

Wolfgang Scholz

*Modell zur datenbankgestützten Planung
automatisierter Montageanlagen*

Wolfgang Scholz

*Modell zur datenbankgestützten Planung
automatisierter Montageanlagen*

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
Friedrich-Alexander-Universität Erlangen-Nürnberg

Tag der Einreichung:	31. Januar 1989
Tag der Promotion:	11. Mai 1989
Dekan:	Prof. Dr.rer.nat. F. Hofmann
Berichterstatter:	Prof. Dr.-Ing. K. Feldmann
	Prof. Dr.-Ing. H. Meerkamm

CIP-Titelaufnahme der Deutschen Bibliothek

Scholz, Wolfgang:

Modell zur datenbankgestützten Planung automatisierter
Montageanlagen / Wolfgang Scholz. Hrsg. von Klaus
Feldmann. - München ; Wien : Hanser, 1989

(Fertigungstechnik - Erlangen)

Zugl.: Erlangen, Nürnberg, Univ., Diss., 1989

ISBN 3-446-15825-1

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, Mikrofilme oder ein
anderes Verfahren), auch nicht für Zwecke der Unterrichts-
gestaltung - mit Ausnahme der in den §§ 53, 54 URG ausdrücklich
genannten Sonderfälle -, reproduziert oder unter Verwendung
elektronischer Systeme verarbeitet, vervielfältigt oder
verbreitet werden.

© Carl Hanser Verlag München, Wien 1989

Herstellung: Copy Center 2000, Erlangen-Eltersdorf

Printed in Germany

Vorwort

Die vorliegende Dissertation entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter 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 wohlwollende Förderung bei der Durchführung dieser Arbeit.

Herrn Professor Dr.-Ing. H. Meerkamm, Leiter der Abteilung für Maschinenelemente und Fertigungsgerechtes Konstruieren, danke ich für die Übernahme des Koreferates, seine fachlichen Anregungen sowie die sorgfältige Durchsicht meiner Arbeit.

Ferner gilt mein Dank allen Kollegen, Studenten und wissenschaftlichen Hilfskräften, die durch ihre Unterstützung zum Gelingen der Arbeit beigetragen haben. Dies gilt insbesondere für Herrn Dipl.-Inf. Walter Streit und Herrn Cand.-Inf. Jens Rohde.

Besonderen Dank möchte ich meinem ehemaligen Kollegen Herrn Dr.-Ing. D. Classe für die moralische Unterstützung und die fruchtbaren Diskussionen aussprechen.

Wolfgang Scholz

**MODELL ZUR DATENBANKGESTÜTZTEN PLANUNG
AUTOMATISIERTER MONTAGEANLAGEN**

-INHALTSVERZEICHNIS-

1.	<u>Einleitung</u>	1
2.	<u>Rechnergestützte Montagesystemplanung</u>	4
2.1	Komponenten der rechnergestützten Montagesystemplanung	4
2.2	Datenbanken in der Montageplanung	6
2.3	Automatisierte Montagesysteme	8
2.4	Verfahrensketten in der Montage	9
3.	<u>Darstellung der Datenbanktechnologie</u>	12
3.1	Allgemeines zu Datenbanksystemen	12
3.1.1	Charakteristika einer Datenbank	13
3.1.2	Die drei Schemaebenen	15
3.1.3	Architektur eines Datenbanksystems	17
3.2	Grundlagen der Datenmodellierung	20
3.2.1	Abstraktionsebenen der Informations- abbildung	20
3.2.2	Die klassischen Datenmodelle	22
3.2.3	Semantische Datenmodelle	29
3.2.4	Abstraktionsmechanismen	31
3.2.5	Integritätsbedingungen	32
3.3	Der Datenbankentwurf	36
3.3.1	Komponenten des Datenbank-Entwurfs- prozesses	37
3.3.2	Bedarfsanalyse und semantische Modellierung	40
3.3.3	Die Normalisierung	43
3.3.4	Abbildung semantischer Schemata auf klassische Datenmodelle	44

3.4	Datenbanksysteme für technische Anwendungen	49
3.4.1	Anforderungen an Datenbanksysteme für technische Anwendungen	49
3.4.2	Integrationsstufen der Datenhaltung	53
4.	<u>Systemtechnische Betrachtung automatisierter Montageanlagen</u>	58
4.1	Allgemeines zur Systemtechnik	58
4.1.1	Begriffe der Systemtechnik	58
4.1.2	Klassifikationssystematik für Systeme	62
4.1.3	Allgemeines Modell eines Transformationssystems	65
4.2	Die Montage als Transformationssystem	68
4.2.1	Schichten-Architektur-Modell für die Produktion	70
4.2.2	Bauteile als Operanden des Montagesystems	73
4.2.3	Der Montageprozeß	77
4.2.4	Die Operatoren des Transformationssystems Montage	82
5.	<u>Systematische Planung automatisierter Montageanlagen</u>	92
5.1	Allgemeines zur Montageplanung	92
5.1.1	Einordnung der Montageplanung in die Funktionsbereiche eines Produktionsbetriebes	93
5.1.2	Ablauf der systematischen Montageanlagenplanung	96
5.2	Analyse der Montageaufgabe	100
5.2.1	Das Produktmodell als Basis der Montageplanung	101
5.2.2	Analyse des Produktaufbaus	103
5.3	Erstellung eines Montagekonzepts	106
5.3.1	Strukturelle Beschreibung des Montageablaufs	108
5.3.2	Operationale Beschreibung der Montageaufgabe	112
5.3.3	Auswahl von Wirkorganen	115

5.4	Die Montagelayouterstellung	117
5.4.1	Abtaktung der Montagestationen	118
5.4.2	Geräteauswahl und Zusammenstellung zum Montagelayout	120
5.5	Optimierung des Layouts durch Simulation	121
5.5.1	Bewegungssimulation	122
5.5.2	Prozeßsimulation	124
5.6	Wirtschaftlichkeitsrechnung	127
6.	<u>Entwurf eines konzeptionellen Schemas für die Planung automatisierter Montageanlagen</u>	131
6.1.	Klassifizierung der Montagedaten	132
6.2	Konzeption eines allgemeinen Strukturmodells unter Berücksichtigung von Varianten und Versionen	135
6.2.1	Die Entstehung von Alternativen, Varianten und Versionen	136
6.2.2	Entwicklung eines semantischen Schemas zur Verwaltung von Varianten und Versionen	138
6.3	Rekonstruktion der Begriffe für die einzelnen Schritte der Montageplanung	151
6.3.1	Rekonstruktion der Begriffe der Montageaufgabenanalyse	151
6.3.2	Rekonstruktion der Begriffe zur Beschreibung des Montageablaufs	153
6.2.3	Rekonstruktion der Begriffe der Geräteauswahl und Layouterstellung	155
6.3.4	Rekonstruktion der Begriffe der Bewegungssimulation	159
6.3.5	Rekonstruktion der Begriffe der Prozeßsimulation	161
6.3.6	Rekonstruktion der Begriffe der Wirtschaftlichkeitsrechnung	162
6.4	Integration der externen Schemata in ein konzeptionelles Schema	163

7.	<u>Exemplarische Implementierungen</u>	168
7.1	Implementierungsumgebung	168
7.2	Implementierung des Datenbankschemas	170
7.3	Implementierung von Anwenderprogrammen	175
8.	<u>Zusammenfassung</u>	183
9.	<u>Literaturverzeichnis</u>	185

1. Einleitung

Die Montage verursacht bis zu 70 % der Herstellkosten für feinerwerktechnische Produkte. Dies ist überwiegend auf die in der Montage vorherrschenden manuellen Tätigkeiten zurückzuführen [1]. Die Montage bietet im Vergleich zur Vorfertigung noch erhebliche Rationalisierungsreserven [2]. Die gegenwärtigen Automatisierungsbestrebungen in der Produktion richten sich deshalb vor allem auf den Bereich der Montage, um den Anteil der lohn- und damit kostenintensiven Handmontage zu verringern. Zusätzlich sollen durch eine rechnerintegrierte Montage flexible Reaktionen auf sich ändernde Marktanforderungen ermöglicht, sowie eine Steigerung der Qualität und die Verkürzung von Durchlaufzeiten erreicht werden.

Bislang gibt es jedoch nur wenige komplexe und flexibel automatisierte Montageanlagen. Begründet wird die weiterhin bestehende Rationalisierungslücke in der Montage vor allem mit dem bislang nur geringen, systematischen Wissen über automatisierte Montageprozesse, die weitgehend nichtmontagegerechte Konstruktion und den Mangel an effizienten Planungswerkzeugen [3].

Die vorliegende Arbeit soll einen Beitrag dazu leisten, das Wissen über die Montage zu systematisieren und die Planungssystematik für automatisierte Montagesysteme zu vervollständigen. Weiterhin soll sie zeigen, wie durch den Einsatz einer Datenbank verschiedene rechnergestützte Planungswerkzeuge für die Montage zu einem Gesamtsystem integriert werden können. Der Aspekt der Bereitstellung relevanter Montageinformationen für die Produktkonstruktion wird im Rahmen dieser Arbeit nicht näher betrachtet.

Zunächst wird im Kapitel 2 kurz erläutert, welche Problemkreise die Entwicklung eines durchgängigen Gesamtsystems für die Montageplanung beinhaltet. Es wird gezeigt, daß eine wesentliche Voraussetzung für ein effektives System zur Montageplanung eine integrierte Datenbank ist.

Allgemein lassen sich durch den Einsatz von Datenbanken unter-

schiedliche Anwendungsprogramme integrieren. Durch die gemeinsame Nutzung eines Datenbestandes wird der Eingabeaufwand verringert, und es werden Inkonsistenzen vermieden.

Trotz dieser Vorteile werden Datenbanken für technische Anwendungen heute noch kaum eingesetzt. Der Grund liegt darin, daß derzeitige verfügbare herkömmliche Datenbanksysteme Mängel beim Einsatz für technische Problemstellungen aufweisen, und daß der Datenbankentwurf sich als erheblich komplizierter als für kommerzielle Anwendungen darstellt, da im technischen Bereich wesentlich komplexere Strukturen vorherrschen.

Mit der vorliegenden Arbeit soll auch gezeigt werden, wie sich heute verfügbare Datenbankmanagementsysteme für die Montageplanung nutzen lassen. Die Bestrebungen richten sich dabei nicht auf die Entwicklung neuer Datenbankkonzepte, wie sie derzeit vielfach im Gange sind (siehe Kapitel 3.4). Der Schwerpunkt liegt auf der Datenmodellierung. Die Bedarfsanalyse als Grundlage für die Datenmodellierung nimmt einen breiten Raum ein. Für exemplarische Implementierungen wurde ein gegenwärtig verfügbares Datenbankmanagementsystem eingesetzt und damit bewußt Einschränkungen in bezug auf die Möglichkeiten der Definition von Integritätsbedingungen und das Laufzeitverhalten in Kauf genommen.

Kapitel 3 enthält eine kurze Einführung in die Begriffswelt der Datenbanktechnologie. Es wird die Architektur einer Datenbank vorgestellt, und die Vorgehensweise beim Datenbankentwurf sowie die verschiedenen dabei einsetzbaren Methoden und Hilfsmittel werden aufgezeigt.

Im Kapitel 4 wird eine systemtechnische Betrachtung von Montageanlagen vorgestellt. Die Systemtechnik liefert als interdisziplinäre Wissenschaft Methoden, Verfahren und Hilfsmittel zur Analyse und Beschreibung komplexer Systeme. Sie eignet sich daher besonders für die Begriffsfindung als Voraussetzung zur semantischen Datenmodellierung. Die Analyse der Montage als System dient dazu, die Abläufe in der Montage und die daran

beteiligten Komponenten systematisch zu erfassen und darzustellen. Die Durchführung der Systemanalyse erfolgt anhand eines soziotechnologischen Modells der Montage.

Kapitel 5 zeigt den Ablauf der systematischen Planung automatischer Montageanlagen. Es werden zu den einzelnen Planungsschritten bereits bekannte Methoden und Werkzeuge vorgestellt und erläutert. Weiterhin wird dort ein neues Verfahren zur Beschreibung von Montageabläufen eingeführt.

Kapitel 4 und 5 bilden die Grundlage für den im Kapitel 6 beschriebenen semantischen Entwurf eines konzeptionellen Schemas zur Unterstützung der Montageplanung. Dabei werden zunächst zu den einzelnen in Kapitel 5 vorgestellten Planungsschritten sogenannte externe Schemata entworfen und diese anschließend zu einem konzeptionellen Schema integriert. Zur Darstellung der Schemata wird eine modifizierte Form des Entity-Relationship-Modells verwendet.

Zum Abschluß der Arbeit werden im Kapitel 7 exemplarische Implementierungen dieses konzeptionellen Datenbankschemas und einiger Anwenderprogramme zur Montageplanung, die mit einer nach diesem Schema aufgebauten Datenbank arbeiten, vorgestellt.

2. Rechnergestützte Montagesystemplanung

Der Einsatz rechnergeführter Montagesysteme ist zu einem Schwerpunkt fertigungstechnischer Entwicklung geworden [4]. Eine wirtschaftliche Produktion bedingt flexibel automatisierte Montageanlagen um kostenintensive Handmontage zu vermeiden. Flexibel automatisierte Montageanlagen erfordern wegen ihrer Komplexität höhere Investitionskosten und eine detailliertere Planung. Wettbewerbsgründe führen zu immer kürzeren Produktlebenszyklen und immer größerer Variantenvielfalt. Dadurch wird die zur Verfügung stehende Planungszeit zusätzlich verkürzt und die Komplexität erhöht. Es besteht daher ein Bedarf nach geeigneten, rechnergestützten Hilfsmitteln für die Planung automatisierter Montageanlagen.

Durch konsequentes Nutzen der heute von der Datenverarbeitung gebotenen Möglichkeiten läßt sich der Planungsaufwand und damit die Planungszeit auch für komplexe automatische Montagesysteme erheblich verringern und die Planungsqualität erhöhen.

2.1 Komponenten der rechnergestützten Montagesystemplanung

Der Rechnereinsatz in den verschiedenen Arbeitsstufen begünstigt zunehmend die optimale Planung und Gestaltung von Montagesystemen [4]. Auf der Grundlage der seit Mitte der siebziger Jahre entwickelten Planungssystematiken für die Montage sind eine Reihe von Rechnerprogrammen für die Durchführung einiger Teilschritte des Planungsablaufs für automatische Montagesysteme entstanden. Die heute verfügbaren CAD-Systemen ermöglichen bereits eine intensive Unterstützung bei der Layoutplanung und Gestaltung von Montagesystemen [4]. Was fehlt ist die Ergänzung sowie die Integration dieser rechnergestützten Planungswerkzeuge zu einem Gesamtsystem, mit dem die Montageplanung von der Aufgabenanalyse bis zur Zusammenstellung eines optimierten Layouts der Montageanlage durchgeführt werden kann.

Ausgangspunkt einer durchgängig rechnergestützten Montagesystemplanung ist das in einem CAD-System beschriebene zu montierende Produkt, das zusammen mit den Auftragsdaten die Montageaufgabe kennzeichnet. In den verschiedenen Stufen der Montagesystemplanung werden daraus die Montagefolge, die Fügetechnik und das Montagekonzept abgeleitet. Die systematische Montageplanung liefert als Ergebnis die Beschreibung eines automatisierten Montagesystems, mit dem das Produkt zusammengesetzt werden kann. Eine zentrale Stellung in einer derartigen, durchgängig rechnerunterstützten Montageplanung nimmt eine integrierte Datenbank ein (Bild 1).

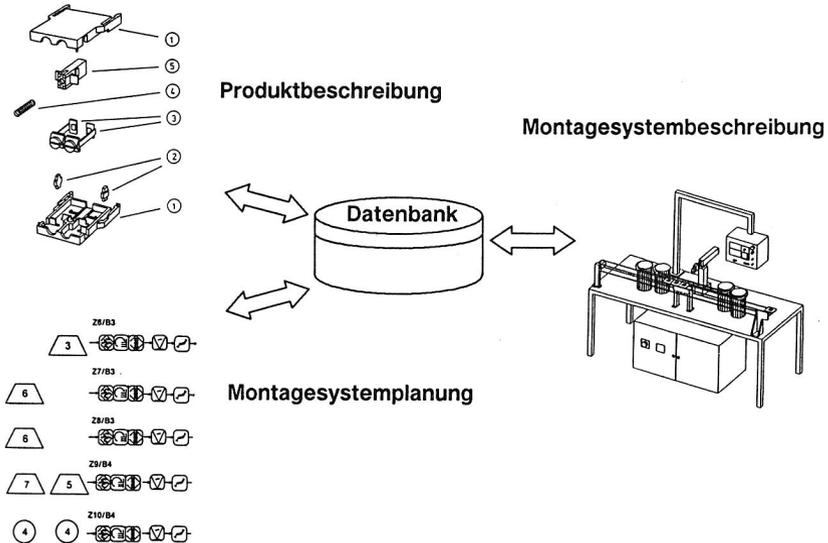


Bild 1: Rechnergestützte Montageplanung

Durch die Datenbank soll die zentrale Verwaltung aller für die Montage relevanten Daten gewährleistet werden. Dies beinhaltet sowohl die Daten über das zu montierende Produkt, als auch die im Laufe der rechnergestützten Montageplanung benötigten und erzeugten Informationen sowie die Beschreibung der geplanten automatisierten Montageanlage.

2.2 Datenbanken in der Montageplanung

Eine wesentliche Voraussetzung für eine effiziente Rechnerunterstützung der Montageplanung ist eine integrierte Datenbank. Sie bildet die Grundlage für die Verkettung einzelner, bei der Planung von Montagesystemen eingesetzter, rechnergestützter Planungswerkzeuge zu einem Gesamtsystem (Bild 2).

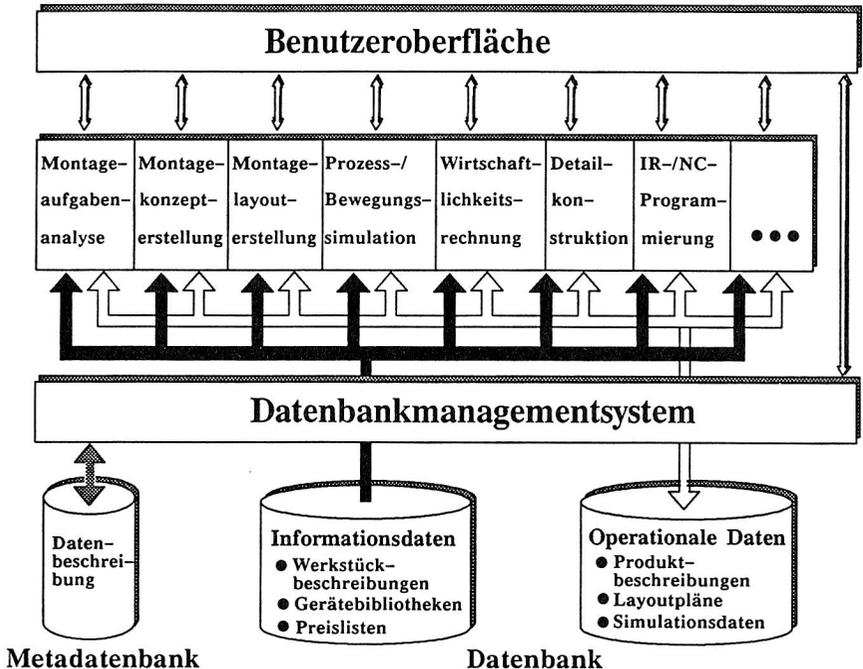


Bild 2: Architektur eines Systems zur Montageplanung auf der Basis einer integrierten Datenbank

In dieser Datenbank werden sämtliche Montage­daten, d.h. sowohl die operationalen Daten über das Montagesystem als Planungsgegenstand als auch für die Planung benötigte Informationsdaten,

verwaltet. Die operationalen Daten stellen die integrierende Komponente dar. Sie umfassen alle in den einzelnen Planungsschritten erzeugten Daten, die damit für nachfolgende Planungsschritte wieder zur Verfügung stehen. Durch den Einsatz einer Datenbank für diese Aufgaben werden viele Nachteile von datei-orientierte Systemen wie übermäßige Redundanz der Daten, fehlende Integritätskontrolle, mangelnde Datenstrukturierung usw. vermieden sowie Standardisierung und Austausch von komplexen Datenstrukturen verbessert. Es entfällt der Aufwand für die Dateneingabe von bereits in früheren Planungsprogrammen generierten Daten, wodurch zusätzlich Fehler vermieden werden.

Trotz dieser Vorteile kommen in der technischen Datenverarbeitung Datenbanken bisher kaum zum Einsatz. Die auf dem Markt verfügbaren kommerziellen Datenbanksysteme orientieren sich vorwiegend an den Bedürfnissen des Dienstleistungssektors und vernachlässigen technisch-wissenschaftliche Anforderungen [5]. Es besteht vor allem ein mangelnder Anwendungsbezug bei der Datenmodellierung. Obwohl erste rechnergestützte Hilfsmittel zum automatischen und systematischen Datenbankentwurf bereits existieren, scheinen diese eher für konventionelle Anwendungen geeignet. In den "nichtkommerziellen" Bereichen ist man hingegen weiterhin auf eigene Entwurfserfahrungen angewiesen [6].

Neben den Mängeln kommerzieller Datenbankmanagementsysteme beim Einsatz für technische Anwendungen, stellt also die Komplexität des Datenbankentwurfs das Haupthindernis beim Aufbau einer Datenbank für die Montageplanung dar. Ein Großteil des Kapitels 3 ist daher der Datenmodellierung gewidmet, um die Vorgehensweise von der Bedarfsanalyse bis zum fertigen Datenbankschema aufzuzeigen.

Von entscheidender Bedeutung für die Qualität einer Datenbank ist die Bedarfsanalyse als Basis des Datenbankentwurfs. Sie beinhaltet die Abgrenzung und Klassifikation von Begriffen und trägt zur Systematisierung und Standardisierung bei. Durch sie wird festgelegt, welche Daten in der Datenbank verwaltet werden, und die Möglichkeiten der späteren Nutzung werden

vorbestimmt. Die Bedarfsanalyse ist deshalb besonders gründlich durchzuführen.

Der Aufbau einer Datenbank für die Montageplanung beinhaltet eine zweifache Aufgabenstellung im Rahmen der Bedarfsanalyse. Zum einen ist zu analysieren, welche Informationsdaten benötigt werden, zum anderen, welche Daten (operationale Daten) in den einzelnen Arbeitsschritten der Montageplanung erzeugt werden (siehe auch Bild 2).

2.3 Automatisierte Montagesysteme

Die Betrachtung automatisierter Montagesysteme als Objekte der Montagesystemplanung bildet die Grundlage für den Aufbau einer Informationsdatenbank für die Montage.

Die moderne Montagetechnik strebt einerseits danach, die im Bereich der Serienfertigung bewährten Automatisierungskonzepte durch Integration von NC-Achsen und Steuerungskomponenten zu flexibilisieren. Andererseits werden durch neue Montagekonzepte weitere Automatisierungsbereiche erschlossen.

Auch in zukünftigen Montagestrukturen werden jedoch klassische Montagelinien, Roboterlösungen und Handarbeitsplätze zusammenwirken. Bei den Montageaufgaben ist, durch die vermehrte Produktion elektronischer Geräte, eine deutliche Verschiebung hin zu den besonderen Aufgaben der Flachbaugruppenbestückung erkennbar [2].

Allgemein ist der Trend erkennbar, Montagezellen als standardisierte Grundelemente mit definierten Schnittstellen für den effizienten Aufbau größerer Montagelinien zu schaffen. Dies gilt sowohl für den mechanischen als auch für den elektronischen Bereich (Bild 3) [2].

Für den Aufbau der Montagezellen wird wiederum versucht, möglichst standardisierte Komponenten einzusetzen. Dies verringert

die Kosten und erleichtert die rechnergestützte Konstruktion und Planung von Montagesystemen.

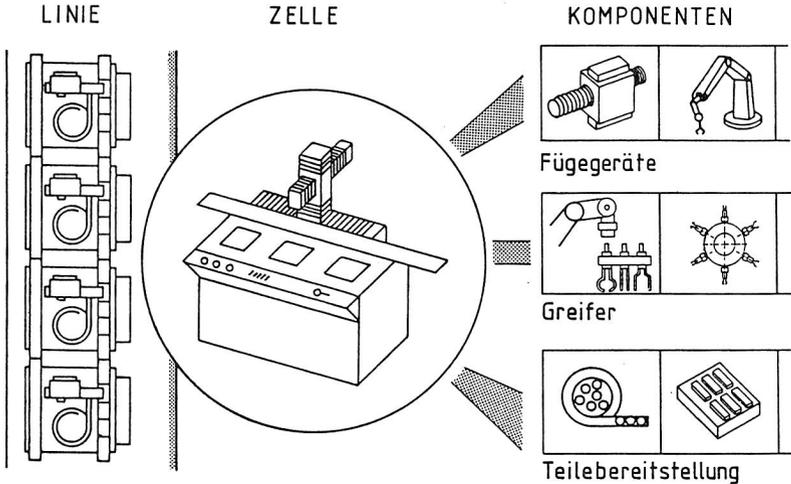


Bild 3: Struktur und Komponenten von Montagelinien [2]

Zur weiteren Standardisierung von Montagekomponenten kann eine systemtechnische Betrachtung der Montage, wie sie in Kapitel 4 vorgestellt wird, einen wesentlichen Beitrag leisten. Sie ermöglicht eine Klassifizierung der an der Montage beteiligten Komponenten und schafft damit auch die Voraussetzung für die effiziente Verwaltung von Informationsdaten für die Montageplanung in Datenbanken.

2.4 Verfahrensketten in der Montage

Die Verfahrenskette beinhaltet die durchgehende Fertigungsdatenverarbeitung, von der Produktbeschreibung bis zur Prozeßführung [2]. Sie umfaßt neben der Planung und Konstruktion von Montageanlagen auch die Offline-Programmierung und direkte Datenkopplung zwischen Planungsbereichen und Werkstattebene.

Die möglichen Verfahrensketten für die mechanische und die elektronische Montage zeigen Ähnlichkeiten, aber auch strukturelle Unterschiede im Ablauf (Bild 4). Bedingt durch den planaren Charakter der reinen Bestückungsaufgabe, sind CAD/CAM-Linien in der Elektronik bereits mehr verbreitet. Die Bestückungsprogramme lassen sich sicher einfacher aus dem Leiterplattenlayout ableiten als bei mechanischer Montage mit sehr unterschiedlichen Fügeteilen und zumeist komplexen Montageabläufen [2].

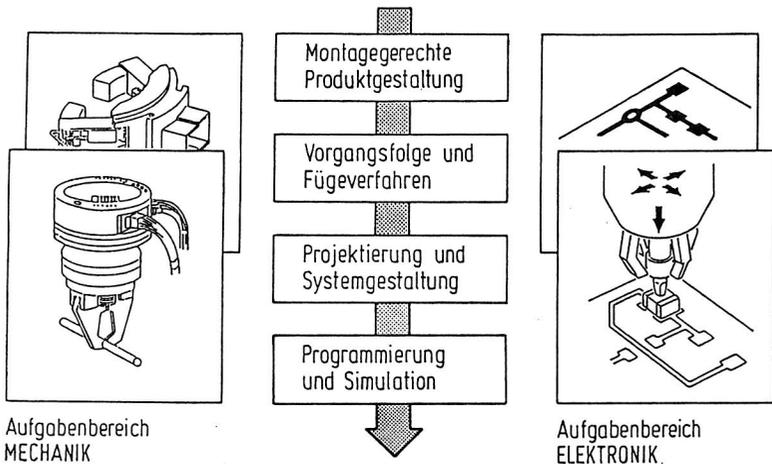


Bild 4: Verfahrensketten in der Montage [2]

Für die Elektronikproduktion werden in der Regel standardisierte Montagezellen eingesetzt. Diese sind flexibel für die Bestückung sehr unterschiedlicher Leiterplatten, es muß nur jeweils ein neues Bestückprogramm erstellt werden.

Im Gegensatz dazu sind Programmänderungen für eine vorhandene Montageanlage in der mechanischen Montage sehr selten und erfordern zumeist auch konstruktive Veränderungen [2].

In der mechanischen Montage wird also die Flexibilität des Betriebes einer automatisierten Montageanlage viel stärker durch die Planung vorbestimmt als in der elektronischen Montage.

Die rechnergestützte Montageanlagenplanung stellt somit einen wesentlichen Teil der Verfahrenskette dar, insbesondere in der mechanischen Montage. Es ist daher ein besonderes Bestreben, ein durchgängiges rechnergestütztes Konzept für die Planung automatisierter Montageanlagen zu entwickeln.

Eine Betrachtung verfügbarer Planungswerkzeuge für die Montage zeigt, daß vor allem in den Bereichen der Montageaufgabenanalyse, der Festlegung der Montagereihenfolge und der Beschreibung des Montageablaufs noch kaum geeignete Rechnerunterstützung zu finden ist. Im Kapitel 5 wird daher ein neues Verfahren zur rechnergestützten Montageablaufbeschreibung vorgestellt.

Dieses Kapitel enthält auch eine Gesamtbetrachtung der einzelnen Schritte der systematischen Montageplanung im Sinne einer Bedarfsanalyse für die Datenmodellierung. Zum Aufbau einer Datenbank für die Montageplanung ist eine derartige Analyse des systematischen Planungsablaufs unerlässlich.

3. Darstellung der Datenbanktechnologie

Für technische Rechneranwendungen werden auch heute noch meist Dateisysteme zur permanenten Datenspeicherung eingesetzt. Jeder Benutzer erzeugt und liest seine privaten Dateien. Die Dateien werden nur von wenigen Programmen verwendet bzw. geändert und sind im Normalfall nur im Kontext dieser Programme interpretierbar (Bild 5). Die primäre logische Struktur der Daten und die Weise, in der der Programmierer sie sieht, sind im wesentlichen identisch mit ihrer physischen Speicherungsstruktur. Diese physische Datenabhängigkeit führt dazu, daß der direkte Benutzer einer Datei zur Formulierung seiner Anwendung die Dateistruktur genau kennen muß. Da jeder Benutzer normalerweise seine eigenen Dateien für seine speziellen Zwecke anlegt, kommt es häufig zu Mehrfachabspeicherung der gleichen Datenbestände.

Bei Änderungen tritt die Gefahr von Unstimmigkeiten zwischen den mehrfach abgespeicherten Daten auf, und es können Inkonsistenzen entstehen, da jeder Benutzer eigenverantwortlich seine Dateien verwaltet. Die Integration der Daten, d.h. die Zugänglichkeit sowohl für verschiedene Anwendungen als auch für verschiedene Benutzer, ist gering. Diese Probleme wurden schon frühzeitig beim EDV-Einsatz im kommerziellen Bereich erkannt und haben zur Entwicklung von organisierten Datensammlungen, genannt Datenbanken, geführt.

3.1 Allgemeines zu Datenbanksystemen

Mit dem Einsatz eines Datenbanksystems wird die Organisation der Daten zentralisiert (Bild 5). Zwei Hauptgründe sprechen dafür die Daten zentralisiert anzuordnen [7]:

- Eine integrierte Datenverarbeitung über mehrere Fachabteilungen und Einzelanwendungen hinweg ist mit lokalen Daten, die häufig in mehreren Kopien gehalten werden, nicht möglich. Eine Datenbank bildet die Grundlage für ein umfassendes

Informationssystem.

- In einem Datenbanksystem werden Integritätskontrollen (Konsistenzkontrollen) global vorgenommen, es entfallen die in konventionellen Programmen einen breiten Raum einnehmenden Kontrollroutinen, um Datenfehler jedweder Art zu entdecken.

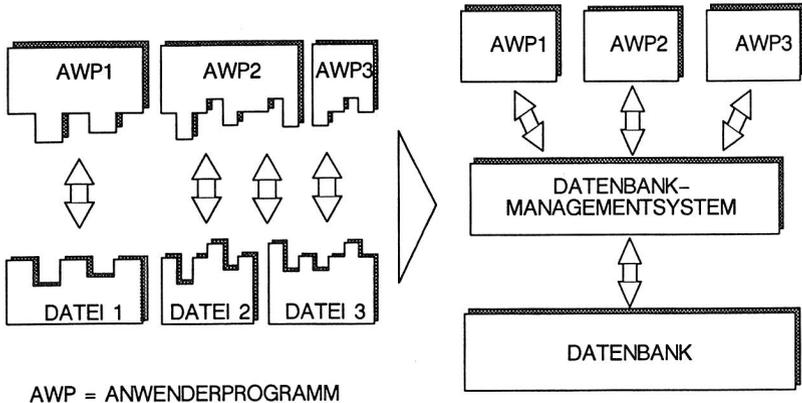


Bild 5: Zentralisation der Daten in einer Datenbank

3.1.1 Charakteristika einer Datenbank

Die sich aus der Integration und zentralen Verwaltung der Daten ergebenden Charakteristika eines Datenbanksystem lassen sich wie folgt zusammenfassen (vgl. [8,9,10,11]):

1. Strukturierung der Daten

Das Datenbanksystem ermöglicht die Bearbeitung strukturierter Daten. Die Datenorganisation ist intern so gestaltet, daß sich der Benutzer mit einer Operation auf bestimmte Daten und Datengruppen beziehen kann, ohne daß ungeordnete Mehrfachspeicherung auftritt (kontrollierte Redundanz).

2. Datenunabhängigkeit und Flexibilität

Daten und Anwendungen sind vollständig getrennt (datenunabhängig). Reorganisationen innerhalb des Datenbanksystems tangieren die Anwenderprogramme nicht, und umgekehrt. Die Datenbank ist nicht nur für die momentan vorhandenen Anwendungen einsetzbar. Die Daten können auch nach anderen Gesichtspunkten als den bisher vorgesehenen ausgewertet werden, sind also flexibel nutzbar.

3. Datenintegrität

Integrität der Datenbank bedeutet ganz allgemein Korrektheit und Vollständigkeit der abgespeicherten Daten. Dabei können drei Aspekte unterschieden werden:

Datenkonsistenz: Unter Konsistenz ist die Übereinstimmung des Inhalts der Datenbank mit der Datenbeschreibung und die Widerspruchsfreiheit zu verstehen. Dazu muß es möglich sein, Konsistenzbedingungen zu formulieren, deren Einhaltung während des Systembetriebs automatisch überwacht wird. Von Bedeutung für die Konsistenzhaltung ist die "Transaktion". Unter einer Transaktion wird eine Kombination von elementaren Operationen auf die Datenbank verstanden, welche diese von einem konsistenten Zustand in einen nächsten überführt. Transaktionen werden entweder ganz oder garnicht ausgeführt und eine abgeschlossene Transaktion wird durch spätere Ereignisse nicht beeinträchtigt.

Datensicherung: Durch die Integration aller Datenbestände in einer Datenbank stellt diese eine wertvolle Ressource ihrer Benutzer dar. Es sind daher umfangreiche Maßnahmen zum Schutz vor Verlust von Daten durch unbeabsichtigte oder beabsichtigte Zerstörung erforderlich.

Datenschutz: Die mißbräuchliche Verwendung von Daten wird durch Regelung der Zugriffsberechtigungen unterbunden.

4. Spezifische Benutzersichten

Für verschiedene Benutzer müssen die gewünschten Informationen über den Umweltausschnitt ("Miniwelt des Benutzers") in einer ihnen angepaßten Weise strukturiert werden. Die Benutzer brauchen keine Rücksicht auf gegebene Speichermedien und deren Speicher- und Zugriffspfadstrukturen zu nehmen.

Von entscheidender Bedeutung ist, daß die enge Abhängigkeit zwischen Anwenderprogrammen und Dateiorganisation aufgehoben wird. Der Anwender sollte keine Kenntnis über die physische Organisation der Daten benötigen, und Änderungen in der Datenspeicherung sollten nicht zu Änderungen in den Anwenderprogrammen zwingen. Diese Form der Unabhängigkeit wird als "physische Datenunabhängigkeit" bezeichnet (vgl. [12,8]). Zusätzlich wird im gewissen Umfang eine "logische Datenunabhängigkeit" angestrebt. Die logische Sicht eines Programmes auf die Daten sollte möglichst unabhängig sein von der Veränderung der logischen Gesamtsicht auf den Datenbestand.

3.1.2 Die drei Schemaebenen

Die Forderung, möglichst große physische und logische Datenunabhängigkeit zu erreichen, wird weitgehend erfüllt durch den Aufbau einer Datenbank nach dem 3-Schema-Konzept im Sinne des Normenvorschlags des ANSI/SPARC-Ausschusses [13]. Die Strukturierung der Daten erfolgt dabei in drei verschiedenen Ebenen in Form von Schemata (Bild 6):

- Die Sichten einzelner Anwenderprogramme bzw. Benutzergruppen werden in **externen Schemata** beschrieben.
- Das **konzeptionelle Schema** liefert eine logische Gesamtsicht aller Daten, die in der Datenbank verwaltet werden sollen. Es stellt den stabilen Bezugspunkt des Systems dar.
- Die physischen Speicherstrukturen und Zugriffspfade werden in **internen Schemata** definiert.

Das konzeptionelle Schema beschreibt ein Umweltmodell. Es enthält die Typen aller Gegenstände, Merkmale und Beziehungen, die den zu repräsentierenden Ausschnitt der realen Welt, die interessierende Umwelt ausmachen. Es wird davon ausgegangen, daß das konzeptionelle Schema eine stabile Basis sowohl für die externe Ebene als auch die interne Ebene bildet. Es stellt sozusagen den "ruhenden Pol" dar, der die Datenunabhängigkeit garantieren soll [14].

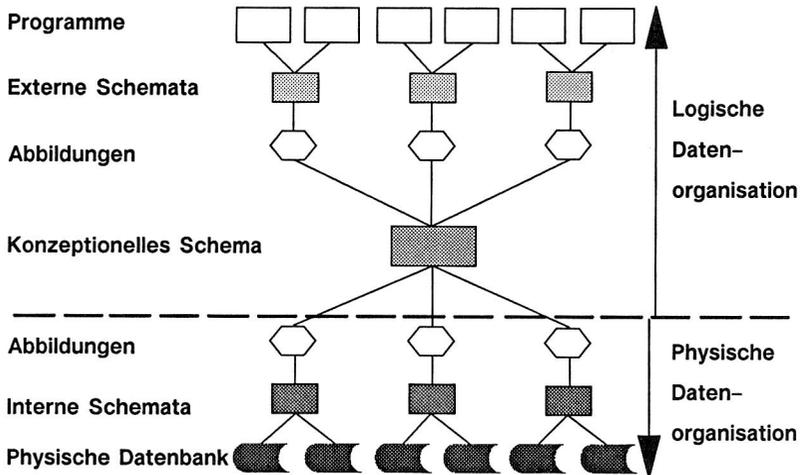


Bild 6: Datenbank-Aufbau nach ANSI/SPARC

Die internen Schemata beschreiben dasselbe Umweltmodell mit Begriffen der Speicher- und Zugriffspfadtechnik. Der Zusammenhang zwischen internen Schemata und konzeptionellem Schema wird über Abbildungen beschrieben zu deren Bestandteile u.a. Namenskorrespondenzen, Typkonvertierungen und Umformatierungen, Partitionierungen und Zusammenfassungen, Umsortieren, Verschlüsselungen, Anlegen von Zugriffspfaden, Zugriffsregelungen gehören [14]. Änderungen der internen Schemata haben bei gleichbleibendem konzeptionellem Schema Änderungen der Abbildungen zur Folge,

mit anderen Worten die Anpaßbarkeit der Abbildungen zwischen konzeptioneller und interner Ebene sichert die geforderte Datenunabhängigkeit.

Jeder individuellen Datenbasissicht einer Benutzergruppe auf der externen Ebene entspricht ein externes Schema. In ihm ist festgelegt, welche Ausschnitte aus dem Umweltmodell und damit welche Teile des konzeptionellen Schemas einer bestimmten Anwendung zugänglich sein sollen und in welcher Form.

Anwenderprogramme haben nur über ein externes Schema Zugang zur Datenbasis, wobei dasselbe externe Schema von einer beliebigen Zahl von Anwenderprogrammen genutzt werden kann, sofern diese mit demselben Umweltausschnitt zu tun haben. Für die Abbildungen zwischen externen Schemata und konzeptionellem Schema gilt sinngemäß das oben Gesagte. Muß ein externes Schema an eine veränderte Anwendung angepaßt werden, so ist nur die Abbildung zu ändern, das konzeptionelle Schema bleibt dasselbe. Die externen Schemata und das konzeptionelle Schema können als die Beschreibung der logischen Strukturen der in der Datenbank gespeicherten Daten (logische Datenorganisation), die internen Schemata als technische Beschreibung der physischen Strukturen (physische Datenorganisation) betrachtet werden (siehe Bild 6).

Der Datenbank-Aufbau nach ANSI/SPARC wurde einstimmig akzeptiert, so daß insbesondere Datenbankverwaltungssysteme von großen Software-Herstellern unter Einhaltung der Dreischichtigkeit konzipiert sind [7].

3.1.3 Architektur eines Datenbanksystem

Eine prinzipielle Datenbanksystem-Architektur, mit welcher die vier weiter oben angeführten Charakteristika verwirklicht werden können, ist in Bild 7 dargestellt. Es wird zwischen der Generierung und dem Betrieb einer Datenbank unterschieden, wobei vier Aktivitäten berücksichtigt werden: Die Datenbank-Generierung,

die Anwenderprogramm-Generierung, die schematische und gleichförmige Nutzung der Datenbank über Anwenderprogramme und die freie (ad hoc) Benutzung, bei der spontan (nicht langfristig vorbereitet) Informationen aus der Datenbank abgefragt werden (nach [15]).

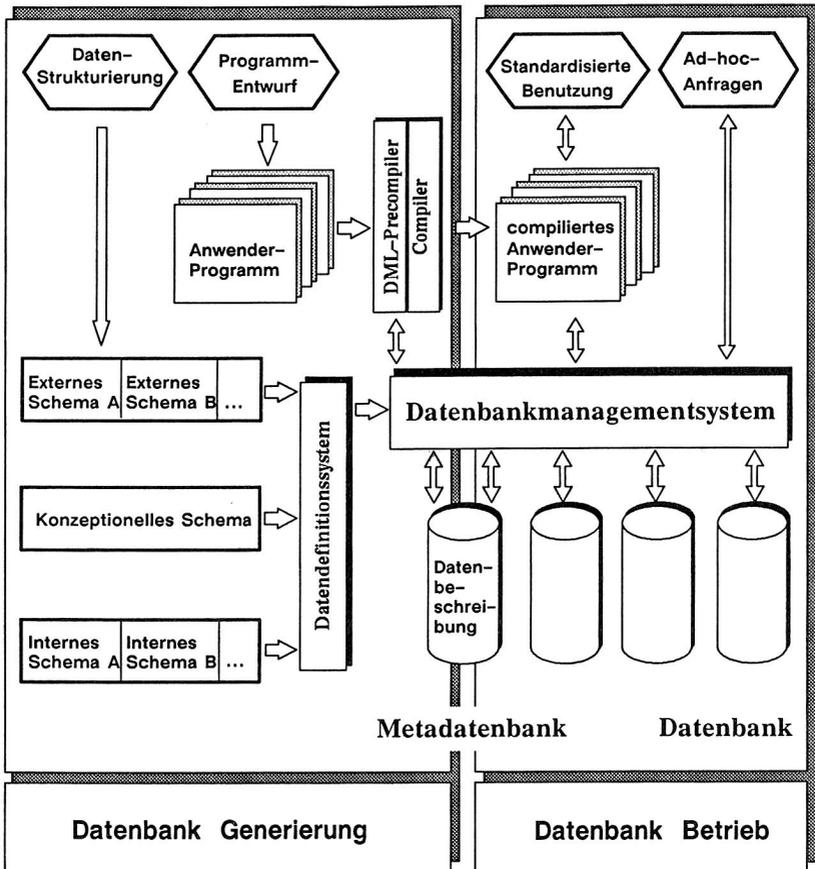


Bild 7: Architektur eines Datenbanksystem

Die Aufgaben der einzelnen Komponenten des Datenbanksystems sowie deren Zusammenspiel werden im folgenden kurz erläutert (siehe auch [8,9,11]).

Die Struktur der aufzubauenden Datenbank muß in einem bestimmten Datenmodell beschrieben werden. Je nach System werden die externen, die konzeptionellen und die internen Spezifikationen zur Generierung der Datenbank mittels entsprechender Datendefinitionssprache (DDL = Data Definition Language) mit Hilfe des Datendefinitionssystems als unterschiedliche Schemata formuliert. Diese Datenbeschreibung wird in einer Metadatenbank verwaltet. Der Einsatz einer Metadatenbank hat den Vorteil, daß für systeminterne Aufgaben Software verwendet werden kann, die für den Betrieb der Datenbank ohnehin vorhanden ist [11].

Die Anwenderprogramme werden in der Regel in einer herkömmlichen höheren Programmiersprache (Wirtssprache) geschrieben und enthalten zusätzliche, in einer DML (Data Manipulation Language) formulierte, Lese/Schreibbefehle als Verbindung zur Datenbank. Die Übersetzung von Anwenderprogrammen wird je nach verwendeter Datenmanipulationssprache mit einem vollständigen Sprachcompiler oder mit einem vorgeschalteten Precompiler vollzogen. Die für den Anschluß an das Datenbankverwaltungssystem erforderlichen Deklarationen werden aus der Metadatenbank geholt.

Das Datenbankverwaltungssystem (DBMS = Data Base Management System) ist ein Paket von Systemroutinen für die üblichen Funktionen auf Datenbanken (Suchen, Lesen, Schreiben), welche den Anwendern zur Verfügung stehen, wobei sämtliche Benutzer von der physischen Datenspeicherung abgeschirmt werden. Daneben umfaßt es noch eine ganze Anzahl nur intern aufrufbarer Funktionen, welche vor allem Maßnahmen zur Gewährleistung der Datenintegrität und der Synchronisation, d.h. der Koordination gleichzeitig auf der Datenbank arbeitender Benutzer, umfassen.

Die vorbereitete (standardisierte) Benutzung der Datenbank

erfolgt mittels der übersetzten Anwenderprogramme. Der Benutzer arbeitet nicht direkt mit den Komponenten des Datenbanksystems, sondern mit einem auf seine Bedürfnisse zugeschnittenen Anwenderprogramm, wo er zu vorbereiteten Standardoperationen höchstens noch gewisse Parameterwerte eingeben muß.

Die zweite Art der Benutzung, die freie Datenmanipulation, erlaubt es dem Benutzer, spontan formulierte Anfragen an das Datenbanksystem zu richten. Dazu steht ihm ein DML-Interpreter zur Verfügung, welcher nebst der Dialogfunktion zusätzliche Darstellungs- und Formulierungshilfen bietet.

3.2 Grundlagen der Datenmodellierung

Eine Datenbank stellt die rechnerinterne Repräsentation eines Ausschnittes der realen Welt dar. Das Original wird zwecks automatischer Verarbeitung durch ein Datenmodell ersetzt. Ein Datenmodell beschreibt in irgendeiner Sprache, die wegen der Forderung nach Genauigkeit möglichst eine formalisierte und keine natürliche Sprache sein soll, die Objekte und deren Verbindungen, d.h. die Struktur des Gegebenen. Je nach Modellwahl sind unterschiedliche logische Beschreibungen des Ausschnittes der realen Welt und auch unterschiedliche Implementierungen, d.h. auf einem Rechner lauffähige Systeme, die Folge [7].

3.2.1 Abstraktionsebenen der Informationsabbildung

Ziel des Datenbankentwurfs ist die möglichst naturgetreue Abbildung des durch Selektion gewonnenen, in der Datenbank zu repräsentierenden Ausschnittes (Universe of Discourse) der perzeptierbaren (durch Anschauung wahrnehmbaren) realen Welt auf ein vom Rechner interpretierbares Datenmodell (siehe Bild 8). Die Modellierung, d.h. der Abbildungsprozeß eines Originals auf ein Modell, entspricht der Zuordnung von relevanten Eigenschaften des Originals zu Eigenschaften des Modells.

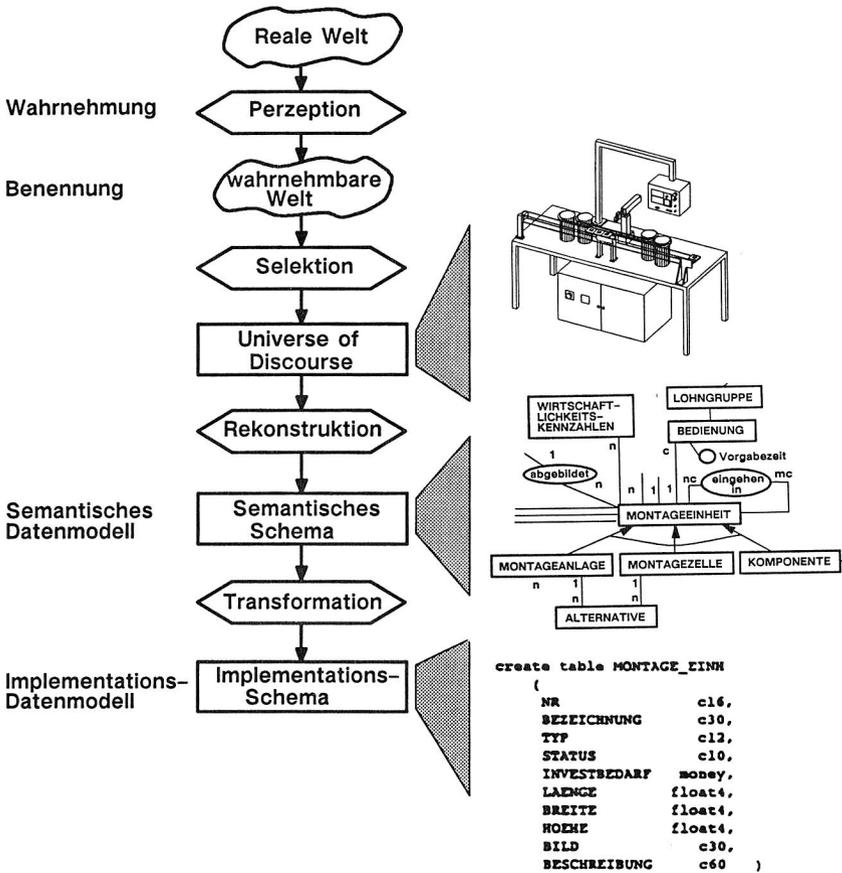


Bild 8: Abstraktionsebenen der Informationsabbildung

Die Vernachlässigung nicht relevanter Aspekte des Originals bei der Abbildung auf ein Modell wird auch Abstraktion genannt. Abstraktion ist die Fähigkeit, Details zu verdecken und sich auf das Wesentliche zu konzentrieren [16]. Die Qualität der durch die Abbildung auf ein Modell erzeugten Vernachlässigungen definiert die Höhe des Abstraktionsniveaus oder den dazu

umgekehrt proportionalen Detaillierungsgrad der Abbildung [17]. Der Abbildungsprozeß kann schrittweise erfolgen und sich über mehrere Modellebenen (Modelle auf verschiedenen Abstraktionsniveaus) erstrecken. Das Modell der Modellebene niederen Abstraktionsniveaus dient dabei als Original für die Modellierung des Modells der Modellebene nächsthöheren Abstraktionsniveaus.

