentationsebene zugeschnitten sein.
- Die gewählten Repräsentationen sollten effizient sein.
- Client-Systeme sollten durch Software darin unterstützt werden, Daten für den Server zu kodieren und zu dekodieren.
- Die Kommunikation zwischen Server und Client sollte gute Werkzeuge zur Fehlerkorrektur unterstützen.

Diese Forderungen wurden in COSMA auf der Grundlage der folgenden, neuen Idee umgesetzt: Die Schnittstelle soll deklarativ sein, indem sie keine Kommandos zwischen Client und Server austauscht, sondern (partielle) Beschreibungen der jeweils vorhandenen sowie der gewünschten Information.

Als grundlegender Kommunikationskanal dienen TCP/IP-Verbindungen zwischen den Prozessen. Die Schnittstellensprache basiert auf der externen Darstellung von Merkmalsstrukturen

OPTIONS:	<i>options</i>
SEQUENCE:	[PREV: { <i>sequence</i> , NULL} THIS: <i>tag</i> ]
CONTEXT:	[UTTERER: <i>agent</i> GROUP: <i>agent</i> N-PARTICIPANTS: { <i>nat</i> , SEVERAL, MANY} TIME: <i>time</i> ]
TEXT:	[ENCODING: {ASCII, ...} VALUE: <i>string</i> ]
STRUCTURED-TEXT:	<i>structured_text</i>
MORPH:	< [STEM: <i>string</i> MORPHOLOGY: <i>morph</i> ], ... >
SYN:	<i>syn</i>
SEM:	<i>sem</i>
GROUND:	<i>ground</i>
CORE-SA:	[TYPE: <i>speech_act_label</i> AGENT: <i>agent</i> INITIATOR: <i>agent</i> CONTENT: { <i>action</i> , <i>proposition</i> }]
DEBUG-QUERY:	[OPERATION: {COPY, GENERATE} OPERAND: <i>operand</i> ]
DEBUG-RESPONSE:	<i>debug_response</i>

Abbildung 1: Struktur zur Bearbeitung von Aufträgen eines Anwendungssystems.

- die Rolle von COSMA bei der Kommunikation zwischen menschlichen und maschinellen Agenten festgelegt, und
- die Repräsentation von Zeitangaben in beiden Systemen abgestimmt.

Das PASHA-System von CoMMA-Plat adaptiert bei Dialogen mit mehreren maschinellen Agenten derzeit eine sternförmige Interaktionsstruktur, d.h. der Initiator hat Kontakt mit allen anderen Teilnehmern, diese jedoch nur mit dem Initiator und nicht untereinander. Im Rahmen des ersten Meilensteins wird COSMA als natürlichsprachliche Schnittstelle bei einem Teilnehmer eingesetzt, der mit dem Initiator über elektronische Post kommuniziert. COSMA kann hierbei andere Interaktionen des Initiators ignorieren.

Die in COSMA aufgrund eines Textes identifizierten Sprechakte werden innerhalb von PASHA in Domänenaktionen übersetzt. So wird z.B. eine "Anforderung" zu Beginn eines Dialogs als Terminvorschlag (*propose*) interpretiert. Die Terminbeschreibungen können je nach Eingabe unterspezifiziert sein (z.B. *am 12. nachmittags*). Die Rekonstruktion der in PASHA bevorzugten vollständigen Spezifikation erfolgt in Verantwortung des Client-Systems, das entsprechende Fragen an den im Server gespeicherten linguistischen Kontext richten kann.

Umgekehrt spezifiziert PASHA stets vollständige Terminbeschreibungen, die dann im Server entweder vollständig in natürlicher Sprache ausgedrückt oder nach Maßgabe des linguistischen Kontexts verkürzt werden können.

### 3.3 Publikationen

- [HS94] E.HINKELMAN und S.SPACMAN: *Communicating with Multiple Agents*. In: *Proceedings of the 15th International Conference on Computational Linguistics, COLING-94*, Kyoto, Japan, 1994.

### 3.4 Sonstige Ereignisse

DFKI GmbH ist Partner im Projekt TAMIC (Transparent Access to Multiple Information for the Citizen), das von der EG im Rahmen von MLAP im Juni 1994 als Pilotprojekt bewilligt wurde (Partner: Quinary SpA., Mailand (I), Cap Gemini Innovation, Boulogne (F), Istituto per la Ricerca Scientifica e Tecnologica, Trento (I), DFKI (D)). Thema des Projekts ist der natürlichsprachliche Zugang zu verschiedenartigen Informationssystemen. Der Bezug zu COSMA besteht zum einen in der zentralen Frage der Schnittstelle zwischen natürlichsprachlichem System und den Technologie in beiden Projekten. TAMIC ist zunächst für ein Jahr bewilligt (Start November 1994).

## 4 IMCOD: Intelligent Manager for Comprehensive Design

### 4.1 Kurzüberblick und Projektfortschritt

Der moderne Produktentwurf stellt hohe Anforderungen an die Informationsverarbeitung: Um den stets wachsenden Anforderungen an neue Produkte gerecht zu werden, gilt es, zum frühestmöglichen Zeitpunkt eine Fülle von Daten über den gesamten Produktlebenszyklus zu verarbeiten und vielfältige Expertise bereits beim Produktentwurf zu berücksichtigen.

Vor diesem Hintergrund entwickelt das IMCOD-Projekt (BMBF Förderkennzeichen ITW 9304/3) Verfahren zur Integration heterogener, den Entwurfsprozeß unterstützender Systeme in ein homogenes Gesamtsystem. Grundgedanke ist dabei die Systemkopplung und Kooperation unter Vermittlung und Kontrolle durch einen zentralen Manager.

Durch den Einsatz dieses Managers stellt das entstehende Gesamtsystem dem Benutzer die Expertise aller beteiligten hochspezialisierten Systeme, der sogenannten "lokalen Experten", in homogen integrierter Weise bereit. Die so erreichte optimale Unterstützung erlaubt unter anderem eine Beschleunigung des anfallenden Wissenstransfers, ein Verkürzen der Umlaufzeiten und die bedarfsweise direkte Rückkopplung mit den einzelnen Experten, um so nicht nur eine erhöhte Flexibilität sondern auch qualitativ bessere Resultate zu erzielen.

Im Berichtszeitraum wurden ausgehend von einem exemplarischen Anwendungsszenario sowohl die einsetzbaren Experten als auch der IMCOD-Manager näher charakterisiert. Die wesentlichen Funktionalitäten des IMCOD-Managers wurden entwickelt und prototypisch implementiert. Zentrale Ergebnisse sind im nächsten Abschnitt kurz dargestellt, gefolgt von einer Auflistung der aktuellen Publikationen. Für eine ausführliche Darstellung der Ergebnisse wird auf den Abschlussbericht des Projekts verwiesen.

Einige der Ergebnisse wurden im Berichtszeitraum im Rahmen von Studien und implementierten Prototypen industriellen Anwendungen verfügbar gemacht. erzielten Resultate.

### 4.2 Wissenschaftliche Ergebnisse

Die Arbeiten im Berichtszeitraum begannen mit der Detaillierung eines möglichen Anwendungsszenarios einer allgemeinen Spezifikation. In diesem Zusammenhang wurden Methoden zur Wissensakquisition in geeigneter Weise erweitert bzw. neu entwickelt. Anschließend wurden sowohl die Experten als auch der IMCOD-Manager in ihren jeweiligen Funktionalitäten genauer untersucht. Schwerpunkte bildeten die Aspekte Integration, Kommunikation und Semantische Unifikation, Verwaltung und Steuerung durch den Manager und Entscheidungsfindung und -unterstützung.

Anhand prototypischer Implementierungen wurden die entwickelten Konzepte evaluiert. Erste Anwendungen ermöglichen eine weitere Erprobung im industriellen Umfeld.

#### Anwendungsszenario

In Zusammenarbeit mit dem Institut für Verbundwerkstoffe GmbH (IVW) in Kaiserslautern wurde der Entwicklungsprozeß von Druckbehältern aus Faserverbundwerkstoffen als ein geeignetes Szenario für IMCOD identifiziert: Das Anwendungsgebiet reicht vom ersten Sammeln von Anforderungen über die Entwurfsphase, die Auslegungsphase bis zum abschließenden Prozeß der Layout- und Kostenbestimmung. Dieses Anwendungsszenario ist somit hinreichend komplex. Interviews und eine "Laut-Denk"-Studie zeigten, daß neben einem (Design-)Manager mindestens vier Experten an dem Entwicklungsprozeß beteiligt sind. Die Aufgaben der Experten umfassen die Konstruktion, die Berechnung, die Kostenschätzung und die Fertigungssimulation. Der Ma-

nager ist sowohl für den korrekten Gebrauch von existierenden Systemen, wie z.B. CAD- oder Simulationsprogramme, als auch für das Erfragen fehlender Information zuständig. Somit deckt sich diese Praxissituation mit dem von IMCOD vorgesehenen Design-Szenario.

### **Wissensakquisition**

Das Modell zum methodischen Konstruieren wurde in ein KADS-Modell transformiert, um eine Dokumentation der Expertise zu erhalten und gleichzeitig ein Modell für das IMCOD-System zu erstellen. Die Inferenzstruktur des KADS-Modells ist sehr komplex, so daß mehrere generische Elemente der KADS Bibliothek zur Entwicklung des Modells benötigt wurden.

Das im ARC-TEC-Projekt entwickelte Wissensakquisitionswerkzeug COKAM+ wurde erweitert, so daß es nun angepaßt werden kann. Diese erweiterte Version heißt MIKADO-KIT. Das KADS-Modell der neuen Domäne wurde in MIKADO-KIT integriert und Wissenseinheiten aus Texten erhoben. Damit ist Wissen über den Prozeßablauf in Hinblick auf den IMCOD-Manager spezifiziert. Motiviert durch die Erfordernisse der neuen Domäne wurde MIKADO-KIT um die Möglichkeit zur Definition unterschiedlicher Sichten auf das KADS-Modell erweitert. Auf diese Weise wurde die Möglichkeit geschaffen, das Wissen aus der Sicht der einzelnen lokalen Experten des IMCOD-Szenarios zu spezifizieren.

### **Integration lokaler Experten**

In einer ersten Stufe wurden die Ideen zur Integration der lokalen Experten prototypisch für die im Projekt ARC-TEC erstellten Systeme demonstriert. Im Bereich der "lokalen Experten" wurde für die Domäne von ARC-TEC eine zusätzliche Komponente realisiert, deren Expertise die Konsistenz beim Entwurf rotationssymmetrischer Teile überprüft. Da dieses Expertensystem in CLP(R) (logisches Programm und algebraische Constraints) realisiert wurde, kann man diesen lokalen Experten als Repräsentanten einer Klasse von Expertensystemen ansehen; nämlich derjenigen Expertensysteme, die mit Hilfe der logischen Programmierung (PROLOG und Erweiterungen) realisiert wurden. Expertensysteme dieser Klasse können methodisch in gleicher Weise in das IMCOD-System integriert werden.

IMCOD geht von einer Reihe von existierenden Systemen aus, die in das Gesamtsystem integriert werden sollen. Um die unterschiedlichen Terminologien dieser lokalen Experten zu integrieren, ist es erforderlich, eine semantische Unifikation zwischen ihnen zu erreichen. Unsere Untersuchung vorhandener wissenschaftlicher Arbeiten auf diesem Gebiet haben gezeigt, daß dies durch eine Abbildung der individuellen Ontologien der lokalen Experten in die Ontologie des Managers erreicht werden kann. Ontologie bezeichnet in diesem Zusammenhang die explizite Spezifikation der Konzeptualisierung der Domäne, d.h. die Semantik der benutzten Bezeichner, die in der Wissensbasis, bzw. an der Schnittstelle eines Expertensystems auftreten.

Zunächst wurde untersucht, welche Ausdrucksmächtigkeit eine Wissensrepräsentationssprache haben muß, die es erlaubt die Ontologie einer Domäne zu repräsentieren. Deswegen wurden zu Beginn von IMCOD die bereits vorhandenen Wissensbasen von ARC-TEC strukturiert und ihre Ontologien repräsentiert. Für die Ausdrucksmächtigkeit eines Repräsentationswerkzeugs und das Ziel, damit langfristig die semantischen Unifikation heterogener Informationssysteme zu erzielen, ergeben sich zwei wesentliche Forderungen:

- Eine Menge von Ontologien muß hierarchisch aufgebaut werden. Das heißt, daß man mit relativ kleinen und einfachen Ontologien beginnt und die darin benutzten Bezeichner und ihre Semantik in komplexere Ontologien "importiert".

- Bei der Definition der Semantik einer Ontologie muß man auf die Semantik einzelner Bezeichner aus anderen Ontologien zugreifen können. Damit hat man die Möglichkeit, Methoden der Theorieinterpretation, wie sie schon in den 60er Jahren in der mathematischen Logik entwickelt wurden, anzuwenden. Aufbauend auf diesen Ergebnissen wurde ein Wissensrepräsentationstool implementiert, das beim Knowledge Engineering von Ontologien Unterstützung leistet.

Innerhalb der Anwendungsdomäne wurden in zwei Teilbereichen Ontologien erstellt:

1. Eine Ontologie, die 70 Begriffe aus dem Bereich der Auslegung von Druckbehältern aus Faserverbundwerkstoff erfaßt.
2. Eine Ontologie, die den prinzipiellen Aufbau der Materialien, wie sie aus der Materialwissenschaft bekannt ist, beschreibt [BS94]. Die Ontologie wurde formal in der Sprache *Ontolingua* repräsentiert.

Das Problem der Heterogenität beim Wissensaustausch zwischen Expertensystemen wurde auf drei Ebenen untersucht:

1. unterschiedliche Syntax der Austauschformate
2. unterschiedliche Semantik der Bezeichner, die beim Austausch benutzt werden (unterschiedliche Ontologien) unterschiedliche Kommunikationsschichten.

Die Problematik der unterschiedlichen Syntax wird dabei nicht weiter untersucht, da Lösungsansätze hierfür auf die speziellen Teilsysteme bzw. lokalen Expertensysteme zugeschnitten sind und deswegen kaum in einer wissenschaftlich interessanten und wiederverwendbaren Art behandelt werden können. Damit wird vorausgesetzt, daß alle beteiligten Systeme als Schnittstellensprache (Teilmengen von) Prädikatenlogik erster Stufe (PL1) benutzen. Ist diese Voraussetzung für ein konkretes System nicht gegeben, muß ein spezieller Übersetzer von PL1 in die Schnittstellensprache des Systems definiert werden.

Im Beispiel des Entwurfs von Druckbehältern aus Faserverbundwerkstoffen wurde jedoch die gemeinsame Basis in Form eines integrierten Produktmodells realisiert.

### Unterschiedliche Ontologien

Die Problematik der unterschiedlichen Ontologien, d.h. im wesentlichen die Integration der unterschiedlichen Terminologien, die vom Design-Manager und von den lokalen Experten benutzt werden, wurde theoretisch (semantischen Unifikation) und softwaremäßig Interpretation zwischen Theorien der mathematischen Logik wird benutzt, um zwei Formeln, die in unterschiedlichen Sprachen (der des Manager und der eines lokalen Experten) ausgedrückt werden, aber semantisch das gleiche bedeuten, ineinander zu übersetzen, d.h., eine Formel in einer Sprache wird als wahr angenommen genau dann, wenn die ihr entsprechende in der anderen Sprache auch wahr ist. Aufbauend auf diesem Konzept wurde eine Softwarekomponente entwickelt, die es erlaubt, die notwendigen Bedeutungen von Bezeichnern zu spezifizieren. Es entstehen dabei Domänenontologien, die die minimale Grundlage für eine Interpretation zwischen Theorien liefern. In einem weiteren Schritt wurde dieses Tool ergänzt, so daß man ausgehend von Definitionen einer Ontologie komplexe Formeln "automatisch" in eine andere Ontologie übersetzen kann.

### **Unterschiedliche Kommunikationsschichten**

Es wurde ein Tool entwickelt, mit dem die Kommunikationsschichten aller beteiligten lokalen

Experten definiert werden. Es entsteht dabei eine Bibliothek von Kommunikationsprimitiven (*Ask, Tell, Retract, Command, Compose*), die für den Manager eine homogene Schnittstelle zu *allen* lokalen Experten darstellt, d.h., zur Laufzeit benötigt der Manager keine Kenntnisse über die individuellen Schnittstellen der Einzelexperten, sondern benutzt die Einträge in der Bibliothek, die hauptsächlich durch den tatsächlichen *Inhalt* der Frage gekennzeichnet sind.

