

Andreas Hemberger

*Innovationspotentiale in der
rechnerintegrierten Produktion
durch wissensbasierte Systeme*

Andreas Hemberger

*Innovationspotentiale in der
rechnerintegrierten Produktion
durch wissensbasierte Systeme*

Herausgegeben von

Professor Dr.-Ing. Klaus Feldmann,

Lehrstuhl für

Fertigungsautomatisierung und Produktionssystematik

FAPS



Carl Hanser Verlag München Wien

Als Dissertation genehmigt von der Technischen Fakultät
der Universität Erlangen-Nürnberg

Tag der Einreichung: 12. Januar 1987
Tag der Promotion: 10. April 1987
Dekan: Prof. Dr.-Ing. H. Brand
Berichterstatter: Prof. Dr.-Ing. K. Feldmann
Prof. Dr.rer.pol. H. Wedekind

CIP-Titelaufnahme der Deutschen Bibliothek

Hemberger, Andreas:
Innovationspotentiale in der rechnerintegrierten
Produktion durch wissensbasierte Systeme/Andreas Hemberger.
Hrsg. von Klaus Feldmann.- München ; Wien : Hanser, 1988.
Zugl.: Erlangen, Nürnberg, Univ., Diss., 1987

ISBN 3-446-15234-2

Dieses Werk ist urheberrechtlich geschützt.
Alle Rechte, auch die der Übersetzung, des Nachdrucks
und der Vervielfältigung des Buches oder Teilen daraus,
vorbehalten.
Kein Teil des Werkes darf ohne schriftliche Genehmigung
des Verlages in irgendeiner Form (Fotokopie, Mikrofilm
oder ein anderes Verfahren), auch nicht für Zwecke der
Unterrichtsgestaltung - mit Ausnahme der in den §§ 53,
54 URG ausdrücklich genannten Sonderfällen -, reproduziert
oder unter Verwendung elektronischer Systeme verarbeitet,
vervielfältigt oder verbreitet werden.

© Carl Hanser Verlag München, Wien 1988
Herstellung: Druckerei R. Angermann, Höchststadt/Aisch
Printed in Germany

V O R W O R T

Die vorliegende Dissertation entstand während meiner Tätigkeit als wissenschaftlicher Assistent am Lehrstuhl für Fertigungsautomatisierung und Produktionssystematik der Universität Erlangen-Nürnberg.

Herrn Professor Dr.-Ing. K. Feldmann, dem Leiter des Lehrstuhls für Fertigungsautomatisierung und Produktionssystematik am Institut für Fertigungstechnik, danke ich für die großzügige Förderung bei der Durchführung dieser Arbeit.

Herrn Professor Dr.-Ing. H. Wedekind, dem Leiter des Lehrstuhls für Rechnergestützte Aktivitäten am Institut für Mathematische Maschinen und Datenverarbeitung, danke ich für die Übernahme des Korreferates und für seine wohlwollende Unterstützung.

Ferner gilt mein Dank allen Studenten, die durch ihr Mitwirken wesentlich zum Gelingen der Arbeit beigetragen haben. Das gleiche gilt für die wissenschaftlichen Hilfskräfte, Herr cand. inf G. Geyer, Herr cand. ing. J. Franke und Herr cand. ing. J. Sturm.

Andreas Hemberger

INNOVATIONSPOTENTIALE IN DER RECHNERINTEGRIERTEN PRODUKTION

DURCH WISSENSBASIERTE SYSTEME

-INHALTSVERZEICHNIS-

	Seite
1. <u>EINLEITUNG</u>	1
2. <u>METHODEN UND HILFSMITTEL DER WISSENSVERARBEITUNG</u>	4
2.1 Entwicklungslinien der Wissensverarbeitung	4
2.2 Abgrenzung zur Datenverarbeitung	7
2.3 Wissensarten	9
2.4 Methoden der Wissensrepräsentation	10
2.4.1 Semantische Netze	11
2.4.2 Prädikatenlogik	13
2.4.3 Produktionsregeln	15
2.4.4 Frames	17
2.5 Methoden der Wissensmanipulation	19
2.5.1 Ablaufstrategien	20
2.5.2 Mustervergleich	20
2.5.3 Konfliktlösung	21
2.5.4 Suchverfahren	22
2.5.5 Unsicherheiten	23
2.6 Methoden der Wissensakquisition	24
2.6.1 Automatischer Wissenserwerb	24
2.6.2 Konsistenzerhaltung	25
2.7 Programmierstile	28
2.8 Softwarewerkzeuge zur Wissensverarbeitung	29
2.8.1 Die Programmiersprache LISP	31
2.8.2 Werkzeugsysteme	32

3.	<u>EINSATZGEBIETE WISSENSBASIRTER SYSTEME IN DER</u>	36
	<u>PRODUKTIONSTECHNIK</u>	
3.1	Diagnose technischer Systeme	37
3.2	Planung von Fertigungsabläufen	40
3.3	Konstruktion von technischen Produkten	45
3.4	Simulation von Fertigungsprozessen	49
3.5	Kriterien für den Einsatz von Expertensystemen	53
3.6	Entwicklungsprozeß von Expertensystemen	55
4.	<u>EIN WERKZEUGSYSTEM ZUR ENTWICKLUNG VON EXPERTENSYSTEMEN</u>	59
4.1	Die Dialogkomponente	61
4.1.1	Allgemeine Anforderungen	61
4.1.2	Dialogtechniken	62
4.1.3	Realisierte Dialogformen	64
4.2	Das Wissensrepräsentationssystem	65
4.2.1	Objekt-Frames	66
4.2.2	Hierarchische Vererbung	69
4.2.3	Assoziativzugriff	71
4.2.4	Strukturierung von Wissensbasen	72
4.2.5	Produktionsregeln	74
4.3	Die Problemlösungskomponente	77
4.3.1	Allgemeine Funktionsweise	77
4.3.2	Kontrollstrategien	79
4.3.3	Der Regelinterpretier	81
4.4	Der Simulator	86
4.4.1	Simulationsobjekte	88
4.4.2	Simulationsereignisse	91
4.4.3	Integration in das Werkzeugsystem	93
4.4.4	Simulationsablauf	95
4.5	Die Wissenserwerbskomponente	97
4.5.1	Allgemeine Anforderungen	97
4.5.2	Der Editor	98

4.6	Die Erklärungskomponente	102
4.6.1	Grundsätzliche Möglichkeiten	102
4.6.2	Realisierte Erklärungsfunktionen	103
5.	<u>MOPLAN - EIN EXPERTENSYSTEM ZUR PLANUNG VON MONTAGEANLAGEN</u>	106
5.1	Aufgabenstellung	106
5.1.1	Struktur von Montageanlagen	107
5.1.2	Der Planungsablauf des Montageexperten	110
5.2	Beurteilung der Planungsaufgabe	114
5.2.1	Möglichkeiten konventioneller DV-Systeme	114
5.2.2	Einsatzmöglichkeiten wissensbasierter Systeme	116
5.3	Der Planungsablauf mit dem Expertensystem	119
5.3.1	Beschreibung der Montageaufgabe	121
5.3.2	Lösen der Montageaufgabe	125
5.3.2.1	Überprüfung des Rationalisierungspotentials	125
5.3.2.2	Festlegung der Montagestruktur	126
5.3.2.3	Funktionskomplexe und Typauswahl	129
5.3.2.4	Geräteauswahl	130
5.3.2.5	Ermittlung der Zeiten	135
5.3.2.6	Blockung der Montageanlage	138
5.3.2.7	Simulation des Einsatzverhaltens	139
5.3.2.8	Erstellung des Groblayouts	140
5.3.2.9	Ermittlung der Kenndaten	142
5.3.3	Ausgabe der Planungsergebnisse	143
5.4	Die Wissensbasen	145
5.4.1	Die Gerätewissensbasis	146
5.4.2	Die Teilewissensbasis	149
5.4.3	Die Technologiewissensbasis	150
5.4.4	Die Planungsregeln	151
5.4.5	Die Metaregeln	153

5.5	Der Wissenserwerb	154
5.5.1	Systematik des Wissenserwerbs	154
5.5.2	Wissensdokumentation	157
6.	<u>DIE CAD-KOPPLUNG</u>	159
6.1	Das CAD-System ROMULUS	159
6.2	Modellieren von rechnergeführten Produktionssystemen	161
6.3	Die CAD-Bibliotheken	163
6.4	Die Kopplung MOPLAN - ROMULUS	168
7.	<u>INTEGRATION IN EIN CIM-KONZEPT FÜR DIE MONTAGE</u>	169
7.1	Erläuterungen zu CIM	169
7.2	Integration in das Anwendungskonzept	172
7.3	Integration in die Softwareumgebung	180
7.4	Integration in die Systemumgebung	182
8.	<u>ANWENDUNGSBEISPIEL</u>	186
9.	<u>ALTERNATIVE EINSATZMÖGLICHKEITEN DES WISSENSBASIERTEN WERKZEUGSYSTEMS</u>	190
9.1	Planung von Flexiblen Fertigungssystemen	190
9.2	Planung von Fertigungsabläufen	192
10.	<u>ZUSAMMENFASSUNG</u>	195
11.	<u>LITERATURVERZEICHNIS</u>	197

1. Einleitung

Die Erschließung von Innovationspotentialen in der Produktionstechnik war schon immer ein wichtiger Garant für den technischen Fortschritt und die Wettbewerbsfähigkeit der Unternehmen. Gerade in der heutigen Zeit werden in verschiedenen Schlüsseltechnologien wieder Grundsteine für zukünftige Marktpositionen und Marktanteile gelegt. Es muß deshalb ein vordringliches Ziel der Forschungsaktivitäten sein, die technologische Abhängigkeit in einigen "high-tech"-Bereichen zu verringern oder gar aufzuheben.

Einer dieser Bereiche, in denen technologische Rückstände zu Japan und den USA existieren, ist zweifellos die "Künstliche Intelligenz" und Wissensverarbeitung. Experten erwarten von dieser Technologie, daß sie die Informationsverarbeitung in den nächsten zehn bis zwanzig Jahren revolutionieren wird. Hierzulande ist die Informationsverarbeitung aber gerade erst dabei, auf breiter Basis in die Produktionstechnik Einzug zu halten. In diesem Zusammenhang liegt es deshalb nahe, die Einsatzmöglichkeiten wissensbasierter Konzepte in der Produktionstechnik zu untersuchen.

Betrachtet man nun davon unabhängig die Produktionstechnik im Hinblick auf Möglichkeiten zur Produktivitätssteigerung genauer, so kristallisieren sich dabei zwei Schwerpunktsthemen heraus. Zum einen ist dies der Produktionsbereich der "Montage" und zum anderen die informationstechnische Integration der einzelnen Unternehmensbereiche zur rechnergeführten Fabrik unter dem Schlagwort CIM [1][2].

Schon seit geraumer Zeit werden rechnergestützte Verfahren zur Automatisierung isolierter Teilbereiche in den Unternehmen eingesetzt [3][4]. Schlagworte wie CAD, CAP, CAM oder CAQ* stehen für

- * CAD = Computer Aided Design
- CAP = Computer Aided Planning
- CAM = Computer Aided Manufacturing
- CAQ = Computer Aided Quality Ensurance

diese "Automatisierungsinselfn". Die Fabrik der Zukunft soll nun durch Integrationen der Informationsstrukturen im Rahmen von CIM entstehen [5]. Doch zur Realisierung dieser Zielsetzung fehlt noch eine wesentliche Komponente, denn die "konventionelle" Datenverarbeitung erreicht ihre Grenzen dort, wo unstrukturirtes, vages Wissen verarbeitet werden soll und wo die Abläufe nicht durch fest formulierte Algorithmen erfassbar sind. Hier eröffnen die neuen Technologien der Künstlichen Intelligenz und Wissensverarbeitung vielfältige Möglichkeiten.

Die zweite große Produktivitätsreserve liegt im Bereich der Montage. Während in der spanenden Fertigung beispielsweise die NC-Technik längst etabliert ist, laufen Handhabungs- und Montagevorgänge noch weitgehend manuell ab. Die Gründe hierfür liegen in der Komplexität und dem Schwierigkeitsgrad der Prozesse. Das gleiche gilt für die entsprechenden Planungsvorgänge. Auch hier treten komplexe Problemstellungen auf, die von einer Vielzahl unterschiedlicher und schwer qualifizierbarer Einflußfaktoren abhängig sind. Aber die Fabrik der Zukunft muß auch die Montage in ihre Informationsstrukturen integrieren.

Aus dieser Sicht der Situation und Entwicklung ergeben sich nun klar die Zielsetzungen der vorliegenden Arbeit:

- 1) Konzeption und Realisierung eines Werkzeugsystems zur wissensbasierten Planung, Konfiguration und Simulation von technischen Systemen.
- 2) Somit Erschließung von Innovationspotentialen in der Produktionstechnik.
- 3) Exemplarische Anwendung des Werkzeugsystems bei der Planung von Montageanlagen.
- 4) Integration in ein Konzept der Rechnerintegrierten Produktion (CIM).

Zu diesem Zwecke werden eingangs Methoden und Hilfsmittel der Wissensverarbeitung vorgestellt. Die Zielsetzung dieses Kapitels ist es, einen groben Überblick zum Stand der Technik zu geben.

Das nachfolgende Kapitel beschäftigt sich mit dem Einsatz wissensbasierter Systeme in der Produktion. Zu Beginn werden existierende Ansätze aus verschiedenen Bereichen vorgestellt. Eine Beschreibung der Einsatzkriterien und der Entwicklungsphasen einer erfolversprechenden Realisierung schließen sich an.

Im Rahmen des vierten Kapitels wird die Konzeption des wissensbasierten Werkzeugsystems (WWS) zur Lösung von Planungs- und Konfigurationsaufgaben vorgestellt.

Als exemplarische Anwendung dieses Systems beschreibt Kapitel fünf ein Expertensystem zur Planung von Montageanlagen (MOPLAN). Den Aufbau der CAD-Bibliotheken und die Integration in ein CIM-Konzept erläutern Kapitel sechs und sieben. Im Anschluß daran zeigt Kapitel acht Planungsergebnisse des Expertensystems MOPLAN anhand eines praktischen Beispiels.

Alternative Einsatzgebiete für das wissensbasierte Werkzeugsystem und damit weitere Innovationspotentiale werden im letzten Kapitel beschrieben.

2. Methoden und Hilfsmittel der Wissensverarbeitung

2.1 Entwicklungslinien der Wissensverarbeitung

Wohl kaum ein Begriff aus dem Bereich der Informationsverarbeitung wurde in der Vergangenheit derart strapaziert wie der der Künstlichen Intelligenz (KI). Analog zu vielen anderen Schlagworten kommt auch der Ausdruck KI aus dem Englischen und steht für "Artificial Intelligence" (AI). Erstmals benutzt wurde der Begriff von J. McCarthy 1958, als er das Ziel einer Konferenz möglichst kurz und prägnant zusammenfassen wollte.

Lange Zeit wurde die Künstliche Intelligenz als ein unbedeutendes Nebenprodukt der wesentlichen Informatikgebiete betrachtet. Außer auf militärischem Gebiet sah man keine lohnenden Anwendungsmöglichkeiten. Dies änderte sich schlagartig, als in Japan das Programm zur Entwicklung der 5. Computergeneration aufgestellt wurde. In diesem Programm, das im wesentlichen die Entwicklung leistungsfähigerer und intelligenterer Computer zum Ziel hat, spielt die Künstliche Intelligenz eine zentrale Rolle [6].

Dem japanischen Programm folgten bald darauf ähnliche Anstrengungen anderer Staaten. Auch in der Bundesrepublik entstanden staatlich geförderte Verbundprojekte [7]. Die Forschung und Lehre an den Universitäten wurde forciert. Die KI stand nun im Mittelpunkt des Interesses und wird als bedeutende Komponente bei der Bewältigung der Zukunftsaufgaben angesehen. Dabei ist die Künstliche Intelligenz kein exakt definierter Begriff, sondern entwickelt sich im Verständnis derer, die auf dem Gebiet arbeiten, ständig weiter. Die Teilgebiete der KI reichen von der Sprach- und Bildverarbeitung über Robotik, Expertensysteme, Programmiersprachen und Programmierumgebungen bis hin zu Kognitionswissenschaften, Spielen, Theorembeweisern und Planungssystemen (Bild 1) [8]. Dabei kann man zwischen grundlegenden KI-Methoden und deren Anwendungen in komplexen Systemen unterscheiden.

Parallel zu der ständigen Ausdehnung von Anwendungs- und Einsatzgebieten entwickelt sich auch der Markt für KI-Produkte

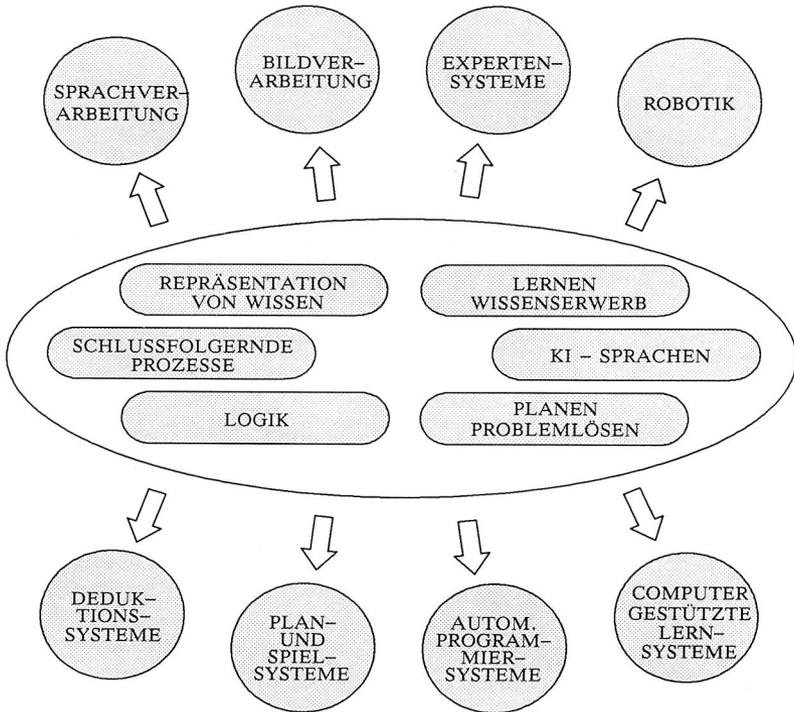


Bild 1: Teilgebiete der KI

(Bild 2). Die Computerindustrie erwartet hier erhebliche Zuwachsraten, die das ohnehin überdurchschnittliche Wachstum weiter stimulieren sollen. Die Grenzen dieser Entwicklung können vom derzeitigen Standpunkt aus noch nicht erkannt werden. Zukünftige KI-Produkte werden mit konventionellen Systemen verschmelzen und nicht mehr isoliert arbeiten [9]. Auf längere Sicht ist es somit fraglich, ob dann noch von einem gesonderten KI-Markt gesprochen werden kann.

Ein wichtiges Anwendungsgebiet von KI-Methoden sind die wissensbasierten Systeme. Eine spezielle Ausprägung davon wird vielfach auch Expertensysteme genannt. Eine allgemein akzeptierte Be-

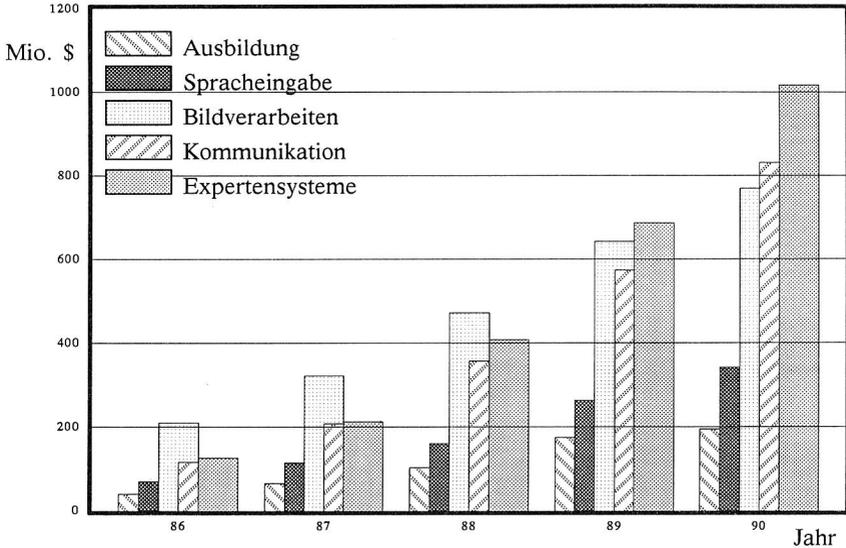


Bild 2: Marktentwicklung von KI-Produkten

Quelle: [10]

griffsbestimmung von Expertensystemen kann nicht angegeben werden; wohl aber einige Aspekte, die zur Unterscheidung und Klassifizierung hilfreich sind [9]:

- Expertensysteme sind wissensbasierte Programmsysteme zur automatischen oder interaktiven Lösung spezieller Symbolmanipulationsprobleme. Sie sind wissensbasierte Problemlöser, die auf der Basis einschlägigen Fachwissens durch gezieltes, schlußfolgerndes Verknüpfen unter Einbeziehung problemspezifischer heuristischer Methoden arbeiten (methodischer Aspekt).
- Die zu lösenden Probleme entspringen aus realen Bedürfnissen existierender Tätigkeitsbereiche (programatischer Aspekt).
- Die zu lösenden Probleme sind von beträchtlicher Schwierigkeit und erfordern zu ihrer Bearbeitung normalerweise hochqualifizierte Experten (qualitativer Aspekt).

Danach bilden Expertensysteme bereits vorhandenes Expertenwissen nach, ersetzen die menschliche Expertise jedoch nicht. Sie übernehmen vielmehr Aufgaben qualifizierter Spezialisten.

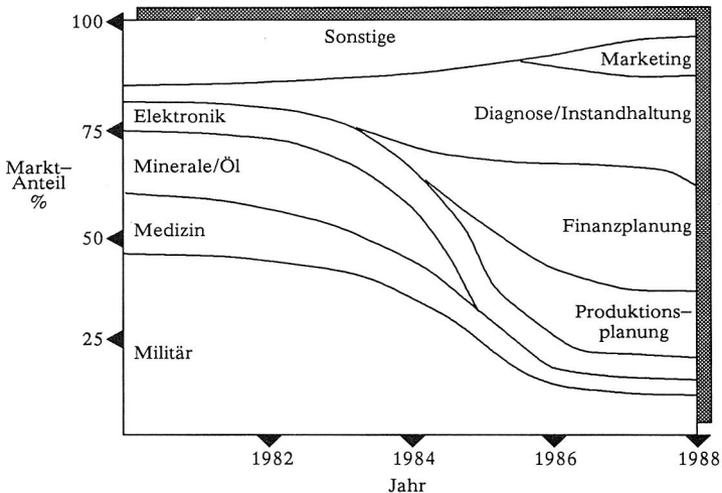


Bild 3: Einsatzgebiete von Expertensystemen

Quelle: [10]

Die Einsatzgebiete von Expertensystemen sind zeitlichen Verschiebungen unterworfen (Bild 3). Dabei wird eine deutliche Verlagerung der Schwerpunkte vom militärischen in den zivilen Bereich erkennbar. Neben der Finanzplanung haben die Diagnose und Instandhaltung sowie die Produktionsplanung die größten Marktanteile. Hieraus geht hervor, daß gerade im Bereich der Produktionstechnik die wichtigsten Innovationspotentiale zu sehen sind.

2.2 Abgrenzung zur Datenverarbeitung

Seit Expertensysteme und Wissensverarbeitung ins Schlaglicht des Interesses getreten sind, wird immer wieder die Frage nach dem Unterschied zwischen den konventionellen Computertechniken und

der Wissensverarbeitung gestellt. Eine exakte Definition zu diesem Thema findet man nirgends. Das hat seinen Grund. Die Künstliche Intelligenz und Wissensverarbeitung ist keine exakt definierte und abgeschlossene Wissenschaft. Im Rahmen ihrer Evolution hat sie verschiedene Entwicklungen hervorgebracht und entwickelt sich verstärkt auch weiter. Dem gegenüber werden auch konventionelle Softwaresysteme immer "intelligenter". Die Künstliche Intelligenz sollte deshalb auch mehr als eine Sammlung von Methoden zur Steigerung der Intelligenz von Computerprogrammen verstanden werden [8]. Wie man sich leicht vorstellen kann, schaffen hierbei unterschiedliche Anwendungen auch unterschiedliche Methoden. An dieser Stelle soll deshalb auch nur der Versuch einer groben Abgrenzung unternommen werden, der nicht den Anspruch auf Vollständigkeit und Gültigkeit in jedem Einzelfall erhebt. Angestrebt sind vielmehr generelle und tendenzielle Aussagen. Danach ist eine Klassifizierung sinnvoll, die zwischen

- Datenverarbeitung,
- Informationsverarbeitung und
- Wissensverarbeitung

unterscheidet. Die Datenverarbeitung beschränkt sich auf vorhersehbare und exakt planbare Abläufe, bei denen die Problemlösung bis ins Detail exakt programmiert ist. Das dazu notwendige Wissen steckt in den programmierten Algorithmen und Dateien. Die Lösungsstrategie ist fest implementiert und kann nur durch Neuprogrammierung geändert werden.

Bei der Informationsverarbeitung weisen die manipulierten numerischen und nichtnumerische Daten komplexe Strukturen auf, die beispielsweise in Datenbankmanagementsystemen [11] repräsentiert sind. Speicherung, Aktualisierung und Wiederbeschaffung von Informationen kann mit Sprachen der 4. Generation erfolgen. Die Informationsbestände sind in verteilten Systemen logisch miteinander verknüpft.

In Abgrenzung dazu existiert bei der Wissensverarbeitung ein internes Schlußfolgerungsmodul zur Lösung von gestellten Aufgaben.

Spezielles Wissen beschreibt, welche Möglichkeiten das System zur Problemlösung hat, und wann welche Lösungswege einzuschlagen sind. Der exakte Lösungsweg ist nicht per Programm vorgegeben, sondern wird vom System in Abhängigkeit von erreichten Zwischenergebnissen selbständig ermittelt. Spezielles Metawissen (Wissen "über" Wissensverarbeitung) steuert dabei übergeordnete Strategien.

2.3 Wissensarten

Beginnt man den Begriff "Wissen" genauer zu untersuchen, stößt man auf eine Vielzahl unterschiedlicher Aspekte. Nach STOYAN [8] hat Wissen einen Bezug, und zwar auf

- Objekte,
- Ereignisse,
- Prozesse und
- Handlungen.

Dabei hat Wissen prädikativen Charakter, wenn es den Dingen Namen, Eigenschaften und Beziehungen zuweist. Wissen ist abstrahierend, wenn es zur Klassifikation von Objekten verwendet wird. Bezieht sich Wissen auf Prozesse, so beschreibt es Änderungen von Prozeßeigenschaften, Veränderungen von Objektbeziehungen, Eintreten von Ereignissen und Abhängigkeiten verschiedener Ereignissen untereinander. Bezieht sich Wissen auf Handlungen, so beschreibt es die Auswirkungen und die Träger von Handlungen mit ihren Zielen und den auslösenden Fakten.

Neben einem Bezug hat Wissen auch Eigenschaften. Wieder unterscheidet man

- statisches,
- dynamisches und
- vages

Wissen. Statisches Wissen bleibt immer unverändert. Objekte be-

halten ihren Namen, ihre Beziehungen, ihre Eigenschaften. Daten behalten ihren Wert. Es handelt sich um Fakten.

Dynamisches Wissen ist Änderungsprozessen unterworfen. So ändert sich der Wert einer Variablen, die den Zustand eines Systems beschreibt. Ereignisse sind meist Auslöser dieser Veränderungen.

Vages Wissen ist nicht sicher. Es existiert keine exakte Aussage. Nur mit Hilfe von Vermutungen und Wahrscheinlichkeitsangaben lassen sich vage Aussagen verarbeitbar machen.

Die zentrale Problematik bei der Entwicklung von Expertensystemen ist der Umgang mit dem Wissen. Hierbei unterscheidet man Methoden zur

- Wissensrepräsentation,
- Wissensmanipulation und
- Wissensakquisition.

Im Rahmen der vorliegenden Arbeit soll nur ein Überblick zu den aufgeführten Methoden gegeben werden. Ausführliche Beschreibungen und Diskussionen finden sich in der entsprechenden Literatur [12][13][14].

2.4 Methoden der Wissensrepräsentation

Die Strukturierung, Speicherung und Verwaltung des Wissens ist von entscheidender Bedeutung für die Effizienz und Leistungsfähigkeit wissensbasierter Systeme. Die einzelnen Methoden zur Problemlösung und Wissensmanipulation werden weitgehend von der Repräsentationsform beeinflusst. Es liegt also nahe, daß ein Expertensystem umso besser funktioniert, je treffender es die realen Problemstellungen und Abläufe intern abbilden und repräsentieren vermag. Das heißt aber auch, daß es bis heute noch keinen Ansatz mit universellem Anwendungsbereich gibt, sondern, daß für jede Problemstellung speziell abgestimmte Lösungsmethoden vorzuziehen sind.

Im nächsten Abschnitt soll nun ein Überblick zu den bekannten Methoden der Wissensrepräsentation gegeben werden. Die wichtigsten sind:

- Semantische Netze,
- Prädikatenlogik,
- Produktionsregel,
- Frames.

Im Hinblick auf ein tieferes Verständnis sei an dieser Stelle auf die weiterführende Literatur verwiesen.

2.4.1 Semantische Netze

In einem Semantischen Netz wird das Wissen in Form von Graphen gespeichert. Dabei stellen die Knoten des Graphen Fakten oder Ereignisse dar, während die Kanten die Beziehungen der einzelnen Objekte zueinander repräsentieren [15]. Bild 4 zeigt ein einfaches Beispiel.

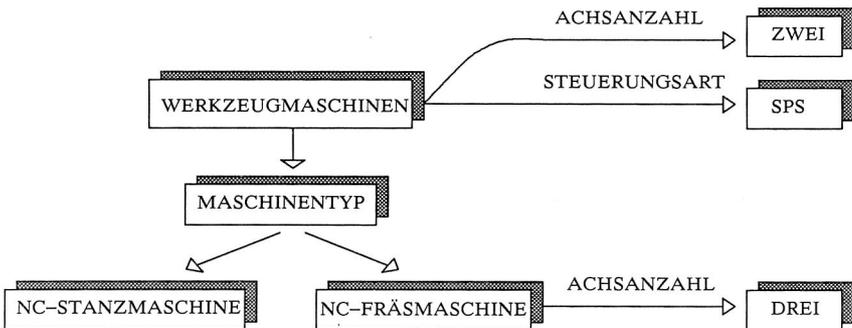


Bild 4: Einfaches Semantisches Netz

Dieses semantische Netz enthält explizit die folgenden fünf Tatsachen:

- (E1) Werkzeugmaschinen besitzen speicherprogrammierbare Steuerungen (SPS).
- (E2) Werkzeugmaschinen besitzen zwei Verfahrachsen.
- (E3) NC-Stanzmaschinen sind Werkzeugmaschinen.
- (E4) NC-Fräsmaschinen sind Werkzeugmaschinen.
- (E5) NC-Fräsmaschinen besitzen drei Verfahrachsen.

Diese Tatsachen sind angesichts der graphischen Darstellung des Semantischen Netzes unmittelbar einsichtig. Damit ist bereits ein Vorteil der Semantischen Netze erwähnt. Sie entsprechen genau der menschlichen Vorstellung über die Strukturierung und Verknüpfung von Wissensfakten. Ein weiterer Vorteil ist auch der günstige Zugriffsmechanismus. Alle zu einem Objekt gehörenden Daten sind vom jeweiligen Objektknoten aus direkt oder indirekt erreichbar, ohne daß die gesamte Wissensbasis durchsucht werden muß.

Semantische Netze ermöglichen zudem noch einen einfachen Zugriff auf implizites Wissen über mit Kanten verbundene Untermengen. Automatische Schlußfolgerungsprozesse laufen entlang der Kanten ab. In dem Beispiel aus Bild 4 sagen die Kanten MASCHINENTYP aus, daß sowohl NC-STANZMASCHINEN als auch NC-FRÄSMASCHINEN zu den WERKZEUGMASCHINEN gehören. Somit treffen alle Eigenschaften der Obermenge (WERKZEUGMASCHINEN) auch auf die Untermengen (NC-FRÄS-, NC-STANZMASCHINEN) zu, sofern diese nicht eigene Kanten mit identischer Bedeutung besitzen.

Im Beispiel von Bild 4 kann man über die STEUERUNGSART von NC-STANZ-, und NC-FRÄSMASCHINEN schließen, daß beide speicherprogrammierbare Steuerungen (SPS) besitzen. Da sie keine explizite Kante STEUERUNGSART haben, sie aber über die Untermengenkante MASCHINENTYP als Werkzeugmaschinen identifizierbar sind, erben sie deren STEUERUNGSART. Dieser Mechanismus wird deshalb auch als Vererbung von Eigenschaften bezeichnet. Im Falle der ACHSENZAHL funktioniert diese Vererbung im Bezug auf die NC-STANZMASCHINEN in derselben Weise. Bei den NC-FRÄSMASCHINEN aber wird sie durch

die explizite Kante ACHSENZAHL des betreffenden Knotens unterdrückt.

Bei dem vorliegenden Beispiel liefert der Vererbungsmechanismus zu den fünf expliziten Tatsachen zusätzlich noch die drei impliziten (I1), (I2) und (I3):

- (I1) NC-Stanzmaschinen besitzen speicherprogrammierbare Steuerungen (SPS).
- (I2) NC-Fräsmaschinen besitzen speicherprogrammierbare Steuerungen (SPS).
- (I3) NC-Stanzmaschinen besitzen zwei Verfahrachsen.

Diese drei Tatsachen sind nicht explizit im Programm enthalten - weder als Programmcode noch als Daten. Ein Programm, welches nur durch Kenntnis der Fakten (E1) - (E5) auch die Fakten (I1) - (I3) herleiten kann, zählt man gegenwärtig zum Bereich der Künstlichen Intelligenz.

Die Nachteile der Semantischen Netze sind ihre schnell wachsende Komplexität, sobald eine größere Menge an Umweltwissen zu repräsentieren ist sowie ihre doch recht starre Struktur.

2.4.2 Prädikatenlogik

Das Prädikatenkalkül 1.Ordnung aus dem Bereich der formalen Logik ist ein klassischer Ansatz zur Wissensrepräsentation [16]. Aussagen über den darzustellenden Wissensbereich werden in logischen Formeln, den sogenannten Klauseln übersetzt. Diese Klauseln werden als Axiome in das System integriert. Zur Anwendung kommen ausschließlich die folgenden Operatoren und Quantoren:

- | | | | | | | |
|---|---------------|---------------|---|-----------------|---|-----------|
| - | UND : | \wedge | - | ALLQUANTOR | : | \forall |
| - | ODER : | \vee | - | EXISTENZQUANTOR | : | \exists |
| - | NICHT : | \neg | | | | |
| - | IMPLIKATION : | \rightarrow | | | | |

Mit diesen Hilfsmitteln der formalen Logik läßt sich etwa der folgende Sachverhalt darstellen:

Tatsachen:

(E2) Werkzeugmaschinen besitzen zwei Verfahrsachsen.

(E3) NC-Stanzmaschinen sind Werkzeugmaschinen.

Logikform:

(E2') $\forall X(X \text{ ist Werkzeugmaschine} \rightarrow X \text{ hat zwei Verfahrsachsen})$

(E3') $\forall X(X \text{ ist NC-Stanzmaschine} \rightarrow X \text{ ist Werkzeugmaschine})$

Interpretation:

Wenn X eine Werkzeugmaschine ist, dann besitzt X zwei Verfahrsachsen.

Wenn X eine NC-Stanzmaschine ist, dann ist X eine Werkzeugmaschine.

Mit Hilfe der Ableitungsregeln der formalen Logik (im vorliegenden Fall die Transitivität) läßt sich nun sofort die folgende implizite Tatsache generieren:

(I3') $\forall X(X \text{ ist Stanzmaschine} \rightarrow X \text{ hat zwei Verfahrsachsen})$

Dies entspricht genau der impliziten Tatsache (I3) aus dem vorherigen Beispiel (Kap. 2.4.1). Im Unterschied zu den Semantischen Netzen ist im Prädikatenkalkül die Tatsache (I3') allerdings beweisbar. Ausgehend von den explizit gegebenen Fakten sind die abgeleiteten impliziten Tatsachen aufgrund der logischen Ableitungsregeln garantiert korrekt. Diese Garantie kann keine andere Methode der Wissensrepräsentation gewährleisten.

Ein weiterer Grund für die Beliebtheit des Prädikatenkalküls ist die Automatisierbarkeit der Ableitungen. Es wird damit möglich, automatische Theorembeweiser zu programmieren, die überprüfen, ob bestimmte Tatsachen aus einem vorgegebenen Sachverhalt abgeleitet werden können und somit Gültigkeit besitzen. Gegen die Anwendung der Prädikatenlogik als Methode der Wissensrepräsentation spricht allerdings die hier besonders ausgeprägte kombinatorische Ex-

plosion von Wissenselementen während der Prozesse des Problemlösens. Bei zunehmender Komplexität der Wissensbereiche stößt dieses Verfahren an die Grenzen der Performanzkriterien. Neuerdings versucht man die unkontrollierte Ausdehnung der Such- und Lösungsräume durch gezielte Strategien beim Schlußfolgerungsprozeß einzudämmen.

2.4.3 Produktionsregeln

Diese Methode der Wissensrepräsentation wird heute in den meisten bekannten Expertensystemen angewendet [17]. Sie basiert auf der Darstellung des Wissens in Form von WENN-DANN - Regeln, den sogenannten Produktionen. Die Produktionen setzen sich aus einer linken und einer rechten Regelseite zusammen, wobei die linke Seite eine Folge von Bedingungen und die rechte Seite eine Folge von Aktionen beschreibt.

Linke Seite: WENN ein Werkstück aus dem Puffer entnommen wird
 UND der Puffer dadurch leer wird

Rechte Seite: DANN fülle den Puffer aus dem Lager wieder auf
 UND melde dies dem Zellenrechner

Im Bedingungsteil einer Regel sind alle Voraussetzungen aufgeführt, die erfüllt sein müssen, damit die im Aktionsteil definierten Handlungen und Aktionen angestoßen oder ausgeführt werden können.

Das vollständige System besteht nun aus einer Menge derartiger Produktionen, aus einer aktuellen Beschreibung des Systemzustandes (Arbeitsspeicher) sowie aus Funktionen, die das Systemverhalten kontrollieren. Letztere haben die Aufgabe, diejenigen Regeln zu finden, die auf den augenblicklichen Systemzustand anwendbar sind. Dabei wird geprüft, inwieweit der Bedingungsteil der Regeln vom Systemzustand erfüllt ist.

Für die weite Verbreitung, die Produktionssysteme im Bereich der

Wissensverarbeitung gefunden haben, sind als Gründe ihre Modularität und ihre Verständlichkeit anzuführen. Das gesamte Wissen ist in Form von Regeln gegeben, die voneinander völlig unabhängig sind. Damit können jederzeit einzelne Wissensbausteine geändert, gelöscht oder hinzugefügt werden, ohne daß sonstige umfangreiche Änderungen am System erforderlich sind. Insbesondere große Expertensysteme mit mehreren hundert Regeln werden so komponentenweise erstellt und auch im praktischen Einsatz ständig an neue Gegebenheiten angepaßt. Aufgetretene Fehlleistungen des Systems können in vielen Fällen durch Austausch oder Änderung der Produktionen korrigiert werden. Dadurch sind Produktionssysteme vielseitig einsetzbar und in einem Maße wartungsfreundlich, wie es bei keiner anderen Methode der Wissensrepräsentation der Fall ist.

Die Verständlichkeit von Produktionssystemen ist gegeben, weil das Wissen in genau der Form gespeichert wird, wie es ein menschlicher Experte benutzt, um seine Entscheidungen zu erklären und zu begründen. Dadurch fällt es dem Experten relativ leicht, sein Wissen auf den Rechner zu übertragen. Dies wird insbesondere dann noch begünstigt, wenn das Expertensystem über eine komfortable Kommunikationsschnittstelle verfügt.

Diesen Vorteilen stehen allerdings auch einige Nachteile gegenüber. Gerade die strenge Modularität von reinen Produktionssystemen verursacht eine Reihe von Effizienzproblemen. Vor jedem einzelnen Schritt muß die Anwendbarkeit aller Produktionen erneut überprüft werden. Es gibt keine Möglichkeit der Interaktion von Produktionen. Sie können sich nicht gegenseitig aufrufen oder beeinflussen. Hierarchische Kontrollstrukturen sind in reinen Produktionssystemen nicht vorgesehen.

Weiterhin sind Produktionssysteme zwar sehr gut geeignet zur Speicherung von Wissen in Form von Ereignissen. Wissen in Form von Fakten hingegen ist nur implizit vorhanden.

Ideal wäre eine Art Mischform zwischen Semantischen Netzen zur Faktenrepräsentation auf der einen Seite und Produktionen zur

Ereignisrepräsentation auf der anderen Seite. Das im folgenden Abschnitt beschriebene Frame-Konzept ist ein Ansatz in diese Richtung.

2.4.4 Frames

Das Konzept der Rahmen (Frames) geht auf MINSKY [18] zurück und stellt eine jüngere Entwicklung auf dem Gebiet der Wissensrepräsentation dar. Hier ist es erstmals möglich, in einem gemeinsamen Ansatz sowohl deklarativ, statisches als auch prozedural, dynamisches Wissen einheitlich darzustellen. Das zugrunde liegende Prinzip ähnelt dem der Semantischen Netze, wobei es aber bei Frames möglich ist, Prozeduren sowohl zur Daten- als auch zur Flußkontrolle zu integrieren. Bild 5 zeigt ein Demonstrationsbeispiel, welches an den Sachverhalt von Bild 4 anknüpft.

RAHMEN: NC-STANZMASCHINE XYZ		
MASCHINENTYP	WERT:	NC-STANZMASCHINE
STEUERUNGSART	WERT:	---
	STANDARD:	---
ACHSENZAHL	WERT:	---
	STANDARD:	2
STANZKRAFT	WERT:	---
	STANDARD:	300 KN
MAXIMALE BLECHSTÄRKE:	WERT:	---
	STANDARD:	---
	ERMITTLUNG:	BERECHNE_ÜBER_STANZKRAFT
DREHBARE WERKZEUG-AUFNAHME	WERT:	---
	STANDARD:	NEIN
	WENN_ÄNDERUNG:	ACHSENZAHL_ANPASSEN

Bild 5: Beispielrahmen einer NC-Stanzmaschine

Ein Rahmen besteht zunächst aus einer Menge von sogenannten Slots. Diese untergliedern sich wiederum in Subslots auch Facet-

ten genannt. Mit Hilfe dieser Struktur kann man nun vielfältige Wissens-elemente verwalten und gezielte Zugriffsstrategien realisieren.

Im Beispielrahmen von Bild 5 sind als Facetten die Subslots WERT, STANDARD, ERMITTLUNG und WENN_ÄNDERUNG dargestellt. Hierbei werden die Fakten des Systems in der Facette WERT (value) abgelegt. Die Facette STANDARD (default) hat die Aufgabe, immer dann Informationen zu liefern, wenn die Facette WERT nicht belegt ist. Sind sowohl WERT als auch STANDARD unbesetzt, so tritt analog zum Semantischen Netz der Vererbungsmechanismus in Kraft, der über den Eintrag MASCHINENTYP gesteuert wird (z.B. Steuerungsart:SPS).

Zuletzt nun die wichtigste Zugriffsmöglichkeit, die Frames von den übrigen Methoden der Wissensrepräsentation unterscheidet. Unter dem Eintrag ERMITTLUNG (if-needed) kann eine Berechnungsvorschrift angegeben werden, die den gewünschten Wert ermitteln soll. Üblicherweise besteht diese Vorschrift aus einem Prozeduraufruf (attached procedure). Dieses Verfahren gewährleistet nahezu uneingeschränkte Flexibilität. Im Beispiel von Bild 5 würde etwa eine Prozedur anhand der gegebenen Stanzkraft und anderer Faktoren bestimmen, bis zu welcher Dicke Stahlbleche bearbeitet werden dürfen (MAX. BLECHSTÄRKE: 10 MM).

Zusätzlich zu den bisher erwähnten Untereinträgen, die den Zugriff auf das gespeicherte Wissen steuern, sind weitere Möglichkeiten zur automatischen Datenanpassung gegeben. So kann es Prozeduren geben, die automatisch aufgerufen werden, wenn sich der Wert eines Eintrags ändert. Man bezeichnet solche Prozeduren oft auch als "Dämonen" (demons), die über den entsprechenden Slot bzw. Subslot wachen.

Dadurch ist es möglich, die Konsistenz der in der Wissensbasis gespeicherten Daten und Fakten zu kontrollieren und in gewissem Umfang zu gewährleisten (z.B. DREHBARE_WERKZEUGAUFNAHME: bei jeder Änderung dieses Eintrags muß auch der Wert des Eintrags ACHSENZAHL geändert werden).

Das Rahmen-Konzept ist somit die einzige Methode der Wissensrepräsentation, die es gestattet, sowohl statisches als auch dynamisches Wissen explizit und zum jeweiligen Problemkreis gehörig zu speichern. Der unmittelbare Zugriff auf statisches Wissen über die WERT- bzw. STANDARD-Untereinträge sowie den Vererbungsmechanismus erfolgt schnell und ohne großen Rechenaufwand. Bei Bedarf kann auch eine beliebige, problemspezifische Prozedur aktiviert werden, um das gesuchte Wissen in gewünschter Weise zu bestimmen.

2.5 Methoden der Wissensmanipulation

In Analogie zu den multiplen Möglichkeiten der Wissensrepräsentation sind auch vielfältige Manipulationsmethoden bekannt. Manche Methoden sind direkt abhängig von der Wissensrepräsentation. Andere wiederum sind auf verschiedene Formen anwendbar. Das weite Spektrum der Künstlichen Intelligenz verschließt sich auch hier wieder einer klassifizierenden Systematik. Im Laufe der Entwicklung haben sich eben spezielle Methoden für spezielle Aufgaben herausgebildet.

Da ein umfassender Überblick zu diesem Thema das Ziel der Arbeit verfehlen und ihren Umfang sprengen würde, sei auch an dieser Stelle der Hinweis auf die weiterführende Literatur erlaubt. [12][13][14][19].

In dem vorliegenden Abschnitt soll vielmehr auf eine weitverbreitete Form wissensbasierter Systeme, nämlich auf Produktionssysteme, eingegangen werden. Nach Lehmann [20] bestehen diese aus drei Hauptkomponenten:

- 1) Eine Menge oder Folge von Produktionsregeln als Wissensbasis.
- 2) Eine Menge vorgegebener oder aktuell erzeugter Ausdrücke oder Datenelemente, die oftmals als Arbeitsspeicher, Kontext oder Kurzzeitgedächtnis bezeichnet werden.
- 3) Ein Interpretationsmechanismus, der Regeln auswählt, entsprechende Aktionen ausführt und so zur Erzeugung neuer Ausdrücke oder Modifikationen des Arbeitsspeichers beiträgt. Die globale

Ablaufstrategie solcher Systeme ist ein permanenter Zyklus des Erkennens und Handelns (recognize-act-cycle).

2.5.1 Ablaufstrategien

Dieser letztgenannte Interpretationsmechanismus führt die eigentliche Lösung der Problemstellung durch. Die zwei prinzipiellen Vorgehensweisen bei der Ablaufstrategie sind vorwärts- und rückwärtsgesteuerte Verfahren. Vorwärtsgesteuerte Verfahren sind datengetrieben, d.h. abhängig von Zwischenergebnissen und Zwischenzuständen werden neue Ergebnisse und Zustände ermittelt. Bei den Produktionsregeln wird zuerst der Bedingungsteil, auch die linke Regelseite genannt, untersucht. Sind die darin beschriebenen Voraussetzungen erfüllt, kann die Ausführung des Aktionsteils, also der rechten Regelseite, erfolgen. Für diese Strategie hat sich der Ausdruck "forward-chaining" eingebürgert.

Rückwärtsgesteuerte Verfahren sind zielgerichtet. Ausgehend von diesem Ziel werden Subziele gesucht, welche näher am Ausgangszustand liegen und somit zu diesem hinführen. Der Bedingungsteil einer Regel beschreibt also einen Zielzustand, während der Aktionsteil mögliche Teilziele oder Wege zum Erreichen dieser Teilziele enthält. Die Abarbeitungssequenz ist wieder von der linken Regelseite zur rechten Regelseite gerichtet. Im Gegensatz zum "forward-chaining" spricht man hier von "backward-chaining".

Die Wahl der Ablaufstrategie ist in hohem Maße von der vorgegebenen Aufgabenstellung abhängig. So sind rückwärtsgesteuerte Verfahren hauptsächlich für Diagnose- und Konsultationssysteme geeignet. Vorwärtsgesteuerte Verfahren eignen sich dagegen mehr für Konfigurations- und Syntheseaufgaben.

2.5.2 Mustervergleich

Die eigentlichen Schlußfolgerungsprozesse beim Interpretieren der Produktionsregeln laufen meist nach einem Mustervergleichsver-

fahren (Pattern Match) ab. Bei diesem Verfahren werden die Elemente des Bedingungssteils einer Regel mit den Elementen des sog. Arbeitsbereichs verglichen. Der Arbeitsbereich, vielfach auch "Working-Memory" genannt, beschreibt den aktuellen Zustand des Problemlösungsprozesses und kann auch als das rechnerinterne Abbild der Expertenaufgabe verstanden werden. Die zu vergleichenden Elemente können aus Objekten, Variablen und Wertangaben bestehen. Fällt der Vergleich positiv aus, so werden die Elemente des Arbeitsbereichs an die der Regel gebunden. Man spricht dann von einer Instanzierung. Sind in einer Regel mehrere Bedingungen konjunktiv verknüpft, dann müssen alle Bedingungen auch verglichen werden. Es werden nur diejenigen Regeln instanziiert, bei denen der Mustervergleich für alle Elemente erfolgreich war.

2.5.3 Konfliktlösung

Entscheidend für den Erfolg eines Expertensystems ist neben dem vorhandenen Wissen auch die Existenz effizienter Kontrollstrategien. So bildet die Menge aller prinzipiell in einer bestimmten Situation ausführbaren Regeln eine Konfliktmenge, auf die nun verschiedene Auswahlstrategien angewendet werden können. Kriterien hierfür sind nach LEHMANN [20] :

- Globale Ordnungs- und Prioritätsbeziehungen unter den Regeln,
- Prioritätsbeziehungen zwischen den betroffenen Daten,
- Spezifizierungsgrad einer Regel,
- Aktualität von Regeln und Daten,
- Zufall,
- Verhinderung einer Mehrfachausführung derselben Regel.

Das Ziel der Kontrollstrategie ist die Auflösung des vorliegenden Konflikts mit Hilfe derartiger Kriterien. Nur eine Produktionsregel darf letztlich ausgeführt werden. Es wird deutlich, daß derartige Strategien den Ablauf und das Systemverhalten maßgeblich beeinflussen und somit über Erfolg und Mißerfolg entscheiden können. Aus diesem Grunde sollten möglichst viele solcher Strategien verfügbar sein.

2.5.4 Suchverfahren

Fast in jedem Expertensystem fällt die Aufgabe an, die richtigen und geeigneten Regeln, Objekte und Daten zu finden und zu bearbeiten. Hierfür wurden in der Vergangenheit ausgeklügelte Suchverfahren entwickelt. Auch bei anderen KI-Anwendungen nimmt die Suche eine zentrale Stellung ein. Dementsprechend umfangreich und häufig sind auch die Beschreibungen der vielen bisher bekannten Methoden.

Die Suche bezieht sich immer auf den sogenannten Suchraum. Dieser kann als baumartiges Gebilde verstanden werden, in dem alle vergangenen, aktuellen und zukünftig möglichen Zustände des Problemlösungsprozesses repräsentiert sind. Diese Repräsentation kann explizit vorliegen, wie beispielsweise in Form von Semantischen Netzen. Sie kann aber auch nur implizit existieren, wenn zum Beispiel Produktionsregeln in einer Regelmenge gefunden und ausgewählt werden sollen.

Es wird sofort klar, daß mit zunehmender Komplexität eine kombinatorische Zustandsexplosion entsteht, der nur mit effizienten Suchverfahren begegnet werden kann. Beispielhaft für eine Vielzahl möglicher Verfahren sind in Bild 6 "Breitensuche" und "Tiefensuche" dargestellt. Bei der Breitensuche werden zuerst alle möglichen Zustände einer Ebene überprüft, bevor Lösungen auf tieferen Ebenen weiter verfolgt werden. Im Gegensatz dazu geht die Tiefensuche sofort auf tiefere Lösungsebenen über, wenn dies möglich ist. Einen weitreichenden Überblick auch über heuristische Verfahren geben WINSTON [13] und NILSSON [14].

Im direkten Zusammenhang mit den Suchverfahren ist auch die nächste Methode, das sogenannte Backtracking, zu sehen. Backtracking heißt nichts anderes als das Zurückgehen im Suchbaum eines Zustandsraumes, wenn die Suchstrategie keine geeignete Lösung erbracht hat. Bildlich kann man sich diese Situation als einen toten Ast im Suchbaum vorstellen. Um nun im Zustandsraum zurückgehen zu können, müssen natürlich vergangene Zustandsbeschreibungen noch existieren oder aber neu aufgebaut werden. Auch

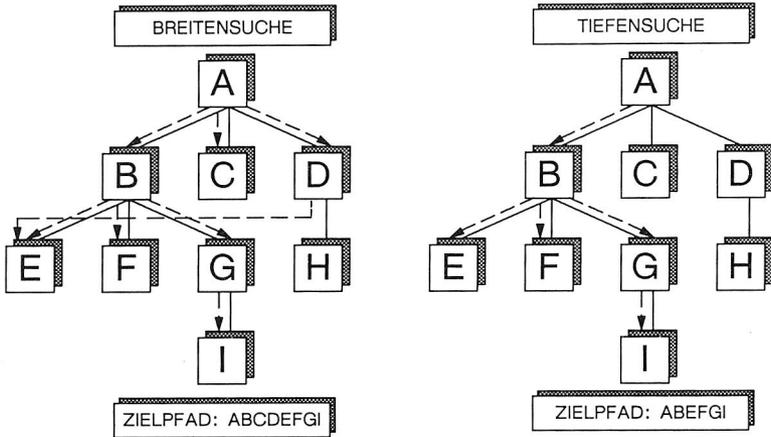


Bild 6: Suchstrategien

hier wurden in der Vergangenheit ausgefeilte Verfahren vorgestellt. Man kann zwischen den sogenannten einfachen und den intelligenten Verfahren unterscheiden. Die einfachen Methoden gehen jeweils um eine Ebene zurück, und versuchen dann wieder neue Zielzustände zu explorieren. Das wird natürlicherweise dann sehr umständlich, wenn man schon zu Beginn des Problemlösungsprozesses, also am Anfang des Baumes, in einen falschen Ast eingestiegen ist. Bei den intelligenten oder auch heuristischen Verfahren wird über mehrere Ebenen rückwärts wieder ein Aufsetzpunkt gesucht, für den noch weitere erfolgversprechende Teiläste bekannt sind. Die Schwierigkeit liegt dann darin, einen solchen Aufsetzpunkt zu erkennen.