Das Ergebnis eines solchen Modellierungsprozesses, die formale Beschreibung eines Modells, wird in der Datenbankterminologie als Schema, die Darstellung konkreter, nach dem Schema organisierter Daten als Schemaausprägung bezeichnet.

3.2.2 Die klassischen Datenmodelle

Zur programmunabhängigen Beschreibung von Daten in computerverständlicher Form ist ein bestimmter formaler Rahmen notwendig, das sogenannte Datenmodell. Die zur Zeit gebräuchlichsten, bei existierenden Datenbanksystemen tatsächlich eingesetzten Datenmodelle, in [12] und [18] auch die "klassischen" Datenmodelle genannt, sind das hierarchische, das netzwerkartige und das relationale Modell.

Die Eigenschaften der klassischen Datenmodelle wurden schon vielfach beschrieben und diskutiert. Es erfolgt hier nur eine kurze, zum Verständnis der weiteren Ausführungen erforderliche, Beschreibung dieser Datenmodelle anhand eines kleinen Beispiels, einem geometrischen Objekt (zwei Linien mit einem gemeinsamen Punkt), wobei jeweils Schema und Schemaausprägung vorgestellt werden. Für eine genauere Betrachtung sei auf die Literatur [20,9,8,7] verwiesen.

In den drei klassischen Datenmodellen wird davon ausgegangen, daß die wahrnehmbare Realität aus diskreten Objekten (Entities), ihren Eigenschaften, und Beziehungen zwischen den Objekten besteht. Durch Klassifikation der betrachteten Objekte werden Objekttypen gebildet, die eine Objektmenge repräsentieren. Die

Objekttypen werden durch Attribute näher beschrieben. Die Attribute sind Eigenschaften von Objekten, und es können identifizierende und beschreibenden Attribute unterschieden werden. Die Menge der identifizierenden Attribute wird Schlüssel genannt.

Zwischen den Objekten bestehen Beziehungen (Assoziationen), die durch einen analogen Klassifizierungsvorgang wie bei den Objekten zu Beziehungstypen zusammengefaßt werden. Die Beziehungstypen sind charakterisiert durch die Anzahl der beteiligten Objekte. Die Komplexität einer Beziehung wird dadurch bestimmt, mit wievielen anderen Objekten ein Objekt eines bestimmten Objekttyps in konkreter Beziehung steht.

Bei binären Beziehungstypen unterscheidet man [12]:

- m : n Beziehungstypen (keine Einschränkung)
- 1 : n Beziehungstypen (mit einem Exemplar des ersten Objekttyps können beliebig viele Exemplare des zweiten Typs verbunden sein, aber mit einem Exemplar des zweiten Objekttyps kann nur eines des ersten Typs verbunden sein)
- 1 : 1 Beziehungstypen (mit einem Exemplar des ersten Typs kann höchstens ein Exemplar des zweiten Objekttyps verbunden sein und umgekehrt)

Sie beschreiben die Funktionalität einer Beziehung [21].

Geht man bei der Beschreibung einer Beziehung zwischen zwei Objektmengen OM1 und OM2 von gerichteten Assoziationen aus [9] und berücksichtigt die Abhängigkeiten [21] zwischen den Objekten, so gelangt man zu einer verfeinerten Darstellung der Beziehungstypen. Eine gerichtete Assoziation ergibt sich dadurch, daß man festlegt, wieviele Objekte aus der Objektmenge OM2 einem Objekt aus der Objektmenge OM1 zugeordnet sein können. Es lassen sich die in Bild 9 dargestellten vier Assoziationstypen unterscheiden.

Assoziationsstyp (OM1->OM2)	Objekte aus OM2, die jedem Objekt aus der Menge OM1 zuzuordnen sind
1 : (einfache Assoziation)	genau eines
c : (konditionelle Assoziation)	keines oder eines (c=0,1)
n(m) : (multiple Assoziation)	mindestens eines (n,m>=1)
nc(mc) : (multiple konditionelle Assoziation)	keines, eines oder mehrere (nc,mc>=0)

Bild 9: Vier Assoziationsstypen (nach [9])

Kombiniert man eine Assoziation (OM1->OM2) dieser Betrachtungsweise mit ihrer Gegen-Assoziation (OM2->OM1), so ergibt sich eine Beziehung zwischen den beiden betrachteten Objektmengen, die die Abhängigkeiten zwischen den Objekten mit berücksichtigt (Bild 10). Zwischen den vier Assoziationsstypen gibt es 16 Kombinationsmöglichkeiten (die 12 im Bild 10 dargestellten plus vier aus der Umkehrung von 1:n in n:1). Berücksichtigt man Symmetrien, so reduzieren sich diese auf zehn.

Attribute dürfen in sich nicht weiter strukturiert sein, sie werden als atomar betrachtet. Eine Strukturierungsmöglichkeit ist nur über die Beziehungen gegeben. Jeder Bestandteil der wahrnehmbaren Realität, dem Charakteristika (Eigenschaften) zugeordnet werden, muß daher als Objekt dargestellt werden.

Im **hierarchischen Modell** werden Beziehungen zwischen Entities zu Hierarchien zusammengesetzt, die durch einen gerichteten Graphen (Assoziationsgraph) dargestellt werden können. Der übergeordnete Entitytyp wird "Vater", der untergeordnete "Sohn" genannt. Entitytypen ohne Vater werden als "Wurzel" bezeichnet. Wesentliches Kennzeichen des hierarchischen Modells

ist, daß jedes Entity maximal einen "Vater" haben darf.

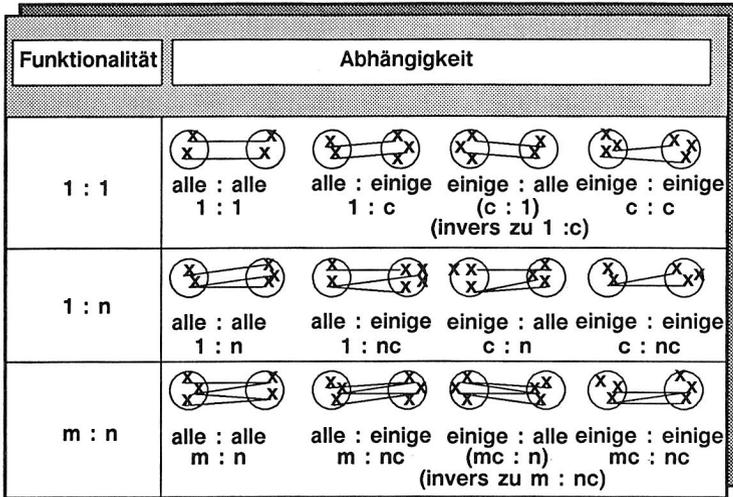


Bild 10: Beziehungstypen unter Berücksichtigung von Abhängigkeiten (nach [21])

Die Beziehungen zwischen den Entities können daher nur vom Assoziationstyp 1:1 oder 1:n sein. Zyklen sind nicht zugelassen. Je nachdem, wieviele Wurzeln auftreten, entstehen ein oder mehrere gerichtete Bäume. Bild 11 zeigt die Darstellung des gewählten Beispiels im hierarchischen Modell. Auf eine Angabe möglicher Attribute (Strichstärke, Farbe für die Linien, Koordinaten für die Punkte) wurde verzichtet.

Die Schemaausprägung zeigt, daß, wegen der Einschränkung auf 1:1 und 1:n Beziehungen, der Punkt P2 zweimal abgespeichert werden muß, wodurch Redundanz und damit die Möglichkeit zu Fehlern gegeben ist.

Im **Netzwerkmodell** wird im Gegensatz zum hierarchischen Modell erlaubt, daß ein Entity in mehreren Beziehungen der unterge-

ordnete Partner sein darf, so daß auch m:n Beziehungstypen dargestellt werden können. Auch Zyklen sind möglich, indem sie durch Einführung eines zusätzlichen Verkettungstyps aufgespalten werden. Zyklen sind jedoch nur im Assoziationsgraphen erlaubt, in der Realisierung darf ein Exemplar eines Objekttyps weder direkt noch indirekt sich selbst übergeordnet sein.

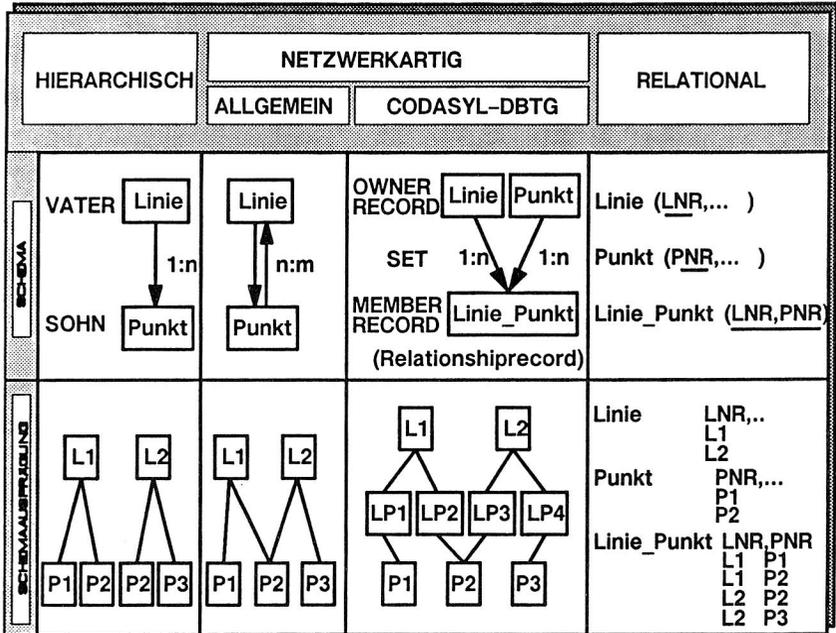


Bild 11: Die klassischen Datenmodelle (nach [17])

Eine weitverbreitete Sonderform des allgemeinen Netzwerkmodells ist das **CODASYL-Modell**. Die Definitionen dieses Modells wurden in einem 1971 von der Data Base Task Group (DBTG) eines CODASYL-Komitees (Conference on Data System Languages) veröffentlichten Report vorgestellt [22]. Kern des CODASYL-Vorschlages ist das SET-Konzept. Die Beziehung zwischen zwei Entitytypen wird als

SET-Typ, die zwischen zwei Entities als SET-Ausprägung bezeichnet. Entities werden RECORDS, Entitytypen entsprechend RECORD-Typen genannt. Das übergeordnete Element eines SET heißt OWNER, das untergeordnete Element heißt MEMBER. Zwischen OWNER und MEMBER ist nur der Beziehungstyp 1:n zugelassen. Zur Darstellung von n:m Beziehungstypen muß ein Beziehungs-RECORD (RELATIONSSHIP RECORD) eingeführt werden, der es ermöglicht diesen Assoziationstyp in zwei 1:n Beziehungstypen aufzuspalten (Bild 11).

Das **relationale Modell** wurde erstmals 1970 von E.F. Codd [23] vorgestellt. In ihm werden Beziehungen allgemein als Relationen wiedergegeben. Relationen stellen sowohl Beziehungen zwischen Entities und Attributen als auch Beziehungen zwischen Entities und Entities dar. Die Relationen können als Tabellen betrachtet werden (Bild 11). Der Name "Relationenmodell" leitet sich davon ab, daß die Beschreibungen der Tabelleninhalte in Anlehnung an den mathematischen Begriff der Relation formuliert werden können [12].

Betrachtet man A_1, A_2, \dots, A_n als Namen von Mengen und $W(A_1), W(A_2), \dots, W(A_n)$ als die zugehörigen Wertebereiche, so ist eine Relation definiert als Untermenge des Cartesischen Produktes der Wertebereiche

$$R(A_1, A_2, \dots, A_n) \subseteq W(A_1) \times W(A_2) \times \dots \times W(A_n).$$

Eine Tabelle besteht aus Zeilen und Spalten. Spaltenbezeichnungen sind Attributnamen, Spaltenwerte entsprechend Attributwerte. Jede Zeile entspricht einem Element (Tupel) der Relation. Die Zeilen sind paarweise verschieden und die Reihenfolge der Zeilen ist ohne Bedeutung. Jeder Tupel kann durch ein Attribut bzw. eine Attributkombination eindeutig identifiziert werden. Dieses Attribut bzw. die Attributkombination wird Primärschlüssel (primary key) genannt und durch Unterstreichen in der Tabelle kenntlich gemacht. Wird in einer Relation R1 ein Attribut verwandt, das in einer anderen Relation R2 Primärschlüssel ist, so spricht man innerhalb R1 von einem Fremdschlüssel (foreign

key). Die Beziehungen zwischen Entities werden über Fremdschlüssel dargestellt. Der Wertebereich einer Spalte heißt Domäne. Die Werte müssen atomar, d.h. nicht weiter zerlegbar sein.

Das relationale Datenmodell unterscheidet sich vom hierarchischen und vom netzwerkartigen Datenmodell im wesentlichen durch die Art der Informationsdarstellung. Während in relationalen Datenmodellen benutzerbezogene Information nur durch den Inhalt von Datensätzen ausgedrückt werden kann, lassen hierarchische und netzwerkartige Datenmodelle die Darstellung von Information durch "Verbindungen" von Datensätzen und durch ihre Anordnung in Speicherungsstrukturen zu [24]. Die Sicht der Daten ist also mit Aspekten des Zugriffs vermischt. Dies führt zu einer stärkeren Bindung der Anwenderprogramme an die vorhandenen Zugriffsstrukturen, so daß ein geringerer Grad an Datenunabhängigkeit erzielt wird. In der Regel ist auch nur ein Zugriff auf die Daten durch prozedurale Sprachen, auch navigierende Sprachen genannt, möglich. Das Charakteristikum des Zugriffs durch Navigation ist, daß ausgehend von einem Startobjekt in der Struktur (Knoten in der Schemaausprägung) der Weg über bestimmte Zwischenobjekte bis zum Zielobjekt festgelegt wird [12]. Die Daten werden satzweise aus der Datenbank gelesen oder satzweise in die Datenbank geschrieben ("one-record-at-a-time-logic") [15]. Der Benutzer muß die möglichen Wege (logischen Zugriffspfade) zwischen den Objekten genau kennen.

Relationale Datenmodelle kennen nicht den Aspekt des Zugriffs, der durch Kenntnisse des Benutzers über Verbindungen und Ordnungen der Datensätze verkörpert wird. Da alle Informationen durch Dateninhalte dargestellt werden, genügen deskriptive Sprachen zum Auffinden einer Information [24]. Deskriptive Sprachen sind mengenorientiert, d.h. der Programmierer kann jetzt ganze Mengen von Sätzen mit gewissen Eigenschaften verlangen.

3.2.3 Semantische Datenmodelle

Die drei klassischen Datenmodelle sind wegen ihres Mangels an Semantik kritisiert worden. Sie zwingen den Benutzer, mehr in Begriffen computerbezogener Konzepte zu denken als an die natürliche Struktur der Daten. Dies ist am offensichtlichsten bei hierarchischen und netzwerkartigen Modellen, wo logische Beziehungen mit physischen Zugriffspfaden assoziiert werden. Auch das relationale Modell beschäftigt sich mehr mit der Organisation der Daten für operationale Zwecke als mit dem Gebrauch natürlicher Strukturen [25]. Sowohl das hierarchische als auch das netzwerkartige Datenmodell basieren also auf Modellierungsmechanismen, die eher eine effiziente Speicherung und Verwaltung der Daten ermöglichen, als die direkte Abbildung der Strukturen und Operationen eines komplexen datenintensiven Anwendungsgebietes. Das relationale Datenmodell abstrahiert noch am meisten von den Einheiten der speziellen Speicherstrukturen [26]. Um die "Semantische Lücke" zwischen der (wahrnehmbaren) realen Welt und den klassischen Datenmodellen zu schließen, entstanden eine Vielzahl weiterer Datenmodelle, deren Modellierungsmerkmale sich mehr an der realen Welt orientieren. Diese Datenmodelle, die sich mit dem Entwurf von Datenbankanwendungen auf einer benutzernahen fachlichen Ebene befassen, faßt man heute unter der Bezeichnung "semantische Datenmodelle" zusammen [27].

Wichtige Klassen semantischer Datenmodelle sind (vgl. [12,18, 25,28,29]):

- Binäre Assoziations-Modelle [30,31]
- Entity (-Attribut) -Relationship Modelle [32]
- Erweiterte relationale Modelle [33,34,35,36].

Das Entity-Relationship Modell (ER-Modell) nach Chen [32] ist gegenwärtig wahrscheinlich das am weitesten verbreitete semantische Datenmodell [37]. Es wurde durch zahlreiche andere Autoren erweitert; viele Erweiterungen wurden in den Procee-

dings von vier Entity Relationship Konferenzen präsentiert, die 1979, 1981, 1983 und 1985 stattgefunden haben [38,39,40, 41].

Das ER-Modell betrachtet die Welt im Sinne von Entities und Beziehungen zwischen Entities. Ein Entity ist ein (reales oder konzeptionelles) Objekt oder ein Ereignis. Eine Beziehung ist eine Assoziation zwischen zwei Entities oder unter mehreren Entities. Entities und Beziehungen können Eigenschaften haben. Diese werden ausgedrückt durch Attribute/Wertebereiche. Zusammen mit dem ER-Modell stellt Chen auch eine Diagrammtechnik (Entity-Relationship Diagramme) vor, zur Repräsentation der logischen Datenstrukturen auf bildliche Weise. In einem Entity-Relationship Diagramm werden Entitytypen durch Rauten, Wertebereiche durch Kreise und Attribute durch Pfeile dargestellt. Die Beziehungstypen (1:1, 1:n, n:m) werden ebenfalls eingetragen (Bild 12).

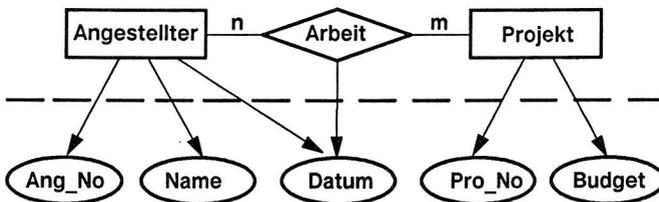


Bild 12: Beispiel eines ER-Diagramm (nach [38])

Die Popularität des ER-Modells als Zugang zum Datenbankentwurf kann wahrscheinlich dieser Diagrammtechnik zugeschrieben werden.

Es gibt zwar Implementierungen semantischer Datenmodelle, ihre Bedeutung, insbesondere die des Entity-Relationship-Modells, liegt heute jedoch noch hauptsächlich im Gebrauch für die Schemadefinition auf benutzernaher Ebene. Sie dienen meist zur Zwischenrepräsentation (auf Papier) bei der Datenmodellie-

rung, wenn für die Implementierung klassische Datenmodelle eingesetzt werden [42].

3.2.4 Abstraktionsmechanismen

Wie schon weiter oben erwähnt, versteht man unter Abstraktion ein Verfahren, um von unwesentlichen Eigenschaften eines Gegenstandes abzusehen und wesentliche hervorzuheben.

Die wichtigsten semantischen Abstraktionskonzepte zur Bildung von Objekthierarchien sind die Klassifizierung, Generalisierung, Aggregation und Gruppierung [26,18].

Die zentrale Struktur zur Beschreibung von Entitäten ist die Klasse. Entitäten werden klassifiziert, und durch die Beschreibung einer Klasse (Objekt) werden die ihr zugehörigen Entitäten bestimmt [43]. Die Klassen können nach drei unterschiedlichen Prinzipien hierarchisch strukturiert werden, nämlich durch Generalisierung, Aggregation und Gruppierung.

Bei der Generalisierung werden eine Gruppe ähnlicher Objekte (Artobjekte) zu einer Objektklasse oder auch zu einem neuen abstrakten Objekt (generic object) höherer Ebene (Gattungsobjekt) zusammengefaßt werden (z.B. KUNDE und LIEFERANT zu GESCHÄFTSPARTNER). Die Umkehrung, d.h. die Bildung von Subtypen, wird Spezialisierung genannt. Auf der Ebene des generischen Objektes werden Unterschiede der Subtypen vernachlässigt während Gemeinsamkeiten nachdrücklich betont werden. Es entsteht eine "is-a"-Beziehung zwischen Objekten.

Aggregation ist eine Form der Abstraktion, bei der Beziehungen zwischen Objekttypen zu einem Objekttyp höherer Ebene zusammengefaßt werden, wie z.B. die Beziehung zwischen den Objekttypen KUNDE, ARTIKEL und DATUM zum Objekttyp AUFTRAG. Es entsteht eine "part-of"-Beziehung zwischen Objekten.

Die Betrachtung von mehreren Entitäten eines Objekttyps als eine einzige, komplexe Entität (Mengenobjekt) ist die Grundlage der Gruppierung. Ein Beispiel ist die Zusammenfassung von MITARBEITERN zu einer ABTEILUNG. Es entsteht eine "member-of"-Beziehung zwischen dem Mitgliedsobjekt und dem Mengenobjekt.

Ähnliche Abstraktionsmechanismen finden sich, unter anderen Bezeichnungen, in den meisten semantischen Datenmodellen. So wird die Klassifizierung auch als Prädikation [27] oder Subsumption [7] bezeichnet oder mit der Generalisierung zusammengefaßt [33,34]. Die Generalisierung entspricht der Inklusion in [27], der Subordination in [7]. Die Aggregation ist gleichzusetzen der Konnexion in [27], der Komposition in [7], der Cartesian Aggregation im RM/T (Relational Model/Tasmania) [12]. Die Gruppierung kann verglichen werden mit der Aggregation in [27], der Reduktion in [7], der Cover Aggregation im RM/T [34] oder der Association von Brodie [44].

3.2.5 Integritätsbedingungen

Die durch die semantische Modellierung strukturierten Daten sollen weitgehend den Abhängigkeiten und Bedingungen gehorchen, denen die durch die Daten repräsentierten realen Objekte unterworfen sind (oft als "logische Konsistenz" bezeichnet) [42]. Die Restriktionen werden als Integritätsbedingungen bezeichnet. Die Integritätsbedingungen können auf verschiedene Weise klassifiziert werden (z.B. [24,25,45,46]).

Prinzipiell wird zwischen Integritätsanforderungen unterschieden, die sich auf den Zustand der in der Datenbank gespeicherten Daten beziehen, und solchen, die die Korrektheit von Transaktionen betreffen, d.h. Operationen zur Änderung der Daten. Die ersten sind statischer Natur und werden unter dem Begriff Datenintegrität zusammengefaßt, die zweiten sind dynamischer Art und gewährleisten die operationale Integrität (Bild 13).

Eine Ermittlung der Datenintegrität im Hinblick auf die Datenbankarchitektur umfaßt nach Steinbauer [46] die in Bild 13 aufgeführten Integritätsanforderungen.

Integritätsbedingungen	
Zustand (statisch) Datenintegrität	Transaktionen (dynamisch) operationale Integrität
<ul style="list-style-type: none">● Physische bzw. Technische Datenintegrität● Strukturelle Datenintegrität● Semantische Datenintegrität● pragmatische Datenintegrität	<ul style="list-style-type: none">● Korrektheit der Implementierung● Ablaufintegrität als strukturelle Eigenschaft● Integrität des Übergangs● deontischer Aspekt operationaler Integrität

Bild 13: Klassifikation der Integritätsbedingungen (nach [46])

Die physische oder technische Datenintegrität, auch häufig Speicherkonsistenz genannt, beinhaltet die Aufrechterhaltung der korrekten Umsetzung der logischen Datenstrukturen in physische Datenstrukturen. Es ist zu gewährleisten, daß ein speicherkonsistenter Zustand der Datenbank existiert bzw. nach verschiedenen Fehlerfällen (Systemausfall, Mediumfehler...) wiederherstellbar ist. Aus der Sicht der Anwendung ist die Wahrung der technischen Datenintegrität gleichbedeutend mit der Gewährleistung der Zuverlässigkeit des Mediums, auf dem die Daten abgelegt sind.

Die Anforderungen struktureller Datenintegrität ergeben sich aus dem verwendeten Datenmodell, sie sind somit datenmodell-

spezifisch. Jedes Datenmodell legt bestimmte Strukturen zugrunde, in denen die Begriffe der Miniwelt darzustellen sind, und bedingt dabei Restriktionen an die Form der Darstellung. Die strukturellen Integritätsanforderungen sind direkt im Datenmodell repräsentiert und können nicht vermieden werden. In bezug auf das Datenmodell werden sie inhärente Integritätsbedingungen genannt. Sie stellen die grundlegenden semantischen Eigenschaften eines Datenmodells dar [18].

Unter dem Begriff semantische Datenintegrität werden in der Literatur eine ganze Reihe von Integritätsanforderungen subsumiert (z.B. auch strukturelle Eigenschaften). Die eigentliche semantische Datenintegrität bezieht sich auf die Notwendigkeit, die durch das konzeptionelle Schema syntaktische möglichen Datenbankzustände durch Angabe von Bedingungen auf eine "sinnvolle" Teilmenge von Zuständen einzuengen, deren dargestellte Sachverhalte in der Miniwelt auch wirklich auftreten können.

Den Restriktionen der semantischen Datenintegrität entsprechen Bedingungen, denen die durch die Daten repräsentierte reale Welt tatsächlich unterliegt. Daneben gibt es noch Bedingungen, denen sie kraft einer normativen Regelung unterliegen sollte. Diese Konsistenzbedingungen aufgrund von gesetzlichen oder organisatorischen Vorschriften sind Gegenstand der pragmatischen Datenintegrität. Aufgabe der semantischen Datenintegrität ist es also, im Sinne der Anwendung "unmögliche" oder "sinnlose" Zustände auszuschließen, während die pragmatische Datenintegrität "unzulässige" Zustände einer Anwendung verhindern soll, die zwar auftreten können, aber nicht erwünscht (d.h. verboten) sind. Unzulässige, verbotene Zustände können durchaus sinnvoll, aber nicht konform mit den geltenden "Gesetzesnormen" sein.

Der operationale Aspekt der Integritätskontrolle beinhaltet die Gewährleistung der Korrektheit der Zustandsübergänge. Analog zur Datenintegrität werden in [46] vier Klassen von operationalen Integritätsbedingungen (Bild 13) unterschieden.

Die Korrektheit der Implementierung gilt als gesichert, wenn man das Datenverwaltungssystem und somit auch die zur Verfügung gestellten Operationen als korrekt annimmt. Als Möglichkeit zur Sicherung der physischen Integrität ist das physische Recovery zu nennen, in dem physische Sicherungskopien von der Datenbank genommen werden. Dieser technische Aspekt wird oft als gegeben vorausgesetzt.

Die "strukturelle" operationale Integrität nach Härder [24] auch Ablaufintegrität genannt, umfaßt die Einhaltung der "Gray'schen Transaktionskriterien" (Konsistenz (Isolation), Atomarität, Dauerhaftigkeit) bei Aufbau und Ausführung von Transaktionen.

Die semantische Datenintegrität grenzt die möglichen Zustände in einer Anwendung ab, dem entspricht auf der operationalen Seite die Festlegung der möglichen Übergänge. Die Übergänge können spezifiziert werden, indem durch Vor- und Nachbedingungen angegeben wird, unter welchen Voraussetzungen das Ausführen einer spezifischen Operation zu welchem Resultat führt, wobei die Wirkung der Transaktion durch die Nachbedingung bestimmt wird.

Der deontische Aspekt operationaler Integrität entspricht der Forderung, daß sich die Ausführung von Transaktionen mit den geltenden Normen in Übereinstimmung befinden. Es ist im konkreten Fall zu entscheiden, ob eine Transaktion normenkonform ist.

Die semantischen und pragmatischen Integritätsbedingungen werden auch als die explizite Integritätsbedingungen bezeichnet, ihr Gebrauch ist fundamentale beim Datenbankentwurf [18]. In welchem Umfang explizite Integritätsbedingungen in einer Datenbankanwendung realisiert werden, wird beim Datenbankentwurf entschieden und wirkt sich entscheidend auf das Leistungsverhalten des Datenbank-Anwendungssystems aus [46]. Meist wird auf eine Unterscheidung zwischen semantischen und pragmatische Integritätsbedingungen verzichtet.

3.3 Der Datenbankentwurf

Der Datenbank-Entwurfsprozeß dient zur Transformation unstrukturierter Information, über verschiedene Zwischenrepräsentationen, in vom Computer interpretierbare Schemadefinitionen.

Der Datenbankentwurf stellt sich als zweifache Aufgabenstellung im Vorfeld der Datenbank-Implementierung dar. Zum einen handelt es sich um die Aufgabe, eine logische Datenbankstruktur aufzubauen (logischer Datenbankentwurf), zum anderen handelt es sich um die Abbildung dieser logischen Strukturen auf physische Speichermedien (physischer Datenbankentwurf) (siehe auch Bild 6). Das Ziel im ersten Teil des Datenbankentwurfs ist, eine flexible logische Datenbankstruktur zu entwickeln, die an sich verändernde Benutzerumgebungen angepaßt werden kann. Der zweite Teil betont das Tunen, um die Performance für gegenwärtig bekannte Anwendungen zu optimieren [47]. Die vorausgesetzte Datenunabhängigkeit, d.h. die strikte Trennung der logischen und physischen Datenorganisation, ermöglicht es, die internen Schemata beim logischen Datenbankentwurf nicht zu berücksichtigen.

Es wurden eine Vielzahl von Methodologien, Techniken, Methoden und Werkzeugen zur Durchführung des Datenbankentwurfs entwickelt. Umfangreiche Bibliographie zu diesem Thema finden sich in [16,48,49,50].

Eine Methodologie für den Datenbankentwurf ist eine integrierte Sammlung von Methoden und Techniken, die den kompletten Datenbank-Entwurfsprozeß unterstützen [50]. Eine Technik in diesem Zusammenhang liefert einen systematischen Weg, um einen Teil dieses Entwurfs durchzuführen. Sie erfüllt nicht die Anforderungen der Integration und Abgeschlossenheit, die von einer Methodologie erwartet werden. Die Technik braucht nicht formalisiert zu sein [50]. Eine Methode ist ein organisierter Satz von Ideen, die benutzt werden, um eine spezielle Handlung auszuführen. Sie ist ein formaler Weg, um nur einen Entwurfsschritt auszuführen [50]. Unter einem Werkzeug wird ein Software-Produkt verstanden,

das die rechnerunterstützte Anwendung von Methoden ermöglicht.

Die folgenden Ausführungen konzentrieren sich auf die Darstellung von Methodologien, Techniken und Methoden.

Es gibt zwar einige rechnergestützte Werkzeuge zur systematischen Durchführung des Datenbankentwurfs, die jedoch mehr für kommerzielle Anwendungen geeignet erscheinen. Im "nicht kommerziellen" Bereich hängt die Qualität des Datenbankentwurfs weiterhin von der Entwurfserfahrung des Designers ab [6].

3.3.1 Komponenten des Datenbank-Entwurfsprozesses

Es lassen sich zwei Klassen von Methodologien zum Datenbankentwurf unterscheiden, die ebenenorientierten und die phasenorientierten. Die phasenorientierten Ansätze (z.B. [47,50,51,53,54]) betrachten mehr den Ablauf des Entwurfsprozesses im Sinne der Durchführung eines Projekts. Die ebenenorientierten Ansätze (z.B. [55,56,57]) sind dagegen mehr produktbezogen und nehmen eine Aufteilung nach verschiedenen Abstraktionsebenen vor, in Anlehnung an Praktiken und Vorgehensweise der Konstruktionssystematik des Maschinenbaus.

Bis jetzt wurde noch keine Einigkeit darüber erreicht, welche Komponenten der Datenbank-Entwurfsprozeß haben muß [49].

Vernachlässigt man einzelne Verfeinerungsstufen, so finden sich in fast allen Methodologien die vier in Bild 14 aufgeführten prinzipiellen Komponenten.

Die erste Komponente ist die Bedarfsanalyse in der, aufgrund einer Aufgabenstellung, ein Bedarf formuliert und eine Analyse der "realen Welt" durchgeführt wird. Sie dient dem Vertrautwerden mit dem Bereich, der modelliert werden soll, und der Festlegung des Funktions- und Leistungsumfangs der angestrebten DV-Lösung. Diese Komponente bildet die Grundlage für die nachfolgenden Modellierungskomponenten. Sie ist jedoch bei deren Beginn noch nicht endgültig abgeschlossen. Üblicherweise sind

baren Datenbankmanagementsysteme haben es ermöglicht, daß der Entwurf der physischen Datenstrukturen nicht mehr die dominierende Rolle spielt [11]. Die Qualität einer Datenbank wird heute durch den logischen Datenbankentwurf bestimmt, weshalb sich die weiteren Ausführungen auch auf diesen Teil konzentrieren.

Die in der Datenbanktechnologie verwendeten Begriffe sind nicht eindeutig festgelegt, es wird häufig über eine gewisse Begriffsverwirrung geklagt (z.B. in [20,49]). In den weiteren Ausführungen werden die Begriffe so verwendet, wie sie weiter oben eingeführt wurden.

3.3.2 Bedarfsanalyse und semantische Modellierung

Bei der Bedarfsanalyse ist festzustellen, welche Informationen der Umwelt für die geplante Anwendung von Interesse sind (die "Miniwelt" des Anwendungsfalls ist festzulegen). Der geplante "Inhalt" des Informationssystems ist also zu ermitteln. Dieser Schritt unterscheidet sich wenig von anderen Arten der Systemanalyse und es bietet sich an, die dort verwendeten Techniken und Hilfsmittel (siehe Abschnitt 4.1) einzusetzen [14].

Durch die semantische Datenmodellierung soll die im Rahmen der Informationsbedarfsanalyse abgegrenzte Miniwelt besser durchdrungen und präzise beschrieben werden. Die unter 3.2.3 genannten semantischen Datenmodelle sind sich inhaltlich zumindest in ihrer Grundkonzeption alle ähnlich, wenn sie auch durch möglichst individuelle Begriffswahl Unterschiedlichkeiten zu betonen versuchen. Generell wird nämlich (implizit) der Systembegriff der allgemeinen Systemtheorie als Gerüst zugrunde gelegt: Die Miniwelt wird beschrieben als eine Menge von Objekten (unserer Anschauung und unseres Denkens), den Systemelementen, zwischen denen wohldefinierte (System-) Beziehungen bestehen [14]. Semantische Datenmodellierung heißt also offensichtlich, eine beobachtete Umwelt mit Begriffen zu beschreiben, deren Bedeutung als bekannt und akzeptiert vorausgesetzt, bzw.

rekonstruktiv [7] erklärt wird.

In dieser Arbeit wird zur semantischen Datenmodellierung eine in Anlehnung an [21] modifizierte Form des Entity-Relationship-Modells (vgl.: [32]) benutzt.

Zur graphischen Darstellung der Schemata dieses semantischen Modells werden die in Bild 15 aufgeführten Bausteine verwendet.

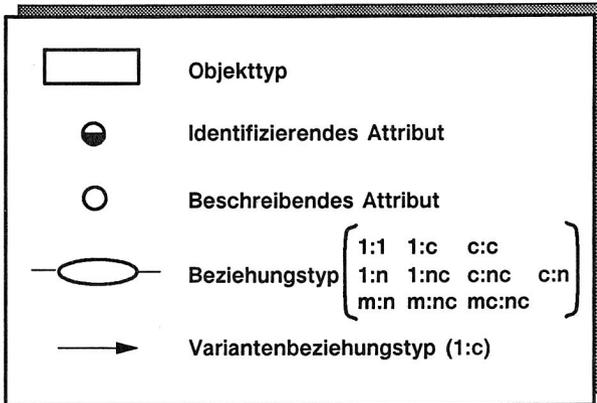
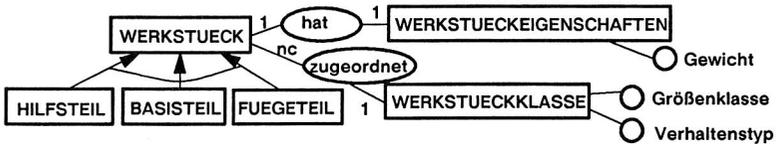


Bild 15: Grundbausteine der graphischen Darstellungsform eines semantischen Schemas

Ein Objekttyp repräsentiert eine Objektmenge. Objekttypen werden durch Attribute näher beschrieben. Attribute sind Eigenschaften von Objekten, und es können identifizierende und beschreibende Attribute unterschieden werden. Ein allgemeiner Beziehungstyp beschreibt die logische Beziehung zwischen Objekten zweier, nicht notwendigerweise verschiedener Objekttypen. Es werden die Beziehungstypen 1:1, 1:n und n:m sowie deren konditionale Ausprägungen 1:c, 1:nc, nc:m usw. (siehe Kapitel 3.2.2) unterschieden. Den m:n Beziehungen können Attribute zugeordnet werden, bei den anderen Beziehungen ist dies nicht sinnvoll. Bei selbsterklärenden Beziehungen wird häufig auf die explizite Benennung der Beziehung verzichtet und diese nur durch eine einfache Verbindung dargestellt.

Neben den allgemeinen Beziehungen wird noch ein weiterer Beziehungstyp unterschieden, der Variantenbeziehungstyp. Er ist graphisch an einer Pfeilspitze zu erkennen. Der Variantenbeziehungstyp kennzeichnet eine durch Generalisierung bzw. Spezialisierung entstandene "is-a" Beziehung [21] und ist vom Beziehungstyp 1:c. Bilden die Artobjekttypen eine Partition (disjunkte Teilmengen) des Gattungsobjekttyps, so wird dies durch eine gestrichelte Linie zwischen den die Partition bildenden Beziehungstypen gekennzeichnet.



WERKSTUECK (W#, Werkstuecktyp, Bezeichnung)

WERKSTUECKEIGENSCHAFTEN (WE#, Gewicht, Laenge, Breite, Hoehe,
Schwerpunktlage_XKOORD, Schwerpunktlage_YKOORD, Schwerpunktlage_ZKOORD)

WERKSTUECKKLASSE (WK#, Groessenklasse, Verhaltenstyp)

Groessenklasse >=1 und <=9
Verhaltenstyp >=0 und <=10

Bild 16: Beispiel eines vollständigen Schemas

Die Generalisierung erfordert bei der Abbildung auf ein formales Datenmodell eine besondere Behandlung, nämlich die Einführung eines diskriminierenden Attributes (siehe Abschnitt 3.3.4), und wird deshalb im semantischen Schema besonders herausgestellt. Die durch Aggregation und Gruppierung entstandenen hierarchischen "part-of" und "member-of" Beziehungstypen (siehe Abschnitt 3.2.4) werden in der Darstellung nicht vom allgemeinen Beziehungstyp unterschieden.

Werden in der graphischen Darstellung alle Einzelattribute mit aufgenommen, so wirkt diese sehr schnell überladen und wird unübersichtlich. Deshalb wird meist auf die Angabe von Attributen verzichtet und die graphische Darstellung ergänzt durch Begriffsschemata, in denen für jeden Objekttyp die zugehörigen Attribute angegeben werden.

Es wird eine Klammerschreibweise in der Form

Objekttypbezeichnung (Schlüsselattribute, weitere Attribute)

verwendet, wobei das oder die Schlüsselattribute durch Unterstreichen gekennzeichnet werden. Ergänzt wird das Schema durch die Angabe von Integritätsbedingungen (Bild 16).

3.3.3 Die Normalisierung

Ein wichtiges Instrument zur Überprüfung des Entwurfs des konzeptionellen Schemas ist die Normalisierung. Bei der Verwendung intuitiv aufgestellter Objekttypen treten häufig Mutationsanomalien auf, wenn unbeabsichtigt redundant gespeicherte Daten nicht mitmutiert werden. Die Normalisierung beschäftigt sich mit funktionalen Abhängigkeiten, mit dem Ziel, solche Anomalien vermeiden zu helfen. Mit dem Erkennen von funktionalen Abhängigkeiten wird ersichtlich, welche Attributwerte aus anderen Attributwerten abgeleitet werden können. Ziel der Normalisierung ist es, die Attribute so zu Objekttypen (und damit zu Relationen) zuzuordnen, daß innerhalb einer Relation keine Redundanzen auftreten. Die Normalisierung wurde im Zusammenhang mit dem Relationen-Modell entwickelt, kann jedoch auf andere Datenmodelle übertragen werden, da die zentralen Überlegungen modellunabhängig sind. So können die Normalisierungsregeln bereits auf die Objekttypen des semantischen Modells angewendet werden.

Die Normalisierung ist in der einschlägigen Datenbankliteratur ausführlich erläutert (z.B.: [7,8,9,20]). Deshalb erfolgt hier

nur eine kurze Darstellung.

1. Normalform: Eine Relation $R(A_1, A_2, \dots, A_n)$ heißt normalisiert oder in erster Normalform (1.NF), wenn ihre Wertebereiche $W(A)$ elementar sind, d.h. die Attribute A sind keine Relationen, d.h. mehrstellige Mengen.

2. Normalform: Eine Relation ist in der zweiten Normalform (2.NF), wenn sie in der ersten Normalform ist und jedes nicht zum Primärschlüssel gehörende Attribut voll funktional abhängig ist von diesem.

3. Normalform: Eine Relation ist in der dritten Normalform (3.NF), wenn sie in der zweiten Normalform ist und kein Attribut, das nicht zum Primärschlüssel gehört, transitiv von diesem abhängt.

Es wurden noch weitere Normalformen entwickelt, die jedoch mehr akademischen Charakter haben und von keiner praktischen Bedeutung sind [7].

Die Normalisierung erfolgt schrittweise. Man beginnt mit der Menge aller möglichen Relationsformen und unterscheidet normalisierte und unnormalisierte Relationen. Unnormalisierte Relationsformen haben als Attribute wieder Relationen.

Befindet sich eine Relation nicht in der gewünschten Normalform, so wird versucht, sie in zwei oder mehr Relationen zu zerlegen, die dieser Normalform genügen (Normalisierung). Dieses Verfahren wird solange durchgeführt, bis alle Relationen in der 3. Normalform sind.

3.3.4 Abbildung semantischer Schemata auf klassische Datenmodelle

Die entworfenen semantischen Schemata müssen im nächsten Schritt auf das Datenmodell eines zur Verfügung stehenden Datenbankmana-

gementsystems abgebildet werden, um es implementieren zu können.

Die meisten der zur Zeit verfügbaren Datenbankmanagementsysteme basieren auf dem Relationenmodell oder dem CODASYL-Modell. In ihrer Fähigkeit, Datenstrukturen abzubilden, unterscheiden sich beide Datenmodelle nicht grundsätzlich. Die wesentlichen Unterschiede zeigen sich in den Möglichkeiten der jeweiligen Datenbanksprachen, insbesondere im Komfort ihrer Benutzerschnittstellen und der Definition von Integritätsbedingungen. So gewährleisten z.B. CODASYL-Datenbankverwaltungssysteme in der Regel einen schnelleren Zugriff auf die Daten (höhere Performance), während relationale Systeme flexiblere Auswertungen der Daten ermöglichen. Zu beachten ist auch, daß die Erweiterung oder Veränderung der Datenstrukturen in CODASYL-Datenbanken nur mit sehr hohem Aufwand zu realisieren ist.

Die folgenden Ausführungen beschränken sich auf die Darstellung der Abbildungsregeln von einem Entity-Relationship-Schema entsprechend Abschnitt 3.3.2 auf ein relationales Schema, da zur Implementierung ein relationales Datenbankmanagementsystem eingesetzt wurde.

Im relationalen Modell stehen zur Darstellung von Information nur Relationen (Tabellen) zur Verfügung. Zur Beschreibung der Relationen wird folgende Darstellung verwendet:

$$R_K (\underline{P_1, \dots, P_n}, A_1, \dots, A_m).$$

R_K ist der Relationenname, P_1 das identifizierende Attribut, bzw. P_1, \dots, P_n die identifizierende Attributkombination, auch Schlüsselattribute genannt. Diese werden gegenüber den nicht-identifizierenden Attributen A_1, \dots, A_m durch Unterstreichen gekennzeichnet. Diese Notation entspricht der weiter oben eingeführten Notation für Begriffsschemata im Entity-Relationship-Schema. Die Beschreibung der Entities kann also direkt in das Relationenschema übernommen und um fehlende Attribute zur Darstellung der Beziehungen ergänzt werden.

Die Beziehungen zwischen Entities werden im Relationenschema durch Fremdschlüssel (siehe Abschnitt 3.2.2) ausgedrückt.

Zwei Relationen stehen genau dann in Beziehung, wenn das Schlüsselattribut (die Schlüsselattribute) der ersten Relation als Teil des Schlüssels oder als Nichtschlüsselattribut(e) in der zweiten Relation auftritt (auftreten). In der zweiten Relation wird (werden) dieses Attribut (diese Attribute) Fremdschlüssel genannt.

Die Beziehungstypen unter Berücksichtigung der Abhängigkeiten (siehe Abschnitt 3.2.2) können nach [9] in drei Klassen eingeteilt werden (Bild 17).

	1	c	n	nc	
1	1:1	c:1	n:1	nc:1	hierarchisch
c	1:c	c:c	n:c	nc:c	konditionell
m	1:m	c:m	n:m	nc:m	netzwerkförmig
mc	1:mc	c:mc	n:mc	nc:mc	

Bild 17: Beziehungstypen zwischen Objekttypen (nach [9])

Die hierarchischen Beziehungen lassen sich mit Hilfe eines Fremdschlüssels direkt abbilden (Bild 18).

Bei konditionellen Beziehungen ist die Einführung einer zusätzlichen Relation erforderlich, weil sonst den verbindenden Attributen in gewissen Fällen ($c=0$) kein Attributwert zugeordnet werden kann (Bild 19).

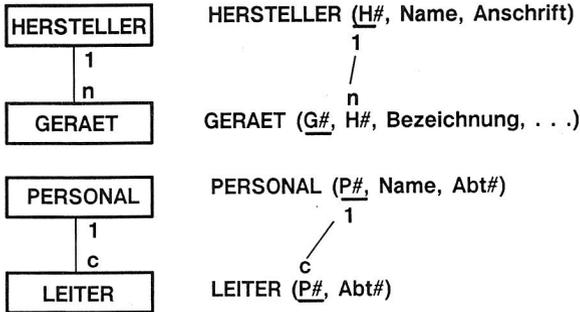


Bild 18: Abbildung hierarchischer Beziehungen vom Entity-Relationship-Modell auf Relationen (nach [9])

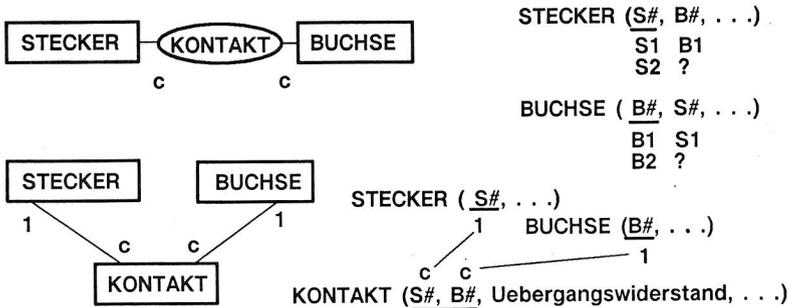


Bild 19: Abbildung konditioneller Beziehungen vom Entity-Relationship-Modell auf Relationen (nach [9])

Bei den netzwerkförmigen Beziehungen ist ebenfalls eine zusätzliche Relation notwendig, die auch die beschreibenden Attribute der m:n-Beziehung aufnimmt. Beim Versuch die Information direkt darzustellen würde eine Relation (im Beispiel die Relation FUNKTION) die 2. Normalform verletzen (siehe Bild 20). Eine m:n-Beziehung wird also auf zwei hierarchische Beziehungstypen 1:n/nc (Bild 20) so transformiert, daß eine Halbordnung [9]

der Relationen erreicht wird.

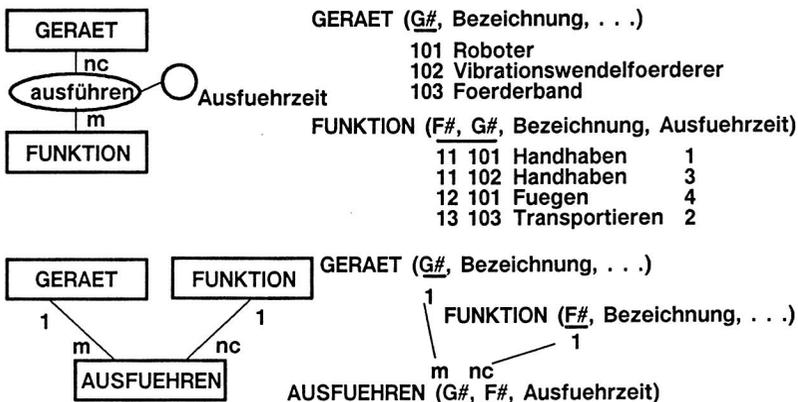


Bild 20: Abbildung netzwerkförmiger Beziehungen vom Entity-Relationship-Modell auf Relationen (nach [9])

Die Abbildung von Variantenbeziehungstypen erfolgt, indem in der Relation des Gattungsobjekttyps ein sogenanntes diskriminierendes Attribut aufgenommen wird, welches auf die Relationen der Artobjekttypen verzweigt. Die Schlüssel der Relation mit den Gattungsobjekten werden in die Relationen der Artobjekte dupliziert (Bild 21).

