



Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

Document

D-96-03

Der DESIGN-ANALYZER – Decision Support im Designprozeß

Winfried Tautges

Mai 1996

**Deutsches Forschungszentrum für Künstliche Intelligenz
GmbH**

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

Stuhlsatzenhausweg 3
66123 Saarbrücken, FRG
Tel.: + 49 (681) 302-5252
Fax: + 49 (681) 302-5341

Deutsches Forschungszentrum für Künstliche Intelligenz

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

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

- Intelligent Engineering Systems
- Intelligent User Interfaces
- Computer Linguistics
- Programming Systems
- Deduction and Multiagent Systems
- Document Analysis and Office Automation.

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

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

Dr. Dr. D. Ruland
Director

Der DESIGN-ANALYZER – Decision Support im Designprozeß

Winfried Tautges

DFKI-D-96-03

This work has been supported by a grant from The Federal Ministry of Education, Science, Research and Technology (FKZ ITW-9304/3).

© Deutsches Forschungszentrum für Künstliche Intelligenz 1996

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

Der DESIGN-ANALYZER – Decision Support im
Designprozeß

Winfried Tautges

9. Mai 1996

Zusammenfassung

Die Entwicklung moderner Industrieprodukte erfordert schon in der Designphase viele frühzeitige Entscheidungen mit weitreichenden Konsequenzen. Verschiedenartige Informationsquellen wie fachspezifische Informationssysteme, Fachexperten und die Anforderungsliste des Kunden stellen sowohl die notwendigen Daten als auch die Bewertungsgrundlage für eine optimale Designentscheidung bereit. Die Arbeit stellt einen Ansatz zur Unterstützung der Auswahl eines besten Konstruktionsvorschlags unter Berücksichtigung der verschiedenen Informationsquellen vor. Die Formalisierung der Produktanforderungen stützt sich auf Methoden des Maschinenbaus. Alle kompetenten Experten und relevanten Daten werden in einer Mehr-Kriterien-Gruppenentscheidung integriert. Der Entscheidungsalgorithmus basiert auf einer Nutzwertanalyse aus der klassischen Entscheidungstheorie. Die Implementation des DESIGN-ANALYZERS orientiert sich an der Dreiteilung von entscheidungsunterstützenden Systemen nach dem DDM-Paradigma.

Modern industrial product development necessitates already during the design phase lots of early decisions with farreaching consequences. Various information resources like domain specific information systems, technical experts, and customer's product requirements contribute both the necessary data and the valuation background for the optimal design. This document presents a basic approach to support the selection of a best design suggestion using the different information resources. The formalization of the product requirements is based on engineering methods. All competent experts and all relevant data are integrated in a multi-criteria multi-participants decision. The decision algorithm is based on an utility analysis taken from classical decision theory. The implementation of the DESIGN-ANALYZER follows the trisection of decision support systems according to the DDM-Paradigma.

Inhaltsverzeichnis

1	Einleitung	3
2	Der Design-Manager - Idee und Konzept	7
2.1	Das IMCOD-Projekt	7
2.2	Die Domäne und der Bereich der Entscheidungsunterstützung	8
2.3	Annahmen	11
3	Entscheidungstheoretische und begriffliche Grundlagen	13
3.1	Entscheidungstheoretische Modelle	13
3.2	Die Entscheidung als Prozeß	16
3.3	Decision Support Systeme	18
4	Die DesignSpezifikation	23
4.1	Das Pflichtenheft des Auftraggebers	23
4.2	Anforderungstypen	23
4.3	Die Abbildung auf Kriterien	25
4.4	Der domänenspezifische Kriterienkatalog	27
4.4.1	Mögliche Erweiterung durch COKAM+	27
5	Die Domänenspezifikation	31
5.1	Basisstruktur	31
5.2	Die Experten	32
5.3	Die Interface-Bibliothek	33
6	Der Entscheidungsalgorithmus	35
6.1	Isolierte Betrachtung	35
6.1.1	Verknüpfung der entscheidungsrelevanten Daten	37
6.1.2	Die Nutzwertkalkulation	39
6.2	Integration in den Produktentwurfsprozeß	43
6.2.1	Darstellung der Konstruktionsvorschläge	45

7	Implementierung	47
7.1	Daten	47
7.1.1	Datenspeicherung	47
7.1.2	Datenverknüpfungen aus DB-Sicht	48
7.2	Nutzwertkalkulation	50
8	Die Benutzeroberfläche	53
8.1	IMCOD-Console	53
8.2	DOMAIN-SPEC-Window	54
8.3	DESIGN-SPEC-Window	54
8.4	Value-Editor	57
8.5	RatingAndRanking-Window	58
8.5.1	Algorithmus einer Beispielsitzung	59
9	Zu guter Letzt	63
9.1	Berechnung statt Entscheidung?	63
9.2	Ausbaumöglichkeiten und andere Groupware-Ansätze	64
9.3	Zusammenfassung	66
A	Pflichtenheft	67
B	Designspezifikation	69

Abbildungsverzeichnis

1.1	Kostenrelevanz der Entscheidungen im Produktlebenszyklus	4
2.1	Die Vermittlerrolle des IMCOD-Managers	8
2.2	Struktur des Produktionsprozesses	10
3.1	Die Abbildung eines Realsystems in ein Modell	17
3.2	Der Entscheidungsprozeß als Problemlösungsprozeß	18
3.3	Die Ausrichtung von DSS auf semistrukturierte Entscheidungsprobleme . .	19
3.4	Das DDM-Modell eines DSS	20
4.1	Anforderungstypen	25
4.2	Die Abbildung relevanter Umweltausschnitte auf Kriterien	28
5.1	Kategorienzugehörigkeit der Kriterien und Experten	32
5.2	Die Architektur der Domänenspezifikation	34
6.1	Der Entscheidungsalgorithmus des Design-Analyzers	36
6.2	Bedeutung der Kriterientypen für die Entscheidung	39
6.3	Einflußparameter des Entscheidungsprozesses	42
6.4	Der Design-Analyzer integriert in den Produktionsprozeß	45
7.1	Die Verknüpfung der Kriterien, Experten und Konstruktionsvorschläge aus Datenbanksicht	48
8.1	Konsole des DESIGN-ANALYZERS	53
8.2	Domain Specification Window mit Kategorie-Auswahl-Fenster und Interface- Bibliothek-Editor	55
8.3	Designspecification-Window des DESIGN-ANALYZERS	56
8.4	Checking-Window mit fehlenden domänenspezifischen Kriterien	56
8.5	ValueEditor zum editieren der Kriterien	57
8.6	Rating-Window des DESIGN-ANALYZERS	58