2.5.5 Unsicherheiten

In vielen Veröffentlichungen wird als Besonderheit wissensbasierter Systeme die Fähigkeit herausgestellt, mit vagem und unsicherem Wissen umgehen zu können. Dies trifft besonders auf Diagnosesysteme und auf Beratungssysteme zu. Die Sicherheit oder Unsi-

cherheit von Wissensselementen wird meist über sogenannte Sicherheitsfaktoren (Certainty Factor) beschrieben. Diese Faktoren können dann Werte aus einem bestimmten Wertebereich einnehmen. Bei MYCIN [21], einem der ersten medizinischen Diagnosesysteme, reicht die Skala beispielsweise von "-1" bis "+1", wobei "-1" für völlig unwahrscheinlich und "+1" für sicher oder sehr wahrscheinlich steht. Andere Systeme vergeben auch Punkte auf Aussagen oder Behauptungen, die dann als Maß für die Glaubwürdigkeit dienen sollen. Als speziell ausgeprägtes Verfahren zur Kalkulation solcher Unsicherheiten soll hier die sog. "Fuzzy Logic" [19] erwähnt werden. Im allgemeinen schließt sich der Verfasser zu diesem Thema der Aussage von Patrick Henry Winston [13] an: "The treatment of uncertainty in expert systems ranges from awfull to questionable".

2.6 Methoden der Wissensakquisition

Der automatische Erwerb von Expertenwissen zählt zu den schwierigsten und aufwendigsten Gebieten der KI-Forschung. Die meisten in der Praxis eingesetzten und kommerziell vertriebenen Systeme beschränken sich auf mehr oder weniger komfortable Editoren und Graphikhilfsmittel um das Expertenwissen interaktiv in das System zu integrieren. So gibt es je nach Wissensrepräsentationsform Regeleditoren, Frameeditoren und sog. "Browser". Mit deren Hilfe können die Objekte und Strukturen der Wissensbasis in Form eines Baumes am Bildschirm graphisch dargestellt werden. Die Interaktion und Modifikation erfolgt dann direkt im Bezug auf diese graphische Darstellung mit der Maus.

2.6.1 Automatischer Wissenserwerb

Zu dieser Fragestellung liegen nur sehr wenige Realisierungen vor. Nach RAULEFS [22] werden dabei zwei Ansätze verfolgt. Einmal ist dies die Synthese neuer Wissensinhalte durch Verknüpfen, Einordnen und Verallgemeinern bestehender Wissenskomponenten. Zum anderen wird ein rechnergestützter Dialog mit dem Bereichsex-

perten geführt. Dabei generiert das System neue Wissensselemente und fügt sie in die Wissensbasis ein. Anschließend bearbeitet der Problemlöser die so erneuerte Wissensbasis bereichsweise und zeigt die Ergebnisse an. Der Bereichsexperte entscheidet dann, ob die neuen Lösungen gut oder schlecht brauchbar sind. Je nach seiner Entscheidung werden die neu zugefügten Wissensinhalte entweder beibehalten oder wieder verworfen. Neuere, teilweise sehr theoretische Ansätze zum Wissenserwerb und automatischen Lernen beschreiben FOHMANN [23] und MITCHELL [24].

Im Rahmen eines BMFT-Verbundprojektes (LERNER) entstand desweiteren das System BLIP [25]. BLIP ist ein Werkzeug zur Unterstützung der Wissensakquisition beim Erstellen von Expertensystemen. Erarbeitet wurde hier ein maschinelles Lernverfahren, das vornehmlich auf der Verwendung von Metawissen basiert. Unter Metawissen versteht man allgemein das Wissen über das Wissen und hier speziell die Kenntnis, wie man neues Wissen erwirbt. Durch eine induktive Lernkomponente sollen aus Fakten Regeln gewonnen werden, die zur Strukturierung des Sachgebietes beitragen. Diese Strukturierung soll unabhängig von der Problemstellung und Lösung sein. Die Definition des Metawissens wird vom Systementwickler vorgenommen.

2.6.2 Konsistenzerhaltung

Genauso schwierig und unerschöpflich wie die Frage des automatischen Wissenserwerbs ist das Problem der Konsistenzerhaltung. Hier existiert natürlicherweise ein direkter Bezug zu der eingesetzten Form der Wissensrepräsentation. Anhand von Produktionsregeln als Repräsentationsform sollen nachfolgend einige Ursachen für Inkonsistenz erläutert werden. Als wichtigste Ursachen findet man in der Literatur [26][27][28]:

- Redundanz,
- Konflikt,
- Subsumption,
- Unvollständigkeit,
- Zirkuläres Schließen.

Unter Redundanz versteht man das mehrfache Vorhandensein gleicher Wissens-elemente. So können dies zwei Regeln sein, die unter den gleichen Bedingungen die gleichen Aktionen veranlassen.

(Regel 1): Wenn A und B
 Dann C

(Regel 2): Wenn B und A
 Dann C

Ein Konflikt entsteht, wenn zwei Regeln in ihrem Bedingungsteil identisch sind, in ihrem Aktionsteil aber verschiedene Aktionen veranlassen, die im Widerspruch zueinander stehen.

(Regel 1): Wenn A
 Dann setze Slot B auf den Wert 5

(Regel 2): Wenn A
 Dann setze Slot B auf den Wert 6

Eine Subsumption liegt vor, wenn zwei Regeln in ihrem Aktions-teil gleich sind, die Bedingungen der ersten aber eine Untermenge der Bedingungen der zweiten Regel bildet. Beide Bedingungen sind dabei erfüllt. Wenn die Regel mit den größeren Einschränkungen ausgeführt werden kann, ist dies auch für die weniger eingeschränkte Regel möglich.

Beispiel:

(Regel 1): Wenn A und B
 Dann C

(Regel 2): Wenn A
 Dann C

Kritisch wird es hier, wenn sich die Aktionsteile der Regeln unterscheiden, ferner die Bedingungen A und B erfüllt sind und die Regel 2 vor der Regel 1 bearbeitet wird. Dann sind beide Regeln

ausführbar und es kann passieren, daß mit Regel 2 die falsche ausgeführt wird. Über spezielle Kontrollstrategien muß dann sichergestellt werden, daß genau die Regel ausgeführt wird, die auch ausgeführt werden soll.

Eine Wissensbasis ist unvollständig, wenn notwendige Elemente fehlen. In den Bedingungssteilen einer abgeschlossenen Gruppe von Produktionsregeln kann beispielsweise eine bestimmte Zahl von Variablen vorkommen, für die es eine endliche Zahl von Kombinationsmöglichkeiten gibt. Es liegt aber nicht für jede dieser Möglichkeiten eine Produktionsregel vor. Es können also Zustände auftreten, für die keine Regel existiert. Diese Situation muß trotzdem durch geeignete Maßnahmen kontrolliert werden.

(Regel 1): Wenn $a > 5$
 Dann C

(Regel 2): Wenn $a < 5$
 Dann D

(Zusatzregel): Wenn $a = 5$
 Dann E

Beim zirkulären Schließen besteht die Gefahr, daß der Regelinterpret in einer Endlosschleife läuft und keine brauchbaren Ergebnisse liefern kann.

(Regel 1): Wenn A Dann B

(Regel 2): Wenn B Dann C

(Regel 3): Wenn C Dann A

Die Existenz solcher Zyklen ist relativ einfach zu erkennen. Dagegen ist es schwierig festzustellen, welche der Regeln falsch ist und ausgetauscht werden muß.

2.7 Programmierstile

Zum weiteren Verständnis der Methoden und Hilfsmittel zur Wissensverarbeitung sollen an dieser Stelle einige Ausführungen über verschiedene Maschinenmodelle und Programmierstile dienen. Nach STOYAN [29] unterscheiden wir zwischen dem

- konventionellen,
- funktionalen und
- objektorientierten

Maschinenmodell. Das konventionelle Maschinenmodell, die sog. von-Neumann-Architektur, geht von schrittweise ablaufenden Folgen von Aktionen bzw. Operationen aus, die über Daten ausgeführt werden. Danach stellen die Programme sequentielle Vorschriften für solche Aktionsfolgen dar. Diese Architektur wird durch die konventionellen CPU-Speicher-Kombinationen realisiert. Bei den ausschließlich für dieses Maschinenmodell konzipierten imperativen Programmiersprachen (FORTRAN, PASCAL) scheinen die Konzepte der Maschine durch: Variablen sind Namen für Speicherzellen oder Gruppen von Speicherzellen (Arrays, Records). Ausdrücke sind Berechnungen auf Speicherinhalten und Wertzuweisungen sind Speicheroperationen, die den Zustand der Maschine verändern. Möglichkeiten zur Abstraktion sind eventuell vorhandene höhere Kontrollstrukturen, Blockstruktur, Prozeduren (Unterprogramme) und abstrakte Datentypen.

Der auf dieses Maschinenmodell abgestimmte Programmierstil benötigt nach STOYAN [29]:

- Sprachelemente für die auszuführenden Grundoperationen,
- Elemente zum Aufbau von Datenstrukturen,
- Mittel zur Kombination und Ablaufplanung von Grundoperationen,
- Mittel zur Unterprogrammtechnik.

Das funktionale Programmkonzept beruht auf dem mathematischen Funktionsbegriff. Aus gewissen Grundfunktionen, also Funktionen,

die nicht zerlegbar oder mit Hilfe anderer Funktionen nicht definierbar sind, können neue, höhere Funktionen definiert werden. Das gesamte Programm repräsentiert so eine große Gesamtfunktion, realisiert durch rekursive und geschachtelte Anwendung von Grundfunktionen und selbsterzeugten Funktionen, die Abbildungen vom Definitionsbereich der Eingangsdaten in den Wertebereich der Ausgangsdaten bewirken. Der Wert der Funktion für ein bestimmtes Element des Definitionsbereiches wird durch Anwenden der Funktion auf dieses Element ermittelt.

Beim objektorientierten Maschinenmodell stellt man sich innerhalb der Rechenanlage eine Welt von Objekten vor, die miteinander in Verbindung stehen und kommunizieren. Die Kommunikation erfolgt durch Senden und Empfangen von Nachrichten. Die Nachrichten selbst stellen wieder spezielle Objekte dar. Das Objekt, das die Nachricht sendet, muß den Empfänger kennen und übergibt ihm in dieser Nachricht Angaben über die zu leistende Aufgabe und die entsprechenden Bezugsobjekte, die zur Erfüllung der Aufgabe benötigt werden. Im allgemeinen soll das empfangende Objekt mit diesen Argumenten Operationen durchführen. Der im objektorientierten Stil arbeitende Programmierer benötigt daher

- Elemente zur Repräsentation von Objekten,
- Mittel zur Benennung, Kombination und Klassifikation von Objekten und
- Mittel zur Beschreibung und Benennung von Nachrichten.

2.8 Softwarewerkzeuge zur Wissensverarbeitung

Im Laufe ihrer Entwicklungsgeschichte hat die KI-Forschung bereits eine Vielzahl von Softwarewerkzeugen hervorgebracht, die man einmal hinsichtlich ihrer Funktionalität und dann von ihren Einsatzmöglichkeiten her unterscheiden kann (Bild 7). Basis und Ausgangspunkt dieser Entwicklungen sind die eigentlichen Programmiersprachen. Hier sind in erster Linie die Sprachen LISP und PROLOG zu nennen. Beide Sprachen sind so universell einsetzbar wie konventionelle Programmiersprachen auch. Der Preis, den man

für diese Universalität zahlen muß, ist mangelnder Komfort. Sämtliche KI-Methoden wie Suchverfahren, vorwärts- und rückwärtsverkettende Kontrollstrukturen (Kap. 2.5) müssen selbst implementiert werden. PROLOG hebt sich hierbei etwas ab, da es über implizite Inferenzmechanismen auf der Basis der Prädikatenlogik verfügt [16].

Aufbauend auf diesen Programmiersprachen wurde inzwischen eine Vielzahl allgemein einsetzbarer Werkzeugsysteme entwickelt. Man spricht deshalb von Werkzeugsystemen, weil bereits verschiedene Wissensrepräsentationsformen, Kontrollstrategien und Inferenzmechanismen in den Systemen integriert sind. Dem Entwickler eines Expertensystems kommt dann "nur" noch die Aufgabe zu, das spezifische Problemwissen zu erwerben, zu strukturieren und in dem System zu implementieren. Dazu stehen dann unterschiedlich komfortable Hilfsmittel zur graphischen Wissensdarstellung (Browser) zum Editieren und zur Fehlersuche (Debugger) zur Verfügung. Mit zunehmender Spezialisierung dieser Systeme werden natürlich auch die möglichen Einsatzgebiete mehr und mehr begrenzt.

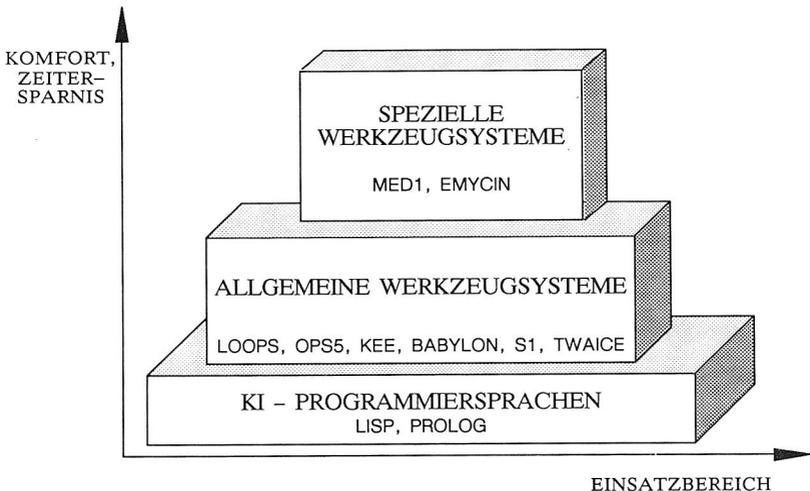


Bild 7: Klassifikation von Softwarewerkzeugen

Die dritte Kategorie von Werkzeugsystemen umfaßt jene, die für den Einsatz in fest vorgegebenen und eng umgrenzten Problemfeldern gedacht sind. Der hohe Grad der Spezialisierung ermöglicht es hier am schnellsten zu Lösungen zu kommen. Jedoch sind die möglichen Einsatzgebiete auch am stärksten begrenzt, sodaß sehr sorgfältig geprüft werden muß, ob sich das Werkzeug tatsächlich für die Anwendung eignet.

Beispiele für solche Systeme sind MED1 [30] und EMYCIN [21]. Die zu lösenden Aufgaben liegen im Bereich der Diagnose medizinischer Probleme. Mit MED1 werden neuerdings auch Diagnosefragen bei technischen Systemen angegangen.

Zum breiteren Verständnis soll in den folgenden zwei Kapiteln etwas näher auf LISP und einige Werkzeugsysteme eingegangen werden.

2.8.1 Die Sprache LISP

Die Entwicklung von LISP geht auf Arbeiten von McCarthy 1958 zurück und hat zum Ziel, eine höhere Programmiersprache zur Manipulation symbolischer Objekte zu schaffen. Im Laufe der Entwicklung entstanden eine Vielzahl von sog. LISP-Dialekten. Einige Beispiele hierfür sind ZETA-LISP für SYMBOLICS-Computer, INTER-LISP für XEROX und SIEMENS-Computer und PORTABLE STANDARD LISP (PSL) für DOMAIN-APOLLO-Computer. Aus Standardisierungsbestrebungen heraus wurde inzwischen COMMON-LISP [31] als Basis weiterer LISP-Aktivitäten vorgeschlagen. COMMON-LISP ist inzwischen auf DEC-VAX- und DOMAIN-APOLLO-Rechnern verfügbar.

LISP steht für "LIST-Prozessor" und damit wird auch schon das Charakteristikum dieser Sprache angedeutet. Daten, Informationen und Wissen werden in Form von Listen repräsentiert. Listen können dann auf vielfältige Art und Weise manipuliert und ausgewertet werden.

LISP ermöglicht nach STOYAN [29] die Verarbeitung hochstrukturi-

rierter Daten. Wenn bei Problemstellungen nicht bekannt ist, wie die Menge und Struktur der heranzuziehenden Daten eskaliert, wird mit herkömmlichen Programmiersprachen irgendwann einmal der Rahmen der eindeutigen Speicherzuordnung gesprengt. LISP bietet dagegen eine automatische dynamische Speicherverwaltung. Aus diesem Grund kann der in LISP vorherrschende Datentyp, die Liste, eine unbestimmte Länge und Tiefe besitzen.

In LISP werden Programme durch Datenstrukturen repräsentiert. Es gibt keinen Unterschied zwischen dem, was man bei herkömmlicher Programmierung als Daten und Programme bezeichnet. Beides sind symbolische Ausdrücke (s-expressions). Insofern kann LISP, ähnlich einer Maschinensprache und im Gegensatz zu anderen höheren Programmiersprachen, Programme erzeugen und im gleichen Verarbeitungsschritt zur Ausführung bringen. Eine typische Anwendung hierfür sind selbstoptimierende und selbstlernende Programme. LISP ist eine funktionale Programmiersprache. Ein Problem wird in kleine Funktionseinheiten zerlegt und diese dann vom Programmierer in definierte Funktionen umgesetzt, die auch rekursiv sein können.

Durch Beschränkung auf Teilbereiche von LISP kann der Programmierer moderne Programmierstile verwirklichen. LISP war Wegbereiter für den funktionalen Programmierstil. Aber auch der objektorientierte Programmierstil wird durch LISP unterstützt.

LISP-Implementationen sind nicht ausschließlich auf Compiler gestützt, mit denen Maschinenprogramme erzeugt werden. Sie sind vornehmlich als interpretierende Verarbeitungssysteme organisiert. Diese Systeme können allerdings auch übersetzte Funktionen abarbeiten. In LISP-Systemen werden die Aufgaben von Editoren, Syntaxprüfern, Testsystemen, Programmverwaltungssystemen, Hilfsmitteln zur Fehlersuche und Compilern integriert wahrgenommen.

2.8.2 Werkzeugsysteme

Softwaresysteme, die die Erstellung von Expertensystemen unterstützen und dabei Möglichkeiten zur Wissensakquisition, Wissensrepräsentation und Wissensmanipulation bieten, werden allgemein als Rahmensysteme (engl. Shell) bezeichnet. Die bekanntesten und auch kommerziell verfügbaren sind KEE [32], SRL+, S.1 [33], ART, DUCK, TWAICE [34], OPS5 [35] und LOOPS [36]. Eine Beschreibung und Gegenüberstellung dazu findet sich bei HAYES-ROTH [37] und ROBINSON [38].

In den Kap. 2.4 und 2.5 wurden einige wesentliche Konzepte der Wissensrepräsentation und Manipulation erläutert. Aufgrund der signifikanten Schwächen einzelner Repräsentationsformen im Hinblick auf allgemeingültige Ansätze wird seit geraumer Zeit die Strategie der Integration multipler Paradigmen verfolgt. Diese Systeme, deren Architektur also auf verschiedenen Formalismen beruht, werden als "hybride" bezeichnet. Theoretisch betrachtet, sind hybride Systeme hinsichtlich der Qualität der Repräsentation nicht besser als uniforme Systeme. Aufgrund der Existenz verschiedener Darstellungs- und Manipulationsmöglichkeiten ist man allerdings in der Lage, die Problemlösungsprozesse genauer und damit besser an die vorgegebenen Aufgaben anzupassen.

Als ein Vertreter von hybriden Systemen soll im folgenden das System BABYLON [39] seiner Anschaulichkeit wegen etwas genauer beschrieben werden. BABYLON ist ein Werkzeugsystem für den Entwurf, die Implementierung und den Betrieb von Expertensystemen und wurde von der Forschungsgruppe "Expertensysteme" bei der Gesellschaft für Mathematik und Datenverarbeitung entwickelt.

Das System besticht sowohl durch seine modulare Architektur, als auch durch die komfortable Arbeitsumgebung. Es verfügt über die folgenden Repräsentationsmöglichkeiten:

- Produktionsregeln,
- Objektorientierte Darstellung,
- Prädikatenlogik (Horn-Klauseln).

Für jeden Formalismus gibt es einen eigenen Sprachprozessor. Der Frame-Prozessor interpretiert die Objektkonstrukte, der Logik-Prozessor die Horn-Klauseln und der Regelinterpretierer die Produktionsregeln. Die Koordination der einzelnen Prozessoren übernimmt ein Metaprozessor. Er verwaltet die zwischen den einzelnen Prozessoren gesetzten Referenzen und Mitteilungen. Dazu verfügt er über eine eigene Wissensbasis, die mit dem Horn-Klausel-Formalismus aufgebaut ist. In dieser Wissensbasis stehen Informationen über die Basisprozessoren, was sie tun können und wie sie zu aktivieren und zu deaktivieren sind.

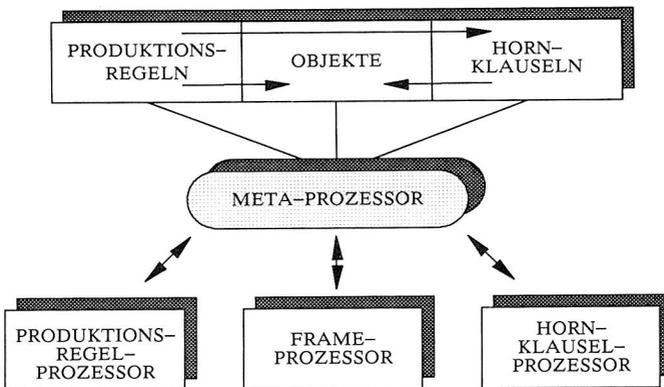


Bild 8: Architektur von BABYLON. Die Referenzen zwischen den Formalismen sind durch Pfeile angedeutet [39].

Die Integration der Formalismen wird dadurch erreicht, daß Prämissen (Bedingungen) von Produktionsregeln Referenzen auf Objektzustände (Frames) und Relationsausdrücke (Horn-Klauseln) sein können. Auf der anderen Seite können aber auch Horn-Klauseln auf Objekte referieren, sodaß sowohl bei der Auswertung von Produktionsregeln als auch von Horn-Klauseln die Vererbungsmechanismen des Frame-Prozessors verfügbar sind. Der Meta-Prozessor erkennt aufgrund der syntaktischen Strukturen die Referenzen auf die Basisprozessoren und delegiert dann die Bearbeitung.

Die Architektur ist streng hierarchisch. Die Basisprozessoren haben keine Kenntnis voneinander. Sie sind allerdings in der Lage zu erkennen, wann Aufgaben außerhalb ihres Gültigkeitsbereiches liegen und wenden sich dann an den Meta-Prozessor. Der Vorteil des Systems besteht in seiner Offenheit in der Breite und auch in der Tiefe. In der Breite bedeutet, daß man zusätzliche Prozessoren hinzufügen oder auch entfernen kann. Die Offenheit in die Tiefe wird über einen objektorientierten Programmierstil erreicht (message passing).

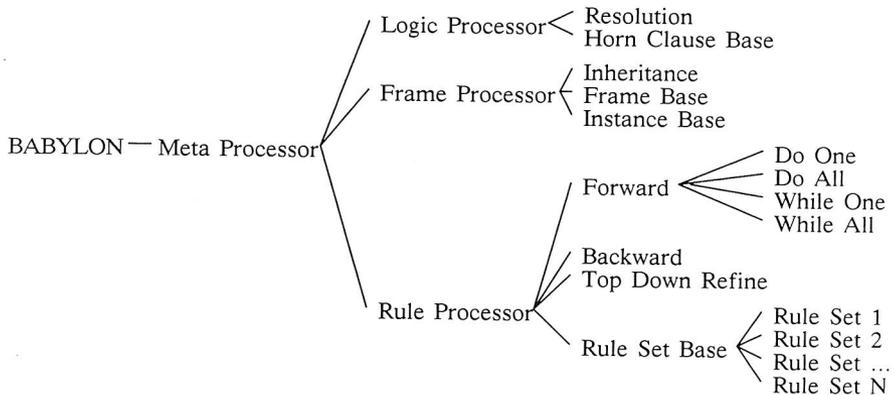


Bild 9: Die hierarchische Struktur von BABYLON

Die Architektur von Babylon könnte sich als nützlich bei einer Kopplung von Expertensystemen mit anderen Informationssystemen erweisen. Sinnvoll wäre hier eine Kopplung mit Datenbanksystemen, Simulations- oder auch CAD/CAM-Systemen. Gedacht wird an eine abstrakte Schnittstelle, über die der Meta-Prozessor Aufgaben arbeitsteilig an Fremdsysteme delegieren könnte.

3. Einsatzgebiete wissensbasierter Systeme in der Produktions- technik

In der Einleitung der vorliegenden Arbeit wurde bereits ein Überblick zu gegenwärtigen und zukünftigen Anwendungsfeldern von KI-Methoden und wissensbasierten Systemen gegeben. Dieses Kapitel beschäftigt sich nun schwerpunktmäßig mit realisierten Lösungen im Umfeld der technischen Produktion. Im Anschluß daran erfolgt ein Überblick zu den Einsatzkriterien und Entwicklungsprozessen von Expertensystemen.

Verständlicherweise kann ein Überblick über eingesetzte Systeme an dieser Stelle nicht vollständig sein. Es werden vielmehr exemplarisch vier Anwendungsfelder herausgegriffen und diskutiert:

- Diagnose technischer Systeme,
- Planung von Fertigungsabläufen,
- Konstruktion von technischen Produkten,
- Simulation von Fertigungsprozessen.

Nicht ausführlich besprochen aber zumindest erwähnt werden soll an dieser Stelle auch McDermotts "Urvater" aller Konfigurationssysteme XCON [40]. XCON ist ein Expertensystem, das Computer der Firma DIGITAL in Abhängigkeit von Kundenspezifikationen aus einer Vielzahl von Einzelkomponenten konfiguriert. XCON zählt zu den wirtschaftlich erfolgreichen Expertensystemen. Nach Angaben von DIGITAL arbeitet es zu 99 Prozent korrekt und spart der Firma jährlich 18 Millionen US-Dollar ein.

Weitere Veröffentlichungen über Expertensystemansätze in den Bereichen

- Produktionsplanung und Steuerung [41],
- Fertigungssteuerung [42][43] sowie
- Werkstattsteuerung und Simulation [44]

unterstreichen das große Innovationspotential wissensbasierter Systeme in der Produktionstechnik.

3.1 Diagnose technischer Systeme

Diagnosesysteme gehören zu den sogenannten Interpretationssystemen. Es handelt sich hierbei um Programme, die Daten über den aktuellen Zustand einer technischen Einrichtung interpretieren, um daraus Aussagen über vergangene, zukünftige und gegenwärtige Zustände herzuleiten. Neben der Diagnose, zu der es gerade im medizinischen Bereich viele Expertensystementwicklungen gibt, findet man solche Interpretationssysteme auch bei Aufgaben der Datenanalyse oder der Vorhersage von Ereignissen. Durch einfache Kontrollstrukturen in Form von Regeln sowie FORWARD- und BACKWARDCHAINING Mechanismen hat man gerade in diesem Bereich wesentlich mehr Fortschritte erreicht als in anderen Anwendungsgebieten. Beispiele für erfolgreiche Implementierungen sind die Systeme:

- ACE (Automated Cable Expertise, BELL Laboratories)
zur Wartung und Fehlersuche bei Telefonleitungen [45],
- FAULTFINDER (NIXDORF Computer AG)
zur Fehlersuche und Erstellung von Reparaturanleitungen
bei EDV-Geräten [46],
- DEX.C3 (FORD Deutschland)
zur Fehlerdiagnose in automatischen PKW-Getrieben [47].

Mit dem System MED1 existiert auch schon ein spezielles Werkzeugsystem, welches für Anwendungen im Bereich der Motordiagnose und Fehlersuche bei EDV-Anlagen eingesetzt wird [30]. MED1 löst diagnostische Probleme mit Hilfe einer HYPOTHEZISE-AND-TEST-Strategie (Bild 10). Das heißt, eine Hypothese zu einem Problem wird anhand vorgegebener Randbedingungen aufgestellt und anschließend im Zuge einer Verdachtsüberprüfung evaluiert.

Die Generierung von Verdachtshypothesen erfolgt mittels vorwärtsverkettender Regeln, die sich in ihrem Bedingungsteil auf verschiedene Randbedingungen beziehen. Diese Randbedingungen gibt der Benutzer während der Initialisierungsphase des Systems ein. Bei der Verdachtsüberprüfung werden die zu einer Hypothese gehörenden Symptome über rückwärtsverkettende Regeln hergeleitet und

vom Benutzer dann interaktiv bestätigt oder auch verworfen. Die neu erfaßten Symptome können dann über vorwärtsverkettende Regeln wieder neue Verdachtshypothesen generieren. In geeigneten Intervallen wird eine Zwischenbilanz gezogen und eine Übersichtsagenda erstellt bzw. aktualisiert. In dieser Agenda sind die einzelnen

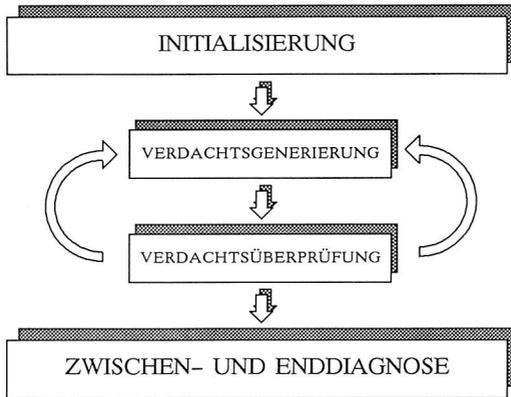


Bild 10: HYPOTHEZISE-AND-TEST-Strategie

Hypothesen entsprechend der Stärke des Verdachts geordnet. Die Verdachtsbewertung erfolgt mit Hilfe eines Punktesystems, wobei die eigentliche Punktevergabe über die Produktionsregeln gesteuert wird. Der ganze Zyklus wird beendet, wenn eine Enddiagnose gefunden wurde, keine noch zu überprüfende Hypothese mehr vorliegt oder der Benutzer den Vorgang interaktiv abbricht.

Einen interessanten Ansatz aus dem Bereich der Montagetechnik stellt das Projekt ROCEDI dar [48] [49]. ROCEDI steht für Robot Cell Diagnosis und dient zur wissensbasierten Diagnose von automatisierten Montagezellen mit Industrierobotern. Die Aufgabe des Systems ist es, alle Fehler zu finden, die bei einem Gerät der Montagezelle eine Störung verursacht haben. Ferner soll das Geräteteil bestimmt werden, welches zu reparieren oder zu ersetzen ist.

ROCEDI ist ein offline-Diagnosesystem. Das heißt, alle zustandsbeschreibenden Daten müssen von einem Bediener eingegeben werden. In ROCEDI werden die Diagnosefunktionen in Abhängigkeit von ihrer Komplexität von zwei verschiedenen Komponenten durchgeführt. Der erste Modul STEP1 ist als regelbasiertes System realisiert und soll den Fehler so eingrenzen, daß speziell abgestimmte Diagnosemethoden zur Störungsbeseitigung angestoßen werden können. Der zweite Modul CELLEX findet Fehler durch Überprüfung der Programm- und Signallogik aller Komponenten der Montagezelle. Für die Zukunft ist eine online-Kopplung zur Übertragung der Sensordaten über eine RS232-Schnittstelle oder ETHERNET geplant.

Das System ist auch deshalb interessant, weil hier größtenteils Standardwerkzeuge zur Wissensverarbeitung eingesetzt werden. Es wurde auf einer Workstation XEROX 1109 mit Hilfe der objektorientierten Wissensrepräsentationssprache LOOPS implementiert.

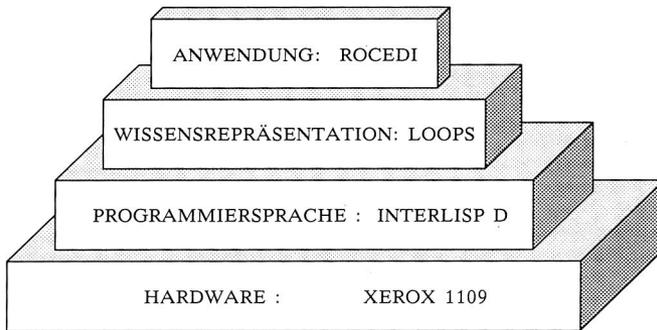


Bild 11: Systemarchitektur von ROCEDI

Wie bei fast allen Expertensystemansätzen ist eine zur Problemstellung passende Wissensrepräsentation ausschlaggebend für Funktion und Effizienz. Alles Wissen wird in LOOPS in Form von Objekten behandelt. Objekte können Ausprägungen der Klassen REGELN, HYPOTHESEN, FRAGEN und BEARBEITUNG sein.

Objekte haben außer ihrem Namen noch vier Instanzvariablen, die angeben,

- welches die nachfolgenden Elemente der Wissensbasis sind,
- ob das Objekt in der Sitzung schon einmal angesprochen wurde,
- bei welchen Regeln das Objekt im Bedingungsteil aufgeführt ist und
- bei welchen Regeln das Objekt im Aktionsteil aufgeführt ist.

Auf diese Weise läßt sich die Wissensbasis geeignet strukturieren, sodaß effiziente Zuordnungen von Regeln und Fragen möglich sind, die den Suchaufwand reduzieren.

Die Kontrollstrategie des Systems ähnelt der von MED1. Mit Hilfe vorwärtsverkettender Regeln werden Diagnosehypothesen generiert und in Abhängigkeit der vorgegebenen Randbedingungen mit Wahrscheinlichkeitsfaktoren belegt. Diese Faktoren bewegen sich auf einer Skala von -1 bis +1. Alle Hypothesen werden in eine Liste eingetragen und in Abhängigkeit von ihrem Wahrscheinlichkeitsgrad abgearbeitet. Der Verdachtsgenerierung schließt sich die Verdachtsüberprüfung mit Hilfe rückwärtsverkettender Regeln an. In beiden Fällen kann das System Fragen an den Benutzer stellen und entsprechende Handlungen veranlassen. Der Benutzer kann aber auch unaufgefordert neue Informationen eingeben, die dann in die Wissensbasis integriert werden. Zur Erklärung des Problemlösungsprozesses gibt das System Auskunft über die Gründe seines Vorgehens und über die dabei beteiligten Objekte.

3.2 Planung von Fertigungsabläufen

Das Planen von Bearbeitungsverfahren und Fertigungsprozessen wird allgemein als wichtiges Innovationspotential für die nächsten Jahre gehalten. Wissensbasierte Ansätze befinden sich hier generell noch im Prototypenstadium. Einige interessante Arbeiten sollen nachfolgend beschrieben werden.

Battelle Laboratories [50] entwickelte einige Systeme für Aufgaben im Bereich der Umformtechnik. Ausgehend von der gewünschten Fertigteilkontur werden dabei die Rohteilkontur und danach auch das Schmiedewerkzeug bestimmt. Ein anderer Prototyp ermittelt bei Tiefziehoperationen die einzelnen Teilschritte, die, vom Rohteil ausgehend, zum fertig gezogenen Werkstück führen.

An der Universität von Linköping entstanden Arbeiten, die sich mit Planungssystemen für das spanende Fertigungsverfahren DREHEN beschäftigen [51]. Die Aufgabenstellung umfaßt hier die Teilschritte

- Auswahl der Bearbeitungsoperationen,
- Auswahl der Spannmittel und Spanungsflächen,
- Festlegung der Bearbeitungsreihenfolge,
- Auswahl der Werkzeugtypen,
- Bestimmung der Hauptabmessungen der Werkzeuge,
- Auswahl spezieller Werkzeuge gemäß Typ und Abmessung,
- Zuordnung der Werkzeuge zu den Bearbeitungsoperationen,
- Anordnen der Werkzeuge in der Maschine,
- Bestimmung der Schnittwerte und
- Bestimmung der Werkzeugwege.

Das Planungsziel ist die Erstellung eines NC-Teileprogramms und eines Werkzeugplanes. Hierbei konzentriert sich die Arbeit auf die Evaluation geeigneter Suchverfahren zur Ablaufplanung.

Ähnliche Arbeiten wurden an der Universität von Grenoble durchgeführt [52]. Das dabei entstandene System GARI dient zur Ablaufplanung von Zerspanungsvorgängen mechanischer Teile. Das System besteht aus einer speziellen Wissensbasis aus Fertigungsregeln und einem Planungsmodul. Die Expertise des menschlichen Experten wird in diesen Fertigungsregeln repräsentiert. Die Regeln sind mit Gewichtungsfaktoren versehen, die die Bedeutung und die Sicherheit der Regel zum Ausdruck bringen sollen. Aufgrund der Komplexität des Problemkreises wurde auf eine ausgeklügelte Kontrollstruktur Wert gelegt. Sie umfaßt:

- Generierung von Hypothesen,
- Deduktion von Fakten,
- Auflösung von Konflikten,
- Fortpflanzung von Randbedingungen (constraint propagation).

Der Planungsmodul verwendet ein Modell des zu fertigenden Teiles. Dieses Modell muß sowohl geometrische als auch technische Angaben enthalten und beschreibt das Teil mit Hilfe verschiedener Merkmale wie Bohrungen, Einkerbungen, Nuten und Spanungsflächen. Ausgehend vom Modell wird zunächst ein allgemein gehaltener Bearbeitungsplan generiert und vorweg die Menge der potentiellen Spannungsvorgänge bestimmt. Den endgültigen Plan erhält man durch eine schrittweise Verfeinerung bzw. iterative Einschränkung und Spezifikation des Anfangplanes.

In GARI wird ein Werkstück über verschiedene Typen von Merkmalen wie beispielsweise SENKLÖCHER, BOHRUNGEN, EINKERBUNGEN, NUTEN und SPANUNGSFLÄCHEN beschrieben. Die Notation von Merkmalen zur Beschreibung der Werkstücke ist ein wichtiges Konzept in GARI. Die Fertigungsregeln machen nämlich Gebrauch von dem semantischen Gehalt, der den Merkmalen zugeordnet ist.

- Phase 1	Machine: FU203 Operation 1.1 Operation 1.2	resting face: FXM roughing cut of FXP finishing cut of N1
- Phase 2	Machine: FU203 Operation 2.1	resting face: FXP finishing cut of FZP
- Phase 3	Machine: FU203 Operation 3.1 Operation 3.2 Operation 3.3	resting face: FZP finishing cut of FZM finishing cut of FXP finishing cut of N2
- Phase 4	Machine: GSP205 Operation 4.1	resting face: FZP finishing cut of H2
- Phase 5	Machine: GSP205 Operation 5.1	resting face: FZM finishing cut of H1

Bild 12: Mit GARI erzeugter Fertigungsplan

Die Beschreibung der verfügbaren Maschinen erfolgt über Namen und Eigenschaften, die der Benutzer für wichtig erachtet. Eigenschaften können dabei MASCHINENTYP, PRÄZISION oder DIMENSIONS-ATTRIBUTE wie beispielsweise der Achsabstand sein. Jedes Merkmal des Werkstücks benötigt maximal zwei Schnittvorgänge, einen Schruppschnitt und einen Schlichtschnitt. In den erzeugten Plänen werden die Schnittvorgänge nach Operationen und Phasen gruppiert (Bild 12). Mehrere Schnittvorgänge unter einer Aufspannung bilden eine Operation. Mehrere Operationen auf der gleichen Maschine bilden eine Phase.

Die Fertigungsregeln bestehen aus Produktionen mit einer linken Regelseite und einer rechten Regelseite:

Bedingung ---> pieces of advice

Die linke Regelseite besteht aus einer konjunktiven Menge von Bedingungen über

- das zu bearbeitende Teil,
- die verfügbaren Maschinen
- und/oder den vorgesehenen Bearbeitungsplan.

Die PIECES OF ADVICE sind schwache Einflußfaktoren (constraints), die sich auf den Fertigungsplan beziehen. Je nach ihrer Bedeutung werden sie mit Werten zwischen 1 und 10 gewichtet. Bei der Planerstellung kommt es häufig vor, daß sich die Ratschläge der rechten Regelseite widersprechen. In diesen Fällen entsteht ein Konflikt, der in Form eines Kompromisses gelöst werden muß. Für diese Konfliktauflösung verwendet das System die angesprochenen Gewichtungsfaktoren. Zur Verdeutlichung sei das folgende Regelbeispiel aufgeführt:

Regel: (> (quality &X) 6.3) --> (9 (not-roughing-cut &X))

Bedeutung: Wenn die Oberflächenqualität eines Objektes höher als 6,3 ist, dann wird mit Gewicht 9 geraten, einen Schruppschnitt zu vermeiden.

Die experimentellen Erfahrungen mit GARI sind durchaus positiv. So wurden Werkstücke mit bis zu hundert beschreibenden Einträgen getestet. Nahezu alle erzeugten Pläne waren akzeptabel. Für die Zukunft ist geplant, GARI mit einem geometrischen Modellierungssystem zu koppeln. Ferner sind Verbesserungen der Beschreibungssprache für die Objekte und die Fertigungsregeln vorgesehen.

In der Bundesrepublik sind vergleichbare Aktivitäten an der Universität Saarbrücken bekannt. Im Rahmen des Sonderforschungsbereichs 314 entstand dort das System FERPLAN [53] zur wissensbasierten Fertigungsplanung in Stanzereien. Bild 13 zeigt eine interessante Gegenüberstellung von wissensbasierten und konventionellen Fertigungsplanungssystemen. Dabei wurden die Systeme

	SYSTEME		
	AUTAP	GARI	FERPLAN
PLANUNGSFUNKTIONEN :			
ARBEITSVORGANGS- FOLGEERMITTLUNG	JA	JA	JA
MASCHINENAUSWAHL	JA	JA	JA
MASCHINENPLAN	NEIN	JA	NEIN
FERTIGUNGSHILFSMITTEL- ZUORDNUNG	JA	JA	JA
VORGABEZEITBESTIMMUNG	JA	NEIN	JA
ARBEITSPLANAUSGABE	JA	NEIN	JA
VERFAHREN :			
STANZEN	JA	NEIN	JA
LOCHEN	JA	NEIN	JA
SCHNEIDEN	JA	NEIN	JA
DREHEN	JA	JA	NEIN
BOHREN	JA	JA	NEIN
FRÄSEN	JA	JA	NEIN
SCHLEIFEN	JA	JA	NEIN
OBERFLÄCHENBEHANDLUNG	JA	JA	NEIN
WERKSTÜCKSPEKTRUM :			
ROTATIONSTEILE	JA	JA	NEIN
BLECHTEILE	JA	JA	JA
ART DES SYSTEMS :	WERKSTÜCKBEZOGEN		

Bild 13: Gegenüberstellung von Fertigungsplanungssystemen [53]

GARI und FERPLAN mit dem sicherlich noch ausgereifteren und auch komplexeren System AUTAP [54] verglichen. Dennoch dokumentiert diese Gegenüberstellung die grundsätzlichen Möglichkeiten wissensbasierter Ansätze.

3.3 Konstruktion von technischen Produkten

Entwicklung und Konstruktion sind diejenigen Unternehmensbereiche, bei denen die kreativen und schöpferischen Tätigkeiten von großer Bedeutung sind. Darüber hinaus ist zusätzlich noch ein hohes Maß an Erfahrungswissen notwendig, um die Realisierbarkeit und Wirtschaftlichkeit neuer Ideen abschätzen zu können. Die konstruktive Gestaltung eines Produktes legt zudem einen Großteil der Kosten für die Fertigung, Montage und Qualitätssicherung fest. Aus dieser Erkenntnis heraus wurden in der Vergangenheit auch schon vielfach Programmsysteme entwickelt, die die Montage- oder Fertigungsfreundlichkeit von Produktkonstruktionen beurteilen sollten. Keine dieser Entwicklungen hat aber den kommerziellen Durchbruch geschafft. Mit Hilfe der Technologie der Wissensverarbeitung eröffnen sich nun vielfältige neue Möglichkeiten.

Als ein Beispiel soll im folgenden ein System vorgestellt werden, das den Teilekonstrukteur bei der Beurteilung der Montagefreundlichkeit seiner Konstruktion berät [55]. Dabei wird untersucht,

- inwieweit das Teil für eine automatische Montage geeignet ist,
- welche Kosten für Handhabungs- und Montagegeräte dabei verursacht werden,
- welche Schwierigkeiten bei der Realisierung auftreten können
- und welche Änderungen an der Teilegestalt sinnvollerweise vorzunehmen sind.

Als Grundlage für diese Arbeiten konnte auf zwei Wissensquellen zurückgegriffen werden. Bei der ersten handelte es sich um den menschlichen Experten eines Unternehmens, das Ausrüstungsgüter für die automatisierte Montage herstellt und das System zur

Unterstützung von Beratungsleistungen für Kunden einsetzt. Die zweite Wissensquelle bestand aus Literatur zur montagegerechten Konstruktion.

Beim Systementwurf wurde versucht, den Problemlösungsprozeß des menschlichen Experten über mehrere Stufen hinweg nachzubilden. Das System unterscheidet vier verschiedene Fragestellungen:

- Möglichkeiten einer automatisierten Teilezuführung,
- Möglichkeiten der Teileorientierung,
- Möglichkeiten der Positionierung und des Transports von orientierten Teilen in die eigentlichen Montagezonen,
- Abschätzen der zu erwartenden Kosten.

Das System wurde in der Sprache PROLOG implementiert und besteht aus den drei Komponenten

- Wissensbasis,
- Regelinterpretier und
- CAD-Schnittstelle.

Die Kontrollstrategie läuft in der Form ab, daß der Regelinterpretier die Regeln der Wissensbasis rückwärtsverkettend abarbeitet und dabei, von bekannten Zielen ausgehend, neue Teilziele ermittelt. Diese werden dann in den folgenden Schlußfolgerungsprozessen weiter bearbeitet. Das System orientiert sich sehr stark an Sicherheitsfaktoren (certainty factors), die den Grad der Bestimmtheit und Glaubwürdigkeit einer Annahme ausdrücken sollen. Hierin liegt nach Meinung des Verfassers auch eine Schwäche des Systems begründet. Die Art und Weise, wie das System mit unsicherem Wissen umgeht, erscheint wenig plausibel. Bis heute existiert in der Wissenschaft jedenfalls noch kein allgemeiner Konsens über die Handhabung von vagen Wissensbereichen mit solchen Sicherheitsfaktoren.

Ein weiterer Schwachpunkt ist sicherlich auch in der CAD-Schnittstelle zu sehen. Es wird zwar klar erkannt, daß ein Konsultationssystem in der Konstruktion auf gestaltbestimmende Daten zu-

rückgreifen muß und daß dies eigentlich nur über ein CAD-System erfolgen kann. Doch benutzt das System eine eigene graphische Repräsentation der Teile in Form von zweidimensionalen Polygonen, deren Eckpunkte als signifikante Koordinatenpaare abgespeichert werden. Anhand einer Liste dieser Koordinatenpaare werden dann Symmetrieeigenschaften bestimmt, die bei der Abarbeitung der Regeln von Bedeutung sind. Der entscheidende Schwachpunkt liegt nun darin, daß komplexe symbolische Schlußfolgerungen auf der spärlichen graphischen Repräsentation nicht durchführbar sind.

Ganz anders ist dies bei einem weiteren Beispiel, dem System ICAD [56], gelöst. Hier wurde die völlige Integration von gestaltbestimmender Graphik und Wissensverarbeitung über einen objektorientierten Ansatz realisiert (vgl Kap. 2.7). Danach kennt das System nur Objekte, die Wissen darüber enthalten, wie sie auf Änderungen ihrer Umwelt zu reagieren haben. Mit anderen Objekten kommunizieren sie über Nachrichten (message passing), die den Adressaten und die auszuführenden Aktionen bestimmen. Mit Hilfe dieser Funktionen lassen sich Produktmodelle definieren, die in der Lage sind, auf der Basis von vorgegebenen Leistungskriterien ein direkt verwertbares Konstruktionsergebnis abzuleiten. Das System ICAD besteht aus den folgenden vier Hauptkomponenten:

- Symbolmanipulationssprache,
- Graphische Darstellungskomponente (Browser),
- Konstruktionswissensbasis,
- Relationale Abfragesprache.

Mit Hilfe der Symbolmanipulationssprache können mechanische Systeme in ihrer hierarchischen Struktur nachgebildet werden. Komplexe Planungsaufgaben lassen sich mit der Sprache in einem objektorientierten Programmierstil deklarieren und lösen. Die integrierte symbolische Referenzierung macht Definitionen lesbar und Variablennamen selbsterklärend.

Die graphische Darstellungskomponente (Browser) visualisiert die Struktur der Konstruktionswissensbasis in Form eines Baumes. Auf alle Objekte und Attribute kann dabei interaktiv zugegriffen

werden. Das erzeugte Bauteil wird am Bildschirm dreidimensional dargestellt.

Die Konstruktionswissensbasis besteht aus Regeln für die Konstruktionsexpertise. Ferner enthält sie Kataloge von Zukaufteilen, Firmenstandards sowie gesetzlichen und wirtschaftlichen Richtlinien und Bestimmungen. Darüber hinaus wird die gesamte Produkthierarchie in Form von Baugruppen, Unterbaugruppen und Einzelteilen repräsentiert.

Die relationale Abfragesprache erlaubt interaktive Abfragen über Tabellen, Kataloge und Objekte, wie sie in der Konstruktionswissensbasis gespeichert sind. Neben diesen interaktiven Funktionen sind auch programmierbare Abfragen zur Einbettung in ein entsprechendes Programmsystem möglich.

Das System ICAD vereinigt also drei wesentliche Softwaretechnologien in sich. Neben den Methoden der Künstlichen Intelligenz sind dies CAD-Verfahren und die Technik der relationalen Datenbanken. Erfolgreich eingesetzt wird das System bei der Projektierung von Wärmetauschern. Diese werden von den Kunden mit detaillierten Leistungsspezifikationen in Auftrag gegeben. Jeder der Wärmetauscher hat seine Besonderheiten und ein Auftrag besteht aus einer Vielzahl von Einzelteilen, für die über 450 Eingabespezifikationen vorhanden sind. Der Umfang der Regeln soll inzwischen mehrere Tausend erreicht haben.

Das System ICAD ist auch deshalb so außerordentlich interessant, weil es neue Wege der CAD/CAM-Technologie aufzeigt. Die Schwächen konventioneller CAD/CAM-Systeme liegen ja bekannterweise in der unzureichenden Semantik der Datenstruktur begründet. So sind Funktionselemente, wie Bohrungen, Fasen und Nuten, den heutigen Systemen größtenteils unbekannt, sofern man die entsprechenden Graphikelemente (Gerade, Kreis, Punkt) nicht zu einer Gruppe zusammengefasst hat.

Mit Hilfe des objektorientierten Ansatzes kann man derartige Funktionselemente einfach definieren und mit Semantik versehen.

Auf diese Weise werden solche Objekte neben einer graphischen Darstellung (Gerade, Kreis, Punkt) auch technologische Bedeutungen besitzen. Diese können dann von den nachgeschalteten

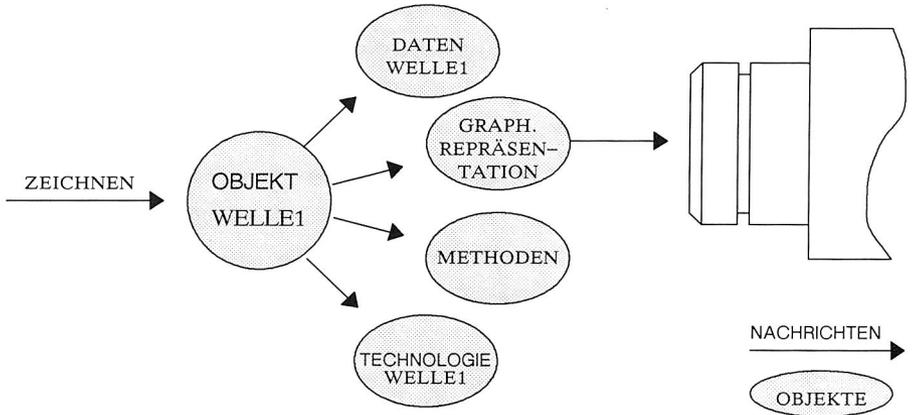


Bild 14: Objektorientiertes CAD-System

Prozessen (z.B.: NC-Programmierung) verwendet und ausgewertet werden. Erste Arbeiten in diese Richtung werden auch in der Bundesrepublik durchgeführt [57]. Auch der klassische Ansatz der Methodenbanken [58] läßt sich in einem objektorientierten Programmierstil verwirklichen.

3.4 Simulation von Fertigungsprozessen

Die Simulation von Vorgängen und Prozessen gewinnt in der Fertigungsplanung immer mehr an Bedeutung. Mit zunehmender Komplexität der Fertigungsanlagen und mit zunehmendem Kapitalaufwand wird nach verlässlichen Aussagen zu dem Einsatzverhalten derartiger Anlagen gesucht.

Die Wissensverarbeitung hat sich schon frühzeitig mit dieser

Thematik beschäftigt [59]. Heute werden mit den Systemen SIMKIT [60] der INTELLICORPORATION und CRL [61] der DANET GmbH auch zwei Systeme kommerziell vertrieben. Charakteristisch für alle wissensbasierten Simulationssysteme ist ein zugrunde liegendes Wissensrepräsentationswerkzeug, das die Modellierungsaufgaben der zu simulierenden Fertigungssysteme übernimmt. Das Simulationsmodell ist das rechnerinterne Abbild der real vorliegenden Fertigungseinrichtungen. Sämtliche Simulationsberechnungen laufen auf diesem Basismodell ab. Bild 15 stellt diesen Zusammenhang dar.

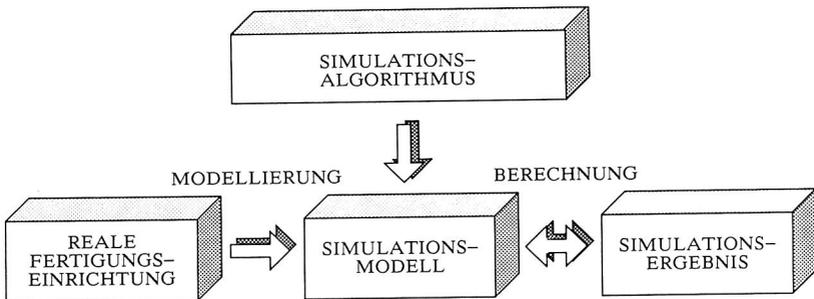


Bild 15: Modellierung und Simulation

Während die ersten Ansätze noch wenig Wert auf eine komfortable Benutzerschnittstelle legten, verfügen die heutigen Systeme über ausdrucksstarke bitmap-Graphik. So werden die Maschinen eines Fertigungssystems am Bildschirm nicht nur symbolisch, sondern auch in ihrer tatsächlichen Gestalt dargestellt. Beim Modellaufbau kann der Benutzer dann über die Maus die gewünschte Maschine aus einer Bibliothek auswählen. Andere Graphikhilfsmittel, sogenannte "pipes" und "valves", zeigen den Füllgrad von Speichern oder Puffern während des Simulationslaufes graphisch über pulsierende Farbsäulen an. Solche Funktionen sind natürlich mehr in den Bereich der Animation zu rechnen, doch waren die wissensbasierten Systeme die ersten, die solche Graphikmöglichkeiten boten.

Am Robotics Institut der Carnegie-Mellon University läuft schon seit 1980 das Projekt "Intelligent Management System" (IMS) [61]. Das langfristig angelegte Projekt ist wohl das bedeutendste seiner Art und hat zum Ziel, die Techniken der Künstlichen Intelligenz im Fertigungsbereich auf breiter Basis einzusetzen. Anwendungsfelder werden beispielsweise in der Fertigungssteuerung, der Prozeßdiagnose, der Fertigungsorganisation und der flexiblen Simulation gesehen.