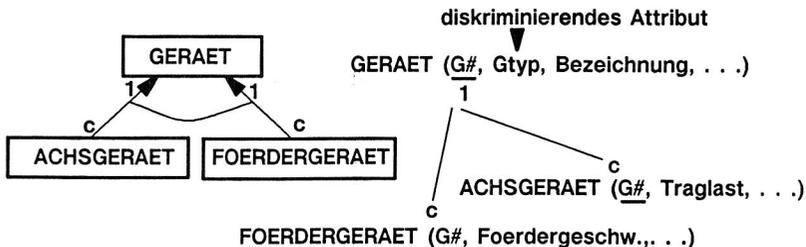


Bild 21: Abbildung von Variantentypbeziehungen des Entity-Relationship-Modells auf Relationen (nach [9])

Sind die Artobjekttypen nicht disjunkt, so muß eine andere Darstellung gewählt werden, bei der sämtliche möglichen Artobjekttypen als Attribute in die Relation des Gattungsobjekttyps aufgenommen werden und durch den Eintrag von "1" oder "0" auf die Artobjekttypen verzweigt wird (siehe [21]).

Die Abbildung von Entity-Relationship-Schemata auf ein CODASYL-Datenmodell funktioniert im Prinzip genauso, indem die Entities zu RECORDs und die Beziehungen zu SETs werden, wobei ähnlich wie im Relationenmodell zur Darstellung von m:n-Beziehungen sogenannte RELATIONSHIP RECORDs eingeführt werden. Eine genauere Darstellung der Transformationsregeln von einem Entity-Relationship-Modell in ein CODASYL-Modell kann beispielsweise in [14] nachgelesen werden.

3.4 Datenbanksystem für technische Anwendungen

In den letzten Jahren vermehrten sich die Bestrebungen, die mit dem Einsatz einer Datenbank verbundenen Vorteile auch für ingenieurwissenschaftliche Anwendungen zu nutzen. Es wurde versucht, die für den administrativ-betriebswirtschaftlichen Bereich entwickelten und bewährten Datenbankmanagementsysteme direkt oder in modifizierte Form auch für Ingenieur Anwendungen einzusetzen. Diese "nichtkommerziellen" Rechneranwendungen werden aus der Sicht heutiger Datenbanksysteme als Non-Standard-Anwendungen bezeichnet. Die Anforderungen an ein Datenbanksystem für Ingenieur Anwendungen unterscheiden sich jedoch teilweise drastisch von denen an ein Datenbanksystem für traditionelle Anwendungen im kommerziellen Bereich.

3.4.1 Anforderungen an Datenbanksystem für technische Anwendungen

In [10,60,61,62] finden sich ausführliche Diskussionen dieser unterschiedlichen Anforderungen. Die wichtigsten abweichenden

Anforderungen ergeben sich aus:

- Komplexen Objekten
- Unterschiedlichen Beschreibungsformen
- Konstruktionsaspekten
- Vielfalt der Strukturen
- Entwurfstransaktionen
- Komplexen Integritätsbedingungen.

Kennzeichen vieler technischer Anwendungen sind **komplexe Objekte**, deren Abbildung zu umfangreichen, stark strukturierten Daten führt (z.B. Layout einer Montageanlage). Komplexe Objekte setzen sich wiederum aus komplexen (Teil-) Objekten zusammen, auf die bei der Konstruktion solcher Objekte zurückgegriffen wird. Dieser Aufbau muß sich in der Datenbank widerspiegeln, wobei die Beschreibungen wichtiger Basisobjekte in entsprechenden Bibliotheken bereitzuhalten sind (informationelle Daten).

Technische Objekte besitzen oft mehrere **unterschiedliche Beschreibungsformen**, die sich als Einzelaspekte zu einer Gesamtbeschreibung ergänzen. So kann das gleiche Objekt in einem 3D-Modell und mehreren Zeichnungen beschrieben sein.

Ingenieurtätigkeiten sind häufig evolutionäre Prozesse, bei denen Zwischenergebnisse und Änderungszustände festgehalten werden müssen. Solche **Konstruktionsaspekte** führen in der Praxis dazu, daß von einem (Teil-) Produkt mehrere Alternativen entwickelt werden, die wiederum Versionen besitzen können. Die Verwaltung dieser Entwicklungsgeschichte sollte vom Datenbanksystem unterstützt werden.

Während kommerzielle Datenbankanwendungen sich durch eine relativ geringe Anzahl von Strukturen mit jeweils sehr vielen Ausprägungen auszeichnen, besitzen Ingenieur Anwendungen eine **Vielfalt der Strukturen**. Die Strukturen sind äußerst vielfältig, aber relativ wenige Datenobjekte sind von gleicher Struktur.

Im administrativ-betriebswirtschaftlichen Bereich involvieren **Transaktionen** gewöhnlich wenige Daten und sind von kurzer Dauer. Im Gegensatz dazu können Konstruktionsvorgänge, also Entwurfstransaktionen, Wochen und Monate dauern und es sind jeweils viele Daten betroffen (ein ganzes, nach obigen Erläuterungen komplexes Objekt).

Die meisten Ingenieur Anwendungen fordern die Einhaltung von recht **komplexen Integritätsbedingungen**, deren Überwachung durch ein Datenbanksystem äußerst schwierig ist. So ist die Widerspruchsfreiheit unterschiedlicher Beschreibungsformen eines komplexen Objektes zu gewährleisten. Weiterhin sind Zustandsübergänge auf Zulässigkeit zu überwachen (Einhaltung von Konstruktionsvorschriften).

Kommerzielle Datenbankmanagementsysteme (DBMS) weisen in bezug auf die genannten Anforderungen einige Schwächen auf [60].

Zur Überwindung dieser Schwächen wurden eine Reihe von Konzepten entwickelt, von denen die wichtigsten kurz erläutert werden. Für genauere Darlegungen wird auf die entsprechende Literatur verwiesen.

Bei der Erweiterung von konventionellen DBMS um **abstrakte Datentypen** können neben den angebotenen Grundtypen durch den Benutzer beliebige anwendungsbezogene Datentypen mit ihren Operatoren eingebracht und Attribute von diesen Typen definiert werden [10,60]. Damit wird es möglich, Kommandos der Datenbankabfragesprache als Attribute von Relationen zu verwenden [20].

Beim **Surrogatkonzept** werden Surrogate als systemweit eindeutige Schlüssel mit invarianten Werten für jedes Tupel vergeben [5]. Die Tupel einer Relation werden über diese Surrogatwerte identifiziert und nicht mehr über benutzerspezifische Merkmale, die zu Schwierigkeiten für die eindeutige Identifikation bzw. zu komplexen Schlüsselbegriffen führen können. Mit Hilfe des

Attributtyps SURROGAT und den speziellen Attributtypen PART_OF, IS_A und REFERENCE können damit in der Datenbank hierarchische und nicht-hierarchische Beziehungen zwischen Tupel unterschiedlicher Relationen bekannt gemacht werden.

Im sogenannten NF²-Relationenmodell [64,65] müssen sich die Relationen nicht mehr in 1. Normalform befinden, d.h. sie dürfen Attribute besitzen, die ihrerseits vom Typ Attribut sind, wobei eine beliebig tiefe Hierarchiebildung möglich ist [10] .

Zur Behandlung lang andauernder Transaktionen wird in [21] und [66] vorgeschlagen, daß der Entwerfer sich zu Beginn der Transaktion ein gewünschtes strukturiertes Objekt aus einer globalen Datenbank in einen lokalen Datenbestand kopiert. Nach Beendigung der langen Transaktion wird das (veränderte) Objekt in die globale Datenbasis zurückgeschrieben und überschreibt einen alten Datenbasiszustand. Das Datenbanksystem sorgt für die notwendige Synchronisation (Ausschaltung von Schreiber-Schreiber-Konflikten). Um bei Systemfehlern während der langen Transaktion nicht zwangsweise auf den Transaktionsanfang zurückgesetzt zu werden (und damit die Ergebnisse einer u.U. langen Entwurfsphase zu verlieren), hat der Anwender die Möglichkeit, bestimmte Datenbasiszustände zu sichern ("safepoints), auf die dann im Fehlerfalle zurückgegriffen werden kann. Für Katalogdaten ergeben sich keine Probleme der Synchronisation, da auf sie von Anwendern nur lesend zugegriffen wird.

Es sind einige Entwicklungen im Gange, in denen versucht wird, solche Konzepte umzusetzen [5,6,61,64,65]; bisher gibt es jedoch noch keine industriell einsetzbaren Produkte, bei denen diese Forschungsansätze verwirklicht wurden. Derzeit ist man für Datenbankrealisierungen immer noch auf die klassischen Datenbanksysteme angewiesen.

3.4.2 Integrationsstufen der Datenhaltung

Der Einsatz von Datenbanken dient der Integration der Daten. Diese Integration der Datenhaltung kann in zwei Richtungen betrachtet werden. Zum einen besteht das Bedürfnis, die Bearbeitung eines Datenbestandes, bzw. sich überschneidender Datenbestände, mit dem gleichen Anwendungssystem, jedoch durch unterschiedliche Benutzer, kontrolliert zu unterstützen (horizontale Integration), zum anderen soll die wechselseitige Bearbeitung von Daten durch verschiedene Anwendungssysteme ermöglicht werden (vertikale Integration).

Der Grad der vertikalen Integration der Datenhaltung für verschiedene Anwendungssysteme läßt sich nach [67] in fünf Integrationsstufen einteilen.

Die Stufe 0 ist durch Isolation gekennzeichnet. Sie entspricht der eingangs geschilderten Situation. Jedes Anwendungssystem arbeitet mit seinen eigenen Dateien. Änderungen in den Dateistrukturen und in den Ausprägungen der Daten des einen Systems werden nicht in das andere System durchgereicht, auch wenn die Dateien nicht disjunkt sind. Man muß mit mehreren Dateizuständen arbeiten, wobei die Frage der Gültigkeit problematisch ist (siehe Bild 22).

Die Stufe 1 wird durch Konversion erreicht (Bild 23). Bei N zu verknüpfenden Dateien werden durch $N \times (N - 1)$ Konverter in beide Richtungen zu definierten Ereignissen die Änderungen durchgereicht. Diese Stufe genügt häufig für kleinere überschaubare Anwendungen, an denen sowohl nur wenige unterschiedliche Benutzer als auch nur wenige verschiedene Anwenderprogramme beteiligt sind.

Zur Stufe 2 gelangt man durch sternförmige Konversion, womit ein entscheidender Schritt hin zur Zentralisation erreicht wird. Änderungen werden in neutrale (anwendungsunabhängige) Dateiformate hinein und aus ihnen heraus übertragen, wodurch

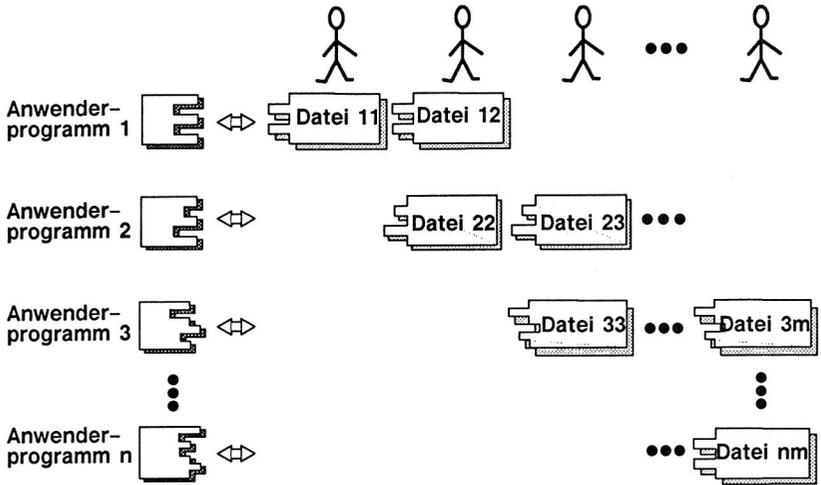


Bild 22: Integrationsstufe 0: Isolierte Datenhaltung in

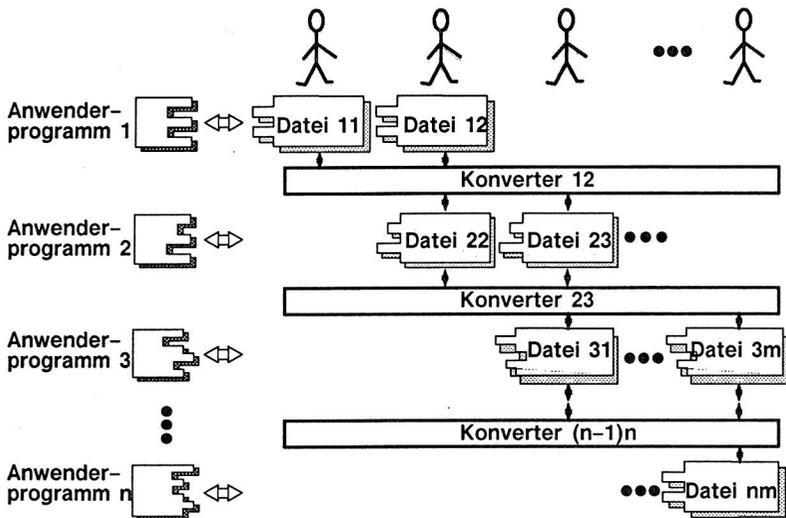


Bild 23: Integrationsstufe 1: Isolierte Datenhaltung in Dateien, die konvertiert werden können

nur noch $2 * N$ Konverter erforderlich sind (Bild 24). Diese Stufe gilt für CAD-Systeme als "state of the art". Mit der ANSI-Norm IGES (Initial Graphics Exchange Specification) wird diese Integrationsstufe angestrebt. Der IGES-Ansatz geht davon aus, daß diverse CAD-Turnkey-Systeme, die mit unterschiedlichen Modellierern arbeiten, über neutrale, normierte Datenformate Geometriedaten und Annotation (Bemaßung, Beschriftung) austauschen. Ähnliche Zielsetzungen wurden verfolgt mit SET (Standard d'Exchange et de Transfer), mit der VDAFS (VDA-Flächenschnittstelle) des Verbandes der deutschen Automobilindustrie, als Nachfolger dieser drei mit STEP (Standard for Exchange of Produkt Model Data) und im Elektronikbereich mit EDIF (Electronic Design Interchange Format).

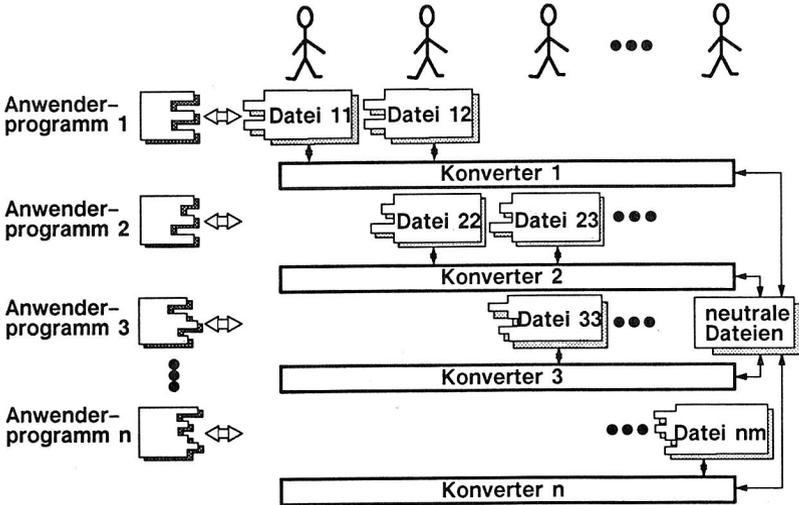


Bild 24: Integrationsstufe 2: Isolierte Datenhaltung in Dateien, mit der Möglichkeit sternförmiger Konversion

Die Integrationsstufe 3 basiert auf herkömmlichen Datenbanksystemen (Bild 25), die um nicht-standardmäßige Komponenten erweitert wurden [5,6,61]. Datenbanksysteme (Integrationsstufe 3) bieten beträchtliche Vorteile gegenüber neutralen Dateiformaten (Integrationsstufe 2), wie Recovery-Mechanismen,

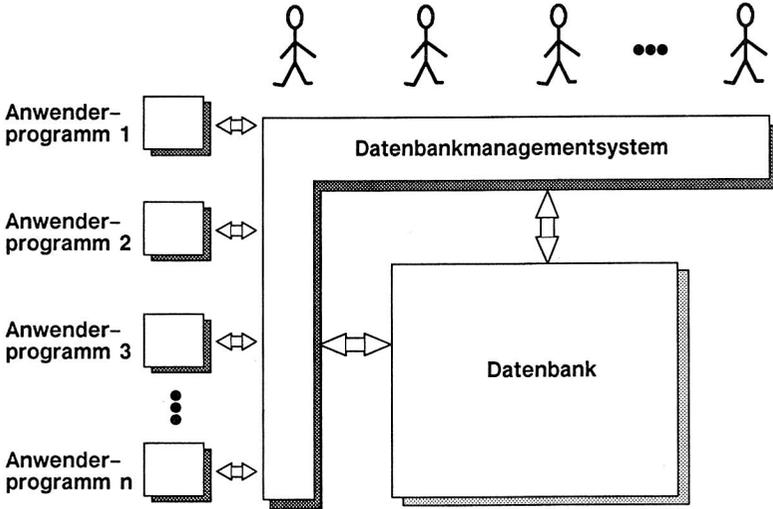


Bild 25: Integrationsstufe 3: Integrierte Datenhaltung in einer Datenbank

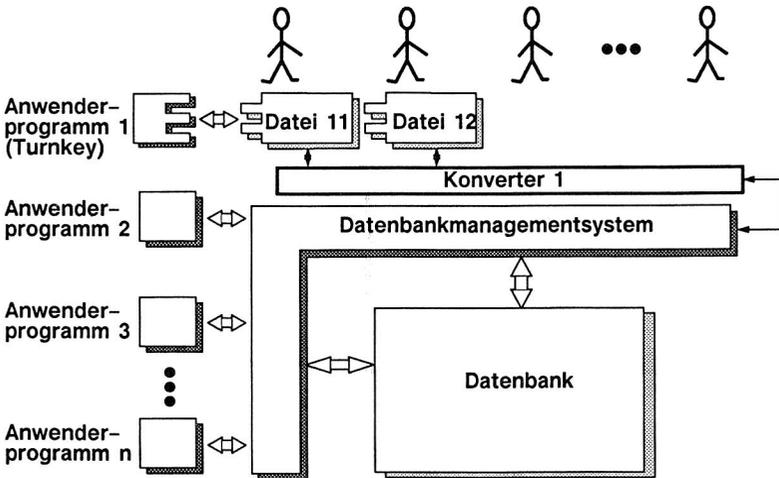


Bild 27: Integrationsstufe 4: Integrierte Datenhaltung in einer Datenbank, mit Einbeziehung von Turnkey-Systemen

geringe Datenredundanz, Transaktionskonzept, Synchronisation für Mehrbenutzerbetrieb und Integritätskontrolle. Wie in Abschnitt 3.4.1 dargelegt, sind diese für technische Anwendungen erweiterten Datenbanksysteme, insbesondere für CAD-Anwendungen, noch nicht ausgereift.

Berücksichtigt man weiterhin, daß die CAD/CAM-Anwendungen des Ingenieurbereichs weitgehend durch Fremdsysteme geprägt sind, so ist eine Integrationsstufe 4 gerechtfertigt. Wesentliches Kennzeichen dieser Integrationsstufe ist, daß schlüsselfertige Systeme ("turnkey systems") integrierbar sein müssen (Bild 26). Vereinfachend kann die Formel $\text{Stufe 4} = \text{Stufe 3} + \text{Stufe 2}$ eingeführt werden [67].

Betrachtet man diese Integrationsstufen unter dem Gesichtspunkt der horizontalen Integration, so zeigt sich, daß diese erst ab Stufe 3 gewährleistet werden kann, wenn konkurrierender Zugriff durch ein zentrales Datenbanksystem geregelt wird.

4. Systemtechnische Betrachtung automatisierter Montageanlagen

Automatisierte Montagesysteme stellen in der Regel komplexe technische Systeme dar. Die Systemtechnik liefert als interdisziplinäre Wissenschaft Methoden, Verfahren und Hilfsmittel zur Analyse und Beschreibung komplexer Systeme; sie eignet sich daher besonders für die Begriffsfindung als Voraussetzung zur semantischen Datenmodellierung.

4.1 Allgemeines zur Systemtechnik

Die Systemtechnik wird als die Technik betrachtet, die Grundlagen der interdisziplinären Systemwissenschaft, einer Wissenschaft vom "zweckrationalen und methodischen Handeln", einsetzt. Sie will damit Methoden, Verfahren und Hilfsmittel zur Analyse, Planung und optimalen Gestaltung komplexer Systeme bereitstellen und betrachtet sich deshalb als eine "Kunst des richtigen Handelns" für die Schaffung optimaler Systeme zu einem bestimmten Zweck [68]. Die instrumentelle Basis als grundlegende Denkweise im Rahmen der Systemtechnik ist der Systemansatz: Jede wissenschaftlich orientierte Arbeit basiert auf einem Ansatz, um Denkprozesse einordnen und Aussagen ableiten zu können. Der Systemansatz ist gekennzeichnet durch inhaltliche Allgemeingültigkeit und formale Abstraktheit, eine umfassende, zugleich strukturierende Betrachtungsweise und durch seine Zweckorientiertheit. Ausgangspunkt des systemorientierten Denkens der Systemtechnik ist die systemorientierte Strukturierung eines Gebildes oder einer gedanklichen Problemstellung als System.

4.1.1 Begriffe der Systemtechnik

Ein System besteht aus einer Menge von Elementen, welche Eigenschaften besitzen und welche durch Relationen miteinander verknüpft sind [69]. Die Menge der Relationen heißt Struktur des

Systems. Die Elemente eines Systems, Subsysteme oder Teilsysteme genannt, haben für sich wiederum Systemcharakter. Ein System kann also auf verschiedenen Hierarchieebenen betrachtet werden.

Ein System ist von seiner Umgebung abgegrenzt. Im Sinne der Systemhierarchie kann jedes System als Teilsystem eines nächsthöheren übergeordneten Systems betrachtet werden, wobei es als Komponente dieses übergeordneten Systems mit anderen Komponenten in Beziehung steht, und zwar in Form des Austausches von Materie, Energie und/oder Information sowie ordnungsmäßig bezüglich Zeit, Ort und Rang. Die Grenze des betrachteten Systems ist somit als Hüllfläche aufzufassen, die sämtliche Beziehungen mit der Systemumwelt schneidet.

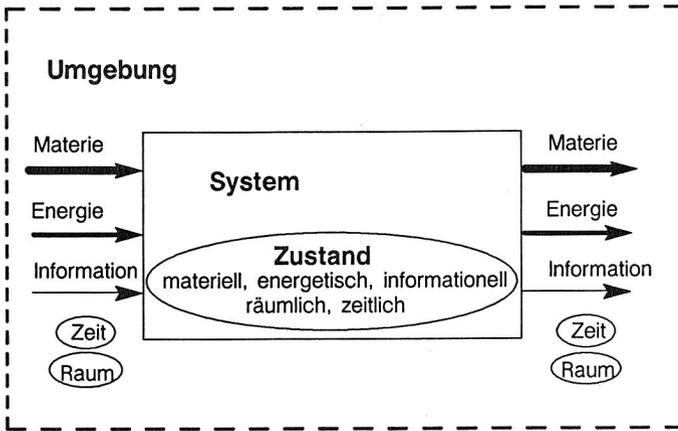


Bild 27: Elementarmodell zur Ermittlung von Systemeigenschaften

Die Festlegung der Systemgrenze wird bestimmt durch die Problemstellung und die Zweckmäßigkeit [69].

Der Begriff "System" besitzt Modellcharakter; es handelt sich bei der Systemvorstellung um ein Modell, das vom menschlichen Denken konstruiert wurde und an beliebige Gegenstände herangezogen werden kann [70]. Die Beschreibung eines Systems basiert

auf der Kenntnis der zeitpunktbezogenen Kombination aller Eigenschaften der systembildenden Elemente. Es lassen sich drei Eigenschaftskategorien unterscheiden:

- raumbezogene Eigenschaften
- zeitbezogene Eigenschaften
- inhaltsbezogene Eigenschaften, materie-energie-informations-
bezogene Eigenschaften

Damit ergibt sich, als Black-Box-Modell, die in Bild 27 gezeigte prinzipielle Darstellung, wenn für Input/Output sowie für den Systemzustand obige Kategorien von Eigenschaften für die Beschreibung herangezogen werden.

Die Struktur eines Systems ist die Abbildung der Menge der die Komponenten eines Systems miteinander verbindenden Relationen.

Je nach Betrachtungsweise können zwei Arten von Strukturen unterschieden werden:

- Aufbaustruktur (Gefüge von Ordnungsrelationen)
- Ablaufstruktur (Gefüge von Flußrelationen).

Wird der Systeminhalt auf sachliche Zusammenhänge hin betrachtet und werden dabei nur statische Beziehungen angegeben, so ergibt sich eine Aufbaustruktur. Der Aufbaustruktur liegt eine hierarchische Gliederung gemäß Bild 28 zugrunde [69].

Ein Teilsystem (Subsystem) enthält eine Teilmenge aller Elemente des Gesamtsystems und erfaßt für diese Teilmenge alle Beziehungen, die sich aus allen für das Gesamtsystem relevanten Gesichtspunkten zwischen diesen Elementen ergeben. Das Systemelement muß, ebenso wie die Systemgrenze, als Ergebnis eines denkökonomischen Prozesses angesehen werden: Systemelemente sind keineswegs unteilbare Bestandteile des Systems, sondern sind nicht sinnvoll weiter zu unterteilende Einheiten desselben

[69]. Die Zerlegung eines gegebenen Ganzen in Komponenten (Top-Down Analyse) bzw. die Zusammenfassung von Komponenten zu einem übergeordneten Ganzen (Bottom-Up Analyse) kann nach objektorientierten oder verrichtungsorientierten Gesichtspunkten erfolgen.

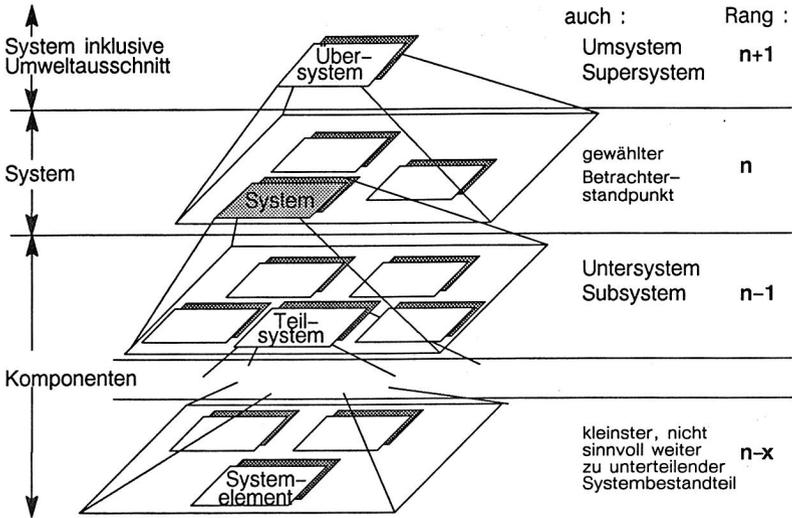


Bild 28: Allgemeine Aufbaustruktur von Systemen (Hierarchische Gliederung)

Die Erfassung einer Ablaufstruktur eines beliebigen Systems setzt voraus, daß das System einen dynamischen Charakter im weiteren Sinne besitzt, daß also Übertragungsabläufe oder Zustandsänderungsabläufe auftreten. Es ist nicht möglich, für abstrakte statische Systeme, wie sie beispielsweise Klassifikationssysteme darstellen, eine Ablaufstruktur zu finden. Die Relationen, aus denen eine Ablaufstruktur sich zusammensetzt, sind zeitlicher oder logischer Art [70] (Bild 29).

Strukturart	Aufbaustruktur	Ablaufstruktur
Charakteristika		
Definition	Darstellung des Systeminhalts gegliedert nach sachlichen Zusammenhängen der in einem System enthaltenen Komponenten	Darstellung der Systemfunktion als zeitlich/logische Anordnung der in einem System enthaltenen Komponenten zum Zwecke einer Aufgabenerfüllung
Betrachtungsweise	statisch	dynamisch
Darstellungsform	<ul style="list-style-type: none"> - Baumstruktur - Hierarchiestruktur - Matrix 	<ul style="list-style-type: none"> - gerichteter Graph (Netzplan, Entscheidungs bäume) - Programmablaufplan Flußdiagramm (Personen, Material, Energie, Daten) Blockschaltbild - Matrix
Analyseverfahren	Aufbauanalyse Systemgliederung	Ablaufanalyse Funktionsanalyse
Charakteristische Analyse/Synthese Begriffe	zerfällt setzt sich zusammen	liegt nach, folgt liegt vor, erfordert

Bild 29: Gegenüberstellung von Aufbaustruktur und Ablaufstruktur (nach [69])

4.1.2 Klassifikationssystematik für Systeme

Systeme können nach verschiedenen Gesichtspunkten klassifiziert werden (Bild 30). Zweck einer derartigen Einteilung der Systeme unserer Erfahrungswelt ist es, Kategorien zu schaffen und Gemeinsamkeiten zur leichteren Bearbeitung herauszuschälen [69].

Bei Montageanlagen handelt es sich um offene, konkrete, künstliche, dynamische Systeme. Jedes künstliche System dient einem bestimmten Zweck, der durch ein Zielsystem beschrieben werden

kann. Ein Ziel ist ein als möglich vorgestellter Sachverhalt, dessen Verwirklichung erstrebt wird; das Zielsystem ist eine Menge von Zielen mit gegenseitigen Beziehungen.

Merkmale	Merkmalsausprägung	
	konkret	abstrakt
Entstehungsart	natürlich	künstlich
Beziehung zur Umwelt	offen	geschlossen
Zeitabhängigkeit (Funktion)	statisch	dynamisch
Zustandsänderungen	kontinuierlich	diskret
Funktionstyp	linear	nicht linear
Störverhalten	stabil	instabil
Zeitabhängigkeit (Stuktur)	starr	flexibel
Grad der Bestimmtheit	deterministisch	stochastisch

Bild 30: Klassifikationssystematik von Systemen nach [69] und [70]

Aus der Sichtweise des Problemlösungsprozesses, d.h. der Überführung von Zielvorstellungen in eine Realisierung als Mittel zur Erreichung der Ziele (Ziel-Mittel-Beziehung), können vier wesensmäßig klar abgrenzbare Systemtypen unterschieden werden [69]. Durch ein Wirksystem wird im Rahmen eines Programmsystems gemäß einem Zielsystem ein Objektsystem verwirklicht (Bild 31).

Zielsysteme, auch Bedarfssysteme genannt, stellen als abstrakte Systeme das angestrebte Ziel, d.h. das Handlungsergebnis als Endzustand, in Form einer Zielhierarchie (aufbaumäßige Zerlegung) sowie u.U. als zeitliche Folge von Teilzielen (ablaufmäßi-

ge Zerlegung) dar. Sie treten in Form von Pflichtenheften, Lastenheften, Anforderungslisten und ähnlichem auf.

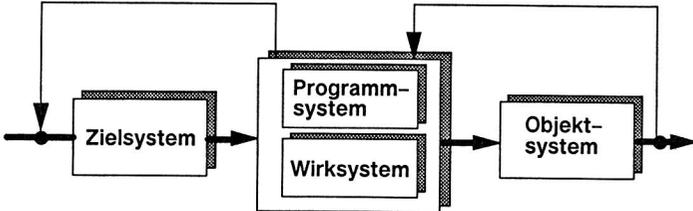


Bild 31: Wichtige Systeme der Systemtechnik

Programmsysteme oder Aufgabensysteme stellen die Aktivitäten als Mittel zur Zielerreichung dar und liefern in Form von Plänen (abstrakte Systeme) Aufschluß über die Art und Weise der durchzuführenden Aktivitäten. Sie treten in Form von Aufgabenplänen oder Vorhabensplänen auf. Ist der Schwerpunkt der Betrachtungen auf den Ablauf gelegt (ablaufmäßige Zerlegung), so spricht man von Prozessen, Ablaufplänen und ähnlichem.

Wirksysteme, auch als Handlungs- oder Arbeitssysteme bezeichnet, sind als konkrete Systeme die Träger der Aktivitäten, haben daher für das Programmsystem unmittelbaren Mittelcharakter. Sie setzen sich in ihrer allgemeinsten Form - als soziotechnische Systeme - aus belebten Komponenten (Menschen, sonstige Lebewesen) und unbelebten Komponenten (Sachmitteln) zusammen, deren Zusammenwirken durch formelle, aber auch informelle Beziehungsmuster (Organisation) ermöglicht wird. Wirksysteme treten als Organisationen, Unternehmungen bzw. als Sachmittelsysteme (Anlagen, Betriebsmittel, Roboter usw.) auf.

Objektsysteme sind als konkrete oder abstrakte Systeme Einwirkungsgegenstand bzw. Handlungsergebnis der von Wirksystemen bei der Realisierung von Programmsystemen mit der Ausrichtung auf Zielsysteme vollzogenen Aktivitäten. Sie treten als beliebige Systeme auf.

Diese Klassifizierung künstlicher Systeme entspricht der Gliederung des Planungsprozesses in

- Zielplanung, als anzustrebender Zustand
- Maßnahmenplanung, als Aktivitätenbündel samt Instrumenten zur Erreichung der Ziele, zerfallend in Ablaufplanung und Mittelplanung
- Ergebnis, als Output des Planungsprozesses (konkretes oder auch abstraktes System).

Diese Gliederung des Planungsprozesses ist allgemein gültig; die aufgeführten Elemente finden sich sowohl bei dem im Abschnitt 3.3 vorgestellten Datenbankentwurf, als auch bei der in Abschnitt 5.1 noch zu behandelnden systematischen Montageplanung.

4.1.3 Allgemeines Modell eines Transformationssystems

In der Theorie Technischer Systeme steht die Frage nach dem Zweck, dem Ziel im Vordergrund: Wofür werden Technische Systeme hergestellt? Stellt man das technische Sachsystem in den Zusammenhang menschlichen Handelns, so gelangt man zum soziotechnologischen Modell [71]. Ausgangspunkt der Modellbildung ist ein zweckentsprechender Transformationsprozeß, der vom Menschen gestaltet wird, um Operanden von einem vorgefundenen Eingangszustand in einen gewünschten Ausgangszustand zu überführen. Bild 32 zeigt das Modell des Transformationssystems.

Operanden sind die passiven Elemente des Systems. Als Operanden kommen Materie, Energie und Information oder Kombinationen daraus in Frage.

Die Änderungen des Zustandes der Operanden werden als Transformationsprozeß bezeichnet. Sie werden verursacht durch Einwirkungen materieller, energetischer oder informationeller Art, die wiederum auf Auswirkungen von Operatoren zurückzuführen sind. Operatoren sind Menschen, Technische Systeme und die

Wirkumgebung. Der Transformationsprozeß ist das verbindende Glied des Transformationssystems.

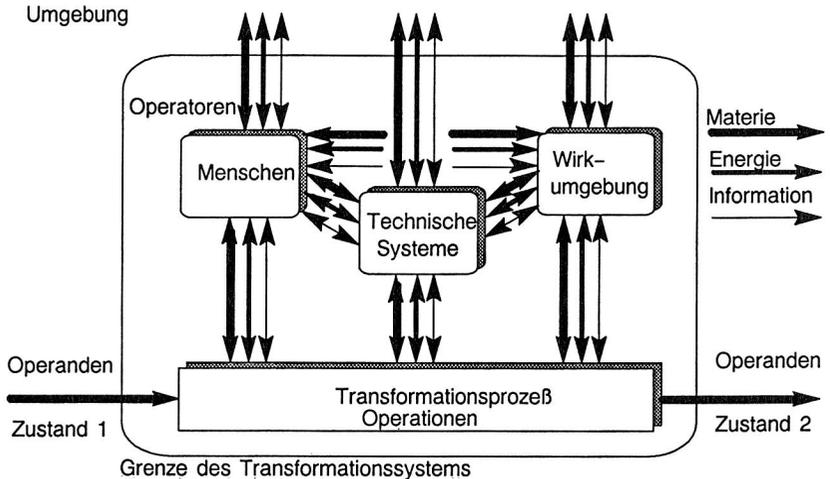


Bild 32: Modell des Transformationssystems (nach [72])

Das Modell deckt generell alle Arten von Transformationen an Operanden aller Art und bei beliebiger Beteiligung der Operatoren an den Auswirkungen ab [72].

Transformationsprozesse werden als Technische Prozesse bezeichnet, wenn als Operatoren Technische Systeme eingesetzt werden. Wenn das "Innere" der Operatoren analysiert wird, werden ganz bestimmte Prozesse festgestellt. Diese internen technischen Prozesse können auch als Transformationen bezeichnet werden. Das Ziel dieser Transformationen ist eine bestimmte Auswirkung auf die Operanden des Transformationssystems, wie im Modell ersichtlich. Wenn die durch die Auswirkungen der Operatoren an diesen Operanden verursachten Transformationsprozesse als primäre Ziele betrachtet werden, so können die nicht unmittelbar mit den Operanden des Transformationssystems in Zusammenhang stehenden Transformationen als sekundäre Transformationsprozesse

bzw. Wirkprozesse oder Wirkketten bezeichnet werden, die andere Operanden zum Gegenstand haben. Welche Prozesse primären und welche Prozesse sekundären Charakter haben hängt somit davon ab, wie die Rollen für Operanden, Operationen und Operatoren verteilt werden.

Zwischen primärem Transformationsprozeß und Wirkprozeß besteht eine kausale Beziehung. Die Zustandsänderungen eines Operanden im Transformationsprozeß werden durch Auswirkungen des Technischen Systems als Ursache hervorgerufen, und diese sind wiederum Folge der Wirkkette im Technischen System ausgehend vom Input des Technischen Systems.

Technische Systeme können wegen ihres offenen Systemcharakters und ihrer daraus folgenden Wechselwirkung mit ihrer Umwelt von zwei Standpunkten aus betrachtet werden. Bei umgebungsorientierter Betrachtungsweise ist die Beschreibung des Systemrandes ausreichend, bei systemorientierter Betrachtung ist die Beschreibung des Systemrandes und des Systeminneren erforderlich [73]. Die Schnittstellen gegenseitigen Wirkungsaustausches zwischen Technischem System und Umgebung bestimmen das Systemverhalten nach außen. Zur Beschreibung des Verhaltens ist, nach der Arbeitsmethode der Systemtechnik, keine Kenntnis der inneren Struktur erforderlich; der innere Systemaufbau hat zunächst keine Bedeutung, allein die Verbindungsgrößen zwischen Umgebung und Systemelementen sind ausreichend [74] (umgebungsorientierte Betrachtungsweise). Je nach Wirkungsrichtung bezeichnet man sie als Ein- oder Ausgangsgrößen, bzw. als Inputs oder Outputs.

Die Funktion eines Technischen Systems ist jener Ausschnitt des Systemverhaltens (Gesamtfunktion), der zur Zweckerfüllung herangezogen wird [69]. Unter Funktion versteht man etwas stabiles, eine gewünschte Wirkungsfähigkeit, die jedoch nicht immer durch das Verhalten des Systems gewährleistet ist; das System kann sich falsch verhalten [72]. Die Funktion des Systems besteht darin, bestimmte Ist-Zustände der Eingangsgrößen in erwünschte Soll-Zustände der Ausgangsgrößen zu transformieren,

die als Auswirkungen auf den Prozeß des Transformationssystems einwirken. Die Funktion wird durch eine umgebungsorientierte Betrachtungsweise beschrieben. Die Transformation wird als Wirkprozeß bezeichnet. Allgemein bedeutet Prozeß einen Vorgang, ein Geschehen, etwas verändert sich im Laufe der Zeit [72]; zur Beschreibung des Wirkprozesses ist eine Betrachtung des Systeminneren erforderlich.

In einem Technischen Prozeß erfahren bestimmte Eigenschaften von Operanden durch eine Reihe von Operationen, die von Operatoren ausgeführt werden, dem Zweck entsprechende Änderungen, wodurch der gewünschte Zustand der Operanden erreicht werden soll. Bei dieser systemorientierten Betrachtungsweise wird nicht mehr von Funktionen gesprochen, sondern von Operationen. Der Funktionsbegriff ist somit Operatoren zugeordnet; er beschreibt, wie Eingangsgrößen in Ausgangsgrößen, genannt Auswirkungen, transformiert werden, während in einem Prozeß Operationen ablaufen, die durch Einwirkungen bewirkt werden, die Auswirkungen von Operatoren sind.

Die Gesamtheit der Werte aller Eigenschaften eines Systems zu einem bestimmten Zeitpunkt nennt man den Zustand des Systems. Bei der Festlegung eines Zustandes abstrahiert man meist von unwesentlichen Eigenschaften [72].

4.2 Die Montage als Transformationssystem

Wissenschaftliche Modelle haben allgemein eine analytische Funktion; das Gegebene soll in zerlegter Form dargestellt werden, um es verstehen und handhaben zu können [7]. Bei Modellen geht es nicht darum, ob sie wahr, sondern ob sie zweckmäßig sind; Modelle bilden zwar Teile der - existierenden oder noch zu schaffenden - Wirklichkeit ab, doch welche Aspekte der Wirklichkeit sie mit welchen Mitteln abbilden, hängt von den jeweiligen Zwecksetzungen der Modellbenutzer ab [75].

Wird das soziotechnologische Modell eines Transformationssystems nach Hubka [72], welches im vorhergehenden Abschnitt vorgestellt wurde, auf die Montage übertragen, so ergibt sich das in Bild 33 dargestellte Modell.

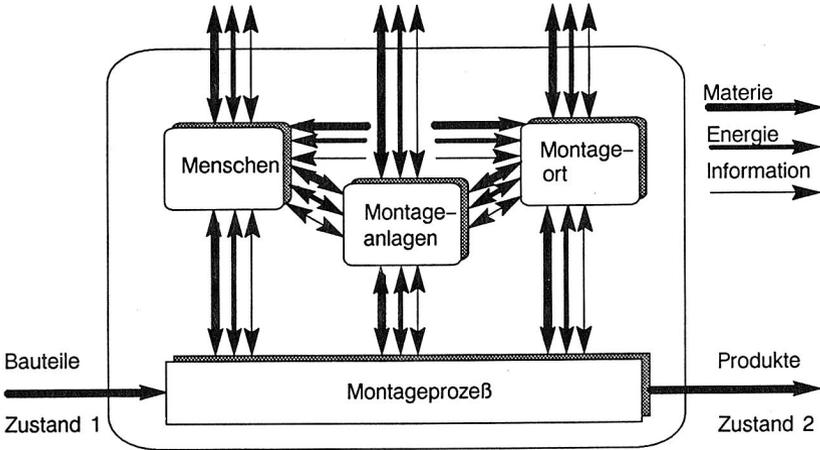


Bild 33: Die Montage als Transformationssystem

Der Montageprozess ist der zielgerichtete Transformationsprozess, in dem aus einzelnen Bauteilen nach einer festgelegten Ordnung Produkte zusammengesetzt werden. Eingangsoperanden sind also Bauteile, die als Produkte das Transformationssystem wieder verlassen. Die Operanden des Transformationssystems Montage sind auch die Operanden des Montageprozesses. Operatoren des Transformationssystems sind Montageanlagen und Menschen, die Montageanlagen bedienen oder Teilmontageprozesse selbst ausführen. Wirkumgebung ist der Montageort.

Die weiteren systemtechnischen Betrachtungen werden anhand dieses Modells durchgeführt. Es wurde gewählt, weil es als besonders zweckmäßig erachtet wird zur Analyse der komplexen Zusammenhänge in der Montage.

4.2.1 Schichten-Architektur-Modell für die Produktion

Die hierarchische Zergliederung von Systemen läßt sich sehr anschaulich in einem Schichten-Architektur-Modell darstellen.

In Anlehnung an ein in [76] und [77] vorgestelltes Schichten-Architektur-Modell eines FFS-Steuerungssystems wird nachfolgend ein derartiges Modell für die Produktion entwickelt, aufbauend auf das oben vorgestellte soziotechnologische Transformationssystem.

Das Transformationssystem wird in hierarchisch geordnete Abstraktionsschichten gegliedert (Bild 34).

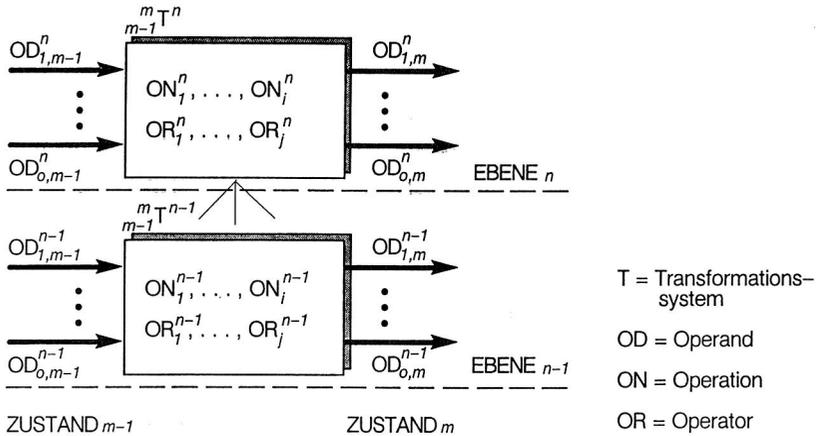


Bild 34: Formale Darstellung des Schichten-Architektur-Modells

Rein formal betrachtet ist das Schichten-Architektur-Modell eine streng geordnete Menge von Abstraktionsschichten, häufig auch Ebenen genannt, wobei jedes Transformationssystem ${}_{m-1}T^n$ jeder Abstraktionsschicht n aus einer Menge von Operationen ON_1^n, \dots, ON_i^n besteht, die durch die Auswirkungen (Funktionen) einer Menge von Operatoren OR_1^n, \dots, OR_j^n bewirkt werden und eine

Menge von Operanden $OD_{l,m-1}^n, \dots, OD_{o,m-1}^n$ des Zustandes $m-1$ in Operanden $OD_{l,m}^n, \dots, OD_{o,m}^n$ des Zustandes m überführen. Der Zustand m eines Operanden $OD_{l,m}^n$ kann durch die Eigenschaften $E_{l,1,m}^{n-1}, \dots, E_{p,1,m}^{n-1}$ beschrieben werden. Zur Realisierung der Funktion eines Operators in der Schicht n werden Operatoren aus der unmittelbar darunterliegenden Schicht $n-1$ verwendet.

Die Abbildung 35 zeigt im Überblick den Aufbau einer Abstraktionshierarchie für das gesamte Produktionssystem eines Betriebes, die in Anlehnung an [78] aufgestellt wurde.

Es werden nur Teilsysteme betrachtet, die Werkstücke als Operanden haben. Welche Operationen in den einzelnen Transformationssystemen ausgeführt und welche Operatoren dazu eingesetzt werden, wird später ausgeführt. Für die Darstellung der hier vorgestellten hierarchischen Gliederung soll es zunächst genügen, die einzelnen Teilsysteme und ihre Ein- und Ausgangsoperanden zu benennen.

Primäres Ziel des Produktionssystems auf der Betriebsebene ist die Herstellung von am Markt absetzbaren Produkten. Die Operanden auf der Ausgangsseite sind aus der Sicht des Produktionsbetriebes Enderzeugnisse, auch wenn es sich z.B. um Halbzeuge, Einzelteile oder Baugruppen handelt. Auf der Eingangsseite werden die Operanden einheitlich als Eingangsmaterialien bezeichnet, egal ob es sich um Rohteile oder Zukaufteile wie Einzelteile und Baugruppen handelt.

Eine Ebene tiefer lassen sich in bezug auf den Operandendurchlauf die drei Betriebsbereiche Materialbeschaffungssystem, Fertigungssystem und Vertriebssystem unterscheiden. Die von Lieferanten bezogenen Eingangsmaterialien durchlaufen das Materialbeschaffungssystem, werden im Fertigungssystem zu Enderzeugnissen veredelt und über das Vertriebssystem an die Kunden ausgeliefert.