Kapitel 1

Einleitung

Unsere Gesellschaft befindet sich in einer Übergangsphase von einer überschaubaren Industrie- zu einer komplexen Informationsgesellschaft. Gute Produkte sind nicht mehr nur auf eine geniale Idee eines Erfinders und eine zielsichere Entscheidung eines Abteilungsleiters oder Firmenmanagers zurückzuführen. Vielmehr erarbeiten viele Experten Teilbereiche eines komplexen Produkts, und mehrere Experten sind notwendig, um Entscheidungen bezüglich der Auswahl einer Produktalternative zu treffen. Schon kleine Fehler oder falsche, emotional bedingte Favorisierung einer Handlungsalternative in dieser Entscheidungssituation können darüber bestimmen, ob ein Produkt ein "Renner" oder ein "Fehlgriff" auf dem konkurrenzstarken Markt wird. Oft erlangt die Firma den entscheidenden Vorsprung, die am besten und schnellsten über Produktionsdaten informiert ist und die bei ihren Entscheidungen die größte Expertise vorzuweisen hat. Im Maschinenbau spielt der Konstruktionsprozeß im gesamten Produktlebenszyklus eine entscheidende Rolle. Auch wenn er vom Kostenstandpunkt gesehen nur einen relativ kleinen Kostenfaktor bestimmt, so ist er doch aufgrund der durch ihn getroffenen Entscheidungen für 80 Prozent der gesamten Produktionskosten "verantwortlich". Sollte eine falsche Entscheidung in der Konstruktionsphase erst in der Fertigungsphase aufgedeckt und revidiert werden müssen, entstehen oft erhebliche Kosten durch Überarbeitung der Konstruktion bis hin zu Rücknahmeaktionen. So müssen im Maschinenbau zum Beispiel teure Fertigungsanlagen umgerüstet werden oder komplette, schon verkaufte Serien aus versicherungstechnischen Gründen zurückgenommen und ersetzt werden. Abbildung 1.1 zeigt schematisiert die Einordnung der Kostenrelevanz in den typischen Produktlebenszyklus.

Obwohl nun die Produktivität des einzelnen Entscheidungsträgers durch verbesserte Informationstechnologie erheblich gesteigert werden konnte, treten bei Mehrpersonenentscheidungen (Gruppenentscheidungen) oft die Dominanz bestimmter Mitglieder, sozialer Gruppendruck oder Geltungsbedürfnisse als Störfaktor einer sachlichen Entscheidungsfindung auf. Moderne Informationstechnologie sollte diesen Einfluß reduzieren und durch effiziente Kombination von Computer, Kommunikation und Decision Support Technologie den Produktionsprozeß optimieren. Wichtig ist in diesem Zusammenhang auch die optimale Ausnutzung des vorhandenen Expertenwissens und die Wiederverwendung der gewonnenen Ergebnisse aus früheren Entscheidungssituationen.

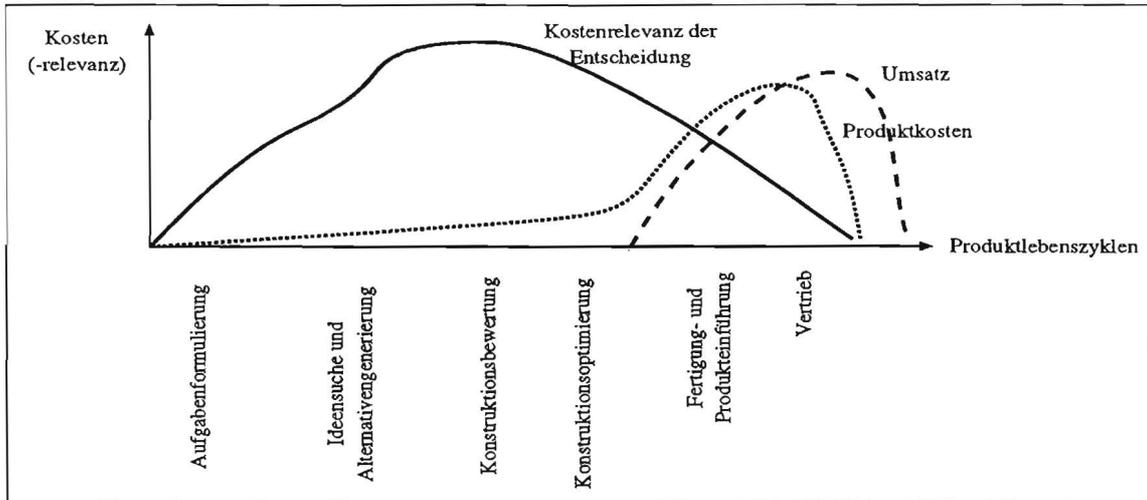


Abbildung 1.1: Kostenrelevanz der Entscheidungen im Produktlebenszyklus

In der vorliegenden Arbeit wird die zweite überarbeitete Version¹ des DESIGN-ANALYZERs vorgestellt, der unter Berücksichtigung dieser Anforderungen die Bewertungsphase von Konstruktionsalternativen im Maschinenbau unterstützt. Das Tool ist Teil des im IMCOD-Projekt am DFKI (Deutsches Forschungszentrum für Künstliche Intelligenz GmbH, Kaiserslautern) entwickelten IMCOD-MANAGERS (siehe Kapitel 2). Das Ziel des Projekts ist es, durch optimale Ressourcenausnutzung vorhandenen Expertenwissens im Designprozeß einen effizienteren und qualitativ höheres Designergebnis zu erzielen. Im Kernpunkt des Projekts und vornehmlich in dieser Arbeit steht dabei das Zusammenspiel vom Computer als formales Informations- und Regelsystem und dem Menschen mit seiner globalen Sichtweise auf das Problem.