Das Kernstück von IMS ist das Werkzeug zur Wissensrepräsentation und Manipulation CRL (Carnegie Representation Language). CRL ist ein hybrides Werkzeug und kann Produktionsregeln, Horn-Klauseln und Objekte interpretieren. Darüber hinaus verfügt es neben einem Editor und Windowmanager auch über eine Datenbankschnittstelle. Aufbauend auf diesem Werkzeug wurde nun das wissensbasierte Simulationssystem KBS (Knowledge Based Simulation) entwickelt [62].

In KBS werden Maschinen und Ereignisse in Form von Objektschemata bearbeitet, die untereinander über Relationen in Verbindung stehen. Ein Vererbungsmechanismus transferiert Objekteigenschaften entlang dieser Relationen. Regeln und Anweisungen sind eben-

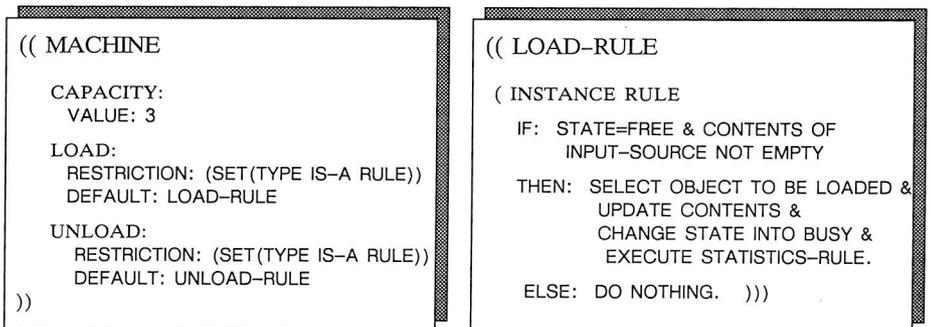


Bild 16: Maschinen- und Regelschema in KBS

falls solche Objektschemata. Sie gliedern sich nach dem Frame-Modell in Slots und Facetten. Beispiele für ein Maschinenschema und ein Regelschema zeigt Bild 16.

Die Simulation in KBS läuft ereignisorientiert ab. Ein Ereignis ist wiederum ein Objekt und wird durch seinen Namen, seine Aktivierungszeit, das Zielobjekt und einige spezifizierende Parameter gekennzeichnet (Bild 17). Den Ablauf der Simulation steuert ein sogenannter Kalender. In ihm sind alle Ereignisse eingetragen und werden nach und nach abgearbeitet. Während der Simulation steuert die Systemuhr die Aktivierung und Deaktivierung der Ereignisse. Der Zustand der Objekte ändert sich ständig.

```

(( EVENT23
  ( INSTANCE EVENT-NOTICE
    EVENT-TIME: 2.8
    EVENT-NAME: LOAD
    EVENT-FOCUS: M-NC-DRILL1
    EVENT-PARAMS: ORDER23 )))

```

Bild 17: Ein Ereignis als Objekt

Der Vorteil und die Eleganz des Systems besteht darin, daß interaktiv beliebige neue Schemata kreiert werden können. Bei konventionellen Programmsystemen ist man hierzu ja auf vordefinierte Klassen angewiesen. Auch sind dort die Simulationsalgorithmen starr programmiert und somit festgelegt. Bei diesem wissensbasierten Ansatz werden auch die Algorithmen als Schemata definiert und sind somit interaktiv zugänglich. Beispielsweise legen einfache Regeln fest, ob eine Maschine kontinuierlich oder diskret beladen und entladen wird. Die Flexibilität erreicht somit völlig neue Dimensionen.

Ein außerordentlich schwieriges Problem stellt die Frage nach Konsistenz und Vollständigkeit in wissensbasierten Systemen dar.

Auch dazu bietet KBS einen interessanten Ansatz. Mit Hilfe der Prädikatenlogik erster Ordnung lassen sich nämlich die Konsistenzbedingungen in Horn-Klauseln definieren. Beispielsweise wird festgelegt, daß für alle Schemata vom Typ FERTIGUNGSBEREICH und für alle Eintragungen im Slot LOKALE MASCHINEN auch tatsächlich Schemata vom Typ MASCHINE existieren.

Jede so definierte Konsistenzbedingung (constraint) wird von KBS ausgewertet. Falls eine Verletzung einer Bedingung erkannt wird, versucht das System den Fehler zu finden und anzuzeigen (Trace-Funktion). Auf diese Weise wird es möglich, inkonsistente Slotwerte und fehlende Schemata zu ermitteln.

3.5 Kriterien für den Einsatz von Expertensystemen

In den vorangegangenen Abschnitten dieses Kapitels wurden einige existierende Systeme und erfolgversprechende Ansätze vorgestellt. Geht man der Frage nach, welches denn die typischen Kriterien sind, die einen erfolgversprechenden Einsatz von Expertensystemen erwarten lassen, dann findet man in der Literatur meist den Hinweis auf diffuse oder vage Wissensdomänen. Dieses Kriterium trifft sicherlich auf die ersten Systeme speziell im medizinischen Bereich zu. Beispiele wie MYCIN oder GLAUCOMA sind bekannt. Betrachtet man aber vornehmlich die technischen Anwendungsfelder und dabei speziell das Beispiel XCON, so tritt der Aspekt des vagen und diffusen Wissens mehr und mehr in den Hintergrund.

In diesem Zusammenhang erscheint der Hinweis von LEHMANN [20] hilfreicher, der von schwach strukturierten Gebieten in Wissenschaft und Technik spricht, für die keine geschlossene Theorie existiert und die sich bisher der Automatisierung durch herkömmliche Computerprogramme verschlossen haben. Darüber hinaus sollte es sich um Aufgaben handeln, bei denen wenig gerechnet, dafür aber in recht differenzierter Weise unter Berücksichtigung vielfältiger Randbedingungen auch auf empirischer Basis geschlußfolgert und geurteilt wird. Komplexe numerische Probleme sind kein geeignetes Anwendungsgebiet. Generell gesprochen erscheinen alle

die Aufgaben prädestiniert, wo bereits heute qualifizierte Fachleute wirtschaftlich erfolgreich als Berater tätig sind.

Eine Aufstellung der Bedingungen, die ein potentiell Anwendungsbereich erfüllen sollte, gibt LEHMANN [20]:

- Das Aufgabengebiet sollte abgrenzbar und nicht zu umfangreich sein.
- Erfahrene Fachleute sollten schon heute auf dem Gebiet als Berater tätig sein.
- Fachbücher und Literatur sollten vorliegen. Ein gewisser Grad wissenschaftlicher Durchdringung und Systematisierung sollte erreichbar sein.
- Mindestens ein, besser aber mehrere menschliche Experten sollten verfügbar sein.
- Der Experte sollte sich durch Spezialwissen, Erfahrung sowie geschultes und differenziertes Beurteilungsvermögen auszeichnen.
- Der Experte muß sowohl bereit als auch dazu in der Lage sein, sein Wissen zu vermitteln und zu transferieren.

Weniger geeignet erscheinen dagegen Gebiete mit folgender Charakteristik:

- Exakte mathematische Lösungsverfahren oder zufriedenstellende automatisierte Lösungen existieren bereits.
- Das benötigte Wissen ist zu umfangreich. Der Aufgabenbereich ist nicht abgrenzbar.
- Das benötigte Wissen ist schwer erfaßbar und wenig formalisierbar. Alltagswissen (common sense knowledge) und gesunder Menschenverstand dominieren.
- Das Aufgabengebiet ist kaum hinreichend verstanden und es existiert kein menschlicher Experte als Erfahrungsträger.

Ein typisches Beispiel wo Expertensysteme sich schwer tun, ist das Wahrnehmen und Erkennen von Objekten oder Ereignissen. Das generelle Verständnis von Wahrnehmungsprozessen ist noch zu wenig ausgeprägt. Ähnlich verhält es sich bei der Beurteilung der Mon-

tagefreundlichkeit eines Produktes. Zwar können hierzu verschiedene Regeln angegeben werden, doch bleibt das Problem der effizienten und universellen Repräsentation der gestaltbestimmenden Daten von Objekten. Darüber hinaus erscheint die Wissensdomäne wenig abgrenzbar und zu komplex. Eine Gewichtung von Einflußfaktoren würde von dem existierenden Geräte- und Ausrüstungspark in entscheidendem Maße abhängen. Die Einflüsse aus anderen Betriebsbereichen müßten in bisher unbekannter Weise mit berücksichtigt werden. So wird man wohl auf ein funktionsfähiges und effizient arbeitendes Expertensystem zur montagegerechten Konstruktion noch einige Zeit warten müssen.

3.6 Entwicklungsprozeß von Expertensystemen

Es gehört zu den wesentlichen Eigenschaften von Softwareprodukten, daß sie einen komplexen Lebenszyklus durchlaufen. Solange sich Software im produktiven Einsatz befindet, sollte sie ständig gepflegt und weiterentwickelt werden. Dies trifft auf wissensbasierte und konventionelle Systeme in gleicher Weise zu. Betrachtet man den Lebenszyklus wissensbasierter Systeme (Bild 18), so fällt auf, daß hieran zwei unterschiedliche Personengruppen beteiligt sind. Neben dem Systemanalytiker, der auch bei der Entwicklung konventioneller Programme beteiligt ist, tritt als neue Komponente der Wissensingenieur hervor. Dabei sind die Aufgaben klar aufgeteilt.

Der Systemanalytiker bestimmt die Entwicklung des aufgabenneutralen Basissystems. Hierbei kann es sich um ein Werkzeugsystem allgemeiner oder spezieller Art oder auch um die Entwicklungsumgebung einer Programmiersprache handeln. Ausgehend von einer Systemanalyse werden Anforderungen definiert, Strukturen und Programme entworfen und anschließend auch implementiert. Einzelkomponenten sind zu integrieren und das Gesamtsystem zu installieren. Während des Probetriebes müssen auftretende Fehler behoben und Maßnahmen zur Effizienz- und Performanzsteigerung ergriffen werden.

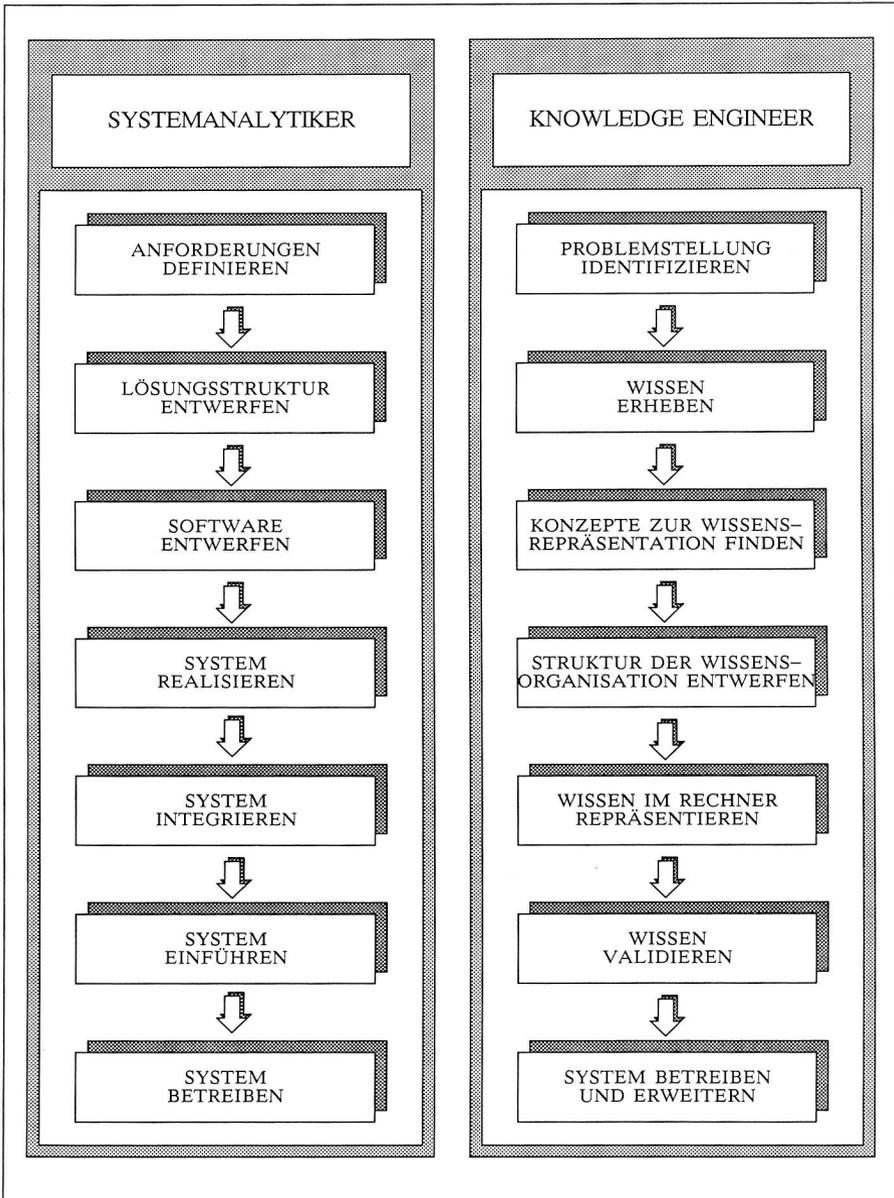


Bild 18: Entwicklungsprozeß von wissensbasierten Systemen
(Nach SCHACHTER-RADIG [63])

Parallel dazu beziehen sich die Aktivitäten des Knowledge Engineers, wie der Wissensingenieur auch genannt wird, auf die anwendungsspezifischen Aspekte eines Expertensystems. Der Knowledge Engineer steht zwischen dem Bereichsexperten, der das Basissystem nicht kennt, und dem Systemanalytiker, dem im Gegensatz dazu die Besonderheiten der Wissensdomäne nicht bekannt sind. Zu seinen Aufgaben zählt es demnach, das Wissen des menschlichen Experten zu erfassen, es geeignet aufzubereiten und zu strukturieren und es letztendlich in das System zu transferieren. Dabei erwächst das Problem, das Expertenwissen möglichst vollständig und auch korrekt zu übertragen.

Die Effizienz und Funktionalität von wissensbasierten Systemen wird ganz wesentlich davon beeinflusst, wie problemgerecht das Domänenwissen im System repräsentiert wird. Problemgerecht heißt in diesem Zusammenhang, wie genau die Inhalte, Strukturen und Abläufe im System nachgebildet werden können. Daraus resultiert auch die Erkenntnis, daß es universell einsatzfähige Werkzeugsysteme augenblicklich nicht gibt, und daß die Entwicklung hin zu multiplen und hybriden Werkzeugen führt. Die Konzepterstellung zur Wissensrepräsentation und die Strukturierung der Wissensorganisation kann aber nur derjenige erfolgreich durchführen, der sowohl mit dem System als auch mit der Wissensdomäne ausreichend vertraut ist. Genau dies ist die Position des Wissensingenieurs.

Expertensysteme arbeiten nicht von Anfang an fehlerfrei und in der gewünschten Weise. Vielmehr treten oftmals unsinnige oder auch falsche Problemlösungen auf, wenn Regeln in falschem Kontext angewendet werden. Selbst bekannte und erfolgreiche Systeme wie beispielsweise XCON mußten in einem mehrmonatigen Prozeß getestet und validiert werden. Hierin liegt eine weitere wichtige Aufgabe des Wissensingenieurs. In Zusammenarbeit mit dem menschlichen Experten müssen die Regeln selbst und ihre Einordnung in die Struktur des Problemlösungsprozesses gegebenenfalls neu überdacht werden.

Dem Wissensumfang eines Expertensystems liegt ein ständiges Wachstum zugrunde. Sinnvollerweise beginnt man mit abgegrenzten

und überschaubaren Wissensbereichen. Wenn man sich dann der korrekten Funktionsweise des Systems in diesem Rahmen sicher ist, erweitert man den Wissensumfang und damit natürlich auch das Einsatzgebiet des Systems. Dies ist einer der generellen Vorzüge von Expertensystemen, daß nämlich eine Funktionserweiterung über eine Erweiterung des Wissens erfolgt und nicht mit zusätzlicher Programmerstellung verbunden ist. Deshalb wird hierzu auch weniger der Systemanalytiker benötigt als vielmehr wieder der Wissensingenieur.

Softwaresysteme sind lebende Systeme, das heißt, sie sollten ständig aktualisiert und gewartet werden. Bei konventioneller Software erfolgt dies durch Daten- und Dateipflege oder durch Austausch von Programmteilen. Analog dazu müssen auch bei Expertensystemen die Wissensbereiche gepflegt und erneuert werden. Der Knowledge Engineer hat dabei die Aufgabe, die Wissensbereiche auf dem neuesten Stand zu halten und dabei auf Konsistenz zu achten, soweit dies nicht automatisch vom System gewährleistet wird.

Zusammenfassend kann man die Funktionen des Wissensingenieurs so beschreiben, daß er als Mittler zwischen dem Bereichsexperten und dem Systemanalytiker Aufgaben zur Akquirierung, Strukturierung und Aktualisierung von anwendungsspezifischen Wissensinhalten wahrnimmt.

4. Ein Werkzeugsystem zur Entwicklung von Expertensystemen

Betrachtet man die Entwicklungsgeschichte der frühen Expertensysteme, so fällt auf, daß man damals das Ziel verfolgte, nur für ein einziges, eng begrenztes und überschaubares Anwendungsgebiet eine spezifische Lösung zu schaffen. Aus den Erfahrungen der ersten Projekte heraus wurde einige Zeit später eine gewisse Zerteilung von Expertensystemarchitekturen erkannt. Da ist einmal das spezifische Wissen zu der ins Auge gefaßten Aufgabe, das auch das Domänenwissen genannt wird. Zum anderen lassen sich Komponenten, Strukturen und Abläufe herausstellen, die nicht auf die Domäne beschränkt sind, sondern auch auf andere Anwendungen übertragbar sind. Diese Erkenntnis führte zur Entwicklung von Werkzeugsystemen (vgl. Kap. 2.8.2). Je nach Mächtigkeit dieser Werkzeuge spricht man entweder von einer Wissensrepräsentations-sprache oder von fertigen Rahmensystemen, auch "Shells" genannt.

Hinter der Entwicklung des vorliegenden Rahmensystems(WWS) stand die Idee, ein wissensbasiertes Werkzeug für Planungs- und Konfigurationsaufgaben im Bereich der Produktionstechnik zu schaffen. Das Wissen in diesem Anwendungsbereich ist durch einen ausgeprägten Objektbezug gekennzeichnet. Es geht vornehmlich um die Entwicklung, die Herstellung oder den Umgang mit technischen Objekten. Diese Objekte sind in erster Linie technische Geräte, Werkstücke oder Werkzeuge. Die Schlußfolgerungsprozesse in diesem Anwendungsbereich laufen meist in der Form ab, daß die einzelnen Planungszustände Schritt für Schritt detailliert werden. Die ablaufenden Aktionen hängen sehr stark von eintretenden Ereignissen oder von vorliegenden Zwischenergebnissen ab.

Für derartige Anforderungen sind Produktionssysteme, die um eine objektbezogene Wissensrepräsentationsform erweitert sind, besonders geeignet. Das Rahmensystem wurde dementsprechend hybride konzipiert. Es besteht aus den Komponenten zur Dialogführung, zur Wissensrepräsentation, zur Problemlösung und Erklärung sowie zum Wissenserwerb. Die Komponenten selbst sind untereinander völlig autark und kommunizieren miteinander über definierte Schnittstellen. Auf diese Weise wird eine funktionsbedingte Modulari-

sierung mit den damit verbundenen Vorteilen der leichten Wartbarkeit und Übersichtlichkeit realisiert.

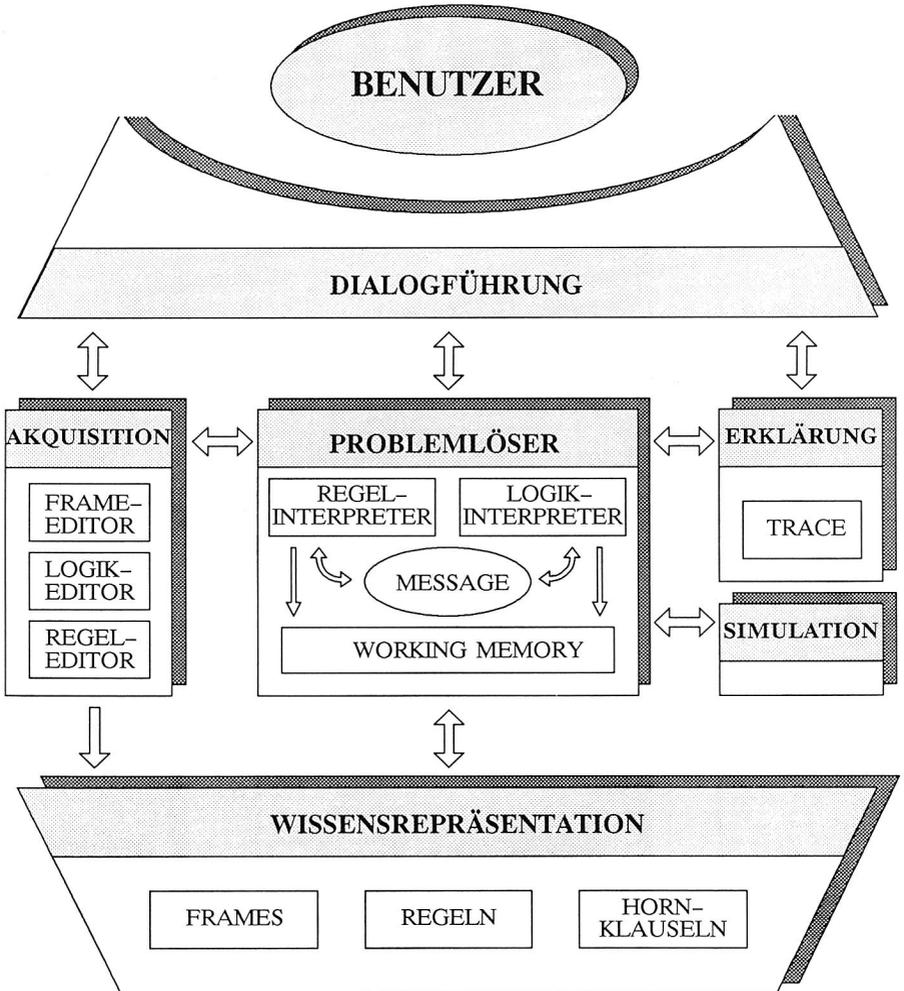


Bild 19: Struktur und Komponenten des Werkzeugsystems

4.1 Die Dialogkomponente

Unter dem Begriff "Dialogkomponente" soll im folgenden der Teil des Werkzeugsystems verstanden werden, der die gesamte Kommunikation mit dem jeweiligen Benutzer abwickelt. Entsprechend der Systemstruktur verbindet sie damit die Erwerbskomponente, den Problemlöser und die Erklärungskomponente mit dem Benutzer. Als Aufgaben dieses Dialogmoduls lassen sich die folgenden Punkte zusammenfassen:

- Bereitstellung geeigneter Hilfsmittel zur komfortablen Beschreibung der Planungsaufgabe,
- Durchführung einer direkten Kommunikation des Problemlösers mit dem Anwender über Nachrichten,
- Bereitstellung geeigneter Hilfsmittel zur Integration von neuen Wissenskomponenten in die Wissensbasis,
- Bereitstellung geeigneter Hilfsmittel zur Dokumentation der Inferenzvorgänge (Trace).

Eine genaue Beschreibung der Dialogtechniken beim Wissenserwerb findet sich in Kap. 4.5.

4.1.1 Allgemeine Anforderungen

Eine wesentliche Voraussetzung für Erfolg und Akzeptanz von interaktiven Programmsystemen ist eine ergonomisch gestaltete Benutzeroberfläche. Der Konzeption und Gestaltung dieser Mensch-Maschine-Schnittstelle sollte deshalb besondere Bedeutung beigemessen werden. In allgemeiner Form lassen sich dazu die folgenden Gestaltungsgrundsätze herausstellen [64]:

- Aufgabenangemessenheit,
- Selbsterklärungsfähigkeit,
- Leichte Erlernbarkeit,
- Fehlertoleranz,
- Steuerbarkeit,
- Verlässlichkeit.

Bei der Erstellung dieser Komponente wurde der Verfasser von Frau cand. inf. B. Bärnreuther unterstützt.

Die Gestaltung von Dialogsystemen sollte darüber hinaus noch unter Berücksichtigung von verschiedenen Benutzermodellen erfolgen. Hier lassen sich unterschiedliche, nach ihrer Geübtheit eingeteilte Benutzerklassen herausstellen:

- Trainierte Benutzer,
- Untrainierte Benutzer,
- Nichtexperten,
- Gelegenheitsbenutzer,
- Experten,

Die Fähigkeiten des Benutzers stehen im direkten Zusammenhang mit der für ihn optimalen Dialogform. Man unterscheidet hierbei [65]:

- Transaktionscodes,
- Kommandosprache,
- Menüsystem,
- Maskensystem,
- Direkte Manipulation,

Die Aufgabe einer Dialogkomponente muß es demnach auch sein, für die zu erwartenden Benutzer die optimale Dialogform bereitzustellen. Für die Visualisierung dieser Dialogform werden spezielle Dialogtechniken eingesetzt. Die wichtigsten und auch die neuesten sind die Piktogramme (icon) und die Fenstertechnik (window).

4.1.2 Dialogtechniken

Im Rahmen der Dialogkomponente des wissensbasierten Werkzeugsystems wurden neben Piktogrammen und Fenstern auch Masken eingesetzt.

Unter einem Fenster versteht man allgemein einen rechteckigen Ausschnitt des Bildschirms mit bestimmter Lage und Größe, dem entweder eine begrenzte oder aber eine unbegrenzte Informationsstruktur zugeordnet ist. Fenster können sich überlappen und gegenseitig verdecken. Dadurch wird der Bildschirm zur Infor-

mationsdarstellung auch zweieinhalbdimensional genutzt. Die Funktionsmerkmale von Fenstersystemen sind [66]:

- größeres Informationsangebot und bessere Ausnutzung des Bildschirms insbesondere bei überlappenden Fenstern,
- gleichzeitiger Zugriff auf mehrere Informationsquellen,
- gleichzeitige Kontrolle über mehrere Prozesse,
- Hilfestellungen und Erinnerungen durch spezielle Fenster,
- Fenster als Rahmen für bestimmte Interaktionsmöglichkeiten.

Das Verändern der Fenstergröße und der Fensterüberdeckung kann interaktiv und parametergesteuert vom Erstellungsprogramm aus erfolgen.

Piktogramme oder Ikonen sind Bilder, die sich auf einen realen Gegenstand beziehen, ihn aber der klaren oder raschen Information wegen, stilisiert oder typisiert wiedergeben. Piktogramme werden mit der Maus als Selektor direkt angewählt und erscheinen dann in einer invertierten Darstellung. Die Erstellung solcher Ikonen erfolgt über eine Manipulation der einzelnen Bildschirmpixel mit der Maus.

Die Zuordnung einer Ikone zu einem referierenden Programmsystem erfolgt über einfache Datei- und Variablenbezeichnungen. Der Knowledge Engineer kann somit jederzeit neue Symbole erzeugen und in die Dialogkomponente integrieren.

Bildschirmmasken werden in der Praxis zur Bewältigung großer Informationsmengen eingesetzt. Speziell geeignet sind sie zur Abfrage alphanumerischer Informationen vom Anwender. Masken zeigen die jeweilige Fragestellung am Bildschirm an und führen den Benutzer in der Reihenfolge der Eingabetätigkeiten. Die eingegebenen Daten werden automatisch in dafür vorgesehene Felder positioniert, wodurch Fehler vermieden werden.

Der Inhalt und der Aufbau der Masken kann durch eine Änderung der vorgegebenen Zeichenkette im Maskenerstellungsprogramm der Dialogkomponente wieder geändert oder auch erweitert werden.

4.1.3 Realisierte Dialogformen

Im Rahmen der vorliegenden Dialogkomponente wurden die Dialogformen der Menü- und Maskentechnik vereint. Dabei kam der Flächenaufteilung des Bildschirms eine wichtige Rolle zu. So haben Untersuchungen ergeben, daß bei einem in vier Quadranten eingeteilten Bildschirm der linke obere Teil zu 40 % der Gesamtzeit beachtet wird [65]. Daraus resultiert die Forderung, daß häufig verwendete Befehle und Hilfsmittel im oberen Bildschirmbereich plaziert sein sollten. Meldungen und Steuerinformationen sind im unteren Bildschirmbereich anzusiedeln.

Als übergeordnete Dialogform wurde eine hierarchisch gegliederte Menütechnik gewählt. Die einzelnen Kommandos werden hierbei in Form von Ikonen repräsentiert. Die Aktivierung eines Kommandos erfolgt durch Anklicken eines Ikons mit der Maus. Dem Anwender steht immer nur der Befehlsvorrat einer Stufe der Kommandohierarchie zur Verfügung. Wird die nächste Hierarchiestufe angesprungen, blendet sich ein weiteres Fenster (pop-up) mit den darin möglichen Kommandos auf.

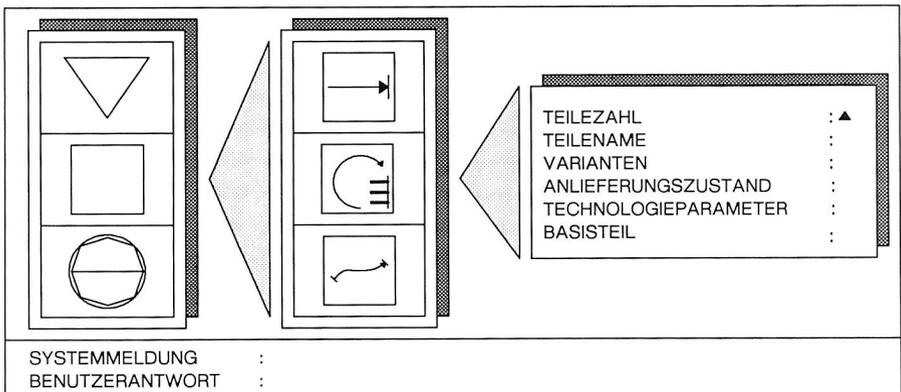


Bild 20: Beispiel für realisierte pop-up-Menüs

Die aktuellen Fensterbereiche bleiben solange aktiv, bis ein Abbruchsignal gegeben wird oder die nächste Hierarchiestufe ausgewählt wird. Meldungen und Anfragen der Problemlösungskomponente werden in einem speziellen Fenster im unteren Teil des Bildschirms dargestellt. Die darauf sich beziehenden Antworten des Benutzers kommen in dem gleichen Fenster zur Anzeige.

Im Rahmen der realisierten Dialogformen konnten die eingangs aufgestellten Gestaltungsgrundsätze bezüglich Aufgabenangemessenheit, Selbsterklärungsfähigkeit, leichter Erlernbarkeit, Fehler-toleranz, Steuerbarkeit und Verlässlichkeit in optimaler Weise berücksichtigt werden.

4.2 Das Wissensrepräsentationssystem

Im Rahmen des wissensbasierten Werkzeugsystems (WWS) hat diese Komponente die Aufgabe, geeignete Repräsentationsformen und Verwaltungsmechanismen für das Planungswissen bereitzustellen. Die Funktionen des Moduls lassen sich wie folgt zusammenfassen:

- Erzeugen, Löschen, Ändern und Verwalten von Produktionsregeln, Objektframes und Horn-Klauseln,
- Zusammenfassen von Regeln zu Regelgruppen,
- Kopplung von Produktionsregeln an Objektframes,
- Aufbau von hierarchischen Framestrukturen über Relationen mit Vererbungsmechanismen.

Wie alle anderen Module ist auch dieses Teilsystem völlig autark. Von spezieller Bedeutung ist in diesem Zusammenhang die klare Trennung zur Problemlösungskomponente. Letztere greift auf die Repräsentationskomponente zur Durchführung der Lese- und Schreibvorgänge nicht direkt, sondern über spezielle Schnittstellenfunktionen zu. Auf diese Weise kann die Konsistenz innerhalb dieses Verwaltungsmoduls gewährleistet werden.

 Bei der Erstellung dieser Komponente wurde der Verfasser von Herrn cand. inf. K.-J. Langer unterstützt.

4.2.1 Objekt-Frames

Die Repräsentationsform für das statische und dynamische Objektwissen sind die Frames. Die realisierten Funktionen entsprechen der Beschreibung aus Kapitel 2.4.4. Danach setzt sich jeder Frame aus einzelnen Slots zusammen. Die Slots selbst können wiederum in Subslots, sogenannte Facetten gegliedert werden. Für jeden Subslot sind die folgenden Facetten möglich:

- VALUE,
- DEFAULT,
- IF-NEEDED,
- RANGE,
- IF-ADDED,
- IF-REMOVED,
- COMMENT.

Die Auswertung dieser Strukturen erfolgt nach einer festen Reihenfolge. Bei einem lesenden Zugriff auf den Slot wird zuerst die VALUE-Facette ausgewertet. Ihr Wert kann eine Variable oder auch ein Datum sein. Im Anschluß daran erfolgt der Zugriff auf die IF-NEEDED-Facette. In ihr kann eine LISP-Funktion hinterlegt sein, die dann automatisch aktiviert wird. Als letztes wird die DEFAULT-Facette ausgewertet. In ihr sind Vorbelegungen von Werten und Variablen definiert.

Bei einem schreibenden Zugriff kann eine Wertebereichsüberprüfung mit der RANGE-Funktion angestoßen werden. Die Facetten IF-ADDED und IF-REMOVED können wiederum spezielle LISP-Funktionen beherbergen, die automatisch aktiviert werden, sobald etwas hinzugefügt oder etwas herausgenommen wird. Zur besseren Lesbarkeit der Slot-Inhalte können Kommentare in die COMMENT-Facette geschrieben werden. In Bild 21 sind diese Möglichkeiten am Beispiel eines Frames für einen Industrieroboter dargestellt.

Zu einer Zusammenfassung mehrerer Frames mit identischem Aufbau dienen die sogenannten Klassen. Für jede Klasse wird ein Typframe definiert (Bild 22). Diese sogenannten PROTOTYPEN legen die An-

FRAME "INDUSTRIEROBOTER"		
SLOT ACHSZAHN	VALUE	6
	DEFAULT	3
	COMMENT	SONDERBAUFORM
SLOT HANDHABUNGS- GEWICHT	RANGE	1-15 KG
SLOT GREIFER	IF_NEEDED	GREIFER_ERMITTELN
SLOT ZUBEHÖR	IF_ADDED	KOSTEN_NEU_BERECHNEN
SLOT SENSORIK	IF_REMOVED	KOSTEN_NEU_BERECHNEN

Bild 21: Aufbau und Funktionalität der Objekt-Frames

zahl und die Namen der einzelnen Slots fest. Zu diesen PROTOTYPEN existieren auch entsprechende Ausprägungen. In ihnen sind die Slots mit dem Wissen zu den realen Objekten belegt. Eine neue

TYPFRAME ROBOTER		AUSPRÄGUNG MANUTEC_R3	
SLOTNAME	SLOTEINTRAG	SLOTNAME	SLOTEINTRAG
BEZEICHNUNG		BEZEICHNUNG	MANUTEC_R3
HERSTELLER		HERSTELLER	MANUTEC
ACHSENZAHL		ACHSENZAHL	6
POSITIONIER- GENAUIGKEIT		POSITIONIER- GENAUIGKEIT	0.1 MM
⋮		⋮	

Bild 22: PROTOTYPEN und Ausprägungen von Frames

Ausprägung kann nur erzeugt werden, wenn der entsprechende PROTOTYP bereits vorliegt. Auf der anderen Seite wirken sich nachträgliche Änderungen am PROTOTYPEN auch automatisch auf dessen Ausprägungen aus. Beim Erzeugen einer Ausprägung wird der entsprechende PROTOTYP einfach kopiert und mit dem spezifischen Wissen der ausgeprägten Objekte gefüllt.

Zur Abbildung der realen Objekthierarchien können die Rahmen zu einer komplexen Struktur aufgebaut werden (Bild 23). Diese Struktur ist ein zusammenhängend gerichteter Graph mit genau einer Wurzel. Zyklische Strukturen sind nicht erlaubt. Der Graph besteht aus Frame-Knoten und Frame-Blättern, wobei die Blätter terminale Frames repräsentieren. Jeder Knoten darf eine beliebige Anzahl von Nachfolgern besitzen.

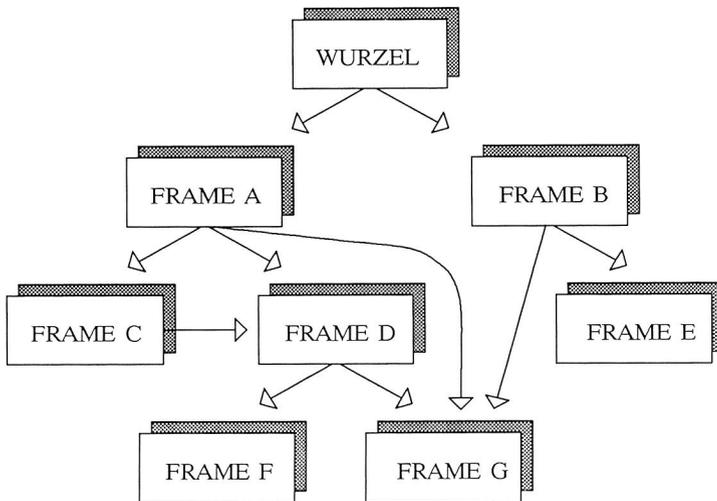


Bild 23: Bildung von Frame-Strukturen

Jeder Rahmen wird bei seiner Erstellung an eine bestimmte Stelle der Struktur platziert, indem sein direkter Vorgänger angegeben wird. Die Struktur kann mit Hilfe von Funktionen manipuliert werden, die neue Verbindungen (RELATIONEN) einfügen oder

bestehende Verbindungen löschen.

4.2.2 Hierarchische Vererbung

Eine wesentliche Eigenschaft von Frame-Systemen ist die Möglichkeit zur Vererbung von Objekteigenschaften. Beim Zugriff auf Frames mit Vererbung werden zusätzlich diejenigen direkten oder indirekten Vorgänger in den Datengewinnungsprozeß mit einbezogen, die ebenfalls über einen Slot mit gleichem Namen verfügen. Der jeweilige Typ des Rahmens ist bei diesem Vorgang ohne Bedeutung. Die Vererbung erfolgt also nicht primär von Rahmen zu Rahmen, sondern von Slot zu Slot.

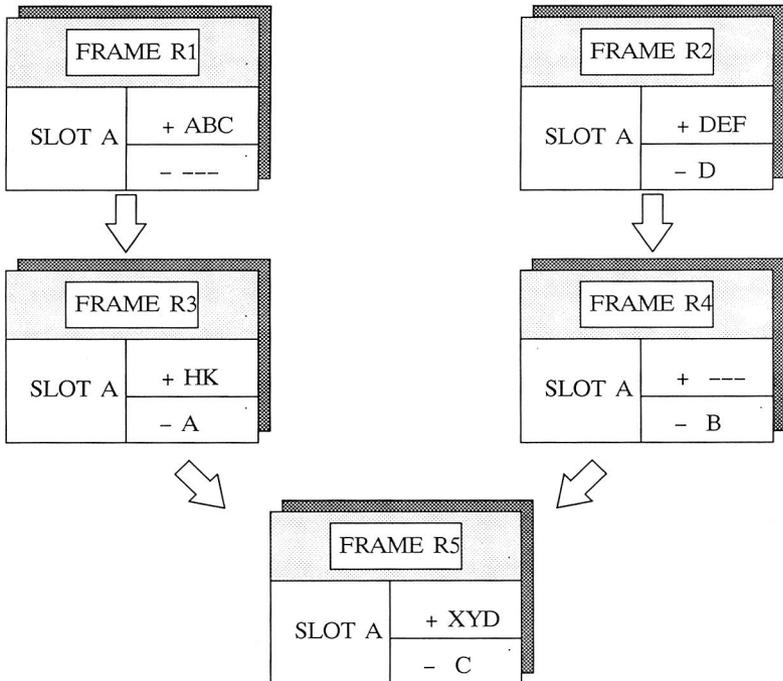


Bild 24: Beispiel für hierarchische Vererbung

Der eigentliche Vererbungsmechanismus läuft über Positiv- und Negativeintragungen in den Slots. Dabei dienen die Negativeinträge zur Unterdrückung des Mechanismus und zur Berücksichtigung von Ausnahmeregelungen.

Ein negativer Eintrag bedeutet also, daß der bezeichnete Slot in dem aktuellen Rahmen und in allen Nachfolgerahmen den eingetragenen Wert nicht besitzt, sofern nicht ein erneuter Positiveintrag erfolgt. Dabei dominiert ein ererbter Negativeintrag gegenüber einem ererbten Positiveintrag. Hingegen sind die gesetzten Slot-Werte gegenüber den ererbten weiterhin dominant. Die tatsächlichen Slot-Werte mit und ohne Vererbung zum obigen Beispiel zeigt die nachfolgende Aufstellung:

RAHMEN	WERTE VON SLOT A	
	OHNE VERERBUNG	MIT VERERBUNG
R1	+ A B C,	A B C
R2	+ D E F, - D	E F
R3	+ H K, - A	B C H K
R4	+ - B	E F
R5	+ X Y D, - C	H K E F X Y D

Bild 25: Slot-Werte bei der Vererbung

Dieser Vererbungsalgorithmus ist mächtiger als die üblicherweise in Rahmensystemen vorhandenen Algorithmen, da er nicht nur eine Vererbung innerhalb einer reinen Baumstruktur gestattet, sondern auch zuläßt, daß ein Rahmen (im Beispiel R5) von mehreren Seiten her Informationen erben kann. Konflikte werden dabei durch die Methode der Positiv- und Negativeinträge aufgelöst. Bei der Bearbeitung von Vorgängern wird zusätzlich noch berücksichtigt, daß gesetzte Positiveinträge die ererbten Negativeinträge überschreiben. Dadurch ergibt sich die Konstellation, wie sie in der rechten Hälfte von Bild 24 dargestellt ist. Der Wert D wird für

die Rahmen R2 und R4 durch den Negativeintrag in R2 unterbunden. Der entsprechende Positiveintrag im Rahmen R5 bewirkt jedoch eine Aufhebung dieser Vererbungssperre.

Mit Hilfe der beschriebenen Vererbungsfunktionen hat der Anwender somit die Möglichkeit, jegliche relevante Information bei nur einmaliger Speicherung allen Komponenten redundanzfrei zugänglich zu machen. Durch die Installation zusätzlicher RELATIONEN kann die an einer Stelle gespeicherte Information praktisch überall innerhalb der Baumstruktur ererbt werden. Der Mechanismus gewährleistet also ein Höchstmaß an Flexibilität bei geringstem Speicherplatzbedarf.

4.2.3 Assoziativzugriff

Eine wesentliche Aufgabenstellung bei der Planung technischer Systeme ist die zeiteffiziente Verwaltung von N:M-Beziehungen. So kann beispielsweise in der Montagetechnik eine Montagefunktion von einer Vielzahl von Geräten ausgeführt werden. Gleichzeitig ist jedes dieser Geräte auch in der Lage, eine Vielzahl anderer Montagefunktionen auszuüben.

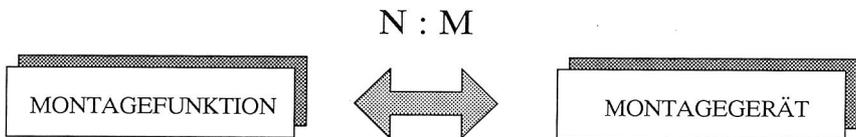


Bild 26: N:M-Beziehungen in der Montagetechnik

Während die einfachen 1:N Beziehungen mit Hilfe von Listen als Slot-Eintrag realisierbar sind, wurde für die N:M Beziehungen eine andere Lösung geschaffen. So kann der Knowledge Engineer jeden beliebigen Slot als Assoziativ-Slot deklarieren. Das System

legt daraufhin eine entsprechende Assoziativliste an. In dieser Liste ist dann vermerkt, in welchem Rahmen der Baumstruktur der Slot einen bestimmten Wert besitzt. Dadurch werden die Lesezugriffe extrem einfach und schnell. Gleichzeitig können einfache Abfragen wie beispielsweise

"Bei welchen Industrierobotern ist ein maximales Handhabungsgewicht von 30 Kg erlaubt?"

gestellt werden. Das System liefert dann die Namen aller Frames vom Typ INDUSTRIEROBOTER, bei denen im Slot HANDHABUNGSGEWICHT der Wert 30 eingetragen ist.

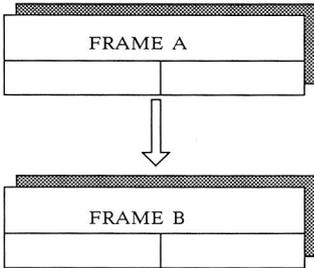
4.2.4 Strukturierung von Wissensbasen

Zur Strukturierung von Wissensbasen bietet das Repräsentationssystem insgesamt vier unterschiedliche Möglichkeiten (Bild 27). Die festen Vererbungsrelationen sind speziell zum Aufbau von festen Strukturen wie beispielsweise der Gerätehierarchie von komplexen Fertigungssystemen geeignet. Diese Relationen bezeichnen jeweils Vorgänger- und Nachfolgerrahmen innerhalb einer solchen Hierarchie.

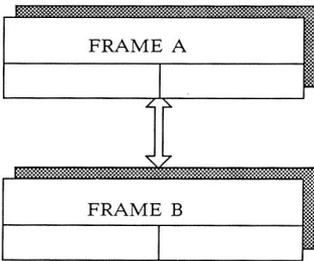
Die Horn-Klauseln können speziell eingesetzt werden, um variable und dynamische Abhängigkeiten und Strukturen zu repräsentieren. Die Klauseln werden in einer speziellen Liste verwaltet und können so in einfacher Weise angelegt und gelöscht werden, ohne daß auf Objekt-Frames zugegriffen werden muß. Dies kann speziell während des Problemlösens von Bedeutung werden, wenn sich Zugehörigkeiten von Substrukturen mehrmals ändern. Zur Abarbeitung dieser Logik existiert ein spezieller Horn-Klausel-Interpreter.

Über Namenseintrag in Slots können ebenfalls variable und zeitlich veränderliche Beziehungen realisiert werden. Ein Beispiel hierfür wäre die Zuordnung eines Werkstücks zu einer Bearbeitungsmaschine während der Bearbeitung. Der Zugriff erfolgt über den jeweiligen Frame-Name. Allerdings ist hier der Verwaltungs-

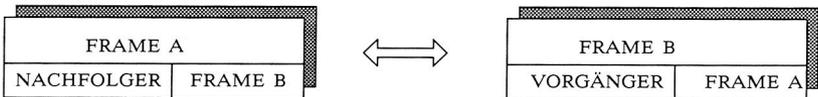
A) FESTE VERERBUNGSRELATIONEN



B) HORN-KLAUSEL-RELATIONEN



C) NAMEN ALS SLOTEINTRAG



D) ASSOZIATIVLISTE

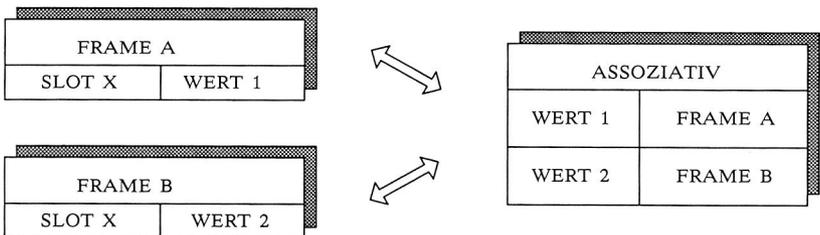


Bild 27: Möglichkeiten der Strukturbildung

aufwand nicht zu unterschätzen, da der Namenseintrag in zwei korrelierenden Frames getätigt und auch wieder gelöscht werden muß.

Die Assoziativliste eignet sich besonders für Abfragefunktionen, wie dies von Datenbankmanagementsystemen her bekannt ist. Allerdings bleibt die Funktionalität der Assoziativliste weit hinter den Möglichkeiten einer QUERY-LANGUAGE zurück. Es können nur einfache Abfragen vorgenommen werden. Der anschließende Zugriff erfolgt direkt über den ermittelten Frame-Name.

4.2.5 Produktionsregeln

Die Repräsentationsform für das deklarative Wissen zu den Planungsabläufen und Prozessen sind die Produktionsregeln. Die Regeln sind von ihrer Syntax her in der klassischen WENN-DANN-Form aufgebaut. Das heißt, sie bestehen aus einem Bedingungsteil und einem Aktionsteil. Jeder dieser Teile ist eine Liste, deren Elemente durch Konjunktion verknüpft sind. Zweckmäßigerweise ist jedes Element wiederum eine Liste, die die Formulierung einer singulären Bedingung oder Aktion enthält. Die genaue Syntax wird in Kapitel 4.3.3 erläutert.

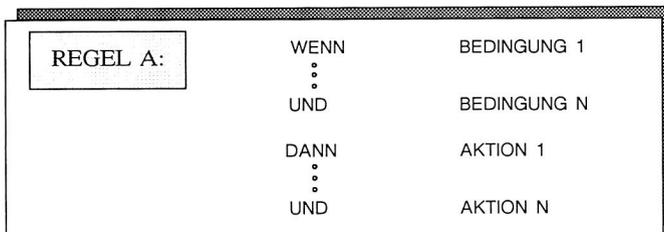


Bild 28: Aufbau der Produktionsregeln

Zur Strukturierung der Regeln bietet das System zwei Möglichkei-

ten. So lassen sich mehrere Regeln zu einer Regelklasse zusammenfassen. Diese Klassenbildung kann der Wissensingenieur nach beliebigen Gesichtspunkten vornehmen. Sie kann funktionsbedingt oder auch ablaufbedingt sein. Funktionsbedingte Regelklassen beziehen sich beispielsweise auf alle Geräte einer Geräteklasse oder auf Eigenschaften von Objekten, wie Werkstückeigenschaften oder kinematische Eigenschaften.

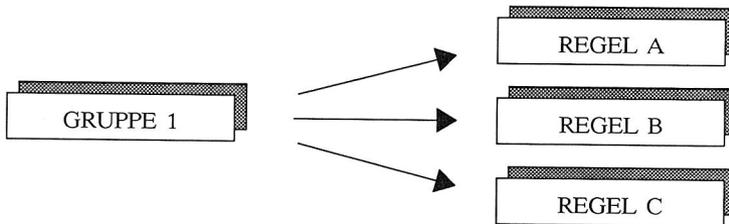


Bild 29: Zusammenfassen von Regeln zu Regelklassen

Diejenigen Regelklassen, die den Ablauf der Planungsvorgänge bestimmen und kontrollieren, werden METAREGELN genannt. METAREGELN repräsentieren das Wissen über das Wissen. Sie definieren beispielsweise, welche Aufgaben vordringlich und welche Aufgaben später bearbeitet werden sollen. METAREGELN sind also dazu gedacht, die Aktivierung der einzelnen funktional gebildeten Regelklassen zu steuern. Damit wird auch der Gesamtablauf der Planungsvorgänge kontrolliert. Wesentlich ist in diesem Zusammenhang, daß diese Steuerung nicht fest vorgegeben ist, wie in einem codierten Programm, sondern daß sie von Ereignissen und Zuständen abhängt, die im Bedingungsteil der Regeln spezifiziert sind. Auf diese Weise wird ein Höchstmaß an Flexibilität erzielt. Änderungen in der Ablauflogik lassen sich in einfacher Weise durch Regeländerungen bewerkstelligen.

Als zweite Möglichkeit zur Regelstrukturierung erlaubt das System die Kopplung von Regeln an Objektrahmen. Auf diese Weise können spezifische objektabhängige Regeln dem entsprechenden Frame

direkt zugeordnet werden. Als Beispiel hierzu sei die Bestimmung eines Greifers (Effektor) für einen bestimmten Roboter in Abhängigkeit verschiedener Randbedingungen genannt. Als Seiteneffekt können diese Regeln auch am Vererbungsmechanismus teilnehmen, indem sie auch für nachfolgende Rahmen zur Verfügung stehen.

Die Kopplung einer Regel an einen Frame kann auch dazu verwendet werden, um Kommentare und Erläuterungen der Regel zu dokumentieren. Beispiele hierfür wären Erstellungsdatum, Kontext und Autor der Regel. Neuzeitliche Werkzeugsysteme verfügen auch über mehrere Repräsentationsformen von Regeln; so beispielsweise eine natürlichsprachliche und eine vom Regelinterpreter interpretierbare Regelversion. Mehrere Versionen können dann an einem Rahmen hängen und so zugreifbar sein.

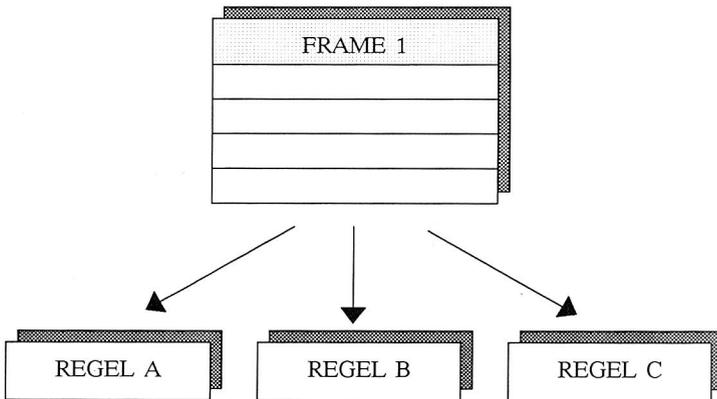


Bild 30: Kopplung von Regeln an Frames

Mit Hilfe der genannten Möglichkeiten können die in Produktionssystemen üblichen komplexen Suchvorgänge nach aktuell aktivierbaren Regeln auf ein Minimum reduziert werden. Dabei bleibt die Forderung nach Unabhängigkeit der einzelnen Regeln voll berücksichtigt.

4.3 Die Problemlösungskomponente

4.3.1 Allgemeine Funktionsweise

Der Problemlöser ist diejenige Komponente des Werkzeugsystems, in der die eigentlichen Schlußfolgerungsprozesse, auch Inferenzen genannt, zur Lösung der gestellten Aufgaben ablaufen. Die Arbeitsweise ist rein vorwärtsverkettend also datengetrieben. Ausgehend von einem aktuellen Planungszustand wird ein neuer Zustand ermittelt, welcher zu der Lösung der gestellten Aufgabe hinführt.

Die Vorgehensweise ist dabei so, daß ein Regelinterpreter die existierenden Produktionsregeln und Horn-Klauseln abarbeitet. Die Regeln repräsentieren bekanntlich das notwendige Planungswissen. Das Finden einer gültigen Regel erfolgt zum einen über eine Mustervergleichsoperation, auch PATTERN-MATCH genannt. Zum anderen wird eine Auswertung der Horn-Klauseln angestoßen. Bei der gesamten Operation (CHECK) werden die Bedingungen aus dem WENN-Teil einer Regel mit dem aktuellen Planungszustand und mit dem Wissen zu den Objekten verglichen. Wenn alle Bedingungen erfüllt sind, gilt die entsprechende Regel als aktiviert.

