



Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

**Research
Report**
RR-92-09

**Perspektiven zur Kombination von
automatischem Animationsdesign und
planbasierter Hilfe**

Winfried Graf, Markus A. Thies

Januar 1992

**Deutsches Forschungszentrum für Künstliche Intelligenz
GmbH**

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

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

Deutsches Forschungszentrum für Künstliche Intelligenz

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

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

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

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

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

Prof. Dr. Gerhard Barth
Director

Perspektiven zur Kombination von automatischem Animationsdesign und planbasierter Hilfe

Winfried Graf, Markus A. Thies

DFKI-RR-92-09

Die vorliegende Arbeit wurde im Rahmen des vom BMFT geförderten Projektes WIP (Wissensbasierte Informationspräsentation, FKZ ITW-8901 8) und des Kooperationsprojektes PLUS (PLan-based User Support) mit dem IBM Labor Böblingen und der IBM Deutschland GmbH am DFKI durchgeführt.

© Deutsches Forschungszentrum für Künstliche Intelligenz 1992

This work may not be copied or reproduced in whole or in part for any commercial purpose. Permission to copy in whole or in 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 Deutsches 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.

Perspektiven zur Kombination von automatischem Animationsdesign und planbasierter Hilfe*

Winfried Graf und Markus A. Thies

Deutsches Forschungszentrum für Künstliche Intelligenz (DFKI)

Stuhlsatzenhausweg 3, W-6600 Saarbrücken 11

Tel.: (+49 681) 302-5252

Fax: (+49 681) 302-5341

E-mail: {graf,thies}@dfki.uni-sb.de

Zusammenfassung

Aktuelle Themen auf dem Gebiet der intelligenten Benutzerschnittstellen behandeln derzeit die automatische Planung multimodaler Präsentationen. Hierbei stand bisher im wesentlichen die koordinierte Generierung von Text und Graphik im Vordergrund. In Zukunft wird hier aufgrund der Komplexität der zu präsentierenden Information zunehmend auch die Einbeziehung realistischer animierter 3D-Graphiken gefordert sein. Einen anderen wichtigen Forschungsschwerpunkt bildet der Einsatz graphischer Ausgabekomponenten für planbasierte Hilfesysteme. Die vorliegende Arbeit hat zum Ziel zunächst einen Überblick über den derzeitigen Stand der Forschung in diesen beiden Bereichen zu geben, als auch neue Anforderungen an die automatische Animationsgenerierung und an Systeme zur planbasierten graphischen Hilfe zu formulieren. Anschließend wollen wir, basierend auf Ergebnissen und Erfahrungen aus WIP und PLUS, Perspektiven für eine mögliche Weiterentwicklung und Integration von Techniken der Animationsplanung und graphischen Hilfe präsentieren.

*Die Autoren danken Wolfgang Wahlster und Thomas Rist für wertvolle Anregungen zu dieser Arbeit.

Inhaltsverzeichnis

| | | |
|---|--|----|
| 1 | Einleitung | 3 |
| 2 | Relevante Arbeiten | 4 |
| 3 | Anforderungen an die Generierung animierter Präsentationen | 6 |
| 4 | Planbasierte graphische Hilfe | 8 |
| 5 | Verknüpfung intelligenter Animationspräsentation und planbasierter Hilfe | 10 |
| 6 | Resümee | 11 |

1 Einleitung

Aktuelle Themen auf dem Gebiet der intelligenten Benutzerschnittstellen behandeln derzeit die automatische Planung multimodaler Präsentationen (vgl. u.a. [ST91, WAB+92, AFG+92, OSS92]). Hierbei stand bisher im wesentlichen die koordinierte Generierung von Text und Graphik im Vordergrund. Neben diesen beiden Modi stellt die Animation von realistischen 3D-Graphiken ein sehr effektives Kommunikationsmedium dar, insbesondere bei der Visualisierung komplexer räumlicher Interaktionen. Das manuelle Design animierter Darstellungen ist jedoch eine zeitintensive und komplexe Tätigkeit, die eine Menge Fachwissen bedingt. Eine Automatisierung könnte hier den Designprozeß effizienter gestalten sowie eine dynamische Animationsgenerierung 'on the fly' ermöglichen, auch wenn deren Inhalt nicht völlig antizipiert werden kann. In solchen Fällen wäre es einem menschlichen Designer nicht möglich eine adäquate Präsentation zu erstellen.

Animation zählt neben gesprochener Sprache zu den *temporalen* Medien, die sich durch ein dynamisches Verhalten auszeichnen. Eine Präsentationskomponente zum automatischen Design animierter Präsentationen hat deshalb im Gegensatz zu statischen Medien wie Text oder Graphik, die derzeit bevorzugt in intelligenten multimodalen Benutzerschnittstellen wie WIP (vgl. [WAGR91, WAB+92, AFG+92]) eingesetzt werden, zeitliche Informationen zu verarbeiten. Die wissensbasierte Erzeugung von bewegten Graphiken erfordert hier unbedingt den Einsatz von Alltagswissen, sowohl zur Festlegung der zeitlichen Reihenfolge, in der Informationen zu präsentieren sind, als auch zur Koordination der Zeitpunkte und Zeitintervalle von Aktionen. Wir interessieren uns hierbei besonders für Fragen, wie sich zeitliche Information mittels animierter Darstellungen ausdrücken läßt und wie eine über der Zeit präsentierte Information von einem Präsentationsdesigner kontrolliert werden kann.

Einen anderen wichtigen Forschungsschwerpunkt bildet die Verwendung objektorientierter graphischer Oberflächen in intelligenten Hilfesystemen (vgl. u.a. [BBD+91, TB92]), z.B. zur Softwareunterstützung von fenster-basierten Applikationen, wie CAD/CAM-Systeme, CASE-Werkzeuge etc. Es bietet sich an, in diesen Systemen wissensbasierte generierte Animationen, sowohl als Hilfestellung, z.B. bei der Menü-Selektion, Cursor-Positionierung unter Verwendung der Maus oder anderen Fenster-Manipulationsschritten, als auch zur Präsentation von visuellen Erklärungen zu verwenden.

Neue 3D-Kommunikationsmöglichkeiten wie etwa *DataGlove* und *DataSuit* (vgl. z.B. [FvDFH90]) oder auch n-dimensionale Ein-/Ausgabemedien beim Aufbau *virtueller Welten* (vgl. auch [FB90]) stellen hier immer höhere Anforderungen bzgl. Erklärungspräsentation und Hilfe bei der Benutzerführung. Dynamische Animation von zeitlicher Information kann hier die Verständlichkeit einer Präsentation erheblich erhöhen.

Wir wollen im folgenden einige grundlegende Techniken des Designs animierter Präsentationen darstellen sowie Ziele, Anforderungen, offene Probleme und Vorschläge zur Realisierung von Systemen, die diese Verfahren automatisch anwenden können, aufzeigen. Hierzu zählen insbesondere intelligente multimodale Benutzerschnittstellen, die wie im

Beispiel von WIP bisher nur die beiden Modi Graphik und Text behandeln, aber auch planbasierte graphische Hilfesysteme. Anschließend wollen wir, basierend auf Ergebnissen und Erfahrungen aus den Projekten WIP und PLUS, Perspektiven für eine mögliche Weiterentwicklung und Integration von Techniken der Animationsplanung und graphischen Hilfe präsentieren.

2 Relevante Arbeiten

Durch die breite Verfügbarkeit von Arbeitsplatzrechnern mit graphisch orientierten Benutzeroberflächen sowie der Entwicklung dedizierter Graphikhardware bzw. innovativer Programmiermethoden (z.B. objektorientierten graphischen Entwicklungsumgebungen) hat heute die wissensbasierte Erzeugung von realistischen Computergraphiken zunehmende Bedeutung für den kommerziellen Einsatz erlangt (vgl. u.a. [KW90, FS91, SF91, RA92]).