Gliederung

Diese Arbeit setzt sich aus den folgenden Kapiteln zusammen:

Kapitel 2

In diesem Kapitel wird IMCOD kurz vorgestellt, um dann die Einordnung der Arbeit in das Projekt vornehmen zu können. Außerdem werden die Voraussetzungen für die praktische Realisierung des entwickelten Konzepts aufgelistet.

Kapitel 3

Entscheidungstheoretische Grundlagen und Begriffe werden in diesem Kapitel kurz

¹Die erste Version des DESIGN-ANALYZER wurde im Zuge einer Diplomarbeit von dem Autor implementiert und vorgestellt. Das vorliegende Dokument ist daher eine Überarbeitung der Diplomarbeit "Rechnerunterstützte Bewertung von Konstruktionsvorschlägen im Maschinenbau mit flexibel gestaltbarer Domänenspezifikation" in den wesentlichen Bereichen der Anforderungs- und Kriterienstrukturierung und der Nutzwertanalyse, Kapitel 4 und 6.

(relativ zur Größe dieser Wissenschaft) erläutert, um das Konzept der Arbeit darstellen zu können. So wird unter anderem zwischen *echter* und *unechter* Entscheidung unterschieden und der Entscheidungsprozeß diskutiert. Darauf aufbauend schließt sich ein Exkurs über Decision Support Systeme und deren Einsatzbereich an.

Kapitel 4

Nicht zu trennen von der Designphase ist die Aufgabenformulierungsphase. Ausgehend von der Anforderungsliste des Auftraggebers wird die Formalisierung der Anforderungen in bewertbare Kriterien beschrieben. Ein anderes Tool des IMCOD-Projekts - MIKADO - ist dann Thema des letzten Abschnitts dieses Kapitels, da es sich zur Formalisierung der Anforderungen eignet.

Kapitel 5

Hier wird das flexibel gestaltbare Domänenmodell beschrieben. Durch diese Spezifikation kann das entscheidungsunterstützende System an Veränderungen der Anwendungsdomäne und veränderte Zusammensetzung des in der Gruppe vorhandenen Wissenspotential angepaßt werden. Außerdem wird auf die Schnittstellenbibliothek der integrierten Expertensysteme eingegangen, die im Zuge einer Diplomarbeit im IMCOD-Projekt entstand.

Kapitel 6

Die im Kapitel 4 beschriebene Designspezifikation und die Domänenspezifikation aus Kapitel 5 werden für den Entscheidungsprozeß kombiniert. Dieser Vorgang und der Bewertungsalgorithmus der Konstruktionsalternativen ist Thema des ersten Abschnitts von Kapitel 6. Im zweiten Teil wird dann der Entscheidungsalgorithmus integriert in den gesamten Designprozeß betrachtet.

Kapitel 7

Dieses Kapitel beschreibt wichtige Details und Zusammenhänge der Implementation des DESIGN-ANALYZER. Außerdem werden wichtige Smalltalkmethoden kurz vorgestellt und die Verknüpfung der Kriterien und des Domänenmodells aus Datenbanksicht beschrieben.

Kapitel 8

Die Benutzeroberfläche des DESIGN-ANALYZER wird hier zusammen mit den wichtigsten Funktionen vorgestellt. Außerdem wird ein grober Ablaufplan für einen Testlauf angegeben.

Kapitel 9

Kapitel 9 betrachtet das entwickelte Tool kritisch in Bezug auf den Grad und die Qualität von Computerunterstützung in Entscheidungsbereichen. Dabei orientiert sich die Kritik auch an Begriffen der Entscheidungstheorie aus Kapitel 3 und bildet einen inneren Rahmen dieser Schrift. Eine Zusammenfassung und einen Ausblick auf mögliche andere Ansätze oder Erweiterungen erfolgen abschließend in diesem letzten Kapitel.

Anhang A

Anhang A zeigt das Pflichtenheft des Auftraggebers, das gleichzeitig Grundlage für die Beispielsitzung der Mitarbeiter des Instituts für Verbundwerkstoffe und Testobjekt für die Funktionen des DESIGN-ANALYZERS war.

Anhang B

Anhang B zeigt die Designspezifikation mit den formalisierten Anforderungen des Pflichtenhefts aus Anhang A.

Kapitel 2

Der Design-Manager – Idee und Konzept

In diesem Kapitel wird zunächst eine kurze Beschreibung des IMCOD-Projekts gegeben, an die sich die Einordnung dieser Arbeit in das Projekt und die notwendigen Annahmen zur Realisierung anschließen.

2.1 Das IMCOD-Projekt

IMCOD ist das Akronym für Intelligent Manager for Comprehensive Design ([BBS96]). Der Begriff 'Manager' weist darauf hin, daß es sich hier um Kommunikation und Koordination zwischen Experten handelt, die an einer Aufgabe arbeiten. Die Aufgabe ist der Produktentwurf. Durch immer vielfältigere Anforderungen an moderne industrielle Produkte wird deren Entwurfsprozeß komplexer und berührt immer mehr Problembereiche. Diese Tatsache erweitert den Lösungsraum für gestellte Aufgaben, verlangt jedoch auch entsprechend leistungsfähigere Werkzeuge für die einzelnen Problembereiche und Expertenwissen, um Lösungsvorschläge zu erarbeiten und auch bewerten zu können.