Bei komplexen Planungsvorgängen tritt leicht der Fall ein, daß nicht nur eine einzige sondern mehrere Regeln gleichzeitig aktiviert werden. Welche dieser Regeln nun auszuführen ist, muß eine geeignete Kontrollstrategie festlegen.

Die Ausführung einer Regel wird oftmals auch mit "feuern" bezeichnet. Dabei werden die im DANN-Teil einer Regel definierten Aktionen ausgeführt. Die Aktionen können sich auf Objekt-Frames, auf Logik-Ausdrücke und auch auf andere Planungsregeln beziehen und modifizieren dabei den aktuellen Planungszustand. Die Vorgänge sind in Bild 31 dargestellt.

Der Regelinterpreter kann grundsätzlich auf einer beliebigen

Bei der Erstellung dieser Komponente wurde der Verfasser von Herrn cand. inf. G. Geyer unterstützt.

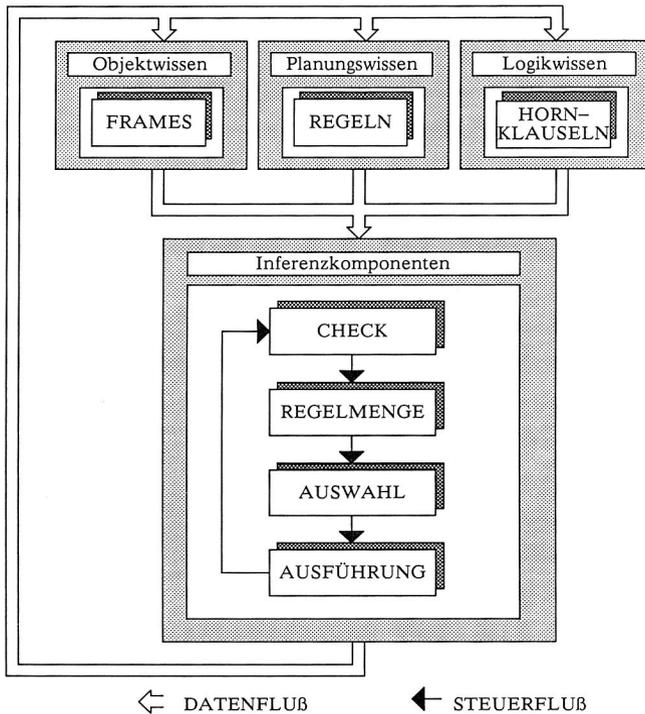


Bild 31: Allgemeine Funktionsweise des Problemlösers

Frame-Struktur arbeiten, solange das Regelwerk zusammen mit den Frames ein ablauffähiges auf einander abgestimmtes System bildet. Aus Gründen der besseren Übersicht ist eine Strukturierung der Wissensbereiche sinnvoll. Regeln, Prototypen und feste Wissensbasen bilden das sogenannte "Long-Time-Memory", also das Langzeitwissen. Der Begriff der PROTOTYPEN wurde bereits in Kapitel 4.2 erläutert. Es handelt sich dabei um Typframes, die in den Arbeitsbereich kopiert werden, falls neue Ausprägungen von Rahmen während des Planungsvorganges erzeugt werden müssen.

Der Arbeitsbereich des Systems wird auch WORKING-MEMORY genannt und bildet im Gegensatz zum "Long-Time-Memory" das sogenannte

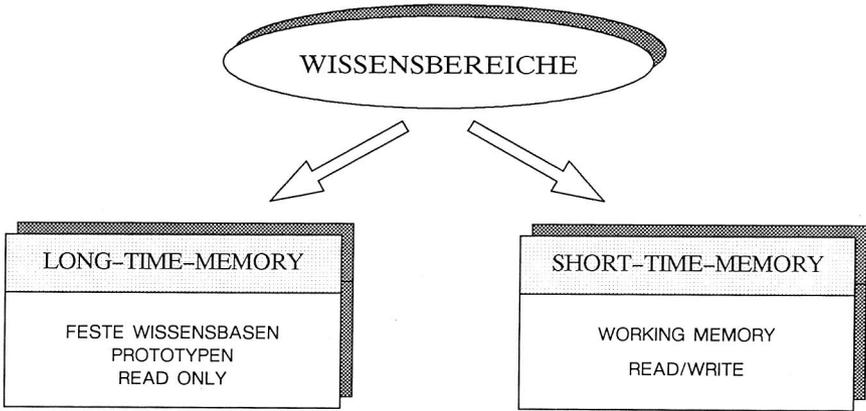


Bild 32: Strukturierung der Wissensbereiche

"Short-Time-Memory". Es handelt sich hier also um das Kurzzeitgedächtnis. In diesem WORKING-MEMORY wird der Verlauf der Planungszustände dokumentiert. Der Regelinterpreter bezieht sich bei seinen schreibenden Aktionen auf dieses WORKING-MEMORY. Zur Repräsentation der Planungszustände werden Frames verwendet. Zur Bildung von Frame-Strukturen und zum Aufbau von Beziehungen kommen die Logikrelationen (Horn-Klauseln) zum Einsatz. Das WORKING-MEMORY muß vom Wissensingenieur einmalig aufgebaut werden, bevor er das Werkzeugsystem für eine Expertensystemanwendung einsetzen kann.

4.3.2 Kontrollstrategien

Zur Arbeitsweise des Regelinterpreters wurden nun die in Bild 33 aufgezeigten drei Kontrollstrategien implementiert. Der einfachste Fall liegt vor, wenn nur eine einzige Regel aktiviert wurde. Sie kann sofort "feuern".

Falls bei mehreren Regeln der Bedingungsteil erfüllt wurde, entsteht eine Konfliktmenge. Zur Auflösung dieses Konfliktes werden

die folgenden Kriterien herangezogen:

- Alter einer Regel,
- Alter eines Elementes aus dem WORKING-MEMORY,
- Komplexität der Regelbedingungen,
- Einmalgültigkeit einer Regel,
- Zufall.

Das Alter einer Regel oder eines Objektes kann leicht ermittelt werden, da eine Systemuhr vorhanden ist, die sich abfragen läßt. Bei einer Regelaktivierung erfolgt dann eine Zuordnung der aktuellen Systemzeit.

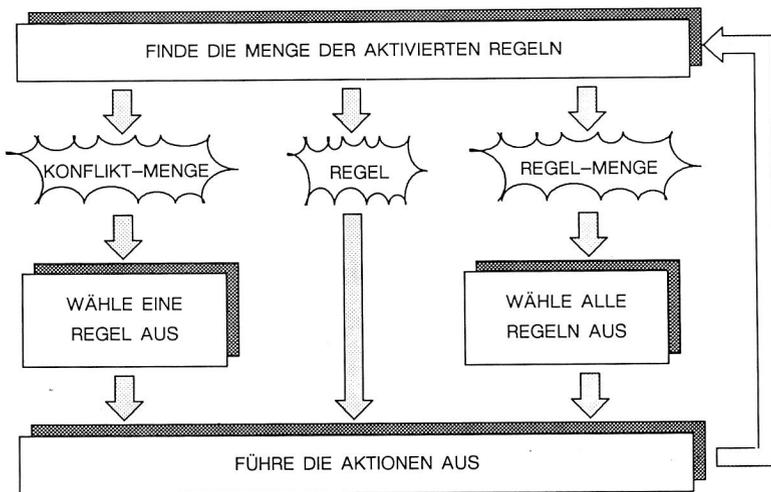


Bild 33: Strategien der Regelbearbeitung

Oftmals kommt es vor, daß mehrere Regeln im gleichen Kontext über unterschiedlich genau spezifizierte Bedingungen verfügen. Dabei spezifizieren komplexe Bedingungsteile die relevanten Zustände meist genauer und sind deshalb den pauschaleren aber ebenfalls

gültigen Anweisungen vorzuziehen.

Um zu verhindern, daß ein und dieselbe Regel ständig feuert und deshalb die Kontrolle nicht mehr abgibt, kann als Kriterium eine einmalige Gültigkeit festgelegt werden. Sobald sie dann gefeuert hat, wird sie automatisch wieder deaktiviert.

Wenn alle diese Kriterien nicht zu einer Auflösung des Konfliktes führen, dann wird als letzte Möglichkeit der Zufall eingeschaltet. Ein Zufallsgenerator liefert hierzu die notwendige Information.

Die dritte und letzte Strategie wird ebenfalls dann verfolgt, wenn eine nicht singuläre Menge aktivierter Regeln vorliegt. Allerdings erfolgt dabei keine Konfliktauflösung. Vielmehr kommen dann alle aktivierten Regeln zur Ausführung. Diese Strategie ist allerdings höchst gefährlich, da sich hier während der Regelausführung die Gültigkeit der Bedingungen verändern kann. Sie sollte deshalb nur sehr zurückhaltend und wohlüberlegt eingesetzt werden.

Die hier dargelegten Ausführungskriterien werden den Regeln bei ihrer Erstellung vom Wissensingenieur direkt zugewiesen. Sie sind somit fest an die Regeln gebunden.

4.3.3 Der Regelinterpretier

Die Arbeitsweise des Regelinterpreters ist die einer permanent durchlaufenen Schleife (Loop). Gemäß der allgemeinen Regelsyntax werden ständig Regeln mit Bedingungen und Aktionen bearbeitet:

```
REGEL:= (( [Bedingung] )( [Aktion] ))
```

Die Bedingungen einer Regel können aus einer beliebigen Anzahl von Muster- und auch von Logikausdrücken (Horn-Klauseln) bestehen. Sowohl für die Muster als auch für die Horn-Klauseln existieren spezielle Interpreter, die die Auswertung dieser

Ausdrücke vornehmen. Die Kontrolle über LOGIK-INTERPRETER und PATTERN-MATCHER übt ein sogenannter CHECKER aus. Der gesamte Mechanismus wurde in einem objektorientierten Ansatz implementiert. Die Kommunikation zwischen den beteiligten Prozessoren erfolgt über Nachrichten (message-passing). Mit den so realisierten Funktionen können die folgenden Ausdrücke verarbeitet werden:

- reine Mustervergleichsoperationen,
- reine Logik-Ausdrücke,
- gemischte Ausdrücke.

Tabelle 1 zeigt die Syntax der Regelbedingungen.

BEDINGUNG	:= ([B-AUSDRUCK]).
B-AUSDRUCK	:= (MUSTER-AUSDRUCK, LOGIK-AUSDRUCK).
MUSTER_AUSDRUCK	:= (FRAME-NAME [MUSTER_FUNKTION])
MUSTER_FUNKTION	:= (>? SLOTNAME VAR_ODER_ZAHL), (<? SLOTNAME VAR_ODER_ZAHL), (>=? SLOTNAME VAR_ODER_ZAHL), (<=? SLOTNAME VAR_ODER_ZAHL), (=? SLOTNAME VAR_ODER_WERT), (#? SLOTNAME VAR_ODER_WERT), (? SLOTNAME [VAR_ODER_WERT]), (/? SLOTNAME VAR_ODER_ZAHL VAR_ODER_ZAHL).
LOGIK-AUSDRUCK	:= ([LOGIK-ZIEL]).
LOGIK-ZIEL	:= (NAME [LOGIK-OBJEKT]), (VARIABLE [LOGIK-OBJEKT]).
LOGIK-OBJEKT	:= ([LOGIK-OBJEKT]), VARIABLE, NAME, ZAHL.

Tabelle 1: Regelsyntax (Bedingungsteil)

Die Tabelle 2 zeigt die Semantik der Regelbedingungen:

- (>? SLOTNAME VAR_ODER_ZAHL),
Führt einen "größer-Vergleich" aus: Ist der Inhalt des Slots
SLOTNAME aus dem aktuellen Frame "größer" als VAR_ODER_ZAHL ?
- (<? SLOTNAME VAR_ODER_ZAHL),
Führt einen "kleiner-Vergleich" aus: Ist der Inhalt des Slots
SLOTNAME aus dem aktuellen Frame "kleiner" als VAR_ODER_ZAHL ?
- (=? SLOTNAME VAR_ODER_WERT),
Führt einen "ist-gleich-Vergleich" aus: Ist der Inhalt des
Slots SLOTNAME aus dem aktuellen Frame "gleich" VAR_ODER_ZAHL ?
- (#? SLOTNAME VAR_ODER_WERT),
Führt einen "ungleich-Vergleich" aus: Ist der Inhalt des Slots
SLOTNAME aus dem aktuellen Frame "ungleich" VAR_ODER_ZAHL ?
- (? SLOTNAME [VAR_ODER_WERT]),
Führt einen "ist-Element-von-Vergleich" aus:
Ist der Inhalt des Slots SLOTNAME aus dem aktuellen Frame
"Element von" der Menge [VAR_ODER_ZAHL] ?
- (/? SLOTNAME VAR_ODER_ZAHL VAR_ODER_ZAHL).
Führt einen "liegt-im-Intervall-von-Vergleich" aus:
Liegt der Inhalt des Slots SLOTNAME aus dem aktuellen Frame
"im-Intervall-von" VAR_ODER_ZAHL bis VAR_ODER_ZAHL ?

Tabelle 2: Bedeutung der Mustervergleichsfunktionen

Die Tabelle 3 beschreibt die Syntax und Semantik der Aktionen:

Aktion :=

- (<set> FRAME_NAME SLOT_NAME WERT)
Setze den Slot SLOT_NAME des Frames FRAME_NAME auf den Wert WERT (mit set-slot).
- (<halt>)
Abbrechen des Regelinterpreterlaufs
- (<create> PROTOTYPE)
Kopieren eines Prototypframes in das WORKING MEMORY
- (<createslot> FRAME_NAME SLOT_NAME)
Hinzufügen des Slots SLOT_NAME in Frame FRAME_NAME.
Alle Frames vom selben Typ haben dann den zusätzlichen Slot
- (<createtype> PROTOTYPE_NAME SLOT_LISTE)
Erzeugen eines neuen PROTOTYPEN
- (<delval> FRAME_NAME SLOT_NAME)
Löschen der VALUE-Facette
- (<delslot> FRAME_NAME SLOT_NAME)
Löschen des Slots SLOT_NAME aus dem Frame FRAME_NAME
- (<delframe> FRAME_NAME)
Löschen des Frames FRAME_NAME aus der Wissensbasis
- (<deltype> PROTOTYPE_NAME)
Löschen eines PROTOTYPEN aus der Wissensbasis
- (<backup> DATEINAME)
Sichern der Wissensbasis unter DATEINAME
- (<restore> DATEINAME)
Einlesen einer Wissensbasis von DATEINAME

Tabelle 3a: Syntax und Semantik der Aktionen

- (<traceon> NUMMER DATEINAME)
Trace NUMMER auf Datei DATEINAME ausgeben
- (<traceoff>)
Trace ausschalten
- (<interpreter> REGEL_GRUPPE)
Aufruf des Regelinterpreters mit den Regeln der REGEL_GRUPPE
- (<simulation>)
Simulator-Aufruf
- (<dialog>)
Dialogkomponenten-Aufruf
- (<read> FRAME_NAME SLOT_NAME)
Einlesen von Benutzereingabe in den Slot SLOT_NAME des Frames
FRAME_NAME
- (<writevar> [VAR_ODER_WERT])
Ausgabe von Variablenwerten oder Werten
- (<writeslot> FRAME_NAME SLOT_NAME)
Ausgabe von Slotinhalt des Slots SLOT_NAME des Frames FRAME_NAME
- (<eval> VAR_ODER_WERT)
Evaluieren einer Variablenbindung oder einer Slotbindung
- (<delrule> REGEL_NAME)
Löschen einer Regel
- (<aktrule> REGEL-GRUPPE)
Aktivieren der Regeln der Gruppe REGEL-GRUPPE
- (<deaktrule>)
Deaktivieren der Regel

Tabelle 3b: Syntax und Semantik der Aktionen

4.4 Der Simulator

In der Produktionstechnik wird die Simulation immer mehr zu einem wichtigen Werkzeug zur Abschätzung des Einsatzverhaltens komplexer technischer Systeme und Anlagen. Wesentlich ist hierbei, daß die zu simulierenden Systeme noch nicht existieren, sondern sich noch im Planungsstadium befinden. Die Simulation dient somit der Optimierung komplexer, technischer Systeme und Abläufe mit dem Ziel, Planungsfehler zu vermeiden und dadurch Zeit und Kosten einzusparen.

Man unterscheidet im wesentlichen zwei Formen von Simulationsmodellen (Bild 34): Einmal die stetigen Systeme, wie beispielsweise die Bewegungssimulation von Industrierobotern, und dann die diskreten Systeme, wie sie auch zur Abschätzung des Einsatzverhaltens automatischer Montageanlagen angewendet werden. Das Verhalten eines zeitdiskreten Systems ist charakterisiert durch eine Folge von diskreten Zuständen, die es aufgrund von Interaktionen seiner Objekte untereinander und mit der Systemumgebung annimmt. Interaktion und Zustandsänderung vollziehen sich plötzlich und übergangslos zu diskreten Zeitpunkten und nicht kontinuierlich wie bei stetigen Systemen. Zustandsänderungen können an das Auftreten von Ereignissen geknüpft sein. Ein Ereignis besitzt keine zeitliche Dauer. Es bezeichnet nur einen Zeitpunkt, zu dem sich der Zustand des Systems ändert. Ein Ereignis leitet im allgemeinen eine Aktivität ein oder beendet sie.

Als interne Taktungsmethoden bei diskreten Simulationssystemen unterscheidet man zwischen Inkrement- und Ereignistaktung. Bei der Inkrementtaktung wird die Systemzeit jedesmal um ein festes Zeitinkrement erhöht, und jedesmal werden alle Objekte daraufhin untersucht, ob eine Zustandsänderung durchzuführen ist. Diese Taktungsweise ist immer dann sinnvoll, wenn die Ereignisse fast ausschließlich gleichzeitig auftreten. Die Ereignissteuerung und Zeitverfolgung wird dann sehr einfach. Bei der Ereignistaktung

Bei der Erstellung dieser Komponente wurde der Verfasser von
Herrn cand. inf. H. Pezolt unterstützt

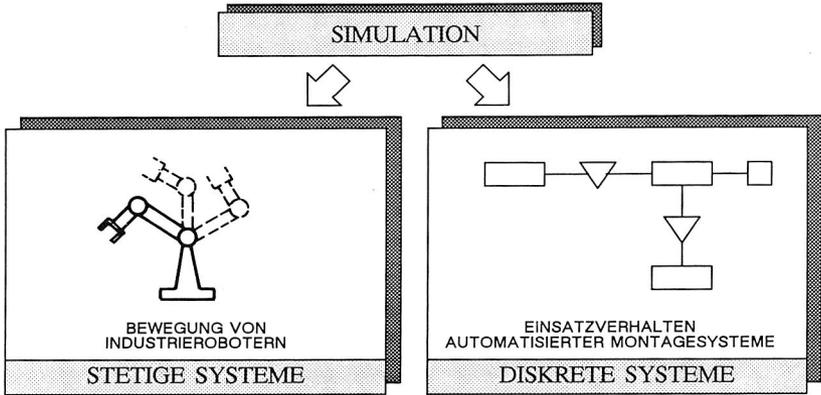


Bild 34: Simulationsmodelle

wird die Systemzeit immer bis zum nächsten Ereignispunkt erhöht. Es werden nur diejenigen Objekte untersucht, bei denen eine Zustandsänderung stattfindet. Das gleiche gilt für eine konsekutive Ereignisauslösung durch das Ereignis bei anderen Objekten. Diese Vorgehensweise bietet sich beispielsweise bei der Simulation von elastisch verketteten Fertigungssystemen an. Die Zustandsänderungen finden hier selten gleichzeitig statt. Durch die Anwendung dieser Taktungsweise kann die Rechenzeit beachtlich gesenkt werden.

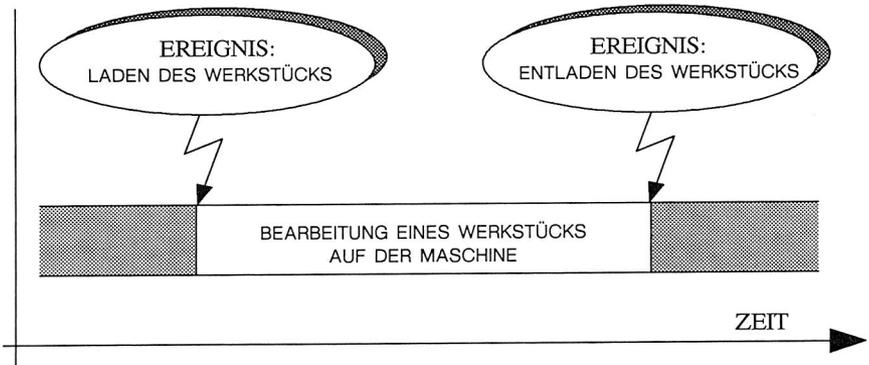


Bild 35: Simulation und Ereignisse

Entsprechend der dargestellten Unterscheidungen wurde der Simulator zeitdiskret und ereignisgetaktet konzipiert. Danach werden die einzelnen Ereignisse in einer Ereignisliste, dem sog. Kalender, festgehalten und in der Reihenfolge ihres Auftretens ausgeführt. Neu generierte Ereignisse werden in Abhängigkeit ihrer Ausführungszeit in den Kalender eingetragen. Ereignisse haben einen Namen und eine Ausführungszeit. Um welche Ereignisart es sich handelt und auf welches Objekt sich das Ereignis bezieht, beschreiben Funktion und Zielobjekt.

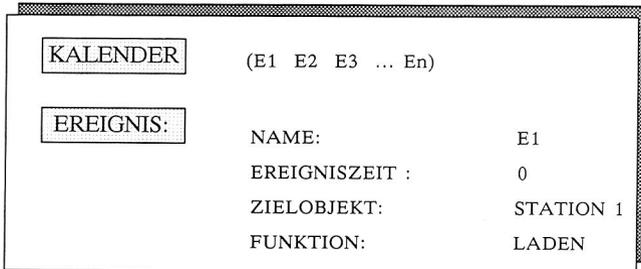


Bild 36: Struktur von Kalender und von Ereignissen

4.4.1 Simulationsobjekte

Als Zielobjekte kann das System zwei Entitytypen unterscheiden. Zum einen ist dies die Objektklasse STATION als Repräsentant aktiver Elemente, wie beispielsweise ein Handhabungsgerät oder eine Bearbeitungsmaschine. Zum anderen werden passive Elemente wie Stautrecken oder Magazine durch die Klasse PUFFER dargestellt. Einen umfassenden Überblick zu den gebräuchlichen Begriffen und Ausdrücken bei der Simulation technischer Maschinensysteme gibt REISCH [67]. Zur weiteren Vertiefung sei diese Arbeit empfohlen. Die möglichen Zustände für Station und Puffer beschreibt Tabelle 4.

WARTEND	Die Station ist inaktiv, da das Vorgängerobjekt kein Werkstück zur Bearbeitung liefert.
BELEGT	Die Station bearbeitet ein Werkstück.
BLOCKIERT	Die Station hat ein Werkstück fertig bearbeitet. Durch eine Blockierung bzw. Störung in den Nachfolgerobjekten konnte das Werkstück nicht vom Nachfolger übernommen werden, sodaß das fertige Werkstück die Station blockiert.
GESTÖRT	Durch eine Störung ist die Station vollkommen ausgefallen bzw. abgeschaltet.
BEREIT	Zwischenzustand ohne zeitlichen Verlauf.

Tabelle 4: Mögliche Stationszustände

Zur Abbildung realer Fertigungssysteme und -vorgänge können mit den beiden beschriebenen Objektklassen in beliebiger Weise Strukturen aufgebaut werden. Nach REISCH unterscheidet man hierbei zwischen:

- loser Verkettung,
- elastischer Verkettung und
- starrer Verkettung.

Bei einer starren Verkettung sind die Maschinen und Geräte starr miteinander verbunden. Fällt eine Station aus, werden die vorge-schalteten Stationen blockiert, da sie ihre fertiggestellten Werkstücke nicht weitergeben können (Blockierdauer). Die nachge-schalteten Stationen warten dagegen auf nachrückende Werkstücke (Wartedauer). Es wird deutlich, daß solche Systeme sehr empfindlich auf Störungen reagieren.

Bei einer losen Verkettung besteht die Anlage aus Einzelstationen, die sich in ihrem Ausfallverhalten nicht beeinflussen und auch nicht untereinander abhängig sind. Störungen werden durch "unendlich" große Puffer aufgefangen. Solche Puffer mit unendlicher Kapazität findet man in der Praxis recht selten. Dazu ist das in den Lagerstätten gebundene Kapital zu groß.

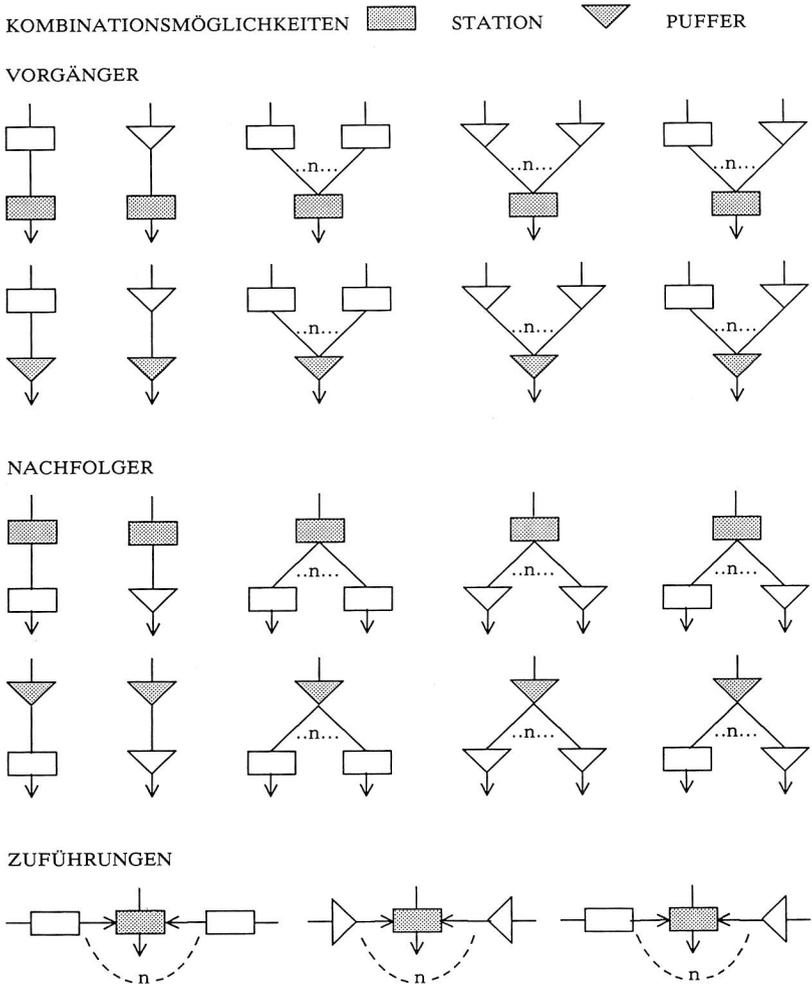


Bild 37: Beispiele für interne Modellstruktur

Die wohl am häufigsten anzutreffende Verkettungsart ist die flexible Verkettung. Die Fertigungs- oder Montageanlage wird durch die Pufferelemente in Einzelabschnitte aufgegliedert, sodaß die Störungszeiten der jeweiligen Stationen kompensiert werden können. Die Verfügbarkeit der Gesamtanlage kann durch optimale Dimensionierung der Pufferkapazitäten stark verbessert werden.

In Bild 37 sind einige Beispiele zu den Modellierungsmöglichkeiten des Simulationssystems gegeben.

4.4.2 Simulationsergebnisse

Die Ereignisse sind die eigentlichen Informationsträger während des Simulationslaufes. Das Zusammenwirken von Objektklassen, Ereignisfolgen und Zustände ist in den Bildern 38 und 39 dargestellt. Danach sind dem System die folgenden Ereignisse bekannt:

- LADEN einer Station,
- ENTLADEN einer Station,
- TRANSPORT eines Werkstücks,
- ANKUNFT eines Werkstücks,
- STÖRUNGSEINTRITT bei einer Station oder einem Puffer,
- STÖRUNGSENDE bei einer Station oder einem Puffer,
- INKREMENT der aktuellen Pufferbelegung,
- DEKREMENT der aktuellen Pufferbelegung.

Hierbei ist zu beachten, daß ein Ereignis stets zu diskreten Zeitpunkten auftritt. Daher sind die Ereignisse LADEN bzw. ENTLADEN nicht als zeitverbrauchende Vorgänge zu verstehen. Sie bezeichnen vielmehr den Zeitpunkt "Eintritt eines Werkstückes in die Station" bzw. "Verlassen einer Station". Die Ereignisse selber stehen untereinander in Wechselbeziehungen. Jedes Ereignis bewirkt die direkte Folge eines zugeordneten anderen Ereignisses.

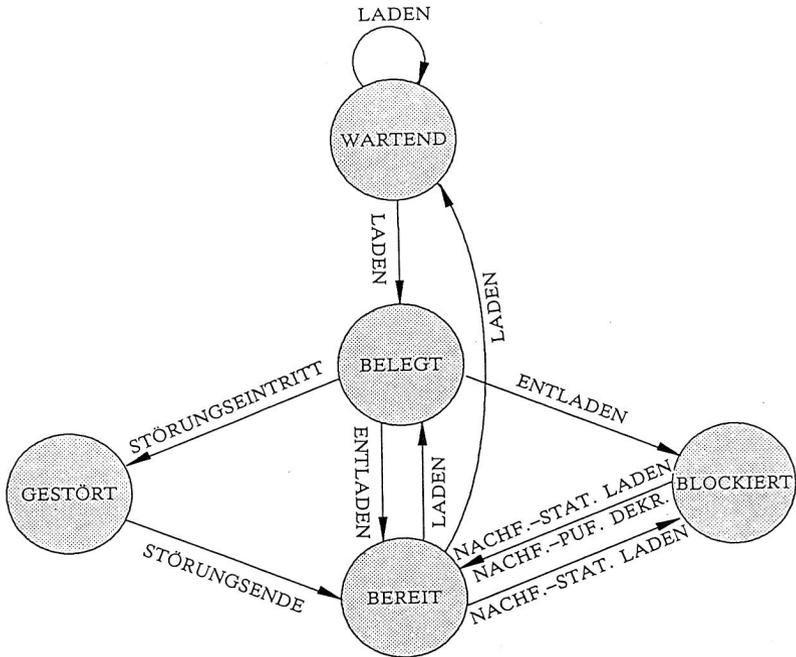


Bild 38: Zustände und Ereignisse einer Station

Das Einbeziehen von Störungen in das Simulationsverfahren stellt naturgemäß eine gewisse Problematik dar. Während der Planungsphase von technischen Systemen liegen noch keine Erkenntnisse über die zu erwartenden Störungen der zukünftigen Anlage vor. Erfahrungen haben aber gezeigt, daß das Eintreten von Störungen einen stochastischen Prozeß darstellt. Solche stochastischen Prozesse kann man mit Hilfe von mathematischen Verteilungsfunktionen nachbilden. Dies wird von den meisten konventionellen Simulationssystemen berücksichtigt. Auch der vorliegende wissensbasierte Simulator arbeitet mit einer solchen Verteilungsfunktion. Es wurde hierzu die Umkehrfunktion der Exponentialverteilung implementiert.

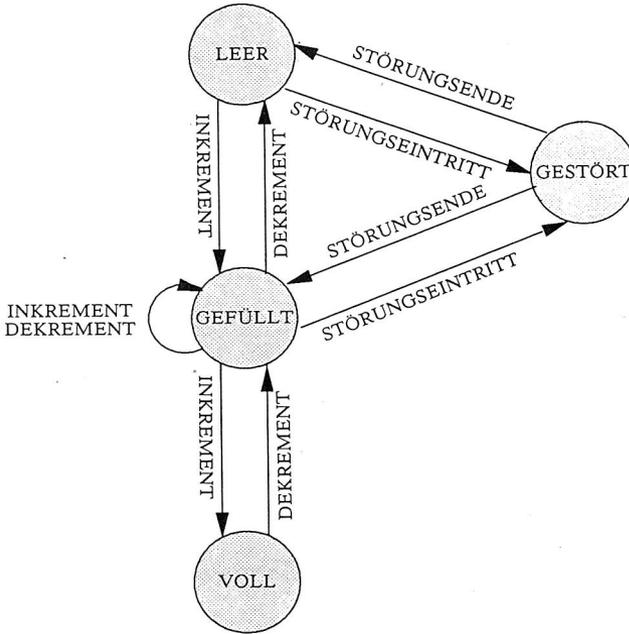
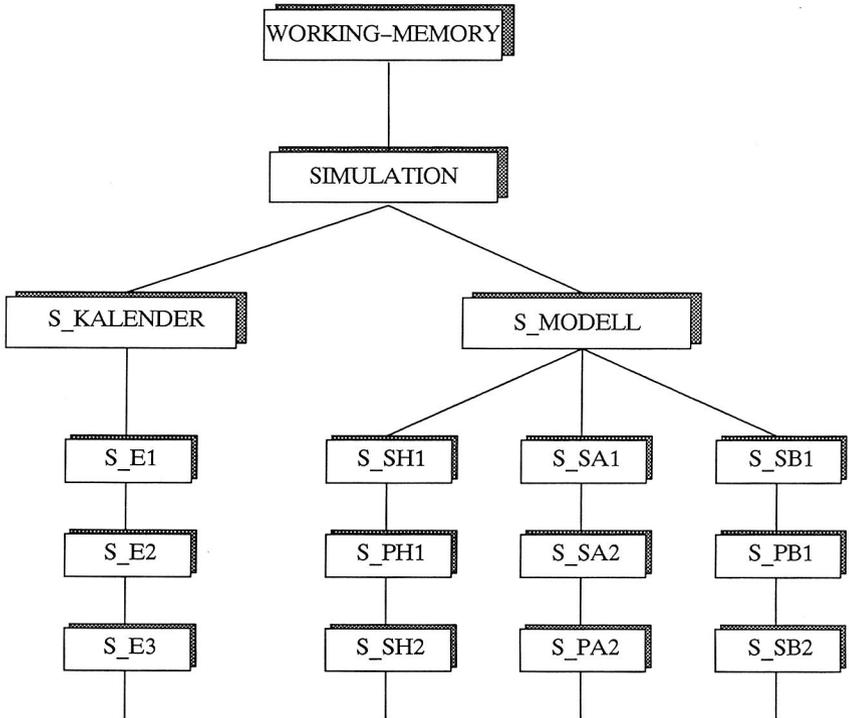


Bild 39: Zustände und Ereignisse eines Puffers

4.4.3 Integration in das wissensbasierte Werkzeugsystem

Der Simulationsmodul ist in Form einer eigenständigen LISP-Funktion implementiert und kann als solcher auch über den entsprechenden Funktionsaufruf aktiviert werden. Hierbei bedient sich der Simulator der Möglichkeiten des Wissensrepräsentations-systems. Das Simulationsmodell besteht danach aus einer Frame-Struktur, wie sie in Bild 40 dargestellt ist. Sowohl für die Ereignisse des Kalenders als auch für Stationen und Puffer des Modells werden Frames angelegt. Die Verwaltung dieser Struktur erfolgt mit den Schnittstellenfunktionen des Wissensrepräsentationsmoduls, wodurch zusätzlicher Aufwand vermieden werden konnte. Gleichzeitig wird eine einheitliche Datenhaltung mit den Vorteilen einer erhöhten Konsistenz gewährleistet.



S_ : SIMULATIONS-RAHMEN VOM TYP
 E : EREIGNIS
 S : STATION
 P : PUFFER

Bild 40: Datenstruktur des Simulationsmodells

Mit der Integration des Simulators in das wissensbasierte Planungs- und Konfigurationssystem (WWS) werden die folgenden wesentlichen Vorzüge erreicht:

1. Der bei konventionellen und anderen wissensbasierten Simulationssystemen beträchtliche Aufwand zur Erstellung eines korrekten Simulationsmodells durch den Benutzer entfällt völlig.

Das System baut zum Simulationsstart die Struktur des in Planung befindlichen Fertigungssystems automatisch auf. Der Benutzer muß also nicht interaktiv tätig werden und Stationen und Puffer aus Bibliotheken konfigurieren.

2. Der Simulator verwendet dieselben Wissensrepräsentationsmechanismen wie das Werkzeugsystem. Dadurch wird eine möglichst einheitliche Wissensverwaltung erreicht und mögliche Informationsverluste beim Wissenstransfer werden vermieden.
3. Ohne Übertreibung kann man feststellen, daß sich durch eine mögliche Wechselwirkung zwischen einer Anwendung des Werkzeugsystems und dem Simulator völlig neue Optimierungsmöglichkeiten eröffnen. Die Simulationsergebnisse können innerhalb eines einzigen Systems von den Produktionsregeln interpretiert werden. Im Zuge der Optimierung wird daraufhin das Planungsmodell korrigiert. Ein Beispiel hierfür wäre die Veränderung einer Pufferkapazität in Abhängigkeit von den simulierten Auslastungen. In einer rekursiven Weise kann daraufhin die neue Planungssituation nochmals simuliert werden, bis dann ein Optimum erreicht ist.
4. Planung und Simulation von komplexen Fertigungssystemen sind zwei in der Planungssystematik direkt aufeinander folgende Tätigkeiten eines menschlichen Experten. Es ist danach nur konsequent, wenn ein rechnergestütztes Werkzeug diese beiden Funktionalitäten in sich vereint.

4.4.4 Simulationsablauf

Der Ablauf der Simulation ist in Bild 41 dargestellt. Nach dem Aufbau des Simulationsmodells und nach der Initialisierung des Kalenders werden die darin vermerkten Ereignisse nacheinander abgearbeitet.

Entsprechend den möglichen Ereignisfolgen ergeben sich ständig neue Zustände. Die Steuerung dieses Ablaufs erfolgt über eine

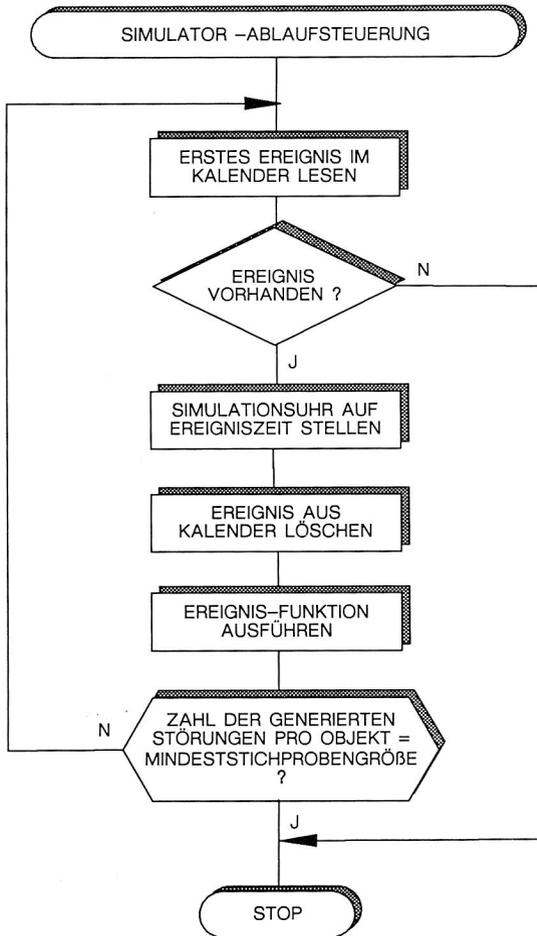


Bild 41: Ablaufsteuerung

Systemuhr. Erst wenn die vorgegebene Simulationsdauer erreicht ist, wird die Simulation beendet und eine Abschlußstatistik der Ergebnisse geliefert. Die Ausgabedaten für Station und Puffer sind in Bild 42 dargestellt. Zur Erläuterung der verwendeten Begriffe sei nochmals auf die Arbeit von REISCH [67] verwiesen.

STATION	PUFFER
-MITTLERE LAUFDAUER	-MITTLERE LAUFDAUER
-MITTLERE AUSFALLDAUER	-MITTLERE AUSFALLDAUER
-MITTLERE WARTEDAUER	-STATIONÄRE VERFÜGBARKEIT
-MITTLERE BLOCKIERDAUER	-MITTLERE BELEGUNG
-STATIONÄRE VERFÜGBARKEIT	-STANDARDABWEICHUNG
-NUTZUNGSGRAD	-FÜLLGRAD
-AUSLASTUNG	-ABWEICHUNGSGRAD
-MITTLERE VERWEILDAUER	-NICHTNUTZUNGSGRAD
-AUSBRINGUNG/TAG	-VOLLAUSNUTZUNGSGRAD

Bild 42: Kennzahlen von Station und Puffer

4.5 Die Wissenserwerbskomponente

4.5.1 Allgemeine Anforderungen

Im Rahmen des Wissenserwerbs muß neben der eigentlichen Wissenserfahrung auch der Aufbau neuer Wissensbasen und die Integration neuer Wissens Elemente in das System durchgeführt werden. Diese Aufgaben nimmt der Wissensingenieur wahr. Erst danach, wenn die Wissensbasen aufbereitet sind, kann der eigentliche Benutzer mit dem Expertensystem arbeiten. Die hier beschriebene Wissenserwerbskomponente soll den Wissensingenieur beim Aufbau und bei der Änderung der Wissensbasen unterstützen, sodaß eine möglichst komfortable interaktive Arbeitsweise möglich wird. Konzepte und Methoden zur automatischen und vom System selbsttätig durchzuführende Akquisition in Form von automatischen Lernprozessen werden nicht verfolgt.

Bei der Erstellung dieser Komponente wurde der Verfasser von Herrn cand. inf. H. Roßmeißl unterstützt.

Entsprechend den verwendeten Repräsentationsformen muß die Erwerbskomponente den Aufbau und die Modifikation von

- Frames,
- Produktionsregeln und
- Relationen

ermöglichen. Hierzu wurde ein entsprechender Editor geschaffen. Seine Benutzeroberfläche kann als Teil der Dialogkomponente des Gesamtsystems gesehen werden. Demzufolge gelten die allgemeinen Anforderungen an einen Benutzerdialog (vgl. Kap. 4.1.1) auch hier in gleicher Weise. Der Dialog mit dem Benutzer wird vornehmlich über Tastatur und Maus geführt. Dazu stehen verschiedene Masken und Befehlsmenüs zur Verfügung.

4.5.2 Der Editor

Der Editor selbst ist hierarchisch strukturiert und bietet dem Wissensingenieur auf der obersten Ebene die folgenden Funktionen zur Auswahl an:

- Erstellen von Frames und Framestrukturen,
- Eingabe von Produktionsregeln und Anbindung an Frames,
- Eingabe von Produktionsregeln und Aufbau von Regelbäumen,
- Definition von Relationen in Form von Horn-Klauseln,
- Ausdrucken von Wissensbereichen,
- Aktivierung der Problemlösungskomponente des Werkzeugsystems.

Der Editor stellt grundsätzlich die gesamte Informationsmenge auf einem Bildschirmmenü dar. Beim Arbeiten mit Objektframes handelt es sich um die jeweiligen Slotnamen und Slotinhalte, sowie die Struktur, in die der bearbeitete Rahmen eingefügt werden soll (Bild 43). Die Vererbungsrelationen innerhalb dieser Struktur werden automatisch angelegt. Innerhalb der Struktur kann man beliebig zu jedem gewünschten Frame springen.

Innerhalb eines Slots können die Facetten

- VALUE,
- DEFAULT und
- COMMENT,

sowie die Funktionen

- IF-NEEDED,
- IF-ADDED und
- IF-REMOVED

angelegt werden. Als Manipulationsfunktionen stehen zur Verfügung:

- EINFÜGEN,
- ÜBERSCHREIBEN,
- ÄNDERN,

Beim Editieren von Produktionsregeln und Regelbäumen verfährt man in der gleichen Weise (Bild 44). Auch hier wird der Regelinhalt, die einzelnen Regeln einer Gruppe und die verfügbaren Regelgruppen am Bildschirm dargestellt. Die Manipulationsfunktionen bleiben erhalten.

Zur Dokumentation der Akquisitionstätigkeiten und der Wissensinhalte leistet die Druckfunktion wertvolle Dienste. Mit ihrer Hilfe können die einzelnen Wissensbasen jederzeit ausgedruckt werden.

Die Arbeit des Wissensingenieurs wird vereinfacht, wenn er die Auswirkungen neuer Regeln und Frames sofort überprüfen kann. Zu diesem Zweck wurde ein direkter Zugang zur Problemlösungskomponente geschaffen. Diese kann dann einen definierten Wissensbereich abarbeiten. Auf diese Weise erhält der Wissensingenieur sofort Aufschluß darüber, ob seine Akquisitionstätigkeiten erfolgreich waren und zu einer weiteren Verbesserung des Systems geführt haben.

Ebene hoeher		cursor up		cursor up		exit			
Ebene tiefer		cursor down		cursor down					
----> Pfad: NODLAW GERAETE_VB HANDABDOMESERKRAFT MANUTECK3		----> Status: NAME TYP ACHSAMZAHL ARBEITSHOEHNE ARBEITSLAENGE KREFTSBREITTE PROFLAENDBREITF LAENGE BREITTE HOEHNE BEWEGUNGSBAHN GESCHWINDIGKEIT POSITIONIEREAKTIONSGRADE HANDABDOMESERKRAFT VERNACHLAESIGUNG KOSTEN LIEFERZEIT PERSONALBEDARF SENSODRUK STEUERUNG KREFTFEX ZUBEHOER		----> Statinhalt: ----> (\$VALUE 2488) (\$CONNEKT NR)		Slotwert: 2488		Slots	
----> springe zu Rahmen: ->		->		----> strukturi osen Slot: ->		Problemloeser		Regelbaum	
springen		einfuegen ueberschr		loeschen assoziativ		einfuegen ueberschreiblen		loeschen strukturi os	
								drucken	

Bild 43: Arbeiten mit dem Frame-Editor

<p>cursor up</p> <p>cursor down</p>	<p>cursor up</p> <p>cursor down</p>	<p>cursor up</p> <p>cursor down</p>	<p>exit</p>
<p>====> Regelgruppen:</p> <pre> MONTAGESTRUKTUR RATIONALISIERUNGSPOT ====> NEUREGELN TYP_WAHL_B TYP_WAHL_V TYP_WAHL_Z TYP_WAHL_Z GERAETE_WAHL_B GERAETE_WAHL_S GERAETE_WAHL_V GERAETE_WAHL_Z BLOCKWARTUNG SIMULATION OPTIMIERUNG KENNDATEI AUSGABE LAYOUT </pre>	<p>====> Regeln:</p> <pre> ====> NEI NEZ NEZ1 NETYP1 NETYP2 NETYP3 NETYP4 NEZEIT NEBLOCKING NESTRUKTION NEOPTIMIERUNG NEKENN1 NEKENN2 NEKENN3 NEKENN4 NEKUS1 NELAYOUT </pre>	<p>====> Regelinhalt:</p> <pre> Wenn: ====>BAUGRUPPEN ZBAU und: ==? MONTAGE_PRINZIP NIL dann: <KONTEXT> RATIONALISIERUNGSPOT -ZBAUGRUPPE ZBAU und: <INTERPRETER> RATIONALISIERUNGSPOT BEST-ONCE </pre>	<p>Slots</p>
<p>====> neue Regelgruppe:</p> <p>einfügen</p> <p>löschen</p> <p>überschreiben</p>	<p>====> neuer Regelname:</p> <p>einfügen</p> <p>löschen</p> <p>aufnehmen</p> <p>überschr</p>	<p>====> Regel editieren:</p> <p>einfügen</p> <p>löschen</p> <p>überschreiben</p> <p>strukturiertes</p>	<p>Regeln</p>
<p>====> neue Regelgruppe:</p> <p>einfügen</p> <p>löschen</p> <p>überschreiben</p>	<p>====> neuer Regelname:</p> <p>einfügen</p> <p>löschen</p> <p>aufnehmen</p> <p>überschr</p>	<p>====> Regel editieren:</p> <p>einfügen</p> <p>löschen</p> <p>überschreiben</p> <p>strukturiertes</p>	<p>Relationen</p>
<p>====> neue Regelgruppe:</p> <p>einfügen</p> <p>löschen</p> <p>überschreiben</p>	<p>====> neuer Regelname:</p> <p>einfügen</p> <p>löschen</p> <p>aufnehmen</p> <p>überschr</p>	<p>====> Regel editieren:</p> <p>einfügen</p> <p>löschen</p> <p>überschreiben</p> <p>strukturiertes</p>	<p>Rahmenbaum</p>
<p>====> neue Regelgruppe:</p> <p>einfügen</p> <p>löschen</p> <p>überschreiben</p>	<p>====> neuer Regelname:</p> <p>einfügen</p> <p>löschen</p> <p>aufnehmen</p> <p>überschr</p>	<p>====> Regel editieren:</p> <p>einfügen</p> <p>löschen</p> <p>überschreiben</p> <p>strukturiertes</p>	<p>Problemlaeser</p>
<p>====> neue Regelgruppe:</p> <p>einfügen</p> <p>löschen</p> <p>überschreiben</p>	<p>====> neuer Regelname:</p> <p>einfügen</p> <p>löschen</p> <p>aufnehmen</p> <p>überschr</p>	<p>====> Regel editieren:</p> <p>einfügen</p> <p>löschen</p> <p>überschreiben</p> <p>strukturiertes</p>	<p>drucken</p>

Bild 44: Arbeiten mit dem Regel-Editor

4.6 Die Erklärungskomponente

4.6.1 Grundsätzliche Möglichkeiten

Immer dann, wenn technische Systeme komplexe Aufgaben autark und selbständig ausführen, wünscht man sich Möglichkeiten zur Überwachung und Kontrolle dieser Abläufe. Auf wissensbasierte Systeme trifft diese Feststellung in gleicher Weise zu. In ihnen laufen selbständige Entscheidungsprozesse ab, die meist von einer Vielzahl von Randbedingungen und Einflußfaktoren bestimmt sind. Das Verhalten der Systeme ist nicht mehr auf den ersten Blick verständlich und plausibel. Es bedarf vielmehr einer Erläuterung.

Bei der Entwicklung von Expertensystemen wurde diese Problematik schon frühzeitig erkannt. Schon die ersten Systeme verfügten über Funktionen, die das Systemverhalten verdeutlichen sollten. Allerdings gab und gibt es noch heute gravierende Unterschiede, was die Funktionalität und den Komfort derartiger Erklärungskomponenten angeht. Die grundsätzlichen Möglichkeiten lassen sich auf zwei Fragen reduzieren:

- 1) Was wurde getan?
- 2) Warum wurde dies getan?

Zur Beantwortung der ersten Frage ist es notwendig, die Entscheidungsprozesse des Systems zu dokumentieren und darzustellen. Komfortable Systeme können mit entsprechender Unterstützung durch die Hardware (XEROX, SYMBOLICS) die Elemente der Wissensbasis am Bildschirm graphisch darstellen (Browser). Die Regeln und andere Wissensrepräsentationsformen werden mit ihrer Struktur in Form eines Baumes angezeigt. Eine Regelauswahl und Ausführung kann durch Invertieren des Graphiksymbols oder durch Blinken kenntlich gemacht werden.

Zur Beantwortung der zweiten Frage ist es darüberhinaus notwendig, alle Randbedingungen und Einflußfaktoren (Constraints) für die einzelnen Entscheidungsschritte zu kennen. Auch sie müssen deshalb während der Laufzeit des Systems festgehalten und doku-

mentiert werden. Komfortable Systeme unterscheiden sich hier von den einfacheren, indem sie ein natürlichsprachliches Frage-Antwort-Spiel mit dem Benutzer ermöglichen. Das System zeigt graphisch an, welche Regeln aktiviert und "gefeuert" wurden und auf welche Objekte dabei referenziert wurde. Auf die Benutzerfrage "Why" hin, erläutert das System in natürlichsprachlicher Form die Gründe, die zur Regelauswahl und Ausführung geführt haben.

4.6.2 Realisierte Erklärungsfunktionen

Der Schwerpunkt der vorliegenden Arbeit lag sicherlich in der Realisierung von Konzepten der Wissensrepräsentation und der Manipulation. Aufwendige Arbeiten an der Erklärungskomponente hätten den Rahmen der Arbeiten deutlich gesprengt und sollen deshalb zukünftigen Verbesserungen vorbehalten sein. Auch bietet die verwendete Hardware nicht die angesprochenen Graphikmöglichkeiten (Browser), um komfortable Darstellungen zu realisieren. Die Erklärungskomponente beschränkt sich auf die Beantwortung der Frage: Was wurde getan?

Hierzu wird der Ablauf der Planungsvorgänge in der Weise dokumentiert, daß die aktivierten Frames und instanziierten Regeln in alphanumerischer Form und in chronologischer Reihenfolge am Bildschirm zu sehen sind. In Abhängigkeit von der gewünschten Ausführlichkeit der Erklärung können verschiedene TRACE-Stufen gesetzt werden:

- 0: Nach jeder ausführbaren Regel den Regelinterpreter stoppen.
- 1: Nach jeder ausführbaren Regelgruppe den Regelinterpreter stoppen.
- 2: Alle Regeln ohne Stop abarbeiten.

Die Erklärungsfunktionen liefern dann Aufstellungen zu

- den passiven Regelgruppen,
- den aktiven Regelgruppen,

- den aktiven Einzelregeln der aktiven Regelgruppe,
- der Konfliktmenge der Regeln und den dabei relevanten Umgebungsbedingungen (Kontext),
- und der gültigen Regelstrategie.

Darüber hinaus wird die Auflösung der Konfliktmenge mit den dabei berücksichtigten Kriterien erläutert und dargestellt. Eine Aufstellung der durch die Regelausführung ausgelösten Aktionen mit der dabei relevanten Umgebung (Kontext) wird ebenfalls gegeben. Der Vorgang ist in Bild 45 dargestellt.

MOPLAN Regelinterpret		.../SCHLITZ/ME/10138
Passive Regelgruppen: (METAREGELN METAREGELN)	TRACESTUFE: 0	-look- **Frame: BGRU_LUEFTERMOTOR_L0090 18 KLASSE : BAUGRUPPEN NAME : LUEFTERMOTOR TEILEZAHL : NIL ZAHL_FS : NIL ZAHL_ZE : NIL MAX GEWICHT : NIL STEUERUNG : NIL NUTZUNGSGRAD : 90 KOSTEN : 300000 PERSONALBEDARF : NIL RAUMBEDARF : NIL STRUKTUR : NIL KONZEPT : NIL LEBENSDAUER : 60 BETRIEBSZEIT : 900 AUSBRINGUNG : 010 MONTAGE_PRINZIP : NIL MONTAGE_STRUKTUR : NIL
Aktive Regelgruppe: MONTAGESTRUKTUR	Aktiven Regeln von MONTAGESTRUKTUR : (RMS1 RMS2 RMS3 RMS4) Warte auf GO oder QUIT !	.../geyer/schraetche/10119 R 1 -look- **REGEL: ME2 wenn: BAUGRUPPEN ?BAU und: ?BAU =? MONTAGE_STRUKTUR _ dann: <KONTEXT> MONTAGESTRUKTUR =BAUGRUPPE ?BAU1 und: <INTERPRETER> MONTAGESTRUKTUR BEST-ONCE
ME2 : (BGRU_BUERSTENBAUGRUPPE_L0106) ME2 : (BGRU_LUEFTERMOTOR_L0090) ME1 : (BGRU_BUERSTENBAUGRUPPE_L0106) ME1 : (BGRU_LUEFTERMOTOR_L0090)	Auswahl nach Alter der Frames : ME2 ME2 ME1 ME1 Auswahl nach Laenge der Regeln : ME1 ME1 ME2 ME2 Auswahl nach Konstantenanzahl : ME2 ME2 ME1 ME1 Auswahl nach Alter der Regeln : ME1 ME1 ME2 ME2 nach Zufallswahl: ME2 Regel: ME2 wird ausgefuehrt mit : ((?BAU . BGRU_LUEFTERMOTOR_	
Aktion: neuer Kontext fuer MONTAGESTRUKTUR Umgebung: (=BAUGRUPPE)		
Aktion: - Regelinterpretaeraufruf -		

Bild 45: Beispiel für TRACE-Funktionen

Mit Hilfe einer speziellen LOOK-Funktion kann sich der Benutzer dann die einzelnen Regeln und Frames im Detail ansehen. Hierzu wählt er mit der Maus den gewünschten Namen aus. Sofort werden dann in zusätzlichen Fensterbereichen die ausführlichen Regeln und Frames dargestellt. Auf diese Weise wird der Benutzer in die Lage versetzt, den Ablauf der Problemlösungsprozesse schrittweise folgen zu können. Die Darstellung in Bild 46 zeigt einen Teil der Frame-Struktur der Wissensbasis sowie zwei mit LOOK geöffnete Frames.