Dagegen bildet die automatische Generierung animierter Graphiken innerhalb der KI noch ein relativ junges Forschungsgebiet. Erste Ansätze zur Bewegungsplanung finden sich u.a. in [Bad87, Rey87, Wil87, TT90]. Hier wird zwar teilweise auf Spezialsoftware für Animation zurückgegriffen, wie etwa prozedurale Programmiersprachen oder auf interaktive Editoren zur Animationsgenerierung, jedoch sind nur wenige Arbeiten bekannt, die sich mit der eigentlichen Entwicklung einer Präsentationskomponente zum automatischen Animationsdesign beschäftigen. Eine solche Komponente hat die Aufgabe, die Selektion der durch Animation zu präsentierenden Information und die Ansteuerung der oben genannten Spezialsoftware wissensbasiert auszuführen.

Eine Pionierarbeit stellt gewissermaßen das 1979 am MIT AILab von Kahn entwickelte System *ANI* (vgl. [Kah79]) dar. Das System generiert animierte 2D-Darstellungen von Handlungsbeschreibungen. *ANI* bestimmt dabei die Geschwindigkeit, mit der sich die Charakter bewegen und wie diese positioniert werden sollen.

Im Gegensatz zu diesen Ansätzen ist bisher der Einsatz von planbasierten Verfahren, wie sie derzeit zur wissensbasierten Graphikgenerierung verwendet werden (vgl. [AR92]), weitgehend unerforscht. Erste rudimentäre Ansätze der automatischen Animationsplanung wurden von Feiner et al. an der Columbia University (vgl. [KF90]) entwickelt und in dem regelbasierten Expertensystem *ESPLANADE* (**E**xpert **S**ystem for **PLAN**ning **A**nimation **D**esign and **E**ding) implementiert. *ESPLANADE* erwartet als Eingabe einen detaillierten Plan von Aktionen, die geometrische Veränderungen und Bewegungen der Objekte in der modellierten Welt beschreiben, sowie mögliche Constraints für die Präsentation. Die *ESPLANADE*-Regeln erzeugen dann einen Präsentationsplan für eine vollständige Animation, d.h. Spezifikation der Einstellparameter für die Kamera etc.

Einen ersten Versuch zur Repräsentation komplexer zeitlicher Pläne wurde von [FLMP91] durchgeführt. Zur Beschreibung zeitlicher Aktionen wurde Allens intervall-basierte Zeitlogik verwandt. Das Ziel der Arbeit bestand in einer Repräsentation temporaler Pläne in Form einer terminologischen Logik.

Verstärkte Bemühungen, die den Präsentationsaspekt in den Vordergrund stellen, wurden in den letzten Jahren auch von der Gruppe um Badler an der UPenn (siehe insb. [BBZ91, BWKE91, Zel91]) durchgeführt. Hier wird Animation aus natürlichsprachlichen Anweisungen generiert, mit dem Ziel sog. *narrated animations*. Der Schwerpunkt dieser Arbeit liegt in der Bewegungskontrolle.

Im folgenden möchten wir zwei Systeme vorstellen, die 2D-Animation in Hilfesystemen verwenden. Das System **GAK** (Graphical Animation from Knowledge, vgl. [Nei82]) erweitert das bestehende wissensbasierte Hilfesystem *CADHELP*, das dem Benutzer natürlichsprachliche Unterstützung im Umgang mit einer kleinen CAD Anwendung bietet, um eine Animationskomponente zur Visualisierung der für bestimmte Applikationsaktionen notwendigen Interaktionsschritte.

Im Gegensatz zum System GAK verzichtet der Ansatz in *Cartoonist* (vgl. [SF90]) auf eine animierte Darstellung der Kausalität der Cursor-Bewegung auf dem Bildschirm. Sind allerdings Mausaktionen wie *linke-Maustaste-drücken* oder *bewegen-der-Maus-mit-niedergedrückter-Taste* von Bedeutung für die Durchführung einer Interaktion, wird die Darstellung des Mausursors entsprechend verändert. Diese Technik der animierten 2D-Darstellung von Mausaktionen wird mit Hilfe von vordefinierten *Characters* erreicht. Sie wird auch zur Animation von Tastatureingaben verwendet. *Cartoonist* erzeugt eine so animierte Sequenz unter Berücksichtigung des aktuellen Bildschirmkontextes.

Die Erzeugung von animierten Darstellungen setzt Informationen über die Benutzerschnittstelle, die Anwendung und die Interaktionsstile voraus. *Cartoonist* greift dabei auf die Wissensrepräsentation des *User Interface Design Environment* (vgl. [FGKK88]) zurück. Das Wissen ist in applikationsspezifisches, schnittstellenspezifisches und interaktionstechnisches Wissen und Relationen zwischen Applikationsaktionen, Schnittstellenaktionen und den Interaktionstechniken unterteilt. Mit Hilfe von Vor- und Nachbedingungen wird deduktiv eine Sequenz von Animationsschritten für eine bestimmte Applikationsaktion generiert. Enthalten die Animationsschritte nicht-instantiierte Parameter, so werden diese, unter Berücksichtigung des aktuellen Bildschirminhalts, bestimmt. Dabei wird die Menge der in Frage kommenden Objekte mit Hilfe von in der Wissensbasis definierten Parameter-Constraints eingeschränkt. Aus der eingeschränkten Menge wird nicht-deterministisch ein Objekt selektiert. Analog wird mit Parametern aus dem numerischen Bereich verfahren. *Cartoonist* eignet sich zur Unterstützung des Benutzers bei Hilfeanforderungen wie "*Zeige mir (exemplarisch) wie ich ...*". Es wird dem Benutzer exemplarisch eine animierte Sequenz von Interaktionsschritten präsentiert.

Die im folgenden beschriebenen Ansätze der automatischen Animationsplanung basieren im wesentlichen auf Verfahren zur wissensbasierten Erzeugung realistischer statischer Graphiken oder 3D-Bildsequenzen zur Beschreibung von Aktionsfolgen wie sie derzeit für die Graphikgeneratoren der multimodalen Präsentationssysteme *WIP* (s. [WAB⁺92, AFG⁺92]) und *COMET* (s. [MF90]).

3 Anforderungen an die Generierung animierter Präsentationen

Computer-Animation behandelt sämtliche Bewegungsprozesse, die visuelle Effekte haben. Dazu zählen sowohl die zeitliche Variation der Position (sog. *motion dynamics*) als auch der Merkmale (z.B. Form, Farbe, Transparenz, Struktur und Annotation, sog. *update dynamics*) von graphischen Objekten sowie Entscheidungen der Kameraplanung. Techniken der Animation fanden bisher bevorzugt Einsatz in Simulationsprogrammen für industrielle Anwendungen, Tutorssystemen und medizinischen Applikationen. Die Forschung in diesem Bereich hat sich bisher im wesentlichen auf Techniken zur Planung von Bewegungen (*motion planning*), wie sie etwa bei der Filmproduktion eingesetzt werden, konzentriert. Die Aufgaben reichen dabei von high-level Verfahren zum Drehbuch-Design (*scripting*) bis zur low-level Bewegungsplanung einzelner Charakter. Für eine ausführlichere Beschreibung elementarer Techniken der Filmerstellung (wie *shooting* und *editing*) sei an dieser Stelle auf die Arbeit von [KF90] bzw. entsprechende Spezialliteratur verwiesen.