Der Computer als Werkzeug stellt für den industriellen Bereich fast überall Expertensysteme und leistungsfähige Informationssysteme (lokale Experten) zur Verfügung. Für die technische Verbindung der verschiedenen Systeme existieren meist sehr praktikable Lösungen durch entsprechende Schnittstellenprotokolle. Die Systeme arbeiten jedoch meist mit unterschiedlichen Produktmodellaten und unterschiedlichen Programmiersprachen und damit unterschiedlichen Variablentypen, so daß Koordinations- und Kooperationsprobleme zwischen den eingesetzten Systemen oft ein reibungsloses, konsistentes Fortschreiten im Produktionsprozeß verhindern.

Im IMCOD-Projekt wird an Konzepten gearbeitet, die die lokalen Experten in ein homogenes System zur Unterstützung des Designprozesses integrieren, um ein optimales Designergebnis zu erzielen. Dabei muß die Möglichkeit bestehen, jederzeit neue lokale Experten integrieren zu können, wenn durch zusätzliche Bedingungen neue Problembereiche berührt werden. Ein prominentes Beispiel hierfür sind zum Beispiel Kriterien der Recyclierfähigkeit des Produkts. Abbildung 2.1 zeigt das Anwendungsszenario des IMCOD-Managers.

In IMCOD werden die iterativen Phasen der Designentscheidung (Suchphase, Entwurfs-

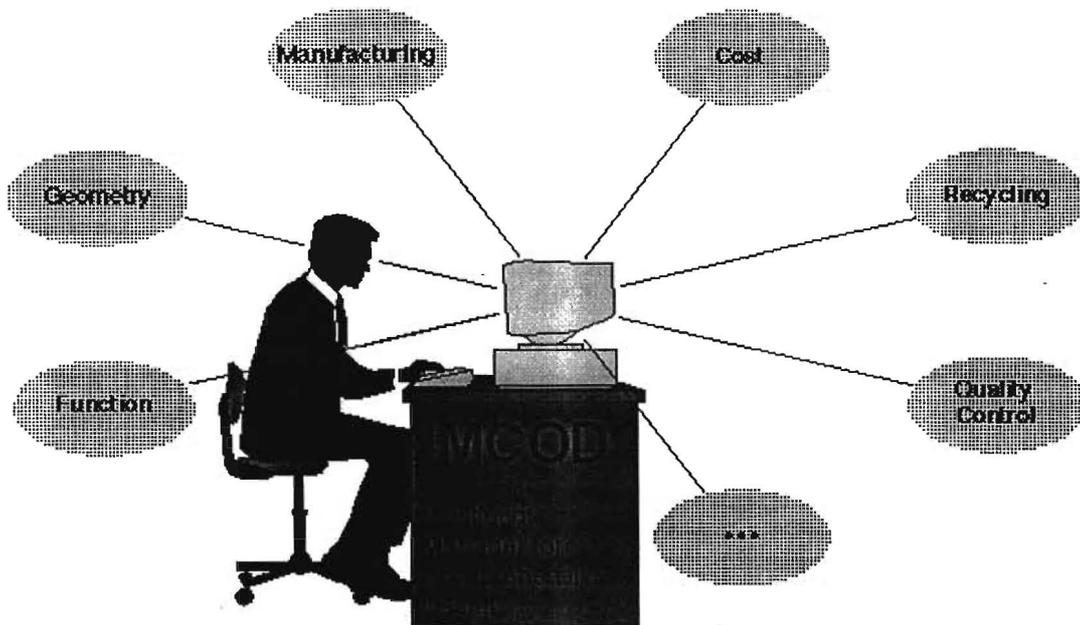


Abbildung 2.1: Die Vermittlerrolle des IMCOD-Managers

phase, Auswahlphase, Umsetzungsphase) im Hinblick auf eine optimale Ressourcenausnutzung der vorhandenen Expertise durch das Konzept des *Zentralen Managements* gesteuert. Das bedeutet: wann welcher Experte eingesetzt wird, entscheidet der Manager. Dazu benötigt er Domänenwissen und Wissen über die verschiedenen Experten, ohne jedoch kompetent in dem betreffenden Problembereich sein zu müssen. Der Experte bleibt lokal, benötigt also kein zusätzliches Domänenwissen. Der zentrale Manager stellt die notwendigen Kommunikationsmechanismen zur Verfügung. Im Gegensatz dazu stehen die Multiagentensystemen, wo die Expertensysteme direkt miteinander kommunizieren können.

2.2 Die Domäne und der Bereich der Entscheidungsunterstützung

Das Institut für Verbundwerkstoffe (IVW GmbH, Kaiserslautern) stellt die notwendigen Informationen über den Designprozeß aus der Sicht des Maschinenbaus zur Verfügung. Im IMCOD-Projekt orientieren sich daher die Konzepte an der Domäne der Druckbehälter aus Verbund-Werkstoffen. Bei der näheren Analyse der Anwendungsdomäne fand eine Sitzung zur Wissenserhebung statt [Ste94], in der Experten exemplarisch den Entwicklungsprozeß eines solchen Druckbehälters durchgespielt haben. Domänenexperten dieses (fiktiven) Szenarios vertraten die Fachsparten Berechnung/Konstruktion, Verarbeitungs-

technik, Wirtschaftlichkeit und Qualitätssicherung. Außerdem war noch ein Vertreter der Kundenfirma anwesend.

Die konventionelle Vorgehensweise bei der Lösung eines konstruktiven Problems beginnt bei der Erfassung der Problemstellung, d.h. die Anforderungen an ein Produkt werden spezifiziert und in einer Anforderungsliste zusammengestellt. Anhand dieses Anforderungskatalogs sollen Designexperten in einer ersten Phase (*First Layout*) alternative Lösungen erstellen. Lösungen können dabei auch Teile von früheren Lösungen enthalten (z.B. aus Konstruktionskatalogen mit Lösungen für Teilfunktionen). Der Ansatz ist so gewählt, daß zunächst eine Sammlung vieler in Betracht kommender Lösungsvorschläge erstellt wird. Wenn das Pflichtenheft von einem oder mehreren Konstruktionsvorschlägen erfüllt werden kann, so kann nach einer geeigneten Auswahl ein Vorschlag in einer zweiten Phase (*Constructional Layout*) weiter optimiert und fertigungsspezifische Fragen (z.B. Werkzeugauswahl) bearbeitet und Verfahrenssimulationen durchgeführt werden. Diese zweite Phase des Designprozesses schließt mit der Berechnung der Herstellungskosten ab (siehe Abbildung 2.2).