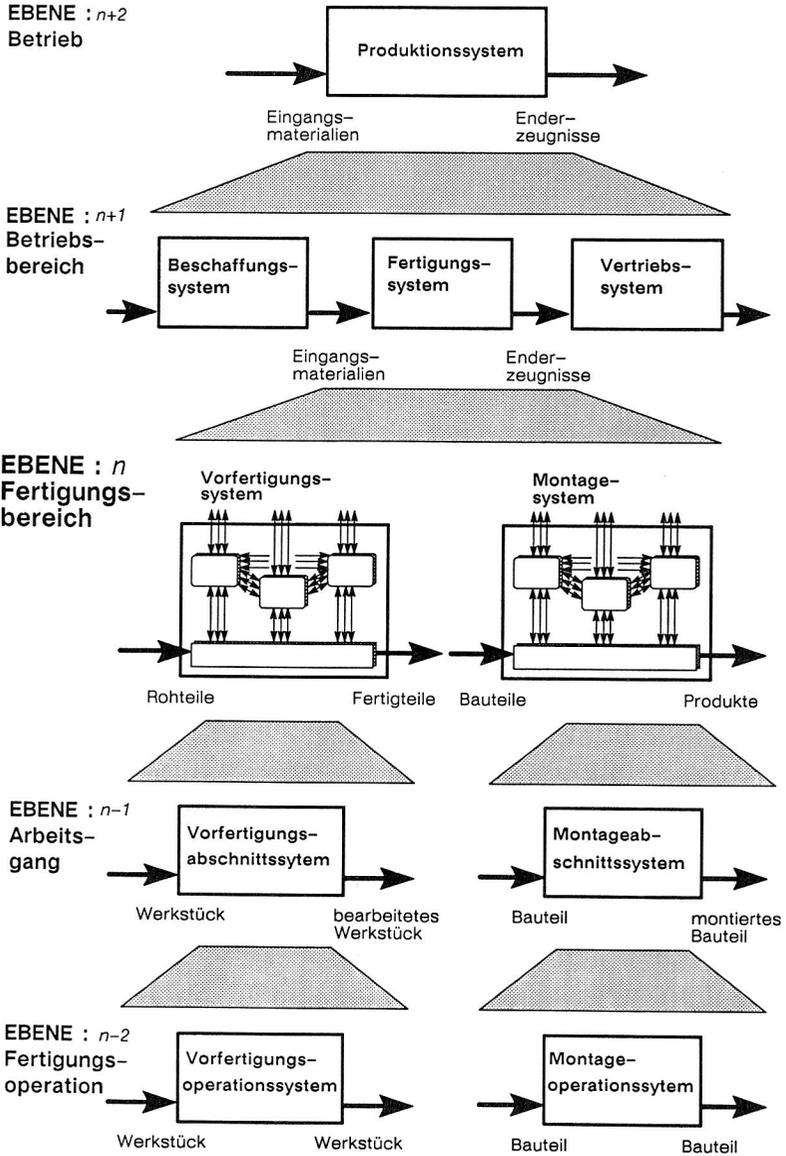


Bild 35: Abstraktionshierarchie eines Produktionssystems

Auf der Ebene der Fertigungsbereiche kann zwischen Vorfertigungssystem und Montagesystem unterschieden werden. Aus Rohteilen werden im Vorfertigungssystem Fertigteile und aus Bauteilen im Montagesystem Produkte erzeugt.

Die Transformationssysteme der Fertigungsbereiche zerfallen auf der Arbeitsgangebene in Vorfertigungsabschnittssysteme, die einzelne Bearbeitungsschritte wie Bohren, Drehen, Fräsen von Werkstücken und in den Montageabschnittssystemen das Montieren einzelner Bauteile zum Inhalt haben.

Die Arbeitsgänge bestehen aus einer Reihe von Fertigungsoperationen, die von Vorfertigungsoperationssystemen und Montageoperationssystemen ausgeführt werden.

Diese Abstraktionshierarchie kann nach oben und unten beliebig erweitert werden. Im Rahmen dieser Arbeit wird der Fertigungsbereich Montage auf den Ebenen n bis $n-2$ betrachtet.

4.2.2 Bauteile als Operanden des Montagesystems

Die Bauteile als Operanden des Transformationssystems Montage und damit auch des Montageprozesses lassen sich je nach ihrem Montagezustand vier verschiedenen Komplexitätsgraden zuordnen (Bild 36).

Einzelteile sind elementare Teile, die nicht zerstörungsfrei weiter zerlegt werden können. Normalerweise stellen sie für den Montagebereich den niedrigsten Komplexitätsgrad dar. Werden mehrere Einzelteile zusammengesetzt, so entsteht eine Baugruppe. Mehrere Baugruppen bilden Geräte, bzw. Maschinen oder Apparate, die sich wiederum zu Anlagen zusammenfassen lassen, wobei in jedes Bauteil eines Komplexitätsgrades höherer Ordnung Bauteile aller niedrigeren Komplexitätsgrade eingehen können.

Da in der Literatur Montageobjekte mit sehr unterschiedlichen Begriffen bezeichnet werden, wird im folgenden eine Festlegung getroffen, welche Begriffe im Rahmen dieser Arbeit im Zusammenhang mit dem Montageprozeß benutzt werden (Bild 37).

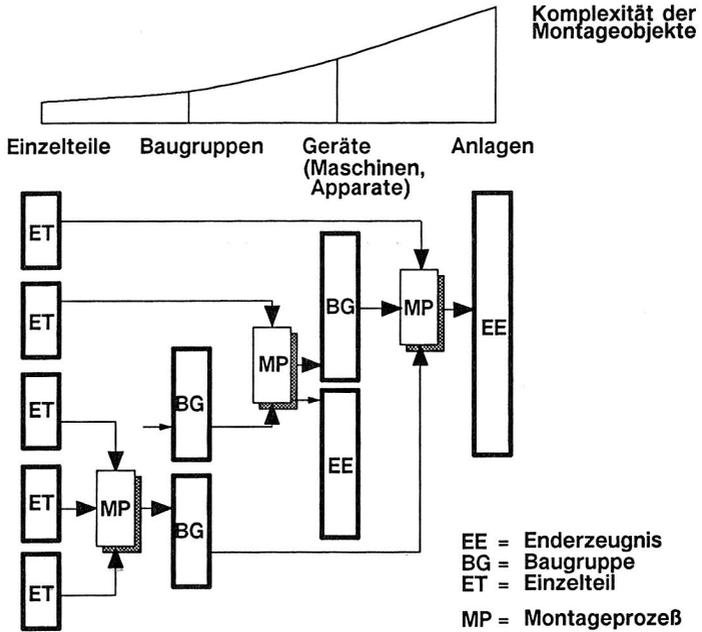


Bild 36: Operanden des Montageprozesses nach Komplexitätsgraden



Bild 37: Bezeichnungen für Operanden des Montageprozesses

Eingangsoperanden des Montageprozesses, die der bestimmungsgemäßen Transformation unterzogen werden, sind Bauteile, die vom Komplexitätsgrad Einzelteil oder Baugruppe sein können. Bestimmungsgemäße Ausgangsoperanden sind Produkte, die entweder als Baugruppen bezeichnet werden, wenn sie Eingangsoperanden eines nachfolgenden Montageprozesses werden, oder als Enderzeugnisse, wenn sie keinen weiteren Montageoperationen mehr unterzogen werden.

Während des Montageprozesses werden die Operanden als Werkstücke bezeichnet, wobei ein Werkstück Basisteil, Fügeteil oder Hilfsteil sein kann. Basisteil ist das Bauteil, welchem weitere Bauteile in der Folge des Montagefortschritts hinzugefügt werden. Das Basisteil wird also mit fortschreitendem Montageprozeß erweitert. Fügeteile sind Bauteile, die dem Basisteil hinzugefügt werden. Hilfsteile sind Werkstücke, die nicht in das Produkt eingehen, also beispielsweise Werkstückträger oder Spannvorrichtungen. Hilfsteile können auch als Nebenoperanden betrachtet werden.

Die für die Durchführung der einzelnen Montageoperationen des Montageprozesses relevanten Zustandsgrößen, die Eigenschaften der Operanden, lassen sich, in Anlehnung an eine Klassifizierung der Werkstückparameter nach [80], in drei Gruppen unterteilen (Bild 38).

Die erste Gruppe beinhaltet dem Werkstück eingeprägte Eigenschaften, die fest vorgegeben sind und dem Werkstück direkt zugeordnet werden können ohne Berücksichtigung des Prozeßablaufs. Sie können aufgeteilt werden in geometrische, physikalische und technologische Werkstückdaten. Die zweite Gruppe beschreibt das Werkstückverhalten, das sowohl vom Werkstück als auch vom Prozeßablauf abhängt. Dabei kann unterschieden werden zwischen dem Werkstückverhalten des einzelnen Werkstückes und des Werkstückverbandes. Die Werkstückeigenschaften und die Werkstückverhaltensmerkmale sind nicht unabhängig von einander. So läßt z.B. die Hauptgeometrie eines Werkstückes

teilweise auf das Bewegungsverhalten schließen; deshalb wird diese Eigenschaft auch manchmal mit Verhaltenstyp bezeichnet.

Werkstückeigenschaften (durch Werkstück bestimmt)			Werkstückverhalten (durch Werkstück und Prozeßablauf bestimmt)		Prozeßgrößen (durch Prozeßablauf bestimmt)	
geometrisch	physikalisch	technologisch	im Verband	einzel	geometrisch	physikalisch
Haupt- geometrie (Verhaltens- typ)	Masse (Gewicht)	kennzeich- nende Formelemente	Haufwerk- verhalten	Stand- sicherheit	Position	Temperatur
Hauptab- messungen	Werkstoff	Oberflächen- empfindlichkeit	Bewegungs- verhalten	stabile Orientierung	Orientierung	Druck
Toleranzen	Schwer- punktlage	Unverträg- lichkeiten (Feuchte, Staub)	Stapel- fähigkeit	Vorzugs- orientierung	Ordnungs- zustand	
Symmetrien	Steifigkeit			Hänge- fähigkeit		
Größenklasse	Festigkeit	Oberflächen- eigenschaften (Reibung, Grat)		Gleitfähigkeit		
				Rollfähigkeit		
				Richtungs- stabilität		

Bild 38: Relevante Zustandsgrößen eines Werkstückes im Montageprozeß

Die dritte Gruppe beschreibt Eigenschaften, die das Werkstück während des Prozesses annimmt, die jedoch erst durch den Prozeßablauf bestimmt werden. So ist der Gegenstand der Veränderungen bei der Montage in erster Linie die Lage der Bauteile, beschrieben durch die Position und Orientierung, während es bei der Vorfertigung die Veränderung der Geometrie ist [79]. Die Werkstücklage kann exakt und klassifizierend beschrieben werden.

Zur klassifizierenden Beschreibung der Werkstücklage eignet sich die Ordnungszustandsmatrix nach Bild 39 [80], mit der die Werkstückfreiheitsgrade beschrieben werden.

Das Element 0.0 sagt aus, daß sich die Werkstücke im ungeord-

neten gespeicherten Zustand befinden, d.h. weder Position noch Orientierung der Werkstücke sind festgelegt. Das Element 3.3 sagt aus, daß die Werkstücke in allen Freiheitsgraden festgelegt sind, also geordnet vorliegen. Durch Symmetrien kann sich der maximale Ordnungsgrad in bezug auf die Werkstückorientierung verringern.

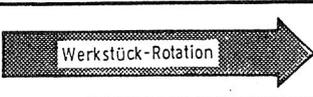
					
	Og 0.0	Og 0.1	Og 0.2	Og 0.3	Haufwerk
	Og 1.0	Og 1.1	Og 1.2	Og 1.3	flächige Verteilung
	Og 2.0	Og 2.1	Og 2.2	Og 2.3	linienförmige Verteilung
	Og 3.0	Og 3.1	Og 3.2	Og 3.3	vereinzelt
	3 Drehfreiheitsgrade unbestimmt	1 Drehfreiheitsgrad bestimmt	2 Drehfreiheitsgrade bestimmt	3 Drehfreiheitsgrade bestimmt	Zustand des Werkstückverbands
				Werkstückorientierung	

Bild 39: Ordnungszustandsmatrix [80]

4.2.3 Der Montageprozeß

Der Montageprozeß als Technischer Prozeß ist ein wesentliches Element des Transformationssystems Montage nach Bild 33. Durch ihn wird die komplette Transformation der Werkstücke als Operanden OD_1^n, \dots, OD_o^n vom Eingangszustand bis zum Ausgangszustand beschrieben, ohne Berücksichtigung des "womit", "wer", "wann" und "wo".

Der "Hauptfluß" im Montageprozeß betrifft die Werkstücke, deren Zustände verändert werden.

Der Montageprozeß auf Fertigungsbereichsebene umfaßt neben dem Zusammenbau der Produkte Operationen des Lagerns und des

Förderns oder Transportierens (Bild 40). Fördern oder Transportieren bezieht sich auf die Ortsveränderung des Fördergutes im Werksbereich [81]. Die zum Zusammenbau benötigten Baugruppen und Einzelteile werden in der Regel in montagebezogenen Lagern oder direkt an Montagestationen zwischengelagert. Ein Transport der Bauteile bzw. Produkte findet zu den Montagestationen, zwischen den Montagestationen selbst und von den Montageanlagen zum Fertigteillager statt, wo die Erzeugnisse vom Vertrieb übernommen werden. Für Lagern und Fördern sind im allgemeinen nur Positionsbedingungen (z.B. Lagerort, Anfangs- und Endpunkt einer Förderbewegung) vorgegeben.

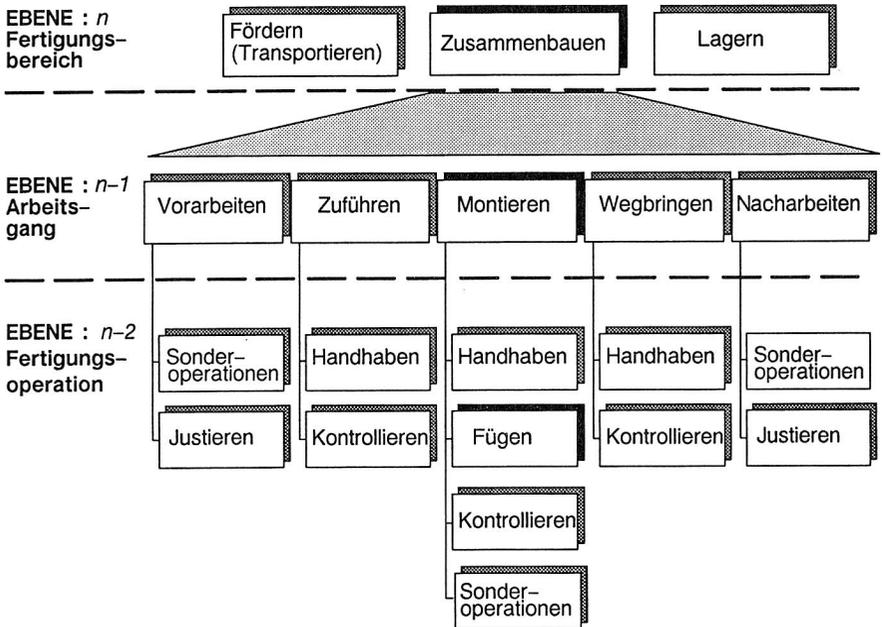


Bild 40: Montageoperationen der Ebenen n , $n-1$ und $n-2$

Die Kernoperation des Zusammenbauens ist auf der Arbeitsebene das Montieren (Bild 40). Während das Zusammenbauen der Fertigungsebene eine Folge von Montageabschnitten um-

faßt, die zum kompletten Produkt (Baugruppe oder Enderzeugnis) führen, beinhaltet das Montieren nur das Hinzufügen eines bzw. mehrerer Fügeteile zu einem Basisteil im Sinne eines Arbeitsganges. Häufig muß dem Montieren noch ein zusätzlicher Bearbeitungsvorgang vorgeschaltet werden wie z.B. Reinigen, Entgraten, Kühlen, Erwärmen, Schmieren. Operationen dieser Art können als Vorarbeiten bezeichnet werden. Eine zweite Art von Operationen dient der Zuführung des zu montierenden Bauteils. Bevor ein Bauteil montiert werden kann, muß es in geeigneter Weise zugeführt werden. Nachdem das Montieren beendet wurde, muß das erweiterte Basisteil bzw. das fertige Produkt aus dem Montagebereich weggebracht werden. Manchmal sind auch noch Nacharbeiten wie Markieren, Entölen, Entstauben, Beseitigen von Fehlern usw. erforderlich.

Auf der Ebene der Fertigungsoperationen $n-2$ können die Montageoperationen in die fünf Klassen Fügen, Handhaben, Kontrollieren, Sonderfunktionen und Justieren eingeteilt werden (Bild 40).

Das Fügen ist der Anteil am Montieren, der den eigentlichen, den aktiven Fertigungsfortschritt am Produkt ausmacht [81]. Fügen beschränkt sich lediglich auf das Zusammenbringen von zwei oder mehr Werkstücken geometrisch bestimmter fester Form bzw. von ebensolchen Werkstücken mit formlosem Stoff, wobei örtlicher Zusammenhalt geschaffen und im ganzen vermehrt wird. Handhaben ist das Schaffen, definierte Verändern oder vorübergehende Aufrechterhalten einer vorgegebenen räumlichen Anordnung von geometrisch bestimmten Körpern in einem Bezugskordinatensystem [82]. Im Unterschied zum Lagern und Fördern ist beim Handhaben zusätzlich immer die Orientierung der zu handhabenden Körper vorgegeben. Das Kontrollieren dient dem Feststellen, ob bestimmte Eigenschaften oder Zustände erfüllt sind. Sonderoperationen sind Hilfsoperationen, die vor oder während des Fügevorganges erforderlich sind. Justieren umfaßt geplante Anpaßoperationen zum Ausgleich fertigungstechnisch bedingter Abweichungen mit dem Ziel, geforderte Funktionen, Funktionsgenauigkeiten oder Eigenschaften von Erzeugnissen innerhalb vorgegebener

Grenzen zu erreichen.

Die Operationen der Arbeitsgangsebene werden jeweils durch eine Reihe von Montageoperationen der Ebene der Fertigungsoperationen verwirklicht (Bild 40). So setzt sich Montieren zusammen aus Füge-, Handhabungs-, Kontroll- und Sonderoperationen. Zuführen und Wegbringen beinhalten Handhabungs- und Kontrolloperationen, Vor- und Nacharbeiten umfassen Sonder- und Justieroperationen.

In Bild 41 sind Beispiele für die den fünf Klassen der Ebene $n-2$ zugeordneten Montageoperationen aufgelistet.

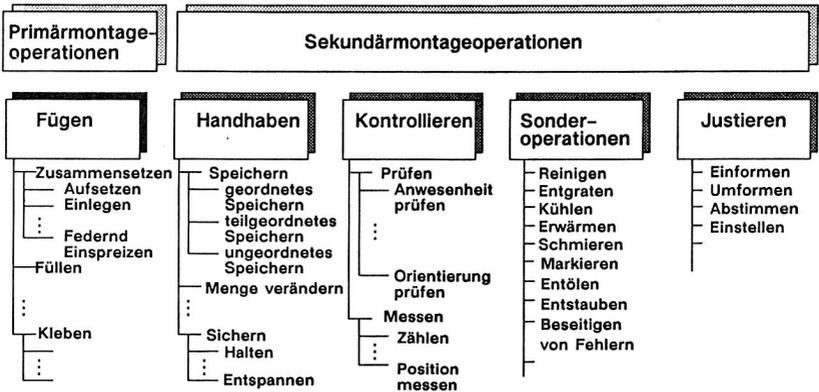


Bild 41: Montageoperationen der Ebene $n-2$

Mit dem Begriff Fügen werden Fertigungsverfahren benannt, die dazu dienen, Werkstücke miteinander zu verbinden. Die DIN 8593 [83] ordnet das Fügen in die Fertigungsverfahren ein und nimmt eine Unterteilung der Fertigungsverfahren Fügen in Gruppen nach dem Ordnungsgesichtspunkt "Art des Zusammenhalts unter Berücksichtigung der Art der Erzeugung" vor.

Die Handhabungsoperationen können entsprechend den in der VDI-

Richtlinie 2860 [82] festgelegten Handhabungsfunktionen gegliedert werden. In der VDI-Richtlinie 2860 werden auch die Kontrolloperationen untergliedert. Eine Einteilung von Justieroperationen findet sich in [81].

An dieser Stelle muß zusätzlich zu den Begriffen "Operation" und "Funktion" der Begriff "Verfahren" eingeführt werden. In der Literatur werden die Begriffe nicht immer eindeutig unterschieden. Ein Verfahren bezeichnet die Art und Weise der Durch- bzw. Ausführung von etwas, während eine Funktion die Verrichtung, Aufgabe, Obliegenheit von Teilen im Rahmen eines Ganzen anspricht. Operationen beschreiben die Zustandsänderungen von Operanden während eines Prozesses. Aus der Sicht der Operatoren, die diese Zustandsänderungen bewirken, werden diese Operationen zu Funktionen. In der VDI-Richtlinie 2860 wird von Handhabungsfunktionen gesprochen, weil die Handhabungseinrichtungen, die diese Funktionen ausführen können, im Vordergrund stehen. Im Rahmen dieser Arbeit werden gemäß dem Transformationsmodell zunächst Handhabungsoperationen definiert. Eine Montageaufgabe wird durch eine Folge von Montageoperationen beschrieben, wobei zur Benennung der einzelnen Montageoperationen sowohl die Gliederung nach DIN 8593 als auch nach VDI 2860 herangezogen wird. Hierbei tritt der entscheidende Unterschied zwischen Funktion und Verfahren zu Tage. Während nämlich zur Beschreibung der Operationsfolge eines Montagevorganges eine Reihe von Handhabungsoperationen benötigt werden, tritt in einer derartigen Folge nur eine einzige Fügeoperation auf, beschrieben durch ein Fügeverfahren. Der Begriff Funktion wird erst benötigt, wenn ein Operator für eine Operation gesucht wird. Die Auswirkung des Operators zur Verwirklichung der Operation wird dann Zweckfunktion des Operators genannt.

Oftmals wird bei der Analyse von Montageabläufen zwischen Primärmontagevorgängen und Sekundärmontagevorgängen unterschieden [84]. Unter Primärmontagevorgängen sind alle Vorgänge zu verstehen, die der Vervollständigung und der direkten Erhöhung der Wertschöpfung eines Produktes während seines Montageorgan-

ges dienen. Dagegen werden Vorgänge, die nicht unmittelbar der Vervollständigung dienen, als Sekundärmontagevorgänge bezeichnet. In diesem Sinne kann das Fügen als primäre Montageoperation, und Handhaben, Kontrollieren, Sonderoperationen sowie Justieren können als sekundäre Montageoperationen bezeichnet werden (siehe Bild 41).

4.2.4 Die Operatoren des Transformationssystems Montage

Um die gewünschten Änderungen der Eigenschaften in einem Technischen Prozeß zu erreichen, muß man in geeigneter Weise auf den oder die Operanden einwirken. Man muß befördern, reinigen, drehen, einlegen usw. Die gesamte Wirkung (W) zur Erzielung beispielsweise einer Transformation des Operanden ($OD_{1,1}^{n-1} \rightarrow OD_{1,m}^{n1}$) besteht aus Teilwirkungen (TW), die gewisse Eigenschaften verändern z.B. ($E_{p,1,1}^{n-1} \rightarrow E_{p,1,m}^{n-1}$). Wirkungen können immer bidirektional betrachtet werden. Sind sie Ausgangsgrößen eines Systems, werden sie als Auswirkungen bezeichnet. Aus der Sicht eines anderen Systems, für das die gleichen Wirkungen Eingangsgrößen sind, werden sie Einwirkungen genannt.

Eine wichtige Frage für die Transformation ist, womit die nötigen Wirkungen im Technischen Prozeß erreicht werden. So kann ein Arbeiter ein Bauteil aus einer Kiste entnehmen, zu einer Fügestelle bringen und dort mit Hilfe einer Vorrichtung in ein Basisteil einpressen. Oder er kippt die Kiste in den Bunker eines Vibrationswendelförderers der die Bauteile automatisch vereinzelt und positioniert. Ein Handhabungsgerät übernimmt ein Bauteil und setzt es in die Vorrichtung ein, die es in das Basisteil einpreßt.

Im ersten Fall übernimmt der Mensch fast alle Wirkungen, die zum Montieren eines Bauteils erforderlich sind. Im Falle der automatischen Montage beschränkt sich der Mensch auf die Bereitstellung der Bauteile, das Ingangsetzen, Überwachen und Warten der Maschinen.

Das Beispiel zeigt auch die enge Abhängigkeit von Operationen und Operatoren. Wird der Mensch als Operator gewählt, so können einzelne Operationen entfallen wie z.B. das Positionieren zwischen dem Vereinzeln und dem Einsetzen in die Vorrichtung. Das bedeutet, die erforderlichen Montageoperationen auf der Ebene $n-2$ sind abhängig von der Wahl der Operatoren, die die benötigten Wirkungen verursachen, um vom Eingangszustand zum geforderten Ausgangszustand der Operanden des Montageprozesses zu kommen.

Die Gesamtheit der nötigen Wirkungen im Montageprozeß wird durch die Beziehung Mensch - Technisches System verwirklicht, mit unterschiedlichen Anteilen der beiden Komponenten.

Im Mittelpunkt der hier angestellten Betrachtungen stehen die Technischen Systeme als Operatoren im Transformationssystem Montage. Sie können entsprechend ihren Wirkungen in drei Klassen eingeteilt werden (Bild 42).

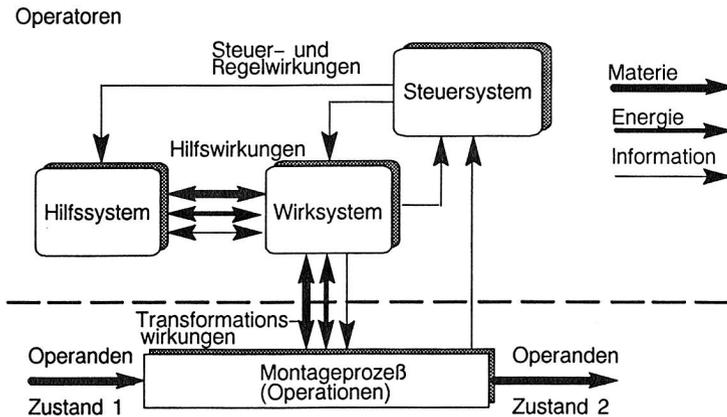


Bild 42: Einteilung Technischer Systeme nach ihren Wirkungen

Wirksysteme sind die Systeme, deren Auswirkungen direkt auf die Operanden des Montageprozesses einwirken um einzelne Montageoperationen zu vollziehen. Diese Wirkungen werden als Arbeits- oder Transformationswirkungen bezeichnet [72]. Sie sind jedoch für die Verwirklichung Technischer Prozesse nicht ausreichend. Um den ordnungsgemäßen Ablauf einer Operationsfolge zu gewährleisten, sind Steuer- und Regelwirkungen erforderlich, die von Steuersystemen erbracht werden. Diese Systeme koordinieren die Wirkungen von Teilsystemen der Wirksysteme und der dritten Klasse, der Hilfsysteme. Die Hilfssysteme, wie beispielsweise Greiferwechselsysteme, sind nicht direkt an der Prozeßdurchführung beteiligt, ermöglichen jedoch teilweise erst die eigentlichen Transformationswirkungen. Diese drei Systemtypen lassen sich grundsätzlich unterscheiden, wenn Technische Systeme in Teilsysteme aufgegliedert werden.

Die Wirksysteme können noch unterteilt werden in Systeme, die direkt auf die Operanden einwirken und solche, die dies nur indirekt über andere Wirksysteme tun. So wirkt ein Handhabungsgerät nur indirekt über den Greifer auf das Werkstück ein.

Die planmäßige und zielgerichtete Auswirkung eines Technischen Systems wird als dessen Funktion bezeichnet. Häufig wird jedoch auch die Fähigkeit des Technischen Systems, seine Eingangsgrößen in relevante Ausgangsgrößen zu überführen als Funktion definiert (Siehe auch Abschnitt 4.1.3). Zur Unterscheidung wird die erste "Zweckfunktion" und die zweite "technische Funktion" genannt. Die beiden Funktionsbegriffe stehen zueinander in einer Beziehung "Ziel - Mittel". Während die Zweckfunktion beschreibt, was das Technische System im Rahmen eines Transformationssystems für eine Aufgabe hat, beantwortet die technische Funktion die Frage, wie die Zweckfunktion erfüllt wird. Dabei kann eine Zweckfunktion durch unterschiedliche technische Funktionen erzeugt werden und umgekehrt.

Technische Systeme nutzen physikalische Effekte um ihre technische Funktion zu erfüllen. Die Bestandteile Technischer Systeme (Teilsysteme) sind so zusammengefügt, daß durch ihren Wirkungszusammenhang ein gewünschtes Geschehen, der interne technische

Prozeß (Wirkprozeß), erzwungen wird, wodurch die geforderte äußere Auswirkung, die Zweckfunktion, erreicht wird. Dieser Wirkzusammenhang kann durch interne technische Funktionen beschrieben und in einer Funktionsstruktur dargestellt werden, oder es werden die Funktionsträger (=Organe), welche die Funktionen ausführen, eingesetzt und die gleiche Wirkweise wird mit der Organstruktur des Technischen Systems beschrieben. Organe stellen prinzipielle Beschreibungen von Funktionsträgern dar. Beide Beschreibungsformen eines Technischen Systems beruhen auf einer rein funktionellen Betrachtungsweise, wobei es möglich ist, einzelne Funktionen durch eine Menge unterschiedlicher Organe auszuüben. Das ausgewählte Organ entscheidet dann über zu erfüllende Funktionen auf der nächstniedrigeren hierarchischen Stufe [72]. Wird die konkrete Gestalt eines Funktionsträgers betrachtet, wobei wiederum jedes prinzipielle Organ unterschiedlich realisiert werden kann, so gelangt man zur Bauform. Ein Technisches System kann also beschrieben werden auf den drei Abstraktionsebenen

- Funktionsstruktur,
- Organstruktur und
- Baustruktur.

Die drei Strukturen beschreiben dasselbe Objekt (Technische System) auf sehr unterschiedlichen Abstraktionsstufen.

Die Erkenntnis, daß ein technisches Gebilde auf verschiedenen Abstraktionsstufen beschrieben werden kann, hat entscheidenden Einfluß auf moderne Konstruktionssystematiken (z.B. [85,86]) ausgeübt.

Die Konstruktionssystematik verfolgt das Ziel, die vorwiegend intuitive Arbeitsweise des Konstrukteurs durch eine systematisch-methodische Vorgehensweise zu ersetzen.

Schon vor mehr als hundert Jahren hat Reuleaux eine Getriebe-Konstruktionssystematik entworfen [87] aus der Franke [88] eine allgemeingültige Konstruktionslehre zu entwickeln versuchte. Diese Getriebe- und Konstruktionslehren beziehen sich

inhaltlich vor allem auf Maschinen zum Energieumsatz. Claussen hat die wichtigsten Begriffe dieser beiden Systematiken zusammengestellt und an Beispielen erläutert [89]. Er kommt zu dem Schluß, daß diese Lehren noch heute angewendet werden können. Vor allem kann man aus den beiden Methoden ableiten, wie eine moderne Konstruktionslehre formal aussehen muß: Der Konstruktionsvorgang muß aufgelöst werden in eine Folge von einzelnen Arbeitsschritten, die iterativ durchlaufen werden.

Der Konstruktionsablauf wird durch eine Reihe unterschiedlicher Parameter beeinflusst, von denen die wichtigsten die verschiedenen Konstruktionsphasen, die Konstruktionsart und die Komplexität der Funktionsträger sind [90]. In Bild 43 sind diese konstruktionsbestimmenden Einflußgrößen zusammengestellt.

Konstruktionsphase	Konstruktionsart	Komplexität der Funktionsträger
Funktionsfindung	Neukonstruktion	Anlagen
Prinziparbeit	Anpassungskonstruktion	Einzelmaschinen
Gestaltung	Variantenkonstruktion	Funktionshauptgruppen
Detaillierung	Prinzipkonstruktion	Funktionsgruppen
		Funktionskomplexe
		Funktions-elemente
		Elementare Funktionsträger

Bild 43: Die wichtigsten konstruktionsbestimmenden Parameter (nach [90])

Die verschiedenen Konstruktionsphasen beinhalten sehr unterschiedliche Tätigkeiten. In der Phase der Funktionsfindung

werden in einer Analyse der Aufgabenstellung die zur Lösung erforderlichen Funktionen herausgearbeitet und in einer Funktionsstruktur dargestellt. In der sich anschließenden Phase der Prinzipierarbeitung werden für die erforderlichen Funktionen geeignete Funktionsträger gesucht und zu einer Organstruktur zusammengestellt, die ein optimales Arbeitsprinzip gewährleistet. Diese beiden Phasen sind eng miteinander verknüpft und werden deshalb häufig zusammenfassend als Konzeptphase bezeichnet. Die enge Verknüpfung ergibt sich aus der Vorgehensweise bei der Funktionsfindung, in der eine durch die Aufgabenstellung gegebene Gesamtfunktion systemtechnisch soweit aufgegliedert wird (z.B. durch das sogenannte "Black-Box-Modell"), bis sich für jede Teilfunktion geeignete Funktionsträger finden lassen. In der Gestaltungsphase werden die Bauteile für das optimale Arbeitsprinzip dimensioniert und gestaltet. Es wird eine Geometrie bestimmt, die Hauptabmessungen werden mit Hilfe von Auslegungsrechnungen ermittelt, und Werkstoffe, Oberflächenbeschaffenheit usw. werden festgelegt. Das Ergebnis der Gestaltungsphase ist der fertige Entwurf, die Baustruktur der Konstruktion. In der letzten, der Detaillierungsphase, wird dieser Entwurf ausgearbeitet. Es werden die Fertigungsunterlagen wie Einzelteilzeichnungen und Stücklisten erstellt. Da es für eine Gesamtfunktion viele Funktionsstrukturen, genannt Funktionsprinzipien und für jedes Funktionsprinzip viele Organstrukturen (Arbeitsprinzipien) und dafür wiederum viele Baustrukturen gibt, schließt sich an jede Konstruktionsphase eine Bewertung an, durch die aus den vielen Lösungsmöglichkeiten eine möglichst optimale herausgesucht wird.

Die vier in Bild 43 unterschiedenen Konstruktionsarten sind dadurch gekennzeichnet, daß bei ihrer Bearbeitung verschiedene Konstruktionsphasen durchlaufen werden.

Bei der Neukonstruktion sind dem Bearbeiter so gut wie keine Lösungsvorschläge für die Aufgabenstellung bekannt. Sie durchläuft alle Konstruktionsphasen. Bei der Anpassungskonstruktion entfällt die Phase der Funktionsfindung, da bereits ein Funktionsnetz vorliegt. Es handelt sich um Weiterentwicklungen oder

Änderungskonstruktionen. Werden nur noch Gestalt und Dimension einer Konstruktion geändert, so spricht man von Variantenkonstruktion. Bei der Prinzipkonstruktion liegt das Funktionsprinzip, das Arbeitsprinzip und die Gestalt fest. Es werden nur noch die Abmessungen geändert, d.h. die Fertigungsunterlagen überarbeitet.

Der dritte wesentliche Parameter bei Konstruktionsaufgaben ist die Komplexität. Hierunter versteht man im Konstruktionsbereich den Grad der sinnvollen Zerlegbarkeit in Einheiten (Funktionsträger) eines Erzeugnisses, die aus einer funktionalen Gliederung hervorgehen. Die Funktionsträger auf den verschiedenen Komplexitätsebenen sind durch folgende Eigenschaften gekennzeichnet [90]:

- Elementare Funktionsträger sind technisch herstellbare Flächen mit bestimmter Oberflächenbeschaffenheit und einem bestimmten Werkstoff. Sie haben i.a. keine räumliche Begrenzung, auf sie können die Regeln der Gestaltungs- und Bemessungslehre nicht angewendet werden. (Beispiele: Rotationsflächen, Planflächen usw.)
- Funktionselemente dienen zur Erfüllung eines physikalischen Effektes und entstehen durch eindeutige Zuordnung von zwei oder mehreren Elementaren Funktionsträgern. Sie haben eine räumlich definierte Abmessung, und auf sie sind erstmals die allgemeinen Regeln der Bemessungs- und Gestaltungslehre anwendbar. Ein physikalischer Effekt beinhaltet dabei die gesetzmäßige Umwandlung oder Veränderung von physikalischen Größen [85] (Beispiel: Abstreifschragen von Ordnungseinrichtungen).
- Funktionskomplexe sind funktionale Einheiten, die sich aus mindestens zwei oder mehreren Funktionselementen zusammensetzen und mindestens ein Einzelteil umfassen. Sie dürfen noch keine Baugruppe darstellen und müssen mehr als nur einen physikalischen Effekt realisieren (Beispiel: Ordnungseinrichtungen von Vibrationswendelförderern).
- Funktionsgruppen (Baugruppen) setzen sich aus mindestens

zwei oder mehreren Funktionskomplexen zusammen. Sie sind in der Lage, definierte Eingangsgrößen zu verarbeiten, weiterzuleiten oder in gewünschte Ausgangsgrößen umzuwandeln (Beispiel: Sortiertöpfe für Vibrationswendelförderer).

- Funktionshauptgruppen setzen sich aus mindestens zwei oder mehreren Funktionsgruppen und weiteren Funktionsträgern niederer Komplexität zusammen und erfüllen eine Vielzahl von Elementarfunktionen. Sie umfassen Hauptbaugruppen mit ihren Anschlußelementen zu angrenzenden Gruppen (Beispiel: Vibrationswendelförderer).
- Einzelmaschinen sind technische Gebilde, die sich aus einer sinnvollen Kombination von Funktionshauptgruppen, Funktionsgruppen sowie weiteren Funktionsträgern niederer Komplexität ergeben (Beispiel: Montagezellen).
- Anlagen setzen sich zusammen aus einer oder mehreren Einzelmaschinen sowie Funktionsträgern niederer Komplexität (Beispiel: Montageanlagen).

Der Konkretisierungsgrad einer Konstruktion ist sowohl vom vorliegenden Komplexitätsgrad als auch von der jeweils durchlaufenen Konstruktionsphase abhängig (Bild 44).

Es kann zwischen der Tätigkeit der "Erfindenden Konstruierens" und der des "Kombinierenden Konstruierens" unterschieden werden. "Erfindendes Konstruieren" ist dadurch gekennzeichnet, daß auf keiner Komplexitätsebene ein Funktionsträger für die Lösungsfindung bekannt ist. "Kombinierendes Konstruieren" ist das systematische Zusammenfügen bekannter Funktionseinheiten niederer Komplexität zu einer Funktionseinheit höherer Komplexität.

Bei der Konstruktion von Montagesystemen handelt es sich zumeist um Neukonstruktionen die alle Konstruktionsphasen durchlaufen; es herrscht jedoch die Tätigkeit des "Kombinierenden Konstruierens" vor. Die Analyse des Montageprozesses führt zu einer Reihe von erforderlichen Montageoperationen, für die geeignete Operatoren gesucht werden müssen. Aus der Sicht der Operatoren werden diese Montageoperationen zu Zweckfunktionen. Zur Erfüllung dieser Zweckfunktionen werden in der Regel Funktionsträger

höherer Ordnung wie Handhabungsgeräte, Greifer, Ordnungseinrichtungen usw. eingesetzt, die in geeigneter Weise zu einer Montageanlage zu kombinieren sind. Teilweise genügt für die Lösung derartiger Konstruktionsaufgaben die Betrachtung der "Zweckfunktion" und es muß nicht die "technische Funktion" genauer untersucht werden.

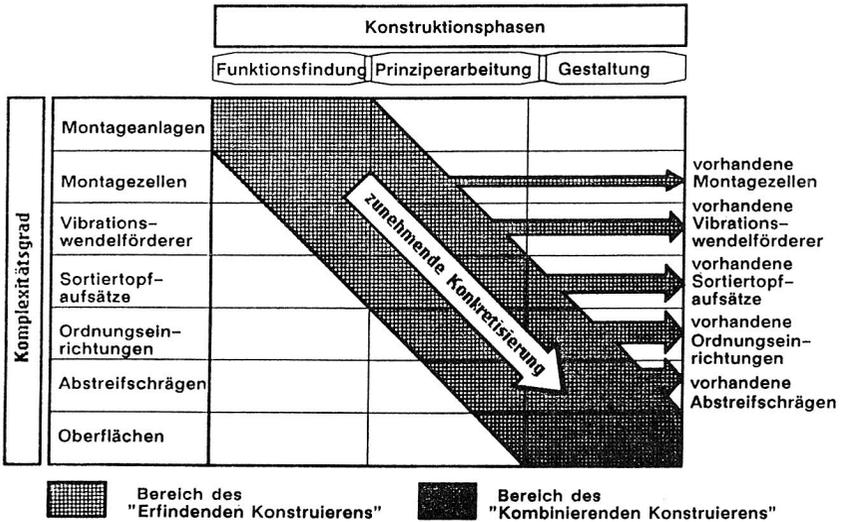


Bild 44: Konkretisierungsgrad einer Konstruktionsaufgabe (nach [90])

Dies bedeutet, daß der Wirkprozeß, also die innere Struktur der Technischen Teilsysteme bei der Konstruktion von Montageanlagen von untergeordneter Bedeutung ist. Er spielt nur eine Rolle bei speziellen Vorrichtungen und besonderen Anpassungen von Komponenten standardisierter Montageeinrichtungen, wie z.B. den Schikanen von Vibrationswendelförderern. Vielmehr geht es darum, die in Montageanlagen zum Einsatz kommenden Technischen Systeme zu klassifizieren und zuzuordnen, für welche "Zweckfunktionen" zur Ausführung von Montageoperationen sie

eingesetzt werden können, wobei gemäß obiger Einteilung zwischen Wirksystemen, Steuersystemen und Hilfssystemen unterschieden werden kann.

Betrachtet man die Montageoperationen auf den Ebenen Fertigungsbereich n , Arbeitsgang $n-1$ und Fertigungsoperation $n-2$, so lassen sich auf jeder Ebene den einzelnen Montageoperationen Wirksysteme und Hilfssysteme zuordnen (Bild 45). Aus den Operationsfolgen können, unter Berücksichtigung der eingesetzten Wirksysteme und Hilfssysteme, erforderliche Steuer- und Regelfunktionen abgeleitet und für diese wiederum entsprechende Funktionsträger bestimmt werden.

	Montageoperationen	Wirkorgane/ Hilfsorgane	Steuer-/Regelfunktionen	Steuerorgane
EBENE : n Fertigungsbereich	Fördern (Transportieren) Zusammenbauen Lagern	Montageanlagen Fördersysteme	Montagesteuerung	Leitrechner
EBENE : $n-1$ Arbeitsgang	Vorarbeiten Zuführen Montieren Wegbringen Nacharbeiten	Montagezellen Transportsysteme	Zellensteuerung	Zellenrechner
EBENE : $n-2$ Fertigungsoperation	Handhaben Kontrollieren Fügen Justieren Sonderoperationen	Handhabungsgeräte Vibrationswendelförderer Greifer Gestelle Greiferwechseinrichtungen	Gerätesteuerung	Robotersteuerung (RC) Numerische Steuerung (NC) Speicherprogrammierbare Steuerung (SPS)

Bild 45: Funktionsträger nach Hierarchiestufen

Die Steuer- und Regelfunktionen auf den Ebenen n und $n-1$ haben mehr organisatorischen Charakter im Gegensatz zu denen auf der Ebene $n-2$, die rein technischer Art sind.

5. Systematische Planung automatisierter Montageanlagen

Nach [77] lassen sich zwei Aspekte unterscheiden, unter denen Schichten-Architektur-Modelle genutzt werden können:

1. Realisierungsaspekt:

Im Hinblick auf die Realisierung eines entsprechenden konkreten Systems soll das Modell Anhaltspunkte für eine schrittweise Planung des Systems liefern.

2. Referenzaspekt:

Für den Vergleich oder die Klassifikation von realisierten Systemen bzw. zur Einordnung von Teilsystemen in ein Gesamtsystem bietet das Modell begriffliche Rahmen.

Nachdem im vorigen Kapitel der Referenzaspekt des in Abschnitt 4.2.1 vorgestellten Schichten-Architektur-Modells für die Montage im Vordergrund stand, soll in diesem Kapitel der Realisierungsaspekt betrachtet werden.

5.1 Allgemeines zur Montageplanung

Die rechnergestützte Montageanlagenplanung ist eine Teilfunktion im Rahmen eines rechnerintegrierten Produktionsbetriebes. Bevor nun die systematische Montageanlagenplanung genauer analysiert wird, soll sie zunächst in die Funktionsbereiche eines Produktionsbetriebes eingeordnet und Schnittstellen zu anderen Bereichen aufgezeigt werden.

5.1.1 Einordnung der Montageplanung in die Funktionsbereiche eines Produktionsbetriebes

In einer rechnerintegrierten Produktion, heute meist mit dem Schlagwort CIM (Computer Integrated Manufacturing) belegt, lassen sich bei der Auftragsbearbeitung zwei Datenströme unterscheiden (Bild 46).

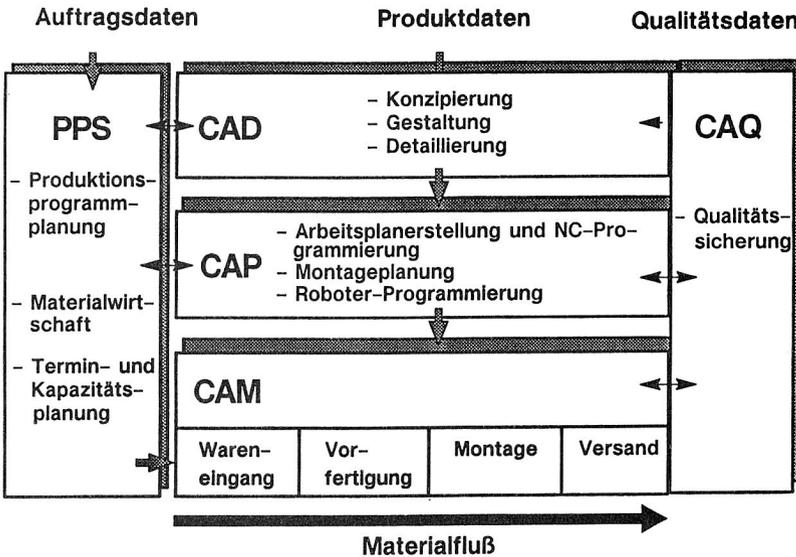


Bild 46: Funktionsbereiche der computerintegrierten Produktion

Neben den Produktdaten, die mehr den geometrisch/technologisch orientierten Bereichen der technischen Auftragsabwicklung zugeordnet werden können, gibt es noch die Auftragsdaten, die von der betriebswirtschaftlich orientierten administrativen Auftragsabwicklung verarbeitet werden.

Aufgabe der administrativen Auftragsabwicklung, der Produktionsplanung und -steuerung (PPS), ist die Planung der Produktionsab-

läufe sowie die Steuerung der Ausführung. Die Funktionen der Planungsphase umfassen die Produktionsprogrammplanung, die Mengenplanung (Materialwirtschaft) und die Termin- und Kapazitätsplanung (Zeitwirtschaft). Zur Produktionssteuerung gehören Auftragsveranlassung und Auftragsüberwachung.

In der technischen Auftragsabwicklung werden die Bereiche Konstruktion, Arbeitsvorbereitung und Fertigung durchlaufen. CAD (Computer Aided Design) in der Konstruktion umfaßt die Festlegung von Funktion und Gestalt des Produktes. CAP (Computer Aided Planning) als Teil der Arbeitsvorbereitung beinhaltet die rechnergestützten Planungsaufgaben, die für die Herstellung des Produktes notwendig sind. Die Ergebnisse der CAP-Funktionen sind Arbeitspläne und Steuerinformationen, die mit CAM (Computer Aided Manufacturing) in der Fertigung umgesetzt werden. Die Vorfertigung liefert dabei Einzelteile, die in der Montage zu Baugruppen oder Enderzeugnissen zusammengesetzt werden. Mit CAQ (Computer Aided Quality Assurance) werden die rechnergestützt ausgeführten Funktionen der Qualitätssicherung bezeichnet, die im Sinne einer Rückkopplungsfunktion den gesamten Produktentstehungsprozeß begleiten. In der Fertigung fließen die technischen und die administrativen Daten zusammen. Jedoch auch schon vorher bestehen sehr vielfältige Beziehungen zwischen den PPS-Funktionen und den CAD- und CAP-Funktionen, die einen regen Datenaustausch erfordern.

Stehen für die Herstellung eines Produktes keine geeigneten Betriebsmittel, d.h. Werkzeuge, Vorrichtungen und Maschinen, zur Verfügung, was insbesondere im Montagebereich häufig vorkommt, so ist es Aufgabe des Fertigungsmittelbaus derartige Betriebsmittel herzustellen oder zu beschaffen.

Der Fertigungsmittelbau, der der Arbeitsplanung zugeordnet ist, kann wie ein kleiner Produktionsbetrieb im Produktionsbetrieb betrachtet werden (Bild 47). Er benötigt wie dieser eine Konstruktion, die Betriebsmittelkonstruktion, in der die Funk-

tionen und die Gestalt der Betriebsmittel festgelegt werden, sowie eine Arbeitsvorbereitung, die die Vorfertigung und Montage der Betriebsmittel plant und steuert.

In Bild 47 ist dargestellt, wie die Montagesystemerstellung als Teilaufgabe des Fertigungsmittelbaus in den Produktionsablauf eingeordnet werden kann.

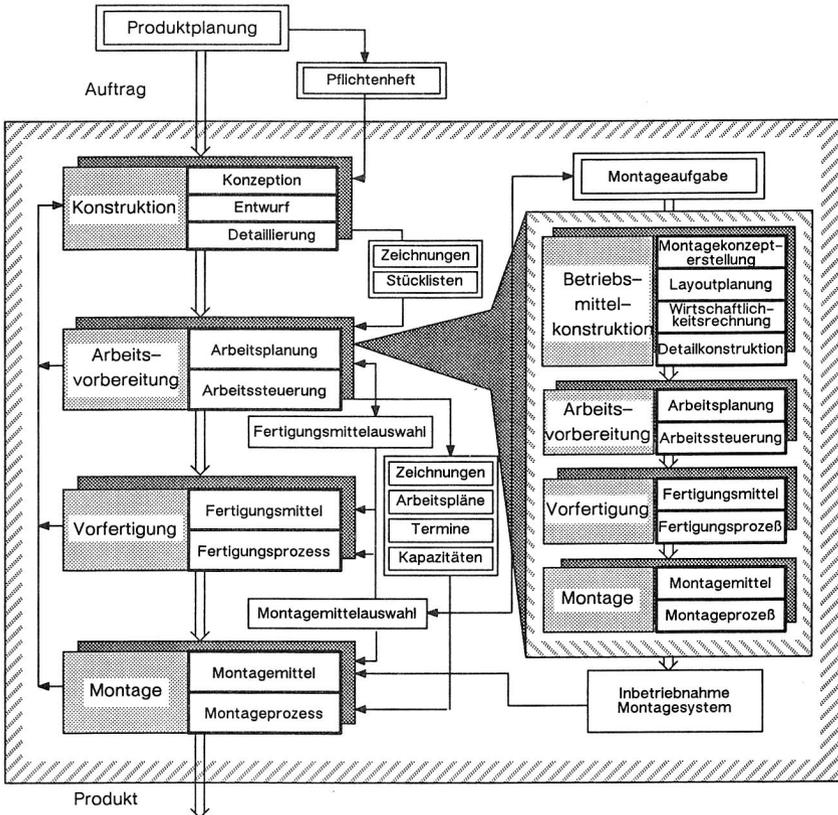


Bild 47: Einordnung der Montagesystemerstellung in den Produktionsablauf

Die Montagesystemerstellung spiegelt den gesamten Ablauf des Produktionsprozesses im Kleinen wider.