Damit wurde die Klasse der lokalen Experten, die von dem IMCOD-Manager angebunden werden können, festgelegt: Es sind gerade die, die mit den obigen konzeptionellen und softwaremäßigen Methoden bezüglich ihres Schnittstellenverhaltens definiert werden können.

### **Ein integriertes Produktmodell für Druckbehälter aus Faserverbundwerkstoffen**

Als grundlegendes Repositorium für alle produktionsrelevanten Daten wurde ein integriertes Produktmodell für Druckbehälter aus Faserverbundwerkstoffen entworfen. Das Modell orientiert sich an bestehenden Standards, trägt aber den Eigenheiten der Domäne Rechnung. Das einzelne



entsprechender Teilgeometrien. Jede einzelne Lage wird explizit beschrieben und ermöglicht die Angabe der Materialkenngrößen. Die Lagenfolge wird (vergleichbar der Topologie) durch Ordnungselemente bestimmt.

Wichtige Grundgrößen, die den ganzen Behälter betreffen (wie etwa minimaler und maximaler Druck), sind in Form von Attribut-Wert-Paaren dem Behälter als ganzem zugeordnet (Vessel).

Da das IMCOD-Projekt die Unterstützung des Produktentwurfs zum Ziel hat, ist es notwendig, im Produktmodell die notwendigen Mittel zur Darstellung von erst partiell bestimmten Produkten bereitzuhalten. Zu diesem Zweck kann jeder Wert durch eine freie Variable ersetzt werden. Der Gültigkeitsbereich einer Variablen umfaßt das gesamte Produktmodell. Umfang

### **Verwaltung und Steuerung von Experten als Aufgabe des Managers**

Der Grundgedanke des IMCOD-Systems ist, heterogene lokale Experten so mit Hilfe des Managers zu integrieren, daß für den Benutzer eine homogene Sicht auf das Design-System entsteht.

In einem ersten Prototyp wurden die technischen Aspekte einer solchen Kopplung untersucht. Der Manager integriert das oben erwähnte System zur geometrischen Konsistenzprüfung CAD-System und den aus dem ARC-TEC-Projekt übernommenen Planungswerkzeugen. Die einzelnen Komponenten verwenden dabei unterschiedliche Hardware und abweichende Systemumgebungen, dennoch wurde eine Integration durch zentral gesteuerte Kommunikation erreicht. Die ontologischen Differenzen wurden durch die im vorigen Abschnitt dargestellten Methoden überbrückt. Der Manager steuert entsprechend der fest vorgegebenen Einsatzreihenfolge den Ablauf der verschiedenen Komponenten.

In einem zweiten Prototypen wurde diese Funktionalität des Managers erweitert. Die explizite Modellierung der beteiligten lokalen Experten ermöglicht dem Manager die dynamische Steuerung des Einsatzes anhand der für jeden Experten angegebenen Kompetenz. Das implementierte System führt eine automatische Bewertung der vom Benutzer generierten Designvorschläge durch. Basis der Bewertung ist dabei ein auftragsspezifischer Kriterienkatalog, erweitert um einschlägige Standards und Normen. Das System präsentiert sich als homogenes entscheidungsunterstützendes System, das sowohl eigenständige Bewertungen vornehmen als auch auf Benutzereingriffe mit *Was wäre wenn?*-Analysen reagieren kann.

### **Entscheidungsfindung als Aufgabe des Managers**

Eine weitere Aufgabe des Managers ist die Integration der verschiedenen Bewertungen der beteiligten lokalen Experten. Die Differenzen betreffen sowohl die Aspekte, die überhaupt bewertet werden, wie auch unterschiedliche Bewertungen eines Aspektes. Quelle dieser Differenzen sind die naturgemäß vorliegenden unterschiedlichen Präferenzen und abweichenden Kompetenzen aller Beteiligten. Da auch der Benutzer an der Entscheidungsfindung beteiligt ist, gilt der Formalisierung der oftmals vagen Benutzerwünsche ein besonderes Augenmerk.

Der entwickelte IMCOD-Prototyp realisiert ein zweistufiges Berechnungsverfahren, mit dem diesen Aufgaben Rechnung getragen wird. Die Wünsche des Benutzers werden in Kriterienkatalogen formalisiert und nach Wichtigkeit kategorisiert. Die Trennung in *unabdingbar* einerseits und *wünschenswert mit unterschiedlicher Präferenz* andererseits ist dabei von besonderer Bedeutung. Die Kompetenz der beteiligten Experten ist für jedes Kriterium separat dargestellt. Auf der Basis dieser Informationen erfolgt eine Bewertung der Designvorschläge durch die kompetenten Experten, d.h. es wird beurteilt, inwieweit die Vorschläge die Kriterien erfüllen. Unter Berücksichtigung der Präferenzen werden die Bewertungen kombiniert und die Vorschläge entsprechend priorisiert. Ergebnis ist die Identifikation des unter diesen Rahmenbedingungen besten

---

Vorschlags.

Da diese Bewertung des Vorschlags auf formal repräsentierten Kriterien und Verfahren beruht, kann das Ergebnis jederzeit eindeutig und nachvollziehbar begründet werden.

### **IMCOD als Decision Support System zur Vorauslegung von Druckbehältern**

Durch die erwähnten Verfahren trifft der IMCOD-Prototyp Entscheidungen und präsentiert sie dem Benutzer in leicht verständlicher Weise. Diese Realisierung eines wohldokumentierten Verfahrens liefert im Sinne der Vorgaben korrekte Ergebnisse.

Der globale Ablauf, die Steuerung des Experteneinsatzes und die Art der bearbeiteten Kriterien sind dabei durch das Anwendungsszenario *Vorauslegung von Druckbehältern aus Faser-verbundwerkstoffen* bestimmt: Die in einem Brainstorming-Prozeß von Menschen skizzierten Vorschläge werden einer Bewertung unterzogen, der beste Vorschlag wird weiter bearbeitet.

Der Wunsch nach Entscheidungsunterstützung im Sinne eines intelligenten Assistenten geht jedoch über diese Funktionalität hinaus: Der Assistent soll mit dem Benutzer in Dialog treten

Der IMCOD-Prototyp ermöglicht auch diese Funktion: Der Benutzer kann jederzeit die von ihm vorgegebenen Kriterien und Präferenzen modifizieren. Auch auf die Auswahl der beteiligten lokalen Experten kann er (bedingt) Einfluß nehmen. Der Manager veranlaßt die durch solche Eingriffe nötig gewordenen erneuten Bewertungen und präsentiert dem Benutzer unmit-

- [NPS94] S. NEUBERT, T. PIRLEIN und G. SCHMIDT: *Top-Down Knowledge Acquisition*. Proceedings of International Conference on Expert Systems for Development, 1994.
- [Sch94a] G. SCHMIDT: *Die Rolle des Experten im Knowledge Engineering*. In: KUNZE, J. und STOYAN, H. (Herausgeber): *KI-94 Workshops*, Gesellschaft für Informatik e.V., 1994.
- [Sch94d] G. SCHMIDT: *Wissensakquisition aus Texten: Hypertext versus sprachverstehende Systeme*. In: NEUBERT, S. und SCHMIDT, G. (Herausgeber): *Hypertext und Künstliche Intelligenz \* Aktuelle Trends 1993*, Bericht 289, Institut für Angewandte Informatik und Formale Beschreibungsfahren, Universität Karlsruhe (TH), 1994.

#### 4.4 Sonstige Ereignisse

Die bisher in IMCOD erzielten Resultate konnten im Berichtszeitraum in mehrfacher Weise konkreten Anwendungen zugänglich gemacht werden. Die entwickelten Verfahren und Werkzeuge zur Wissensakquisition fanden vielfältige Anwendung. Die Konzepte zur Kooperation unter zentralem Management fanden Beachtung im Bereich der Unterstützung räumlich verteilter Teams.

## 5 HYDRA: Hybride Werkbank zur Konstruktion von deduktiven Problemlösern in wissensbasierten Systemen: Berech-

### 5.1 Kurzüberblick und Projektfortschritt

Hydra (BMBF Förderkennzeichen ITW 9105) entwickelt die KI-Programmiersprache Oz, die weltweit als erste Sprache mächtige Problemlöseverfahren, wie sie bisher nur im Constraint Logic Programming zu finden waren, mit der für die verteilte KI entscheidenden Fähigkeit, multiple Agenten auszudrücken, vereint.

Zum Ende des Berichtszeitraumes erfolgte die Freigabe von DFKI Oz 1.0, der von Hydra entwickelten interaktiven Implementierung von Oz. Anwendungsbeispiele in den Bereichen Multiagenten-Programmierung, nebenläufige Systeme, Scheduling- und Optimierungsprobleme sowie Echtzeitanwendungen demonstrieren das Anwendungspotential von Oz.

### 5.2 Wissenschaftliche Ergebnisse

Im Berichtszeitraum konzentrierten sich die Arbeiten auf die Veröffentlichung von DFKI Oz 1.0. Nachdem eine Vorversion ab September 1994 freigegeben wurde, konnte die offizielle Freigabe von DFKI Oz 1.0 am 23. Januar 1995 erfolgen.

#### 5.2.1 Freigabe von DFKI Oz 1.0

DFKI Oz ist die erste Implementierung der in Hydra entwickelten Programmiersprache Oz. Oz wurde entwickelt für Anwendungen, die komplexe symbolische Berechnungen, Organisation in kooperierende Agenten und Echtzeitkontrolle erfordern. Es basiert auf einem neuen Berechnungsmodell, das eine einheitliche Grundlage für funktionale Programmierung, constraintbasierte logische Programmierung und nebenläufige Objekte mit mehrfacher Vererbung bietet. Das System umfaßt eine auf GNU Emacs basierende interaktive Programmierumgebung, eine objektorientierte Schnittstelle zu dem Graphiksystem Tcl/Tk, einen inkrementellen Übersetzer und ein Laufzeitsystem mit Speicherbereinigung. DFKI Oz gliedert sich in die offene Softwarearchitektur von UNIX (Sockets, Anschluß von C und C++) ein. DFKI Oz zeigt eine mit kommerziellen Lisp- und Prolog-Systemen vergleichbare Performanz. Mitgeliefert werden Anwendungsbeispiele für Multiagenten-Programmierung, Programmierung nebenläufiger Systeme, graphische Benutzeroberflächen, sowie Lösungen für Scheduling- und Optimierungsprobleme und Echtzeitanwendungen. DFKI Oz ist für die wichtigsten UNIX-basierten Plattformen verfügbar: Sparc Architekturen (SunOS und Solaris), DEC (Ultrix), Hewlett-Packard (HP-UX), Intel 386/486/586 (Linux), IBM (AIX), sowie Silicon Graphics (IRIX).

Die Dokumentation zu Oz und DFKI Oz umfaßt momentan ca. 630 Seiten. Neben einer Übersicht stehen ein Tutorial sowie Beschreibungen der Constraintprogrammierung in Oz, der graphischen Programmierung in Oz, der in Oz realisierten Programmierparadigmen, der offe-

zusammen, ca. 220 mal kopiert. Die Übersicht des Oz-Systems wurde 580 mal angefordert, das Tutorial 320 mal. Die meisten Interessenten stammen aus Europa und den USA (wobei fast die Hälfte der US-amerikanischen Interessenten aus dem kommerziellen Bereich stammen—für Europa ist der Anteil der kommerziellen Interessenten leider nicht zu ermitteln), aber auch Australien, Japan und die jungen High-Tech Länder wie Singapur, Indien, Malaysia und Südkorea sind vertreten. Diskussionen über die Sprache Oz, das System und mögliche Anwendungen werden auf einer internationalen Mailing List geführt.

### 5.2.2 Weiterentwicklung von Oz

Eine wesentliche Verfeinerung des Sprachentwurfes ist die Beschreibung der Abarbeitungsstrategie mit Hilfe von Threads. Die Aktoren einer Oz-Berechnung werden in Threads gruppiert, denen gemäß ihrer Priorität von einem Scheduler Rechenzeit zugeteilt wird. Die Priorität der Threads kann vom Programm verändert werden. Innerhalb eines Threads haben sogenannte eigentliche Aktoren, die bereits zur Abarbeitung vorbereitet sind, Priorität während Elaboratoren gemäß

können "Externe Ausdrücke" benutzt werden, um Operationen auszuführen, die den Datenfluß des Zustandes in dieser Methode nicht beeinflussen.

Die Kommunikation in Oz wurde von Kanälen auf Zellen umgestellt. Zellen können an genau eine Variable gebunden sein, diese Bindung kann dann im Programmablauf durch neue Bindungen ersetzt werden. Dieses Modell ist konzeptionell bedeutend einfacher und leichter zu realisieren als das alte Kanal-Modell. Mithilfe dieses neuen Konzeptes wurden die Objekte erheblich vereinfacht. Objekte sind nun Prozeduren, die Zugriff auf eine Zelle haben, und die durch vererbte Information angereichert sind. Das Versenden von Nachrichten ist nun realisiert als die Applikation von Objekten auf Daten. Mithilfe von Zellen wurde auch das Konzept von Autonomen Objekten revidiert.

Eine wichtige Weiterentwicklung des Oz-Systems betrifft das Graphikmodul von Oz. Das bisher verwendete, auf Interviews basierende Graphiksystem hat sich als zu langsam und inflexibel erwiesen. Wir haben uns dazu entschlossen, ein verbessertes Graphiksystem auf der Basis von Tcl/Tk zu entwickeln. Dabei handelt es sich um ein konzeptionell einfaches, aber sehr mächtiges Graphikpaket, das von Prof. Ousterhout (Berkeley) entwickelt wurde. Es ist abzusehen, daß Interviews von Tcl/Tk als Standard Graphikpaket abgelöst wird. Wir haben eine erste Testversion eines Tcl/Tk basierten Graphiksystems für Oz entwickelt.

Das Versenden von Nachrichten und die Applikation von Methoden wird vom Übersetzer optimiert, wenn das Objekt zur Laufzeit bekannt ist und wenn die Nachricht zur Übersetzungszeit bekannt ist. In diesem Fall ist das gesamte Protokoll in C++ abgesenkt. Mit dieser Optimierung für den in der Praxis wichtigsten Spezialfall der Methodenapplikation wird diese nun genauso schnell wie die Prozedurapplikation.

Des weiteren wurden suspendierende Builtins realisiert. Dies bedeutet, daß die Builtins nun ihr Verhalten im Suspensionsfall direkt spezifizieren können. Diese Funktionalität mußte bisher durch eigene Oz-Prozeduren simuliert werden. Durch die direkte Realisierung suspendierender Builtins konnte die Laufzeit und der Speicherplatzverbrauch von Builtins insbesondere bei den endlichen Bereichen deutlich verbessert werden. Darüber hinaus kann nun der Übersetzer Aufrufe an diese Builtins deutlich besser optimieren (z.B. bei flachen Guards), da nun deren Verhalten auch im Suspensionsfall bereits zur Übersetzungszeit bekannt ist.

### 5.2.3 Wissenschaftliche Kontakte

Innerhalb der DFKI GmbH besteht eine intensive Zusammenarbeit mit dem Esprit-Projekt ACCLAIM an der Weiterentwicklung von Oz. Die mit AKA-MOD (BMBF Förderkennzeichen ITW 9104) aufgenommene Zusammenarbeit an Anwendungen von Oz in der Verteilten KI wurden im Berichtszeitraum weitergeführt. Weiterhin bestand, wie auch schon in den vorherigen Berichtszeiträumen, eine enge Zusammenarbeit mit VERBMOBIL an Feature Constraint Systemen. Die Ergebnisse dieser Zusammenarbeit sind in [BT94a] dokumentiert.

Auf nationaler und internationaler Ebene bestanden zahlreiche wissenschaftliche Kontakte im Rahmen dreier Esprit-Projekte:

- Die Esprit Basic Research Action ACCLAIM (Advanced Concurrent Constraint Languages: Applications, Implementation and Methodology) verbindet die führenden europäischen Forschungsgruppen auf dem Gebiet der nebenläufigen Constraintprogrammierung. In diesem Rahmen arbeiten wir insbesondere eng mit dem Swedish Institute of Computer Science (SICS) zusammen. Der wissenschaftliche Direktor von SICS, Prof. Seif Haridi, sowie Dr. Sverker Janson besuchten im August 1994 das Projekt Hydra.
- Ein weiteres Standbein von Hydra, die Entwicklung und Kombination neuer Constraint-

systeme, wird in der Esprit Working Group CCL (Construction of Computational Logics) untersucht.