Bedingt durch die Entwicklung von dedizierter Graphikhardware, wird die Computer-Animation in naher Zukunft auch im Bereich der Künstlichen Intelligenz einen hohen Stellenwert erlangen, z.B. als Erweiterung von intelligenten multimodalen Schnittstellen, die wie in WIP bisher auf die statischen Medi Text und Graphik beschränkt waren. Die automatische Erzeugung von realistischen 3D-Animationen erfordert jedoch neben der Verarbeitung von Wissen über Filmerstellungstechniken auch den Einsatz von *Alltagswissen* über Inhaltsauswahl, Inhaltsdarbietung, Erhaltung der visuellen Kontinuität und Kohärenz etc.

Der Prozeß der Animationsgenerierung läßt sich damit in die folgenden beiden wesentlichen Teilbereiche untergliedern (vgl. auch [KF90]):

1. Aufbau einer animierten virtuellen Welt
 - (i) Spezifikation der Objekte der zu animierenden Welt sowie der Aktionen an denen diese beteiligt sind,
 - (ii) High-level Scripting,
 - (iii) Low-level Objektmodellierung,
 - (iv) Bestimmung der *frame-to-frame* Bewegungen interagierender Objekte in einer *Newton-Welt*.
2. Planung wie diese Objekte und Aktionen präsentiert werden sollen
 - (i) Selektion der zu präsentierenden Inhalte,
 - (ii) Bestimmung der Zeitpunkte zu denen die Information präsentiert werden soll,
 - (iii) Bestimmung der Reihenfolge in der die Information präsentiert werden soll,

- (iv) Definition der virtuellen Kamera, durch die die Welt betrachtet wird (d.h. u.a. Wahl der Betrachtungsspezifikationen, der Viewports und der Transitionen zwischen Kameras innerhalb von Viewports).

Hieraus ergibt sich schließlich der Unterschied zwischen einer animierten Präsentation und einer animierten virtuellen Welt.

In bestimmten Situationen kann es auch intendiert sein, dem Benutzer Präsentationsentscheidungen über den Aufbau einer virtuellen Welt zu überlassen oder Entscheidungen unter Berücksichtigung von vorher spezifizierten Kommunikationszielen zu treffen.

Bei der wissensbasierten Erzeugung animierter Darstellungen im Rahmen eines multimodalen Präsentationssystems ergeben sich im Hinblick auf eine Koordinierung der verschiedenen Modalitäten u.a. die folgenden im Moment **offenen Probleme**:

- *Synchronisation statischer und dynamischer Modi*

Die Kombination und Koordinierung verschiedener Modalitäten, insbesondere zwischen statischen und bewegten Bildern, aber auch zwischen Text (z.B. Annotationen) und bewegten Graphiken, oder zwischen einem fixem Hintergrund und im Vordergrund bewegten Bildern (sog. $2\frac{1}{2}$ -dim. Darstellungsweise) läßt sich mit den bisherigen Methoden nicht zufriedenstellend behandeln.

- *Repräsentation temporaler räumlicher Information*

Einen zentralen Punkt bei der Behandlung eines *temporalen* Mediums wie der Animation nimmt die Verarbeitung *temporaler Information* ein. Hierunter wollen wir Entitäten und Relationen verstehen, die über der Zeit definiert sind, wie z.B. Zeitpunkte, Zeitintervalle, Aktionen, Ordnungsrelationen und Überlappungsintervalle. Die für die Beschreibung statischer Graphiken verwendete propositionale Wissensbasis reicht zur Repräsentation von Bewegungen i.a. nicht aus. Hier sind u.a. Informationen über die Geschwindigkeit, die Trajektorien, den kürzesten Weg zwischen zwei Objekten in animierten Aktionen etc. zu modellieren. Um ein einigermaßen effizientes Verhalten zu garantieren, scheint eine Verwendung von Heuristiken, die durch Regeln oder Defaults repräsentiert sind, sinnvoll zu sein. Darüberhinaus sind Fragen der Sichtbarkeit von Objekten und der Perspektivenwahl zu klären.

Objekte bewegen sich in einer physikalischen Welt nicht unbedingt auf geraden Linien, sondern üben relativ zu anderen Objekten nichtlineare zusammengesetzte Bewegungen aus. Zur Spezifikation animierter Sequenzen und deklarativen Repräsentation struktureller (z.B. semantisch-pragmatischer) dynamischer Zusammenhänge auf einem hohen Abstraktionsniveau werden in letzter Zeit häufig innovative Programmiermethoden wie logische und constraint-basierte Formalismen verwendet.

- *Layout dynamischer Präsentationen*

Eine wichtige Fragestellung wirft auch das Design des Layouts von zeitlich variierenden Präsentationen auf. Die für statische Dokumente verwendeten Gridansätze sind für animierte Präsentationen zu unflexibel. Ähnlich dem graphischen Layout von

Text-Bildkombinationen (vgl. [GM91]) können Überlappungsfreiheit und Ordnungsrelationen zwischen animierten Objekten durch Constraints garantiert werden. Erste Ansätze constraint-basierter Animation finden sich in den Systemen *Sketchpad* (vgl. [Sut63]) und *ThingLab* (vgl. [Bor81]). Erweiterung von Constraint-Systemen kann man sich hier bzgl. der Verarbeitung von Constraint-Hierarchien, sowie die Modellierung von potentiellen Bewegungen physikalischer Körper und deren strukturellen Materialeigenschaften mittels Constraints, vorstellen

- *Benutzermodellierung*

Im Gegensatz zum statischen Fall, bei dem der Benutzer sehr leicht einen Bezug zwischen den dargestellten Objekten und entsprechenden Weltobjekten herstellen kann, ist bei animierten Aktionen (z.B. Kamerabewegungen oder speziellen Filmtechniken wie *Zooming*) schwer zu entscheiden, was der Betrachter zu welchem Zeitpunkt der zeitlichen Präsentation weiß und damit, welche Information repräsentiert werden muß. Hierzu zählt u.a. auch die Frage, auf welche der dargestellten Objekte z.B. mittels anderer Modalitäten referenziert wird und welche Objekte den Kontext bilden (sog. *cross references*, vgl. [WAGR91]). Da sich insbesondere intelligente Präsentationsysteme wie WIP zum Ziel gesetzt haben, alle Information, die präsentiert wird, auch zu repräsentieren, ist die Frage *Was ist zu repräsentieren?* offen.

4 Planbasierte graphische Hilfe

Neben der Generierung realistischer animierter Darstellungen, ist die Entwicklung planbasierter graphischer Hilfesysteme zur Unterstützung des Benutzers bei der Interaktion mit Softwareapplikationen ein aktuelles Forschungsthema. Konventionelle und wissensbasierte Hilfesysteme, wie etwa die Systeme *WIZARD* (s. [Fin83]), *AKTIVIST* (s. [FLS85]), *UC* (s. [WCL⁺88]), *SC* (s. [WDHK90]) und *PHI* (s. [BBD⁺91]) leisten dem Benutzer hinsichtlich der Terminologie, des Sachproblems und des konzeptuellen Vorgehens im Zusammenhang mit einer Applikation Unterstützung. Die meisten intelligenten Hilfesysteme verwenden Pläne als zentrale Hilfeinformation.

Allerdings stoßen diese Systeme mit einer rein textuellen Hilfe an ihre Grenzen, sobald der Benutzer Hilfe zur Durchführung einzelner Interaktionsschritte einer Applikation benötigt, wenn es also um die Beantwortung von Fragen oder Aufforderungen folgender Form geht: *“Wie verbinde ich zwei Objekte?”* oder *“Zeige mir bitte wie ich nur die Objekte A, B und C angezeigt bekomme.”*. Ein Hilfetext, der möglicherweise von obigen Hilfesystemen generiert wird, könnte lauten: *“Bewege die Maus zu der Position des ersten Objektes und drücke die linke Maustaste nieder. Lasse die Maustaste wieder los und bewege die Maus an die Position des zweiten Objektes...”*. Es wird deutlich, daß eine graphische Visualisierung der Interaktionsschritte dem Benutzer eine adäquatere Unterstützung bietet.