5.1.2 Ablauf der systematischen Montageanlagenplanung

Die Bemühungen, die Planung von Montageanlagen zu systematisieren und sie damit einer Rechnerunterstützung zugänglich zu machen (wobei dies natürlich nicht immer der Beweggrund war), wurden in Deutschland Mitte der siebziger Jahre stark intensiviert. In dieser Zeit entstanden zahlreiche, zum Teil grundlegende Beiträge zu dieser Thematik [80,91,92,93,94,95,96,97,98,99].

Diese Versuche, die Montagesystemplanung zu systematisieren, führten zu einer Vielzahl von Planungssystematiken. Übereinstimmend liegt allen Systematiken eine gestufte Vorgehensweise zugrunde. Der in Bild 48 wiedergegebene Planungsablauf ist in seinen wesentlichen Teilen allgemein anerkannt; Abweichungen sind bei anderen Autoren nur in Detailfragen zu finden wie z.B. in der Anzahl und somit Differenziertheit der Planungsstufen sowie der Planungshilfsmittel [100].

Die Montagesystemplanung beginnt mit der Analyse der Montageaufgabe. Aus der Analyse der Produktionsbedingungen werden die Anforderungen an die zu planende automatische Montageanlage abgeleitet. Es müssen verbindliche Aussagen des Vertriebes über die zu erwartende Lebensdauer und die Produktionsmenge sowie ein Mengengerüst über die Varianten des auf der Anlage zu montierenden Produktes vorliegen. Die Lebensdauer, d.h. die voraussichtliche Verkaufsdauer des Produktes, bestimmt die Nutzungsdauer der automatisierten Anlage. Die geforderte Produktionsmenge dient zur Abschätzung der notwendigen Sollproduktionsleistung.

Die Analyse des Produktes gibt Aufschluß über die Komplexität der Anlage. Die Sollproduktionsleistung und Teilezahl des

Produktes sind maßgeblich für die Größe der automatisierten Montageanlage [84].

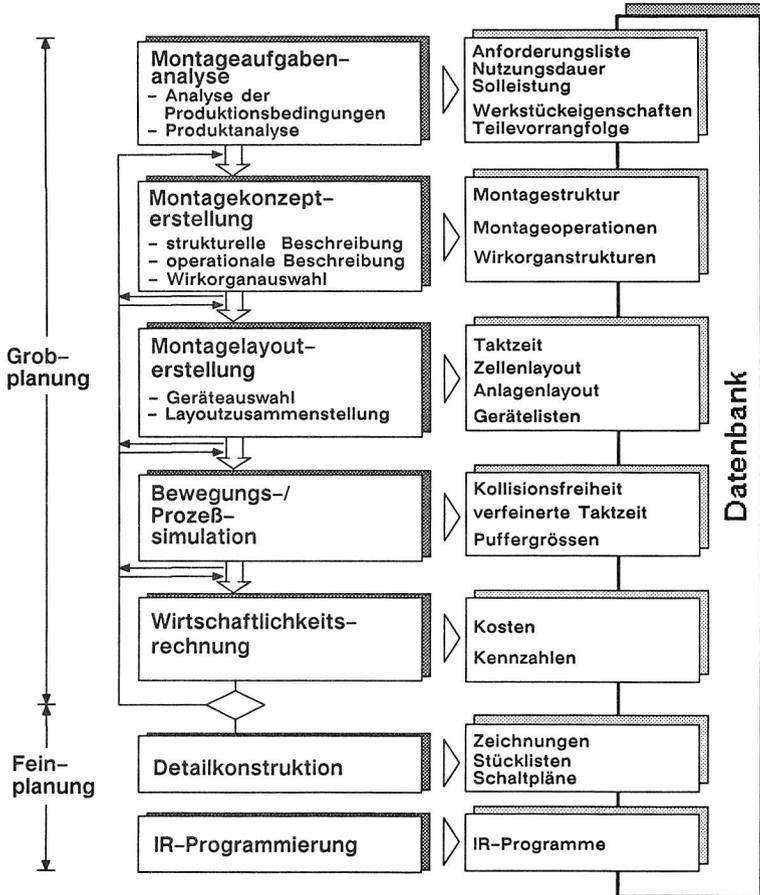


Bild 48: Ablauf der systematischen Montageanlagenplanung

Die Produktanalyse soll weiterhin Informationen liefern über den Anlieferungszustand der Teile, über Handhabungseigenschaften und über die möglichen Reihenfolgen, in denen die Bauteile zum Produkt zusammengebaut werden können.

Aus diesen möglichen Reihenfolgen wird eine ausgewählt und für diese Reihenfolge ein Montagekonzept erstellt, das eine strukturelle und operationale Beschreibung der Montageaufgabe mit den erforderlichen Montage- und Handhabungsoperationen sowie den zugeordneten Wirkorganen enthält. Häufig wird auf eine explizite operationale Beschreibung des Montageablaufs verzichtet, weil sie zu aufwendig ist. Sie existiert dann nur im Kopf des Planers, und es wird direkt zur Auswahl von Wirkorganen übergegangen. Anhand der operationalen bzw. falls diese fehlt der strukturellen Beschreibung des Montageablaufs werden entsprechende Funktionsträger (Wirkorgane) ausgewählt, mit denen sich die geforderten Montage- und Handhabungsoperationen für die jeweils zu montierenden Teile verwirklichen lassen.

Bei der Montagelayouterstellung werden zunächst die Wirkorgane unter Berücksichtigung der erforderlichen Handhabungszeiten und der angestrebten Taktzeit, die sich aus der geforderten Sollproduktionsleistung und einer geschätzten Verfügbarkeit ermitteln läßt, zu Montagezellen zusammengefaßt.

Im Anschluß daran werden für die Wirkorgane und für erforderliche Hilfs- und Steuerorgane konkrete Geräte ausgewählt und in einem Layout der einzelnen Montagezellen sowie der gesamten Montageanlage zusammengestellt.

Die Bewegungssimulation dient dazu, mögliche Kollisionen der Montagegeräte während der Ausführung der Montageaufgabe bei ihrer Anordnung entsprechend dem detaillierten Layout aufzudecken. Durch die Simulation des Anlagenverhaltens kann die geplante Lösung optimiert werden. Dabei geht es darum, die Pufferelemente und Transportsysteme zwischen den einzelnen Komponenten der konzipierten Montageanlage hinsichtlich ihrer Kapazität zu dimensionieren, einzelne Komponenten neu zusammenzustellen und so die Struktur der Montageanlage zu verändern oder höhere Anforderungen an die Einzelverfügbarkeiten der Verfügbarkeitsgrade der Anlage zu erkennen.

Die Wirtschaftlichkeitsrechnung soll Aufschluß darüber geben, inwieweit der Einsatz der geplanten automatischen Montageanlage nach wirtschaftlichen Gesichtspunkten sinnvoll ist. Sie dient als Entscheidungskriterium dafür, ob die Montageanlage nach dem Konzept ausgeführt wird oder nicht.

Die Grobplanung der Montageanlage ist damit beendet, und es schließt sich die Feinplanung an. Sie umfaßt die Detailkonstruktion, bei der Zusammenbauzeichnungen von den Montagezellen, Einzelteilzeichnungen von "Nichtzukaufteilen" und Stücklisten, sowie Schaltpläne für pneumatische, hydraulische und elektrische Antriebs- und Steuereinheiten erstellt werden.

Bereits zum Bereich der Arbeitsplanung im Rahmen der Montagesystemerstellung gehört die Programmierung von Roboterarbeitszellen. Die bei der Bewegungssimulation erzeugten Bewegungsprogramme können ergänzt um Technologieanweisungen zur Offline-Programmierung von Industrierobotern genutzt werden.

Da es sich bei der Planung von Montageanlagen um eine Konstruktionsaufgabe handelt, finden sich in diesem Planungsablauf auch die wesentlichen Konstruktionsphasen (siehe Abschnitt 4.2.4) wieder, die Funktionsfindung in Form der strukturellen und operationalen Montageablaufbeschreibung, die Prinzipierarbeit in Form der Wirkorganauswahl und die Gestaltung in Form der Montagelayoutzusammenstellung während der Grobplanung, sowie die Detaillierung in der Feinplanung.

Der skizzierte Planungsablauf ist ein iterativer Prozeß, d.h. die einzelnen Planungsschritte werden nicht rein sequenziell durchlaufen, sondern es erfolgen immer wieder Rücksprünge auf frühere Planungsschritte, in denen aufgrund gewonnener Erkenntnisse Variationen vorgenommen werden.

Für den in Bild 48 vorgestellten Planungsablauf existieren bislang keine durchgängigen rechnergestützten Lösungen. Einzelne

isoliert arbeitende Programmsysteme wurden für verschiedene Teilaufgaben entwickelt und erfordern meist ein hohes Maß an Eingabeaufwand.

Im folgenden werden die Planungsschritte der Grobplanung näher betrachtet. Es geht dabei weniger um die genaue Darstellung der dabei eingesetzten Verfahren; vielmehr soll im Vordergrund stehen, welche Daten verwendet und erzeugt werden. Die angestellten Betrachtungen dienen ja im Sinne einer Bedarfsanalyse dazu, ein Datenbankschema zur durchgängigen Rechnerunterstützung des gesamten Planungsablaufs zu entwerfen. Sie basieren zum einen auf verfügbarer Literatur zur Montageplanung und zum anderen auf einer in [101] beschriebenen Analyse des Informationsflusses und der Informationsinhalte bei der Montagesystemerstellung in der Praxis, die in einem Betrieb durchgeführt wurde, der elektro-mechanische Kleinteile fertigt und dazu zahlreiche Montageanlagen in Eigenleistung plant und herstellt.

Es werden nur die ersten Planungsschritte ausführlicher dargestellt, da zu diesen eigene Entwicklungen gemacht wurden; die anderen werden nur insoweit erläutert, wie es zum Verständnis erforderlich ist.

5.2 Analyse der Montageaufgabe

Die Analyse der Montageaufgabe setzt sich aus zwei Teilen zusammen, die zugleich die Schnittstellen zu den Funktionsbereichen PPS und CAD einer computerintegrierten Produktion nach Bild 46 charakterisieren.

Die Analyse der Produktionsbedingungen stützt sich vor allem auf Daten aus dem PPS-Bereich. Der PPS-Bereich mit seinen material- und zeitwirtschaftlichen Aufgaben hat gegenüber den Bereichen CAD, CAP und CAM eine lange Tradition und bereitet für den Einsatz konventioneller Datenbanken keine Schwierigkei-

ten [67]. Auf eine genauere Darstellung der Analyse der Produktionsbedingungen wird deshalb hier verzichtet; die Beschreibung in Abschnitt 5.1.2 wird für ausreichend gehalten. Soweit Daten dieser Analyse für spätere Planungsschritte benötigt werden, werden sie dort mitbetrachtet.

Die Produktanalyse erfolgt anhand des mit einem CAD-System erstellten Produktmodells. Das CAD-System ist auch das wesentliche Werkzeug zur Durchführung dieser Analyse.

Im Gegensatz zur Vorfertigung, wo die direkte Nutzung der in der Konstruktion mit CAD erzeugten Produktdaten nahezu Stand der Technik ist (CAD/CAM-Verfahrensketten), wird der Montageplaner bisher kaum unterstützt [102]. Die folgenden Ausführungen konzentrieren sich daher auf die Produktanalyse.

5.2.1 Das Produktmodell als Basis der Montageplanung

Ausgangspunkt für die Produktanalyse, die die eigentliche Grundlage für alle weiteren Montageplanungsschritte bildet, ist das mit einem CAD-System erstellte Produktmodell. Das Produktmodell repräsentiert die rechnerinterne Darstellung der geometrischen und topologischen Eigenschaften des Produktes.

In Bild 49 ist ein allgemeines konzeptionelles Schema für die Beschreibung von Produktmodellen in CAD-Systemen wiedergegeben, wie es beispielsweise dem geometrischen Modellierer Romulus zugrunde liegt.

Fundamental ist die Unterscheidung von topologischen und geometrischen Begriffen [67]. Den topologischen Elementen können vom Benutzer weitere benutzerdefinierte Attribute zugewiesen werden. Die Ausprägung (Instance) eines Körpers (Body) repräsentiert ein Einzelteil, und mehrere Instances zu einem Assembly zusammengefaßt repräsentieren eine Baugruppe.

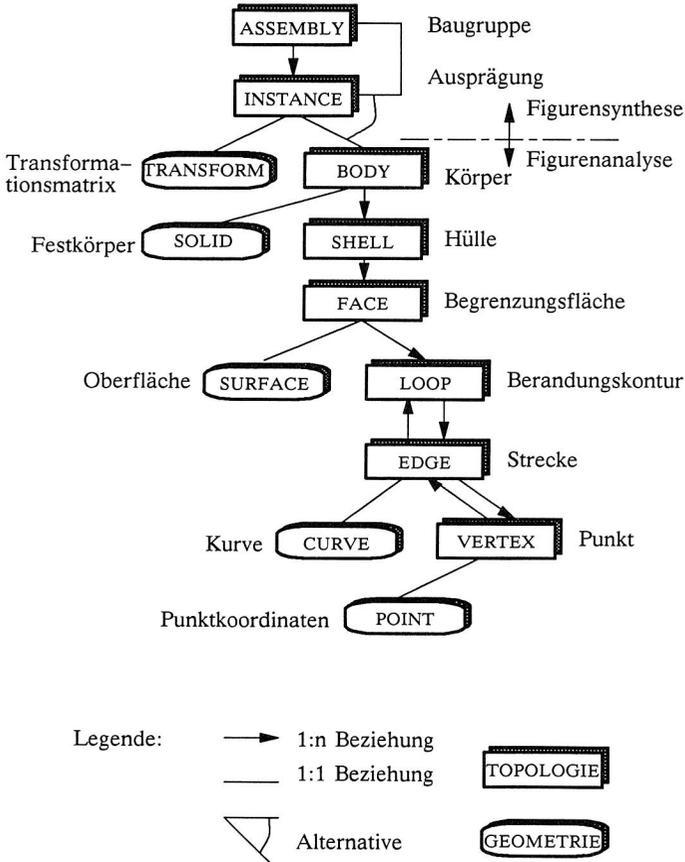


Bild 49: Allgemeines konzeptionelles Schema für CAD-Systeme (nach [67])

In der Regel ermöglichen CAD-Systeme die Bestimmung von Volumina, Schwerpunktlagen usw., so daß die in Bild 38 aufgeführten Werkstückeigenschaften und das Werkstückverhalten entweder analytisch ermittelt oder visuell bestimmt und den Einzelteilen und Baugruppen direkt zugeordnet werden können.

5.2.2 Analyse des Produktaufbaus

Durch die Analyse des Produktaufbaus soll ermittelt werden, aus wieviel Teilen das Produkt besteht, welches Teil sich als Basisteil eignet und in welchen möglichen Reihenfolgen das Produkt montiert werden kann.

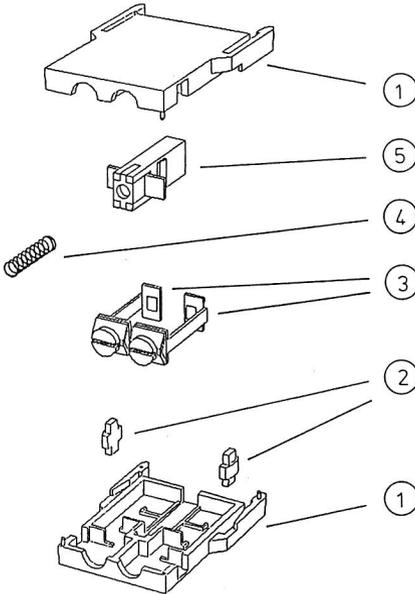


Bild 50: Explosionsdarstellung eines Schalters (Pos.1: Seitenteile; Pos.2: Keile; Pos.3: Klemmen; Pos.4: Feder; Pos.5: Schieber)

Zur Bestimmung der möglichen Montagereihenfolgen eignet sich ein algorithmierbares Verfahren nach Gairola [103], mit dem ein Teilevorranggraph aus einer Fügeflächenmatrix ableitbar ist. Dieses Verfahren wird anhand eines einfachen Beispiels, einem elektromechanischen Schalter (Bild 50) erläutert. Position 3 und 5 sind Baugruppen. Im Sinne einer Baukastenstückliste [104]

werden immer nur jeweils zwei Ebenen einer Erzeugnisgliederung betrachtet.

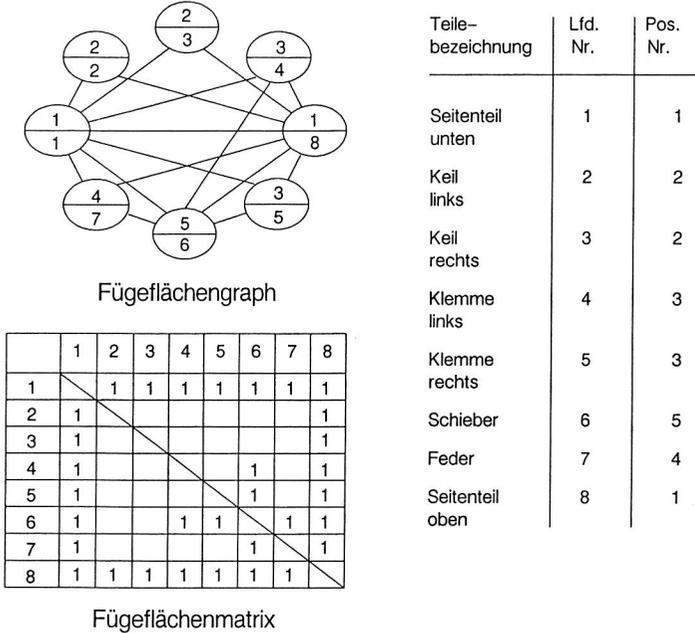
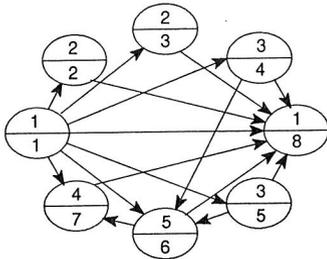


Bild 51: Fügeflächengraph und Fügeflächenmatrix

Die Bestimmung der möglichen Montagereihenfolgen beginnt damit, daß aus dem rechnerinternen Produktmodell automatisch abgeleitet wird, welche Teile sich berühren. Das Ergebnis kann in Form eines Fügeflächengraphen dargestellt werden, wobei Teile durch Knoten und Berührbeziehungen durch Kanten repräsentiert werden (Bild 51 links). Eine adäquate Darstellungsform ist die Fügeflächenmatrix (Bild 51 rechts). Die Indizierung der Spalten und Zeilen der Matrix ist identisch. Sie entspricht laufenden Nummern der Bauteile, die sich aus den vom Konstrukteur vorgegebenen Positionsnummern ableiten.

Wiederholt vorkommende Bauteile (Instances), die mit der gleichen Positionsnummer gekennzeichnet sind, müssen hier unterscheidbar sein. Sie erhalten jeweils eine eigene Nummer. Es ergibt sich eine quadratische Matrix in der die Fügeflächenbeziehungen durch eine eins (1) gekennzeichnet sind.



Gerichteter Graph

	1	2	3	4	5	6	7	8
1		1	1	1	1	1	1	1
2								1
3								1
4						1		1
5						1		1
6							1	1
7								1
8								

Obere Dreiecksmatrix

Bild 52: Gerichteter Fügeflächengraph und Fügeflächen-Dreiecksmatrix

Nach der Festlegung eines Basisteils kann der Fügeflächengraph überführt werden in einen gerichteten Graphen (Bild 52 links). Entsprechend entsteht aus der Fügeflächenmatrix eine symmetrische Matrix die in eine Dreiecksmatrix aufgespalten werden kann (Bild 52 rechts).

Der Fügeflächengraph, bzw. die Fügeflächenmatrix, liefert anhand der Anzahl der Beziehungen pro Bauteil Hinweise für die Auswahl eines Basisteils.

Geeignete Basisteile weisen nach [93] folgende Merkmale auf:

- große Anzahl gemeinsamer Fügeflächen mit anderen Bauteilen
- möglichst viele Fügekombinationen ohne Vorgängerbeziehung
- geeignete Spann- und Auflageflächen (Werkstückträgereigenschaften)

- möglichst nur horizontale oder vertikale Fügebewegungen
- kein Wenden der teilmontierten Baugruppe erforderlich.

Für das Beispiel wird das untere Seitenteil als Basisteil gewählt.

Aus der Dreiecksmatrix kann nun in einem weiteren Optimierungsschritt automatisch der Teilevorranggraph bestimmt werden.

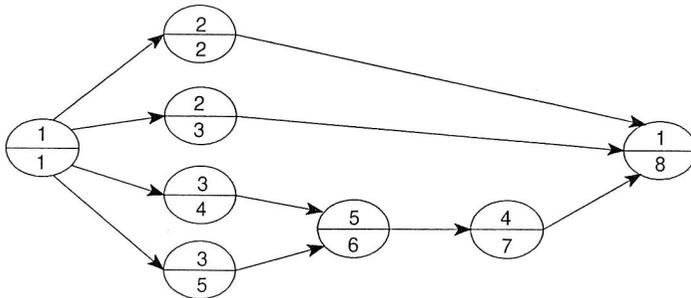


Bild 53: Teilevorranggraph zur Darstellung der möglichen Montagereihenfolgen

Der Teilevorranggraph (Bild 53) enthält alle Reihenfolgebeziehungen der Bauteile, die technologisch von Bedeutung sind um das Produkt montieren zu können. Er bildet die Grundlage für alle weiteren Planungsschritte.

5.3 Erstellung eines Montagekonzepts

Durch das Montagekonzept wird der Ablauf der Montage des Produktes beschrieben. Es wird eine Struktur bestimmt, wie die zum Zusammenbau des Produktes erforderlichen Montageabschnitte in den Montageablauf einzuordnen sind. Für diese Struktur werden dann die Operationsfolgen festgelegt, durch die die einzelnen Bauteile zum Produkt zusammengebaut werden.

Anschließend werden zur Ausführung dieser Operationsfolgen geeignete Wirkorgane ausgewählt.

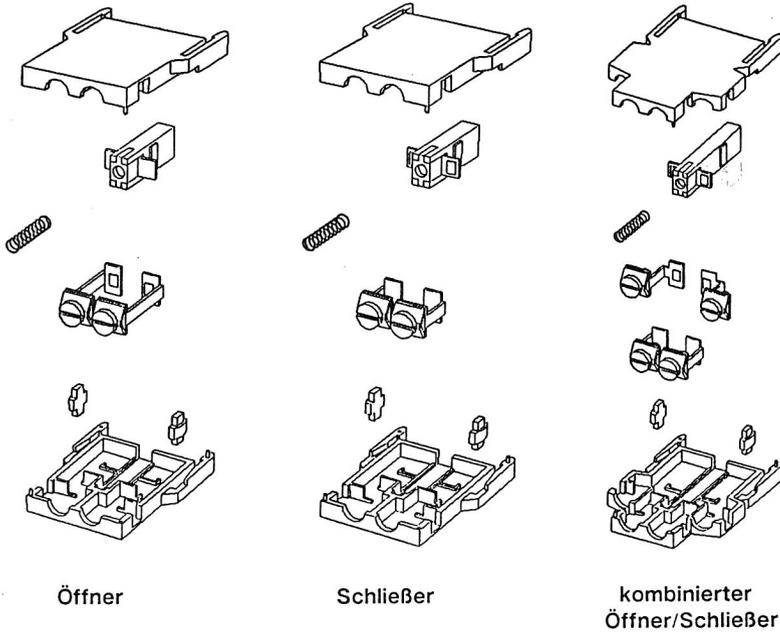


Bild 54: Varianten des Schalters

Zur Durchführung und Darstellung der ersten beiden Arbeitsschritte bei der Erstellung des Montagekonzeptes wurde ein neues Verfahren [100] entwickelt, in das die bei der systemtechnischen Betrachtung der Montage (Abschnitt 4.2) gewonnenen Erkenntnisse eingeflossen sind. Das Verfahren zeichnet sich dadurch aus, daß mit formalen Mitteln die strukturelle Beschreibung des Montageablaufs direkt in eine operationale Beschreibung überführt werden kann.

Für die Beschreibung des Verfahrens möge wieder das Beispiel von oben dienen. Es wird erweitert um zwei Varianten des Schal-

ters (Bild 54). Bei dem in Bild 50 dargestellten Schalter handelte es sich um einen "Öffner". Bei Betätigung des Schiebers wird der Kontakt zwischen den beiden Klemmen unterbrochen. Wechselt man den Schieber und die Klemmen gegen andere Varianten dieser Baugruppen aus, so entsteht ein "Schließer" (Bild 54 mitte) und durch Kombination der beiden Varianten in einem neuen Gehäuse bestehend aus zwei anderen Seitenteilen wird daraus ein "Öffner und Schließer" (Bild 54 rechts).

5.3.1 Strukturelle Beschreibung des Montageablaufs

Die Beschreibung der Montagestruktur baut auf den Teilevorranggraphen auf.

Aus den Teilevorranggraphen der Schaltervarianten nach Bild 54 wird eine Teileeingangsfolge festgelegt (Bild 55), indem die einzelnen Teilefolgen jeder Variante von oben nach unten in Spalten nebeneinander angeordnet werden.

Die Werkstücke des Montageprozesses werden dabei durch die folgenden Symbole gekennzeichnet:

- Kreise für Einzelteile
- Trapeze für Baugruppen
- zwei konzentrische Kreise für Produkte
- Rauten für Hilfsteile.

Teile verschiedener Varianten mit den gleichen Handhabungseigenschaften werden in einer Zeile angeordnet.

Diese Teileeingangsfolge wird nun zu einer strukturellen Beschreibung des Montageablaufes erweitert (Bild 56).

Zunächst werden erforderliche Hilfsteile in der Teileeingangsfolge ergänzt. Im Beispiel wird an der ersten Stelle der Teileeingangsfolge ein Werkstückträger für das Basisteil eingefügt.

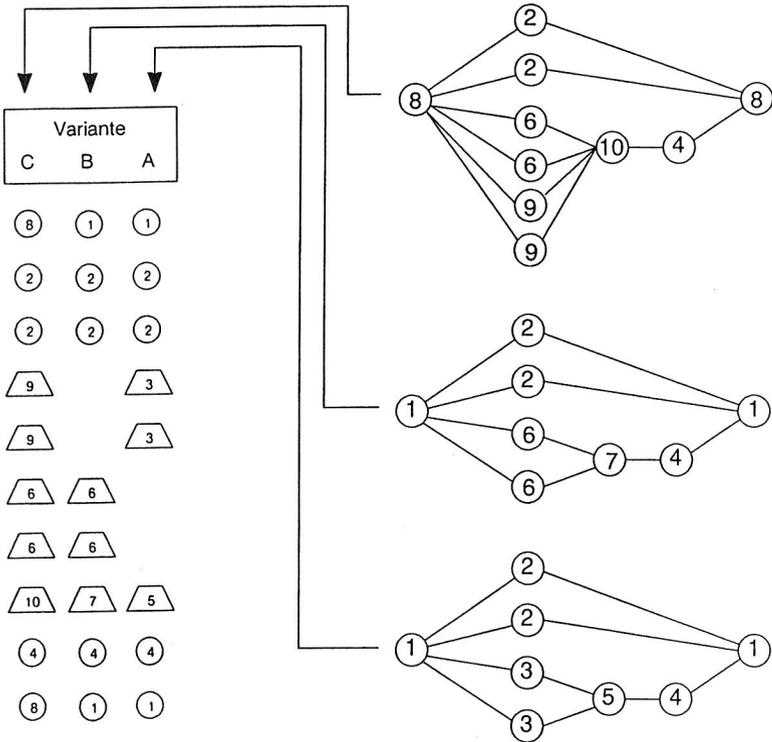


Bild 55: Erstellung der Teilleeingangsfolge

Zur Beschreibung des eigentlichen Montageablaufs werden zwei Strukturelemente benutzt. Handhabungspfade werden durch Linien mit Pfeilen dargestellt und Montageblöcke durch Rechtecke. Die Wahl dieser beiden Elemente stützt sich auf die in Abschnitt 4.2.2 vorgestellte Untergliederung der Montageoperationen auf den Ebenen $n-1$ und $n-2$. Auf der Arbeitsgangebene $n-1$ wurde zwischen Vorarbeiten, Zuführen, Montieren, Wegbringen und Nacharbeiten unterschieden (Bild 40). Auf der Ebene der Fertigungsoperationen $n-2$ (Bild 40) setzten sich Vorarbeiten und Nacharbeiten aus Sonderoperationen und Justieren, Zuführen

und Wegbringen aus Handhaben und Kontrollieren zusammen. Da in der VDI-Richtlinie 2860 Handhabungs- und Kontrolloperationen unter Handhabungsfunktionen zusammengestellt sind, werden sie auch hier gemeinsam als Handhabungspfad betrachtet.

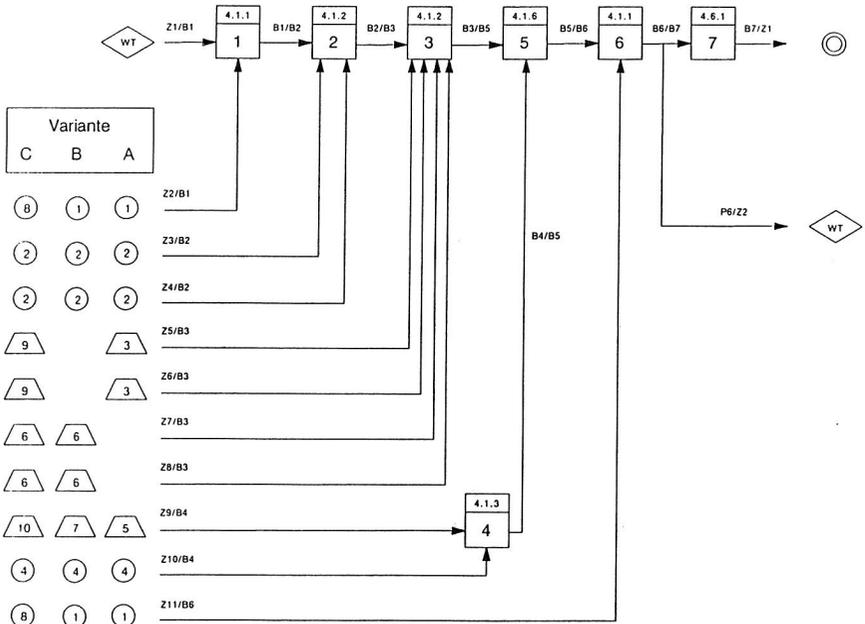


Bild 56: Strukturelle Beschreibung des Montageablaufes

Vorarbeiten und Nacharbeiten sind meist in den Ablauf des Zuführens oder Wegbringens integriert. Sie sind daher für die strukturelle Beschreibung des Ablaufs der Montage von untergeordneter Bedeutung, und auf eine explizite Darstellung wird auf dieser Beschreibungsebene verzichtet. Sollte es sich um umfangreichere eigenständige Operationen handeln, wie z.B. Entgraten, so können diese mit dem Strukturelement Montageblock erfaßt werden.

Das Montieren als zentraler Vorgang wird durch Montageblöcke dargestellt. Sie beinhalten eine Fügeoperation weshalb sie mit der Nummer des Fügeverfahren nach DIN 8593 [3] bezeichnet und in der Reihenfolge ihrer Ausführung numeriert werden. Die Handhabungspfade repräsentieren die dem Montieren vorangehenden und nachfolgenden Operationen. Dabei symbolisieren Handhabungspfade, die in Montageblöcke münden, das Zuführen, und solche, die ihren Ursprung in Montageblöcken haben, das Wegbringen.

Handhabungspfade werden mit den Abkürzungen der Elemente ihres Anfangs und ihres Endes gekennzeichnet. Beginnt ein Pfad an Zeile 5 der Werkstückeingangsfolge und endet im Montageblock 3, so ist er als Z5B3 identifizierbar.

Am rechten Rand der graphischen Beschreibung des strukturellen Montageablaufs steht die Werkstückausgangsfolge.

Nach diesen Konventionen läßt sich der in Bild 56 für das Beispiel der drei Schaltervarianten dargestellte strukturelle Montageablauf wie folgt interpretieren:

Zunächst wird der Werkstückträger zugeführt (Pfad Z1/B1). Im Montageblock 1 wird das im Handhabungspfad Z2/B1 zugeführte untere Seitenteil auf den Werkstückträger "aufgesetzt" (Fügeverfahrensnummer 4.1.1). In Pfad B1/B2 wird der Werkstückträger mit aufgesetztem Seitenteil weggebracht und zum "Einlegen" (Fügeverfahrensnummer 4.1.2) der zwei, in den Handhabungspfaden Z3/B2 und Z4/B2 zugeführten, Keile dem Montageblock 2 zugeführt. Pfad B2/B3 symbolisiert das Wegbringen und Zuführen der Werkstückträgers mit aufgesetztem Seitenteil und eingelegten Keilen für das gemeinsame "Einlegen" der zwei in Pfad Z5/B3 und Z6/B3 zugeführten Klemmen der Öffnervariante im Montageblock 3. Die Klemmenvarianten für Schließer und kombinierten Öffner und Schließer werden in den Pfaden Z7/B3 und Z8/B3 zugeführt. Die Pfade Z9/B4 und Z10/B4 beinhalten die Handhabungsfolge für das Zuführen des Schiebers und der Feder. Sie werden im Montageblock 4 "ineinandergeschoben" (Fügeverfahrensnummer 4.1.3) und mit zusammengedrückter Feder im Montageblock 5 "federnd einge-

spreizt" (Fügeverfahrensnummer 4.1.6). Im Block 6 wird das Seitenteil von Pfad Z11/B6 "aufgesetzt" (Fügeverfahrensnummer 4.1.1). Im Handhabungspfad B6/B7 wird der zusammengebaute Schalter vom Werkstückträger heruntergenommen, und im Block 7 wird die Montage durch die Operation "Ultraschallverschweißen" (Fügeverfahrensnummer 4.6.1) abgeschlossen.

5.3.2 Operationale Beschreibung der Montageaufgabe

Die strukturelle Beschreibung des Montageablaufs entspricht der Arbeitsgangebene $n-1$ (Bild 40). Auf der Ebene der Fertigungsoperationen $n-2$ (Bild 40 und 41) kann dieser Ablauf detaillierter beschrieben werden.

Die Handhabungspfade der strukturellen Beschreibung kennzeichnen auf der Arbeitsgangebene die dem Montieren vor- oder nachgelagerten Montageoperationen. Auf Ebene der Fertigungsoperationen werden sie mit Hilfe von Handhabungssymbolen nach VDI-Richtlinie 2860 [82] genauer beschrieben. In Bild 57 ist beispielhaft für den Pfad Z2/B1 aus Bild 56 eine derartige Verfeinerung gezeigt.

Die Seitenteile sind zunächst ungeordnet gebunkert, werden dann vereinzelt und orientiert. Nach einer Anwesenheitsprüfung werden die Seitenteile geführt weitertransportiert und an einer bestimmten Stelle positioniert. Der Handhabungspfad Z2/B1 endet mit der Prüfung der Position.

Der Montageblock steht für den Vorgang des Montierens auf der Arbeitsgangebene. Operational kann dieser Vorgang untergliedert werden in Handhabungsoperationen nach VDI-Richtlinie 2860 [82] und in eine Fügeoperation, die nach einem Verfahren aus DIN 8593 [83] genauer benannt werden kann. Die Handhabungsoperationen werden mit Hilfe der VDI-Symbole untergliedert, während die Fügeoperation den Charakter einer Elementaroperation hat und nicht weiter unterteilt wird. Die Unterscheidung zwischen

Handhabungsoperationen der Handhabungspfade und denen eines Montageblockes ist Ausdruck der gerätetechnischen Trennung zwischen der Teilezuführung und dem Montieren, wie sie in der industriellen Praxis vorzufinden ist [79].

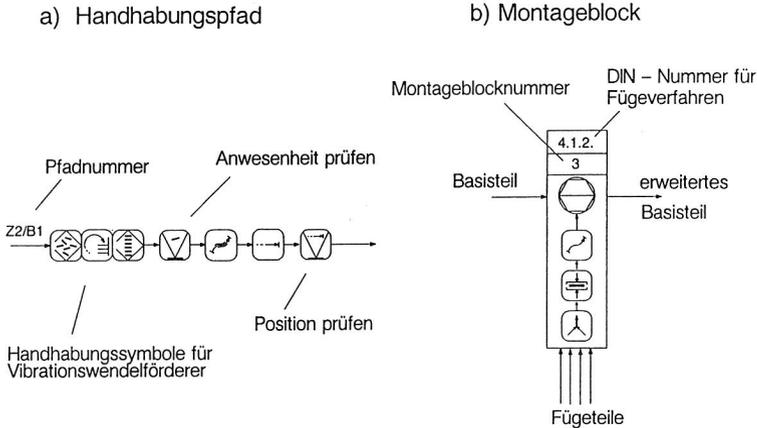


Bild 57: Operationale Beschreibung von Handhabungspfad und Montageblock

Bild 57 zeigt eine operationale Verfeinerung am Beispiel des Montageblock 3 aus Bild 56. Während sich ein Handhabungspfad jeweils nur auf ein Einzelteil, ein Hilfsteil, eine Baugruppe oder eine Ansammlung von zusammengebauten Teilen bezieht, sind in der Regel an den Operationen des Montageblocks zwei Operanden beteiligt, das Fügeteil und das Basisteil, in das gefügt wird. Demzufolge hat der Montageblock einen Eingang, an dem das Basisteil zugeführt, und einen Ausgang, an dem das gegebenenfalls erweiterte Basisteil weggebracht wird. Die Darstellung des Querspades im Montageblock zwischen diesem Eingang und Ausgang reduziert sich auf das VDI-Symbol für Fügen. Alle anderen Handhabungsoperationen sind dem vorgelagerten bzw. sich anschließenden Pfad zuzuordnen. An der Unterseite des Montageblocks enden

die Handhabungspfade die die Zuführung der Füge­teile beschreiben. Diese Pfade setzen sich im Montage­block fort, da die Füge­teile noch eine Reihe von Handhabungs­operationen durchlaufen, die ausschließlich vom Füge­gerät ausgeführt werden, und enden in der Füge­operation. Wenn mehrere Handhabungspfade von Füge­teilen an einem Montage­block enden, so müssen sich die Fortset­zungen dieser Pfade im Montage­block an mindestens einer Stelle vereinigen, sonst sind getrennte Montage­blöcke einzuführen.

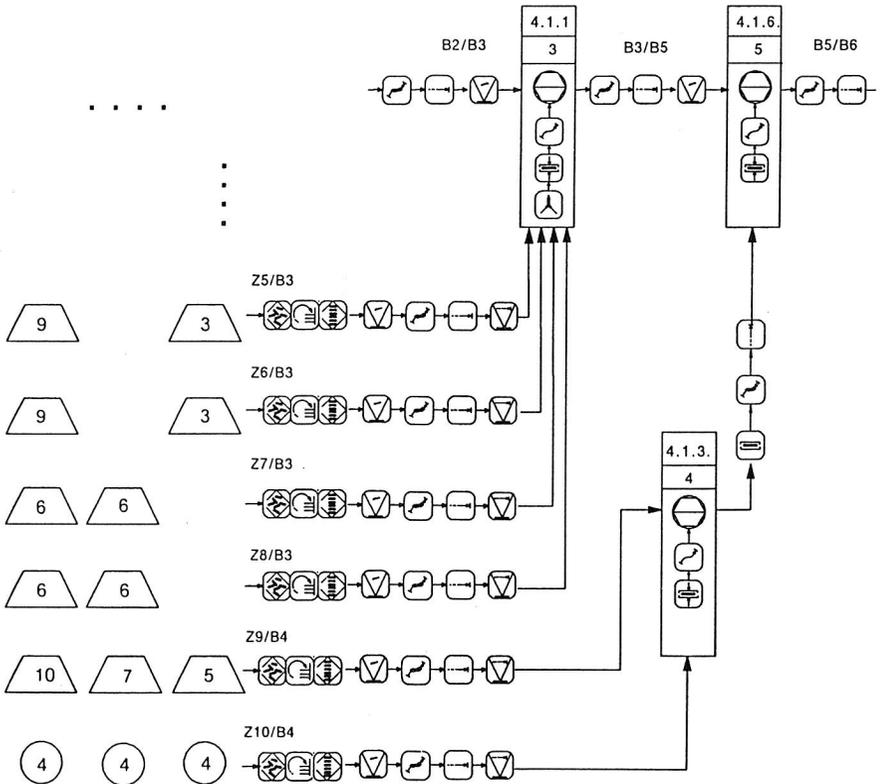


Bild 58: Operationale Beschreibung des Montageablaufs

Werden alle Handhabungspfade und Montageblöcke der strukturellen Darstellung in dieser Weise operational verfeinert beschrieben, so ergibt sich die in Bild 58 dargestellte operationale Beschreibung des Montageablaufs.

Das vorgestellte Verfahren bietet die Möglichkeit, den Montageablauf auf operationaler Ebene auch für komplexe Montageaufgaben anschaulich und übersichtlich zu beschreiben. Die Erstellung einer solchen Beschreibung des Montageablaufs ist jedoch nur mit Rechnerunterstützung sinnvoll, da der Aufwand für das manuelle Zeichnen der operationalen Darstellung, insbesondere bei großer Teile- und Variantenzahl, viel zu groß ist. Es muß zudem berücksichtigt werden, daß nicht sofort im ersten Planungsdurchlauf eine optimale Lösung gefunden wird und deshalb immer wieder Änderungen an der strukturellen und operationalen Beschreibung vorzunehmen sind.

5.3.3 Auswahl von Wirkorganen

Die operationale Beschreibung des Montageablaufs bildet die Grundlage für die Auswahl geeigneter Operatoren, mit denen die benötigten Operationen bewirkt werden können. Es müssen geeignete Funktionsträger (Wirkorgane) gefunden werden, deren "Zweckfunktionen" (vgl. Abschnitt 4.2.3) den Operationen der operationalen Beschreibung entsprechen.

Wirkorgane stellen prinzipielle Beschreibungen von technischen Einrichtungen als Funktionsträger für Montage- und Handhabungsfunktionen dar. Häufig lassen sich durch eine technische Einrichtung mehrere Funktionen realisieren, wobei einzelne Zweckfunktionen den Charakter von Hauptfunktionen annehmen können. Hauptfunktionen sind Zweckfunktionen, die die wesentliche Aufgabe einer Einrichtung beschreiben [105]. Entsprechend diesen Hauptfunktionen lassen sich die Handhabungseinrichtungen in Geräteklassen einteilen (Bild 59).

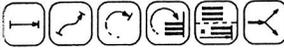
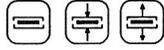
Handhabungseinrichtungen							
Speicher-einrichtungen 		Einrichtungen zum Verändern der Menge, Position und Orientierung 			Spanneinrichtungen 		Kontroll-einrichtungen 
Bunker	Maga-zine	Zuführ-einrich-tungen	Ord-nungs-einrich-tungen	Zuteil-einrich-tungen	Greifer	Spann-vorrich-tungen	Kontroll-einrich-tungen
-Trommel -Trichter	-Palette -Schacht-magazin -Ketten-magazin	-Program-mier-bares Hand-habungs-gerät -Pendel-schritt-förderer -Roll- u. Gleit-kanal	-Vibra-tions-wendel-förderer -Schlepp-ketten-förderer -Trommel-bunker	-Schleiber -Schleuse -Weiche	-Backen-greifer -Vaku-um-greifer -Abform-greifer	-Backen-futter -Elektro-magnet-spanner -Zangen-spanner	-Prüf-taster -Photo-zelle -Ober-flächen-meiß-gerät

Bild 59: Gliederung von Handhabungseinrichtungen in Geräte-
klassen nach Gerätehauptfunktionen (nach [106])

Eine Sonderstellung innerhalb der Handhabungseinrichtungen nehmen die Einrichtungen für die Handhabungsfunktionen ein, die in der VDI-Richtlinie unter dem Begriff "Bewegen" zusammengefaßt sind. Bezüglich der Funktionsvielfalt lassen sie sich einteilen in Bewegungseinrichtungen mit fester Hauptfunktion und solche mit variabler Hauptfunktion [82]. Zu denen mit fester Hauptfunktion zählen die Ordnungseinrichtungen und Zuteileinrichtungen aus Bild 59.

Von besonderer Bedeutung für die automatisierte Montage sind die programmgesteuerten Bewegungsautomaten, die zu den Bewegungseinrichtungen mit variabler Hauptfunktion zählen. Sie lassen sich wiederum unterteilen in Einlegegeräte und Industrieroboter. Einlegegeräte sind fest programmierte Bewegungsautomaten und werden vorzugsweise in der Massenfertigung eingesetzt [106]. Immer größere Bedeutung in der Montage gewinnen die frei programmierbaren Bewegungsautomaten, die Industrieroboter.

Wesentliches Merkmal für die Auswahl von Bewegungsautomaten als Wirkorgane für Handhabungsfunktionen ist ihre Kinematik. Der Arbeitsraum eines Bewegungsautomaten wird durch die Anordnung und die Zahl seiner Bewegungsachsen bestimmt.

Im dreidimensionalen Raum ist die Lage eines starren Körpers durch seinen Ortsvektor und seine Orientierung definiert. Ist er frei beweglich, so besitzt er sechs Freiheitsgrade und kann durch drei Rotationen und drei Translationen in eine beliebige andere Lage gebracht werden.

Entsprechend setzen sich die meisten bekannten Industrieroboter aus drei Bewegungsachsen (zum Positionieren) sowie bis zu drei weiteren Handachsen (zum Orientieren der Werkstücke) zusammen.

Für einige Fügeoperationen können die programmierbaren Bewegungsgeräte ebenfalls eingesetzt werden. So können die Fügeverfahren des Einsetzens, Schraubens, Schweißen und Klebens mit Industrierobotern ausgeführt werden, indem sie indirekt über andere Wirkorgane wie Greifer, Schrauber, Schweißzangen usw. auf die Werkstücke einwirken. Diese Verfahren haben nach [93] einen Anteil von 80 % in maschinellen Montageanlagen. Für einige Fügeverfahren wie z.B. Einpressen und Fügen durch Umformen sind meist spezielle Vorrichtungen erforderlich, die eigens konstruiert werden müssen.

5.4 Die Montagelayouterstellung

Die Montagelayouterstellung hat die Gestaltung der Montageanlage und deren Darstellung in einem Anlagenlayout zum Ziel. Dazu werden zunächst die Wirkorgane entsprechend den ihnen zugeordneten Arbeitsinhalten zusammengefaßt. Anschließend werden geeignete Geräte für die Wirkorgane ausgewählt und deren Anordnung in einem Layout dargestellt.

5.4.1 Abtaktung der Montagestationen

Die Wahl der Wirkorgane wird entscheidend beeinflusst von der Wahl der Organisationsform, d.h. der Aufteilung von Arbeitsinhalten und räumlichen Anordnung der Wirkorgane.

Die für die automatisierte Montage wichtigsten Organisationsformen sind die Punktmontage (Bild 60) und die Linienmontage (Bild 61) [107]. Bei der Linienmontage genügen häufig Handhabungsgeräte mit einer kartesischen Achskinematik für die Ausführung der Fügefunktion, während bei der Punktmontage kompliziertere Bewegungsabläufe erforderlich sind, die meist nur von Industrierobotern in Gelenkbauweise ausgeführt werden können. Die Wahl der Organisationsform hängt wiederum von der geforderten Produktionsmenge der Montageanlage und der zur Verfügung stehenden Montagefläche ab.

Bei der Punktfertigung werden mehrere Montagearbeitsgänge zu einer Montagestation zusammengefaßt. Der für die Montageanlage benötigte Platzbedarf ist dadurch geringer, es sinkt jedoch auch die Ausbringung, d.h. die mögliche Produktionsmenge.

Bei der Linienmontage werden die Arbeitsinhalte entsprechend der Montagereihenfolge auf verschiedene Stationen aufgeteilt. Dadurch verkürzen sich die erreichbaren Taktzeiten und die Ausbringung ist entsprechend höher. Es steigt jedoch auch der für das Aufstellen der Montageanlage erforderliche Platzbedarf.

Die ausgewählten Wirkorgane werden zu Montagestationen, auch Montagezellen genannt, zusammengefaßt. Dabei sollten die Arbeitsinhalte in den einzelnen Montagezellen so zusammengefaßt werden, daß sich eine möglichst einheitliche Taktzeit ergibt, die unter der notwendigen Taktzeit liegt. Die notwendige Taktzeit errechnet sich aus der geforderten Sollproduktionsleistung und einer geschätzten Verfügbarkeit der Montageanlage.