Der oder die verantwortlichen Produktplaner müssen als Entscheidungsträger ihre Entscheidungen, welcher Konstruktionsvorschlag aus dem Lösungsraum einer weiteren intensiveren Untersuchung in der zweiten Phase unterzogen wird, auf die Informationen und Aussagen von Experten stützen können. Durch ein entscheidungsunterstützendes System (*Decision Support System - DSS*) in der First-Layout-Phase können notwendige Informationen schnell beschafft und dokumentiert werden. Sie dienen den Experten als Entscheidungshilfe oder können direkt zur Bewertung bestimmter Anforderungen herangezogen werden. So fehlte in der Beispielsitzung zum Beispiel ein Vertreter der Werkstoffkunde, wodurch viele eigentlich einfach zu erhaltende Werkstoffkennzahlen nicht präsent waren. Diese Informationslücke könnte ein einfaches Expertensystem mit einer entsprechenden Datenbank schließen, wenn es in das Decision Support System integriert wäre.

Die Konzeption und Implementation des entscheidungsunterstützenden Werkzeugs ist der Inhalt dieser Arbeit. Das Tool ist als DESIGN-ANALYZER in den IMCOD-MANAGER integriert. Zwei wichtige Ziele standen bei dem Entwurf des DESIGN-ANALYZERS im Vordergrund. Erstens ein geeignetes Konzept zu entwickeln, mit dem menschliche und computerunterstützte Expertise zusammengeführt werden. Zweitens den kreativen Freiraum, den die Aufgabenstellung der Auftragsfirma bietet, durch exakt definierte Einschränkungen ausschöpfen zu können und damit eine Basis für die Diskussion kontroverser Ideen und Meinungen in einer Gruppenentscheidung zu bilden.

Um diese Ziele zu erreichen, mußte der Entscheidungsprozeß systematisiert und strukturiert werden, und anschließend die Ergebnisse übersichtlich dokumentiert werden.

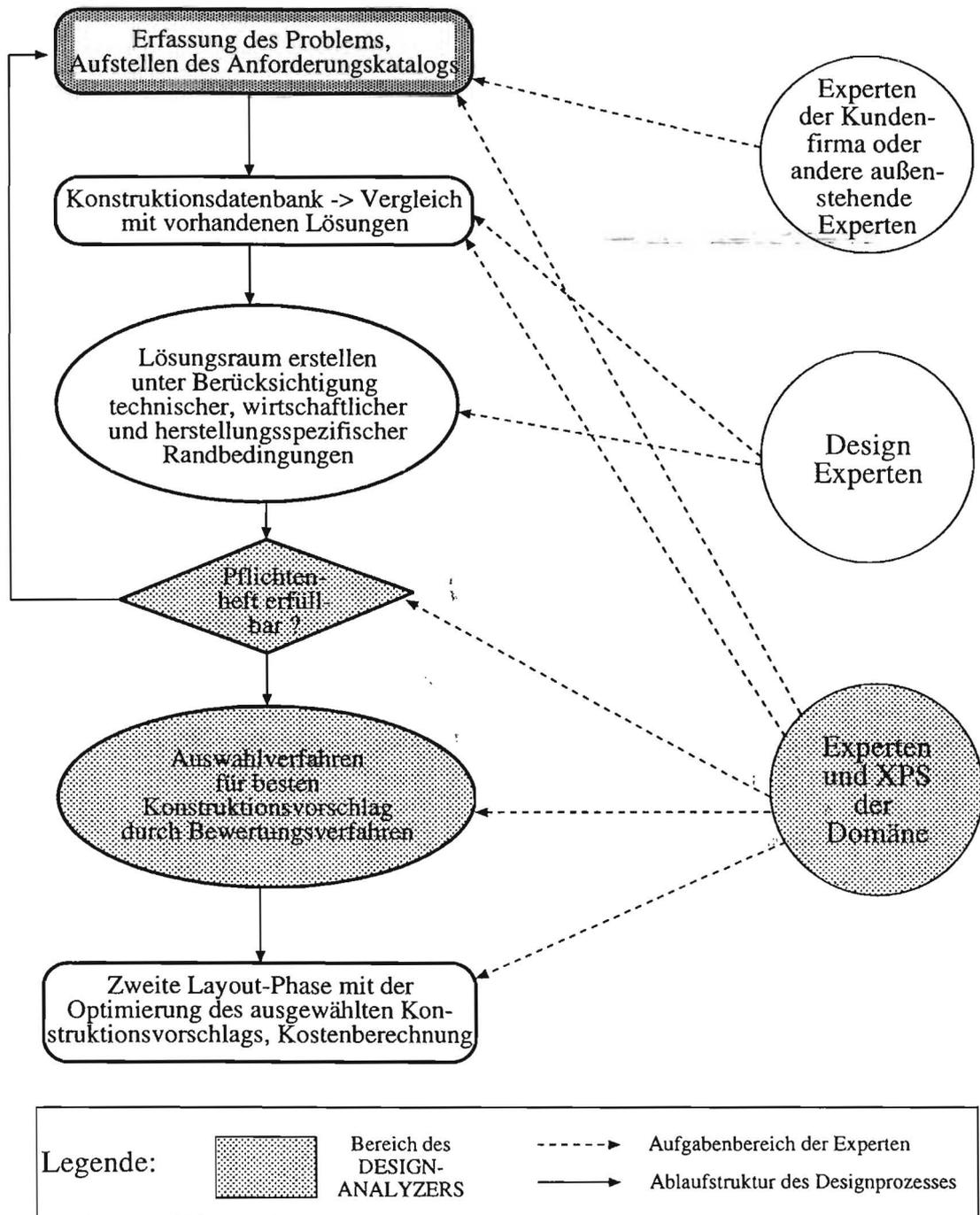


Abbildung 2.2: Struktur des Produktionsprozesses

2.3 Annahmen

Um die praktische Anwendung des Systems zu realisieren, wird von folgenden Annahmen ausgegangen:

- Die Schnittstelle zwischen einem integrierten Expertensystem und dem Manager liefert auf eine Anfrage auf jeden Fall ein Ergebnis an den Manager zurück, das er verwerten kann (siehe dazu auch Abschnitt 5.3).
- Die Konstruktionsvorschläge als Alternativen im Entscheidungsprozeß sind durch das gleiche Produktmodell repräsentiert und lassen sich damit der gleichen Ontologie zuordnen. Dies erleichtert die vergleichende Bewertung der Vorschläge, da sie annähernd nach den gleichen Kriterien bewertet werden können.
- Die Anforderungen an das Produkt lassen sich auf formal strukturierte Kriterien abbilden (siehe Abschnitt 4.2). Diese Annahme prädestiniert den Design-Manager für den Konstruktionsprozeß im Maschinenbau, aber auch für jede andere Domäne, auf die diese Annahme zutrifft.

Kapitel 3

Entscheidungstheoretische und begriffliche Grundlagen

Dieser Exkurs stützt sich hauptsächlich auf eine sehr interessante Arbeit von Johannes Brischar "Berechnung statt Entscheidung" [Bri93], die sich mit den Grenzen und der 'Kritik des Einsatzes von Formalen Systemen zur Entscheidungsunterstützung' beschäftigt. Die Bemerkungen zu Decision Support Systemen sind an [Hä93] angelehnt. Auf das Themengebiet der Entscheidungstheorie wird hier aus zwei Gründen verwiesen, die miteinander verwoben sind:

Bei der Einarbeitung in das Thema *Decision Support* stellt man leicht fest, daß der Begriff Entscheidung vielfältig genutzt wird und dabei selten von den Autoren vorher aus ihrer Sicht definiert wird. Der/die geneigte LeserIn kann die breit gefächerten Ausprägungen selbst eindrucksvoll erleben, wenn er/sie sich an ein Literaturrechercheprogramm setzt und nach Referenzen zum Schlagwort "Entscheidung" suchen läßt. Daher scheint es angebracht, bei den weiteren Überlegungen auf eine einordnende Definition eines Experten zu bauen, in diesem Fall Hermann Lübbe [Lü71]:

"Eine Entscheidung ist fällig, wenn es angesichts alternativer Möglichkeiten zu handeln gilt, ohne daß entscheidende Gründe für die eine Möglichkeit gegen die andere oder umgekehrt vorhanden sind oder zu beschaffen wären."

Der zweite Grund ist, daß sich jeder Entwickler eines entscheidungsunterstützenden Systems der Aussagekraft und der Konsequenzen des Einsatzes von Rechnerleistung bei Entscheidungen, die ja immer auch alternative Möglichkeiten als 'Opfer' im Entscheidungsprozeß mit sich bringen, bewußt sein sollte.

3.1 Entscheidungstheoretische Modelle

Das Grundmodell

Die Entscheidung läßt sich als die Situation einer rationalen Wahl beschreiben:
Eine Person selektiert aus mehreren faktisch möglichen Handlungen diejenige, die gemessen

an gegebenen Ziel- oder Wertvorstellungen den Nutzen oder Wert optimiert und dadurch letztendlich subjektive Befriedigung erlangt.

Die Definition bezieht sich gleichermaßen auf Ziele und Werte, die jedoch nicht gleich zu setzen sind. Die Differenzierung verlangt die Erläuterung der verwendeten Begriffe.

Handlungen und ihre Folgen sind Elemente von möglichen Welten, die jeweils durch einen *Zustand* beschrieben werden können. Ein Ziel ist ein Beschreibungsaspekt eines solchen Zustands. Das Problem der Zustandsbeschreibung liegt darin, daß zur vollständigen Beschreibung kein relevanter (Ziel-)Aspekt fehlen darf. Eine Person verfolgt Ziele und handelt diesbezüglich rational, wenn ihre Handlungen geeignet sind, die Ziele zu erreichen. Falls die Person mehrere mögliche Handlungen zur Auswahl hat - also verschiedene Ziele verfolgen kann - , muß sie sich für eine entscheiden. Durch die Definition einer Ordnung über dem Raum der Handlungen wird versucht, eine einzelne vorzuziehende Handlung auszuzeichnen.

Will man die ausgewählte Handlung in Bezug auf die gegebenen Ziel- oder Wertvorstellungen diskutieren, dann muß man den Anspruch an die Modellierung des Entscheidungsverhaltens bestimmen. Dieser wird in der *normativen* und der *deskriptiven* Theorie des Entscheidungsverhaltens diskutiert (siehe auch [Ric89]). Im normativen Ansatz soll das Modell bei geeigneter Parameterwahl die ergiebigste, beste oder sonst einem beliebigen Ziel dienlichste Alternative auszeichnen, was eine Skala für "besser" verlangt. Ein solches Ziel läßt sich in einen bewertbaren, formalen Aspekt abbilden.

Die deskriptive Theorie analysiert Verhaltensweisen von Personen in Entscheidungssituationen und versucht die Prinzipien in empirisch-sozialwissenschaftlich abgesicherten Modellen zu beschreiben, wodurch das Problem in die Wissensakquisition verlagert wird. Hier werden Wertvorstellungen des Entscheidungsträgers stärker bei der Konstruktion des Handlungsraums berücksichtigt. Wertvorstellungen - unter die zum Beispiel auch Glücksvorstellungen und Emotionen fallen - sind jedoch sehr schwierig in mathematisch berechenbare, naive Modelle abzubilden , teilweise wohl gar nicht. Da nicht alle Wertvorstellungen einer Person oder Personengruppe in einen formalen Zielaspekt abgebildet werden können, sollte in diesem Fall das im Zielsystem durch Optimierung oder Maximierung gewonnene Entscheidungsergebnis anschließend diskutiert werden.