- In der Esprit Basic Research Action COMPULOG II (Computational Logic) werden Erweiterungen der Logischen Programmierung in Hinblick auf Wissensrepräsentation und Problemlösungsfähigkeiten untersucht.

Die bereits vor Projektbeginn bestehende Zusammenarbeit mit dem Paradise-Projekt am Paris Research Laboratory der Digital Equipment Corporation wurde fortgeführt. Mit Dr. Andreas Podelski, Max-Planck-Institut für Informatik, Saarbrücken, stehen wir in engem Kontakt betreffend Constraintsystemen und Fragen der Constraintprogrammierung.

Vom 31. Oktober 1994 bis zum 5. November 1994 war David N. Turner, University Glasgow, Gast von Hydra. Vasco Vasconcelos, der an der Universidad Nova de Lisboa, Portugal, an nebenläufiger objektorientierter Programmierung arbeitet, besuchte uns vom 20. bis zum 22. Dezember 1994.

### 5.3 Publikationen

- [AKPS94] H. AÏT-KACI, A. PODELSKI und G. SMOLKA: *A Feature-based Constraint System for Logic Programming with Entailment*. Theoretical Computer Science, Vol. 122 (1–2), 1994.
- [BS95] R. BACKOFEN und G. SMOLKA: *A Complete and Recursive Feature Theory*. Theoretical Computer Science, Vol. 146 (1–2), 1995.
- [BT94a] R. BACKOFEN und R. TREINEN: *How to Win a Game with Features*. In: J.-P. JOUANNAUD (Herausgeber): *1<sup>st</sup> International Conference on Constraints in Computational Logics*, München, September 1994. Lecture Notes in Computer Science 845, Springer-Verlag, Berlin, 1994.
- [CT94b] H. COMON und R. TREINEN: *Ordering Constraints on Trees*. In: S. TISON (Herausgeber): *Colloquium on Trees in Algebra and Programming*, Edinburgh, Scotland, April 1994. Lecture Notes in Computer Science 787, Springer-Verlag, Berlin, 1994.
- [DLRW94] P. DEVIENNE, P. LEBEGUE, J. ROUTIER und J. WÜRTZ: *One binary Horn clause is enough*. In: *11th Symposium on Theoretical Aspects of Computer Science*, Caen, Frankreich, February 1994. Lecture Notes in Computer Science 775, Springer-Verlag, Berlin, 1994.
- [HS94a] A. HENSE und G. SMOLKA: *A Record Calculus with Principal Types*. In: J.-P. JOUANNAUD (Herausgeber): *1<sup>st</sup> International Conference on Constraints in Compu-*

- [NS94] J. NIEHREN und G. SMOLKA: *A Confluent Relational Calculus for Higher-Order Programming with Constraints*. In: J.-P. JOUANNAUD (Herausgeber): *1<sup>st</sup> International Conference on Constraints in Computational Logics*, München, September 1994. Lecture Notes in Computer Science 845, Springer-Verlag, Berlin, 1994.
- [SS94] C. SCHULTE und G. SMOLKA: *Encapsulated Search in Higher-order Concurrent Constraint Programming*. In: M. BRUYNNOOGHE (Herausgeber): *Logic Programming: Proceedings of the 1994 International Symposium*, Ithaca, New York, USA, November 1994. The MIT Press, Cambridge, MA, 1994.
- [SSW94a] C. SCHULTE, G. SMOLKA und J. WÜRTZ: *Encapsulated Search and Constraint Programming in Oz*. In: A. BORNING (Herausgeber): *Second Workshop on Principles and Practice of Constraint Programming*, Orcas Island, Washington, USA, Mai 1994. Lecture Notes in Computer Science, Springer-Verlag, Berlin, 1994.
- [ST94] G. SMOLKA und R. TREINEN: *Records for Logic Programming*. Journal of Logic Programming, Vol. 18 (3), 1994.
- [Wür94] J. WÜRTZ: *Towards the verification of concurrent constraint programs in the field of finite domain reasoning*. In: F. DE BOER und M. GABBRIELLI (Herausgeber): *Workshop at ICLP: Verification and analysis of (concurrent) logic languages*, Santa Margherita Ligure, Italy, 1994.

#### 5.4 Sonstige Ereignisse

Das große Interesse an Oz zeigt sich daran, daß wir eingeladen wurden, auf der nächsten Weltkonferenz über Logische Programmierung (ICLP'95) in Tokio ein 75-minütiges Tutorial über Oz zu geben. Eine Zusammenfassung des Besuchs am Institut Della Molla d'Intelligenza Artificiale



## 6 OMEGA: Office Mail Expert for Goal-Directed Analysis

### 6.1 Kurzüberblick und Projektfortschritt

Das OMEGA-Projekt (Office Mail Expert for Goal-directed Analysis, BMBF Förderkennzeichen ITW 94001) hat eine automatische Analyse von Büropost zum Ziel, wobei auch unterschiedliche Beziehungen zu bereits analysierten Dokumenten in die Analyse einfließen. Als Grundlage ist eine homogene Repräsentation der analysierten Dokumente erforderlich, welche die wichtigsten Medien geschriebener Bürokommunikation (Papier, Fax, E-Mail) umfaßt. Insofern wird die in ALV (Automatisches Lesen und Verstehen, Förderkennzeichen ITW 90030) begonnene Forschung konsequent weiterverfolgt. Zentrale Absicht im ALV-Projekt war eine isolierte Verarbeitung einseitiger, gedruckter Geschäftsbriefe, aufsetzend auf dem eingescannten Bild bis hin zu einem gewissen Grad des Dokumentverstehens (Identifikation der Nachricht).

In OMEGA wird die lokale Sicht der isolierten Dokumentanalyse durch Benutzung von Dokumentbeziehungen erweitert. Tabellen als typische Elemente in Büropost sollen gesondert interpretiert werden, da sie in der Regel wichtige und eindeutige Informationen enthalten. Darüber hinaus überträgt OMEGA die bestehenden Ansätze der Analyse gedruckter Dokumente auf mehrere Eingabemedien. Die Verfahren werden für die Analyse mehrseitiger Dokumente ausgelegt, da Bürodokumente oft aus mehreren Seiten bestehen (wie z.B. Fax). Die Kernaufgabe von OMEGA besteht in der Bereitstellung von Mechanismen zur zielgerichteten Analyse von Geschäftspost, so daß eingehende Dokumente für eine effiziente Behandlung den zuständigen Bürostellen bereitgestellt werden.

### 6.2 Wissenschaftliche Ergebnisse

Im Berichtszeitraum wurden folgende wissenschaftlich-technische Ergebnisse erzielt:

**Domäne:** Als Kooperationspartner zur Anwendung des OMEGA-Prototypen konnte die Beschaffungsabteilung der Universität Kaiserslautern gewonnen werden. Der Prototyp soll zur Bearbeitung der eingehenden Dokumente dieser Abteilung entwickelt und getestet werden. Um die notwendigen Massendaten für den Systementwurf und die Tests zu bekommen, wird seit Oktober 1994 täglich die Tagespost der Beschaffungsabteilung eingescannt. Den Hauptanteil der eingehenden Post machen Bedarfsanmeldungen, Angebote, Lieferscheine und Rechnungen aus. Für die Nachrichtentypen wurde im Berichtszeitraum mit der Modellierung begonnen. Die Qualität der Dokumente stellt hohe Anforderungen an die verschiedenen Analysekomponenten. Bereits das Druckbild (Layout) weist eine Reihe von Schwierigkeiten für die gebräuchlichen Segmentierungsalgorithmen auf, ebenso ist die Güte des Zeichendruckes, insbesondere bei den häufig anfallenden Faxen, sehr niedrig. Bei ersten Tests mit kommerziellen Erkennern konnten diese bereits für solche Dokumente als untauglich beurteilt werden. Problematisch sind auch häufig auftretende handschriftliche Eintragungen auf ansonsten gedruckten Dokumenten.

**Systemarchitektur:** Im Berichtszeitraum wurde an der Komplettierung und Verfeinerung der OMEGA Systemarchitektur gearbeitet. Ziel hierbei ist die Erarbeitung eines allgemeinen Konzeptes für eine Dokumentanalyse-Shell, die vielseitig anwendbar und schnell auf unterschiedliche Domänen und Aufgaben adaptierbar ist. Alle initiierten Dokumentanalyseprojekte sollen auf dieser Architektur aufbauen.

**Dokumenterfassung:** Da OMEGA neben gedruckten Dokumenten auch Faxe verarbeiten soll, ist es notwendig, einen direkten Faxanschluß bereitzustellen. Der Umweg über den Faxausdruck mit anschließendem Einscannen ist nicht beschreibbar, da die Auflösung nur maximal

200 dpi (dots per inch) beträgt und bei jedem Scanvorgang ein Qualitätsverlust entsteht. In OMEGA wurde daher die Anbindung eines Faxmodems bereitgestellt, die direkt beim Eintreffen eines Faxes eine Bilddatei erzeugt, die Faxnummer und Anzahl der Seiten enthält. Einer vollautomatischen Analyse steht somit nichts im Wege. Ebenso wurde die wichtigste Funktionalität zum Anschluß von E-Mail erstellt. Dazu gehören vor allem Zerlegungsfunktionen, die eine Separierung von E-Mail-Kopf und eigentlichem Inhalt vornehmen, sowie solche, die auf einzelne Informationen im Kopf zugreifen. Daneben wurde eine vorläufige Transformation der E-Mail in die bestehende Layoutstruktur vorgenommen.

**Layoutextraktion:** Da Fax-Dokumente einen sich deutlich unterscheidenden Aufbau bzgl. den gedruckten Geschäftsbriefen besitzen und die in ALV entwickelte Layoutextraktionskomponente für Geschäftsbriefe optimiert ist, wurde die Entwicklung einer zusätzlichen Segmentierungskomponente notwendig. Diese Komponente sollte Graphik- und Photobestandteile von Text separieren, wobei Text in Graphiken verschachtelt sein darf (wie beispielsweise in Tabellen). Zudem sollte Text in verschiedenen Orientierungen (-5 bis +5), Größen und Schriftarten verarbeitbar sein. Für diese Aufgabe wurde in OMEGA eine Layoutextraktionskomponente entwickelt, die Text in nahezu beliebigen Orientierungen verarbeiten kann. Text kann Schriftgrößen von 6 bis 36 Punkten, nahezu alle gängigen Schriftarten und neben schwarzer Schrift auf weißem Hintergrund auch inverse Textbestandteile besitzen. Das Verfahren wurde anhand von ca. 150 Dokumenten unterschiedlichen Typs getestet. Ergebnisauszüge sind in [HönesLichter 94] beschrieben.

**Logical Labeling:** Aufgrund der Strukturvielfalt von Dokumenten ist die Aufgabe der Identifizierung logischer Dokumentobjekte ein komplexes Problem. Jedes Unternehmen verwendet für seine externen Korrespondenz ein eigenes Briefformat, in dem die einzelnen Objekte des Briefkopfes, wie Firmenangaben, Absender, Datum, usw., individuell angeordnet sind. Um trotzdem in der Lage zu sein, logische Objekte zu identifizieren, wurde ein universell einsetzbarer *Concept Formation* Ansatz entwickelt und mit Dokumenten getestet. Dazu werden Dokumente als Fuzzy-Objekte betrachtet. Durch die Analyse der Strukturmerkmale werden unterscheidende Attributwerte ermittelt und in Strukturkonzepten zusammengefaßt. Durch die iterative Anwendung von Induktion wird eine vollständige Konzepthierarchie erstellt, die dem bereits in ALV erfolgreich verwendeten GTree-Modell entspricht.

**Texterkennung:** Zur Verbesserung der Texterkennungsqualität auf Zeichen- und Wortebene sollen kommerzielle OCR-Systeme in das Gesamtsystem integriert werden. Diese bilden ferner eine wesentliche Grundlage für eine geplante Voting-Komponente. In diesem Zusammenhang wurde bisher zwei kommerzielle Zeichenerkennungssysteme in das Gesamtsystem eingebunden. Die Integration eines weiteren Produktes ist bis zum nächsten Berichtszeitpunkt geplant. Damit ist auch die notwendige Basis (mindestens drei Zeichenerkennungssysteme ausreichender Qualität) für die Voting-Komponente gelegt. Bei der homogenen Integration der kommerziellen Erkennungssysteme in das im Entstehen begriffene OMEGA-System hat sich gezeigt, daß abweichende Segmentierungsergebnisse, sowohl bei der Wort- als auch bei der Zeichensegmentierung, eine nicht zu unterschätzende Herausforderung darstellen. Die besondere Problematik liegt in den kontextsensitiven Erkennungsalgorithmen, die bei den eingesetzten Erkennungssystemen verwendet werden und bei größerem Kontext (Wort oder Zeile) zum Teil deutlich bessere Ergebnisse liefern, so daß eine Ansteuerung auf Zei-

**Voting von Zeichenerkennungsergebnissen:** Zur Ergebnisverbesserung und zur Homogenisierung von Zeichenerkennungsergebnissen wird eine Voting-Maschine entwickelt. Hierfür wurden geeignete Algorithmen und Techniken ausgewählt. Zur besseren Beurteilung wurden unterschiedliche Repräsentations- und Verknüpfungsstrategien implementiert, die jetzt auf ihre Stärken und Schwächen hin untersucht werden sollen. Gleichzeitig wurde eine erste Anbindung der Voting-Maschine an eine vorgeschaltete Lernphase realisiert, die spezifische Erkennungsmerkmale für die Verknüpfung zur Verfügung stellt.

**Lexikalische Nachverarbeitung:** Im Bereich Wortverifikation wurden die innerhalb des Projektes PEP (Online-Handschrifterkennung) entwickelten Verfahren an die Anforderungen eines OCR-Systems angepaßt. Hier zeigte sich, daß eine Verwendung dieser Methoden möglich ist. Gleichzeitig konnten erste positive Ergebnisse mit automatischen Lernverfahren für Stringvergleichs-Funktionen erzielt werden. Anhand ausgiebiger Tests konnte gezeigt werden, daß die Lernverfahren für die Kosten der Editieroperationen zu erstaunlichen Ergebnisverbesserungen führen können. Weiterhin wurde damit begonnen, die Lernverfahren theoretisch zu untersuchen, um grundlegende Erkenntnisse über deren Eigenschaften bzw. ihren Beziehungen zu Lernverfahren aus anderen Bereichen der Informatik zu erhalten. Hierbei konnte sehr schnell eine enge Verwandtschaft mit den sogenannten *Gradient Descendant* Methoden ermittelt werden, die bei Neuronalen Netzen sehr häufig zum Einsatz kommen und bereits beim Perceptron Verwendung fanden. Es konnte z.B. bereits gezeigt werden, daß die in unserem Bereich zum Einsatz kommenden Verfahren, unter gewissen Voraussetzungen, eine Lösung finden, sofern eine solche existiert.

**Lexikonvorverarbeitung:** Im Bereich der Lexikonvorverarbeitung wird an einem SGML-Konverter gearbeitet, der die Aufgabe hat, bestehende elektronische Wörterbücher in ein allgemein verfügbares Austauschformat (SGML) überzuführen. Diese SGML-kodierten Lexikoneinträge sollen anschließend in das OMEGA-Wörterbuch für die inhaltliche Analyse von Dokumenten eingebunden werden. Hierzu wurde eine SGML-Dokumenttypdefinition für Lexikoneinträge und ein erster Prototyp für die SGML-Konvertierung entwickelt, welche im Moment anhand größerer Datenmengen ausführlich getestet werden.

**Klassifikation von Geschäftsbriefen:** Neben der statistischen Klassifikation von Geschäftsbriefen (vgl. INFOCLAS-System) wurde im Berichtszeitraum eine zweite, regelbasierte Klassifikationskomponente mit Namen RULECLAS fertiggestellt. Die Idee besteht darin, daß beide Komponenten, die auf völlig unterschiedlichen Verfahren aufsetzen, sich gegenseitig unterstützen sollen. Kern von RULECLAS ist ein Regelinterpreter, der Worthypothesen als Evidenzen für inhaltliche Konzepte durch ein hierarchisches Regelnetzwerk propagiert. Dieser Regelinterpreter kann mit bewerteten Wortalternativen bzw. Unvollständigkeits der Texterkennung erfolgreich umgehen. Bemerkenswert bei diesem Ansatz ist die Spezifikation einer hierarchischen Klassenstruktur von inhaltlichen Konzepten für die Anwendungsdomäne von Geschäftsbriefen. Eine weitere Charakteristik des Systems zeigt sich durch die geschickte Einbettung zusätzlicher Wissensquellen, um die Klassifikationsergebnisse zu verbessern. Solche kontextuellen Wissensquellen in RULECLAS sind: Lexikonsichten (Views), morphologische Ergebnisse (z. B. Wortkategorie), bestimmte Wortkombinationen und Floskeln, die Brieflänge sowie inhaltliche Suchmuster (Pattern).