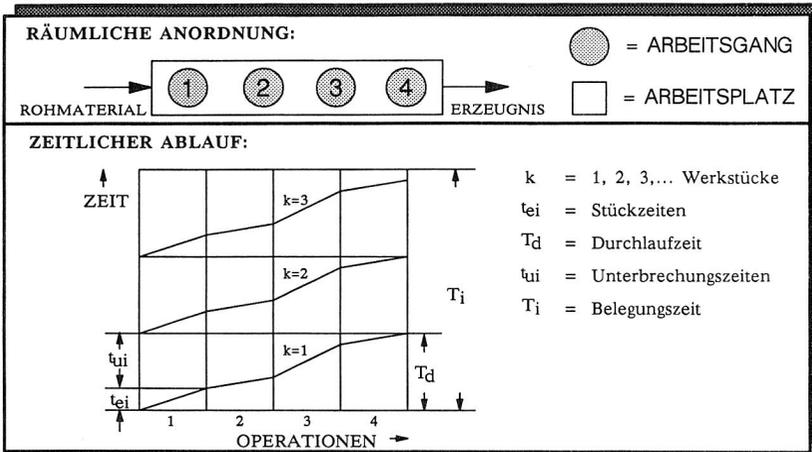


Bild 60: Fertigungsprinzip "Punktfertigung" (Quelle: [106])

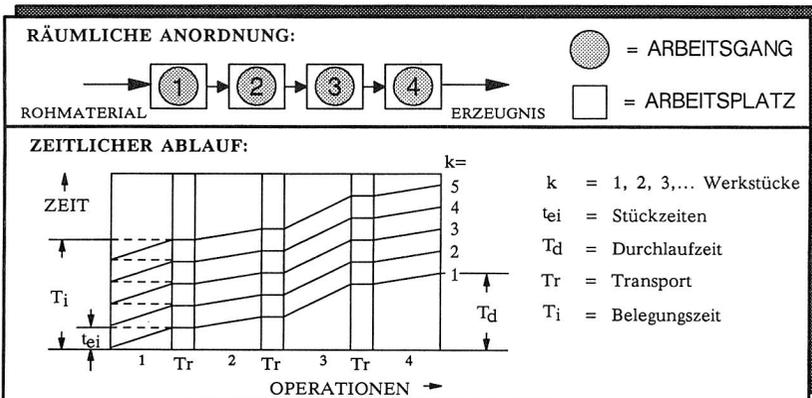


Bild 61: Fertigungsprinzip "Linienfertigung" (Quelle: [106])

Die Taktzeit von Montagezellen kann abgeschätzt werden, indem einzelnen Montageoperationen auf Erfahrungswerten beruhende

mittlere Ausführzeiten zugeordnet werden, die dann für nacheinander ablaufende Montageoperationen einer Montagezelle aufsummiert werden können [108,109].

Die für das gesamte Zusammenbauen des Produktes erforderlichen Montagezellen werden schließlich zu einer Montageanlage zusammengruppiert und durch geeignete Wirkorgane wie Band- oder Gurtförderer für den Transport zwischen den Zellen verbunden.

5.4.2 Geräteauswahl und Zusammenstellung zum Montagelayout

Durch die Bestimmung von Wirkorganen wurden die prinzipiellen Funktionsträger zur Lösung der Montageaufgabe festgelegt. Für jedes Wirkorgan, das das Funktionsprinzip einer Geräteklasse charakterisiert, gibt es eine Vielzahl dieser Klasse zugehöriger konkreter Geräteausprägungen mit unterschiedlichen Bauformen.

Nachdem durch die Wahl von Wirkorganen das Lösungsfeld möglicher Funktionsträger für die benötigten Montagefunktionen eingegrenzt wurde, werden nun einzelne Geräte ausgewählt. Bei der Geräteauswahl sind, neben der geforderten Gerätefunktion, die ja durch die Klassenzugehörigkeit erfüllt ist, viele weitere Einflußfaktoren zu berücksichtigen. Dazu werden die Geräteeigenschaften mit den sich aus der Montageaufgabe und aus sonstigen Randbedingungen ergebenden Anforderungen verglichen.

Technische Anforderungen leiten sich vor allem aus Werkstückeigenschaften, wie Werkstückgröße, Werkstückgewicht etc. ab. So ist z.B. sicherzustellen, daß die Traglast eines gewählten Handhabungsgerätes größer ist als das zu handhabende Gewicht, das sich zusammensetzt aus dem Gewicht des Werkstücks und des Greifers. Weitere Einflußfaktoren sind Eigenschaften des Gerätes wie dessen Raumbedarf und Gewicht. Es sind aber auch nichttechnische Kriterien wie die Kosten und mögliche Lieferzeiten eines Gerätes zu beachten.

Die Darstellung der Anordnung der Geräte in einem Zellenlayout erfolgt meist mit einem CAD-System. Bild 62 zeigt ein derartiges Layout einer Montagezelle.

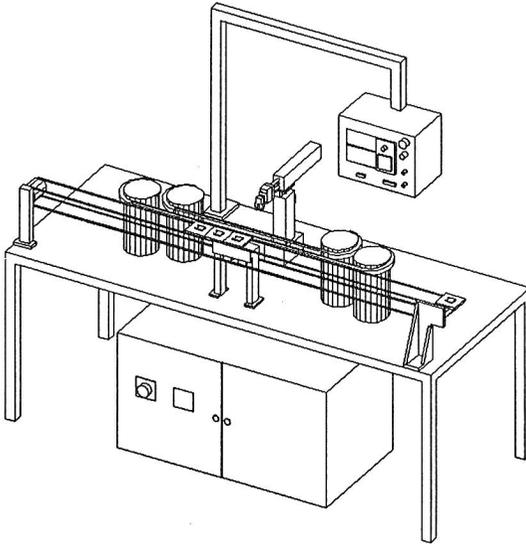


Bild 62: Layout einer Montagezelle

Die für das gesamte Zusammenbauen des Produktes erforderlichen Montagezellen werden zu einer Montageanlage gruppiert, die im Anlagenlayout dargestellt wird.

Zum Abschluß der Layoutplanung werden die einzelnen Komponenten der Montagezellen und der gesamten Montageanlage in Stücklisten erfaßt und zusammengestellt.

5.5 Optimierung des Layouts durch Simulation

Die Simulation dient der Optimierung des Montagelayouts. Grundsätzlich lassen sich zwei Arten der Simulation unterscheiden,

die Bewegungs- und die Prozeßsimulation. Durch die graphische Simulation der Bewegungen von Montagekomponenten und Peripherie lassen sich Kollisionen bereits in der Entwurfsphase erkennen und entsprechende Umgestaltungen an den Montagezellen vornehmen. Die Prozeßsimulation bietet die Möglichkeit, das Einsatzverhalten der Montageanlagen trendmäßig zu analysieren, um so kritische Schwachstellen zu orten, ihre quantitative Wirkung auf die gesamte Montage abzuschätzen und durch Strukturveränderungen und Pufferdimensionierung den Wirkungsgrad zu erhöhen.

5.5.1 Bewegungssimulation

Die Bewegungssimulation bietet die Möglichkeit Kollisions- und Erreichbarkeitsstudien durchzuführen sowie Taktzeiten zu erfassen. Es müssen dazu alle montagerelevanten Systemelemente wie Handhabungsgeräte, Transportsysteme, Greifer, Gestelle und natürlich auch die Werkstücke mit einem 3D-CAD-System modelliert sein.

Für eine Bewegungssimulation genügt jedoch nicht eine rein geometrische Modellierung der Objekte. Das geometrische Modell muß um technologische Daten, vor allem die kinematischen Eigenschaften erweitert werden. So werden für Industrieroboter beispielsweise Typ, Tragkraft, Positioniergenauigkeit sowie Zahl, Lage, Geschwindigkeit, Beschleunigungen und Verfahrbereiche der Achsen eindeutig festgelegt. Sonstige Komponenten werden als beweglich oder unbeweglich klassifiziert [110].

Sind diese Voraussetzungen für das mit einem CAD-System zusammengestellte Layout einer Montagezelle gegeben, so kann der Bewegungsablauf des Montierens programmiert und simuliert werden. Durch eine mitlaufende Uhr kann während der Simulation die erforderliche Zeit für einen Bewegungsablauf gemessen und so die zur Zusammenstellung der Montagezellen geschätzte Taktzeit genauer bestimmt werden.

Die interaktive Programmierung mit Bewegungssimulation ist wegen der geforderten Antwortzeiten nur mit Drahtmodellen sinnvoll. Die während des Programmiervorgangs aufgezeichneten Simulationsprotokolle lassen sich aber auch in 3D-Vollkörperdarstellung mit ausgeblendeten verdeckten Kanten am Bildschirm aufzeigen (Bild 63 [110]).

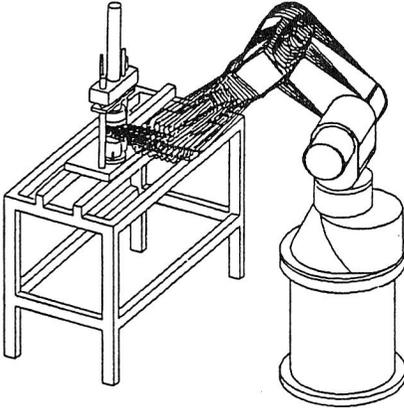


Bild 63: Kollisionsuntersuchung durch Bewegungssimulation
(aus [110])

Die rechenzeitintensive Simulation mit Volumenkörpern liefert dabei nicht nur eine eindrucksvolle Animation von Montagevorgängen, sondern erlaubt auch die automatische Kontrolle von Kollisionen. Dazu werden für jeden Zeitpunkt der vorangegangenen Programmsimulation alle in der Montagezelle vorhandenen Körper auf mögliche Überschneidungen untersucht.

5.5.2 Prozeßsimulation

Die Prozeßsimulation dient zur Optimierung des Montageanlagenlayouts in Hinblick auf Struktur und Pufferdimensionierung.

Zur Durchführung der Simulation muß zunächst ein abstraktes Modell gebildet werden (Bild 64 [112]).

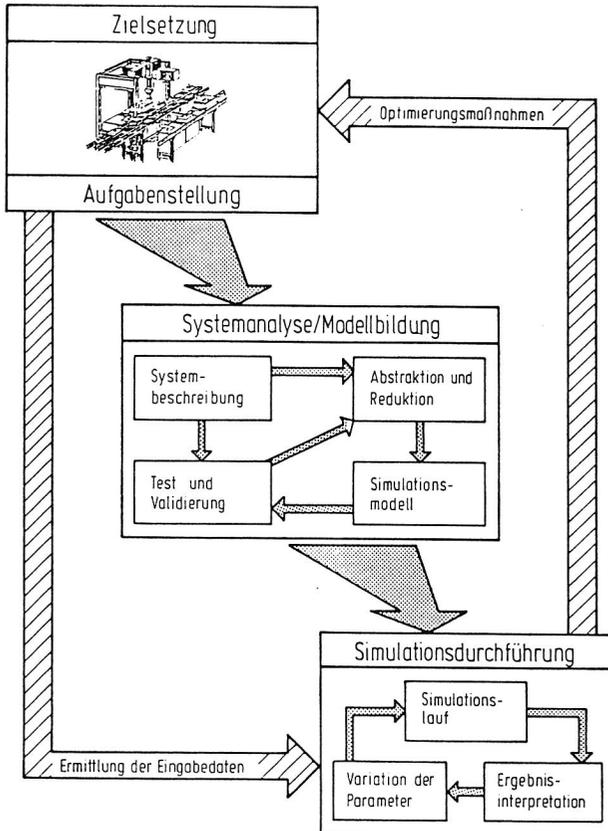


Bild 64: Vorgehensweisen bei Prozeßsimulationsstudien [112]

Die Modellbildung erfolgt in den drei Schritten:

- Abgrenzung gegen die Umwelt
- Bestimmung der Systemelemente und ihrer Eigenschaften
- Definition der Systemstruktur.

Durch die Abgrenzung gegen die Umwelt wird der Systemrahmen für die Simulation festgelegt. So kann die gesamte Montageanlage, bestehend aus mehreren Zellen mit Verbindungselementen, oder auch nur eine einzelne in ihre Bestandteile zerlegte Zelle Gegenstand der Simulation sein.

Für die Simulation von Montagesystemen können zwei elementare Systemelemente unterschieden werden. Zum einen sind dies Stationen. Sie beschreiben die zeitverbrauchenden Elemente wie Handhabungsgeräte, Industrieroboter, Bearbeitungsmaschinen oder auch Handarbeitsplätze.

Zum anderen werden Verkettungseinrichtungen wie Transport- und Stautrecken sowie Magazine, die als Verbindungselemente zwischen den Stationen wirken, unter der Bezeichnung Puffer zusammengefaßt.

Die Stationen werden beschrieben durch ihre Taktzeit, ihre Verfügbarkeit sowie die Häufigkeit und Dauer von stochastischen oder deterministischen Störungen und Ausfällen. Puffer werden beschrieben durch ihre Kapazität und Durchlaufdauer.

Zur Abbildung des Montagesystems können die beiden Systemelemente beliebig kombiniert und so Strukturen aufgebaut werden. Je nach Art der Abhängigkeit von Stationen bezüglich Materialver- und -entsorgung unterscheidet man zwischen [113]:

- loser Verkettung
- elastischer Verkettung
- starrer Verkettung.

Bei loser Verkettung sind die einzelnen Stationen völlig unabhängig voneinander. Störungen werden durch "unendlich" große Puffer aufgefangen (Bild 65).

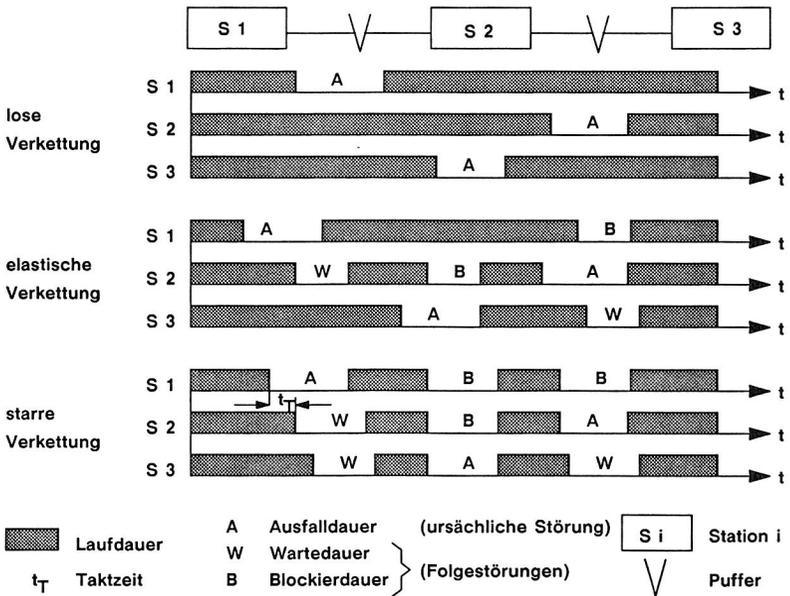


Bild 65: Abhängigkeit der Folgestörungen von der Verkettung (nach [113])

Bei elastischer Verkettung führt ein Ausfall zu einer Folgestörung bei der nachfolgenden Station, wenn der zwischengelagerte Arbeitsvorrat die Ausfalldauer nicht überbrücken kann, und bei der vorangehenden Station, wenn kein Werkstück mehr zwischengelagert werden kann.

Bei starrer Verkettung sind die Stationen starr miteinander verbunden. Fällt eine Station aus, werden vorgeschaltete Stationen blockiert, da sie ihre Werkstücke nicht weitergeben können (Blockierdauer). Die nachgeschalteten Stationen warten dagegen auf nachrückende Werkstücke (Wartedauer).

Bevor die eigentliche Simulation durchgeführt werden kann, muß das erstellte Modell noch validiert werden. Es muß geprüft werden, ob das Modell das Verhalten des realen Systems ausreichend repräsentiert [111].

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 66: Ergebnisdaten der Prozeßsimulation [107]

Die Ergebnisse eines Simulationslaufs sollen Aussagen zur Beurteilung der Stationen und Puffer ermöglichen. Diese dienen dazu, für eine erneute Simulation Veränderungen am Layout vorzunehmen und so das Layout schrittweise zu optimieren. Welche Kennzahlen ein Simulationslauf dazu liefert, ist in Bild 66 zusammengestellt [111].

5.6 Wirtschaftlichkeitsrechnung

Die Entscheidung, ob eine gewählte Lösung für eine Montageaufgabe realisiert werden soll, kann in der Regel erst nach der Durchführung einer Wirtschaftlichkeitsrechnung gefällt werden.

Die Beurteilung der Wirtschaftlichkeit einer Investition in eine automatisierte Montageanlage beruht meist auf dem Vergleich der Kosten, Rentabilität, Amortisationszeiten oder Grenzstückzahl für alternative Systemlösungen, zu denen in der Regel auch die manuelle Montage zählt.

Zur Berechnung können statische und dynamische Verfahren eingesetzt werden. Die statischen Verfahren arbeiten mit Durchschnittswerten, während die dynamischen Verfahren zeitliche Unterschiede beim Anfall von Kosten und Erträgen während der Nutzungsdauer des Investitionsobjektes berücksichtigen.

Der für die Investition in eine Montageanlage erforderliche Kapitaleinsatz setzt sich zusammen aus Planungs- und Entwicklungskosten, Beschaffungskosten für die Komponenten der Anlage wie Geräte, Gestelle, Peripherie, Steuerungen usw. sowie Kosten für die Installation und die Einführung der Anlage (Bild 67).

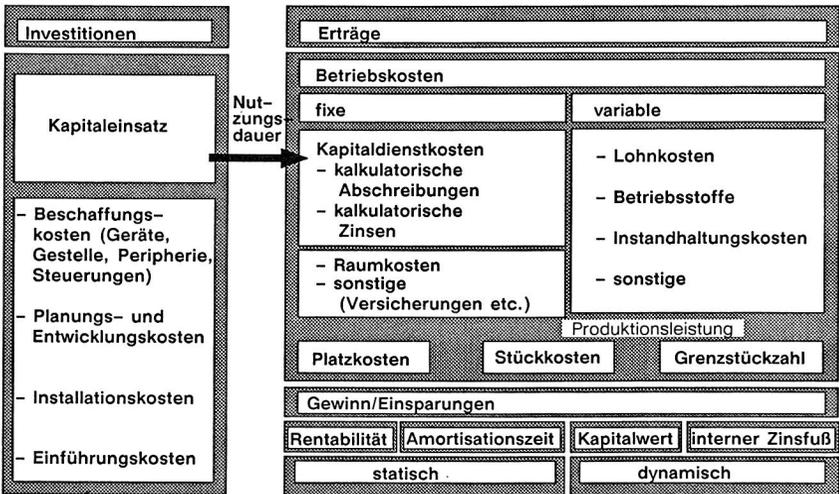


Bild 67: Ablauf von Investitionsrechnungen

Die Betriebskosten der Montageanlage lassen sich aufteilen in fixe Kosten, die unabhängig von der produzierten Menge sind, und in variable Kosten, die von jedem Erzeugnis verursacht werden.

Der einmal aufgewendete Kapitaleinsatz, also die Ausgaben für den Aufbau der geplanten Montageanlage, führen bei der beabsichtigten Nutzung der Anlage zu fixen Betriebskosten in Form von Abschreibungen und Kapitalbindungskosten (kalkulatorischen Zinsen). Weitere fixe Kosten sind die Raumkosten und sonstige Kosten für Versicherungen etc.

Die wichtigsten variablen Betriebskosten sind die Lohnkosten. Sie lassen sich ermitteln aus Vorgabezeiten für die erforderlichen Verrichtungen und einem Lohnfaktor, der abhängig ist von der Lohngruppe des Bedienpersonals. Weitere variable Kosten fallen für Energie, Betriebsstoffe und Instandhaltung an. Die Summe der fixen und variablen Kosten ergibt die Platzkosten [84]. Bei einfacher Kostenvergleichsrechnung dient die Gegenüberstellung der Platzkosten für alternative Investitionen als Entscheidungskriterium.

Sollen Alternativen mit unterschiedlichen Produktionsleistungen verglichen werden, so müssen die Stückkosten verglichen werden [114]. Für solche Anlagen mit unterschiedlichen Kapazitäten läßt sich auch eine Produktionsmenge bestimmen, ab der sich der Einsatz der bei niedrigerer Auslastung teureren Anlage lohnen würde. Die Grenzstückzahl ergibt sich aus der Division der Differenzen der fixen und variablen Kosten.

Werden neben den Kosten auch die Erträge berücksichtigt, so kann der Gewinn aus einer Investition ermittelt werden.

Die Rentabilität einer Investition ergibt sich aus dem Verhältnis des durchschnittlichen Gewinns bzw. der Kostenersparnis zum durchschnittlich dafür eingesetzten Kapital. Die Amortisationszeit sagt aus, nach welchem Zeitraum das für die Investition eingesetzte Kapital wieder in das Unternehmen zurückgeflossen ist. Sie ergibt sich aus dem Verhältnis des Kapitaleinsatzes

zum durchschnittlichen jährlichen Rückfluß [115].

Kostenvergleich, Rentabilitäts- und Amortisationszeitermittlung zählen zu den statischen Verfahren der Wirtschaftlichkeitsrechnung. Die wichtigsten dynamischen Verfahren sind die Kapitalwertmethode und die interne Zinsfußrechnung [115]. Sie berücksichtigen zeitliche und wertmäßige Unterschiede im Anfall der Kosten und Erträge durch die Anwendung der Zinseszins- und Rentenrechnung.

Die Kapitalwertmethode ähnelt der Kostenvergleichsrechnung. Der Kapitalwert ist der auf den Kalkulationszeitpunkt diskontierte Nettogewinn einer Investition. Die interne Zinsfußmethode geht nicht von einer gegebenen Mindestverzinsung aus, sondern sucht den Diskontierungszinsfuß der zu einem Kapitalwert von Null führt.

Für genauere Darstellungen der Verfahren der Wirtschaftlichkeit sei auf einschlägige Lehrbücher [114,115] verwiesen.

6. Entwurf eines konzeptionellen Schemas für die Planung automatisierter Montageanlagen

Das konzeptionelle Schema enthält die Beschreibung aller in einer Datenbank zu verwaltenden Daten. Es stellt den zentralen Bezugspunkt für alle Anwendungen auf den Datenbestand dar.

Die folgenden Ausführungen konzentrieren sich darauf, aus den Ergebnissen der Bedarfsanalyse, durch Begriffsrekonstruktion ein konzeptionelles Schema in einem semantischen Datenmodell zu entwickeln mit dem der gesamte Planungsablauf für automatisierte Montagesysteme unterstützt werden kann.

Während in den Kapiteln 4 und 5 die Montage bzw. die Montageplanung fach- und umgangssprachlich beschrieben wurde, geht es nun darum, die wesentlichen Begriffe der Montageplanung zu rekonstruieren und präzise festzulegen. Zur Darstellung der Ergebnisse dieser Rekonstruktion wird das in Abschnitt 3.3.2 vorgestellte semantische Modell genutzt, das einer nach [21] modifizierten Form des Entity-Relationship-Modells entspricht.

Bei der Erläuterung der semantischen Schemata werden die Objekttypen, Attribute und Beziehungen in der Form

OBJEKTYP Attribut Beziehung

geschrieben. Eine eingehende Beschreibung sämtlicher Objekttypen, ihrer Attribute und das Zustandekommen der Beziehungen würde den Rahmen dieser Arbeit sprengen. Die Erklärungen beschränken sich deshalb auf wesentliche Zusammenhänge, die aus der Darstellung der Schemata nicht ohne weiteres ersichtlich sind. Aus dem gleichen Grund wird auch auf eine explizite Angabe von Attributlisten und Integritätsbedingungen verzichtet. Genauere Aufstellungen der Attribute und Integritätsbedingungen finden sich in [116,117,118,119].

Vor dem eigentlichen Schemaentwurf werden zunächst die anfallen-

den Daten grob klassifiziert und die unterschiedlichen Anforderungen an die Modellierung der verschiedenen Datenklassen beschrieben. Im Anschluß daran wird ein allgemeines Strukturmodell zur Beschreibung technischer Produkte unter Berücksichtigung von Varianten und Versionen entwickelt, welches in die nachfolgenden Schemaentwürfe integriert wird.

6.1. Klassifizierung der Montagedaten

Datenbankmanagementsysteme im technischen Bereich eines Produktionsbetriebes speichern ein Modell technischer Objekte. In der Montage lassen sich zwei Klassen technischer Objekte unterscheiden; die Produkte und die Montageanlagen mit denen die Produkte montiert werden.

Das Montagemodell repräsentiert alle für die Montageplanung relevanten Daten dieser beiden Objektklassen. Es beschreibt alle strukturellen, geometrischen, physikalischen und technologischen Eigenschaften von Produkten und Montageanlagen. Dementsprechend kann das Montagemodell nach [21] gegliedert werden in

- das Strukturmodell,
- das physikalische Modell,
- das technologische Modell und
- das geometrische Modell.

Das Strukturmodell beschreibt die strukturelle Gliederung technischer Objekte in Unterelemente. Typisch für das Strukturmodell ist die hierarchische Gliederung. Produkte setzen sich aus Baugruppen zusammen, die ihrerseits wieder aus Unterbaugruppen bestehen. Diese Unterteilung setzt sich fort bis zu den Einzelteilen. Entsprechend zerfallen Montageanlagen in Montagezellen und Komponenten. Im Strukturmodell werden auch Ähnlichkeiten sowie Varianten und Versionen berücksichtigt.

Das physikalische Modell enthält die physikalischen Eigenschaften der technischen Objekte, wie Massen, Volumina, Schwerpunkte oder Oberflächen.

Das technologische Modell beinhaltet Toleranzangaben für Elemente des Produktes sowie für die Montage des Produktes erforderliche Beschreibungen (Montagepläne, IR-Programme). Auf Seiten der Montageanlage beinhaltet das technologische Modell Angaben über die Funktionen, die deren Komponenten ausführen können. Das technologische Modell bildet die eigentliche Schnittstelle zwischen den beiden technischen Objekten Produkt und Montageanlage.

Im geometrischen Modell wird die Geometrie und Topologie der technischen Objekte beschrieben. Es lassen sich drei Beschreibungsformen unterscheiden [21]. Das 3D-Modell enthält eine eindeutige dreidimensionale Darstellung der Objektinformation auf der Basis des in Bild 51 vorgestellten allgemeinen konzeptionellen Schemas für CAD-Systeme. Zeichnungen enthalten die aus der dreidimensionalen Beschreibung abgeleiteten Projektionen und Ansichten, die um zusätzliche Elemente wie Bemaßungen, Toleranzangaben und Beschriftungen ergänzt wurden. Metafiles sind Plotfiles die nur unstrukturierte Zeichnungsinformation enthalten.

Das strukturelle, das physikalische und das technologische Modell werden detailliert im Montagemodell beschrieben. Für das geometrische Modell wird die Integrationsstufe 4 (Bild 26, Abschnitt 3.4.2) angestrebt. Dies bedeutet, daß für die Verwaltung der Geometriedaten in der Datenbank nur geometrische Objekte auf der Ebene der Figuresynthese des allgemeinen konzeptionellen Schemas für ein CAD-System nach Bild 49 berücksichtigt und für diese die drei genannten Beschreibungsformen (3D-Modell, Zeichnung, Metafile) als Ganzes in das Schema aufgenommen werden, indem ein Attribut auf die Namen der Dateien verweist, in denen die zugehörigen CAD-Beschreibungen gespeichert sind. Es kann dann über die Datenbank nur auf Geometriebeschreibungen für Objekte vom Typ Body, Instance oder Assembly zuge-

griffen werden. Geometriedaten niedrigerer Komplexität sind nur direkt über das CAD-System ansprechbar.

Eine zu der vorgestellten orthogonale Klassifikation ist die in [17] getroffene Unterscheidung zwischen

- informationellen Daten und
- operationalen Daten.

Die informationellen Daten, häufig auch Katalogdaten genannt, sind statischer Art. Sie beinhalten die allgemeinen Informationen, die der Montageplaner zur Durchführung der Montageplanung benötigt. Diese Daten sind weitgehend stabil und bedürfen nur in größeren Zeitabständen einer Aktualisierung, die nur von speziell dazu autorisierten Personen vorgenommen werden darf. Die operationalen Daten dagegen stellen die im Verlauf der Montageplanung anfallenden Daten dar. Der Montageplaner nutzt während einer Planungsstufe immer wieder Ergebnisse früherer Planungsschritte, erzeugt neue Daten und modifiziert und ergänzt aufgrund dabei gewonnener Erkenntnisse die Daten vorangegangener Planungsschritte.

Der wesentliche Unterschied zwischen informationellen und operationalen Daten liegt also darin, daß auf erstere vom jeweiligen Bearbeiter nur lesend zugegriffen wird, während die operationalen Daten laufend verändert und erweitert werden (siehe auch Bild 2).

Bei den informationellen Daten kann zwischen echten Katalogdaten und sonstigen informationellen Daten unterschieden werden. Für diese "Nicht-Katalogdaten" ist die Zugehörigkeit zu den informationellen oder operationalen Daten nicht eindeutig, sie hängt vom jeweiligen Bearbeiter ab. So sind für den Produktkonstrukteur die Daten über das Produkt operationale Daten während sie für den Montageplaner rein informativer Art sind und er keine Veränderungen an ihnen vornehmen darf.

Die Unterscheidung zwischen informationellen und operationalen Daten ist fundamental für die Festlegung von Zugriffsrechten.

6.2 Konzeption eines allgemeinen Strukturmodells unter Berücksichtigung von Varianten und Versionen

Wie im vorhergehenden Abschnitt schon angesprochen, beschreibt ein Strukturmodell die strukturelle Gliederung technischer Objekte in Unterelemente. Sowohl die Produkte, als auch die Montageanlagen, mit denen die Produkte montiert werden, lassen sich hierarchisch aufgliedern. Um Wiederholungen zu vermeiden, wird in den folgenden Ausführungen nur die Untergliederung von Produkten in Baugruppen und Einzelteile angesprochen. Die an den Produkten angestellten Betrachtungen lassen sich natürlich auch auf Montageanlagen übertragen, die in Montagezellen und Komponenten bzw. Geräte und deren Bestandteile untergliedert werden können.

Die Erzeugnisstruktur technischer Produkte wird üblicherweise durch Stücklisten dargestellt.

Prinzipiell setzen sich Stücklisten zusammen aus einem Stücklistenkopf, der die Informationen über das Enderzeugnis bzw. die Baugruppe aufnimmt (bei auftragsgebundenen Stücklisten zusätzlich die Auftragsdaten) und beliebig vielen Positionszeilen, die die Angaben über die verwendeten Gruppen und Teile beinhalten, aus denen der im Stücklistenkopf beschriebene Gegenstand besteht.

Die Datenhaltung erfolgt zumeist durch eine Aufteilung in einen Stammdatensatz und einem Strukturdatensatz. Im Stammdatensatz werden die Teile mit ihren Eigenschaften erfaßt, wobei ein Teil Enderzeugnis, Baugruppe oder Einzelteil sein kann. Im Strukturdatensatz wird der strukturelle Aufbau eines Enderzeugnisses bzw. einer Baugruppe verwaltet, d.h. es wird festgehalten, welche Teile in welcher Anzahl in ein Produkt eingehen. Werden

auch Varianten und Versionen von Erzeugnissen derartig verwaltet, so müssen geeignete Zusatzmechanismen gefunden werden, um die Mehrfachspeicherung gleicher Informationen zu vermeiden oder zumindest zu reduzieren.

Die nachfolgenden Betrachtungen dienen zur Ableitung einer vorteilhaften Erzeugnisdarstellung unter Berücksichtigung von Alternativen, Varianten und Versionen.

Zunächst werden die Begriffe Variante, Version und Alternative gegeneinander abgegrenzt und erläutert und anschließend ein semantisches Schema zur Produktbeschreibung unter Einbeziehung versionenbehafteter Varianten entwickelt.

6.2.1 Die Entstehung von Alternativen, Varianten und Versionen

Die Begriffe Alternative, Variante und Version kennzeichnen sehr unterschiedliche Mechanismen der Produktbeschreibung. Diese verschiedenen Produktbeschreibungen entstehen im Laufe des Lebenszyklus eines Produktes.

Geht man davon aus, daß die Entwicklung eines Produktes eine Neukonstruktion darstellt, so wird der Konstrukteur während den in Abschnitt 4.2.4 geschilderten Konstruktionsphasen mehrere Alternativen zur Lösung der Konstruktionsaufgabe verfolgen, d.h. er fertigt alternative Beschreibungen des Produktes auf unterschiedlichen Abstraktionsebenen an.

In den meisten Fällen wird ein Produkt nicht nur für einen Anwendungszweck konstruiert. Verschiedene Kunden setzen das gleiche Produkt für unterschiedliche Zweckfunktionen ein, z.B. als Komponenten im Rahmen größerer Einheiten und fordern daher Anpassungen des Produktes an ihre speziellen Wünsche. So entstehen Funktionsvarianten (Typen), wie z.B. die in Bild 54 dargestellten Varianten eines Schalters, Gestaltvarianten (Baureihen), die z.B. unterschiedliche Leistungsanforderungen durch

Größenklassen berücksichtigen, und Detailvarianten, die sich z.B. nur in der Farbe, dem Werkstoff oder der Ausführung eines Flansches unterscheiden.

Häufig zeigt sich bei der Vorfertigung und Montage des Produktes, bzw. schon bei deren Planung, daß Änderungen an den Konstruktionsunterlagen erforderlich sind, um Fehler zu beheben oder kostengünstiger fertigen zu können. Auch Mängel während des Einsatzes eines Produktes oder geänderte Kundenwünsche bedingen immer wieder eine Überarbeitung, so daß fortwährend neue Versionen eines Produktes entstehen.

Zusammenfassend können die drei Begriffe wie folgt charakterisiert werden:

Alternativen entstehen im Verlauf der Lösungsfindung und dienen dazu, durch die Gegenüberstellung möglichst vieler Lösungsmöglichkeiten für eine gegebene Aufgabenstellung, eine möglichst optimale Lösung aufzufinden. Sie sind nur für die Produktentwicklung (Konstruktion) von Interesse.

Varianten entstehen durch Diversifikation, d.h. durch die Anpassung des Produktes an unterschiedliche Kundenwünsche. Sie bestehen wie die Alternativen zeitlich nebeneinander, müssen jedoch sowohl bei der Produktentwicklung (Konstruktion) als auch bei der Produktfertigung und dem Produkteinsatz berücksichtigt werden.

Versionen entstehen durch Änderungen der Produktbeschreibung. Sie dokumentieren die Entwicklungsstufen eines Produktes und folgen chronologisch aufeinander. Es werden die aktuellen Versionen der verschiedenen Produktvarianten gefertigt, es muß jedoch auch der Zugriff auf frühere Versionen gewährleistet sein, z.B. für die Ersatzteilerhaltung.

Während also eine neue Version eines Produktes ihren Vorgänger

ablöst und ersetzt bleibt die Variante eines Produktes, von dem eine neue Variante abgeleitet wurde, neben der neuen bestehen.

6.2.2 Entwicklung eines semantischen Schemas zur Verwaltung von Varianten und Versionen

Alternativen können als hypothetische Varianten betrachtet werden [120]. Es wäre folgerichtig sie wie Varianten zu behandeln. Alternativen werden in der Regel jedoch nur in den Phasen der Funktionsfindung, Prinzipiararbeit und Gestaltung während der Konstruktion erstellt und werden bedeutungslos, sobald eine Lösung ausgewählt und detailliert wurde. Sie sind also nur während des Konstruktionsprozesses, bei Neu- und Anpassungskonstruktionen, von Interesse. Für das Stücklistenwesen, das sich durch den ganzen Produktionsbetrieb zieht, vom Einkauf bis zum Vertrieb, sind Alternativen ohne Bedeutung.

Da im Rahmen dieser Arbeit die Produktkonstruktion von untergeordneter Bedeutung ist, werden Alternativen nur für die Montageplanung und nicht für die Produktkonstruktion, berücksichtigt. Der Schwerpunkt der folgenden Ausführungen liegt auf der Entwicklung einer geeigneten Darstellungsform für die Verwaltung von Erzeugnisstrukturen versionenbehafteter Varianten. Zweckmäßigerweise soll die heute allgemein übliche Trennung von teilespezifischer Information (Teilestammdaten) und strukturspezifischer Information (Strukturdaten) beibehalten werden.

Der von Vaszonyi vorgeschlagene Gozinto-Graph vermittelt für in ihrem Umfang beschränkte Erzeugnisstrukturen eine anschauliche Darstellung der Trennung von teile- und strukturspezifischer Information und der damit erreichbaren Redundanzfreiheit. Die Knoten des Graphen repräsentieren Teile (worunter Erzeugnisse, Baugruppen und Einzelteile zu verstehen sind), die gerichteten und mit Mengenangaben bewerteten Kanten zeigen die Strukturierung [121].

Bild 68 zeigt die Erzeugnisstrukturen der drei in Bild 54 vorgestellten Varianten eines Schalters (E1 = Öffner, E2 = Schließer und E3 = Öffner/Schließer) als Gozinto-Graph, wobei variante Teile sich durch die an die Kurzbezeichnung angehängten Ziffern unterscheiden. Dieses relativ einfache Beispiel soll als Grundlage für die weiteren Betrachtungen dienen.

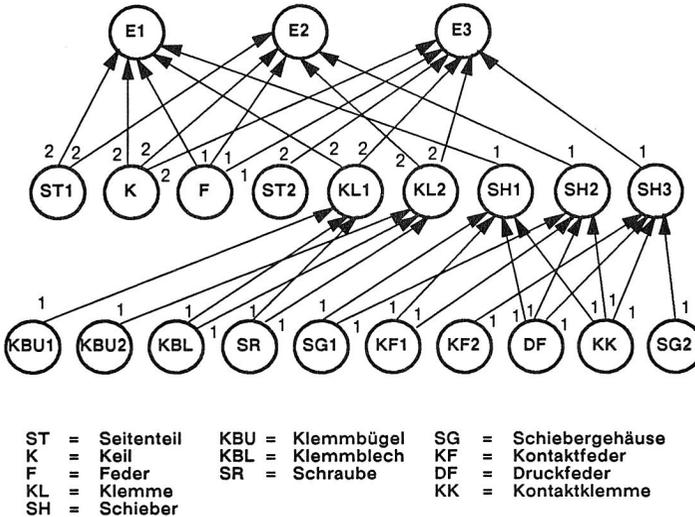


Bild 68: Gozinto-Graph dreier Varianten eines Schalters

Die Strukturdaten geben Auskunft über den strukturellen Zusammenhang eines Erzeugnisses oder einer Baugruppe, also über den eigentlichen Stücklistenaufbau. Es ist zu beachten, daß jedes Teil, das als Stücklistenposition vorkommt, auch einen Stammdatensatz besitzen muß. Es können zwar Teilestammsätze bestehen, deren Teilenummer nicht in einen Strukturdatensatz eingehen, der umgekehrte Fall ist jedoch nicht möglich (Referentielle Integritätsbedingung).

In der Literatur [104,122,123,124] werden für die Beschreibung der Struktur varianter Erzeugnisse vier Variantenarten unterschieden:

1. Die "Mußvariante"

Es steht eine Auswahl von Ausführungen zur Verfügung, es muß aber eine im Erzeugnis enthalten sein, um die Funktion zu gewährleisten (z.B. Kontaktfeder des Schalters mit Gold- oder Silberkontakten).

2. Die "Kannvariante"

Dies sind zusätzliche Teileumfänge, die unabhängig voneinander im Erzeugnis vorhanden sein können. Die Funktion des Erzeugnisses bleibt in jedem Falle erhalten (z.B. Arretierung des Schiebers in Schaltstellung).

3. Die "Mengenvariante"

Die Teile können in unterschiedlicher Menge in ein Erzeugnis eingehen (z.B. zwei oder vier Klemmen in einen Schalter).

4. Die "Dispositionsvariante"

Dies sind Teile/Baugruppen, die untereinander bedingungslos austauschbar sind, aber unterschiedlicher Beschaffungspolitik unterliegen (z.B. Schrauben verschiedener Hersteller).

Die "Dispositionsvariante" erfordert keine gesonderte Stückliste, sie kann über die Lieferantenbeziehung dargestellt werden und wirkt sich nicht auf die Stücklistenstruktur aus.

In bezug auf die Stücklistenstruktur lassen sich die in Bild 69 aufgeführten Möglichkeiten der Speicherung von Variantenstücklisten unterscheiden.

Liegt eine begrenzte, überschaubare Anzahl von Varianten vor, so können diese benummert und auftragsneutral gespeichert werden. Wenn die Zahl der Kombinationsmöglichkeiten sehr groß

wird, so ist es nicht mehr sinnvoll, alle Varianten explizit abzuspeichern. In diesem Fall werden auftragsbezogene Stücklisten zusammengestellt. Produkte mit auftragsneutralen Stücklisten können direkt anhand der Stückliste identifiziert werden, während solche mit auftragsbezogenen Stücklisten nur in Zusammenhang mit einem Auftrag identifiziert werden können.



Bild 69: Überblick über die verschiedenen Arten von Variantenstücklisten (nach [122])

Wird bei der auftragsneutralen Variantenorganisation für jede Variante eine selbständige Variantenstückliste gespeichert, so lassen sich bei den bekannten Verfahren zur Speicherplatzeinsparung zwei Prinzipien erkennen.

Das eine ist die Zusammenfassung von Teilen zu Pseudo-Baugruppen mit dem Ziel, Struktursätze einzusparen. Es werden sogenannte Gleichteile, d.h. Teile, die in alle Erzeugnisse der Variantenreihe mit konstanter Anzahl eingehen, in einer Pseudogruppe zusammengefaßt. Für jede Variante muß statt den Beziehungen zu allen Gleichteilen nur noch die Beziehung zur Pseudogruppe realisiert werden. Die Pseudogruppe erhält eine spezielle Kenn-

zeichnung, damit sie bei der Stücklistenauflösung nicht als konstruktiv vorhandene Baugruppe interpretiert wird [124]. Eine Speicherersparnis ist nur dann gegeben, wenn die Zahl der Gleichteile hoch ist.

Beim zweiten Prinzip wird für eine Grundauführung des Erzeugnisses eine komplette Stückliste gespeichert. Für die Varianten werden zu dieser Grundauführungsstückliste sogenannte Plus-Minus-Stücklisten erstellt, in denen die Teile, die im Vergleich mit der Grundauführung zusätzlich hinzukommen (Plusteile) und die Teile, die gegenüber der Grundauführung entfallen (Minusteile), zusammengefaßt werden. Variantenstücklisten mit Angabe der Zusatz- und Entfallteile sind nur dort vorteilhaft, wo sich die Varianten nur in wenigen Teilen vom Grundtyp unterscheiden, der Kreis dieser Teile aber nicht genau festgelegt ist [122]. Die Verarbeitung von Minuspositionen erfordert einen erhöhten programmtechnischen Aufwand, deshalb ist diese Form der Variantenspeicherung bei maschineller Datenverarbeitung weniger häufig anzutreffen als die Speicherung von Gleichteilestücklisten, die letztlich nur eine vereinfachte Version der Plus-Minus-Varianten darstellen.

Während bei dem Gleichteil- und bei dem Plus-Minus-Verfahren noch für jede Variante ein Teilstammsatz und die zugehörigen Struktursätze geführt werden, beruht das Prinzip der Mehrfachstückliste auf der Zusammenfassung der Varianten zu einem Pseudoerzeugnis und der Erweiterung der Struktursätze um zusätzliche Mengenfelder. Die Struktursätze enthalten pro Variante ein Mengengebiet, in dem angegeben wird, in welcher Anzahl ein Teil in die Variante eingeht.

Dieses Verfahren ist in zahlreichen Modifikationen weit verbreitet [121]. Wenn Teile in einer jeweils gleichen Anzahl oder gar nicht in eine Variante eingehen, so liegen reine Strukturvarianten vor. Für Strukturvarianten genügt es, ein Mengengebiet einzuführen, das die feste Anzahl nennt und eine Angabe, ob das Teil in die Variante eingeht oder nicht. Reine Mengenvarianten unter-

scheiden sich nur durch unterschiedliche Mengen, die bei einzelnen Varianten benötigt werden. Liegen sowohl Struktur- als auch Mengenvarianten vor, so ist eine Kombination der beiden Speicherungsarten möglich, wie sie beispielsweise in [104] ausführlich dargelegt ist.

Je mehr ein solches Verfahren auf Speicherplatzeinsparung ausgelegt ist, desto mehr werden die Erzeugnisstrukturen verschleiert, der Änderungsdienst erschwert und die Auflösung zeitaufwendig [121].

Das dritte Prinzip auftragsneutraler Variantenstücklisten besteht in der Aufzählung der Varianten in einer Sammelstückliste. Es werden neben den Gleichteilepositionen auch die möglichen Variantenpositionen aufgezählt. Die Variantenpositionen erhalten zur Unterscheidung von den in jedem Fall benötigten Positionen ein Variantenkennzeichen, bei Mengenvarianz wird das Mengenfeld nicht ausgefüllt. Die Zusammenstellung einer konkreten Variante erfolgt durch Auswahl.

Ändern sich die Teilekombinationen von Auftrag zu Auftrag oder sind die Varianten kundenbezogen und haben nur eine geringe Wiederholhäufigkeit, so lohnt es sich nicht, die möglichen Varianten in einer neutralen Erzeugnisstruktur abzuspeichern.

Teilweise ergibt sich eine fast unbegrenzte Anzahl von Kombinationsmöglichkeiten, für die eine neutrale Stücklistenorganisation ebenfalls nicht mehr sinnvoll ist. In diesen Fällen werden aus verschiedenen neutralen Einzelstücklisten für Baugruppen auftragsbezogene Variantenstücklisten zusammengestellt, die über die Auftragsnummer identifiziert werden können.

Die Auswahl der gespeicherten Baugruppen erfolgt über Entscheidungstabellen. Damit ist aber die Erzeugnisstruktur inhomogen festgelegt: Die Erzeugnisstruktur der obersten Stufe wird in Entscheidungstabellen, die der weiteren Stufen in einer der

neutralen Stücklistenorganisationsformen geführt [121].

Eine weitere Methode zur auftragsbezogenen Variantenbehandlung wird in [121] vorgestellt. Bei dieser Methode, im folgenden Knotentyp-Methode genannt, wird nicht die Erzeugnisstruktur eines einzelnen Erzeugnisses beschrieben, sondern die Struktur der in einer Variantenreihe enthaltenen Erzeugnisse zusammengefaßt. Sie basiert auf einer grammatikähnlichen Beschreibung der Variantenreihe durch sogenannte Variantenausdrücke, die die Gestalt arithmetischer Ausdrücke besitzen und in Anlehnung an die Boolesche Algebra umgeformt werden können. Auf diesen theoretischen Aspekt wird hier nicht näher eingegangen, er kann in [121] nachgelesen werden.

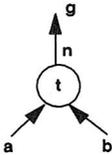
Neben dem Teileknoten des Gozinto-Graphen werden bei dieser Methode drei weitere Typen von Knoten zugelassen: Konjunktivknoten, Alternativknoten und leerer Knoten (Bild 70). Die Teileknoten können als reale Knoten, die anderen drei als virtuelle Knoten aufgefaßt werden.

Ein Teileknoten, dargestellt als Kreis, bestimmt ein Teil, d.h. ein Erzeugnis, eine Baugruppe oder ein Einzelteil. Die Anzahl, mit der das Teil in ein übergeordnetes Teil eingeht, ist durch die der Kante zugeordnete Mengenangabe gegeben.

Ein Konjunktivknoten, dargestellt als Dreieck mit der Spitze nach oben (Δ), bedeutet, daß alle durch die direkt untergeordneten Knoten unmittelbar oder mittelbar bestimmten Teile in das Erzeugnis eingehen. Mit einem solchen Knoten lassen sich Pseudogruppen repräsentieren. Der zugehörige Variantenausdruck entspricht einem logischen UND.

Ein Alternativknoten, dargestellt als Dreieck mit der Spitze nach unten (∇), kennzeichnet eine Verzweigung. Um ein zulässiges Erzeugnis zu bestimmen, muß genau eine der auf den Alternativknoten führenden Kanten ausgewählt werden. Nur die durch

den dieser Kante untergeordneten Knoten unmittelbar oder mittelbar bestimmten Teile gehen in das Erzeugnis ein. Durch diesen Kontentyp lassen sich "Mußvarianten" und "Mengenvarianten" darstellen. Der Variantenausdruck entspricht einem logischen exklusiven ODER.

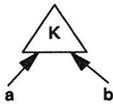


Teileknoten

" t geht n mal ein in g "

" t besteht aus a und b "

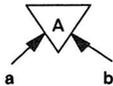
$$t^n$$
$$t = a * b$$



Konjunktivknoten

" a und b müssen hinzu genommen werden "

$$(a * b)$$



Alternativknoten

" Entweder a oder b muß hinzugenommen werden " (exklusives Oder)

$$(a + b)$$



Leerer Knoten

" kein Teil "

$$\Lambda$$

Bild 70: Die vier Knotentypen der Knotentyp-Methode (nach [121])

Der leere Knoten, dargestellt als Kreis mit dem Zeichen Λ in der Mitte, dient ausschließlich dazu, die Wahlmöglichkeit "entfällt", einen Alternativknoten direkt untergeordnet, zuzulassen. Das ist bei "Kannvarianten" der Fall, also bei Teilen, die zu einem Erzeugnis hinzu genommen werden können, aber nicht hinzu genommen werden müssen.

In der Regel werden auch in einem Betrieb, der auftragsbezogen, d.h. kundenspezifisch, variante Enderzeugnisse fertigt, auf-

tragsneutrale, d.h. kundenneutrale, Stücklisten für Vormontagegruppen benötigt.

Bild 71 zeigt die Variantenstrukturen der in Bild 68 explizit als Gozinto-Graph dargestellten drei Schaltervarianten nach der Knotentyp-Methode auf auftragsneutraler und auftragsbezogener Ebene.