Eine solche aufgabenbezogene graphische Hilfe bietet das *PLUS* System (vgl. [TB92]). Im Gegensatz zu den bisherigen Hilfesystemen, die meist für kommando-orientierte Schnitt-

stellen entwickelt wurden, arbeitet PLUS mit Applikationen, die dem Benutzer graphische Bedienoberflächen zur Verfügung stellen, und deren Interaktion auf dem Prinzip eines benutzergeführten Dialoges mittels direkter Manipulation (vgl. [Shn83, Shn87]) basiert, sog. *Direkt-Manipulative Benutzerschnittstellen* (DMI).

Den Kern des Projektes bildet die Beschreibung von Benutzeraktionen mittels planbasierter Verfahren. Die Beschreibung ist anwendungsunabhängig und somit universell einsetzbar. Die Pläne sind um Wissen über generische Schnittstellenkonzepte (nach SAA/Common User Access¹) in Form von Fakten und Regeln ergänzt. Der Beschreibungsrahmen einer Benutzerschnittstelle mittels Plänen und Wissen wird im Rahmen dieses Projektes als Basis für verschiedene Hilfearten eingesetzt. Die zentrale Komponente des Systems ist der Planerkenner. Er versucht vom Benutzer ausgeführte Aktionen auf die in der statischen hierarchischen Planbasis gespeicherten Pläne abzubilden. Als statische Planbasis wollen wir hier die Menge der von einem Plandesigner vordefinierten Pläne und deren Ziele bezeichnen.

Die Planerkennung wird mittels eines *Spreading Activation Algorithmus* realisiert, der zur Laufzeit eine dynamische hierarchische Planbasis erzeugt. Das in der dynamischen Planbasis enthaltene hypothetische Wissen über die aktuell vom Benutzer verfolgten Pläne und Ziele dient zusammen mit einer Wissensbasis über allgemeine Hilfekonzepte als Grundlage für die graphische Komponente *InCome+* (vgl. [Thi90, FT91]) und weiteren Hilfe Komponenten wie animierte Hilfe. *InCome+* visualisiert den Interaktionskontext des Benutzers und stellt darüberhinaus weitere Merkmale wie semantische Undo- und Redo-Möglichkeiten und einen kontext-sensitiven Tutor zur Verfügung.

Die animierte Hilfe arbeitet analog zu *Cartoonist*, indem sie animierte 2D-Darstellungen von Mausaktionen generiert. Gleichzeitig werden diese Mausaktionen imitiert und der Applikation so zugeführt, daß diese auf die simulierten Eingabedaten reagiert, als ob sie von einem Benutzer stammen würden.

Im Gegensatz zur animierten Hilfe in *Cartoonist* erreicht diese Komponente einen stärkeren Bezug zur momentan vom Benutzer verfolgten Aufgabe (Planhypothese). Ein an den Planerkenner und die Planvervollständigungskomponente angebundener Animationsdesigner kann auf Anfrage gezielt für diese Planhypothese eine Sequenz von Animations schritten generieren. Die Generierung erfolgt deduktiv, indem auf die in der Wissensbasis für jede Aktion definierten Vor- und Nachbedingungen zugegriffen wird. Durch die in der statischen Planbasis definierten Parameter-Constraints können durch den Planerkennungsprozeß bekannte Parameter von der Planvervollständigungskomponente propagiert werden. Bleiben auch nach der Propagierung Parameter unspezifiziert, müssen diese vom Benutzer spezifiziert werden.

Durch die Repräsentation von generischen Schnittstellenkonzepten in der Wissensbasis, können auch navigatorische Animationsschritte generiert werden. Wurde als Ziel der

¹SAA/Common User Access beschreibt eine einheitliche Architektur für Benutzerschnittstellen auf IBM Systemen. Dabei wird ein benutzergeführter Dialog mittels direkter Manipulation zugrunde gelegt. Als wesentliches Prinzip gilt: "Wähle zuerst ein Objekt aus und wende darauf eine Aktion an."

Animation 'Verbinden der Objekte A und B' bestimmt und Objekt A ist aus dem sichtbaren Bereich des Applikationsfensters hinausgeschoben worden, generiert die Animationskomponente die entsprechenden navigatorischen Animationsschritte zum Verschieben des sichtbaren Bereiches des Applikationsfensters bis Objekt A sichtbar wird. Nachdem die entsprechenden Animationsschritte zur Selektion der beiden Objekte generiert wurden, wird durch das Aktivieren der Menüfunktion *Connect* das gewünschte Ziel erreicht.

Der an jedes Hilfesystem gestellte Anspruch der Adäquatheit der Unterstützung bedingt auch die Berücksichtigung von Benutzertypen (z.B. unerfahrener Benutzer, Gelegenheitsbenutzer oder Experte) in einer graphischen Hilfe. Einem ungeübten Benutzer muß die Bedienung des Eingabemediums durch eine animierte Darstellung erklärt werden, dagegen ist dies bei einem Experten meist nicht erforderlich. Die Unterstützung zur Eingabe von Text mit Hilfe des Eingabemediums Tastatur (siehe *Cartoonist*) durch eine animierte Darstellung ist selbst für einen ungeübten Benutzer zu detailliert. Dagegen erscheint die graphische Präsentation der Kombination von mehreren Tasten zur Aktivierung von Menüfunktionen sinnvoll, um die Lage der Tasten und die temporale Abhängigkeit der Aktivierung der Tasten auszudrücken.

Im folgenden Kapitel werden wir über die bisher verwendete 2D-Animation hinausgehen und die wissensbasierte 3D-Animationspräsentation mit der planbasierten Hilfe verknüpfen.

5 Verknüpfung intelligenter Animationspräsentation und planbasierter Hilfe

Der Schwerpunkt der Verwendung des Mediums 3D-Animation liegt bei der Verdeutlichung der Zusammenhänge zwischen Bewegung und Position des Eingabemediums und dem *Cursor* auf dem Bildschirm. Die Animationssequenz könnte z.B. exemplarisch visualisieren, wie eine Hand den Stylus aufgreift und auf dem Digitizer-Tableau bewegt. Synchron zu den generierten Bewegungen der Hand samt Stylus wird der *Cursor* auf dem Bildschirm mitbewegt.

Bisherige Hilfesysteme mit 2D-animierter Hilfe (vgl. Kap. 4) zeigen einen guten Ansatz auf. Allerdings beschränkt sich die Verknüpfung von Animation und Hilfekomponente auf die Ebene der Simulation von Mausinteraktionen mit einer mehr oder weniger animierten Darstellung ihrer Kausalität. Wir verwenden bewußt den neutralen Begriff des *Eingabemediums*, um nicht eine Fixierung auf die üblichen Medien wie Maus oder Tastatur zu erhalten, sondern um auch der Entwicklung neuerer Kommunikationsmittel, wie *3D-Mäuse*, *Data-Gloves* oder *Data-Suits* (siehe u.a. [FvDFH90]), Rechnung zu tragen.