**Steuerung:** Für die meisten Analysephasen des OMEGA-Systems werden verschiedene Einzelverfahren entwickelt, die in geeigneter Weise zu kombinieren sind. Darüberhinaus sind Rückkopplungen und Verzweigungen im Analysefluß zuzulassen. Zu diesem Zweck wurde im Berichtszeitraum ein Grundkonzept für eine geeignete Steuerung erstellt. Die Analysesteuerung wird reaktiv vorgehen, d.h. Probleme während der Analyse erkennen und geeignete Problemlösungs-

schritte einleiten. Basiselement werden dabei ein Scheduler und deklarative Beschreibungen von jedem beteiligten Einzelverfahren bilden. Die Analysesteuerung wird reaktiv vorgehen, d.h. Probleme während der Analyse erkennen und geeignete Problemlösungsschritte einleiten.

**Wissensrepräsentation:** Durch die starke Verzahnung des OMEGA-Projektes mit Tandemprojekten entstand das Bedürfnis, Systemkomponenten untereinander besser austauschen zu können. Als Hauptproblem erwiesen sich dabei die unterschiedlichen Domänen (allgemeine Geschäftskorrespondenz, Mitteilungsblätter, Anforderungsschreiben). Durch eine Wissensrepräsentationssprache, in der formal das von allen Analyseverfahren benutzte Wissen beschrieben wird, hoffen wir eine leichtere Adaptierbarkeit auf beliebige Domänen gewährleisten zu können. Die begonnene Entwicklung eines 'Sprachformalismus' zur Beschreibung von Dokumentwissen wurde fortgeführt, und für die entstandene Sprache DRL0 wurde ein Basisinterpreter entwickelt. Um Anforderungen und Problematik der Wissenskompilation bei Verwendung von DRL0 besser einschätzen zu können, wurde ein Analyseverfahren exemplarisch in der Quell- und Zielsprache implementiert; derzeit wird der o.g. Interpreter um eine Inferenzmaschine und Codeerzeugungskomponente erweitert, mit dem Ziel, die Beispielapplikation automatisch übersetzen zu können.

### 6.3 Publikationen

- [BHB94] J. BOON, F. HÖNES, M. BEN HADJ ALI: *Context-driven text recognition by means of dictionary support*. Proc. SPIE Document Recognition, San Jose, California, 1994.
- [DBH<sup>+</sup>94] A. DENGEL, R. BLEISINGER, R. HOCH, F. HÖNES, M. MALBURG und F. FEIN: *OfficeMAID - A system for Automatic Mail Analysis, Interpretation and Delivery*, Proc. Workshop on Document Analysis Systems (DAS94), Int'l Association for Pattern Recognition, Kaiserslautern, Germany, 1994.
- [Hoc94] R. HOCH: *Using IR Techniques for Text Classification in Document Analysis*. Proc. 17th International Conference on Research and Development in Information Retrieval (SIGIR'94), Dublin City, Ireland, 1994.
- [HKH94] R. HOCH, T. KIENINGER und H.-G. HEIN: *Using a Partitioned Dictionary for Contextual Post-Processing of OCR-Results*, Proc. 12th Int'l Conference on Pattern Recognition (ICPR'94), Jerusalem, Israel, 1994.
- [HL94] F. HÖNES und J. LICHTER: *Layout Extraction of Mixed Mode Documents*, Machine Vision & Applications, Vol. 7, Springer-Verlag, Heidelberg, 1994.
- [JHD95] T. JÄGER, F. HÖNES und A. DENGEL: *An Adaptive Metaclassifier for Word Recognition Based on Multiple Independent Classifiers*, Proc. Symposium on Document Analysis and Information Retrieval (SDAIR '95), Las Vegas, Nevada, 1995.
- [WB94] A. WEIGEL und S. BAUMANN: *A Modified Levenshtein-Distance for Handwriting Recognition*. In: S. IMPEDOVO (Herausgeber) *Progress in Image Analysis and Processing III*, World Scientific Pub., London, 1994.
- [WF94] A. WEIGEL und F. FEIN: *Normalizing the Weighted Edit Distance*, Proc. 12th Int'l Conference on Pattern Recognition (ICPR'94), Jerusalem, Israel, 1994.
- [WH95] C. WENZEL und R. HOCH: *Text Categorization of Scanned Documents Applying a Rule-based Approach*, Proc. Symposium on Document Analysis and Information Retrieval (SDAIR '95), Las Vegas, Nevada, 1995.

#### 6.4 Sonstige Ereignisse

Vom 18. bis 20. Oktober 1994 wurde in Kaiserslautern ein internationaler Workshop mit dem Titel *Document Analysis Systems* erfolgreich abgehalten. Veranstalter war die International Association for Pattern Recognition (IAPR), wobei Prof. Dr. Andreas Dengel Chairman war.

## 7 PARADICE: Parameterizable NL Discourse Core Engine

### 7.1 Kurzüberblick und Projektfortschritt

Die Projektziele wurden gemäß der Auflagen des BMFT teilweise revidiert und in verschiedenen Punkten konkretisiert. Die Ergebnisse wurden dem BMFT als Ergänzungen zur Projektbewilligung schriftlich vorgelegt.

Das PARADICE-Projekt baute auf die Module des Kernsystems auf, die in dem vom BMBF-geförderten DISCO-Projekt (FKZ ITW 9002 0) entwickelt wurden. Diese Module wurden in wesentlichen Punkten erweitert und teilweise auch ersetzt.

Im Berichtszeitraum wurde im Bereich Architektur und Diskurs die Konzeption einer Schnittstelle zum Dialog entwickelt und es wurde das Diskursgedächtnis um Mechanismen zur Referen-

zaufösung erweitert.

Bei den linguistischen Wissensbasen lag der Schwerpunkt auf einer Verbreiterung der Grundlagen für die Verarbeitung größerer Textmengen. Die im DISCO-System verwendete Morphologiekomponente wurde durch eine robustere Komponente (MORPHIX) mit wesentlich größerer Abdeckung ersetzt. Durch die Wiederverwendung existierender Ressourcen wurde dabei die Abdeckung dieser Morphologiekomponente auf ca. 100.000 Stämme erweitert. Die Verbindung zu anderen Verarbeitungskomponenten wurde durch eine getypte Schnittstelle realisiert. Die Scanning-Komponente wurde auf textuelle Phänomene angepaßt, die Syntax und das Grammatikfragment wurden kontinuierlich ausgebaut und es wurde mit dem Aufbau einer breit angelegten Lexikon-Datenbank auf der Basis existierender Ressourcen begonnen.

Im Bereich Effiziente Verarbeitung wurde der Parser mit einer Komponente für *Explanation-Based Learning* verbunden. Diese Komponente wurde so erweitert, daß sie auch auf der Ebene von Phrasen für eine effiziente Erkennung und Verarbeitung von partiellen Strukturen einge-

positionale Inhalte und Fehler-Protokolle für vorausgehende linguistische Module (akustisches Fehlverstehen, Mißverstehen) umfassen. Diese Arbeiten zeigten die Notwendigkeit, in den Formalismus einen Begriff von theoretischer *Präferenz* zu integrieren, der in engem Zusammenhang zu Default-Theorien steht. Dieser Begriff ist distinkt sowohl von Verarbeitungs-*Prioritäten*, wie sie im PARADICE-System eingesetzt werden, um die Parsingstrategie und seit neuestem auch Unifikation zu steuern, als auch von empirischen *Wahrscheinlichkeiten*, die zum Beispiel auf der Basis von Trainingsdaten berechnet werden können. Solche Präferenzen würden auf der deskriptiven Ebene die Erfassung einer Reihe von Dialog- und anderen generellen Phänomenen unterstützen. Es wurde gezeigt, daß Ansätze, die die Verarbeitungsprioritäten zu diesem Zwecke ausnutzen, zu einer Verschlechterung der Systemperformanz führen können.

### AP 13: Basis für Diskurs

Das zur Referenzauflösung verwendete Diskursgedächtnis wurde um eine Wissensbasis für Fakten mit einer eigenen PROLOG-artigen Inferenzmaschine erweitert, die für die Resolution einfacher und komplexer Kennzeichnungen eingesetzt wird. Auch wurden Schnittstellen vom NLL-System zur Sortenlogik von TDL implementiert, um für den sortalen Abgleich während der Resolution die Ontologie der Merkmalssemantik verfügbar zu machen. Daneben wurden zusätzliche Mechanismen zur Einbeziehung konzeptuellen Wissens für die Referenzauflösung integriert, insbesondere Teil-Ganze-Beziehungen.

Außerdem wurde der Leistungsumfang der Resolutionskomponente in Richtung auf die Behandlung von Nicht-Identitäts-Anaphern erweitert. Dabei wurden zunächst ein Summationsverfahren für Pluralanaphern und Abstraktionsverfahren für N-bar-Anaphern realisiert. Auch wurde begonnen, das Diskursgedächtnis auf Dialogverarbeitung vorzubereiten, indem Mechanismen zur Verfolgung von Sprecher-Hörer-Wechseln integriert wurden. Am Beispiel von Terminen wurde als zusätzliche Komponente das Konzept des *Diskurs-Topics* eingeführt, welches zusammen mit einer in Entwicklung befindlichen Komponente zur Terminverfolgung in Terminvereinbarungsdialogen ein abstraktes Referenzobjekt in Fällen liefert, in denen ein direkter Match mit Antezedenten nicht möglich ist (Beispiel: *Ich schlage den Montag vor. - Das geht bei mir nicht.*). Damit wurde eine dritte wichtige Phänomenklasse im Bereich der Nicht-Identitätsanaphern bearbeitet.

Bei der Verarbeitung wurden *backtracking*-Mechanismen für alle Resolutionsprozesse implementiert, um alternative Lösungen zugänglich zu machen. Diese Mechanismen sind in die PARADICE-Architektur integriert.

Im Bereich der Syntax-Semantik-Schnittstelle wird die Übersetzung der semantischen Merkmalsstrukturen nach NLL nunmehr vom Typ der Merkmalsstruktur gesteuert statt durch einen eigenen Klassifikator. Dies erhöht die Transparenz und erleichtert die für die Generierung erforderliche Übersetzung von logischen NLL-Strukturen in semantische Merkmalsstrukturen.

## 7.2.2 Linguistische Wissenbasen

### AP 21: Morphologie

Da abzusehen war, daß die in DISCO verwendete Morphologiekomponente X2MORF für die Verarbeitung von größeren Datenmengen nicht über eine ausreichende Robustheit und Effizienz verfügt, wurde diese Komponente durch das Morphologiesystem MORPHIX ersetzt. Der Nachteil, daß MORPHIX, im Gegensatz zu X2Morf, nicht mit den in der Grammatik verwendeten TDL-Datenstrukturen arbeitet, mußte bei der Integration in das Gesamtsystem durch eine Schnittstelle ausgeglichen werden.

Dazu wurden im TDL-Formalismus morpho-syntaktische Typen spezifiziert, die die mögli-

chen morphologischen Formen und paradigmatischen Konfigurationen des Deutschen abdecken und damit eine extensionale Schnittstelle definieren. Jeder Analyse-Output von MORPHIX kann so durch eine einfache Vergleichsoperation auf einen Typnamen abgebildet werden, der dann im TDL-System expandiert werden kann.

Aus diesem Ansatz ergeben sich mehrere Vorteile. Zum einen liegt die Definition der morpho-syntaktischen Typen nunmehr vollständig im Bereich der Grammatik, d.h. die Typdefinitionen können jederzeit unabhängig verändert werden, solange die Typnamen beibehalten werden. Dadurch verfügt die Grammatik über eine getypte Schnittstelle, die die genauen Anforderungen an eine Morphologiekomponente spezifiziert, so daß theoretisch auch andere Morphologiesysteme angeschlossen werden könnten. Zum anderen erhielt MORPHIX dadurch eine Schnittstelle, die die Verbindung zu anderen Verarbeitungssystemen wesentlich vereinfacht und auch die Möglichkeit bietet, MORPHIX als morpho-syntaktischen Tagger einzusetzen.

Das morphologische Lexikon von MORPHIX wurde erheblich erweitert, nämlich um 70.000 neue Nomen- und 14.000 neue Adjektiveinträge. Ausgangspunkt für die Erweiterung waren entsprechende Einträge aus dem SADAW-Lexikon. Die neuen MORPHIX-Einträge wurden dabei automatisch mit Hilfe einer Übersetzungsroutine erzeugt, die im wesentlichen eine Transformation der morphosyntaktischen Information von der SADAW-Kodierung in die entsprechende MORPHIX-Kodierung durchführt. Insgesamt verfügt MORPHIX damit nun über mehr als 100.000 Stammeinträge.

Bei der Verarbeitung dieses wesentlich erweiterten Lexikons zeigte sich, daß die bis dahin verwendete Speicherung der lexikalischen Einträge mittels Hash-Tabellen zu einem erheblichen Effizienzverlust führte. Um diesen Verlust an Performanz wieder aufzufangen, wurde MORPHIX auf die Verwendung von TRIEs (Buchstabenbäumen) angepaßt. Die spezifische Implementierung der TRIEs führte dazu, daß auch das neue, wesentlich größere Lexikon keinen Verlust der Performanz zur Folge hat, sondern der Zugriff nahezu konstant bleibt. Neben den üblichen Operationen auf TRIEs (Einfügen, Zugreifen, Löschen) wurden auch Operationen implementiert, die es erlauben, Information existierender Einträge zu überschreiben und zu erweitern, wobei dies auch für beliebige, nicht MORPHIX-spezifische Information gilt. Des weiteren wurde damit begonnen, Operationen zu definieren, die Zugriffe mittels Mustern möglicher Lemmata erlauben.

#### **AP 22: Text Scanning**

Der Scanner wurde so erweitert, daß nun eine ganze Datei (und nicht nur einzelne Sätze) verarbeitet, erkannt und markiert werden können. Dadurch kann z.B. eine Datei in eine Liste von Paragraphen aufgespalten werden, die wiederum aus Listen von Sätzen bestehen. Schlüsselwörter in tabellarischen Auflistungen können nun speziell behandelt werden. Außerdem werden spezielle Schreibweisen, wie z.B. gesperrt gedruckte Worte oder Trennstriche, erkannt.

#### **AP 23: Lexikon**

Es wurde damit begonnen, die Ressourcen des SADAW-Lexikons in eine Lexikon-Datenbank mit maschinell verarbeitbarer Informationsstruktur zu überführen. In der ersten Phase wurde eine Konzeption von Datenstrukturen entwickelt, die sich einerseits für den Benutzer in transparenter, leicht verständlicher und linguistisch moderner Form darbieten, die aber in der internen Darstellung den Anforderungen eines intelligenten relationalen Datenbanksystems folgen. Des weiteren wurde die Oberflächenfunktionalität definiert, sowie die Strukturen für den Datenim-



matikfragment des Deutschen auf komplexe Sätze. Dazu zählen sowohl subordinierte Komplementsätze als auch Relativsätze. Für die Analyse der Komplementsätze konnte auf die theoretischen Vorarbeiten für funktionale Köpfe und Verbstellung zurückgegriffen werden. Für Relativsätze wurde ein neuartiger Ansatz entwickelt, der im Gegensatz zum traditionellen HPSG-Ansatz eine Analyse ohne leere Knoten ermöglicht und gleichzeitig die Behandlung von Pied Piping Phänomenen erleichtert.

Des Weiteren wurde das Fragment um Auxiliar- und Passivkonstruktionen ausgebaut. Damit steht im Bereich des Verbalkomplexes nunmehr das volle Paradigma an synthetischen und analytischen Tempora zur Verfügung. Die Passivkonstruktionen umfassen momentan die Varianten des Vorgangs- und Zustandspassivs. Durch die Entwicklung eines Sortensystems zur Markierung von Vollverben, Tempus-Auxiliaren und Modalverben, konnten die relevanten kombinatorischen Restriktionen erfaßt werden, ohne daß auf Disjunktionen und Ambiguitäten zurückgegriffen werden mußte.

### 7.2.3 Verarbeitung

#### AP 31: Robustheit

Im Bereich Robuste Verarbeitung wurde vor allem eine Spezifikation von Fehlermeldungsprotokollen festgelegt und es wurden die Grundlagen für die statistische Verarbeitung von Texten geschaffen.