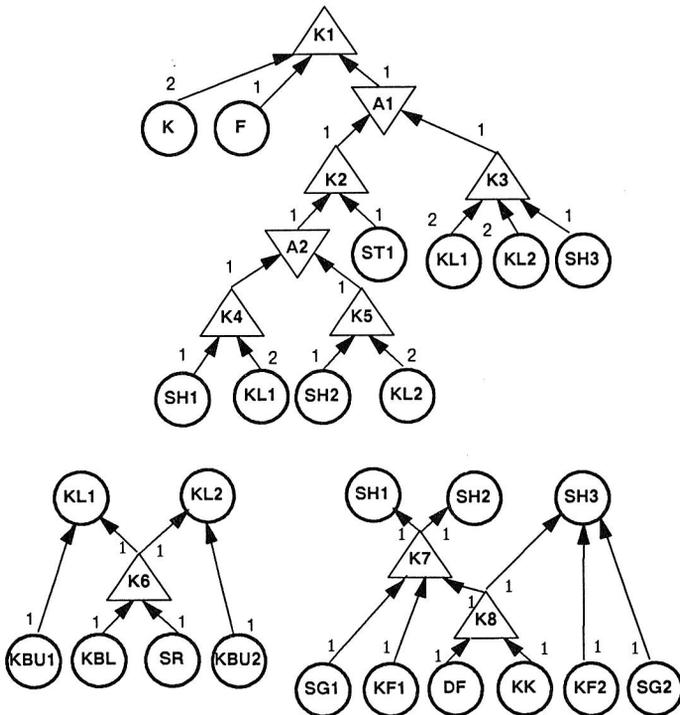


Bild 71: Varianten eines Schalters nach der Knotentyp-Methode

Mit den vier Knotentypen kann die Struktur jeder beliebigen Variantenreihe beschrieben werden, sowohl auftragsbezogen als auch auftragsneutral.

Alternativknoten dürfen nur für auftragsbezogene Variantenreihen eingesetzt werden und sind nur sinnvoll, soweit es sich um Varianten handelt, die im Struktursatz erfaßt werden.

Variante Erzeugnisse, die sich nur in der Art ihrer Zusammensetzung jedoch nicht in den in sie eingehenden Komponenten unterscheiden, lassen sich in einer Darstellung mit Alternativknoten nicht auseinanderhalten. Dies trifft beispielsweise auf die Schieber SH1 und SH2 für die Schaltervarianten Öffner und Schließer nach Bild 54 zu, die sich aus den gleichen Einzelteilen zusammensetzen und nur in der Drehrichtung, in der die Kontaktfeder eingebaut wird unterscheiden. Für derartige Varianten muß eine auftragsneutrale Variantendarstellung erfolgen, in der nur Konjunktivknoten verwendet werden, was einer Variantenstücklistenspeicherung nach der oben angesprochenen Gleichteilmethode entspricht.

Wie weiter oben bei der Beschreibung der Entstehung von Varianten und Versionen festgestellt wurde, bestehen Versionen nicht wie Varianten nebeneinander sondern folgen chronologisch aufeinander; sie beinhalten etwas historisches. Deshalb ist es naheliegend, zur Kennzeichnung von Versionen einen Zeiteintrag zu verwenden, wann eine neue Version gültig wurde.

Nach [125] lassen sich fünf Möglichkeiten der Zeitangaben für die Kennzeichnung von Versionen in Datenbanken unterscheiden:

- 1) (Objekt-Nr.,Gültig-seit, . . .)
- 2) (Objekt-Nr.,Gültig-bis, . . .)
- 3) (Objekt-Nr.,Gültig-seit,gültig-bis, . . .)
- 4) (Objekt-Nr.,Gültig-seit,In-Gebrauch, . . .)
- 5) (Objekt-Nr.,Gültig-bis,In-Gebrauch, . . .).

Bei der ersten Möglichkeit besteht das Problem, daß nicht klar ist, ob das Objekt noch in Gebrauch oder schon längst "tot" ist. Bei der zweiten Darstellungsform ist nicht bekannt, seit wann es das Objekt gibt. Bei der dritten können Lücken in der Versio-

nenfolge auftreten. Praktikabel sind nur die Möglichkeiten vier und fünf, wobei hier der vierten der Vorzug gegeben wird.

Objekte, von denen Versionen existieren, können sowohl reale als auch virtuelle Knoten aus der Variantenstruktur sein, für Strukturen selbst ist jedoch keine Versionendarstellung erforderlich; diese ergibt sich aus den Versionen der Knoten und ggf. durch Aufnahme zusätzlicher, sich bei einer neuen Version verändernder, Kanten in die Strukturdarstellung, wobei die alte Strukturbeschreibung nicht geändert werden darf.

In Bild 72 ist die Versionsfolge der Schiebervarianten SH1 und SH2 der Schalter aus Bild 54 dargestellt, deren Variantenstruktur Bild 71 zeigt. An diesem willkürlich gewählten Beispiel soll die Versionendarstellung erläutert werden.

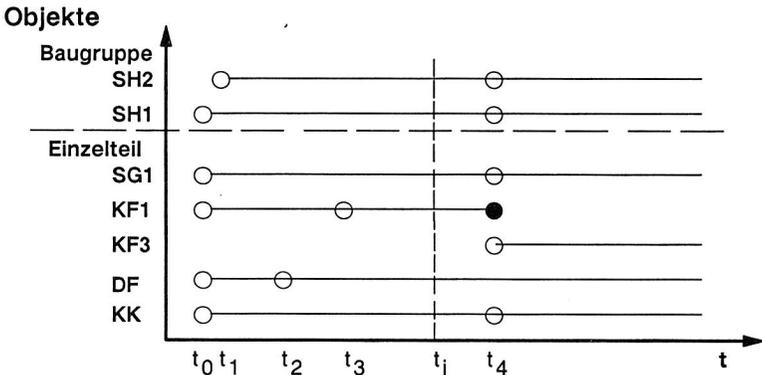


Bild 72: Versionen zweier Schieber SH1 und SH2

Der Schieber SH1 setzte sich zum Zeitpunkt t_0 zusammen aus dem Schiebergehäuse SG1, der Kontaktfeder KF1, der Druckfeder DF und der Kontaktklemme KK.

Zum Zeitpunkt t_1 wurde eine neue Variante SH2 des Schiebers

eingeführt, die sich aus den gleichen Teilen zusammensetzte, nur die Kontaktfeder KF1 wurde umgekehrt montiert, so daß sich bei Betätigung des Schiebers im Schalter ein Kontakt schloß anstatt öffnete.

Ab dem Zeitpunkt t_2 wurde die Druckfeder DF durch eine stärkere ersetzt, was jedoch keine Auswirkung auf die anderen Teile der beiden Schiebervarianten hatte, ebenso wie die Änderung der Kontaktfeder KF1 zum Zeitpunkt t_3 . Es ergaben sich damit auch keine neuen Versionen der Baugruppen.

Eine neue Version einer Baugruppe oder eines Enderzeugnisses entsteht nur durch eine einschneidende Änderung an dem Einzelteil, das das Produkt bestimmt, und nicht bei jeder Änderung eines beliebigen in das Produkt eingehenden Einzelteils. So wird man beim Auto auch nicht von einer neuen Version sprechen, wenn die Schrauben der Kotflügelbefestigung geändert werden, sondern nur dann, wenn eine neue Karosserie eingebaut wird.

Da sich beim Einsatz der Schalter zeigte, daß sich mit der Kontaktfeder KF1, trotz der Änderung zum Zeitpunkt t_3 , keine sichere Kontaktierung erreichen ließ, wurde sie ab t_4 durch zwei schmalere Kontaktfedern KF3 ersetzt. Dies erforderte auch eine Umkonstruktion der Kontaktklemme KK und des Schiebergehäuses SG1. Das Schiebergehäuse SG1 ist das bestimmende Teil der Schieber SH1 und SH2 und deshalb wurden auch diese als neue Versionen gekennzeichnet. Die Kontaktfeder KF3 konnte nicht als neue Version der Kontaktfeder KF1 eingeführt werden, da sie jeweils zweifach in einen Schieber eingeht und sich somit eine Veränderung in der Struktur ergab, die, ohne die historische Struktur zu überschreiben, nur dargestellt werden konnte, indem ein neuer Knoten aufgenommen wurde. Die Kontaktfeder KF1 ist seit diesem Zeitpunkt nicht mehr in Gebrauch.

Die gültige Version eines Produktes zu einem Zeitpunkt t_i ergibt sich aus der Menge aller in das Produkt eingehenden Teile in ihrer gültigen Version zum Zeitpunkt t_i .

Im allgemeinen arbeitet der Benutzer auf einen fixierten Zeitpunkt, den gegenwärtigen; nur für die Ersatzteilhaltung und -fertigung sind frühere Zeitpunkte von Interesse.

Für den gegenwärtigen Zeitpunkt kann mit einer versionsfreien Sicht gearbeitet werden. Die Bezeichnung Version kann durch die Bezeichnung des Objektes ersetzt werden. Die Attribute **Gültig-seit** und **In-Gebrauch** bleiben dann verdeckt.

Bei versionenbehafteter Sicht sind alle Attribute sichtbar.

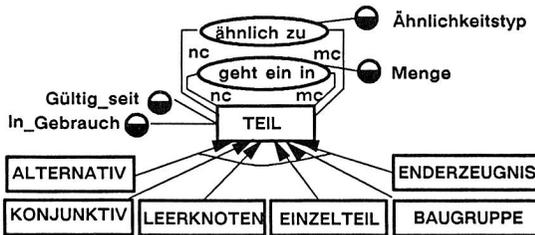


Bild 73: Semantisches Schema zur Darstellung versionenbehafteter Varianten

Ein semantisches Schema für die Speicherung versionenbehafteter Varianten ist in Bild 73 dargestellt.

Die Stammdaten werden unter der Bezeichnung **TEIL** zusammengefaßt. Ein **TEIL** kann sowohl vom realen Knotentyp **EINZELTEIL**, **BAUGRUPPE**, **ENDERZEUGNIS**, als auch vom virtuellen Knotentyp **ALTERNATIV**knoten, **KONJUNKTIV**knoten oder **LEERKNOTEN** sein. Eine Version eines **TEILs** wird über die **Teilenummer**, das Attribut **Gültig_seit**, sowie das Boolesche Attribut **In_Gebrauch** eindeutig identifiziert. Die Teilestruktur wird über die Beziehung geht ein in dargestellt, deren Primärschlüssel sich aus den Attributen **Oberteil**, **Unterteil** und **Menge** zusammensetzt.

Durch den Alternativknoten werden Varianten zusammengefaßt, die sich in Teilen ihrer Struktur unterscheiden. Bei auftrags-

neutrale Speicherung von Produkten und auf der Komplexitäts-ebene der Einzelteile besteht keine Zuordnung zu Variantenreihen. Deshalb wird noch die zusätzliche Beziehung ähnlich zu zwischen realen Teilen eingeführt. Durch das Attribut **Ähnlichkeitstyp** lassen sich damit Typenreihen, Baureihen sowie Teileklassen darstellen.

6.3 Rekonstruktion der Begriffe für die einzelnen Schritte der Montageplanung

Für den nachfolgenden Schemaentwurf werden zunächst zu jedem Schritt des in Kapitel 5 vorgestellten Planungsablaufs Datenschemata entwickelt und diese anschließend im nächsten Kapitel zu einem gemeinsamen konzeptionellen Schema integriert, wobei auch die durch die systemtechnische Betrachtung der Montage gewonnenen Erkenntnisse berücksichtigt werden.

6.3.1 Rekonstruktion der Begriffe der Montageaufgabenanalyse

Im Mittelpunkt der Montageaufgabenanalyse steht das Produkt, das als **TEIL** bezeichnet werden soll. Ein **TEIL** kann sowohl ein reales Teil also **EINZELTEIL**, **BAUGRUPPE** oder **ENDERZEUGNIS** als auch ein virtueller Knoten vom Typ **ALTERNATIVknoten**, **KONJUNKTIVknoten** oder **LEERKNOTEN** sein (Bild 70). Zur Beschreibung der Teilestrukturen, ihrer Varianten, Versionen und Ähnlichkeiten wird das semantische Schema aus Bild 69 übernommen.

Die Geometrie der realen **TEILE** kann in drei Formen beschrieben sein. Ein Teil kann als **3D-CAD-Modell** modelliert, in **ZEICHNUNGEN** beschrieben oder in **METAFILES** dargestellt werden. **ZEICHNUNGEN** werden über die **Zeichnungsnummer** identifiziert.

Ein **TEIL** soll in bestimmten **Stückzahlen** und während einer gewissen **Lebensdauer**, den **PRODUKTIONSDATEN**, produziert werden. Diese Daten stellen für den Montageplaner die Eingangsinformationen dar und sind für ihn informationelle Daten.

Für den Montageplaner werden manche **TEILE** zu **WERKSTUECKEN**, wobei er **HILFSTEILE**, **BASISTEILE** und **FUEGETEILE** unterscheidet. Diese **WERKSTUECKE** analysiert er und bestimmt ihre für die Montage wichtigen **WERKSTUECKEIGENSCHAFTEN**, wie z.B. das **Gewicht**, und zu welcher **WERKSTUECKKLASSE** sie zugeordnet werden können. Die wichtigsten Attribute der **WERKSTUECKKLASSE** sind der **Verhaltenstyp** und die **Größenklasse**. Aus der Struktur der einzelnen Varianten von **BAUGRUPPEN** und **ENDERZEUGNISSEN** leitet er mit Hilfe der Fügeflächenmatrix **VORRANGGRAPHEN** ab, die über die Beziehungen zwischen Vorranggraphknoten (**VGKNOTEN**) und Vorranggraphkanten (**VGKANTEN**) beschrieben werden. Ein **EINZELTEIL** oder eine **BAUGRUPPE** geht ein in einen Vorranggraphknoten.

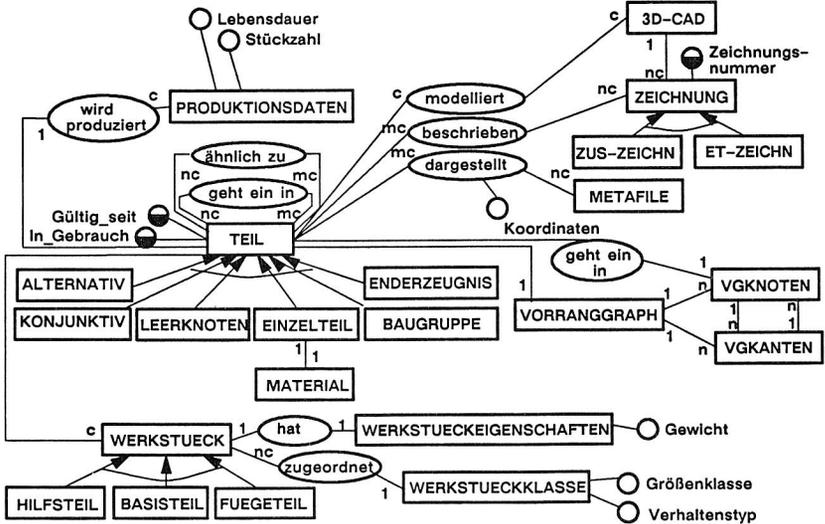


Bild 74: Semantisches Schema zur Montageaufgabenanalyse

Nach dem in Bild 74 dargestellten semantischen Schema können die Daten der Montageaufgabenanalyse verwaltet werden.

6.3.2 Rekonstruktion der Begriffe zur Beschreibung des Montageablaufs

Die Montagekonzepterstellung beginnt mit der strukturellen und funktionalen Beschreibung des Montageablaufs einer **BAUGRUPPE** oder eines **ENDERZEUGNISSES**. Ausgangspunkt sind **VORRANGGRAPHEN** von Varianten oder Baureihen des Produktes, das zusammengesetzt werden soll.

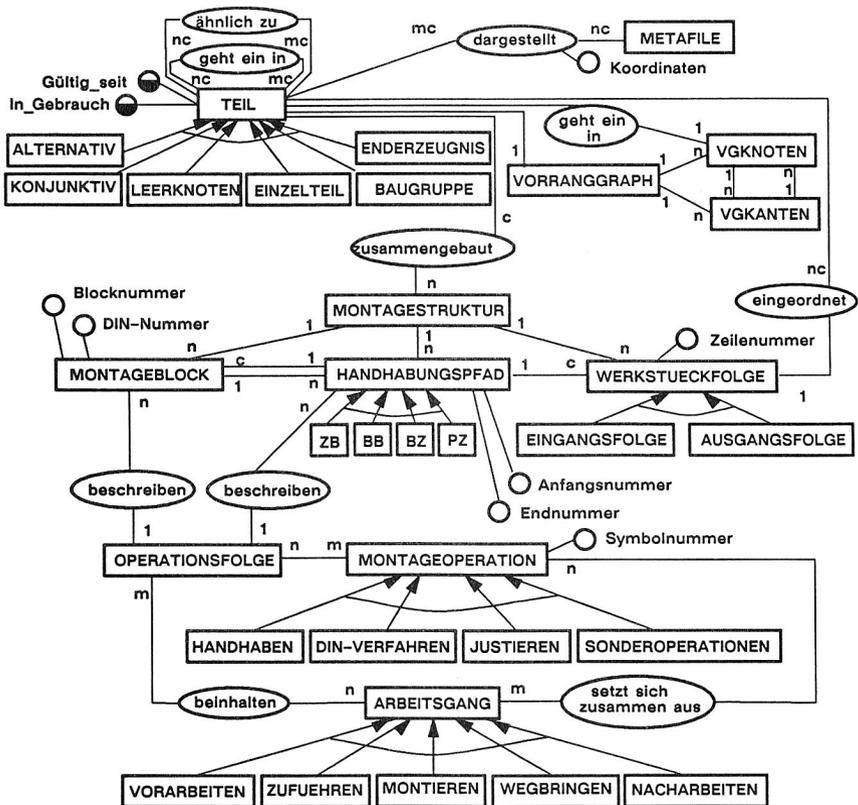


Bild 75: Semantisches Schema zur Montageablaufbeschreibung

Jedes **TEIL**, das in einen Vorranggraphknoten (**VGKNOTEN**) eingeht, wird in eine **WERKSTUECKFOLGE** eingeordnet, die vom Typ **EINGANGSFOLGE** ist. Die Beschreibung der **MONTAGESTRUKTUR**, nach der Varianten oder Baureihen von Produkten zusammengebaut werden, besteht aus solchen **WERKSTUECKFOLGEN**, aus Montageblöcken, die durch die **DIN-Nummer** des durch sie beschriebenen Fügeverfahrens gekennzeichnet werden, und aus Handhabungspfaden. Montageblöcke und Handhabungspfade werden zu den Objekttypen **MONTAGEBLOCK** und **HANDHABUNGSPFAD** zusammengefaßt.

Ein **HANDHABUNGSPFAD** kann eine Zeile einer **WERKSTUECKFOLGE** mit einem **MONTAGEBLOCK** verbinden und ist dann vom Typ **ZB**, oder zwei Montageblöcke (Typ **BB**), oder einen **MONTAGEBLOCK** bzw. einen Handhabungspfad mit einer Zeile der **AUSGANGSFOLGE** (Typ **BZ** bzw. **PZ**). Die Montageblöcke und die Zeilen der Werkstückfolge einer **MONTAGESTRUKTUR** werden fortlaufend mit **Zeilennummern** und **Blocknummern** numeriert. Diese Nummern dienen als **Anfangsnummer** und **Endnummer** zusammen mit dem Typ (**ZB**, **BB**, **BZ** oder **PZ**) zur Identifikation von **HANDHABUNGSPFADEN**, wobei für den Typ **PZ** die Nummer des vorausgehenden Montageblocks verwendet wird. **OPERATIONSFOLGEN** beschreiben die Folge der Operationen, die in einem **HANDHABUNGSPFAD** oder einem **MONTAGEBLOCK** ausgeführt werden. **OPERATIONSFOLGEN** beinhalten einen oder auch mehrere Arbeitsgänge vom Typ **VORARBEITEN**, **ZUFUEHREN**, **MONTIEREN**, **WEGBRINGEN** und **NACHARBEITEN**. Diese setzen sich zusammen aus **MONTAGEOPERATIONEN**, wie in Bild 40 dargestellt. Eine **OPERATIONSFOLGE** mit einem **ARBEITSGANG** des **ZUFUEHRENS** kann einem **HANDHABUNGSPFAD** vom Typ **ZB**, eine mit einem **ARBEITSGANG** des **MONTIERENS** einem **MONTAGEBLOCK** und eine mit einem **ARBEITSGANG** des **WEGBRINGENS** **HANDHABUNGSPFADEN** vom Typ **BB**, **BZ** oder **PZ** zugeordnet werden. **OPERATIONSFOLGEN** mit einem **ARBEITSGANG** des **VORARBEITENS** oder des **NACHARBEITENS** können sowohl zu einem **MONTAGEBLOCK** als auch zu einem **HANDHABUNGSPFAD** gehören.

Als informationelle Daten stehen dem Montageplaner **MONTAGEOPERATIONEN** zur Verfügung, aus denen er die **DIN-Nummer** eines **MONTAGEBLOCKS** und die Operationen einer **OPERATIONSFOLGE** auswählt, wobei zwischen **HANDHABEN**, **DIN-VERFAHREN**, **JUSTIEREN** und

SONDEROPERATIONEN unterschieden wird.

METAFILES ermöglichen es dem Montageplaner sich die geometrische Gestalt der **TEILE** bei der Festlegung der **OPERATIONSFOLGEN** zu betrachten. Jeder **MONTAGEOPERATION** ist ein Symbol durch eine **Symbolnummer** zugeordnet.

6.2.3 Rekonstruktion der Begriffe der Geräteauswahl und Layouterstellung

Eine wesentliche Voraussetzung zur Unterstützung der rechnergestützten Wirkorgan- und Geräteauswahl sowie zur Montagelayouterstellung ist eine Informationsdatenbank in der alle Geräteklassen und Geräteausprägungen verwaltet werden, die zum Aufbau einer Montageanlage erforderlich sind. Aus dem semantischen Datenbankentwurf resultiert jedoch keineswegs ein optimales Schema; im Laufe der Rekonstruktion bieten sich vielmehr Alternativen an, aus denen eine ausgewählt werden muß. So bieten sich für die Modellierung von Geräten der Montage mehrere mögliche Modellstrukturen an.

Eine funktionsorientierte Gliederung der Geräte zeigt, daß viele verschiedene Geräte sich aus ähnlichen Komponenten zusammensetzen. Ein mögliches semantisches Schema für eine funktionsorientierte Gerätegliederung zeigt Bild 76. Die nicht näher benannten Beziehungstypen können als geht ein in betitelt werden. Die einzelnen Komponenten können jedoch auch selbständige Geräte sein und haben teilweise sehr unterschiedliche Attribute, weshalb sie durch eigene Objekttypen repräsentiert werden müssen. Dies hat zur Folge, daß jedes neue Gerät, welches keiner bestehenden Geräteklasse zugeordnet werden kann, eine grundlegende Änderung des Schemas nach sich ziehen würde.

Es wurde deshalb eine komponentenorientierte Gerätestrukturierung gewählt, bei der sich die technischen Daten eines Gerätes beliebig aus den vorhandenen Teilinformationen über einzelne Komponenten zusammensetzen lassen (Bild 77).

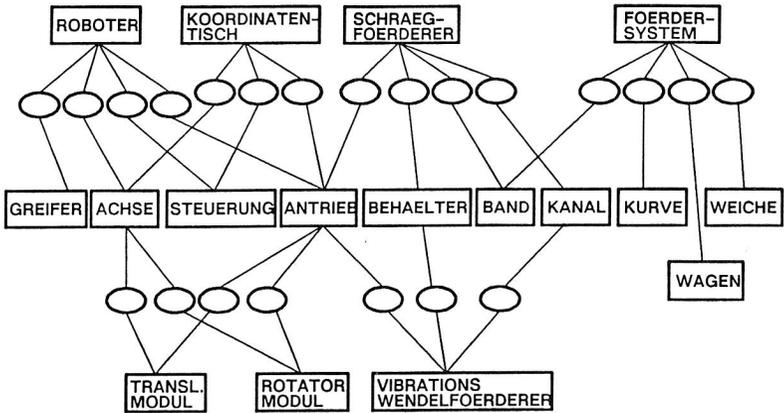


Bild 76: Semantisches Schema einer funktionsorientierten Geräteklassifizierung

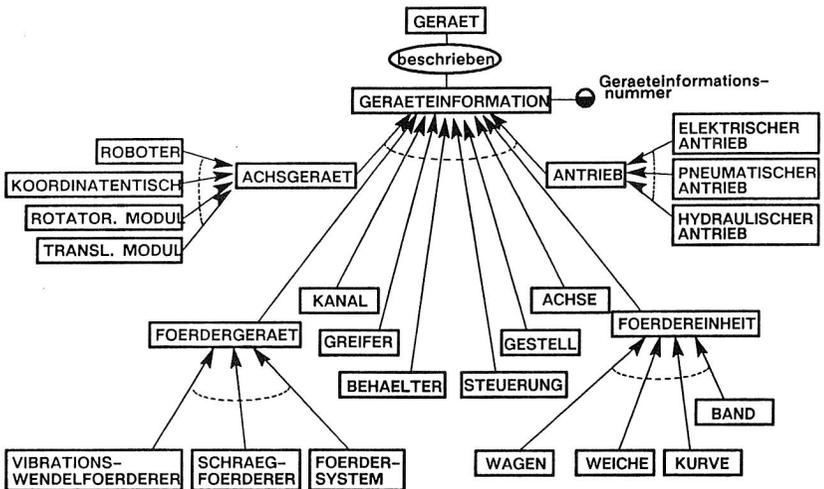


Bild 77: Semantisches Schema einer komponentenorientierten Geräteklassifizierung

Dieses Modell ist offen für Einträge neuer Geräte, da sich die Beschreibung der Geräte an der Zusammenstellung der einzelnen Komponenten orientiert. Im ungünstigsten Fall muß ein neuer Objekttyp eingeführt werden, was eine Erweiterung des Modells darstellt die relativ unproblematisch ist.

Ein **GERAET** wird in diesem Modell beschrieben durch eine Reihe von **GERAETEINFORMATIONEN**, wobei einzelne **GERAETEINFORMATIONEN** mehreren **GERAETEN** zugeordnet sein können.

Zur Vervollständigung des Montagekonzeptes ist die Auswahl von Wirkorganen erforderlich, mit denen die bei der Montageablaufbeschreibung festgelegten **OPERATIONSFOLGEN** ausgeführt werden können. Wirkorgane entsprechen **GERAETEKLASSEN**. Jede **GERAETEKLASSE** kann bestimmte **MONTAGEOPERATIONEN** bewirken, ebenso kann eine **MONTAGEOPERATION** von mehreren **GERAETEKLASSEN** bewirkt werden. Für die Ausführung einer Montageoperation kann eine **Schätzzeit** angegeben werden. Die Geräte einer **GERAETEKLASSE** sind für bestimmte **WERKSTUECKKLASSEN** einsetzbar, die beschreiben werden durch **Größenklasse** und **Verhaltenstyp**.

Damit ist es möglich, für jede **MONTAGEOPERATION** einer **OPERATIONSFOLGE** des Montagekonzeptes eine **GERAETEKLASSE** zu bestimmen, mit der diese ausgeführt werden kann (Bild 78).

Die Aufgabe bei der Montagelayouterstellung besteht darin, geeignete **GERAETE** für **KOMPONENTEN** von **MONTAGEEINHEITEN** auszuwählen, die **FUNKTIONEN** ausüben können, die eine oder mehrere Operationen der an einem **WERKSTUECK** auszuübenden **OPERATIONSFOLGEN** bewirken. Jede dieser **KOMPONENTEN** kann eingehen in **MONTAGEZELLEN** und **MONTAGEANLAGEN** von denen während des Planungsprozesses mehrere **ALTERNATIVEN** zusammengestellt werden können, wobei die erste Ausprägung als die erste **ALTERNATIVE** betrachtet wird. Die **MONTAGEANLAGEN** können verschiedene Varianten von **BAUGRUPPEN** und **ENDERZEUGNISSEN** zusammenbauen und müssen dabei eine notwendige **Taktzeit** einhalten.

eines **WERKSTUECKES**, ein **GERAET** ausgewählt und einer **KOMPONENTE** zugeordnet werden. Mit den **Schätzzeiten** für die von **GERAETEN** einer **GERAETEKLASSE** bewirkten **MONTAGEOPERATIONEN** kann eine ungefähre **Ausführzeit** für die **FUNKTIONEN**, die **MONTAGEEINHEITEN** ausüben, bestimmt werden. **MONTAGEZELLEN** können dann so zusammengestellt werden, daß die **Ausführzeiten** für **OPERATIONSFOLGEN** dieser **MONTAGEZELLE** kleiner als die geforderte **Taktzeit** der **MONTAGEANLAGE** bleiben.

Die geometrische Darstellung der **MONTAGEEINHEITEN** vom Typ **MONTAGEANLAGE** und **MONTAGEZELLE** und damit des Montagelayouts, erfolgt in gleicher Weise wie für die **TEILE**. Die geometrische Darstellung der **KOMPONENTEN** ist normalerweise den **GERAETEN** zugeordnet. Für **FUNKTIONEN** zu denen keine geeigneten **GERAETE** gefunden werden können, müssen neue **KOMPONENTEN** modelliert werden.

6.3.4 Rekonstruktion der Begriffe der Bewegungssimulation

Bei der Bewegungssimulation werden am Graphikbildschirm die Bewegungen von Handhabungsgeräten innerhalb einer Zelle simuliert. Die Bewegungssimulation baut auf das mit einem CAD-System erstelltes 3D-Layout einer **MONTAGEZELLE** auf. Jedes **GERAET**, das einer **KOMPONENTE** der **MONTAGEZELLE** zugeordnet ist, ist als **3D-CAD-Volumenmodell** modelliert. Für die Simulation muß aus dem Volumenmodell einer **MONTAGEZELLE** ein **DRAHTMODELL** abgezogen und für die beweglichen **KOMPONENTEN** die **KINEMATIK** beschrieben werden.

Zur Simulation werden anhand des **DRAHTMODELLS** Bewegungsprogramme für die beweglichen **KOMPONENTEN** erstellt, die **FUNKTIONEN** ausüben, die wiederum **OPERATIONSFOLGEN** an **WERKSTUECKEN** bewirken. Beim Testen entstehen für eine **FUNKTION** meist mehrere Versionen solcher Bewegungsprogramme, die als **IR-PROGRAMME** bezeichnet werden. Diese Versionen werden durch die Attribute **Gültig_seit** und **In_Gebrauch** gekennzeichnet. Zu jedem **IR-PROGRAMM** kann die **Ausführzeit** bestimmt und mit der geforderten **Taktzeit** der

MONTAGEANLAGE verglichen werden in die die **MONTAGEZELLE** eingeht.
Für Kollisionsbetrachtungen kann ein **BEWEGUNGSFILM** für eine **FUNKTION** aus dem **IR-PROGRAMM** erzeugt werden.

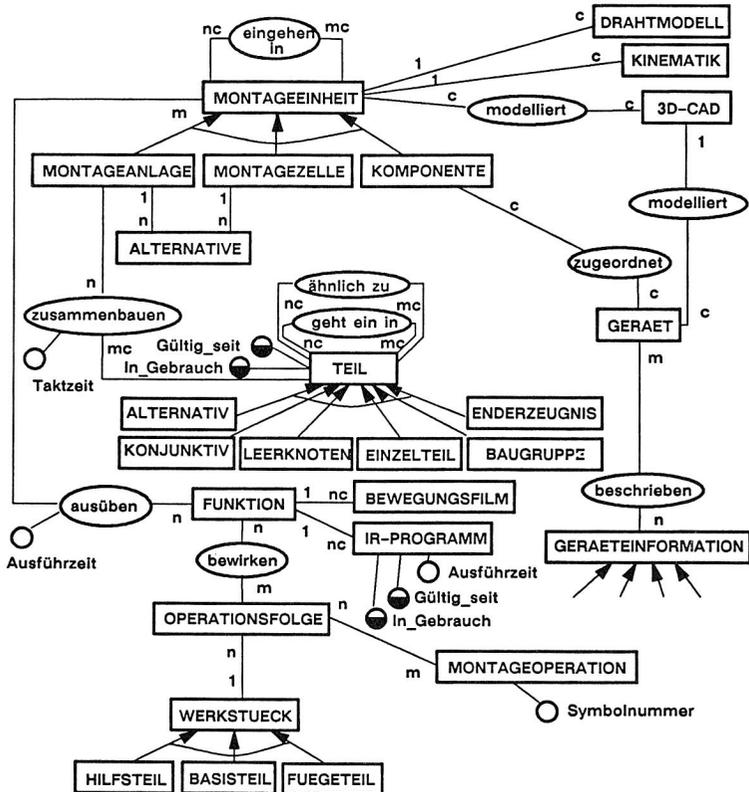


Bild 79: Semantisches Schema zur Bewegungssimulation

Eine **MONTAGEEINHEIT** wird beschafft und verursacht dadurch **Beschaffungskosten**. Den **KOMPONENTEN**, die eingehen in MONTAGEZELLEN und **MONTAGEANLAGEN**, sind **GERAETE** zugeordnet die zu einem bestimmten **Preis** vom **VERTRIEB** eines **GESCHAFTSPARTNERS** vertrieben werden. Damit lassen sich die **Beschaffungskosten** bestimmen, die ergänzt um **Installationskosten** usw. den **KAPITALEINSATZ** für eine **MONTAGEEINHEIT** ergeben.

Für das Zusammenbauen von Varianten einer **BAUGRUPPE** oder eines **ENDERZEUGNISSES** können verschiedene **ALTERNATIVEN** von **MONTAGEANLAGEN** entwickelt und durch **WIRTSCHAFTLICHKEITSKENNZAHLEN** verglichen werden. Jede diese alternativen **MONTAGEEINHEITEN** hat unterschiedliche **BETRIEBSKOSTEN**. Die **BETRIEBSKOSTEN** beinhalten **LOHNKOSTEN**, die sich bestimmen lassen aus der **Vorgabezeit** für die **BEDIENUNG** der **MONTAGEEINHEIT**, dem **Lohnfaktor** für die **LOHNGRUPPE** des Bedienpersonals und der **Stückzahl** aus den **PRODUKTIONSDATEN** nach denen das **TEIL** produziert wird, welches auf der **MONTAGEANLAGE** zusammengebaut werden soll. Die **KAPITALDIENSTKOSTEN** berechnen sich aus dem **KAPITALEINSATZ** und der Nutzungsdauer der **MONTAGEANLAGE**, die der **Lebensdauer** aus den **PRODUKTIONSDATEN** für das **TEIL** entspricht. Für Gewinnvergleichsrechnungen muß noch der **Erlös** der produzierten **TEILE** berücksichtigt werden.

6.4 Integration der externen Schemata in ein konzeptionelles Schema

In einem konzeptionellen Schema wird die Beschreibung aller Daten einer Datenbank zusammengefaßt. Es werden die Datenbeschreibungen für verschiedene Anwendungen auf die Datenbank zu einem gemeinsamen Schema integriert.

Ein Vergleich der in Kapitel 6.3 rekonstruierten semantischen Schemata für die Schritte der Montageplanung zeigt viele Übereinstimmungen der einzelnen Schemata. Dies liegt daran, daß für die meisten Planungsschritte die Ergebnisse vorangegangener

Planungsabschnitte Eingangsdaten darstellen.

Gerade darin zeigt sich ein besonderer Vorteil, der durch den Einsatz einer Datenbank erreicht werden kann. Durch die Nutzung eines gemeinsamen Datenbestands reduziert sich der Eingabeaufwand für die Anwendungsprogramme zur Unterstützung einzelner Planungsschritte erheblich und es werden Fehler durch falsche Dateneingabe und Redundanzen vermieden.

Bei der Integration externer semantischer Schemata geht es vor allem darum, Synonyme (verschiedene Namen für dasselbe Objekt) und Homonyme (gleiche Namen für verschiedene Objekte) zu lokalisieren und zu entfernen. Dies betrifft jedoch vor allen die Objekttypen. Die Wahl gleicher Bezeichnungen für Beziehungstypen läßt sich bei der semantischen Modellierung praktisch kaum vermeiden. Sie ist auch unproblematisch für die Transformation in Implementationsschemata, solange es sich um hierarchische Beziehungen (siehe Bild 18) handelt. Für konditionelle und netzwerkartige Beziehungen müssen bei der Transformation z.B. in ein relationales Datenbankschema eigene Relationen definiert werden, die einen eindeutigen Namen erfordern (Bild 19). Diese eindeutige Namensvergabe kann erreicht werden, indem die Relationen in denen Beziehungen dargestellt werden, mit einer Kombination der Namen der Objekttypen die an der Beziehung beteiligt sind, benannt werden. Für die semantische Modellierung sind dann gleiche Benennungen von Beziehungen zulässig.

Ein konzeptionelles Schema welches aus den oben vorgestellten semantischen Schemata entwickelt wurde, zeigt Bild 82a und 82b. Auf die nochmalige Darstellung der Geräteinformationen aus Bild 77 wurde aus Platzgründen verzichtet.

Bei der Zusammenfassung der einzelnen semantischen Schemata zu einem konzeptionellen Schema macht sich positiv bemerkbar, daß schon bei der Erstellung der Schemata für die verschiedenen Planungsschritte der Montage die Erkenntnisse der systemtechnischen Betrachtung berücksichtigt wurden. Zudem wurde darauf

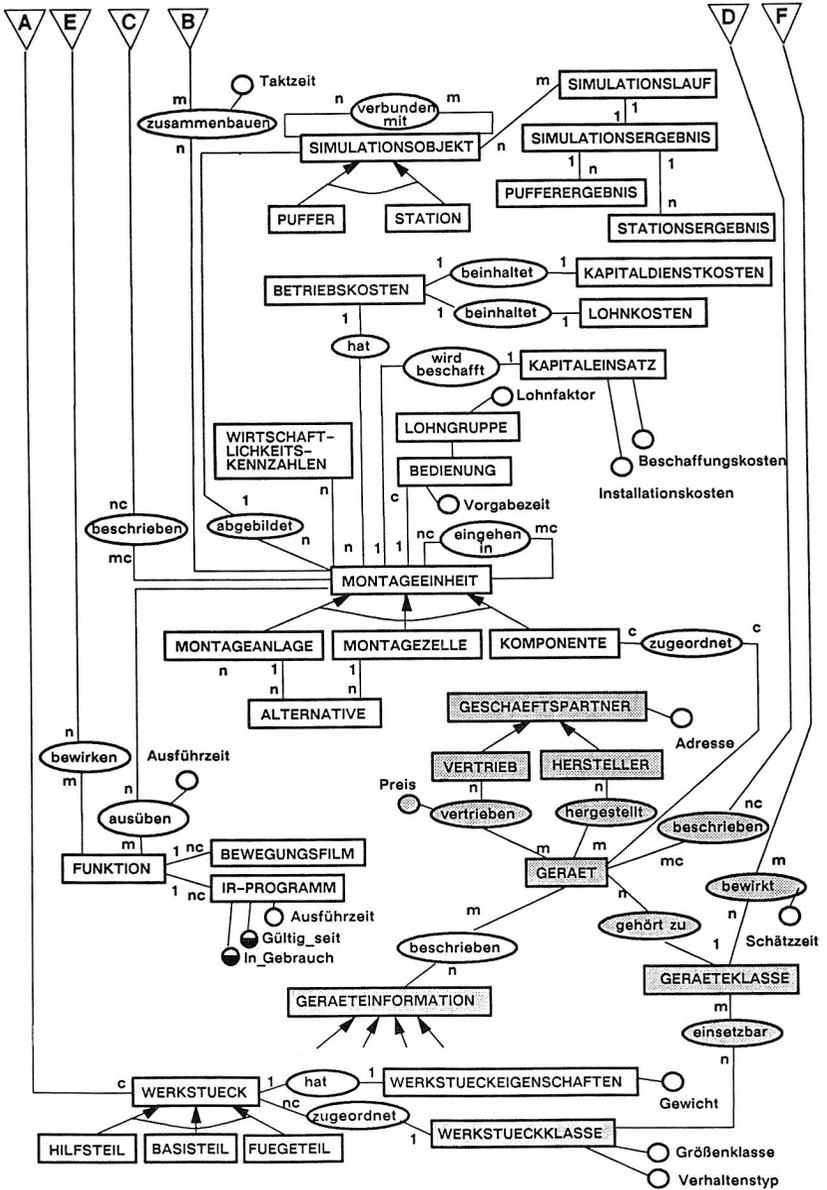


Bild 82b: Konzeptionelles Schema zur Montageplanung

geachtet, möglichst die Begriffe vorangegangener Planungsschritte wieder zu verwenden. Dadurch ließen sich die einzelnen Schemata relativ einfach zusammenfassen.

Lediglich eine gravierende Änderung wurde durchgeführt. Für die verschiedenen geometrischen Darstellungen von **TEILEN**, **MONTAGEEINHEITEN** und **GERAETEN** wurde der generalisierende Objekttyp **GEOMETRIEDATEN** eingeführt, durch den diese beschreiben werden. **GEOMETRIEDATEN** können vom Typ **ZEICHNUNG**, **3D-CAD-Modell**, **METAFILE**, **KINEMATIK** oder **DRAHTMODELL** sein. Bei den **KINEMATIK**daten handelt es sich zwar nicht um echte Geometriedaten, sie sind jedoch eng mit diesen verknüpft und wurden deshalb diesem Objekttyp mit zugeordnet.

Die Objekttypen der informationellen Daten wurden durch eine graue Unterlegung hervorgehoben, wobei die echten Katalogdaten dunkler angelegt wurden.

7. Exemplarische Implementierungen

Das in Kapitel 6 vorgestellte konzeptionelle Schema wurde implementiert und eine Beispiel-Datenbank nach diesem Schema aufgebaut. Zusätzlich wurden zu einigen der beschriebenen Montageplanungsschritte eigene Anwenderprogramme entwickelt, deren Daten in dieser Datenbank verwaltet werden.

Bevor nun die einzelnen Implementierungen beschrieben werden, wird kurz die Implementierungsumgebung vorgestellt.

7.1 Implementierungsumgebung

Hardware-Basis für die Implementierungen waren, über einen Token-Ring vernetzte Apollo Workstations. Jeder Rechner ist mit einem hochauflösenden, graphikfähigen Bildschirm, einer Tastatur mit Funktionstasten und einer Maus ausgestattet. Als Betriebssystem sind sowohl AEGIS als auch UNIX einsetzbar.

Apollo bietet mit den Softwaretools DOMAIN-DIALOGUE und DOMAIN-GMR und GPR komfortable Hilfsmittel zur Gestaltung anwenderfreundlicher Benutzeroberflächen von Programmen an.

Mit DOMAIN-DIALOGUE können dialoggeführte Anwendungsprogramme formuliert werden. Es fungiert dabei als Bindeglied zwischen dem Benutzer und dem Anwendungsprogramm. Es fordert vom Benutzer Aktionen und Daten an, gibt diese über ein "Application Interface" an das Anwendungsprogramm weiter und leitet die Reaktionen über ein "User Interface", durch das die Bildschirmgestaltung festgelegt wird, an den Benutzer zurück. Damit wird der Aufbau von hierarchischen Menues, den sogenannten Popups, und von Masken ermöglicht.

DOMAIN-GMR und GPR sind Programmpakete, die Routinen zur Gestaltung graphischer Anwendungen zur Verfügung stellen. GPR (graphic primitives routines) ist an der Darstellung mit Pixels orientiert bei der Zeichnungen als Bitmaps gespeichert werden. GMR

(graphic metafile resource) ist eine Sammlung von Routinen zum Erzeugen, Anzeigen und Editieren von Bilddaten aus geometrischen Grundelementen wie Linien, Kreisen und Rechtecken. Es wird ein geometrisches Modell beschrieben, das beliebig, also auch in Ausschnitten, auf eine am Bildschirm definierte Graphikfläche (Viewport) abgebildet werden kann.

Als CAD-System stand der geometrische 3D-Modellierer ROMULUS zur Verfügung. ROMULUS wurde an der Universität Cambridge (GB) entworfen und von der englischen Firma Shape Data zur Produktreife entwickelt. Den Datenstrukturen der mit ROMULUS erzeugten CAD-Modelle liegt das in Bild 49 vorgestellte allgemeine konzeptionelle Schema zugrunde, in dem geometrische und topologische Elemente unterschieden werden.

Daneben verfügt ROMULUS noch über zugeordnete (associated) Elemente, die zur Speicherung nichtgraphischer Information vorgesehen sind [107].

Wichtige Funktionen in ROMULUS sind die Programmiersprache MCL (Macro Command Language) und das KERNEL-INTERFACE (KI).

MCL ist eine BASIC-ähnliche Sprache, die innerhalb des ROMULUS-Prozesses abläuft. Der Sprachumfang umfaßt neben einfachen logischen und arithmetischen Anweisungen auch die wichtigsten ROMULUS-Befehle. Das KERNEL-INTERFACE ist eine Anwenderschnittstelle, die es erlaubt, auf alle Funktionen des Modellierers ROMULUS zuzugreifen. Es besteht aus etwa 160 Fortranroutinen, die in eigene Programme eingebunden werden können und einen problemlosen Austausch von Variablenwerten von und zu ROMULUS ermöglichen.

Die Datenbank wurde mit dem relationalen Datenbankmanagementsystem INGRES implementiert. INGRES ist ein Produkt der Relational Technology Inc. (RTI), der Prototyp wurde zwischen 1973 und 1975 in Kalifornien an der Universität Berkeley entwickelt. Auf den Apollo Workstations ist INGRES nur unter dem Betriebssystem UNIX lauffähig. INGRES verfügt über zwei relationale Abfragesprachen. Ursprünglich wurde QUEL (Query Language) als

Abfragesprache für INGRES entwickelt, als jedoch SQL (Structured Query Language) zum internationalen Standard erklärt wurde, wurde auch diese in INGRES integriert; man hat somit die Wahl zwischen beiden.

In der Systemstruktur von INGRES kann zwischen zwei Komponenten unterschieden werden, dem "backend" und dem "frontends" [126] (siehe auch Bild 83). Das backend umfaßt die fundamentalen DBMS-Funktionen, eingeschlossen die volle Unterstützung von QUEL und SQL, wie Daten-definition, Datenmanipulation, Datensicherheit, Datenintegrität usw. Es beinhaltet auch den Transaction Manager, der die nötige Zugriffskontrolle und dem Recovery Support bereitstellt.

Die frontends bilden das INGRES Application Development System und dienen zum Erstellen und Anwenden von Applikationen zur interaktiven Nutzung von INGRES-Datenbanken. Die wichtigsten Komponenten sind QUERY, das Datenbankabfrage, -update und eingabesystem, REPORT zur Definition und zum Schreiben von Reports, QBF mit dem Abfragen in vordefinierten Formularen durchgeführt werden können und ABF zum Testen und Entwerfen von Applikationen. Sie gestatten die komfortable Kommunikation mit einer Datenbank. Genaugenommen gehören auch die QUEL und SQL Terminal Monitoren zu den frontends.

Aus der Sicht des backend sind die frontends nichts anderes als Anwendungsprogramme [126].

7.2 Implementierung des Datenbankschemas

Wie schon erwähnt wurde zur Implementierung des in Bild 82a und 82b vorgestellten konzeptionellen Schemas das relationale Datenbankmanagementsystem INGRES eingesetzt und eine Datenbank ASSDB (Assembly Database) für die Montageplanung nach diesem Schema aufgebaut.

Es wurde die relationale Datenbanksprache SQL gewählt weil

diese weitgehend standardisiert ist. Die wichtigsten Befehle zur Definition von Datenbanken mit SQL sind:

```
CREATE TABLE
CREATE VIEW
CREATE INTEGRITY
CREATE PERMIT
MODIFY.
```

Mit dem Befehl CREATE TABLE werden die Basistabellen (Relationen) definiert. Die Definitionen der Relationen wurden nach den in Kapitel 3.3.4 beschriebenen Transformationsregeln aus dem semantischen Schema nach Bild 82a und 82b abgeleitet.

Da INGRES für die Namensvergabe von Relationen und Attributen nur Zeichenketten der Länge ≤ 12 zuläßt, wurden längere Namen des semantischen Schemas auf 12 Zeichen gekürzt.

Bei der Abbildung von Variantenbeziehungstypen wurden für diejenigen Gattungsobjekttypen eigene Tabellen definiert, die gegenüber den Artobjekttypen zusätzliche Attribute aufweisen. So erübrigen sich z.B. eigene Tabellen für die Gattungsobjekte **ALTERNATIV**knoten, **KONJUNKTIV**knoten und **LEERKNOTEN**, da alle zu diesen Knotentypen gehörenden Informationen in der Relation **TEIL** bereits enthalten sind.

Die Namen den Beziehungen zwischen Objekttypen waren teilweise nicht eindeutig. Deshalb wurden zur Darstellung von netzwerkförmigen Beziehungen die Bezeichnungen der Relationen aus Abkürzungen der an den Beziehungen beteiligten Objekttypen zusammengesetzt.

Für die Verwaltung von Geometriedaten wird nur ein Attribut zum Speichern des Objektnamens in der Datenbank vorgesehen. INGRES unterstützt die Abspeicherung langer Felder nicht. Darum werden die Geometriedaten in normalen Dateien gespeichert und nur die Namen dieser Dateien in der Datenbank verwaltet.

Der Befehl CREATE VIEW erlaubt es, spezielle Sichten auf eine oder mehrere Basistabellen zu definieren. Mit einem View können Spalten und Reihen aus Basistabellen ausgeblendet werden. Ein View kann als eine virtuelle Tabelle betrachtet werden, d.h. als eine Tabelle die, im Gegensatz zu Basistabellen, nicht wirklich existiert sondern sich nur für den Benutzer so darstellt. Ein externes Schema für einen Benutzer kann aus mehreren solchen VIEWS zusammengesetzt werden.

Die Definition von Integritätsbedingungen erfolgt mit dem Befehl CREATE INTEGRITY. Der Qualifikationsteil dieses Befehls darf sich jedoch nur auf die Attribute einer Relation beziehen. Damit besteht keine Möglichkeit, referentielle oder sonstige komplexe Integritätsbedingungen mit diesem Befehl zu formulieren. Es kann nur die Einhaltung von Wertebereichen überwacht werden. Dies wird damit begründet, daß die Antwortzeiten für komplexe Datenoperationen durch die Abprüfung referentieller Integritätsbedingungen unakzeptabel lang würden. Eine Möglichkeit die Einhaltung referentieller Integritätsbedingungen zu gewährleisten bestünde darin, entsprechende Prüfprogramme im Batchbetrieb in angemessenen Zeitabständen ablaufen zu lassen. Semantische Integritätsbedingungen für Wertebereiche wurden für das konzeptionelle Schema definiert.