Animation stellt eine Visualisierung von Handlungssequenzen dar. Im Gegensatz zu den in *Cartoonist* und *PLUS* verwendeten animierten 2D-Graphiken, ist für die Visualisierung von Handlungssequenzen, die eine Manipulation des Eingabemediums zeigen, eine Generierung von realistischer 3D-Animation erforderlich, da nun 3D-Objekte aus der 'realen' Welt (wie Eingabemedium und Hand) umfaßt. Dabei ist es essentiell, realistische 3D-

Animation zu erzeugen, da gerade neue Eingabemedien die dritte Dimension als weiteres Eingabedatum einführen. Interessant ist die Integration dieser generierten Präsentation in Systeme wie PLUS oder Cartoonist, um zusätzlich die Handhabung des Eingabemediums, im Kontext der aktuellen Aufgabe des Benutzers, visualisieren zu können.

Hierzu ein Beispiel: Der Benutzer benötigt Hilfe für die Umsetzung der zum Erreichen seines Aufgabenziels nötigen Aktionen, die er von einer Komponente zur Visualisierung des Interaktionskontextes (wie InCome⁺) in statischer Form präsentiert bekommt, in die korrekte Sequenz von Interaktionsschritten und Schritten zur Manipulation des Eingabemediums. Hier setzt nun die Simulations- und Präsentationskomponente ein. Die Simulationskomponente führt die Interaktionsschritte durch. Gleichzeitig wird die Bedienung des Eingabemediums durch die Präsentationskomponente in einer 3D-Animation erklärt. So kann der Benutzer verfolgen, welche Interaktionsschritte nötig sind und wie diese Interaktionsschritte mit Hilfe des Eingabemediums erreicht werden. Verwenden wir hierzu eine 3D-Maus, ist die Perspektivenwahl der zu generierenden Animation entscheidend für das Verständnis des Benutzers. Eine Animation zur Darstellung des Handlungsablaufs: *'Bewegen der Maus von vorne links unten nach hinten rechts oben mit niedergedrückter Maustaste'*, entspricht dem Verschieben eines Objektes durch den Raum, bedingt sogar eine Perspektivenverschiebung während der Animation (s. Kap. 3).

Arbeiten über planbasierte Hilfesysteme haben gezeigt, daß es schwer ist, aufgrund der beobachteten Aktionen eindeutig auf das Ziel des Benutzers zu schließen. Hieraus wird ersichtlich, daß eine ausschließliche Verwendung von Animation oftmals nicht ausreichen kann, um dem Benutzer eindeutig ein Ziel zu vermitteln. Eine Erweiterung durch Text, der die Animation erklärend ergänzt, erscheint notwendig (siehe [BWKE91], vgl. hierzu auch Erklärungskomponenten [Wah81] und Klärungsdialoge [Feh89]).

6 Resümee

Mit der vorliegenden Arbeit wollten wir einige Zukunftsperspektiven für den Bereich der intelligenten Benutzerschnittstellen aufzeigen, die sich insbesondere aus dem Einsatz dynamischer Medien wie Animation ergeben. Dazu haben wir Anforderungen an eine Präsentationskomponente zur automatischen Planung von Animationen spezifiziert. Anschließend haben wir versucht zu illustrieren, wie sich Ansätze der Generierung realistischer 3D-Animationen in graphischen Schnittstellen planbasierter Hilfesysteme integrieren lassen. Eine solche Kombination führt zu einer Erweiterung der bisherigen Hilfesysteme und zu neuen Kommunikationstechniken bei der Unterstützung des Benutzers. Der Aspekt der animierten Hilfe erreicht durch den Einsatz evtl. n-dimensionaler Ein- und Ausgabemedien in neuen Anwendungsgebieten wie virtuellen Realitäten einen erhöhten Stellenwert. Daraus ergeben sich mögliche Trends für zukünftige Erweiterungen von multimodalen Benutzerschnittstellen.

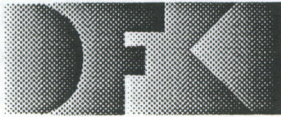
Literatur

- [AFG⁺92] E. André, W. Finkler, W. Graf, T. Rist, A. Schauder, und W. Wahlster. *WIP: The Automatic Synthesis of Multimodal Presentations*. In: M. Maybury (Hrsg.), *Intelligent Multimedia Interfaces*. Menlo Park, CA: AAAI Press, 1992. forthcoming.
- [AR92] E. André und T. Rist. *The Design of Illustrated Documents as a Planning Task*. German Research Center for Artificial Intelligence, DFKI Research Report, January 1992.
- [Bad87] N. Badler. *Computer Animation Techniques*. In: W. Brauer und W. Wahlster (Hrsg.), *Proceedings 2. Internationaler GI-Kongreß Wissensbasierte Systeme*, S. 22–34. Berlin, Germany: Springer-Verlag, October 1987.
- [BBD⁺91] M. Bauer, S. Biundo, D. Dengler, M. Hecking, J. Koehler, und G. Merzinger. *Integrated Plan Generation and Recognition - A Logic-Based Approach*. In: W. Brauer und D. Hernández (Hrsg.), *Proceedings 4. Internationaler GI-Kongreß Wissensbasierte Systeme - Verteilte KI und kooperatives Arbeiten*, S. 266–277. Berlin, Germany: Springer-Verlag, October 1991.
- [BBZ91] N. Badler, B. Barsky, und D. Zeltzer (Hrsg.). *Making Them Move: Mechanics, Control, and Animation of Articulated Figures*. Los Altos, CA: Morgan Kaufmann, 1991.
- [Bor81] A. Borning. *The Programming Language Aspects of ThingLab, a Constraint-oriented Simulation Laboratory*. *ACM Transactions on Programming Languages and Systems*, 3(4):353–387, October 1981.
- [BWKE91] N. Badler, B. Webber, J. Kalita, und J. Esakov. *Animation from Instructions*. In: Badler et al. [BBZ91], S. 51–93.
- [FB90] S. Feiner und C. Beshers. *Worlds within Worlds: Metaphors for Exploring n-Dimensional Virtual Worlds*. In: *Proceedings of the UIST'90 (ACM Symp. on User Interface Software and Technology)*, S. 76–83, Snowbird, UT, October 1990.
- [Feh89] Th. Fehrle. *Menüorientierte, wissensbasierte Klärungsdialoge für ein natürlichsprachliches Auskunftssystem*. Dissertation, Institut für Informatik der Universität Stuttgart, 1989.
- [FGKK88] J. D. Foley, C. Gibbs, W. C. Kim, und S. Kovacevic. *A Knowledge-based User Interface Management System*. In: *CHI'88 Human Factors in Computer Systems, Conference Proceedings*, Washington, D.C., 1988.
- [Fin83] T. W. Finin. *Providing Help and Advice in Task Oriented Systems*. In: *Proc. IJCAI-83*, S. 176–178, Karlsruhe, Deutschland, 1983.