*Fehlermeldungsprotokolle:* Fehlermeldungen wurden als Methoden für ein Objekt der Klasse ERROR definiert. Dieses Objekt verwaltet die Fehlermeldungen und selektiert entsprechende Fehlermeldungsrountinen. Der Vorteil der objekt-orientierten Modellierung ist, daß die Aktivierung von Fehlermeldungen und entsprechenden Routinen automatisch vom zugrundeliegenden OOP-System vorgenommen wird. Die entsprechende Klassendefinition wurde bereits in die Architektur übernommen.

Bei den Fehlermeldungsprotokollen werden "vorhersagbare Fehler" und "unvorhersagbare Fehler" unterschieden. Vorhersagbare Fehler sind solche, die vom Entwickler eines Moduls spezifiziert sind (gemäß eines vorgegebenen Standards). Solche Fehler sind "vorhersagbar", da der Modulzustand genau definiert und damit eine spezifische Fehlerbehandlung durchgeführt werden kann. Für die Definition von vorhersagbaren Fehlern wurde ein erster Prototyp implementiert und ebenfalls in die Architektur integriert. Unvorhersagbare Fehler sind solche, die nicht durch ein Modul als vorhersagbar definiert sind oder externe Ursachen haben (z.B. Hardwarefehler). In diesem Fall ist die Auswahl einer korrekten Fehlerbehandlung nicht eindeutig gegeben.

*Korpusanalysetools:* Die robuste Verarbeitung natürlicher Sprache (z.B. die Behandlung unbekannter Wörter) auf der Basis statistischer Information setzt Mechanismen zur automatischen Korpusanalyse voraus. Dazu wurde im Berichtszeitraum auf der Basis des existierenden Scanners und der Morphologiekomponente MORPHIX ein System implementiert, das für einen beliebigen in ASCII-Format vorliegenden Text, eine morphologisch basierte, automatische Korpusanalyse durchführt. Diese umfaßt eine lokale als auch eine globale Analyse.

In der lokalen Analyse wird für jede Satzeinheit folgende Information protokolliert: Identifikation von Tokens und Wortformen, morphologisch analysierte und nicht analysierte Formen, sowie die präterminale Kette eines Satzes in Form von Kategorien. Die globale Analyse protokolliert die absolute Häufigkeit von unbekanntem Wörtern, von erkannten Wörtern, sowie von Clustern von präterminalen Ketten. Die pro Satz erkannten präterminalen Kategorienketten werden dabei in einem Diskriminationsnetz gespeichert, das später zum Definieren von korpusrelevanten endlichen Automaten herangezogen werden kann und damit als Grundlage für *Shallow Parsing* dienen kann. Des Weiteren werden während der globalen Analyse n-grams für Wortformen, Lem-

mata und präterminale Elemente berechnet. Es ist an dieser Stelle zu betonen, daß die globale Analyse inkrementell arbeitet, also unmittelbar nach jeder lokalen Analyse einsetzt.

Das gesamte System ist bereits vielseitig parametrisierbar. Beispielsweise kann definiert werden, welche der oben aufgeführten Informationen tatsächlich während der Analyse berücksichtigt werden sollen. Auch die Berechnung der n-grams ist parametrisierbar (bi-grams, tri-grams oder eben n-grams) und es kann angegeben werden, auf der Basis welcher Information (Wortform, Lemma oder präterminale Element) die n-grams berechnet werden sollen.

Dieses System wurde auf der Basis eines 700 KB großen Korpusstextes (über 120.000 Token) getestet, wobei die gesamte Analysezeit bei ca. 200 CPU Sekunden liegt.

### **AP 32: Effizienz**

Im Bereich der Effizienten Verarbeitung lag der Schwerpunkt auf dem Ausbau und der Integration der Komponente für Explanation-Based Learning (EBL), sowie auf der Konzeption eines chart-basierten top-down Generators.

*Explanation-Based Learning:* Es wurde ein direktes Interface zwischen dem Parser und der EBL-Komponente implementiert, um zu ermöglichen, daß von EBL gefundene mögliche Ableitungen zusammen mit dem Parser benutzt werden. Der Parser extrahiert auf Anfrage aus einem Parsingergebnis die Ableitungsgeschichte derart, daß sie während der Lernphase von EBL an der entsprechenden Stelle des Diskriminationsnetzes abgespeichert werden kann.

Wenn EBL in der Anwendungsphase eine gültige Kette (oder Teilketten) erkennt, werden diese Strukturen wieder an den Parser übergeben, der sie dazu benutzt, die abgespeicherte Derivation quasi-deterministisch nachzuvollziehen.

Aufbauend auf dieses Interface wurde in einem zweiten Schritt eine erste Version einer engen Verzahnung von Parsing und EBL auf phrasaler Ebene realisiert. Dadurch wurde es möglich in der Trainingsphase Ableitungsmuster für Phrasen zu extrahieren und in der Anwendungsphase entsprechend einzusetzen. Damit erhöht sich nicht nur die Performanz, sondern auch die Flexibilität des Gesamtsystems, das nicht mehr auf vollständige Satzmuster eingeschränkt ist. Im einzelnen wurden dabei die folgende Arbeiten durchgeführt:

- Trainingsphase:
  - Alle möglichen Teilbäume eines analysierten Satzes werden extrahiert, abstrahiert und in einem Diskriminationsnetz abgelegt.
  - Für jeden Teilbaum werden “Chunks” berechnet, d.h. Merkmalsstrukturen, die nur die Informationen des Wurzelknotens und die abstrahierte Terminal-Kette kodieren.
  - Für das Diskriminationsnetz wurde ein graphischer Browser, sowie Operationen zum Sichern und Einlesen der durch EBL berechneten Strukturen implementiert.
  
- Anwendungsphase:
  - Für eine lexikalisch analysierte Wortkette werden alle matchenden Chunks bestimmt und anschließend mit der lexikalischen Information unifiziert.
  - Alle erfolgreich identifizierten Chunks werden dem Parser übergeben, der diese unmittelbar in seine Chart einfügt.
  - Der Parser versucht nun diese Chunks zu kombinieren, wobei Regeln der Kompetenzgrammatik herangezogen werden können.

In der Anwendungsphase können unterschiedliche Strategien verwendet werden, die die Auswahl zwischen alternativen Chunks steuern. So gibt es einen "exhaustiven" Modus, der alle möglichen Chunks bestimmt und diese dem Parser weiterreicht. Der Parser erhält dadurch die Möglichkeit, mit Hilfe seines Agendamechanismus selbst die Auswahl zu bestimmen, wodurch ein "backtracking" in Chunks realisiert ist. Es ist aber auch möglich, daß EBL nur die Chunks mit größter Spannweite liefert. In diesem Fall, muß EBL selbst das Backtracking verwalten. Dieses durch EBL gesteuerte Backtracking wird in einer nächsten Version...

*Generator:* Die Konzeption einer Earley-basierten Grammatikgenerierung wurde abgeschlossen und in eine algorithmische Spezifikation gebracht. Die Vorteile dieser Generierungsmethode gegenüber der im DISCO-System verwendeten Variante der "semantic head driven" Generierung sind die folgenden:

- Durch die Verwendung einer dynamischen Selektionsfunktion ist eine stärkere daten-gesteuerte Kontrolle möglich. Darüberhinaus kann durch Verwendung von Präferenzen zusätzlich eine statistisch-orientierte Auswahl getroffen werden.
- Der neue Algorithmus verwendet eine Chart. Damit werden redundante Berechnungen vermieden, die bei Backtracking entstehen würden.
- Als generische Kontrolle wird ein Agendamechanismus verwendet, wie er auch in ähnlicher Weise bereits im Parser eingesetzt wird. Damit steht auch ein sehr flexibler Regelauswahlmechanismus zur Verfügung, der ebenfalls präferenzbasiert gesteuert werden kann.
- Das neue Verfahren ist aufgrund seiner Flexibilität auf eine größere Klasse von Grammatiken anwendbar. Weiterhin garantiert die Earley-basierte Kontrolle...

- [NNP94] NERBONNE, J., NETTER, K. und POLLARD, C. (Herausgeber): *German in Head-Driven Phrase Structure Grammar*. Nummer 46 in *CSLI Lecture Notes*. CSLI, Stanford, 1994.

#### 7.4 Sonstige Ergebnisse

Prof. Vijay-Shanker von der University of Delaware verbrachte sein Sabbatical an der DFKI GmbH. Zusammen mit ihm wurde die theoretische Basis und ein Algorithmus für die Compilierung von HPSG- in TAG-Grammatiken entwickelt. Diese Arbeit wurde unterstützt durch einen kurzen Gastaufenthalt von Prof. Robert Kasper (Ohio State University). Im Rahmen eines Workshops zum Thema "HPSG and TAGs" am DFKI, Saarbrücken (12./13. 06. 94), kam es zu weiteren intensiven Kontakten mit Prof. Aravind Joshi und Prof. Anthony Kroch (Univ. of Pennsylvania), sowie Dr. Anne Abeille und Dr. Owen Rambow (Université Paris VII).

Die PARADICE-System wurde von Stephan Oepen im Rahmen eines mehrwöchigen Gastaufenthalts am CSLI (Stanford University) für die Gruppe von Prof. Ivan Sag und Dr. Daniel Flickinger (Verbmobil) installiert. Von den Forschern am CSLI wurde dieses System äußerst positiv aufgenommen und wird zur Entwicklung einer englischen HPSG-Grammatik eingesetzt. Über das CSLI entstand auch ein Kontakt zur Simon-Frazier Universität in Vancouver, wo das PARADICE-System ebenfalls installiert wurde und zur Entwicklung einer japanischen HPSG und zu Unterrichtszwecken verwendet wird.

## 8 PPP: Personalized Plan-Based Presenter

### 8.1 Kurzüberblick und Projektfortschritt

Das Projekt 'Personalized Plan-Based Presenter' (*PPP*: BMBF-Förderkennzeichen ITW 9400 7) wurde am 1. Februar 1994 begonnen und läuft bis zum 31. Dezember 1996.

Das Ziel des Projekts ist die Erforschung und Entwicklung innovativer Präsentationstechniken für zukünftige intelligente Benutzerschnittstellen. Die zentralen Problemstellungen des Projekts sind:

- **Planung multimedialer Präsentationshandlungen:**  
Ein Präsentationssystem muß nicht nur in der Lage sein, multimediale Dokumente zu synthetisieren, sondern auch planen, wie dieses Material verschiedenen Benutzern präsentiert werden soll. Eines der Ziele des *PPP*-Projekts ist es, natürlichere und effektivere Präsentationen erzeugen. Dazu soll eine animierte Figur als Präsentator verwendet werden, der das generierte Material zeigt und erklärt.
- **Interaktive multimediale Präsentationen:**  
Da es unmöglich ist, Präsentationen von vornherein so zu gestalten, daß sie den Wünschen und Erfordernissen jedes denkbaren Benutzers gerecht werden, sollte ein Präsentationssystem Interaktion mit dem Benutzer erlauben. Im *PPP*-System sollen sowohl Zwischenfragen bezüglich des Diskursbereichs als auch Metafragen zum Präsentationsstil behandelt werden.
- **Kontinuierliche Effektivitätskontrolle von Präsentationen:**  
Um herauszufinden, ob ein Benutzer eine Anweisung wirklich verstanden hat, muß ein System die Effekte seiner Präsentation kontinuierlich überprüfen. Eine Möglichkeit, diese Kontrolle durchzuführen, besteht darin, das Präsentationssystem mit den zu bedienenden technischen Geräten über einen Datenbus zu verbinden. Mittels einer solchen Verbindung kann das *PPP*-System das Verhalten des Benutzers nachverfolgen und seine Präsentationen an die jeweilige Situation anpassen.
- **Bereitstellung einer ausgereiften Präsentationsgrundlage:**  
Eine einfache Anpassung an neue Diskursbereiche erfordert die Entwicklung ausreichend flexibler und mächtiger Repräsentationstechniken, mit denen ein breites Spektrum potentieller Anwendungen erfaßt werden kann. Darüberhinaus sollten diese Techniken von entsprechenden Inferenzmechanismen begleitet werden, die die Implementation des multimedialen Präsentationssystems unterstützen.

Gemäß den inhaltlichen Schwerpunkten gliedert sich das *PPP*-Projekt in die beiden Arbeitsgruppen Präsentationsplanung und Design (PPD) und Wissensrepräsentation und Reasoning (KR&R). Die PPD-Gruppe konzentriert sich auf Probleme der Dialogplanung in einer multimedialen Umgebung, interaktive graphische Präsentationstechniken, Architekturen für Multimediasysteme und interaktives multimediales Displaymanagement. Die KR&R-Gruppe beschäftigt sich mit Repräsentations- und Reasoningtechniken zur Unterstützung von multimedialen Präsentationssystemen.

Im ersten Halbjahr standen konzeptionelle Arbeiten im Hinblick auf die Realisierung eines interaktiven, multimedialen Präsentationssystems im Vordergrund. Auf der Grundlage dieser Arbeiten wurden im zweiten Halbjahr Implementierungsarbeiten im Hinblick auf die Realisierung

eines interaktiven, multimedialen Präsentationssystems durchgeführt und die aus WIP übernommenen Komponenten (3D-Graphikpaket, Displaykomponente, Präsentationsplaner, RAT) angepaßt und auf CLIM-2 portiert.

## 8.2 Wissenschaftliche Ergebnisse

### 8.2.1 Teilprojekt Präsentationsplanung (PPD)

Multimediale Präsentationen, wie sie beispielsweise im ausgelaufenen BMBF-geförderten WIP-Projekt (ITW 8901 8) generiert wurden, bieten dem Benutzer keine Möglichkeit zur Interaktion. So können weder Rückfragen gestellt werden, noch ist es möglich, während der Präsentation

legten nahe, Präsentationen nicht nur multimedial, sondern auch interaktiv zu gestalten.

Es zeigte sich, daß sich eine ansprechende Art der Interaktion durch Einsatz von Hypermediantechniken vergleichsweise einfach realisieren läßt. Der Grundgedanke bestand darin, Teile einer Präsentationshandlung nicht nur vom System, sondern auch vom Benutzer planen und/oder realisieren zu lassen. Beispielsweise wird bei einer Rückfrage eine bestimmte Präsentationshandlung vom Benutzer initiiert, die Ausführung erfolgt jedoch durch das System. Bei Explorationshandlungen wird hingegen eine bestimmte Präsentationshandlung vom System geplant und vom Benutzer ausgeführt. Je nachdem wie stark der Benutzer in die Präsentation involviert ist, ließen sich vier grundlegende Präsentationsstile unterscheiden.

Um die Struktur interaktiver Präsentationen adäquat beschreiben zu können, wurde zwischen *Präsentationshandlungen* und *Dialoghandlungen* unterschieden. Präsentationshandlungen beziehen sich auf die Präsentation eines Dokuments. Beispiele hierfür sind Aktions- oder Objektbeschreibungen. Dialoghandlungen beziehen sich hingegen auf die Interaktion mit dem Benutzer. Hierunter fallen die Unterbrechung eines Dialogpartners oder die Ablehnung einer Bitte. Diese Dialoghandlungen lassen sich wiederum zu prototypischen Handlungssequenzen zusammenfassen, wie z.B. einer Sequenz bestehend aus einem *Request-for-Elaboration* und einem *Comply-with-Request*.

Um Explorationshandlungen im selben Formalismus behandeln zu können, wurden sie als Folgen von Dialoghandlungen aufgefaßt. Fährt der Benutzer beispielsweise mit der Kamera um ein Objekt herum, so wird dies als Folge von Kommandos mit den entsprechenden Systemreak-

um neue Constraints oder neue Präsentationsziele hinzuzufügen. Glücklicherweise ist diese Vor-

Verfahren sah bereits Rückmeldungen vor, zwar nicht vom Benutzer, jedoch von den Generierungskomponenten. Um auch Benutzerinteraktion zu ermöglichen, wurden Dialogstrategien definiert, mit denen angemessen auf das Verhalten des Benutzers reagiert wird. Stellt der Benutzer beispielsweise mitten in einer Präsentation eine Anfrage, die bereits bearbeitet wurde, so wird eine alternative Präsentationsstrategie verwendet, um die Anfrage auf andere Art und Weise zu beantworten, da die gewählte Methode offensichtlich nicht zum Erfolg führte. Existiert keine andere Präsentationsstrategie, so wird der entsprechende Präsentationssteil graphisch markiert, um dem Benutzer zu verstehen zu geben, wo er die Antwort auf seine Frage findet. Um das Verfahren im Standalone-Betrieb testen zu können, wurden Benutzerrückmeldungen, die sich auf bereits erzeugte

Präsentationsteile beziehen, durch Anklicken graphisch angezeigter Planknoten simuliert, wobei die Auswahl eines Knotens so interpretiert wird, als hätte der Benutzer den dazu korrespondierenden Teil der Präsentation angeklickt.