The screenshot shows the MOPLAN software interface. The main window is titled 'MOPLAN' and contains a tree view of the knowledge base. Two sub-windows are open, displaying detailed information for selected frames.

Main Window (MOPLAN):

- Passive Regelgruppen: (METAREGELN)
- TRACESTUFE: /
- Aktive Regelgruppe: .../scratchtree/10220
- look- **Frame: BGRU_BUERSTENBAUGRUPPE_L0203 ist :

Sub-window 1 (Frame: BGRU_BUERSTENBAUGRUPPE_L0203):

ANLAGE	/ UWEI_G0230
UMWELT	\ UWEI_G0229
VERSORGUNG	/ VHYD_G0220
	VPNE_G0227
	VELE_G0226
	\ VELE_G0225
BAUGRUPPEN	/ BGRU_BUERSTENBAUGRUPPE_L0203
	\ BGRU_LUEFTERMOTOR_L0187

Sub-window 2 (Frame: LUEFTERMOTOR):

KLASSE	: BAUGRUPPEN
NAME	: LUEFTERMOTOR
TEILEZAHL	: 10
ZAHL_FS	: NIL
ZAHL_ZE	: NIL
MAX_GEWICHT	: NIL
STEUERUNG	: NIL
NUTZUNGSGRAD	: 90
KOSTEN	: 300000
PERSONALBEDARF	: NIL
RAUMBEDARF	: NIL
STRUKTUR	: NIL
KONZEPT	: NIL
LEBENSDAUER	: 60
BETRIEBSZEIT	: 900
AUSBRINGUNG	: 810
MONTAGE_PRINZIP	: MASCHINELL
MONTAGE_STRUKTUR	: NEST

Bild 46: TRACE-Funktion von Frames und Frame-Strukturen (Browser)

5. MOPLAN - Ein Expertensystem zur Planung von Montageanlagen

Die Entwicklung des Systems MOPLAN verfolgt im wesentlichen zwei Zielsetzungen. Einmal soll das wissensbasierte Werkzeugsystem (WWS) in seinen Eigenschaften erprobt und überprüft werden. Anhand der gewonnenen Erkenntnisse können daraus Maßnahmen

- zur Beseitigung von Fehlern,
- zur Erweiterung des Funktionsumfangs
- und zur Verbesserung des Laufzeitverhaltens (Performanz)

ergriffen werden. Die zweite wesentliche Zielsetzung war das Bestreben, für den Bereich der Montage Rationalisierungspotentiale zu erschließen. Während man in der spanenden Fertigung schon weitgehend über automatische Prozesse wie die NC- und DNC-Technik für Drehen, Bohren, Fräsen u. a. verfügt [68], ist der Automatisierungsgrad in der Montage augenblicklich noch deutlich geringer ausgeprägt. Dies liegt sicherlich auch in den hohen technischen Anforderungen begründet, die eine automatische Montage erfordert. Auch ist eine Standardisierung der Montagegeräte und deren Programmiermöglichkeiten praktisch nicht vorhanden. Während im Bereich der spanenden NC-Technik wichtige Standards wie die Normierung des NC-Codes und der CL-Data existieren, sind derartige Ansätze in der Montagetechnik derzeit noch nicht erkennbar.

In vielen Bereichen der industriellen Fertigung liegen die Montagekosten zwischen 50 Prozent und 75 Prozent der gesamten Produktionskosten. Dabei sind wesentliche Erfolge in der Produktivitätssteigerung nicht mehr zu erwarten. Die Montagetechnik wird damit zum "bedeutendsten Rationalisierungspotential der Zukunft" [69].

5.1 Aufgabenstellung

Eine wesentliche Voraussetzung für eine erfolgreiche Entwicklungsarbeit auf dem Gebiet der Expertensysteme ist die Konzentration auf ein abgeschlossenes und überschaubares Anwendungsge-

biet (Vgl. Kap. 3.5). Dies gilt für Softwareentwicklungen im universitären Umfeld noch viel mehr. Als erste Anwendung des wissensbasierten Werkzeugsystems (WWS) wurde deshalb die Planung von Montageanlagen in der Elektroindustrie gewählt. Als zusätzliche Eingrenzung ist die spezielle Ausrichtung auf die Großserienmontage kleinvolumiger Produkte zu sehen. Eine Ausdehnung auf andere Aufgabenbereiche ist nur mit quantitativen Wissenserweiterungen und nicht mit qualitativen Systemerweiterungen verbunden.

5.1.1 Struktur von Montageanlagen

Die Struktur von Montageanlagen orientiert sich sehr stark an der Struktur des zu montierenden Produktes. So kann man eine komplexe Anlage entsprechend den vorgegebenen Produktbaugruppen in eine End- oder auch Hauptmontage sowie in die jeweiligen Vor- oder Baugruppenmontagen untergliedern. Diese Baugruppen der Anlage bestehen dann aus einzelnen Zellen. Die Zellen können verschiedene Ausprägungen annehmen. Man unterscheidet hier zwischen manuellen und automatischen Montagekonzepten, wobei dann Handarbeits-

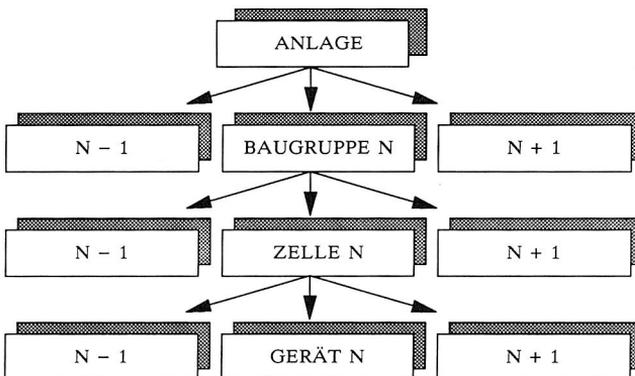


Bild 47: Struktur von Montageanlagen

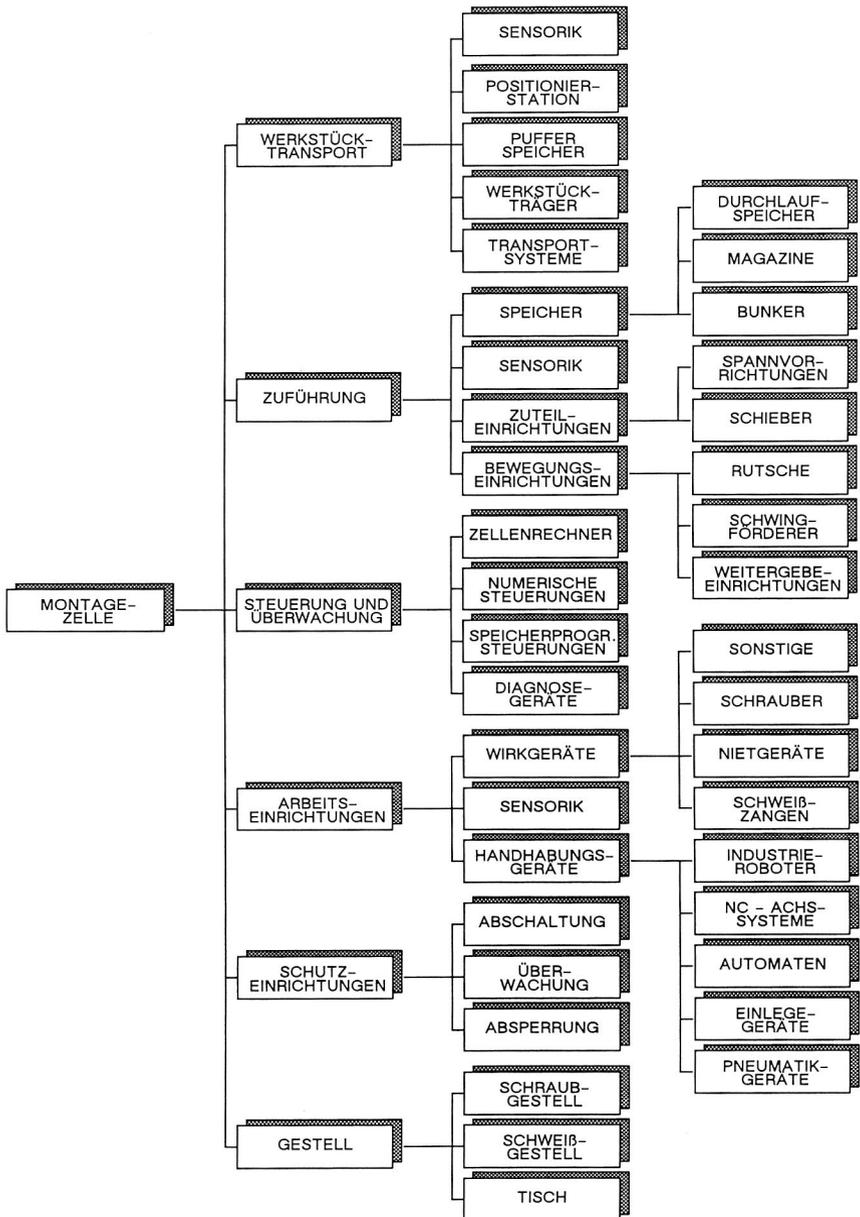


Bild 48: Geräte einer Montagezelle

plätze, Roboterarbeitsplätze und auch autarke Montagezellen anzutreffen sind. Die Verbindung der einzelnen Zellen innerhalb einer Baugruppe erfolgt über mehr oder weniger flexible Transportsysteme. Innerhalb einer Zelle unterscheidet man zwischen Geräten zum Handhaben, Zuführen, Steuern, Bearbeiten und Transportieren (Bild 48). Tragende Funktionen werden von diversen Gestellen übernommen. Beim Einsatz von Industrierobotern und anderen mit Gefahrenmomenten behafteten Geräten sind spezielle Schutzvorrichtungen notwendig. Sensoren arbeiten direkt mit den verschiedenen Geräten zusammen.

Es wird sofort deutlich, daß diese Geräte zu einzelnen Klassen zusammengefaßt werden können, die innerhalb einer Anlage spezielle Funktionen übernehmen. Innerhalb einer Klasse unterscheidet man weiter zwischen verschiedenen Gerätetypen. Diese Typen sind hauptsächlich durch ihre Bauform oder Wirkungsweise bestimmt. Für jeden Gerätetyp existiert eine Vielzahl von Geräteausprägungen oder auch Instanzen.

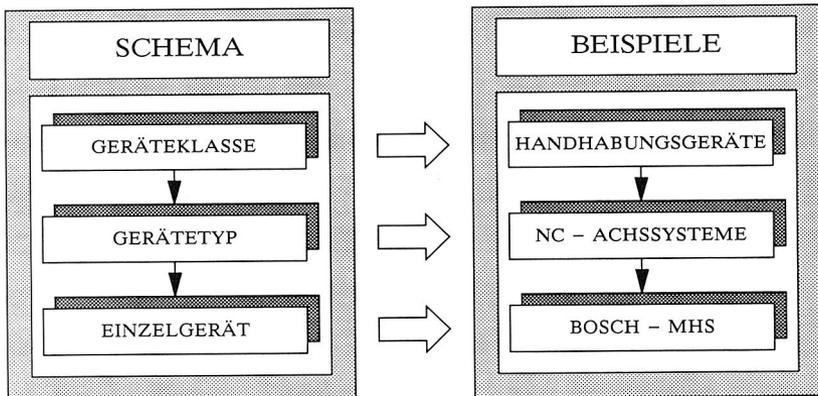


Bild 49: Geräteschema und Ausprägungen

Es ist leicht ersichtlich, daß auf diese Weise auch ein gewisses

Mengenproblem entsteht. Wenn man für jede Geräteklasse durchschnittlich nur fünf verschiedene Typen und für jeden Typ ebenfalls nur fünf Ausprägungen veranschlagt, so kommt man schon auf 25 Geräte je Klasse. Die Zahl der tatsächlich in der Praxis eingesetzten Geräte liegt sicherlich weit darüber.

5.1.2 Der Planungsablauf des Montageexperten

Die Planungsarbeiten des menschlichen Montageexperten laufen über mehrere Phasen ab. In vielen Fällen entsteht dabei die Situation, daß Entscheidungen einer Planungsphase erst über mehrere Folgephasen hinweg evaluiert werden müssen. Das heißt, der Experte bewegt sich über mehrere Phasen hinweg. Lösungsansätze werden vorgetragen und auf ihre Auswirkungen und ihre Machbarkeit hin untersucht. Damit wird der Planungsprozeß auch in einem hohen Maße iterativ. In vielen Fällen müssen vorausgegangene Entscheidungen aufgrund neu entstandener Umstände und Randbedingungen revidiert werden.

Die Gründe und Sachverhalte, die lokal innerhalb einer Planungsphase zu Entscheidungen führen, sind bekannt. Schwierig und nicht algorithmierbar dagegen wird es, diese Entscheidungen im globalen Zusammenhang der gesamten Montageanlage und des gesamten Planungsablaufes zu sehen.

Der Planungsablauf des Montageexperten gliedert sich einmal in Aufgaben zur Grobplanung und zum anderen in Aufgaben zur Feinplanung von Montagesystemen. Zur Feinplanung zählen beispielsweise die Festlegung der Einsteuerfolge von Werkstücken beim Anlagenstart oder bei einem Variantenwechsel und sämtliche Programmieraktivitäten. Die Feinplanung ist im Rahmen der vorliegenden Arbeit nicht von Interesse. Vielmehr sollen schwerpunktmäßig die Aufgaben der Grobplanung untersucht werden (Bild 50).

Der erste Schritt dieser Tätigkeiten ist eine Analyse der Montageaufgabe. In diesem frühen Stadium des Entwicklungsprozesses liegen von dem zu montierenden Produkt (Werkstück) in vielen

Fällen nur Konstruktionszeichnungen bzw. Prototypen vor. Hier ist

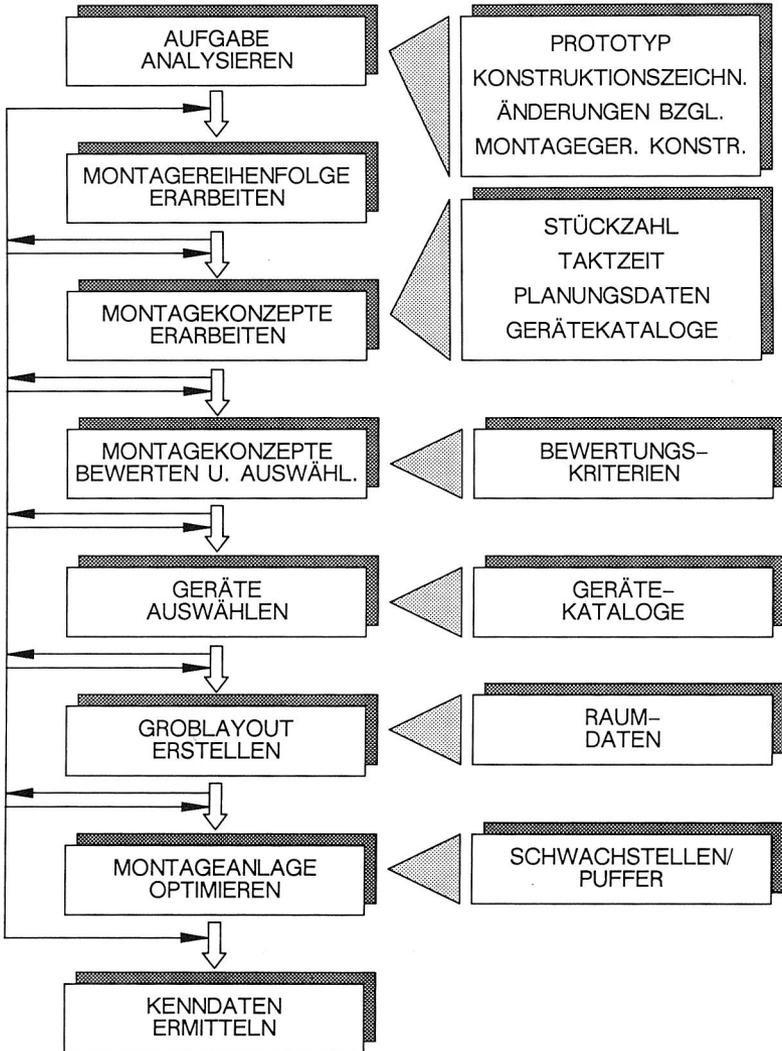


Bild 50: Planungstätigkeiten des Montageexperten (Grobplanung)

es die Aufgabe des Planers, konstruktive Änderungen am Werkstück zur Verbesserung der Montageeignung in Zusammenarbeit mit der Konstruktionsabteilung zu bewirken. Ferner ist die Automatisierbarkeit der Montagevorgänge abzuschätzen. Dazu können Untersuchungen von Handhabungs- und Fügevorgängen vorgenommen werden. Große Bedeutung gewinnt hierbei auch das Wissen des planenden Experten, der aufgrund seiner langjährigen Erfahrung aus ähnlich gelagerten Projekten Abschätzungen vornehmen kann.

In der nächsten Phase wird die Montageaufgabe in einzelne Teilschritte untergliedert und die Reihenfolge der dabei entstandenen Montagevorgänge festgelegt. Vielfach sind mehrere Lösungen möglich, sodaß eine endgültige Fixierung an dieser Stelle noch nicht vorgenommen werden kann.

Ausgehend von der so ermittelten Reihenfolge werden alternative Montagekonzepte aufgestellt. Die Einflußfaktoren hierfür sind die vorgegebenen Plandaten sowie die betrieblichen Randbedingungen. Die Konzepte unterscheiden sich in der Ablauffolge der Montageschritte, in der Struktur und Organisationsform der Anlagen sowie in dem Grad der Automatisierung. In manchen Fällen kann eine abschließende Bewertung und Konzeptauswahl erst erfolgen, wenn die Gerätelösungen ermittelt sind. Die verschiedenen Alternativen müssen also erst vollständig ausgearbeitet werden.

Die Geräteauswahl selbst hängt von einer Vielzahl von Kriterien ab, von denen nur die wichtigsten in Bild 51 dargestellt sind.

Zur graphischen Darstellung der Planungsergebnisse und zur Untersuchung der räumlichen Verhältnisse wird ein Groblayout der Anlage mit allen ihren Komponenten erstellt. Das Layout ist meist zweidimensional und schematisch. Entsprechende Erkenntnisse können auch hier noch zu Änderungen der Geräteauswahl, des Montagekonzeptes, ja sogar der Montagereihenfolge führen.

Nach dem Groblayout muß die Montageanlage in vielen Fällen noch optimiert werden. Dabei geht es darum, die Pufferelemente und Transportsysteme zwischen den einzelnen Stationen hinsichtlich

ihrer Kapazität zu dimensionieren. Das Leerlaufen oder Blockieren der Puffer muß vermieden werden.

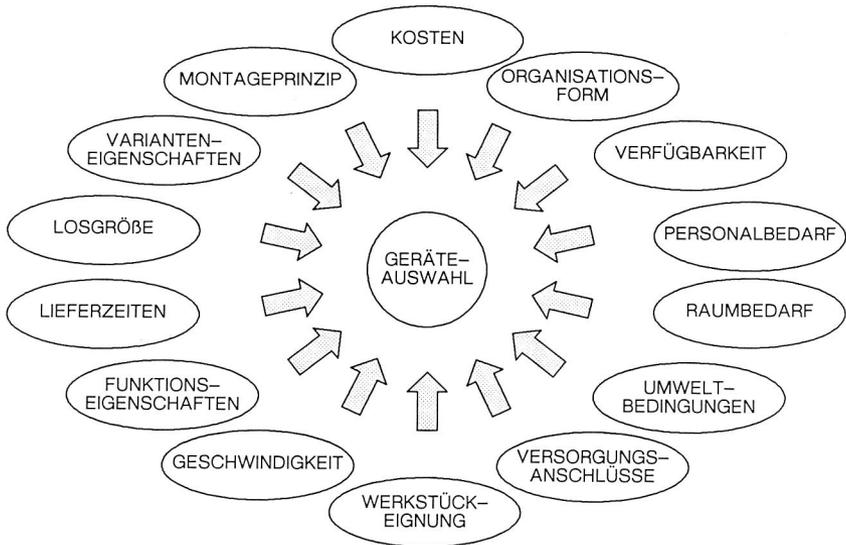


Bild 51: Einflussfaktoren bei der Geräteauswahl

Die Kenndaten einer Anlage sollen Aufschluß über die Güte der Planungsergebnisse geben. Im allgemeinen werden die entstandenen Kosten, verschiedene Wirtschaftlichkeitskennzahlen, der Personal- und Raumbedarf und nicht zuletzt die erzielte Verfügbarkeit der Anlage ermittelt.

Der beschriebene Planungsablauf wird in dieser Form von den Montageexperten der Praxis so durchgeführt. Die verfügbare Literatur bestätigt diese Vorgehensweise im wesentlichen, wobei es je nach Ausrichtung der Montagesysteme auch zu Schwerpunktsverlagerungen kommen kann. Zur weiteren Vertiefung sei auf diese Literatur [70] [71] [72] [73] [74] [75] verwiesen.

5.2 Beurteilung der Planungsaufgabe

5.2.1 Möglichkeiten konventioneller DV-Systeme

Für die in Kapitel 5.1 beschriebene Planungsaufgabe existiert bislang keine durchgängige rechnergestützte Lösung. Einzelne isoliert arbeitende Programmsysteme wurden nur für verschiedene Teilaufgaben entwickelt. Diese Systeme erfordern meist ein hohes Maß an Eingabeaufwand und behandeln die anfallenden numerischen Fragestellungen teilweise nur überschlägig. Dazu zählen Systeme

- zur Planung von Montageablaufstrukturen [76],
- zur Austaktung von Montageanlagen [77],
- zur Geräteauswahl und Betriebsmittelzuordnung [78],
- zur Ermittlung von Wirtschaftlichkeitskenndaten [79],
- zur Einsatzplanung von Industrierobotern [80],
- und zur Montageplanerstellung [81].

In vielen Fällen ist die Arbeitsweise interaktiv. Der Planer trifft Auswahl- und Konzeptentscheidungen und teilt sie dem Programmsystem über die Tastatur mit. Die Systeme übernehmen dann die Lösung der numerischen Probleme. Es liegt also eine klare Unterteilung in numerische Aufgaben und in intellektuelle Aufgaben vor.

Das neben CAD wohl wichtigste rechnergestützte Verfahren zur Planung von Montageanlagen ist die Simulation. Sie dient zur Optimierung der Pufferelemente und zur Ermittlung von Zuverlässigkeitskenndaten. Die ersten Programmsysteme wurden bereits Mitte der 70er Jahre entwickelt. Die bekanntesten gehen von einer vorgegebenen Pufferkapazität aus und simulieren dann die Pufferbelegung über einen vordefinierten Zeitraum [82] [83] [84] [85]. Verschiedene Kenndaten werden ausgegeben. Auch diesen Systemen ist eine klare Trennung von numerischen Verfahren und intellektuellen Entscheidungen eigen. Die Optimierung muß vom Planer ausgehen.

Unverhältnismäßig viel Aufwand wird in die Modellierung und Animation gesteckt. Da die Systeme isoliert arbeiten, muß das ge-

samte Anlagenabbild modelliert werden. Hierfür werden teilweise graphisch-interaktive Methoden, teilweise aber auch noch rein numerische Beschreibungsformen angeboten. Der Zeitaufwand für die Modellbildung durch den Benutzer ist größer als die eigentliche Dauer des Simulationsprozesses. Die Animation in Form von mehrfarbigen und pulsierenden Füllstandsanzeigen und Stationsbelegungen bringt keine neuen Erkenntnisse oder gar Optimierungen. Hier sind die statistischen Aussagen über einen längeren Zeitraum hinweg sehr viel aussagefähiger.

Zusammenfassend kann festgestellt werden, daß nur für isolierte numerische Teilprobleme rechnergestützte Lösungen erarbeitet wurden. Durchgängige Verfahrensketten über alle Teilprobleme hinweg sind nicht bekannt. Die Teillösungen sind darüber hinaus noch auf spezielle Typen von Montagesystemen beschränkt [79] [86]. Intellektuelle Entscheidungsprozesse werden immer dem Montageplaner überlassen.

In Folge des steigenden Rechneinsatzes auch in der Produktionsplanung wird es immer dringlicher, isolierte funktionale Aspekte hinter den Gesichtspunkten der Integration zurückzustellen. Die Zeit der automatisierten Insellösungen in der Datenverarbeitung sollte eigentlich vorbei sein. Softwarewerkzeuge, die nicht auf produktdefinierende und zentral verfügbare Daten zurückgreifen, können den notwendigen Integrationsanforderungen nicht genügen. Zur Verdeutlichung sei nochmals betont, daß diese Integrationsanforderungen ihren Ursprung

- in der Konsistenzgewährleistung,
- im Wartungsaufwand,
- im Modellierungsaufwand

und damit eigentlich in der Wirtschaftlichkeit von DV-Systemen haben. Legt man diese Maßstäbe an, so kann keine der oben aufgeführten Realisierungen bestehen. Keines der Teilsysteme ist in ein CIM-Konzept integrierbar, keines greift auf produktdefinierende Daten eines CAD-Systems oder eines Datenbanksystems zurück.

5.2.2 Einsatzmöglichkeiten wissensbasierter Systeme

Für die Entscheidung, wissensbasierte Konzepte und Lösungen für die Planungsaufgabe zu entwickeln, gab es mehrere Gründe (Bild 52). Dabei wurden die Kriterien für einen erfolgversprechenden Einsatz wissensbasierter Systeme so berücksichtigt, wie sie in Kapitel 3.5 aufgeführt sind.

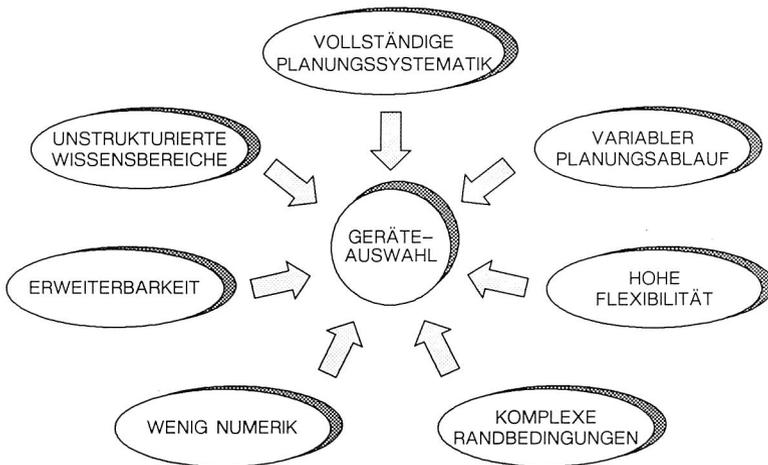


Bild 52: Gründe für den Einsatz von MOPLAN

So existiert eine vollständig beschreibbare Planungssystematik, ohne daß der eigentliche Ablauf der Planungsaktivitäten starr vorgegeben ist. Beispielsweise können Montagekonzepte teilweise erst dann bewertet werden, wenn die Gerätelösungen ermittelt und die Layoutuntersuchungen positiv verlaufen sind. Genausogut können lokale und auch globale Konzeptänderungen dadurch notwendig werden.

Das Planungsproblem selbst besteht einerseits aus numerischen

Verfahren und andererseits aus intellektuellen Entscheidungsprozessen. Einfache numerische Probleme lassen sich auch in einer KI-Programmiersprache wie LISP lösen. Komplexe Probleme, wie beispielsweise die Ermittlung der Verfahrszeiten von Industrierobotern, können auf einfache Darstellungen ohne Gültigkeitsverlust reduziert werden. Die tatsächlichen Verfahrszeiten lassen sich ohnehin nicht exakt angeben, da sie zu sehr von der real vorliegenden Bewegungsbahn und den instationären Bewegungsvorgängen unter Last abhängen.

Darüber hinaus existieren sogenannte vage Wissensbereiche was das Abschätzen der Automatisierbarkeit von Montage- und Zuführvorgängen angeht. Faustregeln und Erfahrungen aus zurückliegenden Projekten sind erforderlich und können in Regelform beschrieben werden.

Ein weiterer wesentlicher Gesichtspunkt sind die Einflußfaktoren zur Geräteauswahl (vgl. Kap. 5.1.). Betrachtet man jeden dieser Faktoren für sich alleine, so kann man durch einfache Zuordnung zu einer vertretbaren Geräteauswahl kommen. Im ganzen gesehen sind viele dieser Bedingungen jedoch widersprüchlich und konträr. Hier eine praktikable Abschätzung der Kriterien untereinander zu finden, ist mit numerischen Verfahren praktisch nicht möglich. Denn das Ziel des Planungsprozesses ist es ja, eine Anlage zu konfigurieren, die eine möglichst hohe technische Verfügbarkeit aufweist, wenig Personal bindet, minimale Kosten verursacht und möglichst wenig Raum beansprucht. Darüber hinaus muß natürlich die geforderte Systemleistung (Ausbringung) gewährleistet sein.

Der Planungsprozeß selbst ist im wesentlichen ein datengetriebener Vorgang. Ausgehend von definierten funktionalen Anforderungen werden Gerätelösungen Schritt um Schritt konkretisiert. Für solche Aufgaben eignen sich vorwärtsverkettende Inferenzmechanismen im Rahmen von regelbasierten Systemen (Produktionssysteme). Als herausragendes Beispiel sei hier das System XCON (vgl. Kap.3) angeführt.

Der letzte und vielleicht wichtigste Grund für den Vorzug eines

wissensbasierten Systems gegenüber einem konventionell numerischen ist die notwendige Flexibilität. Sie ist erforderlich, weil Montageanlagen in hohem Maße

- vom Produkt,
- vom Unternehmen und
- vom Planer

abhängig sind. Die Einflußfaktoren, die vom Werkstück und den geforderten Handhabungsfunktionen herrühren, sind so schwerwiegend, daß sich Anlagen in unterschiedlichen Industriezweigen gravierend unterscheiden. Und da sich viele Unternehmen über Jahre hinweg ein spezielles Know-how erarbeitet haben, unterscheiden sich die Anlagen auch innerhalb einer Branche. Weil dieses Firmen-Know-how meist nur in den Köpfen einiger weniger aber erfahrener Planer konzentriert ist, sind Expertensysteme auch ein Instrument solches Wissen zu sichern und zu transferieren.

Die Flexibilität regelbasierter Systeme zeigt sich auch in der Erweiterbarkeit der Wissensbasis. Man kann hier mit einem eng umgrenzten Problemfeld wie beispielsweise mit Roboterarbeitsplätzen oder mit manuellen Arbeitsplätzen beginnen. Durch Erweiterung der Wissensbasis werden dann neue Aufgabenbereiche wie beispielsweise die Planung von autarken Montagezellen erschlossen. Diese Aussage steht nach Meinung des Verfassers auch nicht mit der Erfahrung im Widerspruch, daß eines der ersten medizinischen Expertensysteme, MYCIN [21], inzwischen nicht mehr betrieben wird, weil die Pflege der Wissensbasis zu aufwendig war. Konventionelle Programmsysteme konnten ja die Aufgaben von MYCIN gar nicht erst lösen. Daß das System trotzdem aufgegeben werden mußte, lag sicherlich an seiner Architektur und dem allzu raschen Wandel des Domänenwissens und nicht an generell mangelnder Flexibilität des regelbasierten Ansatzes.

5.3 Der Planungsablauf mit dem Expertensystem

Das Expertensystem MOPLAN deckt in seiner Funktionalität einen weiten Bereich der Planungstätigkeiten bei der Grobplanung von Montageanlagen ab. Die einzelnen Aktivitäten sind:

- Montagekonzept erarbeiten,
- Geräte auswählen,
- Groblayout erstellen,
- Einsatzverhalten simulieren,
- Montageanlage optimieren,
- Kenndaten der Anlage ermitteln.

Beim Starten des Systems erscheint das Eingangsmenü von MOPLAN auf dem Bildschirm (Bild 53). Die einzelnen Menüfelder orientieren sich an der Struktur des Systems. Der Benutzer kann nun auswählen, welche Tätigkeiten er durchführen will.

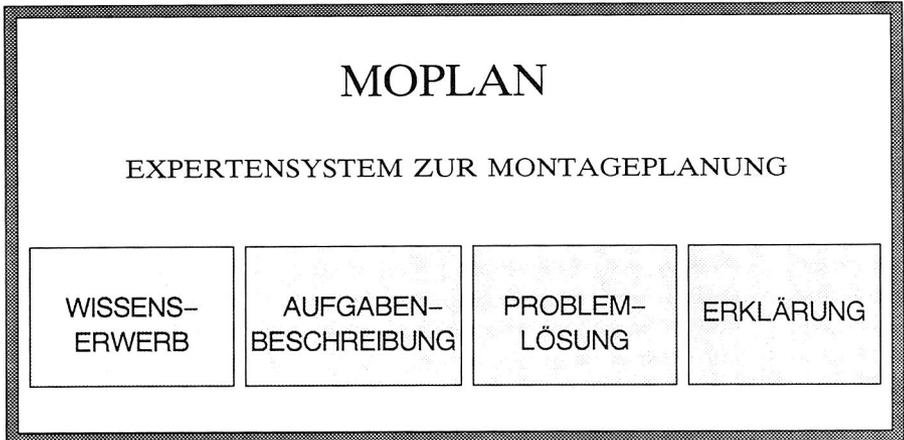


Bild 53: Eingangsmenü von MOPLAN

Bevor überhaupt eine Aufgabe bearbeitet werden kann, müssen natürlich die entsprechenden Wissensbasen angelegt sein. Dies gehört zum Aufgabengebiet des Wissensingenieurs. Er benutzt dazu die Wissenserwerbskomponente.

Der eigentliche Benutzer von MOPLAN wird im Anschluß daran die zu lösende Planungsaufgabe beschreiben und abspeichern. Auf diese Beschreibung kann dann jederzeit wieder zurückgegriffen werden.

Zur Aktivierung der Problemlösungskomponente und der Erklärungsfunktionen sind ebenfalls gesonderte Menüfelder vorhanden.

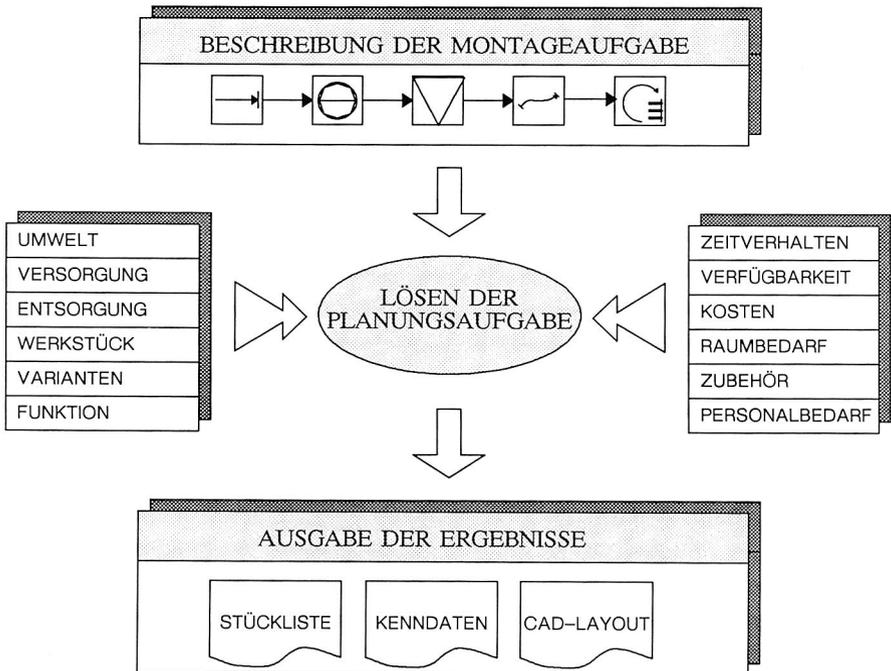


Bild 54: Globaler Planungsablauf von MOPLAN

Im folgenden soll nun die Arbeitsweise mit dem Expertensystem genauer beschrieben werden. Es lassen sich schwerpunktmäßig drei Phasen herausstellen (Bild 54):

- Beschreibung der Montageaufgabe,
- Bearbeitung und Lösung der Montageaufgabe,
- Ausgabe der Ergebnisse.

5.3.1 Beschreibung der Montageaufgabe

Es ist eine altbekannte Tatsache, daß die Ergebnisse eines Computerprogramms qualitativ besser und detaillierter sind, je umfangreicher und genauer das zur Verfügung gestellte Datenmaterial war. Auf der anderen Seite werden Programmsysteme aber auch zunehmend unhandlicher, je langwieriger sich der Prozeß der Dateneingabe gestaltet. Die Akzeptanz solcher Programme ist gering, was zur Folge haben kann, daß funktional hochwertige Programme aufgrund mangelndem Eingabekomfort nicht eingesetzt werden. Die Kunst besteht nun eben darin, ein Optimum als Kompromiß zu finden, das die folgenden zwei Aussagen auf sich vereint:

- So viel Benutzereingabe wie nötig
- So wenig Benutzereingabe wie möglich

Für das Expertensystem MOPLAN gelten diese Aussagen in gleicher Weise. Die Beschreibung der Montageaufgabe erfolgt interaktiv durch den Benutzer. Hierfür steht ihm die Dialogkomponente des wissensbasierten Werkzeugsystems (WWS) zur Verfügung. Die Eingaben untergliedern sich grob in

- allgemeine Randbedingungen und in
- spezielle Informationen zu den einzelnen Montageschritten.

Diese Montageschritte werden im folgenden "Funktionsschritte" genannt (Bild 55). Alle Angaben werden von den Inferenzprozessen während der Problemlösung benötigt.

Die allgemeinen Randbedingungen der Montageanlage werden zu Beginn des Dialogs vom System abgefragt. Neben den Bezeichnungen für die geplante Anlage mit ihren Baugruppen sind dies die veranschlagte Lebensdauer, die tägliche Betriebszeit, die geforderte tägliche Ausbringung und der geplante Nutzungsgrad. Angaben zur Versorgung (Elektro, Pneumatik, Hydraulik), zu den vorliegenden Umwelteinflüssen (Temperatur, Gase, Staub, Säure, Feuchtigkeit, Erschütterungen) und zu den Raumverhältnissen (Länge, Breite, Höhe, Deckenbelastung) schließen sich an. Für die einzelnen Baugruppen können davon abweichende Angaben notwendig werden, wenn die Struktur der Anlage und der Randbedingungen dies erfordert. Damit ist die Eingabe der allgemeinen Randbedingungen abgeschlossen.

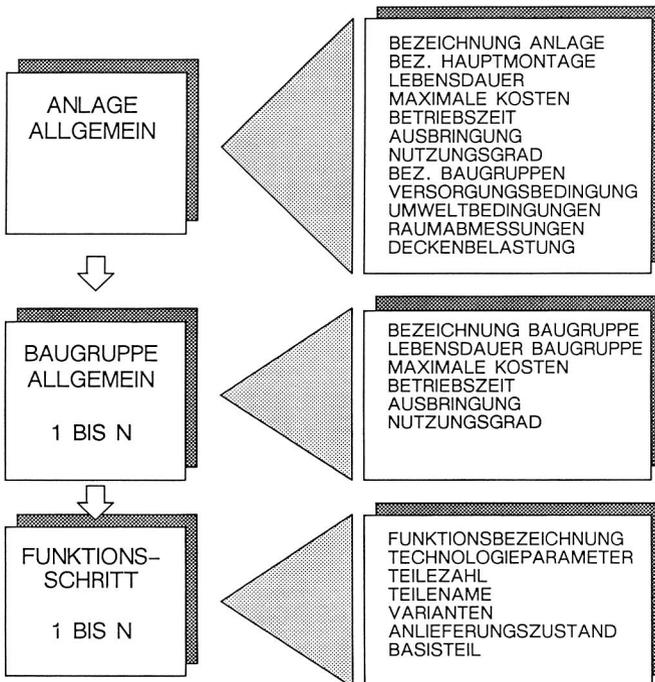


Bild 55: Eingaben zur Beschreibung der Montageaufgabe

Es folgt nun die eigentliche Spezifizierung der Montagefolge. Das System geht hierzu die einzelnen Baugruppen der Anlage nacheinander durch. Eine zusammenhängende Reihenfolge entsteht durch Aneinanderreihen der einzelnen Funktionsschritte. Für jeden Funktionsschritt muß ein entsprechendes Menüfeld angesprochen werden. Zur Spezifikation eines Funktionsschrittes bietet das System Ikonen für die Bereiche

- Handhaben,
- Kontrollieren,
- Lagern,
- Formgeben,
- Formändern,
- Behandeln und
- Fügen

an (Bild 56). Diese Aufteilung mit den dazugehörigen Symbolen entspricht der DIN 8593 [87] und der VDI-Richtlinie 2860 [88]. Der Benutzer kann nun jede gewünschte Funktion definieren. Möchte er auf standardisierte Unterfunktionen des Handhabens und Kontrollierens zurückgreifen, so stehen ihm in zwei weiteren pop-up-windows diese Unterfunktionen zur Verfügung (Bilder 57 und 58) .

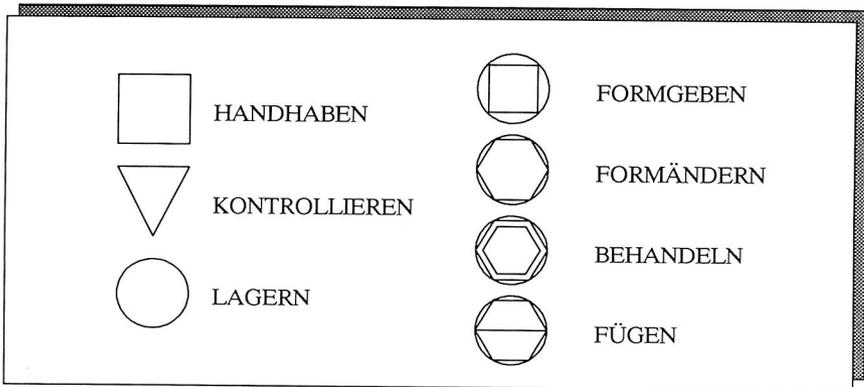


Bild 56: Ikonen zur Beschreibung der Funktionsschritte

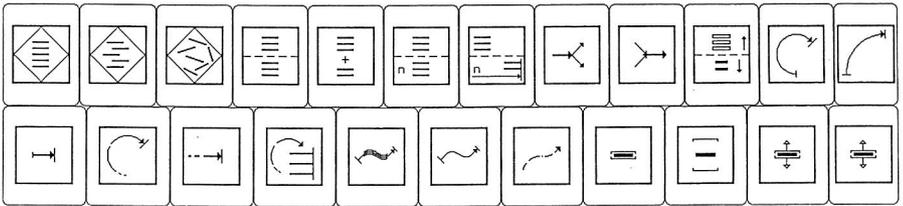


Bild 57: Vordefinierte Teilfunktionen "Handhaben"

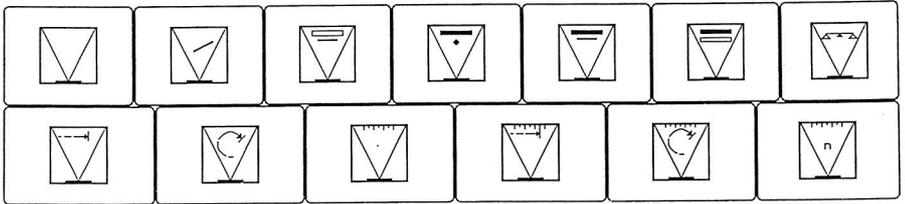


Bild 58: Vordefinierte Teilfunktionen "Kontrollieren"

Zur weiteren Spezifikation eines einzelnen Funktionsschrittes baut das System ein spezielles popup-window auf. Angaben zur Zahl und Bezeichnung der Zuführteile, zu ihren Varianten, zum Anlieferungszustand und zu speziellen Technologieparametern, wie beispielsweise DREHWINKEL oder FÜGEKRAFT, werden über eine entsprechende Maske abgefragt.

Neben den beschriebenen Generierungsfunktionen sind auch Manipulationsfunktionen zum

- Löschen,
- Einfügen,
- Ändern und zur
- Beendigung der Benutzereingabe

implementiert (Bild 59). So können in beliebiger Weise ganze Funktionsschritte gelöscht oder eingefügt werden, ohne daß die

Montagefolge ihre sequentielle Ordnung verliert. Einzelangaben innerhalb eines Funktionsschrittes sind ebenfalls korrigierbar.

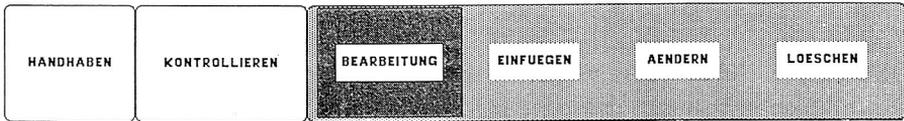


Bild 59: Menü für die Manipulationsfunktionen

Sobald nun alle Eingaben zu den Baugruppen und Funktionsschritten vorgenommen sind, ist die Montageaufgabe vollständig beschrieben. Die einzelnen Funktionsschritte werden mit ihren Zusatzangaben am Bildschirm sequentiell dargestellt. Zu Dokumentationszwecken kann diese Darstellung auch über Drucker oder Hardcopy ausgegeben werden.

5.3.2 Lösen der Montageaufgabe

Das Planen und Konfigurieren von Montageanlagen ist kein geschlossen algorithmierbarer Prozeß, sondern wird von intellektuellen Entscheidungen über mehrere Planungsphasen hinweg geprägt. Demzufolge können an dieser Stelle auch nur die jeweiligen Aktionen der Problemlösungskomponente beschrieben werden, wie sie zur Lösung der Planungsaufgabe notwendig sind. Nicht beschrieben werden kann dagegen die genaue Ablauffolge der Aktionen, weil diese von den ausgewählten Regeln bestimmt wird.

5.3.2.1 Überprüfung des Rationalisierungspotentials

Im Rahmen der Konzeptfindung fällt als erstes die Frage an, inwieweit für die Montageaufgabe eine automatisierte Lösung in Erwägung zu ziehen ist, oder ob manuelle Konzepte vorzuziehen

sind. Die ausschlaggebenden Faktoren hierfür sind:

- veranschlagte Lebensdauer der Anlage,
- tägliche Betriebszeit,
- störende Umwelteinflüsse,
- geforderte Ausbringung,
- Komplexitätsgrad des Montagevorganges.

Ohne eine fundierte Wirtschaftlichkeitsrechnung durchzuführen, kann man hier schon eine grobe Abschätzung vornehmen. Der Experte ist ferner in der Lage, die prinzipielle technische Notwendigkeit und Realisierbarkeit zu beurteilen. Dieses Wissen liegt in Form von Faustregeln vor und kann in einfacher Weise auf Produktionsregeln abgebildet werden.

So spricht eine lange Lebensdauer der Anlage, ein täglicher Dreischichtbetrieb oder schädliche Umwelteinflüsse eindeutig für automatisierte Lösungen. Das Rationalisierungspotential ist entsprechend hoch einzuschätzen. Für eine manuelle Montage würde sprechen, wenn es sich um ein Produkt mit absehbar geringer Lebensdauer und niedrigen Stückzahlen handelt. Dies trifft meist auf Varianten oder "Exoten" zu.

Zur Entscheidungsfindung in dieser Frage greift das System auf die eingegebenen Allgmeindaten der Anlage zurück.

5.3.2.2 Festlegung der Montagestruktur

In der Montagetechnik findet man eine Vielzahl alternativer Organisationsformen, die sich primär durch die Aufteilung der Arbeitsinhalte und die Bewegungen der Montageobjekte unterscheiden [89]. Für den ausgewählten Wissensbereich sind im wesentlichen zwei Organisationsformen mit alternativen Ausprägungen von Bedeutung:

- die Linienmontage (Bild 60),
- die Nestmontage (Bild 61).

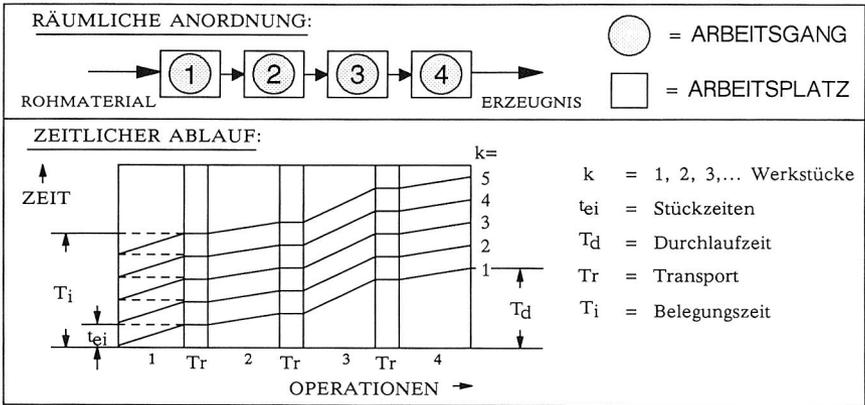


Bild 60: Fertigungsprinzip "Linienfertigung" (Quelle: [89])

Bei der Linienmontage werden die Arbeitsinhalte entsprechend der Montagereihenfolge auf verschiedene Stationen aufgeteilt. Die Ausprägungen derartiger Stationen können automatische Handhabungs- und Bearbeitungszellen oder auch manuelle Arbeitsplätze sein. Das zu montierende Werkstück wird über ein mehr oder weniger flexibles Transportsystem zu den jeweiligen Stationen gebracht.

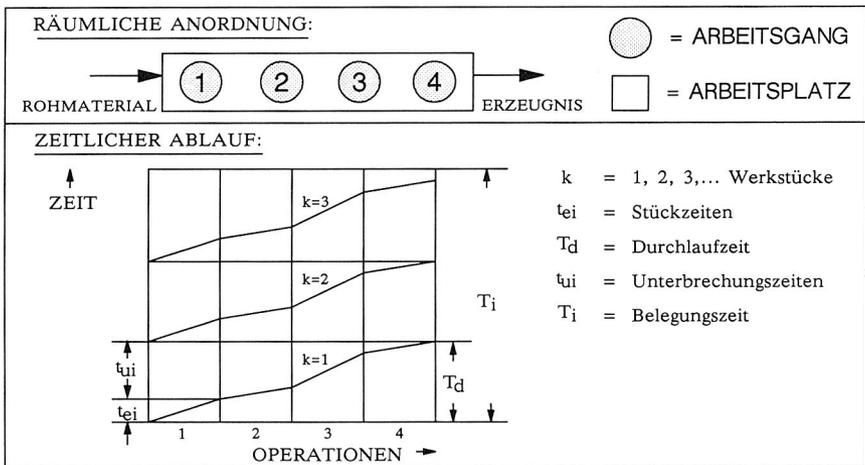


Bild 61: Fertigungsprinzip "Nestfertigung" (Quelle: [89])

Bei der Nestfertigung werden mehrere Montageschritte, wie beispielsweise die Montage einer ganzen Baugruppe, auf einer einzigen Station durchgeführt. Diese Station kann dann ebenfalls entweder Roboterzelle, Bearbeitungsstation oder auch Handarbeitsplatz sein. Erst nach Abschluß aller Montagevorgänge erfolgt der Weitertransport zur nächsten Station.

Die Einflußfaktoren für diese Strukturfestlegung sind:

- die geforderte Ausbringung,
- die Anzahl der Montageschritte je Baugruppe,
- die Anzahl der Zuführteile je Baugruppe,
- die Zeitdauer der einzelnen Montageschritte im Verhältnis zur Gesamttaktzeit der Baugruppe.

Der Trend geht heute hin zu verketteten Montagestationen, wenn die geforderte Ausbringung hoch ist und die geschätzten Taktzeiten dadurch sehr kurz werden. Das Gleiche gilt, wenn die Zahl der einzelnen Montageschritte und die Zahl der Zuführteile groß ist. Bei einer Nestmontage wäre dann einfach nicht genügend Raum für die notwendigen Zuführgeräte vorhanden.

Von der ausgewählten Organisationsform ist die Auswahl der Handhabungs- und Transporteinrichtungen extrem abhängig. So sind für das Linienkonzept beispielsweise die sogenannten NC-Achsen vorzuziehen, während eine Komplettmontage in vielen Fällen einen Werker oder einen Industrieroboter in Gelenkbauweise erforderlich macht.

Aufgrund der Eingabewerte ermittelt das Regelwerk eine Organisationsform, die dem Benutzer dann auch mitgeteilt wird. Dieser kann dann darüber entscheiden, ob er den Vorschlag akzeptiert oder die entsprechende Alternative bearbeitet haben möchte. Auf diese Weise können auch alternative Montagekonzepte verfolgt werden. Der gleiche Fall tritt ein, wenn die Reihenfolge der Montageschritte geändert würde. Auch dann können alternative Konzept bearbeitet werden.

5.3.2.3 Funktionskomplexe und Typauswahl

Zur Strukturierung der Gerätevielfalt wurde eine Zusammenfassung der Montageeinrichtungen nach funktionalen Gesichtspunkten vorgenommen. Es werden vier derartige Funktionskomplexe unterschieden:

- BEARBEITEN,
- ZUFÜHREN,
- VERBINDEN,
- STEuern.

Sämtliche Geräte einer Montageanlage können einem dieser vier Funktionskomplexe zugeordnet werden. Gleichzeitig benötigt eine Montageanlage immer Geräte aus jedem dieser Funktionskomplexe, um überhaupt funktionieren zu können. Die Geräteauswahl selbst reduziert sich also auf eine Auswahl zu den einzelnen Funktionskomplexen. Die Zuordnung der einzelnen Geräte zu den jeweiligen Funktionskomplexen wird in den nachfolgenden Abschnitten beschrieben.