- [FLMP91] S. Feiner, D. Litman, K. McKeown, und R. Passonneau. *Towards Coordinated Temporal Multimedia Presentations*. In: Proceedings of the AAAI-91 Workshop on Intelligent Multimedia Interfaces, Anaheim, CA, July 1991.
- [FLS85] G. Fischer, A. Lemke, und T. Schwab. *Knowledge-based Help Systems*. In: Proceedings CHI-85, San Francisco, CA, 1985.
- [FS91] S. Feiner und D. Seligmann. *Dynamic 3D Illustrations with Visibility Constraints*. In: Computer Graphics International '91, Cambridge, MA, June 1991.
- [FT91] Th. Fehrle und M.A. Thies. *InCome: A System to Navigate through Interactions and Plans*. In: H.-J. Bullinger (Hrsg.), *Human Aspects in Computing: Design and Use of Interactive Systems and Information Management*, Amsterdam, London, New York, Tokyo, 1991. Elsevier Science Publishers B.V.
- [FvDFH90] J. Foley, A. van Dam, S. Feiner, und J. Hughes (Hrsg.). *Computer Graphics: Principles and Practice, 2nd Edition*. The System Programming Series. Reading, MA: Addison-Wesley, 1990.
- [GM91] W. Graf und W. Maaß. *Constraint-basierte Verarbeitung graphischen Wissens*. In: W. Brauer und D. Hernández (Hrsg.), *Proceedings 4. Internationaler GI-Kongreß Wissensbasierte Systeme - Verteilte KI und kooperatives Arbeiten*, S. 243–253. Berlin, Germany: Springer-Verlag, October 1991. Also DFKI Research Report RR-91-35.
- [Kah79] K. Kahn. *Creation of Computer Animation from Story Descriptions*. Dissertation, AI Lab, Massachusetts Institute of Technology, Cambridge, MA, January 1979.
- [KF90] P. Karp und S. Feiner. *Issues in the Automated Generation of Animated Presentations*. In: *Graphics Interface '90*, S. 39–48, Los Altos, CA, May 1990. Morgan Kaufmann.
- [KW90] K. Kansy und P. Wißkirchen (Hrsg.). *Proceedings GI-Fachgespräch 'Graphik und KI', IFB 239*. Berlin, Germany: Springer-Verlag, April 1990.
- [MF90] K. McKeown und S. Feiner. *Interactive Multimedia Explanation for Equipment Maintenance and Repair*. In: DARPA Speech and Language Workshop, S. 42–47, 1990.
- [Nei82] D. Neiman. *Graphical Animation from Knowledge*. In: Proceedings of the 2nd National Conference of the American Association for Artificial Intelligence, S. 373–376, Pittsburgh, PA, July 1982.
- [OSS92] A. Ortony, J. Slack, und O. Stock (Hrsg.). *Computational Theories of Communication and their Application*. Berlin, Germany: Springer-Verlag, 1992.

- [RA92] T. Rist und E. André. *From Presentation Tasks to Pictures Including Depictions of 3D Objects: Towards an Approach to Automatic Graphics Design*. German Research Center for Artificial Intelligence, DFKI Research Report, January 1992.
- [Rey87] C. Reynolds. *Flocks, Herds and Schools: A Distributed Behavioral Model*. In: ACM SIGGRAPH '87, S. 25–34, July 1987.
- [SF90] P. Sukaviriya und J. D. Foley. *Coupling a UI Framework with Automatic Generation of Context-Sensitive Animated Help*. In: Proceedings of the ACM Symp. on User Interfaces Software and Technology, S. 152–166, Snowbird, UT, October 1990.
- [SF91] D. Seligmann und S. Feiner. *Automated Generation of Intent-Based 3D Illustrations*. Computer Graphics, 25(3), July 1991.
- [Shn83] B. Shneiderman. *Direct Manipulation: A step beyond programming Languages*. IEEE Computer, 16, 1983.
- [Shn87] B. Shneiderman. *Designing the User Interfaces: Strategies for effective Human-Computer Interaction*. Massachusetts: Addison Wesley, 1987.
- [ST91] J. Sullivan und S. Tyler (Hrsg.). *Intelligent User Interfaces*. Frontier Series. New York, NY: ACM Press, 1991.
- [Sut63] I. Sutherland. *Sktechpad: A Man-Machine Graphical Communication System*. In: IFIPS Proceedings of the Spring Joint Computer Conference, S. 329–345, 1963.
- [TB92] M. A. Thies und F. Berger. *Planbasierte graphische Hilfe in objektorientierten Benutzeroberflächen*. Submitted to GI-Fachgespräch *Innovative Programmiermethoden für Graphische Systeme*, Bonn, January 1992.
- [Thi90] M. A. Thies. *Interaction Control Manager: Ein System zum Navigieren durch Interaktionen und Pläne*. Diplomarbeit, Fakultät Informatik, Universität Stuttgart, Deutschland, 1990.
- [TT90] N. Magnenat Thalmann und D. Thalmann (Hrsg.). *Computer Animation. Theory and Practice. 2nd edition*. Computer Science Workbench. Berlin, Germany: Springer-Verlag, 1990.
- [WAB⁺92] W. Wahlster, E. André, S. Bandyopadhyay, W. Graf, und T. Rist. *WIP: The Coordinated Generation of Multimodal Presentations from a Common Representation*. In: Ortony et al. [OSS92]. Also DFKI Research Report RR-91-08.

- [WAGR91] **W. Wahlster, E. André, W. Graf, und T. Rist.** *Designing Illustrated Texts: How Language Production Is Influenced by Graphics Generation.* In: Proceedings of the 5th Conference of the European Chapter of the Association for Computational Linguistics, S. 8–14. Berlin, Germany: Springer-Verlag, April 1991. Also DFKI Research Report RR-91-05.
- [Wah81] **W. Wahlster.** *Natürlichsprachliche Argumentation in Dialogschnittstellen.* Informatik Fachberichte 48. Springer-Verlag, 1981.
- [WCL+88] **R. Wilensky, D. N. Chin, M. Luria, J. Martin, J. Mayfield, und D. Wu.** *The Berkeley UNIX Consultant Project.* Computational Linguistics, 14:35–84, 1988.
- [WDHK90] **W. Wahlster, D. Dengler, M. Hecking, und C. Kemke.** *SC: The SINIX Consultant.* In: P. Norvig, W. Wahlster, und R. Wilensky (Hrsg.), *Intelligent Help Systems for Unix - Case Studies in Artificial Intelligence.* Heidelberg: Springer, 1990.
- [Wil87] **J. Wilhelms.** *Towards Automatic Motion Control.* IEEE Computer Graphics and Animation, 7(4):11–22, April 1987.
- [Zel91] **D. Zeltzer.** *Task Level Graphical Simulation: Abstraction, Representation, and Control.* In: Badler et al. [BBZ91], S. 3–33.



Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

DFKI
-Bibliothek-
PF 2080
D-6750 Kaiserslautern
FRG

DFKI Publikationen

Die folgenden DFKI Veröffentlichungen sowie die aktuelle Liste von allen bisher erschienenen Publikationen können von der oben angegebenen Adresse bezogen werden.

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

DFKI Publications

The following DFKI publications or the list of all published papers so far can be ordered from the above address.

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

DFKI Research Reports

RR-91-08

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

RR-91-09

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

RR-91-10

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

RR-91-11

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

RR-91-12

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

RR-91-13

Gert Smolka: Residuation and Guarded Rules for
Constraint Logic Programming
17 pages

RR-91-14

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

RR-91-15

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

RR-91-16

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

RR-91-17

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

RR-91-18

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

RR-91-19

Munindar P. Singh: On the Commitments and
Precommitments of Limited Agents
15 pages

RR-91-20

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

RR-91-21

Klaus Netter: Clause Union and Verb Raising
Phenomena in German
38 pages

RR-91-22

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

RR-91-23

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

RR-91-24

Jochen Heinsohn: A Hybrid Approach for Modeling Uncertainty in Terminological Logics
22 pages

RR-91-25

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

RR-91-26

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

RR-91-27

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

RR-91-28

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

RR-91-29

Hans Uszkoreit: Strategies for Adding Control Information to Declarative Grammars
17 pages

RR-91-30

Dan Flickinger, John Nerbonne: Inheritance and Complementation: A Case Study of Easy Adjectives and Related Nouns
39 pages

RR-91-31

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

RR-91-32

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

RR-91-33

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

RR-91-34

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

RR-91-35

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

RR-92-01

Werner Nutt: Unification in Monoidal Theories is Solving Linear Equations over Semirings
57 pages

RR-92-02

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

RR-92-03

Harold Boley: Extended Logic-plus-Functional Programming
28 pages

RR-92-04

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

RR-92-05

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

RR-92-07

Michael Beetz: Decision-theoretic Transformational Planning
22 pages

RR-92-08

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

RR-92-09

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

RR-92-11

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

RR-92-13

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

RR-92-14

Intelligent User Support in Graphical User
Interfaces:

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

22 pages

RR-92-15

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

RR-92-17

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

RR-92-18

John Nerbonne: Constraint-Based Semantics
21 pages

DFKI Technical Memos**TM-91-01**

Jana Köhler: Approaches to the Reuse of Plan
Schemata in Planning Formalisms
52 pages

TM-91-02

Knut Hinkelmann: Bidirectional Reasoning of Horn
Clause Programs: Transformation and Compilation
20 pages

TM-91-03

Otto Kühn, Marc Linster, Gabriele Schmidt:
Clamping, COKAM, KADS, and OMOS:
The Construction and Operationalization
of a KADS Conceptual Model
20 pages

TM-91-04

Harold Boley (Ed.):
A sampler of Relational/Functional Definitions
12 pages

TM-91-05

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

TM-91-06

Johannes Stein: Aspects of Cooperating Agents
22 pages

TM-91-08

Munindar P. Singh: Social and Psychological
Commitments in Multiagent Systems
11 pages

TM-91-09

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

TM-91-10

*Béla Buschauer, Peter Poller, Anne Schauder, Karin
Harbusch:* Tree Adjoining Grammars mit
Unifikation
149 pages

TM-91-11

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

TM-91-12

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

TM-91-13

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

TM-91-14

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

TM-91-15

Stefan Bussmann: Prototypical Concept Formation
An Alternative Approach to Knowledge
Representation
28 pages

TM-92-01

Lijuan Zhang:
Entwurf und Implementierung eines Compilers zur
Transformation von Werkstückrepräsentationen
34 Seiten

DFKI Documents**D-91-01**

Werner Stein, Michael Sintek: Relfun/X - An Experimental Prolog Implementation of Relfun
48 pages

D-91-02

Jörg P. Müller: Design and Implementation of a Finite Domain Constraint Logic Programming System based on PROLOG with Corouting
127 pages

D-91-03

Harold Boley, Klaus Elsbernd, Hans-Günther Hein, Thomas Krause: RFM Manual: Compiling RELFUN into the Relational/Functional Machine
43 pages

D-91-04

DFKI Wissenschaftlich-Technischer Jahresbericht 1990
93 Seiten

D-91-06

Gerd Kamp: Entwurf, vergleichende Beschreibung und Integration eines Arbeitsplanerstellungssystems für Drehteile
130 Seiten

D-91-07

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

D-91-08

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

D-91-09

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

D-91-10

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

D-91-11

Thilo C. Horstmann: Distributed Truth Maintenance
61 pages

D-91-12

Bernd Bachmann: HieraCon - a Knowledge Representation System with Typed Hierarchies and Constraints
75 pages

D-91-13

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

D-91-14

Erich Achilles, Bernhard Hollunder, Armin Laux, Jörg-Peter Mohren: X_RIS: X_Rnowledge Representation and Inference System - Benutzerhandbuch -
28 Seiten

D-91-15

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

D-91-16

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

D-91-17

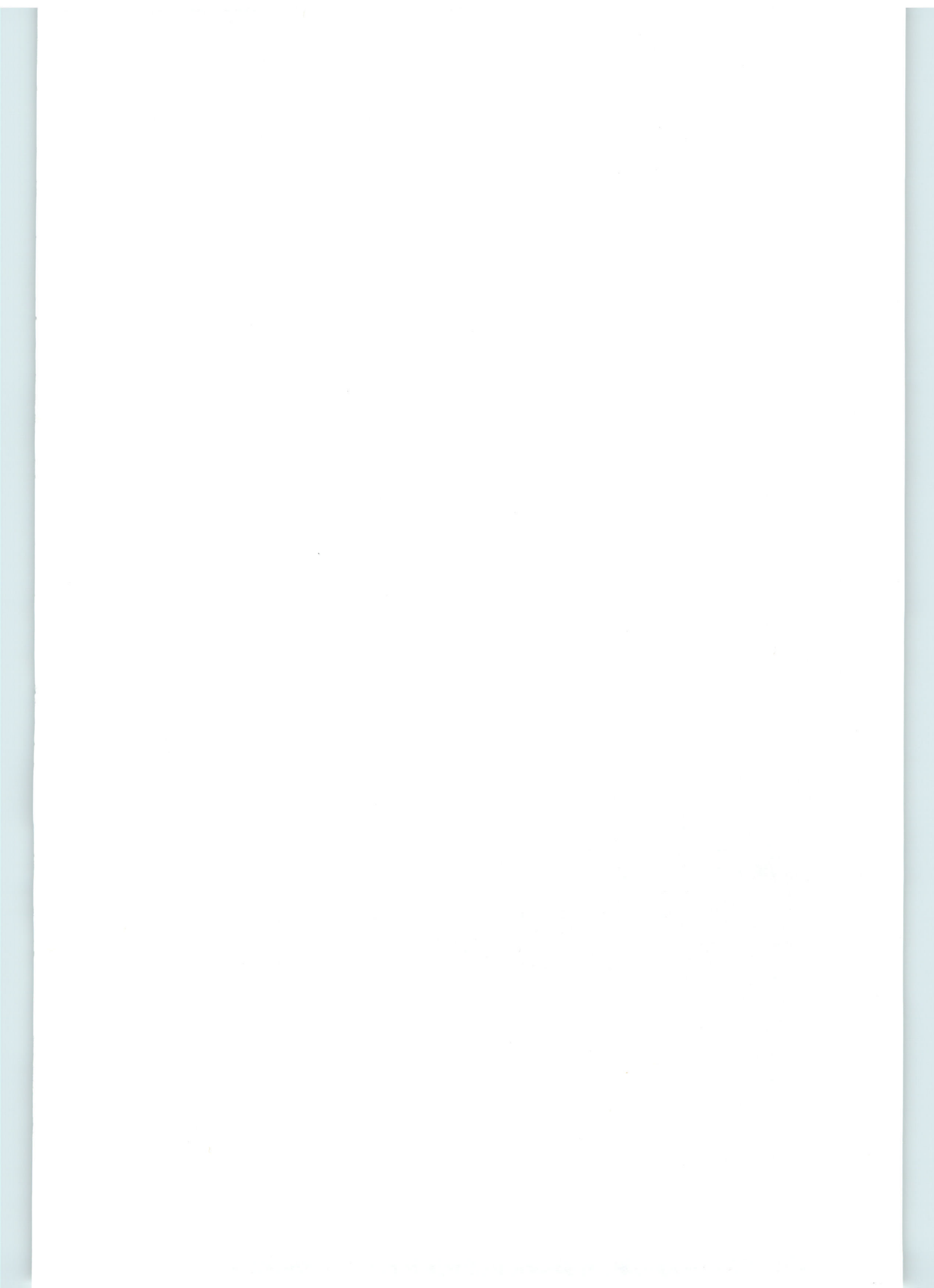
Andreas Becker: Analyse der Planungsverfahren der KI im Hinblick auf ihre Eignung für die Arbeitsplanung
86 Seiten