Laut Antrag soll das PPP-System nicht nur Material erzeugen, sondern auch planen, wie dieses Material dem Benutzer zu präsentieren ist. Um Präsentationen ansprechender zu gestalten, soll eine animierte Figur eingesetzt werden, die das generierte Material zeigt, kommentiert und erklärt. Hierzu konnten wichtige technische Vorarbeiten erfolgreich abgeschlossen werden. Es

Präsentation sehr großer, hierarchisch organisierter Datenmengen, wissensbasiert ausgewählt. Darüber hinaus konnte erfolgreich gezeigt werden, wie sich die Komponente *Inlay* effektiv zur Platzierung von Metapräsentationsobjekten, wie der animierten Figur *PPP-PERSONA*, einsetzen läßt.

### 8.2.2 Teilprojekt Wissensrepräsentation und Reasoning (KR&R)

Das bereits im Vorgängerprojekt WIP entwickelte System RAT zur Repräsentation und Verarbeitung von Aktionen und Plänen unter Zuhilfenahme terminologischer Logiken wurde um Konzepte zur Einbeziehung von Zeit erweitert. Die erste Version des RAT-Systems erlaubte lediglich lineare Sequenzen von Teilaktionen zur Komposition von Plänen. Durch Integration des von Kautz und Ladkin entwickelten und implementierten Systems MATS (Metric Allen Time System) in RAT wurde es möglich, metrische und qualitative temporale Beziehungen zwischen Aktionen und Plänen zu repräsentieren und Inferenzen über diese komplexen zeitlichen Beziehungen zu ziehen. Im Rahmen der Implementierungsarbeiten (die teilweise innerhalb eines Fortgeschrittenenpraktikums durchgeführt wurden) galt es unter anderem, die Algorithmen, die für die einfacher aufgebauten linearen Pläne der ersten Version von RAT implementiert wurden, dieser elementaren Erweiterung der Ausdrucksmöglichkeiten anzupassen. Dies betraf in starkem Maße die Berechnung der allgemeinen Ausführbarkeit von Plänen (Feasibility Test). Die Erweiterung mußte z.B. auch parallel ablaufende Aktionen und Bedingungen, die nur während des Ablaufs einer Aktion gelten (oder gelten müssen), berücksichtigen. Weitere Probleme ergaben sich durch die Tatsache, daß das neu integrierte MATS-System Disjunktionen von zeitlichen Relationen zwischen Planteilen erlaubt. Dadurch stellt die Definition eines Plans nicht mehr nur einen linearen Aktionsablauf dar, sondern sie beschreibt einen komplexen Vorgang, der auf unterschiedliche Weise ablaufen kann. Dies führt dazu, daß mehrere mögliche totale Ordnungen von Planteilen in einem Plan zusammengefaßt werden können, so daß jede mögliche totale Ordnung auf ihre Ausführbarkeit hin überprüft werden muß. Da dies ein aufwendiger Prozeß ist, wurde untersucht, wie sich die Anzahl dieser Tests minimieren läßt.

Neben dieser Erweiterung des RAT-Formalismus wurde die graphische Oberfläche des Systems auf CLIM 2.0 portiert und stark erweitert. Die graphische Schnittstelle bietet nun beispielsweise mehrere Arten der Visualisierung verschiedener temporaler Aspekte der Pläne. Zur Verbesserung der Orientierung innerhalb großer Terminologien wurde außerdem ein Navigationsfenster in die Oberfläche integriert, das die gesamte Hierarchie in verkleinertem Maßstab darstellt und mit dem sich der dargestellte Ausschnitt verändern läßt.

Im Rahmen eines weiteren Fortgeschrittenenpraktikums wurde die Implementierung der Sprache ALCP durchgeführt. ALCP ist als Erweiterung terminologischer Logiken konzipiert und gestattet die Repräsentation und Verarbeitung unsicheren Wissens. Zusammen mit dem mit terminologischen Systemen verarbeitbaren begriffsbezogenen Wissen lassen sich damit in ALCP große Teile des in konkreten Anwendungen auftretenden Wissens modellieren. Das System ALCP wurde in das im *PPP*-Projekt als Kernmodul verwendete RAT-System integriert und kann ebenfalls über die erweiterte CLIM-Oberfläche angesprochen werden.

## 8.3 Publikationen

[AHR94a] ANDRÉ, E., HERZOG, G. und RIST, T.: *Multimedia Presentation of Interpreted Visual Data*. In: *Proceedings of AAAI-94 Workshop on the Integration of Natural Language and Vision Processing*, Seattle, USA, 1994.



- 
- [AHR94b] ANDRÉ, E., HERZOG, G. und RIST, T.: *Von der Bildfolge zur multimedialen Präsentation*. In: *Integration von Bild, Modell und Text '95*, Nummer 46, ASIM, Technische Universität Wien, 1995.
- [AR94a] ANDRÉ, E. und RIST, T.: *Intellimedia-Präsentationssysteme als Komponenten von Angebotssystemen*. In: KUNZE, J. und STOYAN, H. (Herausgeber): *KI-94 Workshop: Angebotssysteme mit wissensbasierten Komponenten*, Gesellschaft für Informatik e.V., 1994.
- [AR94b] ANDRÉ, E. und RIST, T.: *Referring to World Objects with Text and Pictures*. In: *Proc. of the 15<sup>th</sup> COLING*, Band 1, Kyoto, Japan, 1994.
- [AR94c] ANDRÉ, E. und T. RIST: *Presenting Information Employing Textual and Visual Material*. *Artificial Intelligence Review* 9 (2), Special Volume on the Integration of Natural Language and Vision Processing, 1994.
- [AR94d] ANDRÉ, E. und T. RIST: *Multimedia Presentations: The Support of Passive and Active Viewing*. In: *Working Notes of the AAAI Spring Symposium on Intelligent Multi-Media Multi-Modal Systems*, Stanford University, USA, 1994.
- [Hei94a] HEINSOHN, J.: *Probabilistic Description Logics*. In: *Proceedings of the 10th Conference on Uncertainty in Artificial Intelligence*, Seattle, Washington, 1994.
- [Hei94b] HEINSOHN, J.: *The Semantics of Imprecision in Terminological Logics*. In: KRUSE, R., GEBHARDT, J. und PALM, R. (Herausgeber): *Fuzzy Systems in Computer Science*, Artificial Intelligence, Vieweg, Wiesbaden, 1994.
- [Hei94c] HEINSOHN, J.: *ALCP – Ein hybrider Ansatz zur Modellierung von Unsicherheit in terminologischen Logiken*, DISKI-55, INFIX-Verlag, Sankt Augustin, 1994.
- [RKSZ94] RIST, T., T. KRÜGER, G. SCHNEIDER und D. ZIMMERMANN: *AWI - A Workbench for Semi-Automated Illustration Design*. In: *Advanced Visual Interfaces (Proceedings of AVI '94)* Bari, Italy, 1994.

#### 8.4 Sonstige Ereignisse

E. André wurde zur europäischen Vertreterin der ACL Special Interest Group on Multimedia Language Processing (SIGMEDIA) gewählt. J. Heinsohn hatte zum 1. September einen Ruf auf eine Informatik-Professur an der FH Brandenburg erhalten. In eingeschränktem Maß stand er

---

---

der DFKI GmbH zur Fortsetzung bereits begonnener Arbeiten zur Verfügung.

## 9 RAP: Reasoning About Plans

### 9.1 Kurzüberblick und Projektfortschritt

Im Projekt RAP (BMBF Förderkennzeichen ITW 9404 0) wird eine logikbasierte Planungs-umgebung entwickelt. Sie umfaßt ein Werkzeug zur Domänenmodellierung, eine Menge von Basis-Inferenzkomponenten sowie eine Shell, mit deren Hilfe die Inferenzkomponenten flexibel zu komplexen anwendungsspezifischen Plan-Inferenzsystemen kombiniert werden können.

Im Berichtszeitraum konzentrierten sich die Arbeiten auf die Erarbeitung einer gemeinsamen formalen Basis für die Inferenzkomponenten, ihre Weiter- und Neuentwicklung sowie die Implementierung eines ersten Prototyps der Shell zur Konfigurierung komplexer Plan-Inferenzsysteme. Im einzelnen wurden folgende Ziele erreicht. Das Plankonzept wurde um zusätzliche Abstraktionsgrade und Nebenläufigkeit erweitert. Der Planvalidierungsmodul wurde implementiert. Der Prototyp eines interaktiven Werkzeuges zur Erstellung und Inspektion von Planbibliotheken wurde fertiggestellt und getestet. Es wurde ein neues Konzept zur Planmodifikation erarbeitet. Zur Realisierung der RAP-Shell wurde der erste Prototyp eines Modulkombinationswerkzeuges implementiert; er erlaubt den Aufbau komplexer Modulnetze durch graphisches Programmieren auf sehr abstraktem Niveau. Der Ansatz zur Behandlung von Unsicherheit in der Planerkennung wurde auf die Verarbeitung indirekter Beobachtungen erweitert. Die Entwicklung und Implementierung einer Testumgebung für approximative Algorithmen in der Dempster-Shafer-Theorie wurden abgeschlossen.

### 9.2 Wissenschaftliche Ergebnisse

Ziel des Projektes ist die Entwicklung einer logikbasierten Planungs-umgebung, die verschiedene, bereichsunabhängige Inferenzdienste für Pläne zur Verfügung stellt. Eine Shell ermöglicht die flexible Kombination dieser Dienste derart, daß komplexe, anwendungsspezifische Inferenzsysteme leicht realisiert werden können.

#### 9.2.1 Der Repräsentationsformalismus

Basis aller Inferenzkomponenten ist ein gemeinsamer logischer Repräsentationsformalismus. Hierzu muß die modale temporale Planungslogik LLP um die Konzepte *Nebenläufigkeit in Plänen* und *Repräsentation von Zeit* erweitert werden. Zusätzlich hat sich herausgestellt, daß der LLP-Formalismus, der speziell zur Modellierung von Kommandosprachenumgebungen entwickelt wurde, zur Modellierung von "Realwelt"-Szenarien nicht in gleichem Maße geeignet ist. Er legt die Repräsentation des Anwendungsbereiches als Datenstruktur zugrunde und formuliert die Änderungen, die durch Basisaktionen hervorgerufen werden, als Zuweisungen. Diese Darstellung bedeutet für Anwendungsszenarien, die außerhalb der bisher betrachteten Softwareumgebungen liegen, häufig eine unnatürliche Strukturierung. Um auch hier eine adäquate Domänenmodellierung zu ermöglichen, müssen Basisaktionen *Relationen* zwischen *unstrukturierten* Objekten verändern können. Da das RAP-System als eine Plattform für die formale Behandlung von Plänen unterschiedlichster Anwendungen dienen soll, muß der logische Repräsentationsformalismus auch dieser Anforderung gerecht werden.

Daher wurde eine hybride Planungslogik entwickelt. Sie kombiniert eine Variante der Dynamischen Logik mit der temporalen Logik LLP und stellt so einen zweistufigen Repräsentationsformalismus zur Verfügung. Komplexe Basisaktionen werden durch Programme der Dynamischen Logik beschrieben und auf der Ebene der Temporallogik zu Plänen kombiniert. Basisaktionen verwenden elementare *add-*, *delete-* und *update-*Operationen und ermöglichen damit sowohl die

Änderung strukturierter Objekte, als auch die Änderung von Relationen zwischen (unstrukturierten) Objekten. Geeignete Kontrollstrukturen, u.a. auch Rekursion und indeterministische Verzweigung, kombinieren diese Elementaroperationen zu komplexen Basisaktionen. Diese werden auf der Stufe der Temporallogik, wiederum mit Hilfe entsprechender Kontrollstrukturen, zu komplexen Plänen verknüpft. Dabei wurde LLP um neue Kontrollstrukturen erweitert. Die Semantik der hybriden Planungslogik kombiniert die *state transition* Semantik der Dynamischen Logik mit der *execution sequence* Semantik von LLP auf natürliche Weise und erlaubt eine flexible

Deutung gewisser Funktions- und Prädikatsymbole.

Im Bereich der Plangenerierung wurde eine Weiterentwicklung des in der Logik LLP vorhandenen Plankonzeptes vorgenommen. Pläne werden dabei allgemein angesehen als zeitlogische Formeln, die eine Menge von konkreten Aktionsfolgen kompakt beschreiben. Dabei ist es nicht notwendig, daß Pläne *explizit* Aktionen benennen, die in ihnen enthalten sind, sondern die erlaubten Aktionen können auch implizit durch die Formulierung entsprechender Randbedingungen im Plan bestimmt sein. Man erreicht damit einen höheren Abstraktionsgrad in Plänen, der z.B. für den Plankonsumenten "Planerkennung" Vorteile bringt: Dadurch, daß mit einem einzigen Plan eine größere Menge von konkreten Aktionsfolgen beschrieben werden kann, kann somit auch die Erkennungsrate beim Planerkenner erhöht werden.

Die Kontrollstruktur "temporale Abstraktion" zur Formulierung unterbrechbarer Pläne wurde bisher nur in Verbindung mit linearen Plänen verwendet. Als Beitrag zur Erweiterung der Beschreibung von Nebenläufigkeit in Plänen – im wesentlichen dargestellt durch Nichtlinearität – kann nun auch das Konzept der Unterbrechbarkeit verwendet werden, d.h. nichtlineare Pläne dürfen durch nicht explizit spezifizierte Aktionen unterbrochen werden. Zusätzlich ist es möglich, die Planunterbrechungen sowohl quantitativ als auch qualitativ einzuschränken.

### 9.2.2 Die Entwicklungsumgebung

Eine der wesentlichen Aufgaben im Berichtszeitraum war die Spezifikation der prozeduralen Basis der zu entwickelnden RAP-Shell, mit deren Hilfe ein Anwender in die Lage versetzt werden soll, verschiedene Inferenzmechanismen zu komplexen Anweisungsfolgen zu kombinieren. Da die grafikunterstützte Eingabe solcher *Modulnetze* auf einem möglichst abstrakten Niveau erfolgen soll, um technische Details vom Anwender fernzuhalten, war es notwendig, einen zentralen Kontrollmechanismus zu entwickeln, der die fehlerfreie Abarbeitung auch komplexer Anwendungen garantiert. Die Entwicklung der entsprechenden Algorithmen, Daten- und Kontrollstrukturen bildete hierbei einen Schwerpunkt dieses Arbeitspaketes. Auf dieser Basis wurde der erste Prototyp eines *Modulkombinationswerkzeuges* implementiert (Arbeitspakete INT-REAS und SYS-INT). Ein Anwender hat damit die Möglichkeit, aus elementaren Modulen grafikunterstützt *Modulnetze* aufzubauen. Dieses *grafische Programmieren* auf sehr abstraktem Niveau erlaubt das Fernhalten technischer Details vom Anwender. Da im Berichtszeitraum die Anpassung der sich in der Weiterentwicklung befindenden RAP-Inferenzdienste für die Integration in die RAP-Shell noch nicht vorgesehen war, wurde der erste Prototyp mit Testmodulen aus anderen Domänen wie zum Beispiel der Arithmetik erfolgreich getestet. Dabei stand die Evaluierung des zentralen Kontrollmechanismus der RAP-Shell im Vordergrund.

Eine zentrale Komponente des RAP-Systems ist das *Domain Modeling Tool* (DMT), das Benutzer bei der Modellierung ihrer Anwendungsdomäne unterstützt, ohne sie dabei mit dem logischen Basisformalismus zu konfrontieren. Im Berichtszeitraum wurde die graphische Eingabeschnittstelle des DMT spezifiziert. Sie präsentiert dem Benutzer Definitionsschemata, mit deren Hilfe Objekttypen, Aktionen und Pläne der Domäne spezifiziert werden. Eine wichtige Rolle spielt

dabei die Spezifikation der Schnittstelle zum Anwendungssystem, über die bei Bedarf aktuelle Systemparameter ermittelt werden können.

### 9.2.3 Die Inferenzdienste

Der Prototyp eines interaktiven Werkzeugs zur Erstellung und Inspektion von Planbibliotheken innerhalb des RAP-Systems wurde fertiggestellt und getestet. Das Werkzeug ermöglicht es, die Indexierungsstruktur einer Planbibliothek graphisch zu veranschaulichen. Weiterhin unterstützt es die Operationen auf einer Planbibliothek, wie zum Beispiel das Initialisieren, Löschen bzw. Aufnehmen neuer Planeinträge mit automatischer Indexierung und die Informationsbereitstellung zu Planeinträgen.