Vor der eigentlichen Geräteauswahl muß die Zahl der in Frage kommenden Lösungen aus Effizienzgründen weiter reduziert werden. Bei einem Suchraum von vielleicht 300 bis 400 Geräten und bei ca. 10 Prüfungskriterien würde eine kombinatorische Explosion der Inferenzprozesse (ca. 4000) praktisch zur Ineffizienz des Systems führen. Diese Reduktion erfolgt zweckmäßigerweise über eine Selektion derjenigen Gerätetypen, welche überhaupt für die gestellten Anforderungen in Frage kommen. Bei den Handhabungsgeräten unterscheidet man beispielsweise acht verschiedene Typen:

- kartesische Industrieroboter,
- zylindrische Industrieroboter,
- sphärische Industrieroboter,
- Montageroboter (Scara-Bauweise),
- numerisch gesteuerte Achssysteme (NC-Achsen),
- pneumatisch betriebene Handhabungsgeräte,
- mechanisch gesteuerte Einlegegeräte,
- starre Sonderautomaten.

Diese Typselektion wird über spezielle Produktionsregeln der Operationswissensbasis vorgenommen. Die Kriterien hierfür entsprechen denen der eigentlichen Geräteauswahl. Es entsteht also keine qualitative Veränderung der Problemstellung, sondern nur eine quantitative Eingrenzung des Suchraums.

5.3.2.4 Geräteauswahl

Die Auswahl der Geräteausprägungen erfolgt getrennt nach den einzelnen Funktionskomplexen. Dies wird notwendig, weil es keinen allgemeingültigen Kriterienkatalog gibt, der für alle existierenden Geräte verwendbar ist. Demzufolge müssen zu jedem Funktionskomplex spezielle Auswahlkriterien mit einer spezifischen Gewichtung aufgestellt werden.

Funktionskomplex BEARBEITEN

Die Geräte dieses Komplexes haben die Aufgabe, alle Funktionen der Werkstückbearbeitung und Werkstückhandhabung durchzuführen. Demnach fallen hierunter alle Geräte zum

- Handhaben,
- Kontrollieren,
- Formgeben,
- Formändern,
- Behandeln und
- Fügen.

Die Geräteauswahl läuft in zwei Phasen ab. Im Rahmen der ersten Phase muß eine Überprüfung verschiedener Eignungskriterien erfolgen. Diese Bedingungen müssen alle erfüllt sein. Man kann hier also zu klaren Ja/Nein-Entscheidungen kommen. Die Menge der in Frage kommenden Objekte umfaßt somit alle Geräte, die diese Eignungskriterien erfüllen.

Die Kriterien dieser ersten Phase sind:

- funktionale Eignung,
- Umwelteignung,
- Werkstückeignung,
- Varianteneignung,
- Versorgungseignung.

Bei der Prüfung der funktionalen Eignung ist von Interesse, inwieweit ein Handhabungsgerät aufgrund seiner Kinematik und Steuerung überhaupt in der Lage ist, die Aufgabe auszuführen. Die Komplexität und der Schwierigkeitsgrad der Handhabungsfunktionen sind hierbei entscheidend. Im Rahmen eines Expertensystems sind natürlich nur überschlägige Abschätzungen möglich. In der Praxis werden in solchen Fällen spezielle Fügeuntersuchungen entweder an einem Laboraufbau oder am CAD-System vorgenommen. Bei Bearbeitungsgeräten, wie beispielsweise Magnetisier- oder Einpreßvorrichtungen, gestaltet sich diese Prüfung einfacher, da dort die Kinematik nur eine untergeordnete Bedeutung hat.

Im Zuge einer Prüfung der Umwelteignung wird festgestellt, ob das Gerät gegen Erschütterungen oder sonstige chemische und physikalische Einflüsse empfindlich ist. Hier werden die entsprechenden Umweltdaten aus der Benutzereingabe benötigt.

Die Werkstückeignung wird im wesentlichen vom Teilgewicht und von der Teilegröße beeinflusst. Hier sind die jeweiligen Greiferkonstruktionen mit zu berücksichtigen. Die Information zu den Handhabungsteilen stammt aus der Teilwissensbasis.

Varianten des Handhabungsteiles sind in der Teilwissensbasis mit ihren Eigenschaften ebenfalls erfaßt. Auf diese Weise wird auch eine Prüfung der Werkstückeigenschaften von Variantenteilen ermöglicht.

Die Versorgung mit Elektrik und Pneumatik sollte eigentlich immer gewährleistet sein. Geprüft werden Anschlußspannung, max. Stromstärke, Luftdruck und Luftdurchsatz. Die Hydraulik ist ebenfalls

durch Druck und Durchsatz gekennzeichnet.

Mit Hilfe der genannten Prüfungen läßt sich die Zahl der geeigneten Geräte zwar weiter einschränken, aber für die letztendliche Geräteauswahl sind noch zusätzliche Untersuchungen notwendig. So gilt es, aus dem Kreis der noch verbliebenen Geräte das möglichst beste und optimal geeignete Gerät zu finden. Die Kriterien hierfür sind:

- technische Verfügbarkeit,
- benötigter Raum,
- entstehende Kosten,
- notwendige Zusatzausrüstung,
- benötigtes Personal,
- Zeitverhalten.

Es ist sofort einleuchtend, daß diese Fragen nicht im Stile von sogenannten ko-Kriterien mit "Ja" oder "Nein" beantwortet werden können. Die Aufgabe besteht vielmehr darin, entsprechende Kennwerte zu ermitteln und zueinander in Bezug zu setzen. Das optimal geeignete Gerät muß durch Abwägen der Kenngrößen gegeneinander gefunden werden. Erschwerend kommt hinzu, daß die Kenngrößen in ihrer Tendenz teilweise gegenläufig sind. So ist beispielsweise eine höhere technische Verfügbarkeit in vielen Fällen mit höheren Kosten, mehr Zusatzausrüstung und vielleicht auch mit höherem Personalaufwand verbunden. Die Lösung dieses Problems kann nur eine Kompromißentscheidung sein. Solche Entscheidungsprozesse sind nicht algorithmierbar. Nur ein entsprechend formuliertes Regelwerk kann hier zu fundierten Lösungen kommen.

Auch schon die Ermittlung der Kenngrößen gestaltet sich nicht gerade einfach. Die technische Verfügbarkeit kann nur im Zusammenhang mit dem zu handhabenden Werkstück in der tatsächlich vorliegenden Geräteumgebung abgeschätzt werden. Verlässliche Daten liegen im Planungsstadium selten vor. Der Raumbedarf, die Investitions- und Folgekosten sowie der Personalbedarf sind dagegen fest angebbar. Allerdings müssen auch die Zusatzausrüstungen noch berücksichtigt werden. Das Zeitverhalten selbst ist wieder direkt

vom Geschwindigkeitsverhalten des Gerätes und von der Art und Länge der Bewegungsbahn abhängig. Hier können nur grobe Näherungen berücksichtigt werden.

Die letztendliche Geräteauswahl erfolgt regelbasiert durch Abwägen der Einflußfaktoren gegeneinander.

Funktionskomplex ZUFÜHREN

Zu diesem Funktionskomplex werden die folgenden Geräte gezählt:

- Speichereinrichtungen,
- Ordnungseinrichtungen,
- Zuteileinrichtungen,
- Vereinzelungseinrichtungen.

Wichtig ist bei dieser Betrachtungsweise, daß die Geräte nicht im direkten Materialfluß der Montageanlage liegen, sondern die Zuführung von neuen Werkstücken in diese Hauptflußrichtung bewirken sollen. Die Einflußfaktoren für eine Geräteauswahl sind:

- Eigenschaften des Zuführteiles,
- Anlieferungszustand des Zuführteiles,
- geforderte Taktzeit der Anlage,
- geforderte Position und Orientierung des Zuführteiles für nachfolgende Handhabungsfunktionen.

Zur Eigenschaftsbeschreibung von Werkstücken wird eigens eine TEILEWISSENSBASIS organisiert (vgl. Kap. 5.4.2). In dieser TEILEWISSENSBASIS ist auch das Wissen über die Zuführteile abgelegt. Es besteht aus den folgenden Angaben:

- Teiletyp,
- Teilegröße,
- Teilegewicht,
- physikalische Eigenschaften,
- Lager- und Transporteigenschaften.

Der Anlieferungszustand des Zuführteiles und die geforderte Taktzeit der Anlage sind bekannte Größen. Sie werden vom Benutzer im Zuge der Beschreibung der Montageaufgabe eingegeben.

Dagegen bereitet es deutlich mehr Aufwand, die geforderte Position und Orientierung des Zuführteiles festzustellen. Die beiden Größen hängen wesentlich vom Typ des ausgewählten Handhabungsgerätes ab. Auch ist das Funktionsprinzip des eingesetzten Greifers und die Ausbildung seiner Greifflächen von großem Einfluß.

Das Finden geeigneter Zuführlösungen ist von weitreichender Bedeutung für die Funktionsweise der Montageanlage. In vielen Fällen können automatische Handhabungsgeräte nur eingesetzt werden, wenn die Werkstücke auch in geeigneter Weise zuführbar sind. Die Schwierigkeiten bei komplexen Handhabungsoperationen werden in den Bereich der Peripherie verlagert. Am besten läßt sich dieser Trend mit dem Beispiel "Griff in die Kiste" verdeutlichen. Damit ein Industrieroboter ungeordnete Teile aus einer Kiste entnehmen kann, ist sehr viel Aufwand mit Bilderkennungssystemen und Werkstückgestaltung zu treiben. Dagegen bereitet die Handhabung der gleichen Teile dann keine Schwierigkeiten mehr, wenn sie geordnet und vereinzelt zugeführt werden können.

Funktionskomplex VERBINDEN

Die Geräte dieses Funktionskomplexes haben die Aufgabe den Werkstücktransport zwischen den einzelnen Funktionsschritten und Baugruppen zu realisieren. Demzufolge fallen darunter:

- Transportsysteme ggf. mit Werkstückträger,
- Gestelle,
- Pufferspeicher.

Die Geräteauswahl wird ganz wesentlich von der Organisationsform der Anlage und von ausgewählten Handhabungsgeräten bestimmt. Es ist ganz offensichtlich, daß die Transportsysteme bei flexiblen

Montagelinien mit NC-Achsen anders ausgeprägt sind als bei einer Nestmontage mit Industrieroboter. Im letzteren Fall muß beispielsweise der Roboter zusätzlich noch den Werkstücktransport innerhalb der Zelle übernehmen.

Funktionskomplex STEUERN

Die Geräte dieses Funktionskomplexes bauen eine komplette Steuerungshierarchie der Montageanlage auf:

- Anlagensteuerung,
- Baugruppensteuerung,
- Zellensteuerung,
- Gerätesteuerung.

Den einzelnen Hierarchieebenen lassen sich hier in einfacher Weise entsprechende Gerätetypen zuordnen:

- Workstation (WS),
- Personal Computer (PC),
- Numerische Steuerung (NC),
- Speicherprogrammierbare Steuerung (SPS).

Die Steuerung und Überwachung einer ganzen Anlage erfolgt zweckmäßigerweise über eine Art Leitreechner. Während hierfür eine Workstation erforderlich ist, genügt zur Überwachung und Diagnose von Baugruppen schon ein einfacher PC. Die Steuerungsfunktionen in einer Zelle selbst werden teilweise von den Gerätesteuern, wie beispielsweise einer Robotersteuerung, teilweise aber auch von speicherprogrammierbaren Steuerungen übernommen.

5.3.2.5 Ermittlung der Zeiten

Eine wichtige Planungsgrundlage für die Auswahl der Geräte und für die Bildung von Zellen (Blockung) ist die Funktionsschrittzeit. Es handelt sich hierbei um das Zeitintervall, welches ein

Gerät zur Ausführung einer Funktion benötigt. Man muß in diesem Zusammenhang zwischen maschinellen und manuellen Zeiten unterscheiden. Maschinelle Zeiten beziehen sich auf automatische Geräte und manuelle Zeiten auf Werker. Zur Ermittlung von manuellen Zeiten existieren in der Praxis verschiedene Methoden. Die verbreitetsten davon sind:

- Methods Time Measurement (MTM) [90],
- Work Factor nach REFA (WF) [91].

Diese Verfahren basieren hauptsächlich auf empirischem Wissen und auf Näherungswerten. Die Ermittlung der maschinellen Zeiten kann sehr einfach sein, wenn die entsprechende Geräteleistung beispielsweise in Arbeitstakten pro Sekunde gegeben ist. Sie kann sich aber auch äußerst schwierig gestalten, wenn beispielsweise die Handhabungszeit eines Industrieroboters ermittelt werden soll. Die Einflußfaktoren in einem solchen Falle sind:

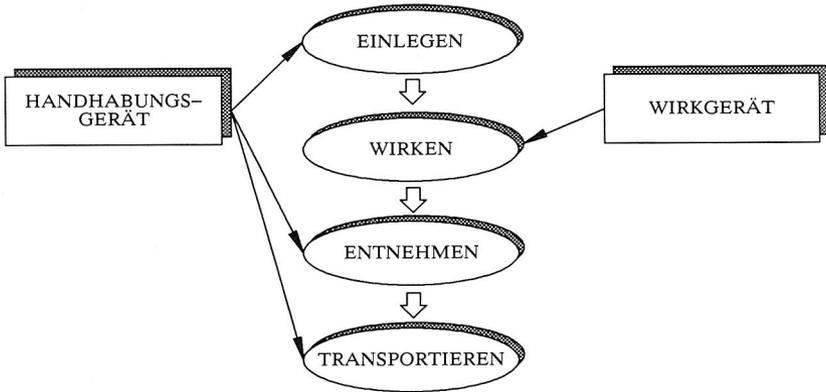
- Geschwindigkeitsverhalten des Gerätes,
- Schwierigkeitsgrad der Handhabungsaufgabe,
- Länge und Komplexität der Bewegungsbahn,
- Form und Verlauf der Bewegungsbahn.

Es wird sofort deutlich, daß während einer Planungsphase hier nur Näherungswerte ermittelt werden können. Das Geschwindigkeitsverhalten eines Industrieroboters unter Last, entlang einer komplexen Bewegungsbahn im Raum, läßt sich praktisch nicht exakt berechnen. Die dynamischen Verhältnisse während der Beschleunigungs- und Abbremsphase sowie beim Überschleifen in Abhängigkeit vom vorliegenden Handhabungsgewicht sind zu komplex.

Die Zeitenermittlung im Rahmen eines Expertensystems könnte theoretisch auch über ein Tabellenwerk ähnlich der Work-Factor-Methode erfolgen. Da dies allerdings umfangreiche numerische Algorithmen erfordert, wurde im Rahmen von MOPLAN darauf verzichtet. Dies bedeutet allerdings nicht, daß solche Erweiterungen in Zukunft nicht möglich wären.

In Rahmen eines wissensbasierten Systems können die Zeiten über Regeln abgeschätzt werden. Diesem Vorgang liegen die Modelle aus Bild 62 zugrunde:

MODELL A : HANDHABUNGSGERÄT MIT WIRKGERÄT



MODELL B : HANDHABUNGSGERÄT OHNE WIRKGERÄT

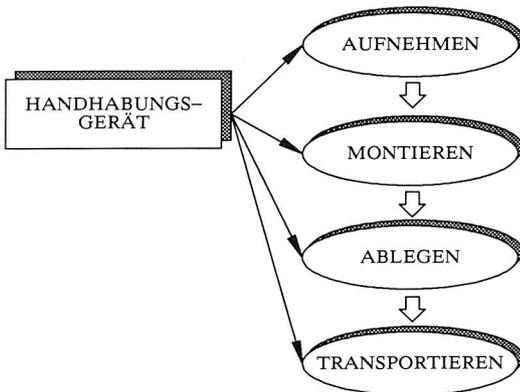


Bild 62: Modelle zur Zeitenermittlung

Beim Modell a benötigt das Handhabungsgerät Zeiten zum Einlegen, Entnehmen und ggf. auch Transportieren des Werkstücks, während das Wirkgerät die Zeit für die eigentliche Funktion verbraucht. Bei dem Modell b führt das Handhabungsgerät sämtliche Funktionen selbst aus. Ein Wirkgerät existiert nicht. Die Zeit pro Funktionsschritt ergibt sich dann als Summe der Einzelzeiten. Diese können dann über Regeln abgeschätzt werden.

5.3.2.6 Blockung der Montageanlage

Unter dem Begriff der "Blockung" einer Montageanlage versteht man das Zusammenfassen von einzelnen Geräten zu den Montagezellen. Dieser Vorgang ist von den folgenden Faktoren abhängig:

- Organisationsform der Baugruppe,
- Zeitbedarf der Einzelgeräte,
- Taktzeit der Baugruppe,
- Verträglichkeit der Geräte untereinander.

Als Organisationsformen kommen im wesentlichen nur Linien- und Nestmontage in Betracht. Allerdings müssen eventuelle Funktionsvervielfältigungen, wie beispielsweise Parallelanordnungen von Geräten oder Einsatz von Vielfachgreifern, berücksichtigt werden. Der Zeitbedarf aller Geräte aus dem Funktionskomplex WIRKEN sollte möglichst nahe an die Taktzeit der Baugruppe herankommen. Ferner sollten nur solche Geräte zu Zellen zusammengefaßt werden, die eine gegenseitige Verträglichkeit aufweisen. Handarbeitsplätze und Industrieroboter sind beispielsweise unverträglich. Generell sollten manuelle und maschinelle Tätigkeiten entkoppelt werden.

Ziel der Zellenbildung ist es dann, zu einer optimalen Austaktung der Montageanlage zu gelangen, bei der die Verlustzeiten möglichst minimiert sind. Hierfür wurden bereits numerische Verfahren entwickelt, die recht genaue Aussagen erlauben [77]. Im Rahmen eines wissensbasierten Systems sind grobe Abschätzungen mit Hilfe von Produktionsregeln ausreichend und auch sinnvoll.

5.3.2.7 Simulation des Einsatzverhaltens

Gerade in der Planungsphase von kapitalintensiven Montageanlagen ist es von großem Interesse, Kenntnisse über das Einsatzverhalten der zukünftigen Anlage zu erlangen. Hierfür gibt es praktisch nur zwei Möglichkeiten. Entweder der planende Experte hat so viel Erfahrung, daß er diese Abschätzung selbst vornehmen kann oder man bedient sich der Simulation als Hilfsmittel. Die Arbeitsweise konventioneller und wissensbasierter Simulationssysteme wurde in den vorausgegangenen Kapiteln 3.4 und 5.2.1 beschrieben.

Im Rahmen der wissensbasierten Planung von Montageanlagen wird der Simulator (vgl. Kap. 4.4) durch das Regelwerk des Systems MOPLAN aktiviert. Daraufhin erfolgt der automatische Aufbau des Simulationsmodells entsprechend der bisher konfigurierten Anlage. Das heißt, für jede Station und jedes Pufferelement werden die entsprechenden Frames des Simulationsmodells angelegt. Der Simulator arbeitet in der beschriebenen Weise und ermittelt je Station und Puffer die Simulationskennzahlen.

Aus dem Bündel der Simulationsergebnisse soll an dieser Stelle die stationäre Verfügbarkeit der Anlage herausgegriffen und etwas genauer betrachtet werden. Dieser Kennwert ist ein Maß für die Störungsempfindlichkeit der Montageanlage. Grundlage der Ermittlung ist die mittlere Laufdauer und die mittlere Ausfalldauer der Komponenten i [67] [92].

$$V_i = \frac{MTBF_i}{MTBF_i + MTTR_i}$$

MTBF := Meantime Between Failure

MTTR := Meantime To Repair

Nach REISCH [67] gilt als Randbedingung für die Berechnung, daß sich der Kennwert auf definierte umgebungs- und funktionsabhängige Beanspruchungsbedingungen innerhalb fester Systemgrenzen bezieht und damit nicht allgemein übertragbar ist. Es wird hier

deutlich, daß es sich bei der Verfügbarkeit um keine analytisch exakt ermittelbare Größe handelt. Vielmehr hat sie einen ausgeprägt "vagen" Charakter. Bei konventionell programmierten Simulationssystemen behilft man sich mit theoretischen und modellhaften Verteilungsfunktionen der Lebensdauer [76]. Diese mathematischen Funktionen (Normalverteilung, Weibull-Verteilung, Exponentialverteilung u. a.) können natürlich in keiner Weise die immer wieder neuen Einflüsse immer wieder neuer Werkstücke und Systemumgebungen berücksichtigen. Hier eröffnen die Methoden der Wissensverarbeitung vielfältig neue Möglichkeiten. Mit Hilfe von Produktionsregeln können Erfahrungen zur Lauf- und Ausfalldauer technischer Systeme sehr viel individueller und genauer gemacht werden.

Die Interpretation der Simulationsdaten mit den entsprechenden Reaktionen kann nun ebenfalls wieder regelbasiert und damit automatisch erfolgen. Beispielsweise werden Schwachstellen durch eine zu geringe stationäre Verfügbarkeit erkannt. Als Reaktion darauf kann dann die Geräteauswahl an dieser Stelle neu durchgeführt werden. Aufgrund der Belegungsstatistik der Pufferelemente können Veränderungen der Pufferkapazität, wie beispielsweise das Vergrößern oder Verkleinern von Staustrecken, veranlaßt werden.

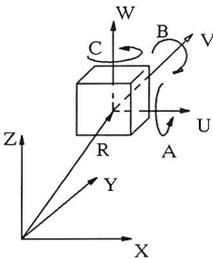
Das Zusammenwirken von Konfiguration, Simulation und regelgesteuerter Reaktion stellt somit eine einzigartige Methode zur automatischen Optimierung von Montageanlagen dar. Die Nachteile isoliert arbeitender Simulationssysteme, insbesondere der hohe Interaktionsaufwand zur Modellbeschreibung und das Fehlen von automatischen Optimierungsverfahren, werden somit vermieden.

5.3.2.8 Erstellung des Groblayouts

Für die Layouterstellung von Montageanlagen sind im wesentlichen die folgenden Kriterien maßgebend:

- Raumbedarf der Geräte und Gesamtanlage,
- funktional korrekte Anordnung der Geräte,
- kollisionsfreie Anordnung der Geräte.

Derzeit existieren keine Möglichkeiten zur vollständigen, automatischen und interaktionsfreien Durchführung dieser Aufgaben. Im großen und ganzen handelt es sich hierbei auch um numerische Probleme. Zu deren Lösung müssen Objekte im Raum mit ihrer Position, Orientierung und Ausdehnung mathematisch eindeutig beschrieben werden. Zweckmäßigerweise bedient man sich hierzu mathematischer Grundlagen.



- X, Y, Z : BEWEGUNGSKOORDINATEN
 U, V, W : KÖRPEREIGENES
 KOORDINATENSYSTEM
 R : ORTSVEKTOR MIT (X, Y, Z)
 (U, V, W = TRANSLATORISCHE
 FREIHEITSGRADE
 A, B, C = ROTATORISCHE
 FREIHEITSGRADE)

Bild 63: Mathematische Beschreibung von Objekten im Raum
 (Quelle: [89])

Regelbasierte Systeme sind für derartige Verfahren gänzlich ungeeignet. Die funktional korrekte Geräteanordnung kann nur über einfache Zuordnung erfolgen. So werden Handhabungsgeräte, Transportsysteme, Zuführungen u. a. in Abhängigkeit von der vorliegenden Montageorganisationsform (Linie, Nest) auf vorgegebene Positionen innerhalb einer Zelle plazierte. Die Orientierung dieser Geräte wird über einen speziellen Orientierungswinkel bestimmt. Hierfür lassen sich geeignete Produktionsregeln aufstellen.

Die kollisionsfreie Geräteanordnung wird dann im wesentlichen noch vom Raumbedarf der Einzelgeräte bestimmt. Hierzu bieten CAD-Systeme bereits implementierte Verfahren, sodaß diese Aufgabe auch nicht mit einem regelbasierten System gelöst werden sollte.

5.3.2.9 Ermittlung der Kenndaten

Zur Beurteilung von Montageanlagen können spezielle Kenndaten herangezogen werden. Die wichtigsten hierbei sind:

- Nutzungsgrad,
- Personalbedarf,
- entstandene Kosten,
- Raumbedarf.

Diese Kenndaten können hierarchisch für die Zellen, die Baugruppen und die Gesamtanlage aus den Daten der eingesetzten Geräte ermittelt werden. Auf diese Weise ergibt sich für das Management ein vollständiges und zugleich übersichtliches Bild der Anlage.

Der Nutzungsgrad ist ein Maß für die Auslastung und die Störunganfälligkeit der Montageanlage. Er wird vom Simulator ermittelt (vgl. Kap. 5.3.2.7).

Der Personalbedarf ergibt sich als die Summe der Personen zum Überwachen, Bedienen und Entstören der Anlage. Zusätzlich sind die Speicher und Bunker der Zuführlösungen mit Zuführteilen nachzufüllen. Auch automatisierte Montageanlagen benötigen also ein gewisses Maß an Personal.

Bei den Kosten und dem Raumbedarf handelt es sich um einfache Kennwerte, die man nur geräteweise addieren braucht. In dem Kostenansatz sind die Aufwendungen für Personal und Gemeinkosten nicht enthalten. Es handelt sich vielmehr um die Summe der Beiträge für Geräte, Zubehör und ggf. Engineeringleistungen. Komplexere Verfahren wie Renditen- oder Kapitalrückflußrechnungen sollten auch nicht im Rahmen eines wissensbasierten Systems erfolgen. Hierfür existieren in vielen Unternehmen bereits fertige Programme konventioneller Art.

5.3.3 Ausgabe der Planungsergebnisse

Nach Beendigung des Planungsvorganges müssen die angefallenen Ergebnisse ausgegeben werden. Es sind dabei die folgenden Listen vorgesehen:

- Stückliste,
- CAD-Liste,
- Kenndatenliste,
- Simulationsdatenliste.

Die Stückliste ist baukastenartig strukturiert und gliedert sich in Anlage, Baugruppen, Zellen und Geräte.

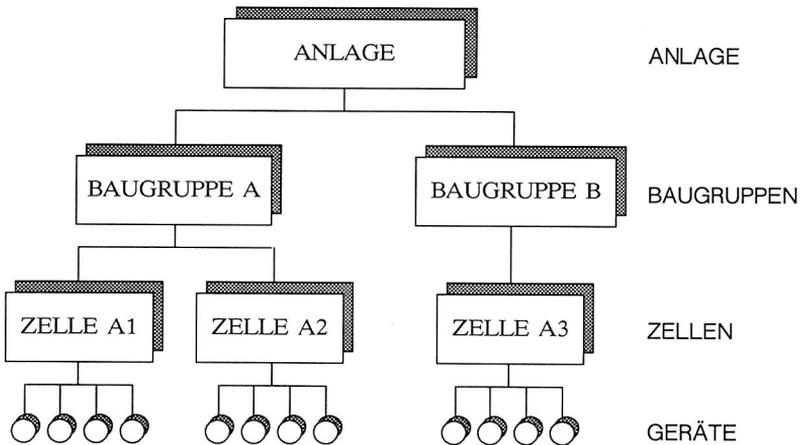


Bild 64: Aufbau der Baukastenstückliste

Mit Hilfe der CAD-Liste wird eine Kopplung zu dem CAD-System ROMULUS hergestellt (vgl. Kap. 6). Sie enthält neben den Bezeichnungen der Geräte (Dateinamen) auch Angaben über Position, Orientierung und Ausdehnung der Geräte. Jeweils das erste Gerät einer Zelle dient als Bezugspunkt für Positions- und Orientierungsangaben.

GERÄT	POSITION	ORIENTIER.	LÄNGE	BREITE	HÖHE
NAME	0-360 GRD	0-360 GRD	Y-AUS- DEHNUNG	X-AUS- DEHNUNG	Z-AUS- DEHNUNG

Bild 65: Aufbau der CAD-Liste

Die Liste der Kenndaten ist hierarchisch organisiert.

ANLAGE:	NAME NUTZUNGSGRAD PERSONALBEDARF KOSTEN RAUMBEDARF
BAUGRUPPE:	NAME NUTZUNGSGRAD PERSONALBEDARF KOSTEN RAUMBEDARF
ZELLE:	NAME NUTZUNGSGRAD PERSONALBEDARF KOSTEN RAUMBEDARF

Bild 66: Aufbau der Kenndatenliste

Die Liste der ermittelten Simulationsdaten wurde bereits in Kap. 4.4.4 beschrieben.

5.4 Die Wissensbasen

Neuzeitliche Expertensysteme sind durch eine strukturelle und architektonische Zweiteilung in anwendungsneutrale und anwendungsspezifische Bereiche gekennzeichnet. Die anwendungsabhängigen Komponenten werden größtenteils durch die Wissensbasis repräsentiert. Bei der Planung von Montageanlagen besteht diese Expertise aus den folgenden Bausteinen:

- das Wissen über die funktionalen und technischen Eigenschaften von Geräten und Maschinen,
- das Wissen über Eigenschaften von Werkstücken, Zuführteilen und zusammengesetzten Baugruppen,
- das Wissen zu den technologischen Einflußfaktoren der Bearbeitungs-, Handhabungs- und Montageprozesse,
- das Wissen zu den Einsatzbedingungen, Auswahlmethoden und Auswahlkriterien der Geräte und Maschinen,
- das Wissen zur Systematik der Montageplanung.

Für jede dieser Wissenskomponenten wurde im Rahmen von MOPLAN eine eigene Organisationsform geschaffen (Bild 67). Dabei dienen Frame-Konstrukte zur Repräsentation von objektbezogenem Geräte- und Teilewissen, während Produktionsregeln das deklarative Wissen der Planungsaufgaben und Planungsabläufe repräsentieren. Zu den Frame-Konstrukten zählen die GERÄTEWISSENSBASIS, die TEILEWISSENSBASIS und die TECHNOLOGIEWISSENSBASIS. Die Produktionsregeln sind in METAREGELN und PLANUNGSREGELN unterteilt.

Im Rahmen der vorliegenden Arbeit konnten und sollten die Wissensbasen natürlich nur in exemplarischer Weise gefüllt werden. Schon in Kapitel 3.6 wurde ausgeführt, inwieweit neue Anwendungen mit neuen Wissensbasen erschlossen werden können. Der Anspruch auf Vollständigkeit ist mit einem beträchtlichen zusätzlichen Aufwand verbunden. Zur Einordnung dieser Aussage ist aber festzuhalten, daß ein solcher Aufwand immer noch weit geringer einzuschätzen ist als bei der Erstellung eines konventionellen Programmsystems.

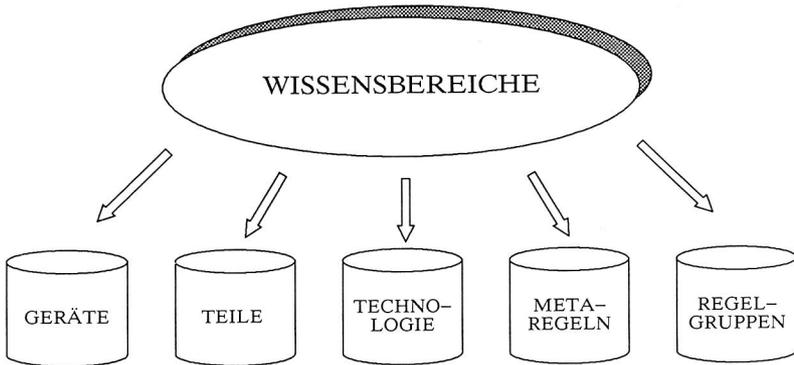


Bild 67: Komponenten der MOPLAN-Wissensbasis

5.4.1 Die Gerätewissensbasis

In der GERÄTEWISSENSBASIS wird Objektwissen über technisch relevante Eigenschaften repräsentiert. In Abhängigkeit von der Funktion der in Montageanlagen vorkommenden Komponentem wurden 17 Geräteklassen gebildet:

- HANDHABUNGSGERÄTE,
- GREIFER,
- SENSORIK,
- STEUERUNGEN,
- WIRKGERÄTE,
- GESTELLE,
- SCHUTZEINRICHTUNGEN,
- PUFFERSPEICHER,
- TRANSPORTSYSTEME,
- WERKSTÜCKTRÄGER,
- POSITIONIERSTATIONEN,
- ZUFÜHRLÖSUNGEN,
- SPEICHEREINRICHTUNGEN,
- ORDNUNGSEINRICHTUNGEN,

- ZUTEILEINRICHTUNGEN,
- BEWEGUNGSEINRICHTUNGEN,
- ZUBEHÖR.

Die Geräte einer Klasse werden je nach Funktionsprinzip in verschiedene Typen weiter unterteilt. Für die HANDHABUNGSGERÄTE existieren beispielsweise die Typen:

- SPHÄRISCHE INDUSTRIEROBOTER (SIR),
- ZYLINDRISCHE INDUSTRIEROBOTER (ZIR),
- KARTESISCHE INDUSTRIEROBOTER (KIR),
- MONTAGEROBOTER (MR),
- NC-ACHSGERÄTE (NC),
- PNEUMATISCHE GERÄTE (PG),
- MECHANISCHE EINLEGEGERÄTE (MG),
- STARRE AUTOMATEN (AT).

FRAME HANDHABUNGSGERÄT	
NAME	
TYP	
ACHSANZAHL	
ARBEITSHÖHE	MM
ARBEITSLÄNGE	MM
ARBEITSBREITE	MM
FLÄCHENBEDARF	M2
LÄNGE	MM
BREITE	MM
HÖHE	MM
BEWEGUNGSBAHN	
GESCHWINDIGKEIT	M/S
POSITIONIERGENAUIGKEIT	MM
HANDHABUNGSGEWICHT	KG
VERSORGUNG	
KOSTEN	DM
LIEFERZEIT	TAGE
PERSONALBEDARF	
SENSORIK	
STEUERUNG	
GREIFER	
ZUBEHÖR	

Bild 68: Frame-Aufbau für HANDHABUNGSGERÄTE

Aufbau und Inhalt der Frames ist für jede GERÄTEKLASSE spezifisch. Bild 68 zeigt beispielhaft einen Frame der Klasse HANDHABUNGSGERÄTE.

Die Zuordnung von Greifern, Sensorik, Steuerung und sonstigem Zubehör erfolgt über einen entsprechenden Slot-Eintrag. Die eingetragenen Namen referieren dann auf einen Frame der jeweiligen Geräteklasse.

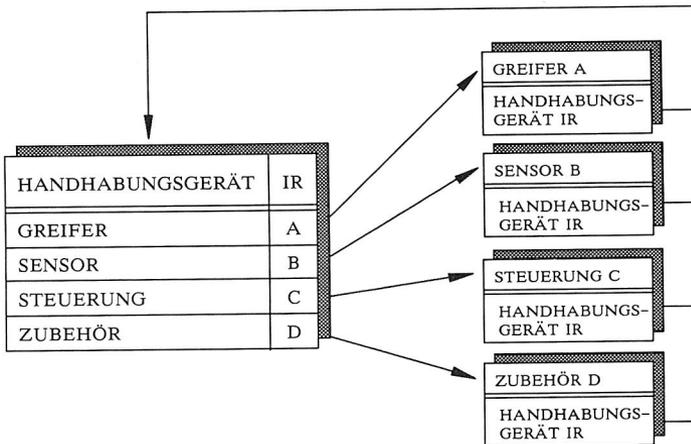


Bild 69: Zuordnung der Zusatzausrüstung zu den Geräten

Mit Hilfe dieser Organisationsform wird ein Höchstmaß an Redundanzfreiheit gewährleistet, ohne daß die Kombinationsmöglichkeiten eingeschränkt sind. Jedes Gerät und jede Ausrüstungskomponente ist nur einmal in der Wissensbasis abgelegt. Dies ist für die Konsistenz sowie den Pflege- und Wartungsaufwand von entscheidender Bedeutung.

5.4.2 Die Teilewissensbasis

In der TEILEWISSENSBASIS wird das Wissen über die relevanten Eigenschaften der Zuführ- und Basisteile verwaltet. Zu diesem Zweck ist eine Aufteilung in die Bereiche SONDERTEILE und NORM-TEILE sinnvoll (Bild 70).

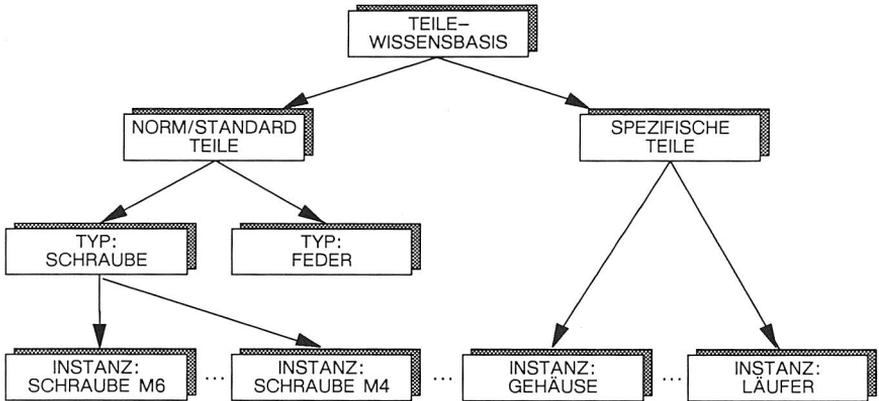


Bild 70: Organisation der TEILEWISSENSBASIS

Während spezifische Teile direkt mit ihrer Instanz vorhanden sind, existieren bei den Norm- und Standardteilen noch zusätzlich Teiletypen wie beispielsweise Schrauben, Federn, Stift u.s.w.. Mit Hilfe des Vererbungsmechanismus können nun Eigenschaften von den Typ-Frames auf die Instanzen vererbt werden. Es sind also nur noch instanzspezifische Eigenschaften zu editieren.

Die Existenz von Teiletypen hat ferner den entscheidenden Vorteil, daß sich das Erfahrungswissen aus der Teilebehandlung oder Zuführung in Form von Produktionsregeln auf diese Teiletypen beziehen kann. Als Beispiel sei hier das Know-how über die Auswahl von Zuführlösungen für Federn genannt. Den Aufbau der eigentlichen Teile-Frames zeigt Bild 71.

FRAME TEILE	
BEZEICHNUNG	TEILENAME
VERHALTENSTYP	(WIRR-, FLACH-, MASSIV-, KUGEL-, BLOCK-, ZYLINDER-, KEGEL-, PYRAMIDEN-, PILZ-, HOHL-, LANG-, FORMTEIL)
GRÖSSEN- KLASSE	(01, 02, 03, 04, 05, 06, 07, 08)
RUHE- UND FÖRDER- VERHALTEN	(ROLL-, GLEIT-, STAPELFÄHIG, STANDFEST)
GEWICHT	TEILEGEWICHT (g)
PHYSIKALISCHE EIGENSCHAFTEN	(BRUCH-, STOß-, OBERFLÄCHENEMPFINDLICH)

Bild 71: Aufbau eines Teile-Frames

5.4.3 Die Technologiewissensbasis

Zur Bereitstellung von technologischem Wissen zu den Bearbeitungs- und Montageprozessen wurde eine spezielle TECHNOLOGIEWISSENSBASIS organisiert. Diese Prozesse werden von einer Vielzahl von Einflußfaktoren bestimmt. In Bild 72 sind beispielhaft einige der wichtigsten Parameter dargestellt.

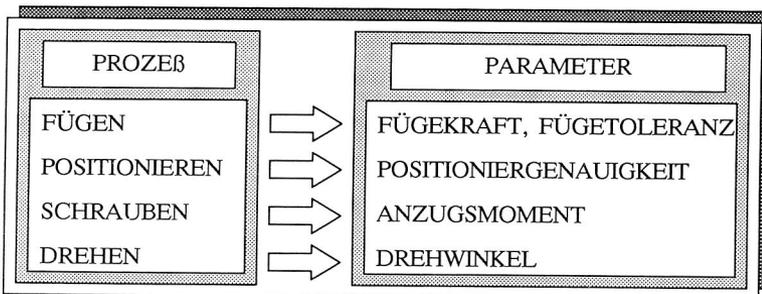


Bild 72: Prozeßbestimmende Einflußfaktoren

Die Repräsentation dieses Technologiewissens kann grundsätzlich sowohl mit Regeln als auch mit Frames erfolgen. Da jedoch die Ausführung einer Regel wegen der Komplexität des Match-Vorganges langwieriger ist als ein einfacher Zugriff auf einen Slot, wurde die TECHNOLOGIEWISSENSBASIS in Frame-Form realisiert.

5.4.4 Die Planungsregeln

Die Produktionsregeln zur Durchführung der einzelnen Planungsaufgaben sind in Gruppen zusammengefaßt. Die Einteilung erfolgte nach funktionalen Gesichtspunkten. So existieren Regelgruppen zu den folgenden Aufgaben:

- Rationalisierungspotential prüfen,
- Montagestruktur ermitteln,
- Gerätetypen auswählen,
- Funktionale Geräteignung prüfen,
- Umweltbedingungen prüfen,
- Werkstückeignung prüfen,
- Varianteneignung prüfen,
- Versorgungsanschlüsse prüfen,
- Kenndaten ermitteln,
- Geräteausprägungen auswählen,
- Zellenbildung (Blockung) durchführen,
- Baugruppen der Anlage verbinden,
- Anlage optimieren,
- Layout erstellen,
- Ergebnisausgabe vornehmen.

Fortschrittliche Produktionssysteme sind in der Lage, mehrere Repräsentationsformen der Produktionsregeln zu verwalten. Sinnvollerweise sollte neben der dem System verständlichen Regelform auch eine für den Benutzer verständliche Form zur Verfügung stehen. Diese benutzerlesbare Regelbeschreibung kann beispielsweise in einem Kommentar-Slot des entsprechenden Regelrahmens abgelegt sein. Nachfolgend sind einige Planungsregeln aufgeführt:

```
=====
Frame-Name : METAREGELN           Rahmentyp : REGEL           Niveau : 3
=====
```

```
1 - AUTHOR
   (($VALUE GEYER))
```

```
2 - DATUM
   (($VALUE 29/5/86R))
```

```
3 - STRATEGIE
   (($VALUE ALL))
```

```
* * * * *
```

```
Regel: RRP1           Gruppe: RATIONALISIERUNGSPOT
RRP1
```

```
wenn:
  =BAUGRUPPE <=? BETRIEBSZEIT 435 <=? LEBENSDAUER 48
dann:
  <SET> =BAUGRUPPE MONTAGE_PRINZIP MANUELL
```

```
* * * * *
```

```
Regel: RRP2           Gruppe: RATIONALISIERUNGSPOT
RRP2
```

```
wenn:
  =BAUGRUPPE <=? BETRIEBSZEIT 870 <=? LEBENSDAUER 24
dann:
  <SET> =BAUGRUPPE MONTAGE_PRINZIP MANUELL
```

```
* * * * *
```

```
Regel: RRP4           Gruppe: RATIONALISIERUNGSPOT
RRP4
```

```
wenn:
  =BAUGRUPPE <=? TEILEZAHL 3 <=? MAX_GEWICHT 7500 <=? AUSBRINGUNG 7500
und:
  UMWELT ?UM
und:
  ?UM @[ EINFLUSS SAUERE LAERM GASE GEFAHR
dann:
  <SET> =BAUGRUPPE MONTAGE_PRINZIP MANUELL
```

Bild 73: Beispiel für Planungsregeln

5.4.5 Die Metaregeln

Metaregeln sollen das Wissen über das Wissen repräsentieren. Sie sind somit dazu bestimmt, die Systematik des Planungsvorganges zu beschreiben und damit den Planungsablauf zu steuern. Dies läuft dann so ab, daß die einzelnen Gruppen der Planungsregeln von den Metaregeln aktiviert und deaktiviert werden.

Metaregeln haben aber auch die Aufgabe, die korrekte Funktionsweise des wissensbasierten Werkzeugsystems (WWS) sicherzustellen. So wird dem System beispielsweise eine Verhaltensvorschrift gegeben, wenn keine ausführbaren Regeln gefunden wurden oder wenn die Ausführung eines Aktionsteiles aus irgendwelchen Gründen unmöglich gemacht ist. Metaregeln können also Kontrollmechanismen sein, die in konventionellen DV-Systemen fest programmiert sind.

```

                Regel:  ME1           Gruppe: METAREGELN
ME1
wenn:
    BAUGRUPPEN ?BAU
und:
    ?BAU =? MONTAGE_PRINZIP _
dann:
    <KONTEXT> RATIONALISIERUNGSPOT =BAUGRUPPE ?BAU
und:
    <INTERPRETER> RATIONALISIERUNGSPOT BEST-ONCE

                * * * * *

```

```

                Regel:  ME2           Gruppe: METAREGELN
ME2
wenn:
    BAUGRUPPEN ?BAU
und:
    ?BAU =? MONTAGE_STRUKTUR _
dann:
    <KONTEXT> MONTAGESTRUKTUR =BAUGRUPPE ?BAU
und:
    <INTERPRETER> MONTAGESTRUKTUR BEST-ONCE

```

Bild 74: Beispiele für Metaregeln

5.5 Der Wissenserwerb

5.5.1 Systematik des Wissenserwerbs

Der Wissenserwerb zu dem System MOPLAN wurde im Rahmen einer dreistufigen Systematik durchgeführt (Bild 75). Dabei konzentriert sich die erste Phase auf die Erfassung des theoretischen Wissens.

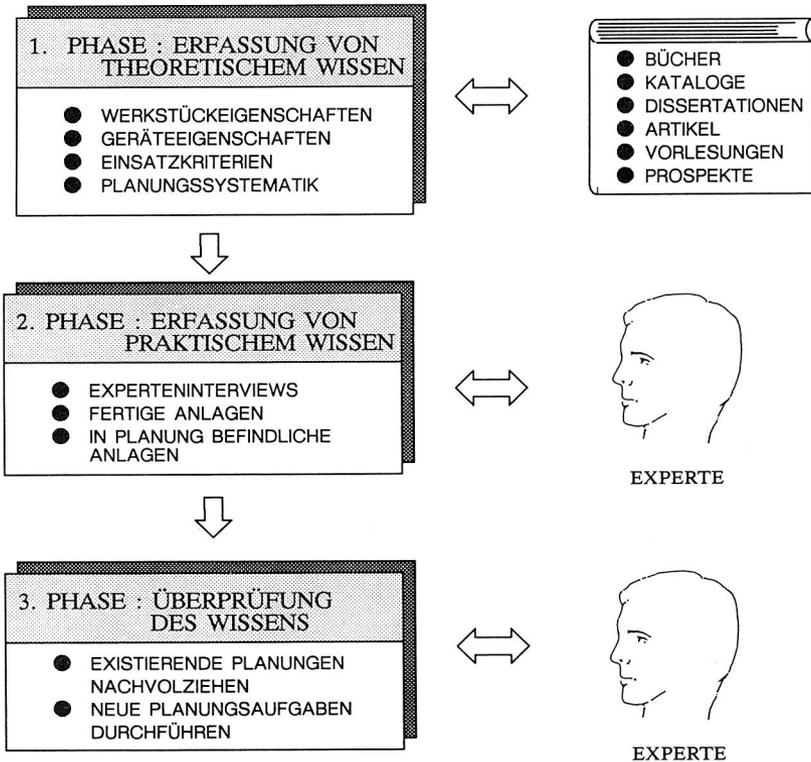


Bild 75: Systematik des Wissenserwerbs

Dieses theoretische Wissen bezieht sich auf die Geräte der Montagetechnik, ihre funktionalen Eigenschaften und ihre Einsatzkriterien. Darüber hinaus sind Beschreibungsformen von Werkstücken und deren Eigenschaften von Interesse. Die Planungssystematik beschreibt die generelle Vorgehensweise und die Einzelaktivitäten. Als Unterlagen stehen Lehr- und Sachbücher, Zeitschriften, Dissertationen, Kataloge und technische Dokumentationen zur Verfügung.

In der zweiten Phase erfolgt dann der Wissenserwerb bei den Praktikern in der Industrie. Entscheidend für die Allgemeingültigkeit und Überprüfbarkeit dieses Wissens ist es, daß es nicht von einer einzigen Person stammt. Denn im ungünstigen Falle können die Aussagen eines Experten auch unvollständig, widersprüchlich oder gar falsch sein.

Wenn an dieser Stelle behauptet wird, daß konsistentes Wissen vollständig, widerspruchsfrei und faktisch richtig sein muß, dann steht dies nicht im Widerspruch zu der Zielsetzung von Expertensystemen, auch sogenanntes vages Wissen verarbeiten zu können. Denn auch die Angabe von Wahrscheinlichkeitsfaktoren dient der Vervollständigung des Wissens.

Zur Gewährleistung der Konsistenz der Expertise wurde der Kreis der befragten Personen einmal auf mehrere Unternehmen und dabei noch auf mehrere Personen dieser Unternehmen ausgedehnt. Durch Vergleich der jeweiligen Aussagen gegeneinander konnte ein gewisser Grad an Vollständigkeit und Widerspruchsfreiheit sichergestellt werden.

Die Kommunikation mit dem Experten wird sinnvollerweise in Form von Interviews geführt. Hierzu ist es erforderlich, daß der Interviewer nicht nur qualifizierte Fragen stellen kann, sondern auch in der Lage sein muß, die Aussagen des Interviewten ad hoc zu verstehen und darauf spontan zu reagieren. Denn nur, wenn die Antworten verstanden wurden, können sie auch auf Konsistenz überprüft werden.

Mit Hilfe spezieller Interviewtechniken kann die Qualität von Expertenaussagen darüber hinaus noch besser überprüft werden. So sollte man die Interviews auf mehrere Zeitpunkte verteilen, um dann jeweils wieder die Aussagen aus einer vorherigen Sitzung hinterfragen zu können. Weitere Möglichkeiten sind die zeitlich versetzte Wiederholung von Fragen aus verschiedenen Zusammenhängen heraus oder die Suggestierung von Aussagen, die zu einem Widerspruch führen würden.

Der Erwerb des praktischen Expertenwissens sollte sich aber nicht nur auf Interviews im Büro beschränken. Von großem Vorteil ist es, wenn bereits realisierte oder gerade in Planung befindliche Anlagen zur Verfügung stehen. Hier kann sich der Interviewer einen viel besseren Eindruck verschaffen. Gleichzeitig ergeben sich beim Anblick der real existierenden Anlagen immer wieder neue Fragestellungen und auch Erklärungen. Ganz abgesehen davon kann man die Expertenaussagen auch anhand der vorliegenden Anlagen überprüfen.

Die dritte und letzte Phase des Wissenserwerbs dient der Überprüfung und Vervollständigung. Sie kann auch als die Erprobungsphase des Expertensystems angesehen werden. Sinnvollerweise sollten zu Beginn erst einmal schon realisierte Planungen von dem Expertensystem nachvollzogen werden. Dabei ist es aber nicht maßgebend, zu den gleichen Planungsergebnissen zu kommen, wie sie auch schon vorliegen. Vielmehr soll das Expertensystem auf der Basis seines Wissens zu Entscheidungen kommen, die montagetech-nisch gesehen sinnvoll und zweckmäßig sind. Zur Beurteilung dieser Entscheidung und zur Korrektur oder Vervollständigung der Wissensinhalte wird nochmals der Montageexperte herangezogen.

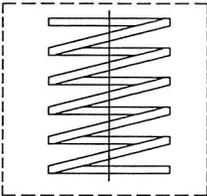
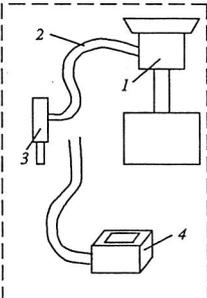
Erst nach dieser Erprobungsphase ist es dann sinnvoll, vollständig neue Planungsaufgaben mit dem System zu bearbeiten. Dabei sollte man sich zu Beginn noch auf eng abgegrenzte Problemstellungen konzentrieren, bevor zunehmend komplexere Aufgaben angegangen werden.

5.5.2 Wissensdokumentation

Zur Dokumentation des erfaßten Wissens empfiehlt es sich einen gewissen Formalismus zu entwickeln. Die Bilder 76 und 77 zeigen die Formblätter, die im Rahmen des Wissenserwerbs zu MOPLAN erstellt und ausgefüllt wurden

1) BEZEICHNUNG	<i>WMW-EINLEGEGERÄT</i>	
2) FUNKTION	<i>POSITIONIEREN, EINLEGEN</i>	
3) EINSATZKRITERIEN	<i>BLOCK-,MASSIV-,KEGEL-,ZYLINDER-, PILZ-,HOHL-,FORM-,PYRAMDENTEILE</i>	
WERKSTÜCKART		
GRÖSSENKLASSE		<i>01, 02, 03</i>
WERKSTÜCKMASSE		<i>MAX 1000 g</i>
AUSTRAGLEISTUNG/ GESCHWINDIGKEIT		<i>30 HÜBE / MIN.</i>
VERFÜGBARKEIT		<i>95 %</i>
VERHALTEN IM STÖRUNGSFALL		<i>STATISCH, TEIL WIRD GEHALTEN</i>
RAUMVERHÄLTNISSE		<i>400 x 200 x300</i>
GEWICHT		<i>8 kg</i>
TEILEVER- U. ENTSORGUNG		<i>-</i>
FLEXIBILITÄT		<i>KURVENSCHLEIBEN</i>
UMWELTBEDINGUNGEN		<i>-</i>
SICHERHEITSBED.		<i>-</i>
INSTANDHALTUNG/ REPERATUR		<i>-</i>
KOSTEN		<i>5000 .- DM</i>
SONSTIGES		
4) ZUBEHÖR	<i>DREHEINHEIT, NORMSPANNZANGE</i>	
5) BESONDERHEITEN		

Bild 76: Formblatt für Einzelgeräte

WERKSTÜCK	: <u>DRUCKFEDER</u>	
VERHALTENSTYP	: <u>WIRRTAIL</u>	
GRÖSSENKLASSE	: <u>1,2,3,4</u>	
ANLIEFERUNGSZUSTAND:	<u>SCHÜTTGUT</u>	
SONST. EIGENSCHAFT	: _____	
ZUBRINGUNG ZU : <u>HANDHABUNGSGERÄT</u>		
GEFORD. ORIENTIERUNG: <u>POSITIONIERT</u>		
GERÄTEKONFIGURATION:	_____	
	<u>1 FEDERENTWIRRGERÄT</u>	
	<u>2 PNEUM. ZUFÜHRUNG</u>	
	<u>3 ZU HHG BEI AXIALER FÜGERICHTUNG</u>	
	<u>4 ZU FEDERZUFÜHREINRICHTUNG BEI RADIALER FÜGERICHTUNG</u>	

PLATZBEDARF	: <u>700 x 400 x 1100</u>	
ZEITVERHALTEN	: <u>40 TEILE / MINUTE</u>	
KOSTEN	: <u>14 000.- DM</u>	
SPEZ. REGELN	: <u>VERSORGUNG DRUCKLUFTNETZ 6 BAR</u>	
	<u>ABHÄNGIG VON FÜGERRICHTUNG ALTERNATIV POSITION 3 ODER 4</u>	

Bild 77: Formblatt für Zufürlösungen

Bei den Tätigkeiten des Wissenserwerbs wurde der Verfasser von Herrn cand. ing. J. Sturm unterstützt.

6. Die CAD-Kopplung

6.1 Das CAD-System ROMULUS

Im Rahmen der vorliegenden Arbeit wurde das Expertensystem MOPLAN mit dem geometrischen Modellierer ROMULUS [93] gekoppelt. Bei ROMULUS handelt es sich um ein dreidimensional arbeitendes CAD-System. Es ist hauptsächlich in FORTRAN 77 geschrieben und auf einer Vielzahl von Computersystemen lauffähig. Ferner wurde ROMULUS auch von verschiedenen anderen CAD-Anbietern als 3D-Teil in deren Systeme integriert. ROMULUS kann für die folgenden Aufgaben eingesetzt werden:

- Erzeugung von Volumenkörpern verschiedener Größe, Gestalt und Komplexität,
- Verändern der Gestalt dieser Objekte,
- Verbinden von mehreren Einzelkörpern auf verschiedene Weise,
- Berechnen von Körpereigenschaften wie Massen und Momente,
- Darstellen der Objekte in verschiedenen Projektionen,
- Speichern und Wiederholen der Objekte mit Hilfe eines Dateiverwaltungssystems.