Ausgehend von dem beschriebenen Grundmodell, in dem eine Person versucht durch rationales Handeln von einem Ausgangszustand zu einem Zielzustand zu kommen, wobei über dem Raum der Handlungen eine Ordnung definiert ist, haben sich in der Entscheidungstheorie zwei Modelle manifestiert.

Das deterministisch ökonomische Modell der Wahl

In diesem Modell wird jeder Handlung der möglichen Handlungen $\{H_1, \dots, H_i\}$ ein Handlungsnutzen $g_{handlung}(H_i)$ und den Handlungsfolgen, mitgeteilt durch die Zustände $\{Z_1, \dots, Z_i\}$, eine (Ziel-) Nutzenfunktion $g_{ziel}(Z_i)$ zugeordnet. Der Zielnutzen gibt an, wie sehr ein bestimmter Zustand einem gewünschten Ziel nahekommt. Ein Kriterium definiert auf dem Wertebereich der Nutzenfunktion $g_{handlung}$ eine vollständige Ordnung. Nimmt man als Kriterium Ordnungsrelationen wie zum Beispiel "größer als" oder "nicht kleiner

als", und als Wertebereich der Nutzenfunktion die natürlichen oder rationalen Zahlen, so erlaubt die maximale Nutzenfunktion die Selektion der "besten" Alternative. Im einfachsten Modell existiert nur ein Ziel und der Handlungsnutzen entspricht dem Zielnutzen. Bei mehreren Zielen und mehreren Entscheidungsträgern wird die Bestimmung des Nutzens, also die metrische Abbildung der Handlungen in eine quantitative (mathematische) Nutzenfunktion mit einer eindeutigen Ordnung auf dem Handlungsraum, schwierig. Die einzelnen Nutzenfunktionen müssen zu einer gemeinsamen Nutzenfunktion aggregiert werden.

Das deterministische Präferenzmodell der Wahl

In diesem Modell definiert jedes Kriterium eine *Präferenzrelation* \succeq_d über dem Alternativenraum, so daß die Person anzugeben vermag, welche von je zwei Alternativen sie bevorzugt.¹ An die Präferenzrelation sind die Eigenschaften Reflexivität, Vollständigkeit und Transitivität einer Ordnungsrelation geknüpft.² Während bei dem ökonomischen Modell die Schwierigkeit in der Bestimmung des Nutzens liegt, ist im Präferenzmodell die Schwierigkeit eher in der Erfüllung eben der Präferenzrelationsbedingungen zu sehen, da man bei umgangssprachlich formulierten Kriterien oftmals "Birnen mit Äpfeln" vergleichen muß. Außerdem können Präferenzinterdependenzen der einzelnen Kriterien zu Inkonsistenzen führen. Weiterhin müssen bei mehreren Akteuren die individuellen Präferenzen zu einer kollektiven Präferenzrelation aggregiert werden (*Präferenzaggregation*).

Das Präferenzmodell ist eine abgeschwächte Form des ökonomischen Modells. Es definiert eine direkte Präferenzrelation auf dem Alternativenraum statt wie bei den an Nutzenrechnungen orientierten Modellen über den Umweg der Metrisierung von Kriterien zu gehen. Das bedeutet im Speziellen, daß das Präferenzmodell und das ökonomische Modell isomorph sind (keine Äquivalenz !!), wenn die Präferenzrelationsbedingungen erfüllt sind. Zu einer einzelnen Präferenzrelation \succeq_d kann unter diesen Bedingungen eine Nutzenfunktion konstruiert werden, die allerdings auf den subjektiven Präferenzen einer Person basiert, statt die subjektive Bewertung einer Person wiederzuspiegeln. Beide Modelle erfahren eine Erweiterung in probabilistischen Modellen. Hier werden den Zuständen (Handlungsfolgen) noch Eintrittswahrscheinlichkeiten zugeordnet. Das Verhalten eines Entscheidungsträgers bei einer Entscheidung durch mehrere Personen wird sich zusätzlich verändern, wenn er von den Entscheidungen der anderen Akteure weiß oder Vermutungen darüber anstellt. Die Ansätze, auf die hier nicht näher eingegangen werden soll, beruhen auf Axiomen der Nutzentheorie, deren Grundlagen auf v. Neumann und Morgenstern zurückgehen (siehe auch [Ric89]).

¹Der Index d steht für "deterministisch".

²Zur Erinnerung: *Reflexivität* : $\forall x[x \succeq_d x]$,
Vollständigkeit : $\forall x \forall y[x \succeq_d y \vee y \succeq_d x]$,
Transitivität : $\forall x \forall y \forall z[x \succeq_d y \wedge y \succeq_d z \rightarrow x \succeq_d z]$

3.2 Die Entscheidung als Prozeß

Der Mensch sieht sich einer Entscheidungssituation gegenüber, wenn er bezüglich mehrerer Handlungsalternativen nicht weiß, was zu tun ist: er zweifelt. Der Grund dafür kann sein, daß er mit Problemsituationen dieses Typs keine Erfahrung hat, die Situation ungewohnt ist, oder die Übung fehlt, also kein Experte ist. Die Folge ist **Unentschiedenheit**.

Der Experte denkt nach, wägt ab, begründet und urteilt aufgrund seiner Erfahrung. Stützt sich die Begründung auf anerkannte Gesetzmäßigkeiten des Problembereichs, z. B. in Form von Regeln, so ist die Entscheidung rational begründet. Mangelt es an Rationalität und Überlegung, so kann dies zu Unentschlossenheit aufgrund von Wissensmangel führen. Genau hier ist der Einsatz entscheidungsunterstützender Computer-Software gefragt. Sie kann dem Entscheidungsträger entscheidende Informationen und damit entscheidende Gründe für eine Alternative liefern. Im Spezialfall sind die Gründe Regeln oder Fakten, die gar keine *echte* Entscheidung mehr verlangen, sondern nur noch eine Handlungsalternative als *die beste* und einzig mögliche kennzeichnen. Diese Vorgehensweise, in der zwar festgestellt werden muß, welche von möglichen Alternativen realisiert werden sollte, jedoch für diesen 'unechten' Entscheidungsprozeß klare und eindeutige Regeln bestehen, bezeichnet Brischar als Wahl.