Die Indexierung von Plänen in Bibliotheken innerhalb von RAP erfolgt durch ein spezielles Kodierungsverfahren [Köh94a]. Dieses Verfahren besteht aus einem bereichsunabhängigen Abstraktionsteil sowie einem domänenspezifischen Abbildungsteil. Die für den domänenspezifischen Teil benötigte Information soll in RAP durch das Domain Modeling Tool bereitgestellt werden. Es wurde deshalb eine Abstimmung zwischen Domain Modeling Tool und Planbibliothek vorgenommen und die Schnittstelle zwischen beiden Komponenten konzipiert.

Im Berichtszeitraum wurde ein neuer Ansatz zur Planmodifikation konzipiert. Zunächst wurde das in [Köh94d] entwickelte formale Modell der Planmodifikation im Hinblick auf die neuen Anforderungen in RAP evaluiert (Arbeitspaket MOD-EV). Prinzipiell ist es möglich, dieses Modell aufgrund seiner Allgemeinheit unverändert beizubehalten. In PHI wurde jedoch bereits festgestellt, daß die Modifikation von Plänen mit Kontrollstrukturen aufgrund komplexitätstheoretischer Eigenschaften der Planmodifikation zu Effizienzverlusten im Vergleich zur Plangenerierung führen kann. Da in RAP zusätzlich nebenläufige Pläne auftreten werden, wurde mit der Entwicklung eines neuen Konzepts zur Planmodifikation begonnen (Arbeitspaket MOD-TH).

Die Grundidee dieses Ansatzes besteht in der Definition sogenannter "*local repairs*", die es ermöglichen, Operatoren in Pläne einzufügen, aus Plänen zu löschen oder durch andere Operatoren auszutauschen, ohne die Ausführbarkeit der übrigen Operatoren zu beeinträchtigen. Als Vorstufe nebenläufiger Pläne wurden zunächst nichtlineare Pläne betrachtet.

Das Planungssystem berechnet dabei zunächst die Abweichung eines gegebenen Plans vom gewünschten Ziel- oder Ausgangszustand. Diese Abweichung definiert eine neue Planungsaufgabe, d.h. mit Hilfe des Plangenerierers wird ein Reparaturplan "*local repair*" berechnet, der die Abweichung korrigiert. Die Korrektheit der "*local repairs*" wird durch die Überprüfung von Operatorinteraktionen gesichert. Es wurden 3 verschiedene Arten der Operatorinteraktion definiert, die in sogenannten *Operator-Abhängigkeitsgraphen* repräsentiert werden.

Der wesentliche Unterschied des neuen Verfahrens im Vergleich zur allgemeinen Planmodifikation besteht in der Lokalitätseigenschaft der Modifikationsoperationen. Diese Eigenschaft macht es unnötig, einen modifizierten Plan noch einmal vollständig zu verifizieren. Darüberhinaus kann die Lokalität effizient durch Analyse der Operator-Abhängigkeitsgraphen überprüft werden. Ein weiterer Vorteil des neuen Ansatzes liegt in der Möglichkeit, fehlerhafte Pläne während der Planausführung zu reparieren.

Ein offenes Problem ist die effiziente Berechnung der Position innerhalb eines nichtlinearen bzw. nebenläufigen Plans, in der die Reparatur ausgeführt werden kann. Dieses Problem erfordert die Behandlung von temporalen und strukturalen Partitionsproblemen. Lösungen dieser Probleme sind aus der Literatur nicht bekannt.

Im Bereich der Planvalidierung (Arbeitspaket REA-VAL), deren Aufgabe es ist, zu entscheiden, ob ein gegebener Plan in einem bestimmten Zustand des Anwendungssystems ausführbar ist, wurden neue Sequenzenregeln entwickelt. Damit konnte eine Beweistaktik zur Validierung linea-

rer Pläne implementiert werden. Zwei deduktive Strategien wurden für den Validierungsprozeß spezifiziert: eine unter dem Einsatz von Regression, die zweite unter Verwendung von Progression. Die Regressionsstrategie erzeugt dabei für einen Plan die zu seiner Ausführung notwendigen *schwächsten Vorbedingungen*. Vorteile und Nachteile beider Verfahren wurden lokalisiert. Die Validierung nichtlinearer Pläne wurde zugunsten der Arbeitspakete INT-REAS und SYS-INT zurückgestellt, um möglichst früh zu erkennen, welche Anforderungen durch das prototypisch implementierte Modulkombinationswerkzeug an die einzelnen Inferenzmechanismen gestellt werden.

Die Realisierung des Planerkenners als Stand-Alone System PLATO (**P**lan Recognition in a Modal Temporal Logic) wurde fertiggestellt. Damit wurde die Basis gelegt für eine Integration der Planerkennung als ein Inferenzprozeß innerhalb der RAP-Shell.

Neben der Arbeit an den Inferenzprozessen wurde mit der theoretischen Formalisierung einiger Demonstratorszenarien für RAP begonnen. Dies beinhaltet das Bestimmen der benötigten Anwendungsprogramme und ihrer Funktionalität. Als vorteilhaft erweisen sich die Verwendung der objektorientierten Bibliothek XPCE zur Entwicklung der Benutzeroberfläche der RAP-Shell und einiger mit Hilfe von XPCE erstellter interaktiver Anwendungsprogramme.

Im Bereich der Behandlung von Unsicherheit stellt sich das Problem der gegenüber dem Vorgängerprojekt erheblich größeren Datenmengen etwa im Bereich der Planerkennung, die nun auch

- [Bau94c] BAUER, M.: *Uncertainty in Plan Recognition—Extended Abstract*. In: *KI-94 Workshops, Extended Abstracts*, Gesellschaft für Informatik, 1994.
- [BSW94] BIUNDO, S., W. STEPHAN und A. WOLPERS: *Formalizing Scenarios for Reactive Plans*. In: *Proceedings of the ECAI'94 Workshop "Logic and Change"*, Wiley & Sons., Chichester, England, 1994.
- [Den94a] DENGLER, D.: *An Adaptive Deductive Planning System*. In: *Proceedings of the 11th European Conference on Artificial Intelligence (ECAI-94)*, Wiley & Sons, Chichester, England, 1994.
- [Den94b] DENGLER, D.: *Generating Consumer-Tailored Plans*. In: *Proceedings of the 4th International Conference on User Modeling (UM-94)*, MITRE corporation, 1994.
- [Köh94a] KÖHLER, J.: *An Application of Terminological Logics to Case-based Reasoning*. In: *Proceedings of the 4th International Conference on Principles of Knowledge Representation and Reasoning*, Morgan Kaufmann, San Francisco, CA, 1994.
- [Köh94b] KÖHLER, J.: *Avoiding Pitfalls in Case-based Planning*. In: *Proceedings of the 2nd International Conference on Artificial Intelligence Planning Systems*, Chicago, IL, 1994. AAAI Press, Menlo Park, 1994.
- [Köh94d] KÖHLER, J.: *Wiederverwendung von Plänen in deduktiven Planungssystemen*, DISKI-65, INFIX-Verlag, Sankt Augustin, Germany, 1994.
- [Köh94e] KÖHLER, J.: *Correct Modification of Complex Plans*. In: *Proceedings of the 11th European Conference on Artificial Intelligence (ECAI-94)*, Wiley & Sons, Chichester, 1994.

#### 9.4 Sonstige Ereignisse

Dietmar Dengler und Dr. Jana Köhler wurden am 9. August 1994 mit dem *Digital Equipment Preis* der European Conference on Artificial Intelligence (ECAI-94) ausgezeichnet. Der Vorsitzende des Programmkomitees dieser bedeutenden internationalen Tagung, Prof. Dr. Antony Cohn, betonte in seiner Laudatio, daß diese Auszeichnung gleichzeitig als Würdigung der wissenschaftlichen Arbeit des gesamten Projektes zu verstehen sei.

## 10 VEGA: Validierung und Exploration von Wissen durch Globale Analyse

### 10.1 Kurzüberblick und Projektfortschritt

In dem Projekt VEGA wird ein System zur Validierung und Exploration von Wissensbasen entwickelt.

### 10.2.1 Die Evolution von Wissensbasen

Die Anpassung von Wissensbasen an neue Anforderungen ist beim Einsatz und bei der Kooperation wissensbasierter Systeme von großer Bedeutung. Es wurde detailliert untersucht, inwiefern Verfahren aus verschiedenen Forschungsgebieten für diese sogenannte *Wissensevolution* eingesetzt werden können. Unter anderem wurde eine partielle Generalisierung entwickelt, die Unifikation und Anti-Unifikation kombiniert, und so die Verallgemeinerung beschränkt.

Die Abduktion spielt auch beim Finden von Unzulänglichkeiten in Wissensbasen eine wichtige Rolle. Hierfür wurde eine Transformation logischer Programme entwickelt, die Methoden zur Beantwortung von Anfragen an deduktive Datenbanken für die Abduktion erweitert. Daneben wurde eine Transformation entworfen, die zur Konsistenzprüfung nach einer Änderung der Wissensbasis eingesetzt werden kann. Die Transformation für diese als *Konsequenzfindung* bezeichnete Inferenz wurde im Rahmen einer Projektarbeit implementiert. Für die effiziente Auswertung logischer Programme wurden Transformationen entwickelt, die die redundante mehrfache Herleitung gleicher Fakten vermeiden.

Im Kontext der Wissensevolution eignen sich *induktive Verfahren*, um neues Wissen bzw. eine neue, kompaktere Repräsentation aus vorhandenen Wissens- bzw. Datenbeständen herzuleiten. Dazu wurde ein induktives System entwickelt, das zu einem gegebenen Literal entsprechende Regeln generiert, die dieses Literal bezüglich eines bestehenden Datenbestandes definieren. Diese und weitere Techniken der Wissensevolution wurden auch zusammen mit der Thematik eines universellen Wissensaustauschformats im Rahmen des Concurrent Engineering diskutiert.

Desweiteren wurde im Berichtszeitraum mit der Klassifikation von Explorationstools durch eine Betrachtung auf der *Wissensebene* (Knowledge level modelling) begonnen.

Nach der Untersuchung und Entwicklung einzelner Verfahren wurde im Berichtszeitraum die Architektur des Gesamtsystems entworfen und mit der Implementierung des Wissensbank-Servers begonnen. Daneben wurde eine Oberfläche konzipiert, die es dem Benutzer erlaubt, die Evolution durch Manipulation graphischer Objekte zu steuern.

### 10.2.2 Deklarative Wissensrepräsentation

Die deklarativen Formalismen zur Wissensrepräsentation, die in dem Project ARC-TEC entwickelt wurden, wurden von der DRL-Gruppe aufgegriffen und entsprechend erweitert. So wurde das relational/funktionale System RELFUN um einen Sortenverband erweitert. Ein Modulkonzept zur besseren Wartbarkeit und Wiederverwendung wurde entworfen. Desweiteren wurde eine Spezifikation für KES-orientierte DRL-Erweiterungen erarbeitet und ihre Implementation in Angriff genommen.

### 10.2.3 Erstellung einer Anwendungswissensbasis

In Zusammenarbeit mit dem Institut für Verbundwerkstoffe, GmbH (IVW) wurde im vorhergehenden Berichtszeitraum im Rahmen des IMCOD-Projekts an dem Aufbau einer Produkt- und Produktionswissensbasis für neue Fertigungsmaterialien (Verbundwerkstoffe) gearbeitet. In diesem Berichtszeitraum wurde nun diese RPPP-Wissensbasis modifiziert und erweitert. So wurde die Materialwissensbasis RTPLAST nach dem in der DRL-Gruppe erarbeiteten Sortenprinzip umstrukturiert. In Ergänzung zu dieser Materialwissensbasis wurde eine Produkt- und Produktionswissensbasis erarbeitet. Die RPPP-Wissensbasis besteht somit jetzt aus drei Komponenten: Der Materialwissensbasis RTPLAST, der Produktwissensbasis zur Parameterspezifikation bei der Fertigung von Schalensitzen mit Verbundwerkstoffen und einer entsprechenden Produktwissensbasis, die man auch als Fallbasis bezeichnen kann.



Die RPPP-Wissensbasis und eine Beschreibung der Erfahrungen mit dem WWW

wurde dann auf dem World Wide Web (WWW) veröffentlicht.

### 10.3 Publikationen

- [AJS94] J. S. AITKEN, L. E. BOURNE JR. und F. SCHMALHOFER: *Descriptions of rational behaviour for reuse: the Sisyphus example*. In: H. VOSS, R. STUDER: *Proceedings of the 4th KADS Meeting, GMD, Sankt Augustin, March, 1994*, Januar 1994.
- [BBK94] H. BOLEY, U. BUHRMANN und C. KREMER: *Towards a Sharable Knowledge Base on Recyclable Plastics*, Januar 1994. To appear in: *TMS'94 Symposium on Knowledge-Based Applications in Material Science and Engineering*, Feb/Mar 1994, San Francisco, USA.
- [BH94] G. BURGUN und K. HINKELMANN: *Knowledge-base rewriting for bottom-up abduction*. In: C. BECKSTEIN und (Herausgeber: *Entwicklung, Test und Wartung deklarativer KI-Programme, Workshop auf der 18. Jahrestagung für Künstliche Intelligenz (KI-94)*), GMD-Studien Nr. 238, 1994.
- [Bol94] H. BOLEY: *Finite Domains and Exclusions as First-Class Citizens*. In: R. DYCKHOFF (Herausgeber): *Fourth International Workshop on Extensions of Logic Programming, St. Andrews, Scotland, 1993, Preprints and Proceedings*, Lecture Notes in Artificial Intelligence, Springer-Verlag, Berlin, 1994.
- [BTS<sup>+</sup>94] A. BIRK, B. TSCHAITSCHIAN, F. SCHMALHOFER, M. THÜRING und H. GERTZEN: *Architecture and User-Interface of the IDEAS Intelligent Documentation System*. In: W. SCHULER, J. HANNEMANN und N. STREITZ (Herausgeber): *Designing User Interfaces for Hypermedia*. Springer-Verlag, Berlin, 1994.
- [HH94] K. HINKELMANN und H. HINTZE: *Computing Cost Estimates for Proof Strategies*. In: R. DYCKHOFF (Herausgeber): *Extensions of Logic Programming, 4th International Workshop ELP '93, St Andrews, U.K., March/April 1993, Proceedings*, Lecture Notes in Artificial Intelligence 798, Springer-Verlag, Berlin, 1994.
- [Hin94] K. HINKELMANN: *A Consequence-Finding Approach for Feature Recognition in CAPP*. In: *Seventh International Conference on Industrial & Engineering Applications of Artificial Intelligence & Expert Systems (IEA/AIE'94)*, Austin, Texas, Gordon and Breach Science Publishers, 1994.