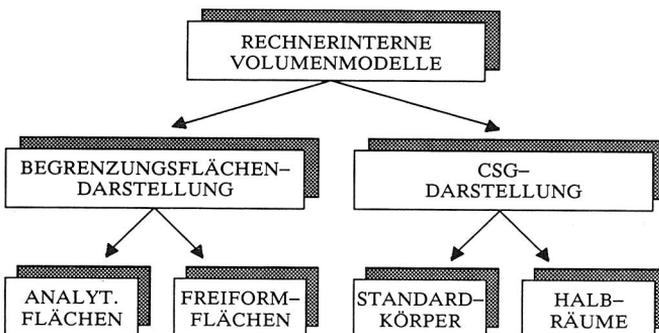


Bild 78: Rechnerinterne Darstellungsformen (RID) bei CAD-Volumenmodellen

Rechnerintern stellt ROMULUS die erzeugten Volumenkörper in Form von einem sog. Begrenzungsflächenmodell dar. Im Gegensatz zu den sog. Vollkörperdarstellungen (CSG = Constructive Solid Geometry) werden nicht die Vollkörper mit ihrer Entstehungshistorie repräsentiert, sondern die Oberfläche, welche den Körper begrenzen [94] [95].

Die Datenstruktur des Systems ist in Bild 79 dargestellt. Demnach verfügt ROMULUS neben rein geometrischen und topologischen Elementen (Entities) auch über zugeordnete (associated) Elemente. Die topologischen Elemente enthalten Aussagen über die Struktur eines Modells, wie zum Beispiel, daß ein Würfel A von den sechs Flächen F1 bis F6 umgeben ist. Die geometrischen Elemente beschreiben die mathematischen Formeln für die Punkte, Kurven, Flächen und Transformationen. Die zugeordneten Elemente sind zur Speicherung nichtgraphischer Informationen vorgesehen.

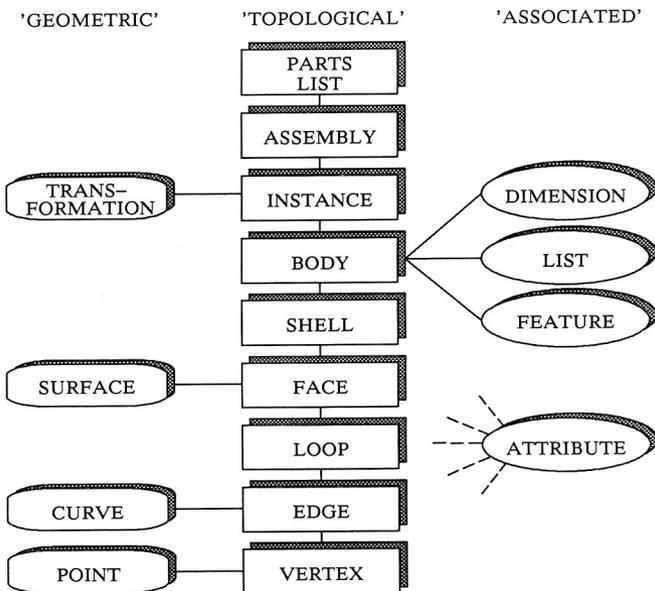


Bild 79: Datenstrukturelemente des Systems ROMULUS

Die geometrische Information in ROMULUS wird im Gegensatz zu den sog. Facettenmodellen (EUCLID, MEDUSA) rechnerintern in mathematisch exakter Form dargestellt.

Eine wichtige Funktion in ROMULUS ist die Programmiersprache MCL (MACRO COMMAND LANGUAGE). Es handelt sich hierbei um eine BASIC-ähnliche Sprache, die innerhalb des ROMULUS-Prozesses abläuft. Der Sprachumfang umfaßt neben einfachen logischen und arithmetischen Anweisung auch die wichtigsten ROMULUS-Befehle. Auf diese Weise wird es möglich, aus einem laufenden Programm heraus CAD-Befehle abzusetzen.

6.2 Modellieren von rechnergeführten Produktionssystemen

Die Struktur von fertigungstechnischen Produktionsanlagen weist eine mehrstufige Hierarchieform auf (vgl. Kap. 5.1). Komplexe Gesamtanlagen gliedern sich in Teilanlagen, die sog. Baugruppen. Diese wiederum bestehen aus einzelnen mehr oder weniger autarken Fertigungszellen. Innerhalb einer Zelle finden wir die einzelnen Geräte, die sich wiederum aus Geräteteilen und Einzelteilen zusammensetzen (Bild 80).

Bei den Einzelgeräten ist zwischen starren, unbeweglichen und beweglichen Geräten zu unterscheiden. Letztere werden auch Mechanismen genannt. Zu den Mechanismen zählen im wesentlichen die Handhabungsgeräte und Transporteinheiten. Sie führen Bewegungen aus. Zwischen den einzelnen Fertigungszellen und den Teilanlagen sorgen zusätzlich Transportsysteme für den notwendigen Materialfluß.

Bei der Modellierung von Produktionssystemen mit CAD muß nun diese Struktur auf das rechnerinterne Modell (vgl. 6.1) abgebildet werden. Entsprechend den Modellierungsfunktionen von ROMULUS besteht jedes Einzelgerät für sich aus einer sog. "Assembly". Diese Assembly hat als "Instances" die jeweiligen Gerätekomponenten. Kommt die gleiche Komponente innerhalb einer Assembly mehrfach vor, so werden zwar mehrere Instances, nicht aber mehrere

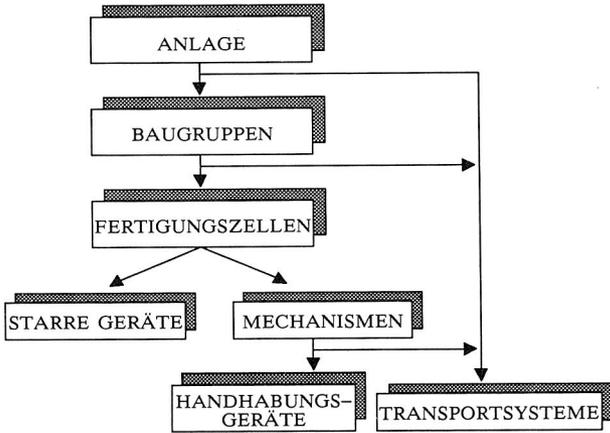


Bild 80: Hierarchie von fertigungstechnischen Produktionsanlagen

Körper selbst erzeugt. Die Daten des Körpers werden nur einmal angelegt. Die Instances fungieren dann quasi als Verweise auf diesen Körper. Bei Handhabungsgeräten sind diese Komponenten die einzelnen Achsen (Bild 81). Die dazugehörigen Effektoren sind selbst wieder eigene Assemblies mit ihren beweglichen Greiferteilen als Instances. Die Geräte- und Effektorachsen selbst werden als Einzelkörper behandelt. Beim Modellierungsvorgang wurden sie aus geometrischen Grundoperationen erzeugt. Die rotatorischen und translatorischen Bewegungsachsen der Mechanismusarme werden diesen direkt zugeordnet.

Die auf diese Weise erzeugten Mechanismen können nun als Grunddaten für weitere Verfahren im Rahmen der "Rechnerintegrierten Produktion" eingesetzt werden (vgl. Kap. 7). So sind sämtliche Manipulationskommandos des CAD-Systems auf diese Modelle anwendbar.

Die Zusammenfassung der Einzelgeräten zu einer Fertigungszelle und dieser zu Baugruppen und Anlagen erfolgt zweckmäßigerweise wieder über Assemblies. Denn nur auf diese Weise wird gewähr-

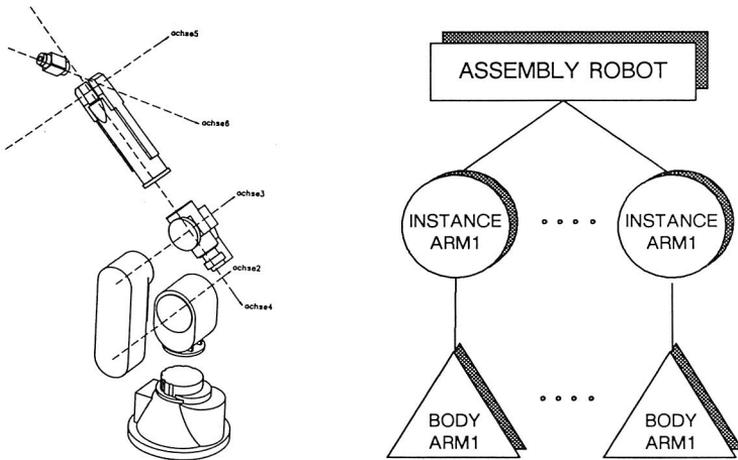


Bild 81: Rechnerinterne Darstellung von Mechanismen mit Hilfe von Assembly-Strukturen

leistet, daß alle organisatorischen Einheiten für das CAD-System manipulierbar sind. Die wichtigsten Funktionen hierzu sind das Drehen, Verschieben, Kopieren, Projizieren, verdeckte Kanten Ausblenden, Abspeichern, Wiederfinden und Plotten.

6.3. Die CAD-Bibliotheken

Entsprechend dem Inhalt der Gerätewissensbasis sind im CAD-System ebenfalls Gerätebibliotheken angelegt. Einige Beispiele zeigen die Bilder 82 - 89.

 Bei der Erstellung der Bibliotheken wurde der Verfasser von den Herren cand.ing. S. Albrecht, cand. ing. R. Köberlein und cand. ing. J. Sturm unterstützt.

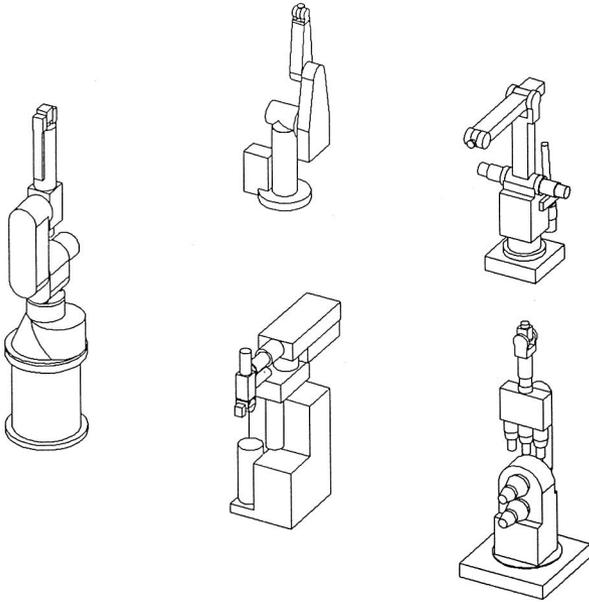


Bild 82: Industrieroboter

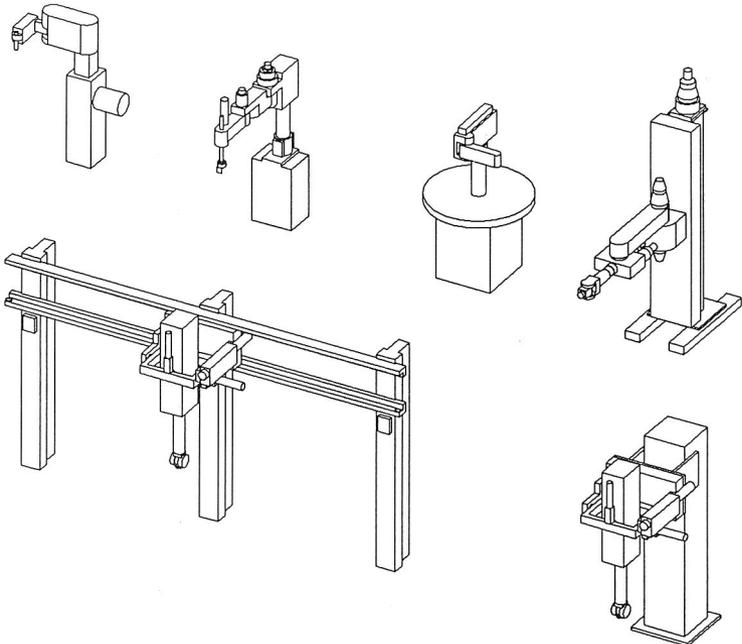


Bild 83: Montageroboter

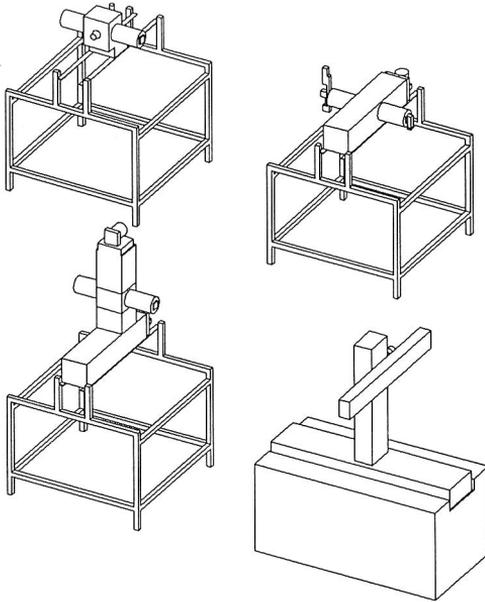


Bild 84: NC-Achssysteme

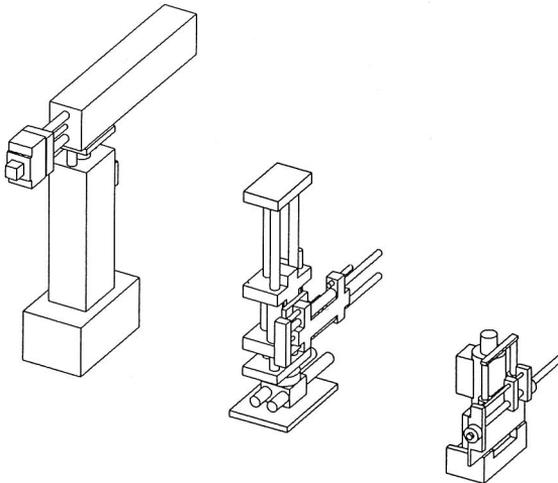


Bild 85: Mechanische und pneumatische Geräte

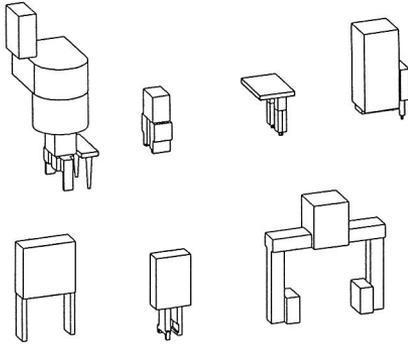


Bild 86: Greifer

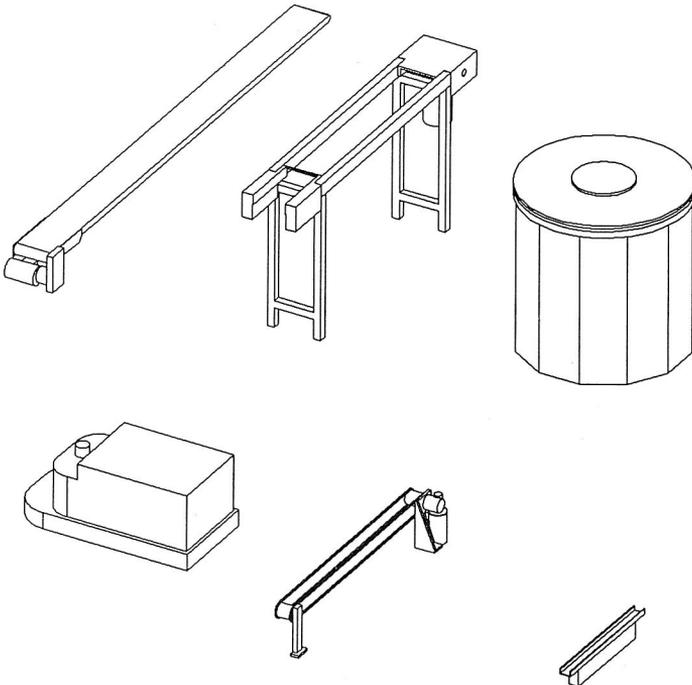


Bild 87: Transportsysteme

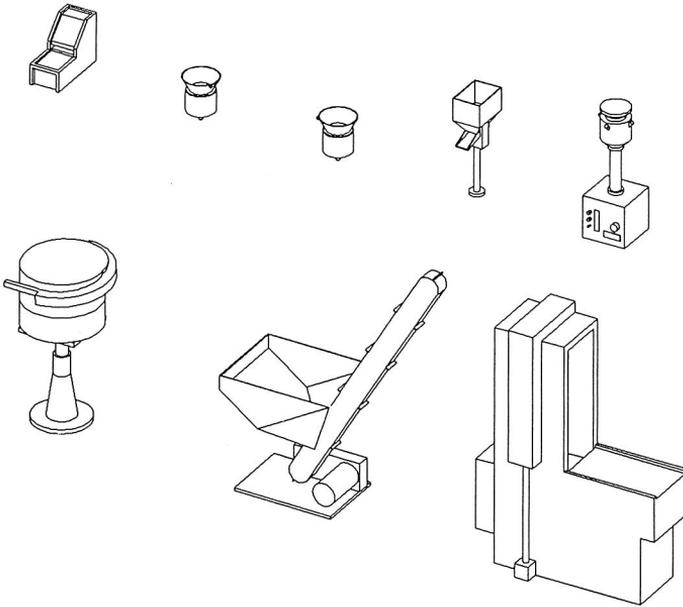


Bild 88: Zuführungen

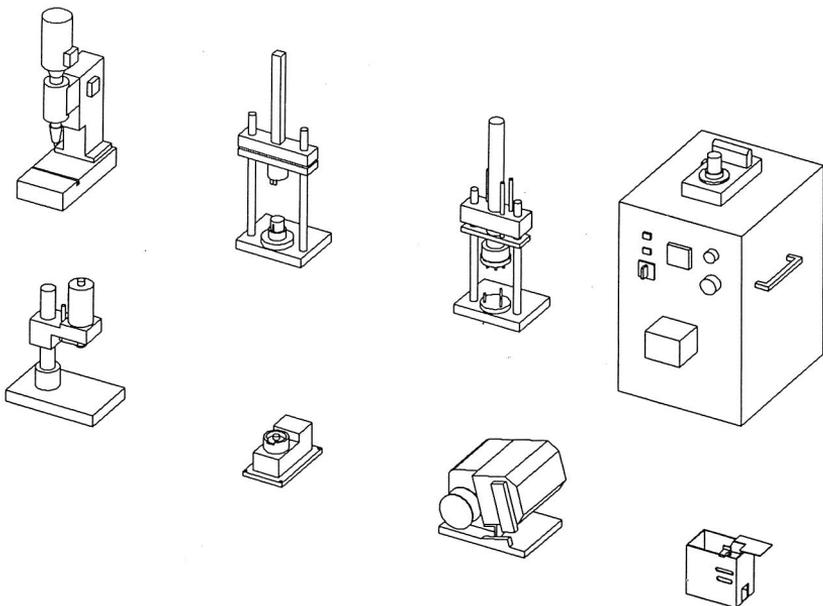


Bild 89: Bearbeitungsgeräte

6.4 Die Kopplung MOPLAN - ROMULUS

Die Kopplung zwischen dem wissensbasierten Planungssystem MOPLAN und dem CAD-System ROMULUS erfolgt über File-Transfer. Da beide Applikationen auf gleichen Rechnern und in gleicher Softwareumgebung ablaufen, wird dies ohne Schwierigkeiten möglich (Bild 90).

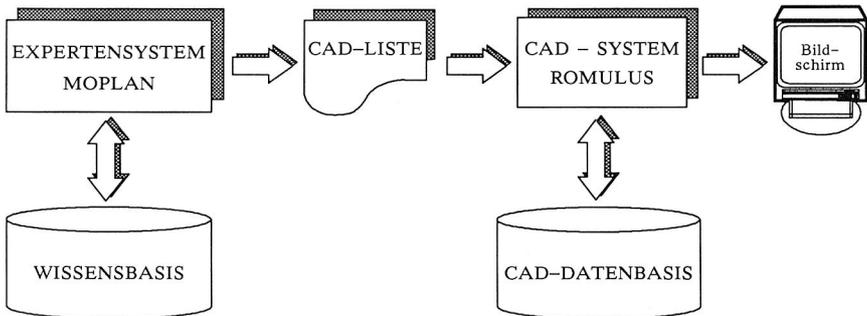


Bild 90: Kopplungsprinzip MOPLAN-ROMULUS

Die Interpretation der Ergebnisdaten aus MOPLAN übernimmt ein spezielles Koppelprogramm, welches in der MACRO COMMAND LANGUAGE (MCL) geschrieben ist. Es arbeitet die CAD-Liste zeilenweise ab und liest die Gerätenamen mit Positions- und Orientierungsangaben ein. Da die Gerätebezeichnungen direkt mit den Files der CAD-Bibliothek korrelieren, können die Geräte direkt aufgerufen und angesprochen werden. Danach ist die Plazierung an der vorgegebenen Position ebenfalls mit CAD-Kommandos möglich.

Bei der Erstellung des Programmes wurde der Verfasser von Herrn cand. ing. J. Franke unterstützt.

7. Integration in ein CIM-Konzept für die Montage

7.1 Erläuterungen zu CIM

Wohl kaum ein Begriff aus dem Bereich der Produktionstechnik hat bislang so sehr für Schlagzeilen gesorgt wie der des "Computer Integrated Manufacturing" (CIM). Trotz oder vielleicht gerade wegen dieser Publizität kann aber eine einheitliche Definition dessen, was unter einer "Rechnerintegrierten Produktion" zu verstehen ist, bislang nicht gegeben werden. Auch werden verschiedentlich die Schwerpunkte und Akzente zwischen den Teilbereichen unterschiedlich gelegt. Einigkeit besteht nur darin, daß im Rahmen von CIM die existierenden rechnergestützten Automatisierungssysteme wie CAD, CAP, PPS, CAQ und CAM miteinander kommunizieren sollen [96] [97].

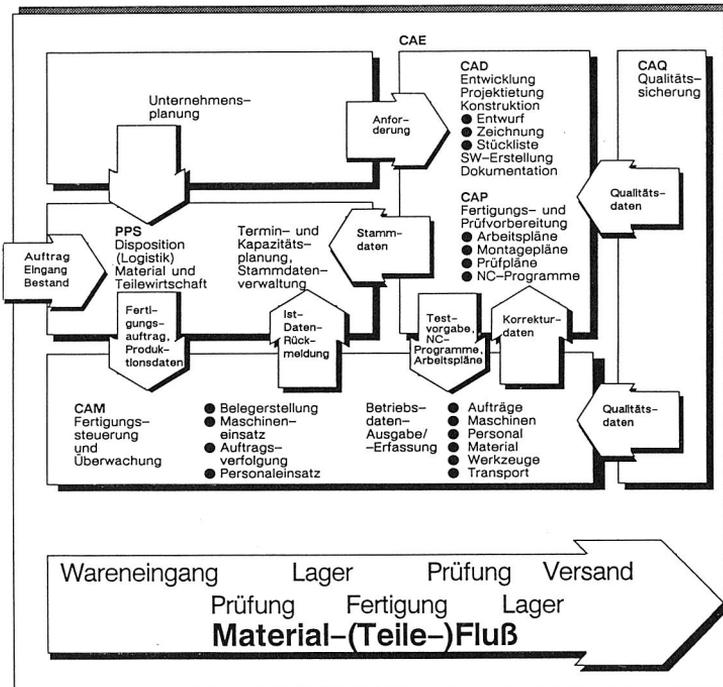


Bild 91: Funktionen des "Computer Integrated Manufacturing"
(Quelle: Siemens)

Erst eine solche Integration läßt maximale Produktivitätssteigerungen erwarten. Bild 92 zeigt, daß diese Steigerungen zwar später einsetzen, dafür aber einen höheren Nutzen bringen als einzelne oder multiple, nicht integrierte Automatisierungsinself. Der verspätete Nutzeinsatz hat seine Ursachen in dem erhöhten Aufwand zur Vorbereitung und Einführung eines CIM-Systems.

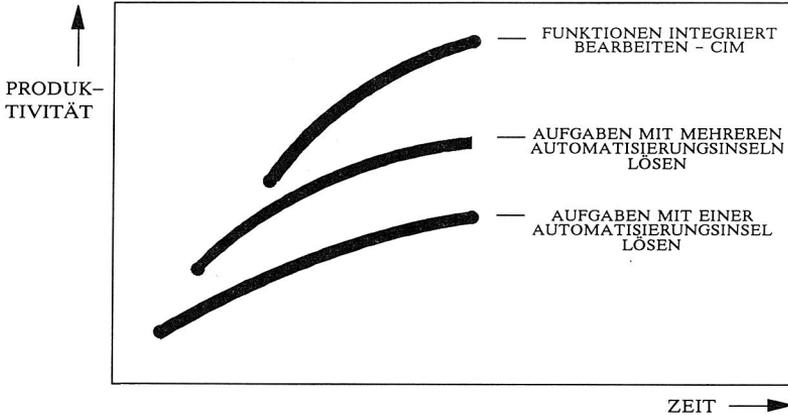


Bild 92: Produktivitätssteigerung durch Integration

Einigkeit besteht darüber hinaus auch in der Einsicht, daß der Begriff CIM gegenwärtig noch nicht Realität ist, sondern vielmehr ein Ziel oder eine Strategie verkörpert. Die Hemmnisse, die diesem Ziel entgegenstehen, sind von vielschichtiger Natur. So sind die einzelnen Automatisierungsinself im Laufe der letzten Jahren entstanden, ohne daß dabei Integrationsaspekte berücksichtigt wurden. Dementsprechend schwierig wird es dann, die Integration "gewachsener" Programmsysteme im nachhinein durchzuführen. Eine Ausnahme macht hierbei nur die CAD-NC-Kopplung. Schon frühzeitig hat man daran gedacht, die geometrischen Daten des CAD-Modells auch für die Programmierung der numerisch gesteuerten Bearbeitungsmaschinen zu nutzen [98]. Speziell die Automobilindustrie hat hierbei eine Vorreiterfunktion übernommen [99].

Das zweite Hemmnis auf dem Wege zu CIM ist das Fehlen eines ge-

eigneten Integrationsmediums. In den Produktionsstätten findet man eine Vielzahl von Computersystemen und Gerätesteuern unterschiedlicher Funktionalität und Komplexität vor. Bis auf wenige und teure Ausnahmen können diese Systeme aber auch von ihrer systemtechnischen Seite her nicht miteinander kommunizieren. Der Ausweg aus diesem Dilemma wird von vielen Leuten in den Standardisierungsbemühungen des MANUFACTURING AUTOMATION PROTOCOL (MAP) [100] gesehen (Bild 93). Nach einer Initiative der Firma GENERAL MOTORS wollen sich viele Computerhersteller auf ein gemeinsames Übertragungsprotokoll im CIM-Bereich einigen.

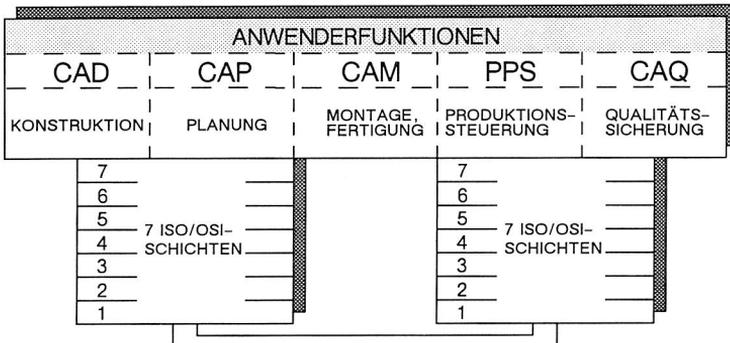


Bild 93: MAP-Anwenderfunktionen auf dem ISO/OSI-Schichtenmodell
(Quelle: [100])

Für das weitere Verständnis soll nun unter CIM als kleinster gemeinsamer Nenner der existierenden Ansichten die folgende Zusammenfassung gelten [101] [102]:

"CIM ist eine konzeptionelle Lösung für die Integration der Entwicklungs-, Planungs- und Fertigungsprozesse durch bereichsübergreifende Informationssysteme und Anwendungslösungen."

Nach dieser Auffassung kann man den CIM-Begriff aus einer systemorientierten und einer anwendungsorientierten Sicht betrachten.

7.2 Integration in das Anwendungskonzept

Das Anwendungskonzept der "Rechnerintegrierten Montage" umfaßt eine Kette von Tätigkeiten und Verfahren, die bislang noch nicht durchgängig rechnerunterstützt ablaufen. Es erstreckt sich auf die Gebiete der Grob- und Feinplanung von Montageanlagen, sowie deren rechnergeführten Betrieb (Bild 94). Damit sind drei verschiedene Funktionsbereiche eines Unternehmens betroffen:

- die Produktkonstruktion,
- die Fertigungsmittelplanung und
- der Fertigungsbetrieb.

Nicht ausführlich betrachtet werden in diesem Zusammenhang Fragestellungen der Qualitätssicherung und Diagnose sowie die Verwaltung der Fertigungsaufträge im Rahmen einer Produktionsplanung und Steuerung. Alle diese Themen fallen in den Bereich "Fertigungsbetrieb" und sollen hier nur der Vollständigkeit halber erwähnt werden.

Ausgangspunkt der Verfahrenskette ist die Produktkonstruktion, wo das zu montierende Produkt entworfen, konzipiert und detailliert wird. Sinnvollerweise arbeitet man dort mit einem dreidimensionalen und volumenorientierten CAD-System und kann damit das konstruierte Produkt in seiner Gestalt rechnerintern vollständig erfassen (Bild 95).

Der Teilekonstrukteur und der Planer der Montageanlage diskutieren anschließend die einzelnen Bearbeitungs- und Fügevorgänge, wie sie zur Montage des Produktes notwendig werden. Hierbei benutzen sie wieder das CAD-System. Sie simulieren die anfallenden Fügevorgänge durch Bewegen der Einzelteile entlang der vorgesehenen Füge-richtung. Der Montageexperte erkennt so sehr schnell die Montagefreundlichkeit des Produkts und kann notwendige Änderungen an der Produktgestalt veranlassen. Diese Änderungen, wie beispielsweise das Anbringen oder Variieren von Fügefasen, können dann wieder direkt am CAD-Bildschirm vorweggenommen und beurteilt werden, ohne daß entsprechender Aufwand in Änderungsdiensten bei

den Konstruktionsunterlagen notwendig wird.

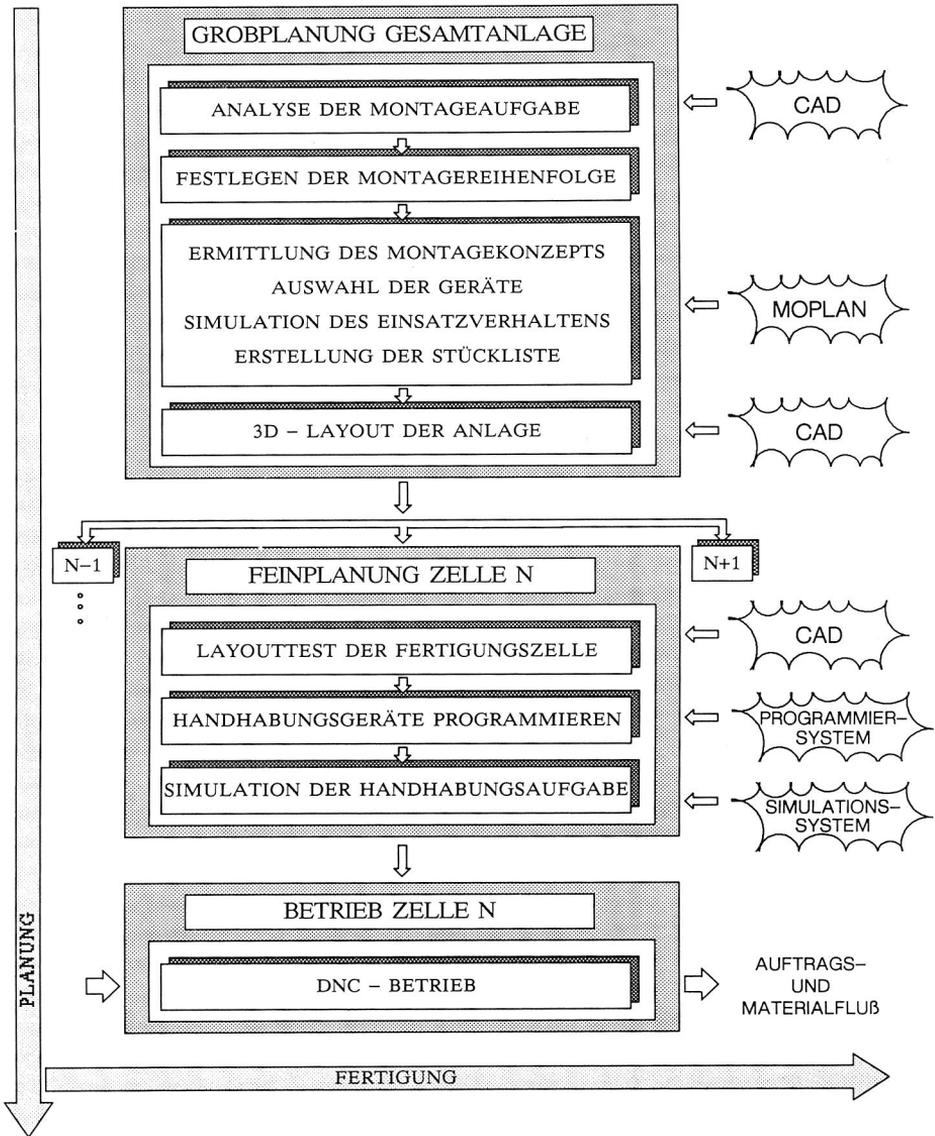


Bild 94: Anwendungskonzept "Rechnerintegrierte Montage"

In der nächsten Phase ergibt sich die Aufgabe, eine optimale Struktur und Reihenfolge der Montageoperationen zu finden. Es wird also untersucht, in welcher Reihenfolge die einzelnen Montageoperationen ablaufen sollen und wie man die Operationen sinnvoll in Vormontage und Endmontage gliedert. In vielen Fällen

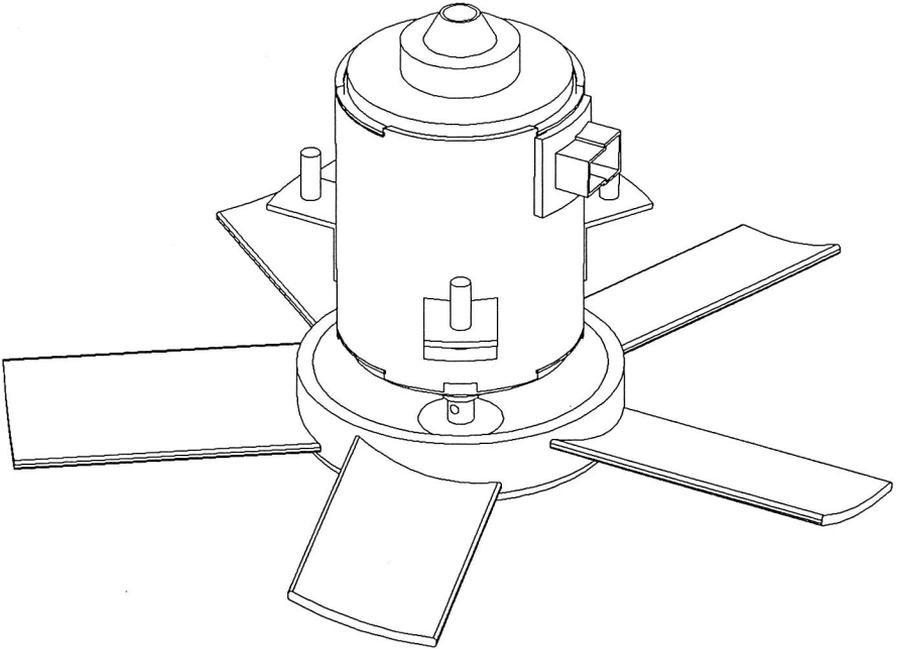


Bild 95: 3D-Produktmodell (Elektromotor mit Lüfter)

geben Gestalt und Struktur des zu montierenden Produkts auf diese Fragen Antwort. Auch hier bietet das CAD-System wirksame Unterstützung. So können alle Fügeflächen und Fügeoperationen im Rahmen einer Explosionsdarstellung gezeigt werden (Bild 96).

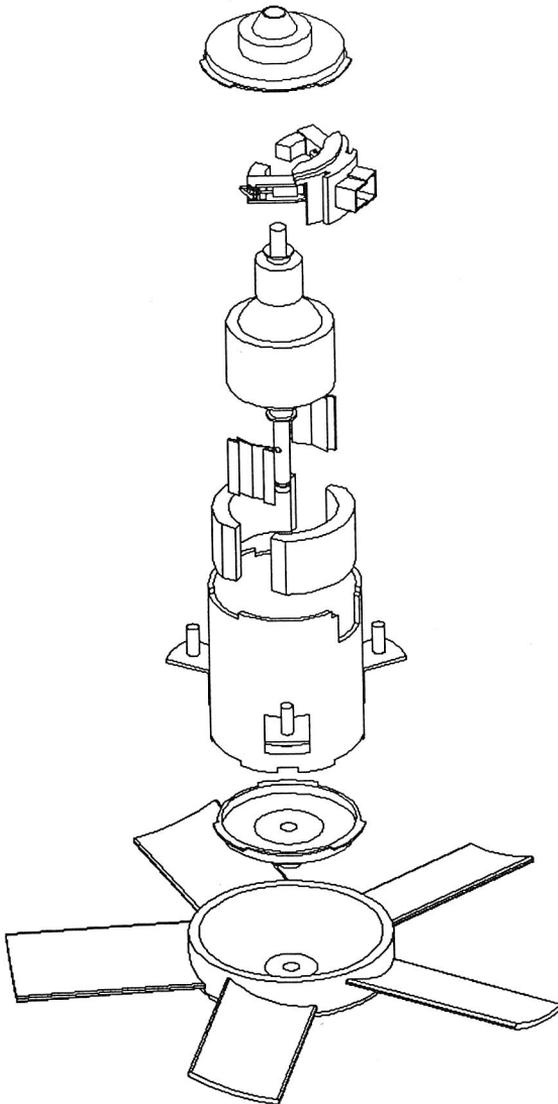


Bild 96: Explosionsdarstellung

Der Fügeflächengraph (Bild 97) zeigt dann, welche Flächen der Einzelteile mit anderen Flächen eine Fügekombination bilden. Der Montageplaner kann hieraus Rückschlüsse für die Auswahl geeigneter

ter Basisteile und die Strukturierung des Ablaufs ziehen.

In diesem Zusammenhang sind bereits mathematische Ansätze von GAIROLA [103] [104] bekannt, die vom Fügeflächengraph ausgehen und sogenannte Fügeflächenmatrizen aufstellen. Dort ist eingetragen, welche Flächen eine Fügekombination in x-, y- oder z-Richtung mit anderen Flächen eingehen. Aus den Matrizen entsteht ein mathematisches Gleichungssystem, welches den einzelnen Füge-teilen Wertungsfaktoren zuweist. Über diese Wertungsfaktoren kann dann ein gerichteter Montagegraph ermittelt werden, der eindeutige Aussagen zur optimalen Reihenfolge und Strukturierung der Montageoperationen macht. Für diesen Ansatz von GAIROLA ist ein Programmsystem denkbar, welches die Fügeflächen im CAD-System automatisch erfaßt, die Fügeflächenmatrizen aufstellt, daraus das Gleichungssystem ableitet und auswertet und so zum endgültigen Montagegraph gelangt.

Allerdings handelt es sich hierbei um einen theoretischen Ansatz, der die immer wieder anzutreffenden Besonderheiten und Ausnahmefälle der Praxis nicht berücksichtigen kann. Demzufolge ist für die nächste Zukunft wohl anzunehmen, daß der Montageplaner eher im Zuge einer interaktiven Arbeitsweise mit dem CAD-System zu dem Montagegraphen (Bild 97) gelangt, daß dieser Vorgang aber nicht voll automatisiert werden kann.

Zur letztendlichen Darstellung der Montagereihenfolge kann durch eine Art Abwicklung des Montagegraphen ein sog. Vorranggraph erstellt werden. In ihm sind die Montageoperationen, die dabei beteiligten Teile und ihre Reihenfolge beschrieben. Das Prinzip des Vorranggraphen stammt aus der Netzplantechnik. Seine Erstellung hat das Ziel, für Programmsysteme zur Austaktung von Montageanlagen eine geeignete Eingabe bereitzustellen. Im Rahmen des vorliegenden CIM-Konzeptes werden solche Programmsysteme nicht beachtet, da das Expertensystem MOPLAN diese Aufgaben wahrnimmt. Der Vorranggraph hat also hier nur dokumentarische Bedeutung.

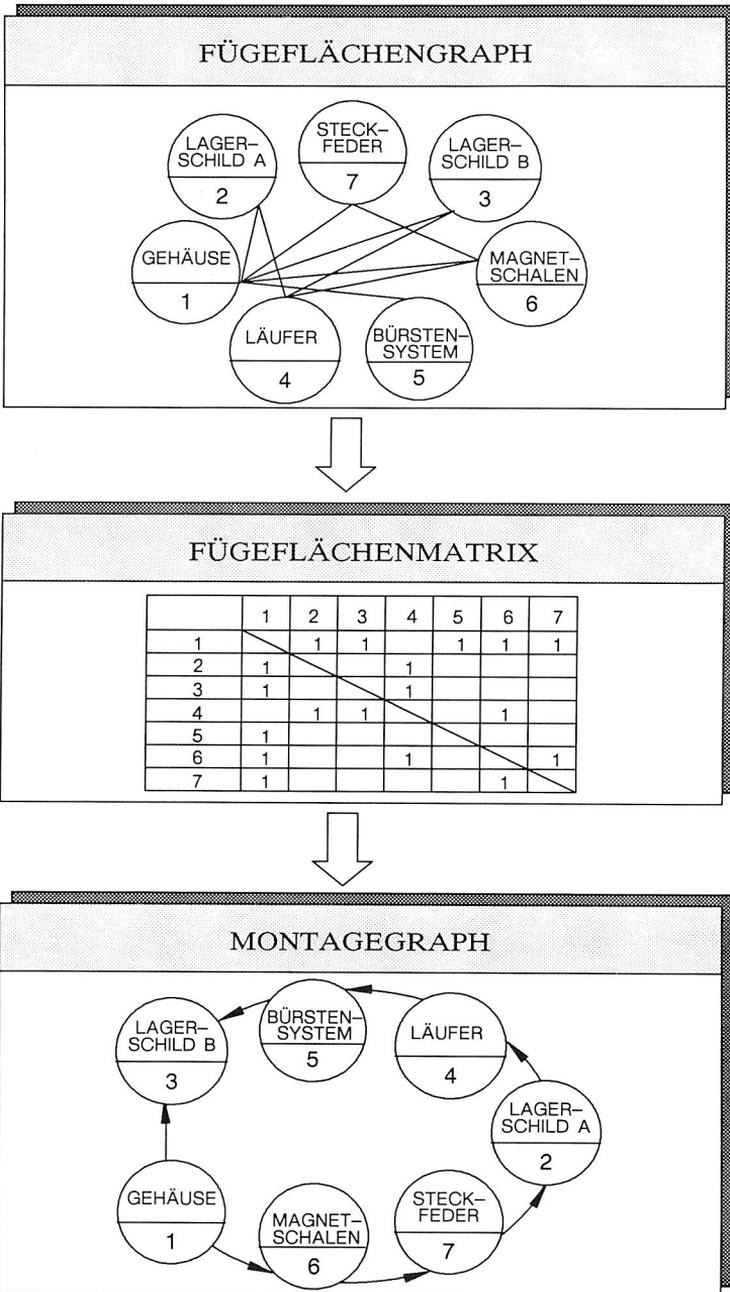


Bild 97: Ermittlung der Montagereihenfolge

Die nun folgenden Tätigkeiten im Planungsablauf werden von dem Expertensystem MOPLAN übernommen. Die Funktionalität des Systems ist in Kap. 5.3 eingehend beschrieben und soll hier nur der Vollständigkeit halber kurz angesprochen werden. Ausgehend von einer interaktiven Beschreibung der Montageaufgabe werden Montagekonzepte ermittelt, die die Festlegung der Organisationsform (Linie, Nest) der Montageanlage und des Montageprinzips (manuell, automatisch) beinhalten. Die Auswahl der Gerätetypen und Geräteausprägungen zur Lösung der Montageaufgabe schließt sich an. Die Überprüfung des Einsatzverhaltens der Anlage und die Beseitigung von Schwachstellen erfolgt über eine integrierte ereignisorientierte Simulation. Die Optimierung ist über Regeln steuerbar. Abschließend wird das Layout der Anlage geplant und eine Stückliste erstellt. Der Aufbau des dreidimensionalen Layouts erfolgt automatisch über die in Kap. 6.4 beschriebene CAD-Kopplung. Damit ist die Phase der Grobplanung abgeschlossen.

Es schließt sich nun die Phase der Feinplanung an, die im Gegensatz zur Grobplanung stark zellenorientiert ist. Zu Beginn der Tätigkeiten muß das bereits vorliegende Groblayout noch verfeinert und interaktiv optimiert werden. Sicherlich sind hierbei die Positionen der einzelnen Geräte noch zu verschieben. Kommen Industrieroboter oder sonstige programmierbare Handhabungsgeräte zum Einsatz, so sind verschiedene Layouttests durchzuführen. Wichtig ist hierbei, daß alle erforderlichen Raumpunkte und Bahnen angefahren werden können. Zu diesem Zwecke werden alle Achsen der Handhabungsgeräte entsprechend ihrem Freiheitsgrad gedreht bzw. verschoben. Die Erreichbarkeit von Raumpunkten, an denen beispielsweise Werkstücke von Peripheriegeräten bereitgestellt werden, läßt sich über eine Koordinatenabfrage am Tool Center Point (TCP) einfach überprüfen. Etwaige Kollisionen mit anderen Körpern können durch eine entsprechende CAD-Funktion automatisch erkannt werden. Unter Bezug auf ein kinematisches oder gar dynamisches Modell der Handhabungsmechanismen werden die Verfahrzeiten berechnet und die Gelenkwinkelstellungen ermittelt.

Die eigentliche Geräteprogrammierung kann off-line über ein entsprechendes Programmiersystem erfolgen. Man unterscheidet hier

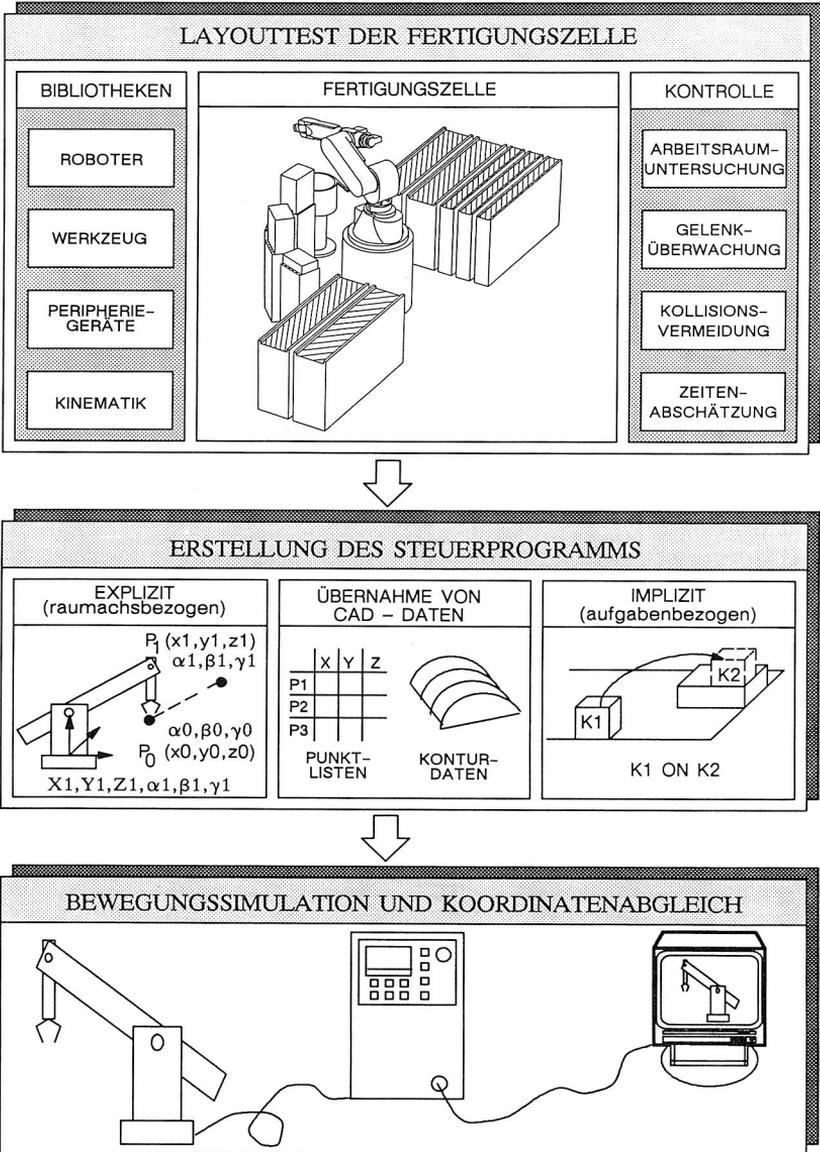


Bild 98: Detailplanung von Montageanlagen

zwischen den sogenannten expliziten und den impliziten Verfahren. Während sich letztere noch im Forschungsstadium befinden, sind explizite Verfahren schon verfügbar. Im Gegensatz zur impliziten Programmierung müssen hierbei die Geometrie und Technologieanweisungen noch explizit eingegeben werden.

Zur Überprüfung und Darstellung des Ablaufs von Steuerprogrammen wird ein Simulationslauf notwendig. Hierbei können die Bewegungen der Handhabungsgeräte (Mechanismen) am Bildschirm graphisch dargestellt und nachvollzogen werden. In der letzteren Phase der Detailplanung mißt der Roboter die reale Fertigungszelle selbst aus, um somit einen Abgleich der realen Koordinaten mit den offline programmierten Angaben zu erhalten. Die Vorgänge sind in Bild 98 dargestellt.

An dieser Stelle sind die Planungsaktivitäten abgeschlossen. Die Montageanlage kann in den Auftrags- und Materialfluß einer Computerintegrierten Produktion eingegliedert werden.

7.3 Integration in die Softwareumgebung

Das Softwarekonzept für die in Kap. 7.2 beschriebenen Anwendungen der "Rechnerintegrierten Montage" ist in Bild 99 dargestellt. Danach verfügt das Expertensystem MOPLAN über eine eigene Wissensbasis, in der das gesamte Planungswissen abgelegt ist (vgl. Kap. 5.4). Darunter fallen auch die Stammdaten der einzelnen Geräte und Maschinen. Im Sinne einer betrieblich gesehen einheitlichen und zentralen Datenhaltung ist hier der Einsatz eines Datenbankverwaltungssystems zur Stammdatenverwaltung denkbar [11]. Als Vorteile einer solchen Integration lassen sich aufzählen:

- Erhöhte Konsistenz der innerbetrieblichen Datenbestände,
- Vermeidung von redundanten Datenbeständen,
- Senkung des Wartungs- und Aktualisierungsaufwands.

Die Kopplung mit einem CAD-System wurde bereits in Kap. 6.4 be-

geschrieben. Das CAD-System verfügt über ein eigenes Verwaltungssystem für lokal benötigte objektdefinierende Daten. Diese objektdefinierende Daten bestehen im wesentlichen aus gestaltbestimmenden Daten (Gestalt = Topologie + Geometrie) und den nicht-geometrischen Attributen. Letztere sollten im Rahmen dieses Konzepts nicht zur Anwendung kommen. Dispositive und organisatorische Daten sollten vielmehr vom Datenbankmanagementsystem des Planungsbereichs verwaltet werden.

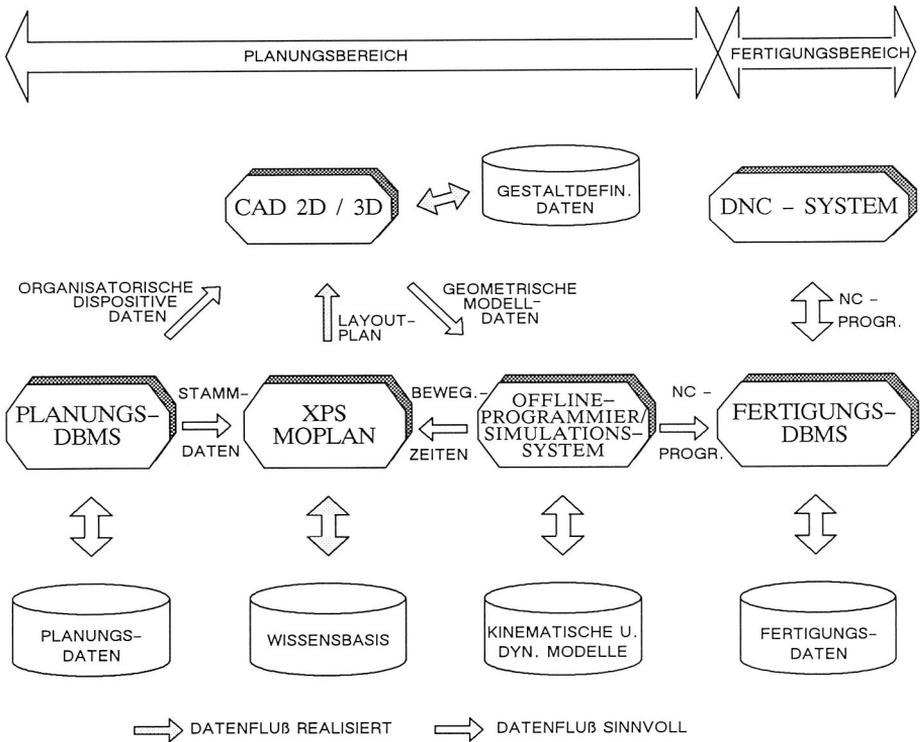


Bild 99: Softwarekonzept für die Rechnerintegrierte Montage

Bei der letzten Komponente des Planungsbereichs handelt es sich um ein Programmier- und Simulationssystem. Mit ihm werden die Aufgaben der Feinplanung durchgeführt. Das rechnerinterne Modell

dieses Systems greift auf kinematische und in Zukunft auch dynamische Daten der Handhabungsgeräte zurück. Wesentlich ist hierbei, daß diese Bestände nicht vom CAD-System verwaltet werden. Bei einer Bewegungssimulation greift vielmehr das Simulationssystem auf die gestaltbestimmenden Daten des CAD-Systems zurück und baut daraus das Simulationsmodell (Kantenmodell) auf.