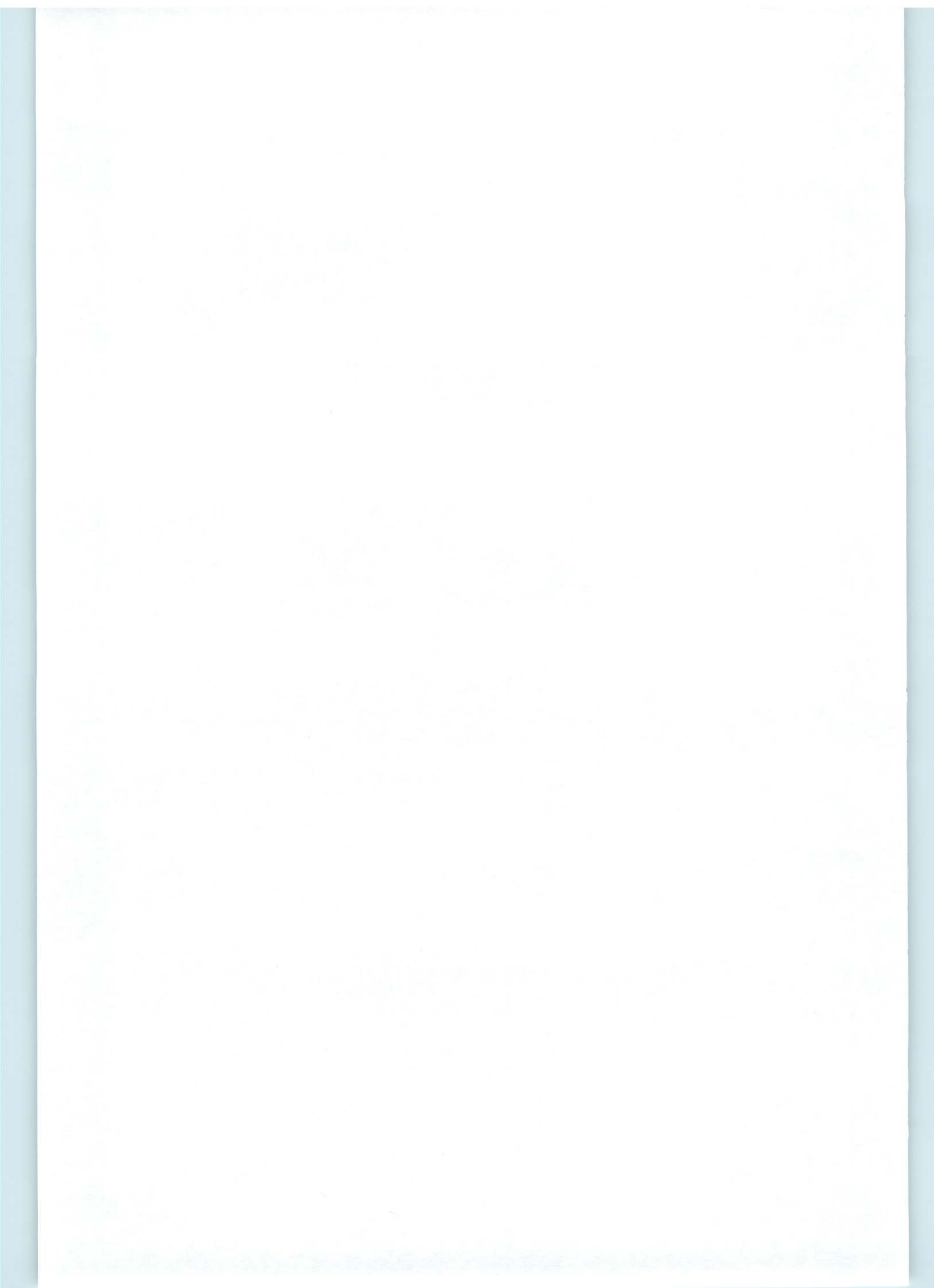
D-91-18

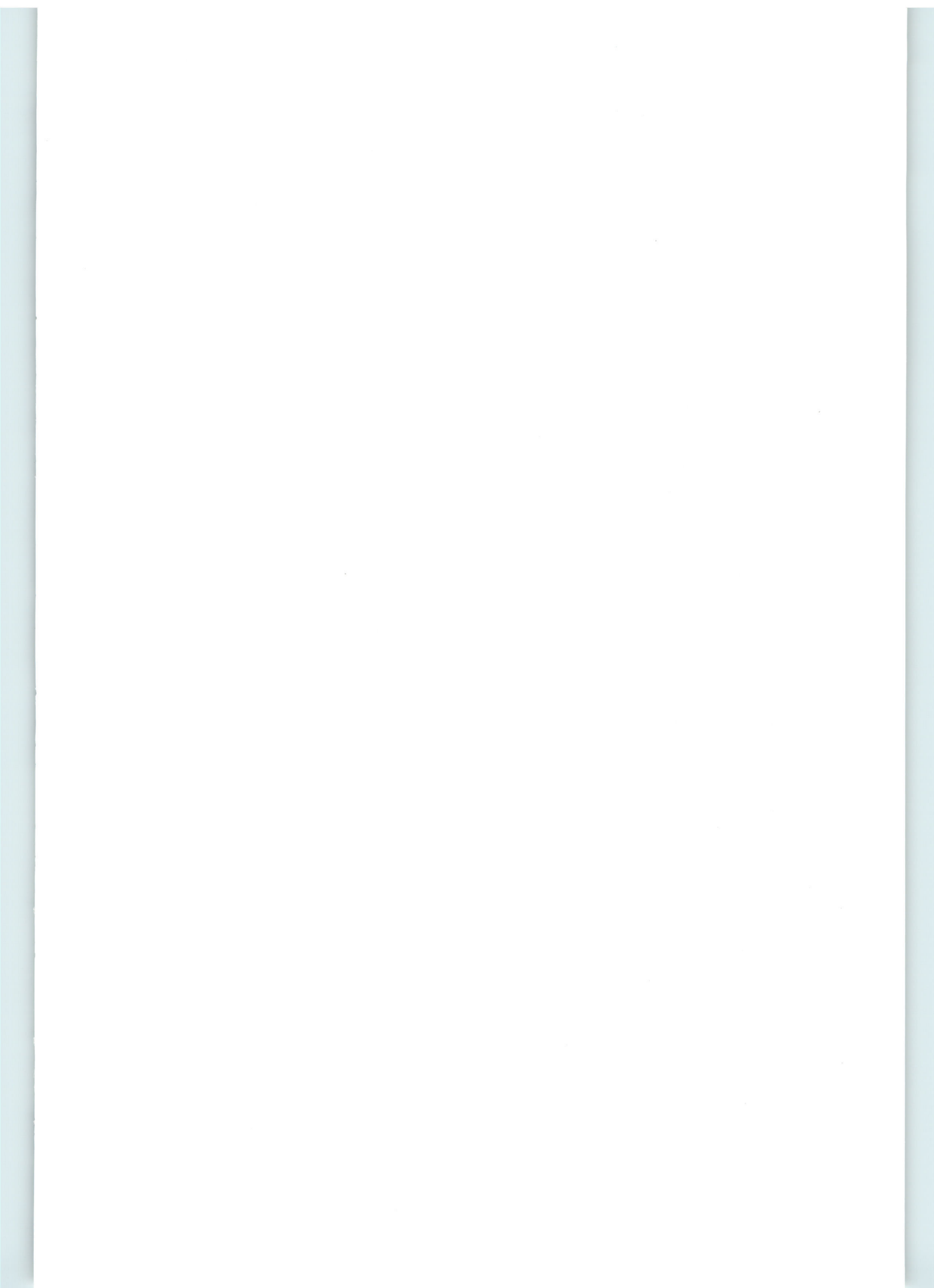
Thomas Reinartz: Definition von Problemklassen im Maschinenbau als eine Begriffsbildungsaufgabe
107 Seiten

D-91-19

Peter Wazinski: Objektlokalisierung in graphischen Darstellungen
110 Seiten







**Perspektiven zur Kombination von automatischem Animationsdesign
und planbasierter Hilfe**

Winfried Graf, Markus A. Thies

RR-92-09
Research Report