- [KK94] O. KÜHN und D. KRECHEL: *Validating DATALOG Design Rules Including Finite Domains and Intervals*. In: *Entwicklung, Test und Wartung deklarativer KI-Programme (Verfahren und Werkzeuge), Workshop auf der 18. Fachtagung für Künstliche Intelligenz (KI-94)*, 1994.
- [KM94] C. KLAUCK und J. MAUSS: *Feature Recognition in Computer-Integrated Manufacturing*. *Integrated Computer-Aided Engineering: AI in Manufacturing and Robotics*, 1994.
- [Küh94] O. KÜHN: *An Ontology for the Conservation of Corporate Knowledge about Crankschaft Design*. *Proceedings of the ECAI'94 workshop on Comparison of Implemented Ontologies*, Amsterdam, 1994.
- [Mey94c] M. MEYER: *Using Hierarchical Constraint Satisfaction for Lathe-Tool Selection in a Computer-Integrated Manufacturing Environment*. *Integrated Computer-Aided Engineering*, 1 (3), 1994.
- [Mey94d] M. MEYER: *Issues in Concurrent Knowledge Engineering: Knowledge Sharing and Knowledge Evolution*. In: M. Sobolewski (Herausgeber): *Proceedings First International Conference on Concurrent Engineering, Research and Applications (CERA'94)*, Pittsburgh. IEEE Computer Press, 1994.
- [MM94] M. MEYER und J. MÜLLER: *Solving Configuration Problems Efficiently using Finite Domain Consistency Techniques*. In: *The International Journal of Applied Intelligence*. Kluwer Acad. Publ., 1994.
- [NPS94] S. NEUBERT, T. PIRLEIN und G. SCHMIDT: *Top-Down Knowledge Acquisition*. *Proceedings of International Conference on Expert Systems for Development*, 1994.
- [RS94] T. REINARTZ und F. SCHMALHOFER: *An Integration of Knowledge Acquisition Techniques and EBL for Real-World Production Planning*, *Knowledge Acquisition*, Vol. 6, Juni 1994.
- [SATvE94] FRANZ SCHMALHOFER, J. STUART AITKEN, BIDJAN TSCHAITSCHIAN und LUDGER VAN ELST: *User-guided machine learning for generating abstractions*. In: *ECAI' 94 Workshop on Intergration of knowledge aquisition and Machine Learning*, Amsterdam, 1994.
- [Sch94a] FRANZ SCHMALHOFER: *A comparison of case-based and comprehension-based architectures*. In: J. KUNZE und (btxeditorslong): *KI-94 Workshops*, 1994.
- [Sch94b] FRANZ SCHMALHOFER: *Expert systems as intelligent documentation and communication tools*. In: A. LESGOLD und F. SCHMALHOFER (Herausgeber: *Expert and Tutoring Systems as Media for Embodying and Sharing Knowledge*, Dagstuhl Seminar Report 9431, Dagstuhl, Germany, 1994.
- [Sch94c] FRANZ SCHMALHOFER: *Unifying knowledge acquisition and machine learning for applications*, *MLnet Workshop: Industrial Applications of Machine Learning*, 1994.
- [Sch94f] MICHAEL SCHMIDT: *Generierung von Regelhypothesen aus relationalen Datenbanken*. In: *Entwicklung, Test und Wartung deklarativer KI-Programme (Verfahren und Werkzeuge), Workshop auf der 18. Fachtagung für Künstliche Intelligenz' (KI-94)*, Saarbrücken, Germany, 1994.

- [STR94] FRANZ SCHMALHOFER, BIDJAN TSCHAITSCHIAN und THOMAS REINARTZ: *A cognitive science approach to learning in complex real world domains*. In: R. BISDORFF (Herausgeber) *Proceedings of the First European Conference Cognitive Science in Industry*, Luxembourg, 1994.



Deutsches  
Forschungszentrum  
für Künstliche  
Intelligenz GmbH

---

-Bibliothek, Information  
und Dokumentation (BID)-  
PF 2080  
67608 Kaiserslautern  
FRG

---

---

Telefon (0631) 205-3506  
Telefax (0631) 205-3210  
e-mail  
dfkibib@dfki.uni-kl.de  
WWW  
http://www.dfki.uni-  
sb.de/dfkibib

---

## Veröffentlichungen des DFKI

Die folgenden DFKI Veröffentlichungen sowie die aktuelle Liste von allen bisher erschienenen Publikationen können von der oben angegebenen Adresse oder (so sie als per ftp erhältlich angemerkt sind) per anonymous ftp von ftp.dfki.uni-kl.de (131.246.241.100) im Verzeichnis 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 (if they are marked as obtainable by ftp) by anonymous ftp from ftp.dfki.uni-kl.de (131.246.241.100) in the directory pub/Publications.*

*The reports are distributed free of charge except where otherwise noted.*

---

## DFKI Research Reports

### 1996

#### RR-96-03

Günter Neumann

Interleaving

Natural Language Parsing and Generation  
Through Uniform Processing

51 pages

#### RR-96-02

E.André, J. Müller, T.Rist:

PPP-Persona: Ein objektorientierter Multimedia-Präsentationsagent

14 Seiten

### 1995

#### RR-95-20

Hans-Ulrich Krieger

Typed Feature Structures, Definite Equivalences,  
Greatest Model Semantics, and Nonmonotonicity

27 pages

#### RR-95-19

Abdel Kader Diagne, Walter Kasper, Hans-Ulrich Krieger

Distributed Parsing With HPSG Grammar

20 pages

#### RR-95-18

Hans-Ulrich Krieger, Ulrich Schäfer

Efficient Parameterizable Type Expansion for Typed  
Feature Formalisms

19 pages

#### RR-95-17

Hans-Ulrich Krieger

Classification and Representation of Types in TDL

17 pages

#### RR-95-16

Martin Müller, Tobias Van Roy

Title not set

0 pages

**Note:** The author(s) were unable to deliver this document for printing before the end of the year. It will be printed next year.

#### RR-95-15

Joachim Niehren, Tobias Van Roy

Title not set

0 pages

**Note:** The author(s) were unable to deliver this document for printing before the end of the year. It will be printed next year.

#### RR-95-14

Joachim Niehren

Functional Computation as Concurrent Computation

50 pages

- RR-95-13**  
*Werner Stephan, Susanne Biundo*  
 Deduction-based Refinement Planning  
 14 pages
- RR-95-12**  
*Walter Hower, Winfried H. Graf*  
 Research in Constraint-Based Layout, Visualization,  
 CAD, and Related Topics: A Bibliographical Survey  
 33 pages
- RR-95-11**  
*Anne Kilger, Wolfgang Finkler*  
 Incremental Generation for Real-Time Applications  
 47 pages
- RR-95-10**  
*Gert Smolka*  
 The Oz Programming Model  
 23 pages
- RR-95-09**  
*M. Buchheit, F. M. Donini, W. Nutt, A. Schaerf*  
 A Refined Architecture for Terminological Systems:  
 Terminology = Schema + Views  
 71 pages
- RR-95-08**  
*Michael Mehl, Ralf Scheidhauer, Christian Schulte*  
 An Abstract Machine for Oz  
 23 pages
- RR-95-07**  
*Francesco M. Donini, Maurizio Lenzerini, Daniele Nardi, Werner Nutt*  
 The Complexity of Concept Languages  
 57 pages
- RR-95-06**  
*Bernd Kiefer, Thomas Fettig*  
 FEGRAMED  
 An interactive Graphics Editor for Feature Structures  
 37 pages
- RR-95-05**  
*Rolf Backofen, James Rogers, K. Vijay-Shanker*  
 A First-Order Axiomatization of the Theory of Finite  
 Trees  
 35 pages
- RR-95-04**  
*M. Buchheit, H.-J. Bürckert, B. Hollunder, A. Laux, W. Nutt, M. Wójcik*  
 Task Acquisition with a Description Logic Reasoner  
 17 pages
- RR-95-03**  
*Stephan Baumann, Michael Malburg, Hans-Guenther Hein, Rainer Hoch, Thomas Kieninger, Norbert Kuhn*  
 Document Analysis at DFKI  
 Part 2: Information Extraction  
 40 pages
- RR-95-02**  
*Majdi Ben Hadj Ali, Frank Fein, Frank Hoenes, Thorsten Jaeger, Achim Weigel*  
 Document Analysis at DFKI  
 Part 1: Image Analysis and Text Recognition  
 69 pages
- RR-95-01**  
*Klaus Fischer, Jörg P. Müller, Markus Fischel*  
 Cooperative Transportation Scheduling  
 an application Domain for DAI  
 31 pages
- 1994**
- RR-94-39**  
*Hans-Ulrich Krieger*  
 Typed Feature Formalisms as a Common Basis for Linguistic Specification.  
 21 pages
- RR-94-38**  
*Hans Uszkoreit, Rolf Backofen, Stephan Busemann, Abdel Kader Diagne, Elizabeth A. Hinkelman, Walter Kasper, Bernd Kiefer, Hans-Ulrich Krieger, Klaus Netter, Günter Neumann, Stephan Oepen, Stephen P. Spackman.*  
 DISCO—An HPSG-based NLP System and its Application for Appointment Scheduling.  
 13 pages
- RR-94-37**  
*Hans-Ulrich Krieger, Ulrich Schäfer*  
 TDL - A Type Description Language for HPSG, Part 1: Overview.  
 54 pages
- RR-94-36**  
*Manfred Meyer*  
 Issues in Concurrent Knowledge Engineering. Knowledge Base and Knowledge Share Evolution.  
 17 pages
- RR-94-35**  
*Rolf Backofen*  
 A Complete Axiomatization of a Theory with Feature and Arity Constraints  
 49 pages

- RR-94-34**  
*Stephan Busemann, Stephan Oepen, Elizabeth A. Hinkelman,*  
*Günter Neumann, Hans Uszkoreit*  
 COSMA – Multi-Participant NL Interaction for Appointment Scheduling  
 80 pages
- RR-94-33**  
*Franz Baader, Armin Laux*  
 Terminological Logics with Modal Operators  
 29 pages
- RR-94-31**  
*Otto Kühn, Volker Becker, Georg Lohse, Philipp Neumann*  
 Integrated Knowledge Utilization and Evolution for the Conservation of Corporate Know-How  
 17 pages
- RR-94-23**  
*Gert Smolka*  
 The Definition of Kernel Oz  
 53 pages
- RR-94-20**  
*Christian Schulte, Gert Smolka, Jörg Würtz*  
 Encapsulated Search and Constraint Programming in Oz  
 21 pages
- RR-94-19**  
*Rainer Hoch*  
 Using IR Techniques for Text Classification in Document Analysis  
 16 pages
- RR-94-18**  
*Rolf Backofen, Ralf Treinen*  
 How to Win a Game with Features  
 18 pages
- RR-94-17**  
*Georg Struth*  
 Philosophical Logics—A Survey and a Bibliography  
 58 pages
- RR-94-16**  
*Gert Smolka*  
 A Foundation for Higher-order Concurrent Constraint Programming  
 26 pages
- RR-94-15**  
*Winfried H. Graf, Stefan Neurohr*  
 Using Graphical Style and Visibility Constraints for a Meaningful Layout in Visual Programming Interfaces  
 20 pages
- RR-94-14**  
*Harold Boley, Ulrich Buhrmann, Christof Kremer*  
 Towards a Sharable Knowledge Base on Recyclable Plastics  
 14 pages
- RR-94-13**  
*Jana Koehler*  
 Planning from Second Principles—A Logic-based Approach  
 49 pages
- RR-94-12**  
*Hubert Comon, Ralf Treinen*  
 Ordering Constraints on Trees  
 34 pages
- RR-94-11**  
*Knut Hinkelmann*  
 A Consequence Finding Approach for Feature Recognition in CAPP  
 18 pages
- RR-94-10**  
*Knut Hinkelmann, Helge Hintze*  
 Computing Cost Estimates for Proof Strategies  
 22 pages
- RR-94-08**  
*Otto Kühn, Björn Höfling*  
 Conserving Corporate Knowledge for Crankshaft Design  
 17 pages
- RR-94-07**  
*Harold Boley*  
 Finite Domains and Exclusions as First-Class Citizens  
 25 pages
- RR-94-06**  
*Dietmar Dengler*  
 An Adaptive Deductive Planning System  
 17 pages
- RR-94-05**  
*Franz Schmalhofer, J. Stuart Aitken, Lyle E. Bourne jr.*  
 Beyond the Knowledge Level: Descriptions of Rational Behavior for Sharing and Reuse  
 81 pages
- RR-94-03**  
*Gert Smolka*  
 A Calculus for Higher-Order Concurrent Constraint Programming with Deep Guards  
 34 pages
- RR-94-02**  
*Elisabeth André, Thomas Rist*  
 Von Textgeneratoren zu Intellimedia-Präsentationssystemen  
 22 Seiten
- RR-94-01**  
*Elisabeth André, Thomas Rist*  
 Multimedia Presentations: The Support of Passive and Active Viewing  
 15 pages

---

## DFKI Technical Memos

1994

1995

**TM-95-04**

*Klaus Schmid*

Creative Problem Solving  
and

Automated Discovery

— An Analysis of Psychological and AI Research —

152 pages

**TM-95-03**

*Andreas Abecker, Harold Boley, Knut Hinkelmann, Holger Wache,*

*Franz Schmalhofer*

An Environment for Exploring and Validating Declarative Knowledge

11 pages

**TM-95-02**

*Michael Sintek*

FLIP: Functional-plus-Logic Programming  
on an Integrated Platform

106 pages

**TM-95-01**

*Martin Buchheit, Rüdiger Klein, Werner Nutt*

Constructive Problem Solving: A Model Construction  
Approach towards Configuration

34 pages

**TM-94-04**

*Cornelia Fischer*

PAnUDE - An Anti-Unification Algorithm for Expressing Refined Generalizations

22 pages

**TM-94-03**

*Victoria Hall*

Uncertainty-Valued Horn Clauses

31 pages

**TM-94-02**

*Rainer Bleisinger, Berthold Kröll*

Representation of Non-Convex Time Intervals and  
Propagation of Non-Convex Relations

11 pages

**TM-94-01**

*Rainer Bleisinger, Klaus-Peter Gores*

Text Skimming as a Part in Paper Document Understanding

14 pages

---

## DFKI Documents

1995

**D-95-12**

*F. Baader, M. Buchheit, M. A. Jeusfeld, W. Nutt (Eds.)*

Working Notes of the KI'95 Workshop:

KRDB-95 - Reasoning about Structured Objects:

Knowledge Representation Meets Databases

61 pages

**D-95-11**

*Stephan Busemann, Iris Merget*

Eine Untersuchung kommerzieller Terminverwaltungssoftware im Hinblick auf die Kopplung mit natürlichsprachlichen Systemen

32 Seiten

**D-95-10**

*Volker Ehresmann*

Integration ressourcen-orientierter Techniken in das wissensbasierte Konfigurierungssystem TOOCON

108 Seiten

**D-95-09**

*Antonio Krüger*

PROXIMA: Ein System zur Generierung graphischer Abstraktionen

120 Seiten

**D-95-08**

*Technical Staff*

DFKI Jahresbericht 1994

63 Seiten

**Note:** This document is no longer available in printed form.

**D-95-07**

*Ottmar Lutzy*

Morphic - Plus

Ein morphologisches Analyseprogramm für die deutsche Flexionsmorphologie und Komposita-Analyse

74 pages

**D-95-06**

*Markus Steffens, Ansgar Bernardi*

Integriertes Produktmodell für Behälter aus Faserverbundwerkstoffen

48 Seiten

**D-95-05**

*Georg Schneider*

Eine Werkbank zur Erzeugung von 3D-Illustrationen

157 Seiten

**D-95-04**

*Victoria Hall*

Integration von Sorten als ausgezeichnete taxonomische Prädikate in eine relational-funktionale Sprache  
56 Seiten

**D-95-03**

*Christoph Endres, Lars Klein, Markus Meyer*

Implementierung und Erweiterung der Sprache *ALCP*  
110 Seiten

**D-95-02**

*Andreas Butz*

*BETTY*

Ein System zur Planung und Generierung informativer Animationssequenzen  
95 Seiten

**D-95-01**

*Susanne Biundo, Wolfgang Tank (Hrsg.)*

PuK-95, Beiträge zum 9. Workshop „Planen und Konfigurieren“, Februar 1995  
169 Seiten

**Note:** This document is available for a nominal charge of 25 DM (or 15 US-\$).

**1994**

**D-94-15**

*Stephan Oepen*

German Nominal Syntax in HPSG

— On Syntactic Categories and Syntagmatic Relations

80 pages

**D-94-14**

*Hans-Ulrich Krieger, Ulrich Schäfer*

TDL - A Type Description Language for HPSG, Part

**D-94-10**

*F. Baader, M. Lenzerini, W. Nutt, P. F. Patel-Schneider (Eds.)*

Working Notes of the 1994 International Workshop on Description Logics  
118 pages

**Note:** This document is available for a nominal charge of 25 DM (or 15 US-\$).

**D-94-09**

*Technical Staff*

DFKI Wissenschaftlich-Technischer Jahresbericht 1993

145 Seiten

**D-94-08**

*Harald Feibel*

IGLOO 1.0 - Eine grafikunterstützte Beweisentwicklungsumgebung  
58 Seiten

**D-94-07**

*Claudia Wenzel, Rainer Hoch*

Eine Übersicht über Information Retrieval (IR) und NLP-Verfahren zur Klassifikation von Texten  
25 Seiten

**D-94-06**

*Ulrich Buhrmann*

Erstellung einer deklarativen Wissensbasis über recyclingrelevante Materialien  
117 Seiten

**D-94-04**

*Franz Schmalhofer, Ludger van Elst*

Entwicklung von Expertensystemen: Prototypen, Tiefenmodellierung und kooperative Wissensentwicklung  
22 Seiten

**D-94-03**

*Franz Schmalhofer*

72 pages

**D-94-12**

*Arthur Sehn, Serge Autexier (Hrsg.)*

Proceedings des Studentenprogramms der 18. Deutschen Jahrestagung für Künstliche Intelligenz KI-94

Betrachtung

54 Seiten

**Note:** This document is no longer available in printed form.