Zugriffsrechte werden mit dem Befehl CREATE PERMIT vergeben. Damit wird festgelegt, welche Benutzer auf welche Basistabellen und Views nur lesend zugreifen dürfen und welche das Recht zum Einfügen, Ändern und Löschen haben. Die zugriffstechnische Trennung zwischen informationellen und operationalen Daten ist somit ohne weiteres möglich.

Die Speicherungsstrukturen werden mit dem Befehl MODIFY definiert. Dieser Befehl wird auch benötigt, um Primärschlüssel zu bestimmen, indem bei der Festlegung der Speicherungsstrukturen entsprechende Attribute mit **unique** gekennzeichnet werden.

Zur Datenmanipulation stehen in der relationalen Abfragesprache SQL die Befehle:

```
SELECT
UPDATE
DELETE und
INSERT
```

zur Verfügung.

Es können an dieser Stelle nicht alle Möglichkeiten relationaler Datenmanipulationen gezeigt werden. Zur Darstellung der Abfragemöglichkeiten diene ein Beispiel zur Auswahl eines Gerätes aus der Informationsdatenbank. Die dargestellte Anfrage wählt die Geräte aus, die bruchempfindliche (Werkstückeignung >5), kegelförmige Werkstücke der Werkstück-Größenklasse 6 weitergeben können:

```
select Nr, Bezeichnung from GERAET where
    Ws_ei_bruch > 5 and
    Grkl_max > 6 and
    Nr in ( select Genr from EINSETZBAR where
        Wfnr = 'kegel' ) and
    Gknr in ( select Gknr from BEWIRKT where
        FunktionsNr = 'weitergeben' ).
```

Das Beispiel zeigt, wie durch die Verknüpfung über mehrere Tabellen die Auswahl einer begrenzten Menge von Daten erfolgt. Es verdeutlicht auch die Vorteile relationaler gegenüber CODASYL-Abfragesprachen. Der "join" ermöglicht die direkte Verknüpfung der Basistabellen GERAET und BEWIRKT über das Attribut Gknr, in CODASYL hätte der Record GERAETEKLASSE berücksichtigt werden müssen.

Die Befehle UPDATE, DELETE und INSERT dienen zum Ändern, Löschen und Einfügen von Attributwerten in der Datenbank.

Wie die Datenbank genutzt wird zeigt die Systemarchitektur in Bild 83. Es ist dabei zu unterscheiden zwischen dem Datenbank-administrator und den Benutzern der Datenbank.

Der Datenbankadministrator ist mit besonderen Rechten ausgestattet. Ihm obliegt es, erforderliche Veränderungen an den Definitionen der Basistabellen, Views und Speicherungsstrukturen vorzunehmen und Zugriffsrechte zu vergeben. Er wird sich dabei der durch die frontends von INGRES gebotenen Möglichkeiten bedienen.

Für die eigentlichen Nutzer der Datenbank wurde mit DOMAIN-DIALOG eine einheitliche Benutzeroberfläche geschaffen.

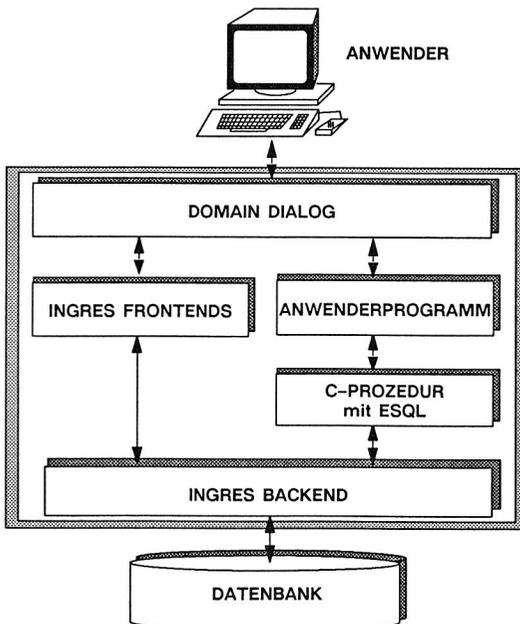


Bild 83: Systemarchitektur zur Datenbanknutzung

Typischerweise greift der Montageplaner nur selten unmittelbar auf den Datenbestand direkt zu, sondern er tut dies mittelbar über Entwicklungswerkzeuge. Im Hauptmenue der Benutzeroberfläche sind die für die einzelnen Planungsschritte zur Verfügung stehenden Anwenderprogramme aufgelistet. Diese Liste läßt sich bei Bedarf jederzeit ohne großen Aufwand erweitern, so daß neue Anwenderprogramme eingehängt werden können.

Der Benutzer kann mit der Maus eines der Programme anwählen und aktivieren. Werden im Rahmen dieses Anwenderprogrammes Datenbankzugriffe erforderlich, so werden entsprechende in der Programmiersprache C geschriebenen Prozeduren aufgerufen, in die Embedded-SQL-Statements eingebaut sind, die wiederum über das INGRES-backend mit der Datenbank kommunizieren.

Während normale Nutzer auf Katalogdaten nur Leserechte haben, sind für deren Pflege einige Benutzer mit erweiterten Rechten ausgestattet. Für diese können mit den INGRES-frontends z.B. ABF spezielle Applikationen bereitgestellt werden, die das direkte, interaktive Arbeiten mit der Datenbank (auch aus dem Hauptmenue der oben erwähnten Benutzeroberfläche anwählbar) ermöglichen.

7.3 Implementierung von Anwenderprogrammen

Zur Unterstützung des in Kapitel 5 geschilderten Planungsablaufs für automatisierte Montageanlagen und zum Testen der Montagedatenbank wurden einige Anwenderprogramme entwickelt.

Für die Planung einer Montageanlage mit den hier vorgestellten Anwendungsprogrammen wird davon ausgegangen, daß das Produkt, welches auf diese Anlage zusammengebaut werden soll, mit dem CAD-System ROMULUS modelliert wurde.

Zur Montageaufgabenanalyse stehen die Programme FuegAn (Fügeflächenanalyse [127]) zur Fügeflächenanalyse und WeKe (Werkstückkennwerte [128]) zur Ermittlung von Werkstückeigenschaften

ten zur Verfügung.

Das Fortran-Programm FuegAn dient zur Bestimmung der möglichen Teilfolgen für das Zusammenbauen. Mit ihm werden über das KERNEL-INTERFACE die Fügeflächen des mit ROMULUS modellierten Produktes ermittelt und daraus mit dem in Abschnitt 5.2.2 beschriebenen Verfahren über eine Fügeflächenmatrix ein Teilevorranggraph bestimmt und in die Datenbank geschrieben, wo auch der Objektname des CAD-Modells verwaltet wird (Bild 84). Diese Analysen der Teilfolgen für die Montage werden für alle Varianten des Produktes durchgeführt.

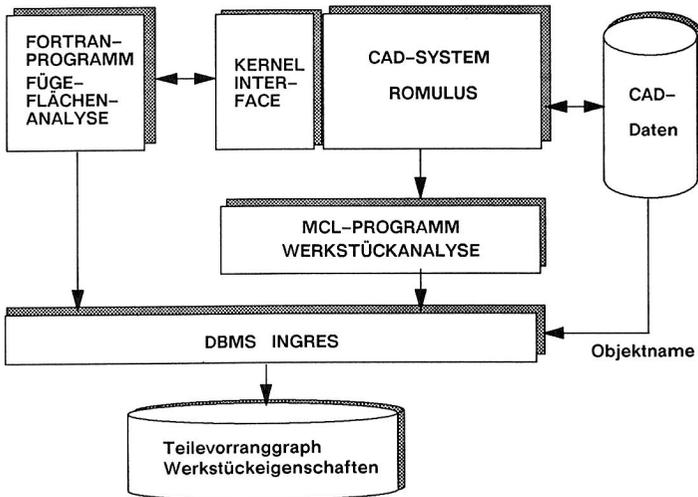


Bild 84: Ablauf der Montageaufgabenanalyse

Mit einem MCL-Programm WeKe werden aus dem CAD-Modell ableitbare Eigenschaften, wie Volumen, Schwerpunktlage usw. errechnet. Zusätzlich werden weitere für die Montage wichtige Eigenschaften wie z.B. der Verhaltenstyp interaktiv abgefragt und der Planer gibt die entsprechenden Daten nach visueller Beurteilung ein.

Die ermittelten Werkstückkennwerte und Werkstückeigenschaften werden ebenfalls der Produkt- bzw. Werkstückbeschreibung in der Datenbank zugeordnet.

Die mit dem Programm FuegAn bestimmten Teilevorranggraphen werden in das Programm MonStrEd (Montagestruktureditor [119]) zur Montageablaufbeschreibung eingelesen und für jede Variante wird eine Teileingangsfolge festgelegt. Die Benutzeroberfläche dieses Programms wurde mit DIALOG und GMR realisiert.

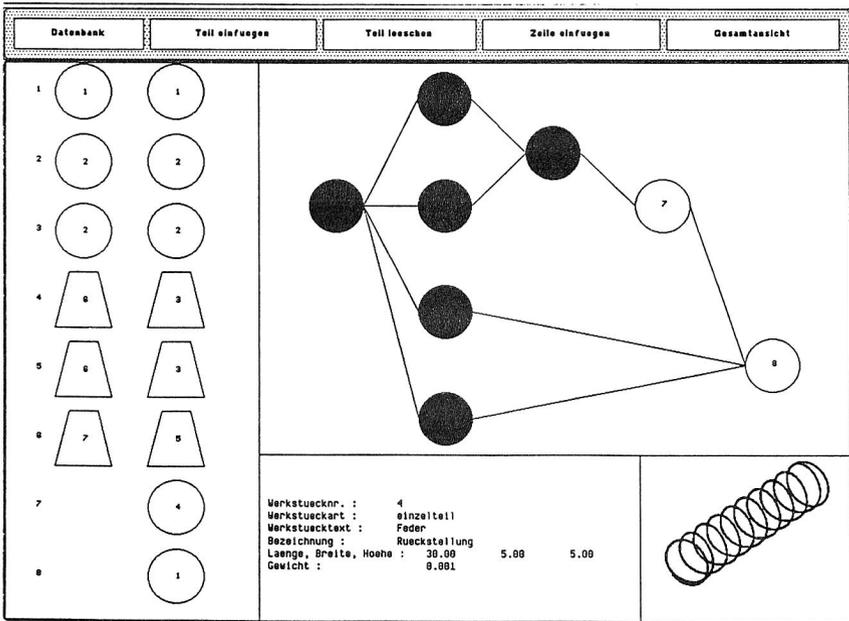


Bild 85: Benutzeroberfläche für die Erstellung der Teileingangsfolge

Der Vorranggraph wird graphisch dargestellt (Bild 85). Der Benutzer kann mit der Maus einen Knoten des Vorranggraphen anwählen und bekommt dann die Werkstückinformation in einem

Fenster angezeigt. Zusätzlich wird noch ein Metafile mit der graphischen Darstellung des Werkstück eingeblendet.

Der Montageplaner kann anhand dieser Informationen das Werkstück in die Teileeingangsfolge einordnen.

Vom Programm wird überwacht, daß der Teilevorranggraph nur auf einem zulässigen Weg abgearbeitet wird.

Nachdem der Planer die Teile aller Varianten in die Teileeingangsfolge eingeordnet hat, kann er nun eine Montagestruktur entwerfen. Er ordnet zusätzliche Hilfsteile, z.B. Werkstückträger in die Teilefolge ein, definiert Montageblöcke, in denen die Teile gefügt werden und verbindet diese durch Handhabungspfade (Bild 86).

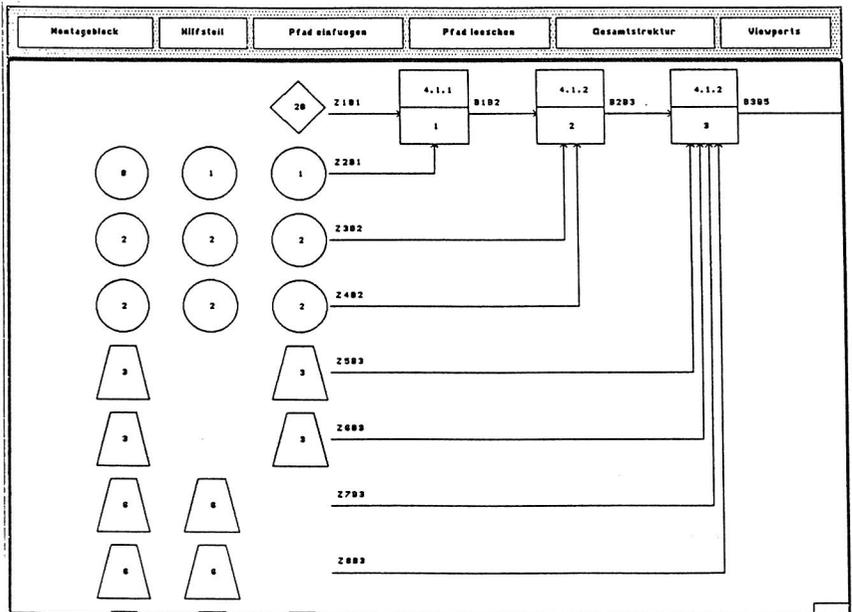


Bild 86: Benutzeroberfläche zur Montagestrukturbeschreibung

Er kann dabei beliebig in der Darstellung "zoomen" und hat auch die Möglichkeit den Bildschirm in zwei Viewports aufzuteilen. Auch kann er bei Bedarf aus der Datenbank abrufen, welche Fügeverfahren es gibt und sich durch eine kurze Beschreibung über diese informieren.

Hat der Planer die strukturelle Montagebeschreibung beendet, so erfolgt die operationale Beschreibung des Montageablaufs.

Der Montageplaner stellt aus den, den einzelnen Montageoperationen zugeordneten, Symbolen eine Operationsfolge zusammen und weist diese einem oder mehreren Handhabungspfaden oder Montageblöcken zu (Bild 87).

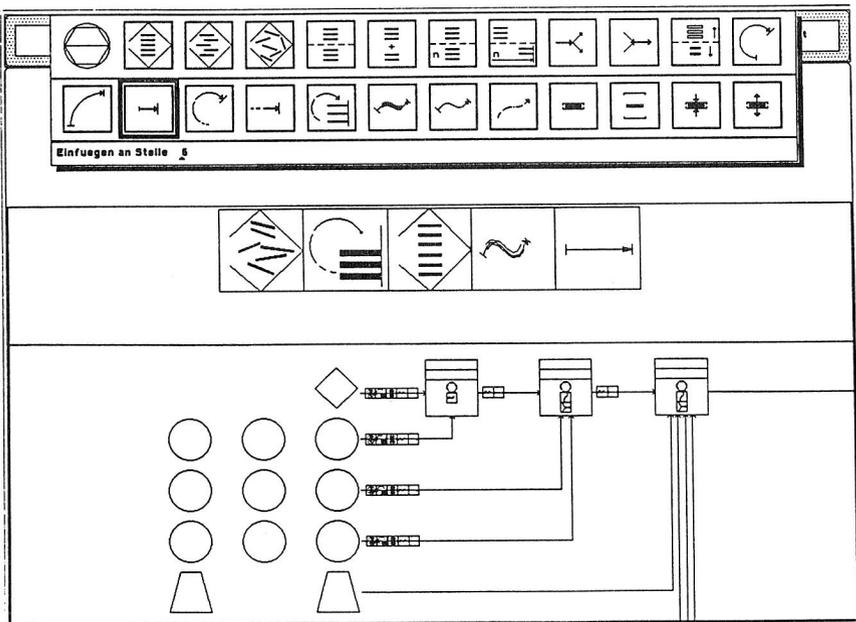


Bild 87: Benutzeroberfläche für die operationale Montageablaufbeschreibung

Das auf diese Weise erstellt Montagekonzept wird in der Datenbank gespeichert und dient als Ausgangsbasis für die Geräteauswahl.

Zur Wirkorgan- und Geräteauswahl steht das Programm Layout zur Verfügung. Mit ihm können, anhand der mit dem Programm MonStrEd erzeugten Operationsfolgen, für die Montage eines Produktes geeignete Geräteklassen (Wirkorgane) und diesen Klassen zugehörige Geräte ausgesucht werden.

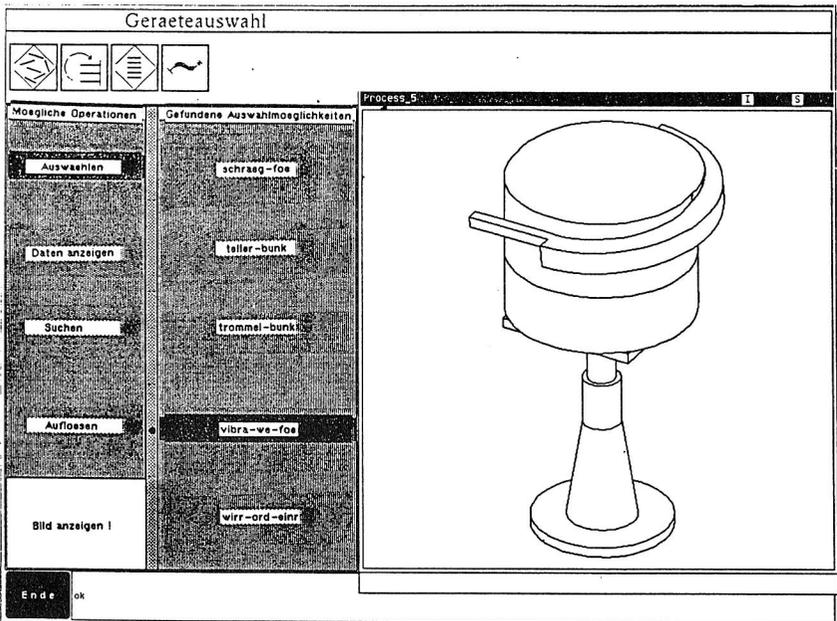


Bild 88: Benutzeroberfläche für die Geräteauswahl

Zunächst wird für die gesamte Operationsfolge zur Montage eines Werkstückes eine Geräteklasse gesucht, mit der diese bewirkt werden kann. Dabei wird die Werkstückklasse berücksichtigt, der das Werkstück zugeordnet ist. Wird keine geeignete Geräteklasse

gefunden, so kann die Operationsfolge in Teilketten aufgespalten werden, so lange bis für alle Operationen Geräteklassen zugeordnet werden können, wobei der Benutzer bei mehreren möglichen Lösungen eine Auswahl treffen muß. Eine geräteklassentypische Darstellung kann am Bildschirm ausgegeben werden (Bild 88).

Nachdem der Benutzer eine Geräteklasse festgelegt hat, kann er zu der Geräteklasse gehörende Geräteausprägungen aus der Datenbank auswählen.

Er kann sich dabei zu jedem Gerät gewünschte Informationen wie Kosten, Lieferzeiten und auch technische Daten anzeigen lassen. Die ausgewählten Geräte werden mit dem CAD-System ROMULUS zu Montagezellen und Montageanlagen zusammengestellt. Zur Aktualisierung der Datenbank kann ein von ROMULUS erzeugtes Journalfile genutzt werden.

Die Optimierung des Montagelayouts könnte mit dem Simulationssystem SIMU [112] unterstützt werden. Sämtliche für dieses Programm erforderlichen Daten wurden beim Entwurf des konzeptionellen Schemas berücksichtigt. Das Programm ist bisher jedoch nur auf einem PC installiert und müßte erst auf Apollo-Workstations portiert werden.

Zur Bewegungssimulation steht das Programm Caro [110] zur Verfügung. Die Benutzeroberfläche dieses Programms zeigt Bild 89.

Caro ist ein Werkzeug zur graphisch interaktiven Layoutplanung und Programmierung von Montagesystemen. Es können auf der Basis des CAD-Systems ROMULUS, einer Modelldatenbank und eines Moduls zur Programmierung und Simulation, Montagezellen gestaltet, programmiert und getestet werden.

Schließlich ist es möglich, die offline erstellten und getesteten Programme an die Robotersteuerungen zu übergeben. Es bietet die Möglichkeit Einbauuntersuchungen am Produktmodell, sowie Erreichbarkeitsstudien mit dem Montagesystemmodell durchzuführen. Es eignet sich auch zum Testen und Optimieren von Roboterprogrammen am Graphikbildschirm, unter anderem mit

Taktzeitermittlung [96].

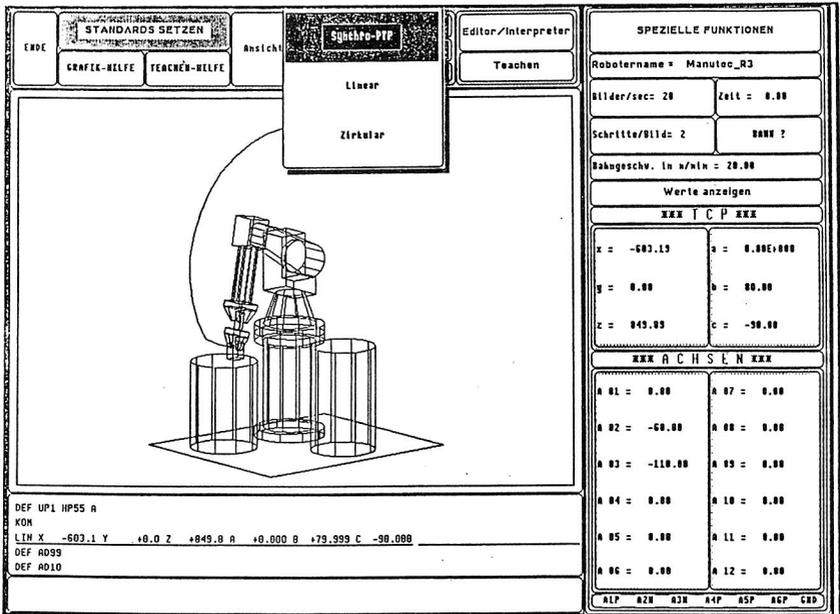


Bild 89: Caro-Benutzeroberfläche [110]

Die Anbindung von Caro an die Montagedatenbank ist derzeit in Arbeit.

Ein umfassendes Programmsystem zur Kostenrechnung befindet sich ebenfalls momentan im Aufbau und steht kurz vor der Vollendung. Dieses Programm arbeitet voll datenbankgestützt und nützt die oben beschriebenen Relationen.

8. Zusammenfassung

Die Entwicklung rechnergestützter Planungswerkzeuge für die Montage ist für die gegenwärtigen Automatisierungsbestrebungen in der Montage von grundlegender Bedeutung.

Die vorliegende Arbeit verfolgte eine dreifache Zielsetzung. Eine systemtechnische Betrachtung der Montage sollte dazu dienen, das Wissen über an der Montage beteiligte Komponenten zu systematisieren und zu klassifizieren. Es sollte zusätzlich ein Beitrag geleistet werden, bestehende Lücken in der Planungssystematik für die Montage zu schließen. Ein besonderes Anliegen war es, zu zeigen, wie Datenbanken entworfen werden können, mit denen einzelne Planungswerkzeuge zu einem Gesamtsystem integriert und damit effektiver genutzt werden können.

Zunächst wurden die entsprechenden Aufgabenstellungen in Kapitel 2 kurz vorgestellt und gezeigt, daß dem Einsatz von Datenbanken in einer Verfahrenskette zur Montageplanung eine zentrale Bedeutung zukommt.

Bisher wurden Datenbanken im Bereich der Montageplanung jedoch kaum genutzt. Dies wird hauptsächlich darauf zurückgeführt, daß die vorwiegend für kommerzielle Anwendungen entwickelten Datenbankmanagementsysteme für technische Anwendungen nur unzureichend geeignet sind und der Datenbankentwurf sich als eine sehr komplexe Aufgabenstellung darstellt.

Kapitel 3 enthält eine Einführung in die Begriffswelt der Datenbanktechnologie. Besonderer Wert wurde auf die Darstellung der Methoden zum Datenbankentwurf gelegt. Es wurde der Weg von der Bedarfsanalyse bis zum implementierfähigen konzeptionellen Schema aufgezeigt.

Eine systemtechnische Betrachtung der Montage wurde in Kapitel 4 durchgeführt. Die Montage wurde dazu auf ein soziotechnologisches Modell abgebildet, welches eine klare Abgrenzung der einzelnen an der Montage beteiligten Komponenten ermöglicht.

Die Ergebnisse dieser Systemanalyse dokumentieren, daß die Systemtechnik ein unerläßliches Hilfsmittel darstellt, um zu einer strukturierten Beschreibung eines so komplexen Problemkreises, wie ihn die Montage darstellt, zu gelangen.

In Kapitel 5 wurde der Ablauf der systematischen Montageanlagenplanung analysiert. Um die Lücke in der systematischen Montageplanung zwischen der Beschreibung der Montageaufgabe und der Erstellung des Montagelayouts zu schließen, wurde ein neues Verfahren zur Montageablaufbeschreibung entwickelt. Dieses Verfahren zeichnet sich dadurch aus, daß anhand einer operationalen Beschreibung des Montageablaufs direkt, geeignete Wirkorgane und Geräte zur Durchführung der Montage, aus einer Datenbasis ausgewählt werden können.

Diese beiden Kapitel bildeten die Grundlage für die Entwicklung eines konzeptionellen Schemas, mit dem der gesamte Ablauf der Montageplanung, wie er in Kapitel 5 dargestellt wurde, unterstützt werden kann. Es wurden zunächst semantische Schemata für die einzelnen Planungsschritte entworfen und diese dann zu einem gemeinsamen konzeptionellen Schema integriert.

Zum Abschluß der Arbeit wurden die Implementierung einer Datenbank nach diesem konzeptionellen Schema und exemplarische Realisierungen von Anwenderprogrammen, die mit dieser Datenbank arbeiten, vorgestellt.

Die Arbeit hat gezeigt, daß schon durch die für die Vorbereitung eines Datenbankeinsatzes erforderliche systematische Betrachtung der Montage wesentliche Erkenntnisse zur Montageplanung gewonnen werden können. Zudem kann auch mit derzeit verfügbaren Datenbankmanagementsystemen, trotz der in Kapitel 3 angesprochenen Unzulänglichkeiten, eine wirkungsvolle Unterstützung der rechnergestützten Montageplanung erreicht werden, indem der Eingabeaufwand für einzelne Anwendungsprogramme erheblich reduziert und Fehler durch redundante Datenspeicherung weitgehend vermieden werden.

9. Literaturverzeichnis

1. Gairola, A.;
Voigt, B.: Maßnahmenkatalog für das montagegerechte Konstruieren, VDI-Berichte 556, Automatisierung der Montage in der Feinwerktechnik, VDI-Verlag 1985, S.61-84
2. Feldmann, K.: Rechnerintegrierte Montagesysteme, Fachtagung "Rechnerintegrierte Produktionssysteme", Erlangen 27.-28. Okt. 1987, S.279-294
3. Milberg, J.;
Diess, H.: Optimierung der Montagetechnik durch rechnergestützte Planungssysteme, Zwf 82 (1987) 4, S.190-195
4. Feldmann, K.: Rechnergestützte Planungsmethoden für Montagesysteme, Kolloquium automatische Produktionssysteme, München 14.-15. Febr. 1985, S.414-426
5. Meier, A.: Erweiterung relationaler Datenbanksysteme für technische Anwendungen, Habilitationsschrift ETH, Zürich 1986
6. Härder, Th.
et al: KUNICAD - ein datenbankgestütztes geometrisches Modellierungssystem für Werkstücke, Informatik Forsch. Entw. (1987)2, S.1-18
7. Wedekind, H.: Datenbanksysteme I, Bd.I, 2. Auflage, Mannheim 1981
8. Schlageter, G.;
Stucky, W.: Datenbanksysteme: Konzepte und Modelle, Stuttgart 1983
9. Zehnder, C.A.: Informationssysteme und Datenbanken, Teubner, Stuttgart 1985
10. Dittrich, K.R.
et al: Datenbankunterstützung für den ingenieurwissenschaftlichen Entwurf, Informatik-Spektrum (1985)8, S.113-129
11. Rebsamen, J.: Datenbankentwurf im Dialog - Integrierte Beschreibung von Strukturen, Transaktionen und Konsistenz, Diss. ETH Zürich 1983
12. Vinek, G.;
Rennert, R.;
Tjou, A.: Datenmodellierung: Theorie und Praxis des Datenbankentwurfs, Würzburg 1982
13. ANSI/X3/SPARC: Study Group on Data Base Management Systems, Interim Report 75-02-08 in FDT Bullerin ACM-SIGMOD Vol.7, Nr.2, 1975

14. Lockemann, P.C.; Schmidt, J.W. (Hrsg): Datenbank-Handbuch, Springer-Verlag, Berlin 1987
15. Wedekind, H.: Supporting the design of conceptual schemata by database systems, Int. Conf. on DATA ENGINEERING 1984, IEEE Computer Society Order Nr.533, S.434-438
16. Tsichritzis, D.C.: Office Automation: Concepts and Tools, Springer Berlin 1985
17. Eigner, M.: Semantische Datenmodelle als Hilfsmittel der Informationshandhabung in CAD-Systemen und deren programmtechnische Realisierung auf Kleinrechnern, Fortschritts-Bericht VDI-Z, Reihe 10, Nr.9, 1980
18. Brodie, M.L.: On the Development of Data Models in [19] S.19-48
19. Brodie, M.L.; Mylopoulos, J.; Schmidt, J.W. (Hrsg): On conceptual modelling: perspectives from artificial intelligence, databases and programming languages, Berlin, Springer Verlag 1984
20. Date, C.J.: An Introduction to Database Systems, Fourth Edition, Addison-Wesley Publ. Comp. 1986
21. Eberlein, W.: Architektur technischer Datenbanken für integrierte Ingenieursysteme; Diss. Uni Erlangen 1984
22. CODASYL SYSTEM COMMITTEE: Feature Analysis of Generalized Data Base Management Systems, May 1971
23. Codd, E.F.: A relational model for large shared data banks, in: Comm. ACM, Vol.13 (1970), No.6, S.377-387
24. Härder, Th.: Implementierung von Datenbanksystemen, Carl Hanser Verlag, München 1978
25. Gibbs, S.J.: Conceptual Modelling and Office Information Systems, in [16], S.193-228
26. Lamersdorf, W.: Semantische Repräsentation komplexer Objektstrukturen, Diss. Uni Hamburg 1985
27. Ortner, E.: Semantische Modellierung - Datenbankentwurf auf der Ebene der Benutzer Informatik-Spektrum (1985)8, S.20-28

28. Eberlein, W.;
Wedekind, H.: At Methodology for Embedding Design
Databases into Integrated Engineering
Systems, Proc. of the IFIP WG 5.2,
Working Conference on File Structures
and Data Bases for CAD, Sept. 81,
S.3-37
29. Grabowski, H.;
Eigner, M.: Database Description with SMD: A Semantic
Database Model, Proc. of the IFIP
WG 5.2 Working Conference on File
Structures and Data Bases for CAD,
Sept. 81, S.117-144
30. Bracchi, G.;
Paolini, P.;
Pelagatti, G.: Binary Logical Associations in Data
Modelling, in Modelling in Data Base
Management Systems, Proc. of the IFIP
Working Conference, Freudenstadt 1976,
North-Holland Publ. Comp. 1976,
S.125-148
31. Senko, M.E.: Data Description Language in the Con-
text of a Multilevel Structured Des-
cription: DIAM II with FORAL, in Proc.
"Data Base Description" North-Holland
Publ. Comp. 1977
32. Chen, P.P.: The entity-relationship model: to-
wards an unified view of data, Tran-
sactions on Database System, 1,11
(1976), S.77-84
33. Smith, J.M.;
Smith, D.C.P.: Database Abstraction, Aggregation and
Generalization, in ACM Transaction on
Database Systems Vol.2, No.2 (1977),
S.105-133
34. Codd, E.F.: Extending the Database Relational Model
to Capture More Meaning, ACM Tran-
sactions on Database Systems, Vol.4,
No.4, 1979, S.397-434
35. Thurnherr, B.: Konzepte und Sprachen für den Entwurf
konsistenter Datenbanken, Diss. ETH
Zürich 1980
36. Hammer, M.;
McLeod, D.: Database Description with SDM: A Semantic
Database Model, ACM Transactions
on Database Systems, Vol.6, No.3, 1981,
S.351-386
37. Furtado, A.L.;
Neuhold, E.J.: Formal Techniques for Data Base Design,
Springer Verlag, Berlin 1986
38. Chen, P.P.
(Hrsg.): Entity-relationship approach to systems
analysis and design, North Holland
(1980)

39. Chen, P.P. (Hrsg.): Entity-relationship approach to information modelling and analysis, ER-Institute (1981)
40. Davis, C.G. et al: Entity-relationship approach to software engineering, North Holland 1983
41. Chen, P.P. (Hrsg.): Entity-relationship approach: the use of ER concept in knowledge representation, fourth Int. Conf. on Entity-Relationship Approach (Chicago, Okt. 1985), Washington D.C.: IEEE Computer Society Press 1985
42. Gebhardt, F.: Semantisches Wissen in Datenbanken, Informatik-Spektrum (1987)10, S.79-98
43. Schiel, K.: Ein semantisches Datenmodell für konzeptionelle Schemata und ihre Abbildung auf interne relationale Schemata, Diss. Uni Stuttgart, 1984
44. Brodie, M.L.: Association: A Database Abstraction for Semantic Modelling, in [39]
45. Steinbauer, D.: Transaktionen als Grundlage zur Strukturierung und Integritätssicherung in Datenbank-Anwendungssystemen, Diss. Uni Erlangen 1983
46. Steinbauer, D.; Wedekind, H.: Integritätsaspekte in Datenbanksystemen, Informatik-Spektrum (1985)8, S.60-68
47. Teorey, T.J.; Frey, J.P.: Design of Database Structures, Prentice Hall 1982
48. Batini, C.; Lenzerini, M.; Navathe S.B.: A Comparative Analysis of Methodologies for Database Schema Integration, ACM Computing Surveys, Vol.18, No.4, December 1986, S.323-364
49. Agosti, M.; Johnson, R.G.: A framework of reference for database design, Database, Summer 1984, Vol.15, No.4, S.3-9
50. Agosti, M.: Database design: a classified and annotated bibliography, Cambridge University Press 1986
51. Ferrare, F.M.; Batini, C.: GDOL: A tool for computerized design and documentation of database systems, Database Summer 1984, Vol.15, No.4, S.15-20

52. Yao, S.B.
et al (Hrsg.): Data Base Design Techniques I: Requirements and Logical Structures Lecture Notes, in Computer Science, No.132, Springer Verlag, Berlin 1982
53. Yao, S.B.
et al: An Integrated Approach to Database Design, in [52] S.1-30
54. Krieger, R.;
Lausen, G.;
Stucky, W.: Methodischer Datenentwurf beim Aufbau betrieblicher Informationssysteme, Angewandte Informatik 3/82, S.191-199
55. Schiel, U.;
Furtado, A.L.: Towards Multi-Level and Modular Conceptual Schema Specifications, Information Systems 1984, Vol.9, No.1, S.43-57
56. Ortner, E.: Aspekte einer Konstruktionsprache für den Datenbankentwurf, Darmstadt 1983
57. Wedekind, H.: Systematisches Konstruieren von Datenbankanwendungen, München 1980
58. Vetter, P.;
Maddison, R.N.: Data Base Design Methology, Prentice Hall, New York 1981
59. Tsichritzis, D.C.;
Lochovsky, F.H.: Data Models, Prentice-Hall 1982
60. Schön, D.: Die Einbindung heterogener rechnergestützter Entwicklungssysteme in die technische Dokumentation, Diss. Uni Erlangen 1987
61. Härder, Th.;
Reuter, A.: Architektur von Datenbanksystemen für Non-Standard-Anwendungen, in Informatik-Fachberichte Band 94: Datenbank-Systeme für Büro, Technik und Wirtschaft, Proc. GI-Fachtagung, Karlsruhe, 20.-22. März 1985, S.253-286
62. Lockemann, P.C.
et al: Anforderungen technischer Anwendungen an Datenbanksysteme, in Informatik-Fachberichte Band 94: Datenbank-Systeme für Büro, Technik und Wirtschaft, GI-Fachtagung, Karlsruhe 1985, S.1-26
63. Stonebraker, M.
et al.: QUEL AS A DATA TYPE, ACM SIGMOD, Boston 1984, S.208-214
64. Dadam, P.
et al.: Objektorientierte Datenhaltung für die Roboterprogrammierung, Informatik Forsch. Entw. (1987) 2, S.151-170

65. Schek, H.-J.;
Weikum, G.: DASDBS Konzepte und Architektur eines neuartigen Datenbanksystems, Informatik Forsch. Entw. (1987) 2, S.105-121
66. Gotthard, W.
et al.: Das Projekt DAMOKLES und sein Entwurfsobjekt-Datenmodell, in "Die Zukunft der Informationssysteme", Fachtagung ÖGI und GI in Linz, Sept. 1986, S.277-296
67. Wedekind, H.: Die Problematik des Computer Integrated Manufacturing (CIM), Informatik Spektrum (1988) 11, S.29-39
68. Beitz, W.: Systemtechnik im Ingenieurbereich, VDI-Berichte Nr. 173, 1971, S.5-12
69. Patzak, G.: Systemtechnik - Planung komplexer innovativer Systeme, Springer Verlag, Berlin 1982
70. Ropohl, G.: Systemtechnik - Grundlagen und Anwendung, Hanser Verlag, München 1975
71. Hubka, V.;
Ropohl, G.: Was ist ein technisches System?, VDI-Zeitung, Bd.128 (1986), Nr.22 Nov.(II), S.869-874
72. Hubka, V.: Theorie Technischer Systeme, Springer Verlag, Berlin 1984
73. Müller, H.: Gibt es ein ideales Prozeßmodell für das konstruktive Entwickeln technischer Objekte?, Maschinenbautechnik 22 (1973), Heft 5, S.230-232
74. Beitz, W.: Übersicht über Konstruktionsmethoden, Konstruktion 24 (1972), S.68-72
75. Stachowiak, H.: Allgemeine Modelltheorie; Springer - Verlag; Wien/New York 1973
76. Wedekind, H.
(Hrsg.): Untersuchungen über den Einsatz von Datenbanken in Flexiblen Fertigungssystemen, Arbeitsberichte des IMMD der FAU Erlangen, Bd. 20, Nr.2, Erlangen, März 1987
77. Zörntlein, G.: Flexible Fertigungssysteme: Belegung, Steuerung, Datenorganisation, Diss. Uni. Erlangen 1987
78. Dilling, H.-J.: Methodisches Rationalisieren von Fertigungsprozessen am Beispiel montage-rechter Produktgestaltung, Diss. TH Darmstadt 1978

79. Classe, D.; Scholz, W.: Verfahrensbeschreibende Parameter - Schlüssel zur systematischen Montageautomatisierung, wt Werkstattstechnik 77 (1987) S.684-689
80. Warnecke, H.-J.; Weiss, K.: Zubringeeinrichtungen, Krausskopf Mainz 1978
81. Spur, G. (Hrsg.): Handbuch der Fertigungstechnik, Bd.5: Fügen, Handhaben und Montieren, München 1986
82. VDI Richtlinie 2860: Handhabungsfunktionen, Handhabungseinrichtungen, Blatt 1: Begriffe, Definitionen, Symbole, Entwurf Oktober 1982
83. DIN 8593: Fertigungsverfahren Fügen; Einordnung, Unterteilung, Begriffe, Juni 1984
84. Lotter, B.: Wirtschaftliche Montage, Ein Handbuch für Elektrogerätebau und Feinwerktechnik, VDI-Verlag Düsseldorf 1986
85. Koller, R.: Konstruktionslehre für den Maschinenbau, Berlin 1985
86. Roth, K.: Konstruieren mit Konstruktionskatalogen, Berlin Springer-Verlag 1982
87. Reuleaux, F.: Theoretische Kinematik, 2 Bde., Vieweg, Braunschweig 1875/1900
88. Franke, R.: Vom Aufbau der Getriebe, 2 Bde., VDI-Verlag, Düsseldorf 1951/58
89. Claussen, U.: Konstruieren mit Rechnern, Springer Verlag, Berlin/Heidelberg/New York 1971
90. Baatz, U.: Bildschirmunterstütztes Konstruieren; Funktionsfindung, Prinzipiarbeitung, Gestaltung und Detaillierung mit Hilfe graphischer Datenverarbeitungsanlagen, Diss. TH Aachen 1971
91. Miese, M.: Systematische Montageplanung in Unternehmen mit Kleinserienproduktion, Giradet Verlag 1976
92. Frank H.E.: Das Verhalten von Werkstücken bei automatisierter Handhabung in der Fertigung, Diss. Uni Stuttgart 1974
93. Löhr, H.G.: Eine Planungsmethode für automatische Montagesysteme, Diss. Uni Stuttgart 1976

94. Herrmann, G.: Analyse von Handhabungsvorgängen im Hinblick auf deren Anforderungen an programmierbare Handhabungsgeräte in der Teilefertigung, Diss. Uni. Stuttgart 1976
95. Schimke, E.W.: Planung von Handhabungssystemen, Diss. TH Aachen 1976
96. Metzger, H.: Planung und Bewertung von Arbeitssystemen in der Montage, Diss. Uni. Stuttgart 1977
97. Schraft, R.D.: Systematisches Auswählen und Konzipieren von programmierbaren Handhabungsgeräten, 1977 ISBN-3-7830-0115-3
98. Hoeschen, R.D.: Planung von Montagesystemen im Rahmen der technischen Investitionsplanung, Diss. TH Aachen 1978
99. Hirschbach, O.: Rechnergestützte Montageplanerstellung, Stuttgart Krausskopf 1978
100. Scholz, W.;
Classe, D.: Modell zur Beschreibung von Montageaufgaben, Arbeitsvorbereitung AV 25, Juli 1988, Heft 4 S.136-139
101. Scholz, W.: Technisches Informationssystem und Entwurf integrierter Ingenieurdatenbanken, Zwischenbericht an die DFG zum Forschungsvorhaben We 763/7-1, Aug. 1985
102. Weule, H.;
Friedmann, Th.: Rechnergestützte Produktanalyse in der Montageplanung, VDI-Z Bd.129 (1987) Nr.12, S.59-63
103. Gairola, A.: Montagegerechtes Konstruieren; Ein Beitrag zur Konstruktionsmethodik, Diss. Uni. Darmstadt 1981
104. Mertens, P.: Industrielle Datenverarbeitung, Bd.1, Administrations- und Dispositionssysteme, Gabler 1982
105. Altenwerth F.
et al.: Arbeitssysteme mit integrierten Handhabungsgeräten; BMFT Schriftenreihe: Humanisierung des Arbeitslebens Bd.56, VDI-Verlag Düsseldorf 1984
106. Warnecke, H.-J.;
Schraft, R.: Handbuch Handhabungs-, Montage- und Industrierobotertechnik Bd. I-III, Verlag moderne Industrie, Landsberg am Lech 1984

107. Hemberger, A.: Innovationspotentiale in der rechnerintegrierten Produktion durch wissensbasierte Systeme, Diss. Uni Erlangen 1987
108. Schweizer, M. et al.: Methodenentwicklung zur Taktzeitprognose bei Robotern, CAE-Journal 2/87, S.56-60
109. Warnecke, H.-J. et al.: Taktzeitprognose für flexible Montagestationen, VDI-Z Bd.129 (1987) Nr.3, S.53-56
110. Feldmann, K.; Eisele, R.; Kleineidam, G.: Verfahrenskette zur Planung und Programmierung von Montagesystemen, ZWF 82 (1987) 9, S.521-527
111. Schlüter, K.: Planung einer flexiblen Montagestraße für Schaltgeräte mit Hilfe des Simulators GPSS-FORTRAN Version III, Angewandte Informatik 8/9 (87), S.340-350
112. Schlüter, K.: Simulation als Hilfsmittel für Planung und Betrieb von Produktionssystemen, atp 29.Jahrgang, Heft 9/1987, S.416-422
113. Janisch, H.-W.: Optimierung der Puffer bei elastisch verketteten Fertigungssystemen, Diss. Uni Hannover 1979
114. Olfert, K.: Investitionen, Ludwigshafen 1982
115. Warnecke, H.-J. et al.: Wirtschaftlichkeitsrechnung für Ingenieure, München - Wien 1980
116. Streit, W.: Entwurf und Implementierung eines konzeptionellen Schemas für eine Datenbank zur Unterstützung der Montageanlagenplanung, Studienarbeit am Lehrstuhl für Fertigungsautomatisierung und Produktionssystematik, UNI. Erlangen 1987
117. Yau, Ch. K.: Entwurf eines konzeptionellen Schemas zum Aufbau einer Informationsdatenbank für automatische Montageanlagen, Studienarbeit am Lehrstuhl für Fertigungsautomatisierung und Produktionssystematik, UNI. Erlangen 1986

118. Rister, R.: Konzeption und Implementierung einer Datenbank für Komponenten automatischer Montageanlagen mit dem Datenbanksystem UDS in Verbindung mit dem CAD-System CADIS, Diplomarbeit am Lehrstuhl für Fertigungsautomatisierung und Produktionssystematik, UNI. Erlangen 1986
119. Rohde, J.: Entwurf und Implementierung eines Systems zur datenbankgestützten, graphischen Beschreibung von Montageaufgaben", Studienarbeit am Lehrstuhl für Fertigungsautomatisierung und Produktionssystematik, UNI. Erlangen 1988
120. Katz, R.H.;
Lehman, T.J.: Database Support for Versions and Alternatives of Large Design Files, IEEE Transactions on Software Engineering, Vol.10, No.2, 1984 S.191-200
121. Wedekind, H.;
Müller, T.: Stücklistenorganisation bei einer großen Variantenzahl, Angewandte Informatik 9/81, S.377-383
122. Grupp, B.: Elektronische Stücklistenorganisation in der Praxis, Forkel-Verlag, Stuttgart 1976
123. Hackstein, R.: Produktionsplanung und Steuerung, VDI-Verlag Düsseldorf 1984
124. Grupp, B.: Stücklisten- und Arbeitsplanorganisation mit Bildschirmeinsatz Forkel-Verlag, Wiesbaden 1985
125. Müller, T.;
Steinbauer D.;
Wedekind, H.: Control of Versions of Database Application, IBM Germany TR 83.09.003, Heidelberg 1983
126. Date, C. J.: A guide to INGRES, Addison-Wesley Publishing Company 1987
127. Wagner, N.: Erstellung eines Montagevorranggraphen durch Auswertung des Produktaufbaus im CAD, Studienarbeit am Lehrstuhl für Fertigungsautomatisierung und Produktionssystematik, UNI. Erlangen 1988
128. Frank, A.: Erweiterung eines CAD-Modells um verfahrensbeschreibende Parameter, Studienarbeit am Lehrstuhl für Fertigungsautomatisierung und Produktionssystematik, UNI. Erlangen 1988

Lebenslauf

- 1955 Wolfgang Scholz, geb. am 25. Oktober 1955
in Erlangen
Eltern: Gerhard Scholz, verstorben am
9.3.1959
Sabine Scholz, geb. Böhm
- 1962 - 1968 Besuch der Volksschule in Ebrach
- 1968 - 1972 Besuch der Realschule in Ebrach
- 1972 - 1974 Besuch der Fachoberschule in Bamberg
- 1974 - 1978 Studium der Fachrichtung Maschinenbau,
Studienschwerpunkt Entwicklung und
Konstruktion, an der Fachoberschule
Würzburg - Schweinfurt in Schweinfurt
Ingenieurprüfung und Hochschulreife
am 28. Juli 1978
- 1978 - 1980 Ingenieur in der Abteilung Entwicklung/
Planung der Fa. Gg. Noell GmbH, Würzburg
- 1980 - 1983 Studium des Maschinenwesens, Studienrich-
tung Theorie und Forschung, an der
Technischen Universität in München
Diplomhauptprüfung am 25. November 1983
- 1982 - 1983 Wissenschaftliche Hilfskraft am Lehrstuhl
für Förderwesen und am Lehrstuhl für
Maschinenelemente der TU München
- Ab 1984 Wissenschaftlicher Mitarbeiter am Lehrstuhl
für Fertigungsautomatisierung und
Produktionssystematik des Instituts für
Fertigungstechnik der Universität Erlangen-
Nürnberg in Erlangen
Lehrstuhlinhaber: Prof. Dr.-Ing. K. Feldmann

Reihe Fertigungstechnik Erlangen

Band 1

Andreas Hemberger
**Innovationspotentiale in der rechnerintegrierten Produktion durch
wissensbasierte Systeme**
208 Seiten, 107 Bilder. 1988. Kartoniert.

Band 2

Detlef Classe
**Beitrag zur Steigerung der Flexibilität automatisierter Montage-
systeme durch Sensorintegration und erweiterte Steuerungskonzepte**
194 Seiten, 70 Bilder. 1988. Kartoniert.

Band 3

Friedrich-Wilhelm Nolting
Projektierung von Montagesystemen
201 Seiten, 107 Bilder, 1 Tabelle. 1989.
Kartoniert.

Band 4

Karsten Schlüter
**Nutzungsgradsteigerung von Montagesystemen durch den Einsatz
der Simulationstechnik**
177 Seiten, 97 Bilder. 1989. Kartoniert.

Band 5

Shir-Kuan Lin
Aufbau von Modellen zur Lageregelung von Industrierobotern
168 Seiten, 46 Bilder. 1989. Kartoniert.

Band 6

Rudolf Nuss
**Untersuchungen zur Bearbeitungsqualität im Fertigungssystem
Laserstrahlschneiden**
206 Seiten, 115 Bilder, 6 Tabellen. 1989. Kartoniert.

Band 7

Wolfgang Scholz
**Modell zur datenbankgestützten Planung automatisierter
Montageanlagen**
194 Seiten, 89 Bilder. 1989. Kartoniert.