Das fertige Steuerprogramm kann nach dem Simulationstest dem Fertigungsbereich übergeben werden. Heutige Gerätesteuierungen erlauben einen solchen DNC-Betrieb nur in einem sehr begrenzten Umfang. Die Zukunft könnte so aussehen, daß hier ein spezielles Datenbankverwaltungssystem für den Fertigungsbereich existiert [105]. Dieses könnte dann neben den Fertigungsaufträgen auch die Steuerprogramme für die Geräte verwalten und ein entsprechendes DNC-System damit versorgen.

Wie aus den Ausführungen deutlich hervorgeht, ist dieses Konzept sehr stark in die Zukunft ausgerichtet und bisher nur für zwei Teilbereiche realisiert. Dies entspricht aber auch voll und ganz der eingangs erwähnten Feststellung, daß es sich bei CIM nicht um eine Realität, sondern um ein zukünftiges Ziel handelt.

7.4 Integration in die Systemumgebung

Die Systemumgebung im Bereich der Computerintegrierten Produktion ist in hohem Maße komplex und inhomogen. So findet man eine Vielzahl von Rechner-, Steuerungs- und Überwachungssystemen mit unterschiedlicher Leistungsfähigkeit. In vielen Fällen bilden sie getrennte Automatisierungsinselfn, wobei teilweise auch Funktionsüberschneidungen auftreten können. Zur Überwindung dieses Zustandes ist eine klare Systemstruktur notwendig (Bild 100).

Eine solche Struktur ist hierarchisch in verschiedene Funktionsebenen [102] getrennt. Auf der Konstruktions- und Planungsebene werden Zentralrechner in Verbindung mit Bereichsrechnern und autonomen Arbeitsstationen eingesetzt. Die Kommunikation dieser Systeme erfolgt meist über Lokale Netze (LAN) oder Direktver-

bindungen. Beispiele hierfür sind der TOKEN-Ring von DOMAIN oder DECNET von DIGITAL. Über spezielle Gateways werden Verbindungen zu anderen Rechnersystemen realisiert.

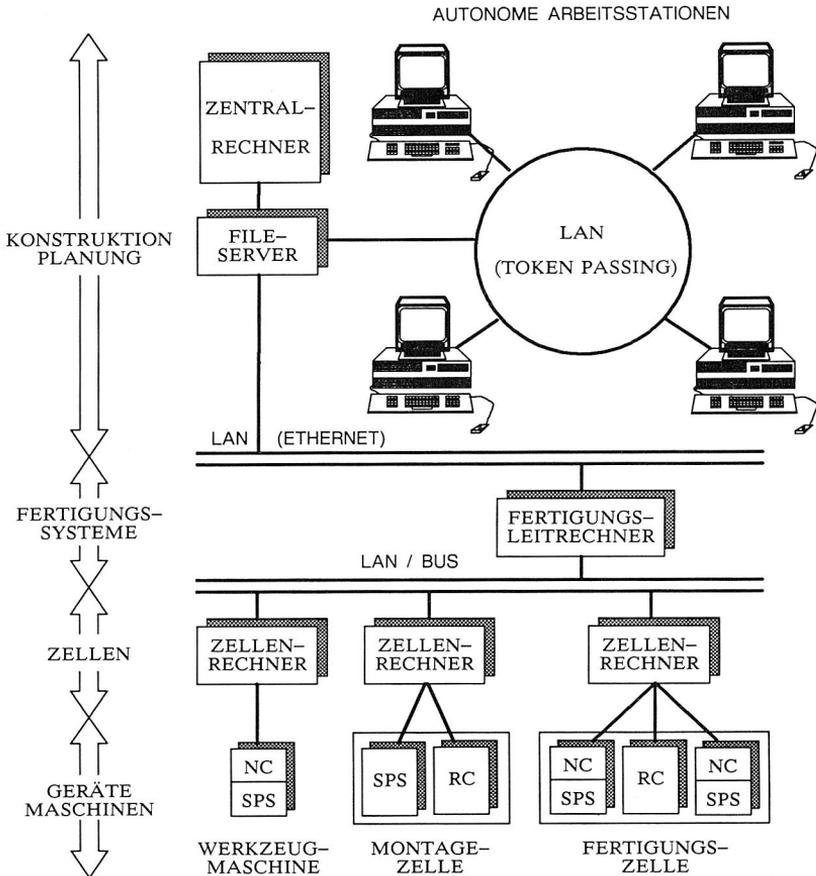


Bild 100: Systemstruktur für CIM

Auf der Ebene der Prozeßsteuerung kommen vielfach Prozeß- bzw. Fertigungsleitreehner zum Einsatz. Diese kommunizieren mit den untergeordneten Systemen der Zellenebenen und üben überwachende und steuernde Funktionen aus. Die Kommunikation erfolgt entweder

direkt oder über Lokale Netze (ETHERNET) und Bussysteme.

Die prinzipiellen Möglichkeiten der Integration und Kommunikation zwischen CIM-Komponenten wurden von SCHEER [106] in fünf Integrationsgrade eingeteilt (Bild 101). Auf der ersten Stufe liegt nur eine organisatorische Verbindung der CIM-Komponenten vor. Die jeweiligen Programm- und Datenbereiche bleiben voneinander getrennt. Die Kommunikation erfolgt über den Benutzer, der zwei Terminals am Arbeitsplatz bedienen muß. Angewendet auf Expertensysteme würde dies bedeuten, daß diese auf speziellen KI-Rechnern laufen und die CAD-Anwendungen auf speziellen CAD-Rechnern. Der Übergang von der Stückliste zum 3D-Layout müßte vom Benutzer realisiert werden.

Auf der zweiten Integrationsstufe sind die beiden Systeme nach wie vor getrennt, doch ist ein Personal Computer (PC) dazwischengeschaltet, auf dem spezielle Kommunikationssoftware abläuft. Diese Form hat den Nachteil, daß für die Verbindung zweier Systeme erst einmal ein komplexes drittes System geschaffen werden muß.

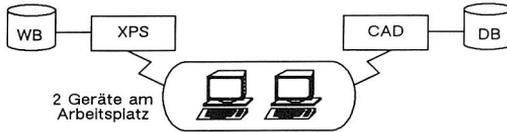
Die nächst höhere Integrationsstufe ist erreicht, wenn beide Systeme zwar noch über getrennte Datenbestände verfügen, aber schon auf dem gleichen Rechnersystem ablaufen. Die Kommunikation erfolgt über eine Art Briefkasten (Mailbox) im File-Transfer-Betrieb. Genau dieser Integrationsgrad wurde mit der MCL-Kopplung von MOPLAN und ROMULUS erreicht. Nach SCHEER ist dies der höchste Integrationsgrad, der unter einer Beibehaltung existierender Systemstrukturen derzeit erreichbar ist.

Die Stufen 4 und 5 sind in die Zukunft hin gerichtet. Gekennzeichnet sind sie durch gemeinsame Datenbestände und Zusammenwachsen der Softwaremodule durch eine Programmintegration. Ein Ansatz zur gemeinsamen Datenbasis bei der Integration von Expertensystemen ist sicherlich im Bereich der relationalen Datenbanksysteme zu finden. Expertensysteme auf der Basis solcher Datenbanksysteme werden derzeit entwickelt [107]. An dieser Stelle tritt nach Meinung des Verfassers eine zentrale Problematik bei

der industriellen Nutzung der Wissensverarbeitung zu Tage. Wie lassen sich in Zukunft vermehrt eingesetzte Datenbanken und Wissensbasen vernünftig organisieren und verwalten? Auf diesem Gebiet sind sicherlich noch viele Fragen zu klären.

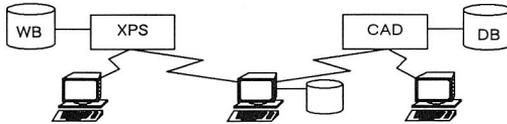
1. STUFE

ORGANISATORISCHE VERBINDUNG EDV - TECHNISCH UNVERBUNDENER SYSTEME



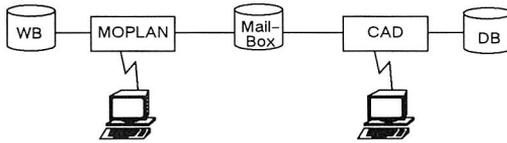
2. STUFE

INTEGRATION DER UNVERBUNDENEN SYSTEME DURCH TOOLS (PC, QUERY, ...)



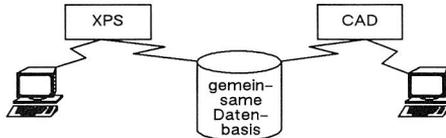
3. STUFE

DATEITRANSFER ZWISCHEN DEN SYSTEMEN



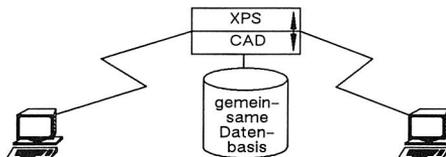
4. STUFE

GEMEINSAME DATENBASIS DER SYSTEME



5. STUFE

ANWENDUNG/ ANWENDUNG- BEZIEHUNG DURCH PROGRAMMINTEGRATION



WB : WISSENSBASIS; XPS : EXPERTENSYSTEM; DB : DATENBASIS

Bild 101: Integrationsgrade und -möglichkeiten (Quelle: [106])

8. Anwendungsbeispiel

Die Erprobung des Systems MOPLAN erfolgte anhand einer beispielhaften Anwendung aus der Modellfabrik des Lehrstuhls für Fertigungsautomatisierung und Produktionssystematik. In dieser Modellfabrik sind vielfältige und automatisierte Montagesysteme aufgebaut.

Anhand dieser Systeme werden Forschungsarbeiten zur Rechnerintegrierten Fabrik durchgeführt. Die Aufgaben erstrecken sich u.a. auf die Gebiete der Steuerungstechnik, der Sensortechnik, der Diagnose und der Verfügbarkeitssteigerung. Die einzelnen Montagesysteme haben schwerpunktmäßig Zellencharakter. Sie sind über ein fahrerloses Transportsystem vom Materialfluß her verbunden. Auf jedem der sog. Teilsysteme werden Montageaufgaben von unterschiedlichem Charakter durchgeführt. Dies bedingt zwangsläufig den Einsatz unterschiedlicher Montagekonzepte und unterschiedlicher gerätetechnischer Ausführungen.

Die Zielsetzung dieser Modellfabrik besteht also nicht in einer durchgängige Fertigung eines komplexen Produktes über mehrere Stationen hinweg. Vielmehr sollen unterschiedliche Technologien und Montagesysteme demonstriert werden, die weitgehend abgegrenzt und autark arbeiten.

Als Aufgabe für den Einsatz des Expertensystems MOPLAN wurde die Montage eines Elektromotors für einen PKW-Lüfter ausgewählt. Die Aufgabe gliedert sich in eine Vormontage und eine Hauptmontage. Im Rahmen der vorliegenden Arbeit wurde die Endmontage der in Bild 102 dargestellten Teile geplant und konfiguriert.

Die Bilder 103 bis 106 zeigen die Aufgabenbeschreibung und die Planungsergebnisse in Form von Stückliste und 3D-Layout für diese Musteraufgabe.

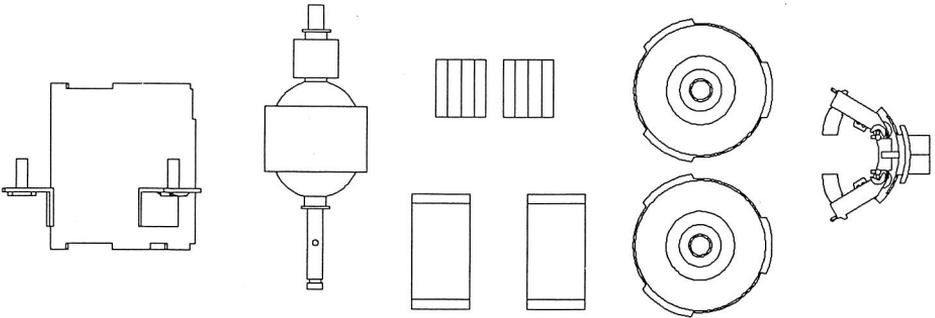


Bild 102: Hauptmontage des Elektromotors (Teile)

ALLGEMEINDATEN FUER DIE GESAMTANLAGE

ANLAGEN-BEZEICHNUNG : GESAMTMONTAGE-ELEKTROMOTOR

HAUPTMONTAGEN-BEZEICHNUNG: LUEFTERMOTOR

LEBENSDAUER (MONATE) : 60

MAX. KOSTEN (DM) : 250000

BETRIEBSZEIT (MINUTEN/TAG) : 900

AUSBRINGUNG (STUECK/TAG) : 810

NUTZUNGSGRAD (%) : 90

DECKENBELASTUNG (KP/M2) : -

RAUMLAENGE (MM): 5000

RAUMBREITE (MM): 5000

RAUMHOEHE (MM): 5000

ANGABEN ZU DEN ANSCHLUESSEN

Bild 103: Aufgabenbeschreibung "Allgemeindaten"

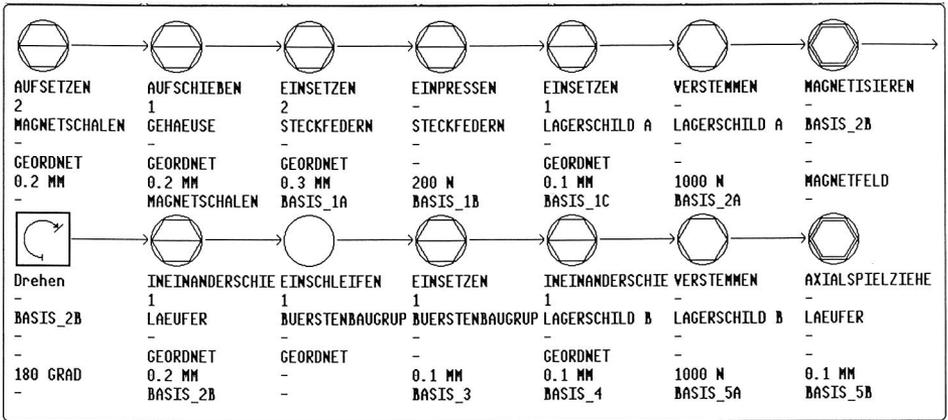


Bild 104: Aufgabenbeschreibung "Montagereihenfolge"

STÜCKLISTE MOTORMONTAGE

NR.	BEZEICHNUNG	ANZAHL
1	MANUTEC R3	1
2	MEHRFACHGREIFER	1
3	STANDFUß R3	1
4	STECKFEDEREINPREßVORRICHTUNG	1
5	MAGNETISIERVORRICHTUNG	1
6	VERSTEMM/AXIALSPIELZIEH-VORRICHTUNG	1
7	BÜRSTENEINSCHLEIFVORRICHTUNG	1
8	MODULARGESTELL	1
9	STEUERUNG MANUTEC R3	1
10	STEUERUNG SIMATIC S5	1
11	BOSCH_DOPPELGURTBAND_2000mm	4
12	BOSCH_DOPPELGURTBAND_800mm	8
13	HUBQUEREINHEIT	14
14	INDEXIERSTATION	8
15	CODESETZER	8
16	CODELESER	4
17	INDUKTIVSENSOR BERO	14
18	STOPPER	12
19	VERLADEBAHNHOF	4
20	GEHÄUSEWERKSTÜCKTRÄGER	20
21	MAGNETSCHALENWERKSTÜCKTRÄGER	10
22	LÄUFERWERKSTÜCKTRÄGER	20
23	LAGERDECKELWERKSTÜCKTRÄGER	20
24	STECKFEDERWERKSTÜCKTRÄGER	10
26	BÜRSTENSYSTEMWERKSTÜCKTRÄGER	20
27	MOTORENWERKSTÜCKTRÄGER	20
28	SCHUTZKETTENSTÄNDER INCL. KETTE	6

Bild 105: Stückliste

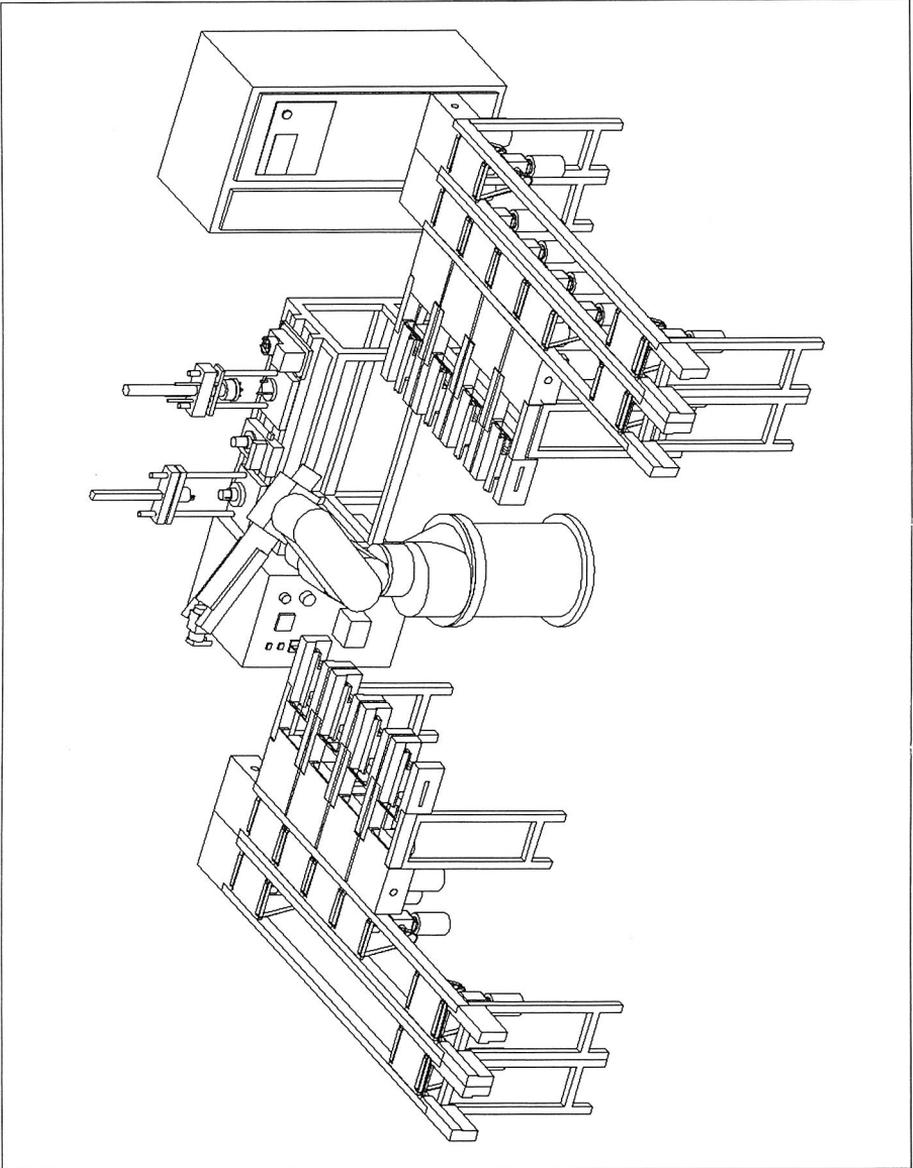


Bild 106: 3D-Layout der Fertigungszelle

9. Alternative Einsatzmöglichkeiten des wissensbasierten Werkzeugsystems (WWS)

Schon in der Einleitung der vorliegenden Arbeit wurde als Motivation und Zielsetzung die Erschließung von Innovationspotentialen in der Computerintegrierten Produktion herausgestellt. Die Entwicklung des Planungssystems MOPLAN ist in diesem Zusammenhang nur als exemplarische Realisierung einer Anwendung in diesem Bereich zu sehen. Weitere denkbare Ansätze sollen nun in den folgenden Abschnitten kurz angedeutet werden. Ein wesentlicher Vorteil wissensbasierter Rahmensysteme liegt in dem hohen Maß an Flexibilität begründet, das mit ihrer Hilfe erzielt werden kann. Diese Flexibilität erstreckt sich auf:

- die verschiedenen Technologien,
- die beteiligten Objekte und
- die anzuwendenden Planungsprozesse.

Sie wird durch interaktives Ändern der entsprechenden Wissensinhalte sowie durch Ergänzen oder Austauschen der Wissensbasen erreicht. Im Gegensatz dazu müssen bei den konventionellen Planungssystemen über einen Austausch von Daten hinaus auch noch Programme neu geschrieben werden.

9.1 Planung von Flexiblen Fertigungssystemen

Die Planung von Flexiblen Fertigungssystemen läuft in ähnlicher Weise ab wie die Planung von Montageanlagen [108]. Auch wurden hierfür bereits erste rechnergestützte Hilfsmittel in konventioneller Softwaretechnik entwickelt [109].

Ausgangspunkt ist das zu fertigende Produkt. Die Fertigungsschritte werden analysiert und zu einer Reihenfolge zusammengestellt. Zu jedem einzelnen Fertigungsschritt sind dann die entsprechenden Fertigungsmittel wie Maschinen, Geräte, Vorrichtungen und Werkzeuge zu planen (Bild 107).

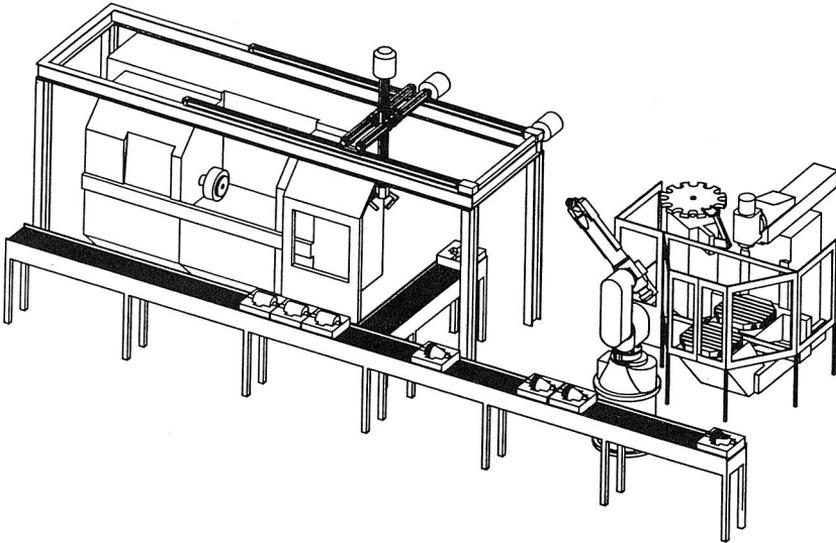


Bild 107: Beispiel einer Fertigungszelle

Ausschlaggebend für den planerischen Erfolg ist:

- das Finden einer geeigneten Anlagenstruktur,
- das Bereitstellen der benötigten Kapazitäten,
- das Sicherstellen einer definierten Verfügbarkeit,
- die Einhaltung des Kostenrahmens und
- die Gewährleistung der geplanten Systemausbringung.

Es ließen sich hier sicher noch weitere Kriterien aufstellen, doch wird auch schon so deutlich, daß sich die Zielsetzungen von denen der Montageplanung nicht wesentlich unterscheiden. Was die eingangs aufgestellten Flexibilitätskriterien wissensbasierter Rahmensysteme betrifft, so unterscheiden sich Fertigungssysteme von den Montagesystemen natürlich schon. Es sind andere meist spannende Technologien, andere Maschinen und Geräte und sicherlich auch andere Planungsstrategien zu berücksichtigen.

Aber die Beschreibung der Fertigungsaufgabe kann mit den bereits

vordefinierten Ikonen und den dazugehörigen Randbedingungen in gleicher Weise durchgeführt werden.

Die Technologien unterscheiden sich in den sie bestimmenden Parametern und Einflußgrößen. Diese werden in einer speziellen Technologiewissensbasis verwaltet und müssen neu ermittelt und integriert werden.

Die zu betrachtenden Objekte differieren einmal, was die Geräte angeht, und zum anderen, was die zu produzierenden Teile betrifft. Für beide Anforderungen bietet die Wissensakquisitionskomponente des Systems komfortable Möglichkeiten, um eine neue Teile- und Gerätewissensbasis zu erstellen. Letztendlich sind nur Framestrukturen und deren Inhalte neu zu definieren.

Die Bestimmung der Planungsprozesse, die Durchführung der Planungsaufgaben unter Vorgabe bestimmter Kriterien erfolgt mit Hilfe der Produktionsregeln. Auch sie müssen neu definiert werden.

Aber die Verwaltung der Wissensbasis, die interne Systemstruktur, die Steuerung des Problemlösungsvorganges und die Ergebnisausgaben in Form von Stücklisten und 3D-Layouts bleiben unverändert. Selbst der Simulator zur Ermittlung von Schwachstellen und zur Pufferoptimierung kann unverändert eingesetzt werden.

Es zeigt sich also, daß das Planungssystem auch für die Planung von Flexiblen Fertigungssystemen voll einsatzfähig ist, wenn man die Wissensinhalte zu den Technologien, Objekten und Prozessen jeweils ergänzt.

9.2 Planung von Fertigungsabläufen

Neben der Planung der Fertigungs- und Betriebsmittel gehört die Planung der eigentlichen Fertigungsabläufe in einer Computer-integrierten Produktion zu den essentiellen Bestandteilen der Arbeitsplanung. Dabei geht es im wesentlichen um die Erstellung

von Steuerprogrammen für numerisch gesteuerte Werkzeugmaschinen auf der einen und für Industrieroboter auf der anderen Seite.

Auf den ersten Blick unterscheiden sich die beiden Aufgaben ganz erheblich von denen der Fertigungsmittelplanung. In der Tat werden auch ganz andere Ergebnisse verlangt. Es ist hier keine Stückliste und kein 3D-Layout gefordert. Vielmehr wird eine Handlungsvorschrift benötigt, die

- sequentiell aufgebaut ist,
- aus elementaren Einzelanweisungen besteht und
- Anforderungen hinsichtlich Format und Codierung

zu genügen hat.

Ausgangspunkt der Planungsaufgabe ist aber wieder das zu fertigende Produkt. Die einzelnen Fertigungsschritte werden in ihrer Reihenfolge festgelegt. Technologische und organisatorische Kenn-
daten wie Schnittwerte und Werkstoffeigenschaften legen die Rahmenbedingungen für die Planung fest. In Abhängigkeit von der Reihenfolge und den existierenden Randbedingungen wird ein Steuerprogramm erstellt. Die notwendigen Werkzeuge werden ermittelt und die Fertigungszeiten berechnet.

Bei genauerer Betrachtungsweise fällt auf, daß die Planungssystematik eigentlich die gleiche ist wie bei der Fertigungsplanung. Auch hier handelt es sich um eine Konfigurationsaufgabe unter vielschichtigen Randbedingungen. Auch hier kann die Fertigungsreihenfolge mit Hilfe der Dialogkomponente komfortabel beschrieben werden. Faßt man darüber hinaus die verschiedenen Einzeloperationen noch als Objekte auf, so bietet das Wissensrepräsentationssystem Möglichkeiten, diese Objekte zu repräsentieren und zu verwalten. Das gleiche gilt für spezielles Wissen über Technologieparameter wie Schnittwerte oder Werkstoffeigenschaften.

Die Systematik, nach der jetzt zur Erstellung einer Anweisungsfolge vorgegangen wird, sowie die Kriterien und Randbedingungen,

die dabei zu berücksichtigen sind, können wieder in Form von Produktionsregeln beschrieben werden. Diese Regeln konfigurieren dann die Anweisungsfolge aus den in Form von Objekt-Frames vorliegenden Einzelanweisungen. Die Umwandlung der Anweisungsliste in ein für Roboter- bzw. Maschinensteuerungen lesbares Format kann nachfolgend von einem internen oder externen Programm übernommen werden (Bild 108).

Kennzeichnend für diesen Ansatz ist, daß Einzelanweisungen als Objekte verstanden werden, und daß Produktionsregeln die Konfiguration dieser Einzelanweisungen zu einem Gesamtprogramm durchführen und steuern. Ansätze dieser Art werden bereits international weiterverfolgt [2] [110] [111].

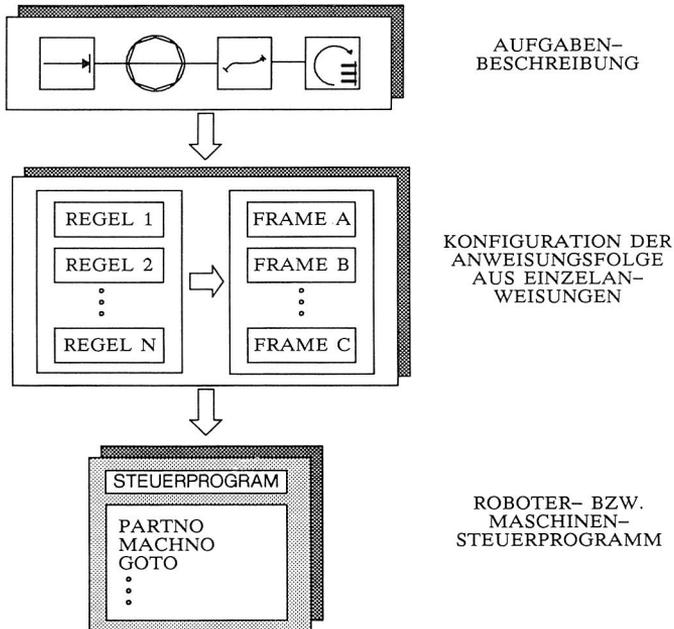


Bild 108: Ablauf einer impliziten Programmierung

10. Zusammenfassung

Die Realisierung einer Rechnergeführten Fabrik unter dem Schlagwort CIM ist eine der größten Herausforderungen für die industrielle Produktionstechnik. Komplexe Informations- und Automatisierungssysteme steuern und überwachen die Fabrik der Zukunft. Doch die konventionelle Informations- und Datenverarbeitung erreicht ihre Grenzen dort, wo Wissen und Erfahrung zur Problemlösung im Vordergrund steht, und wo komplexe, unstrukturierte und nicht algorithmierbare Zusammenhänge angetroffen werden. Hier eröffnen die Methoden der Künstlichen Intelligenz und Wissensverarbeitung vielfältig neue Möglichkeiten. Unter diesen Randbedingungen will die vorliegende Arbeit Innovationspotentiale in der Rechnerintegrierten Produktion durch den Einsatz wissensbasierter Systeme erschließen.

Dazu werden eingangs die grundsätzlichen Methoden und Hilfsmittel der Wissensverarbeitung erläutert. Diese Ausführungen erstrecken sich auf den Wissensbegriff selbst, auf die Methoden zur Wissensrepräsentation, Manipulation und auch Akquisition. Eine grobe Klassifizierung der Softwarehilfsmittel in Programmiersprachen und Werkzeugsysteme schließt sich an.

Das nächste Kapitel beschäftigt sich mit dem Einsatz wissensbasierter Systeme in der Produktion allgemein. Erfolgreiche Systeme und interessante Prototypen aus den Anwendungsgebieten Diagnose, Arbeitsplanung, Konstruktion und Simulation werden vorgestellt. Die Wissensverarbeitung erfordert eine neue Qualifikation an Engineeringleistung. Die Aufgaben eines Wissensingenieurs werden im Zusammenhang mit dem Entwicklungsprozeß von wissensbasierten Systemen erläutert.

Im anschließenden Kapitel wird ein wissensbasiertes Rahmensystem (WWS) für die Lösung von Planungs- und Konfigurationsaufgaben vorgestellt. Es besteht aus Komponenten für den Dialog, für die Wissensrepräsentation, für die Problemlösung und für den Wissenserwerb. Ein ereignisorientiertes Simulationssystem ist in die Problemlösungskomponente voll integriert. Mit Hilfe dieser logi-

schen und programmtechnischen Integration von Konfigurations- und Simulationswerkzeugen ist es erstmals gelungen, völlig neue Möglichkeiten der Optimierung von Planungstätigkeiten in einem ganzheitlichen und wissensbasierten Ansatz zu erschließen.

Innerhalb der industriellen Produktion gilt die Montagetechnik als weitgehend unerschlossenes Rationalisierungspotential. Als exemplarische Anwendung des wissensbasierten Werkzeugsystems (WWS) wurde das Expertensystem MOPLAN zur Planung von Montageanlagen implementiert. Als einziges System seiner Art ist es hardware- und softwareseitig voll in ein CIM-Konzept für die Montage integriert und kommuniziert mit einem dreidimensionalen Modellierer (ROMULUS). Damit steht der Montageplanung erstmals ein rechnergestütztes Werkzeug zur Verfügung, das für einen Großteil der Aufgaben bei der Grobplanung eingesetzt werden kann.

Das letzte Kapitel beschäftigt sich mit alternativen Einsatzmöglichkeiten für das wissensbasierte Werkzeugsystem WWS. Hier ist in erster Linie die Planung von produktionstechnischen Anlagen im allgemeinen und die Planung von Flexiblen Fertigungssystemen im speziellen zu nennen. Aber auch zur Planung von Fertigungsabläufen und Fertigungsaufträgen kann das Werkzeug eingesetzt werden. Für die implizite offline-Programmierung von Industrierobotern wird hierzu ein Beispiel gegeben.

Die vorliegende Arbeit zeigt das Spektrum der Einsatzmöglichkeiten wissensbasierter Systeme in einer Rechnerintegrierten Produktion auf. Angefangen bei der Konstruktion, über die Fertigungsplanung und -steuerung, bis hin zur Diagnose können mit Hilfe von wissensbasierten Konzepten vielfältige Innovationspotentiale erschlossen werden. Es wird deutlich, daß die Wissensverarbeitung eine wesentliche Komponente in der Fabrik der Zukunft darstellt.

Mit dem Rahmensystem WWS und dem Expertensystem MOPLAN ist es gelungen, breit einsetzbare Werkzeuge als Basis für viele weiterführende Arbeiten im Bereich der Planung und Konfiguration zu schaffen. Damit wird auch ein Beitrag dazu geleistet, die Wissensverarbeitung in Forschung und Lehre zu etablieren.

11. Literaturverzeichnis

1. Spur, G.: Technischer Informationsfluß beim Einsatz der EDV.
WT-Zeitschrift für industrielle Fertigung, 73 (1983), Nr.7, S.435-445.

2. Rembold, U;
Blume, C.;
Dillmann, R.: Computerintegrated Manufacturing Technology and Systems.
Marcel Dekker Inc., New York, 1985.

3. Feldmann, K.: Rechnerunterstütztes Konstruieren im Fertigungsmittelbau.
VDI-Bericht Nr.473, S. 41-53,
VDI-Verlag, Düsseldorf, 1982.

4. Feldmann, K.: Rechnerunterstützte Planungsmethoden für Montagesysteme.
Tagungsband zum Kolloquium "Automatische Produktionssysteme", TU München, S. 414-426, 1985.

5. Bullinger, H.-J.;
Warneke, H.J.;
Lentes, H.-P.: Toward the Factory of the Future.
Toward the Factory of the Future, S. 29-54, Springer Verlag, Berlin, 1985.

6. Simons, G.L.: Die Fünfte Computergeneration.
Carl Hanser Verlag, München, 1986.

7. Isensee, S.: Forschungsförderung des BMFT auf dem Gebiet der Wissensverarbeitung und Mustererkennung.
Informatik Fachberichte 112, Wissensbasierte Systeme,
Springer Verlag, Berlin, 1985.

8. Stoyan, H.: Einführung in die Probleme der Künstlichen Intelligenz.
Vorlesung am Institut für Mathematische Maschinen und Datenverarbeitung, Universität Erlangen, 1986.

9. Lehmann, E.;
Schwärtzel, H.;
Schweppe, H.: Industrielle Nutzung wissensbasierter Systeme.
Informatik Fachberichte 112, S. 347-369, Springer Verlag, Berlin, 1985.

10. Scharf, A: Künstliche Intelligenz.
Hard and Soft, Industrielle Mikroelektronik, 0/85, S. 31-35,
VDI-Verlag, Düsseldorf, 1985.
11. Wedekind, H.: Datenbanksysteme I.
BI Verlag, Mannheim, 1981.
12. Barr, A.; Feigenbaum, E.A.: The Handbook of Artificial Intelligence.
Vol.I, Pitman Books Limited, London, 1981.
13. Winston, P.H.: Artificial Intelligence.
Second Edition, Addison-Wesley Publishing Company, London, 1984.
14. Nilsson, N.J.: Principles of Artificial Intelligence.
Springer Verlag, Berlin, 1982.
15. Trost, H.: Wissensrepräsentation in der AI am
Beispiel Semantischer Netze.
Artificial Intelligence, S. 47-72,
Teubner Verlag, Stuttgart, 1984.
16. Clocksin, W.F.; Mellish, C.S.: Programming in Prolog.
Springer Verlag, Berlin, 1984.
17. Brownston, L.; Farrell, R.; Kant, E.: Programming Expert Systems in OPS 5.
Addison-Wesley Publishing Company, Reading, MA, 1985.
18. Minsky, M.: A Framework for Representing Knowledge.
The Psychology of Computer Vision,
Mc Graw Hill, New York, 1975.
19. Zadeh, L.A. A Formalization of Commonsense Reasoning Based on FUZZY Logic.
Informatik Fachberichte 112,
S. 398-402, Springer Verlag,
Berlin, 1985.
20. Lehmann, E.: Expertensysteme - Überblick über den
aktuellen Entwicklungsstand.
BMFT - geförderter Bericht 083 0616,
Siemens AG, München, 1983.

21. Buchanan, B.;
Shortliffe, E.: The MYCIN Experiments of the Stanford
Heuristic Programming Project.
Rule-Based Expert Systems,
Addison-Wesley, London, 1984.
22. Raulefs, P.: Expertensysteme.
Informatik Fachberichte 59,
S. 61-98, Springer Verlag,
Berlin, 1982.
23. Fohmann, L.: Wissenserwerb und maschinelles
Lernen.
Künstliche Intelligenz und
Expertensysteme, S. 125-200,
Oldenburg Verlag, München, 1985.
24. Mitchell, T.M.: Learning in Knowledge-Base Management
Systems.
On Knowledge-Base Management
Systems, S.403-406, Springer Verlag,
Berlin, 1986.
25. Marik, K.;
Thieme, S.: Metawissen - domainabhängig oder
domainunabhängig?
Interner Arbeitsbericht 15, Projekt-
gruppe Künstliche Intelligenz und
Textverstehen im BMFT-Verbundprojekt
LERNER, Institut für angewandte
Informatik, Berlin, 1986.
26. Carlisle, A;
Suwa, M.;
Shortliffe, E.H.: Completeness and Consistency in a
Rule-Based System.
Rule-Based Expert Systems,
S. 159-170, Addison-Wesley
Publishing Inc. Reading, MA, 1984.
27. Bramer, M.A.: Expert Systems: The Vision and the
Reality.
Research and Development in Expert
Systems, Proceedings 4th Technical
Conference of the British Computer
Society, S. 1-11, December 1984.
28. Doyle, J: A Truth Maintenance System.
Artificial Intelligence 12,
S. 231-272, 1977.
29. Stoyan, H.;
Görz, G.: LISP - Eine Einführung in die
Programmierung.
Springer Verlag, Berlin, 1984.

30. Puppe, F.: Erfahrungen aus drei Anwendungen mit MEDL.
Informatik Fachberichte 112, S. 234-245, Springer Verlag, Berlin, 1985.
31. Steele, G.L.: COMMON LISP - The Language.
Digital Press, USA, 1984.
32. N.N. Knowledge Engineering Environment, KEE.
Produktbeschreibung, Intellicorp, Menlo Park, 1985.
33. N.N. S1, Product Description.
FRAMENTEC S.A.M., Monte-Carlo Sun, Monaco, 1984.
34. Mescheder, B.: Funktionen und Arbeitsweise der Expertensystem-Shell TWAICE.
Künstliche Intelligenz und Expertensysteme, S. 57-90, Oldenburg Verlag, München, 1985.
35. Forgy, C.L.: OPS 5 User Manual.
Department of Computer Science, Carnegie-Mellon University, Pittsburgh, 1981.
36. Stefik, M.;
Bell, A.G.;
Bobrow, D.G.: Rule Oriented Programming in LOOPS.
Technical Report XEROX Corporation, XEROX PARC, 1983, KB-VLSF-82-22.
37. Hayes-Roth, F.;
Waterman, D.;
Lenat, D.B.: Building Expert Systems.
Addison-Wesley Publishing Company, Reading MA, 1983.
38. Robinson, L.G.: The Artificial Intelligence Report.
August 1985 Vol 2, No. 8,
Artificial Intelligence Publications, Palo Alto, 1985.
39. Di Primio, F.;
Bungers, D.;
Christaller, T.: BABYLON als Werkzeug zum Aufbau von Expertensystemen.
Informatik Fachberichte 112,
S. 70-79, Springer Verlag, Berlin, 1985.

40. McDermott, J.: R1: A Rule-Based Configurer of Computer Systems. Artificial Intelligence 19, S. 39-87, 1982.
41. Hörnig, K.M.: Methoden der Künstlichen Intelligenz in der Produktionsplanung und -steuerung. Produktionsplanung, Produktionssteuerung in der CIM-Realisierung, S. 425-432, Springer Verlag, Berlin, 1986.
42. Wiendahl, H.-P.; Lüssenkop, T.: Ein neuartiges Produktionsprozeßmodell auf der Basis eines Expertensystems für die Fertigungssteuerung. Produktionsplanung, Produktionssteuerung in der CIM-Realisierung, S. 433-454, Springer Verlag, Berlin, 1986.
43. Fox, M.S.; McDermott, J.: The Role of Databases in Knowledge-Based Systems. On Knowledge Base Management Systems, S. 407-449, Springer Verlag, Berlin, 1986.
44. Tönshoff, H.K.; Horns, A.: Petrie-Netze als spezielle Inferenzsysteme und ihre Anwendung in der Werkstattsteuerung und -simulation, Produktionsplanung, Produktionssteuerung in der CIM-Realisierung, S. 455-480, Springer Verlag, Berlin, 1986.
45. Versonder, G.T.: ACE: An Expert System for Telephone Cable Maintenance. Proc. 8th International Conference on Artificial Intelligence, Karlsruhe, 1983, pp 116-121.
46. Savory, S.E.: Fault Finder: A NIXDORF Expert System for Fault-Finding and Repair Planning. Artificial Intelligence, S. 119-128, Elsevier Science Publishers B.V. (North Holland), 1985.

47. Henne, P.;
Klar, W.;
Wittur, K.-H.:
DEX.C3 - Ein Expertensystem zur Fehlerdiagnose in automatischen Getrieben.
Informatik Fachberichte 112,
S. 105-120, Springer Verlag,
Berlin, 1985.
48. Winkelmann, K.:
Automated Reasoning in the Domain of Robotics.
Interner Bericht, Siemens Research and Technology Laboratories, 1986.
49. Reimann, J.A.:
STEP 1: An Expert System for Diagnosis of Technical Systems.
Interner Bericht, Siemens AG
Erlangen, 1986.
50. N.N.
Expert Systems for Metal Forming.
Engineering and Manufacturing Technology Cooperation, Center for Metal Processing, Battelle Columbus Laboratories, Columbus, Ohio, 1984.
51. Tengvald, E.:
The Design of Expert Planning Systems: An Experimental Operation Planning System for Turning.
Institute of Technology, Linköping, Sweden, 1983.
52. Descotte, Y.;
Latombe, J.C.:
GARI: An Expert System for Prozessplanung.
Solid Modelling by Computers: From Theory to Applications, S. 329-347,
Plenum Press, New York, 1984.
53. Grasmück, R.;
Guldner, A.:
Wissensbasierte Fertigungsplanung in Stanzereien mit FERPLAN.
SFB 314, Künstliche Intelligenz - Wissensbasierte Systeme, KI-Labor am Lehrstuhl für Informatik IV, Universität Saarbrücken, Memo Nr.2, 1985.
54. Fuchs, H.:
Automatische Arbeitsplanerstellung - Ein Baustein im Rahmen der integrierten Fertigungsunterlagenerstellung.
Dissertation, RWTH Aachen, 1981.

55. Runciman, C.;
Swift, K.: Expert System Guides CAD for Automatic Assembly.
Assembly Automation, August 1985,
S. 147-150.
56. Rudershausen, E.: ICAD: Ein wissensbasiertes System für CAD.
Künstliche Intelligenz, Heft 1, 1986,
CAD/CAM, Verlag München.
57. N.N.: Wie die CAD/CAM-Systeme der Zukunft aussehen.
CAD-CAM, Zeitschrift für Computer-Anwendungen, 5. Jahrgang, Nr.1,
1986, S. 34-37.
58. Maier, H.: Verwaltung von Methoden und ihre
Kopplung an ein Werkstückmodell in
einem integrierten CAD-System.
VDI-Fortschrittsberichte, Reihe 10,
Nr.13, VDI-Verlag, Düsseldorf, 1983.
59. Borning, A.: Thinglab - An Objekt Oriented System
for Building Simulations Using
Constraints.
Proceedings IJCAI 77, Cambridge MA,
1982.
60. N.N. SIMKIT: Technische Produktbeschreibung.
IntelliCorp, München, 1985.
61. Fox, M.S.: Intelligent Management System IMS.
Project Sampler, Intelligent Systems
Laboratory, The Robotics Institute,
Carnegie-Melon University, Pittsburgh,
PA, 1985.
62. Reddy, Y.V.;
Fox, M.S.: KBS: An Artificial Intelligence
Approach to Flexible Simulation.
Carnegie-Melon University, Pittsburgh,
Pa, 1982, CMU-RI-TR-82-1.
63. Schachter-
Radig, M.-J.: Wissenserwerb und -formalisierung
für den kommerziellen Einsatz
Wissensbasierter Systeme.
Informatik Fachberichte 112,
S. 314-332, Springer Verlag,
Berlin, 1985.

64. N.N. DIN 66234 Bildschirmarbeitsplätze.
Dialoggestaltung Teil 8, Entwurf vom
15.07.83, Beuth Verlag, Berlin.
65. Balzert, H.: Mensch-Computer-Kommunikation.
Vorlesungsunterlagen an der Uni-
versität Erlangen-Nürnberg, 1984.
66. Bullinger, H.-J.: Grundsätze der Dialoggestaltung.
Tagungsunterlagen zum Seminar
Software-Ergonomie, Technische
Akademie Esslingen, Juni 1984.
67. Reisch, D.: Die Berücksichtigung der Zuverlässig-
keit und Verfügbarkeitsanforderungen
bei der Planung von Maschinensystemen.
Dissertation, Universität Hannover,
1978.
68. N.N. EXAPT-Computer Aided Engineering.
EXAPT NC Systemtechnik GmbH, Aachen,
1983.
69. Lotter, B.: Arbeitsbuch der Montagetechnik.
Vereinigte Fachverlage Krausskopf,
Ingenieur Digest, Mainz, 1982.
70. Schimke, E.-F.: Planung von Handhabungssystemen.
Dissertation, RWTH Aachen, 1976.
71. Metzger, H.: Planung und Bewertung von Arbeits-
systemen in der Montage.
IPA-Schriftreihe, Krausskopf Verlag,
München, 1977.
72. Eversheim, W.: Montage richtig planen.
VDI-Fortschrittsberichte, Nr. 19,
VDI-Verlag, Düsseldorf, 1981.
73. N.N. Handhabungssysteme.
BMFT-Schriftenreihe "Humanisierung
des Arbeitslebens", Band 11,
VDI-Verlag, Düsseldorf, 1981.
74. Peffekoven, K.H.: Planung und Steuerung des Montageab-
laufs in Unternehmen der Einzel- und
Serienfertigung.
Dissertation, TH Aachen 1982.

75. Walther, J.: Systematische Planung flexibler automatisierter Montageanlagen. VDI-Z, Band 127 (1985) Nr.9, S.313 - 318.
76. Ammer, E.-D.: Rechnergestützte Planung von Montageablaufstrukturen für Erzeugnisse der Serienfertigung. IPA-Bericht Band 81, Springer Verlag, Berlin, 1985.
77. Lutz.: Abtakten von Montagelinien. Krausskopf Verlag, Mainz, 1973.
78. Mayer, S.: LAPEX- Ein rechnerunterstütztes Verfahren zur Betriebsmittelzuordnung. IPA-Bericht Nr.70, Springer Verlag, Berlin, 1983.
79. Haller, E.: Rechnergestützte Gestaltung ortsgebundener Montagearbeitsplätze dargestellt am Beispiel kleinvolumiger Produkte. IPA-Bericht Nr.65, Springer Verlag, Berlin, 1982.
80. Schmidt-Streier, U.: Eine Methode zur rechnergestützten Einsatzplanung von programmierbaren Handhabungsgeräten. Dissertation, TU Stuttgart, 1981.
81. Hirschbach, O.: Rechnergestützte Montageplanerstellung. IPA-Schriftenreihe Nr. 6, Krausskopf Verlag, 1978.
82. Bullinger, H.-J.; Schweizer, W.: Interaktive Simulation von Montagesystemen. Wt-Zeitschrift für industrielle Fertigung, Heft 7, 1985, S.421-424.
83. Vettin, G.: Simulation von Ablaufsystemen in der verketteten Fertigung, Maschinenmarkt, Würzburg, Heft 9, S. 147-150, 1980.

84. Pritschow, G.;
Storr, A.: SIFLA, Simulation flexibler Anlagen.
Pflichtenheft, Institut für
Steuerungstechnik, Universität
Stuttgart, 1985.
85. Viehweger, B.;
Wenke, B.: Rechnerunterstützte Planungshilfen
für Fertigungssysteme, ZWF 81 (1986)
Nr.1, S. 23 - 28.
86. Furgac, I.: Aufgabenbezogene Auslegung von
Robotersystemen.
Carl Hanser Verlag, München, 1985.
87. N.N.: DIN 8593, Fertigungsverfahren.
Beuth Verlag GmbH, Berlin, 1985.
88. N.N.: VDI 2860 Blatt.
VDI- Richtlinien "Montage und Hand-
habungstechnik" (Entwurf),
VDI-Gesellschaft Produktionstechnik
(ADB), Düsseldorf, 1982.
89. Warneke, H.-J.;
Schraft, R.D.: Handbuch Handhabungs-, Montage- und
Industrierobotertechnik.
Verlag Moderne Industrie, Landsberg,
1984.
90. N.N.: MTM-Handbuch.
Deutsche MTM-Vereinigung e.V.,
Hamburg, 3.Auflage, 1981.
91. N.N.: REFA - Methodenlehre der Planung und
Steuerung, Carl Hanser Verlag,
München.
92. Janisch, H.-W.: Optimierung der Puffer bei elastisch
verketteten Fertigungssystemen.
Dissertation, TU Hannover, 1979.
93. N.N.: The Romulus Solid Modelling System,
V6.0.
User's Reference Manual, Shape Data,
Cambridge, 1985.
94. Spur, G.;
Krause, F.-L.: CAD-Technik.
Carl Hanser Verlag, München, 1984.

95. Eigner, M.;
Maier, H.: Einstieg in CAD.
Carl Hanser Verlag, München, 1985.
96. Grabowski, H.: Die automatisierte Fabrik - Die deutsche Antwort auf die japanische Herausforderung ?
CAD/CAM Forum zur Hannover Messe,
Prime Computer GmbH, Wiesbaden, 1983.
97. Gausemeier, J.: Von CAD zu CIM - Gesichtspunkte für den Aufbau von ganzheitlich automatisierten Produktionsprozessen.
Zeitschrift für wirtschaftliche Fertigung Zwf81, Nr.9 1986,
S. 467-472.
98. Hellwig, H.-E.;
Paulus, M.: Die Kopplung von CAD und CAM, Teil 1 und Teil 2.
VDI-Z 125 Nr.10, 1983, S. 355-360,
VDI-Z 125 Nr.11, 1983, S. 455-460.
99. Encarnacao, J.;
Schuster, R.;
Vöge, E.: Product Data Interfaces in CAD/CAM-Applications, Springer Verlag,
Berlin, 1986.
100. Heger, D.: MAP - Leitlinien für die Kommunikation in der Automatisierungstechnik.
Produktionsplanung, Produktionssteuerung in der CIM-Realisierung,
S. 171-202, Springer Verlag,
Berlin, 1986.
101. Krallmann, H.: Expertensysteme für die computerintegrierte Fertigung.
Produktionsplanung, Produktionssteuerung in der CIM-Realisierung,
S. 395-424, Springer Verlag,
Berlin, 1986.
102. Warneke, H.-J.: Synchronisation: Notwendige Voraussetzungen für die CIM - Realisierung.
Produktionsplanung, Produktionssteuerung in der CIM - Realisierung,
S. 21-40, Springer Verlag,
Berlin, 1986.
103. Gairola, A.: Montagegerechtes Konstruieren - Ein Beitrag zur Konstruktionsmethodik.
Dissertation, TH Darmstadt, 1981.

104. Gairola, A.: Design Analysis for Automatic Assembly. Toward the Factory of the Future, S. 441-447, Springer Verlag, Berlin, 1985.
105. Wedekind, H.: Eine konzeptionelle Basis für den Einsatz von Datenbanksystemen in Flexiblen Fertigungssystemen. Informatik-Forschung und Entwicklung, Heft 1/87, Springer Verlag, Berlin.
106. Scheer, A.-W.: Anforderungen an Datenverwaltungssysteme im CIM-Konzept. Produktionsplanung, Produktionssteuerung in der CIM-Realisierung, S. 121-142, Springer Verlag, Berlin, 1986.
107. Brodie, M. L.; Mylopoulos, J.: Knowledge Bases vs Databases. On Knowledge Base Management Systems, S. 83-94, Springer Verlag, Berlin, 1986.
108. Seliger, G.: Wirtschaftliche Planung automatisierter Fertigungssysteme. Carl Hanser Verlag, Berlin, 1983.
109. Eversheim, W.; Fromm, W.: Planung und Simulation flexibler, automatisierter Fertigungssysteme. Zeitschrift für wirtschaftliche Fertigung ZwF81, Nr.10 1986, S. 541-547.
110. Michie, D.: Expert System and Robotics. Handbook of Industrial Robotics, John Wiley & Sons, New York, 1985.
111. Kak A.C.; Boyer K.L.; Chen, C.H.: A Knowledge - Based Robotic Assembly Cell. IEEE EXPERT, IEEE Computer Society, New York, Spring 1986

Lebenslauf

- 1955 Andreas Hemberger, geb. am 22. Februar 1955 in
Mannheim
Eltern: Andreas Hemberger
 Edith Hemberger geb. Zieher
- 1962 - 1966 Besuch der Grundschule in Neckarhausen
- 1966 - 1974 Besuch des Carl-Benz-Gymnasiums in Ladenburg
- 1974 - 1976 Wehrdienst
- 1976 - 1982 Studium der Fachrichtung Maschinenbau an der
Technischen Hochschule Karlsruhe
- 1979 - 1980 Hilfsassistent am Institut für Maschinenkon-
struktionslehre, Abteilung Kraftfahrzeugbau
- 1982 Diplom-Hauptprüfung im Fachbereich Maschinenbau
an der TH Karlsruhe am 29. Oktober 1982
- Ab 1983 Wissenschaftlicher Mitarbeiter am Lehrstuhl für
Fertigungsautomatisierung und Produktions-
systematik des Instituts für Fertigungstechnik
der Universität Erlangen-Nürnberg
Lehrstuhlinhaber: Prof. Dr.-Ing. K. Feldmann
- 1983 -1987 Akademischer Rat am gleichen Lehrstuhl
- Ab 1986 Oberingenieur für Rechnergestützte Konstruktion
und Planung am gleichen Lehrstuhl
- 1987 Promotionsprüfung am 10. April 1987 an der Tech-
nischen Fakultät der Universität Erlangen-
Nürnberg