Definition: *Eine Wahl ist die Selektion einer bestimmten Alternative aus mehreren möglichen Alternativen anhand von allgemeinen entscheidenden Gründen.*

Ein *entscheidender Grund* ist alles, was als Berechnungsschritt in einem Verfahren formuliert werden kann. Wenn das Verfahren genau und exakt durchführbar ist, bezeichnet Brischar dieses Verfahren als *effektiv* und setzt es gleich mit dem Begriff des Algorithmus und des Regelsystems. In der Betriebswirtschaft wird diese Art von Entscheidung auch programmierte oder strukturierte Entscheidung genannt (siehe auch [Hä93]). Ihnen gegenüber stehen die nichtprogrammierten oder unstrukturierten Entscheidungen. Diese Entscheidungssituationen entstehen, wenn zum Beispiel die Entscheidungsprobleme zu komplex sind oder bei Innovationen wichtige Daten fehlen.

Da durch den Anspruch der Berechenbarkeit an den Begriff der Wahl eine Realisierung nur in der Welt der Zeichen und formalen Systeme möglich ist (das "Urteil" eines formalen Systems ist bezüglich der beschriebenen "Welt" unzweifelhaft, wenn z.B. eine 'closed world assumption' vorausgesetzt wird), wird noch ein weiterer Begriff eingeführt, der eigentlich nur ein Spezialfall des allgemeinen Begriffs 'Wahl' ist.

Entscheidende Gründe, die wie ein Maßstab wirken, mit dem man Alternativen in 'bessere' und 'schlechtere' aufteilen kann, werden Kriterien genannt. Werden verschiedene Möglichkeiten durch Kriterien verglichen, ohne daß ein eindeutiges Verfahren die Entscheidung durch Regeln herbeiführt, so bezeichnet Brischar diese Vorgehensweise als **Vorzugswahl**. Die Zusammenstellung der Kriterien, die letztendlich die Entscheidung herbeiführen sollen, verlangt auf jeden Fall eine domänenumfassende Urteilsfähigkeit des Experten. Die Kriterien dienen der Verfeinerung der Entscheidungsfindung und sollen die Entscheidungssituation möglichst realitätsnah von einem Realsystem in ein Modell abbilden (siehe Abbildung 3.1 nach Kirchgäßner, 1983, S.14, [Kir83]). In dieser Modellbildung liegt auch die Schwierigkeit des Einsatzes entscheidungsunterstützender Systeme. Das formale System der Zeichen und Regeln zur Manipulation dieser Zeichen - nichts anderes ist ein Computerprogramm

- sollte möglichst genau den beobachteten Sachverhalt modellieren. Werden die Kriterien

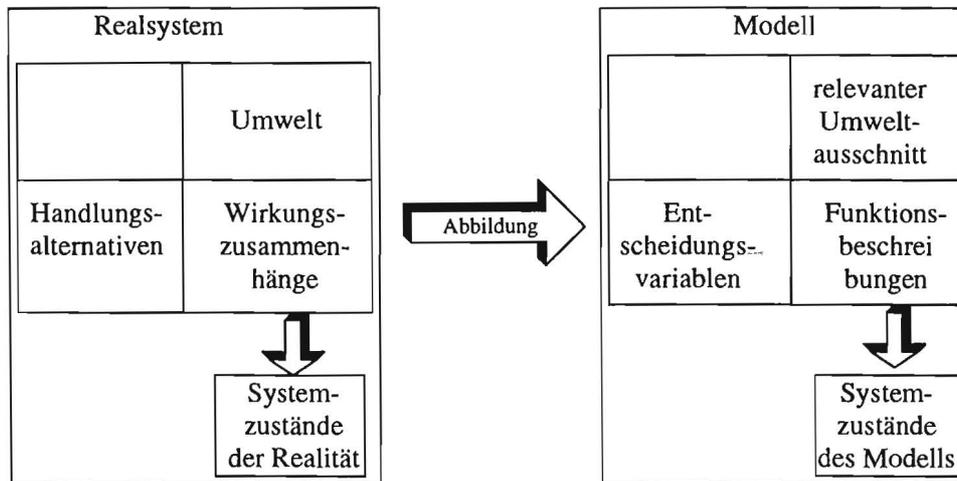


Abbildung 3.1: Die Abbildung eines Realsystems in ein Modell

nur von einem formalen System bewertet und die Alternativen zueinander in Beziehung gesetzt, so liegt für diesen Teilschritt wiederum eine Wahl vor, da die dahinterstehenden Algorithmen kausaldeterminiert agieren. Der urteilende Mensch hingegen handelt nicht nur kausaldeterminiert, sondern urteilt auch intuitiv und trifft eine *echte* Entscheidung oft aus Mangel an entscheidenden Gründen oder aufgrund psychologischer Faktoren, z.B. "Vorweg-Favorisierung" einer Alternative. Stellt er die Kriterienliste für einen komplexen Entscheidungsprozeß auf, so ergeben sich zwangsläufig Situationen echter Entscheidung. Dies wird durch die Schwierigkeit bei der Übersetzung der realen Entscheidungssituation in die Rechnerwelt begünstigt. Die echte Entscheidung beruht nicht wie die Wahl allein auf entscheidenden (regelbasierten) Gründen, sondern auf dem Personsein des Experten, wodurch Elemente des Indeterminismus in den Entscheidungsprozeß einfließen. Nicht alle Wertvorstellungen einer Person lassen sich als Ziele eindeutig formalisieren.

Der Entscheidungsprozeß kann auch als Problemlösungsprozeß bezeichnet werden, wenn man vor die Auswahl der Handlungsalternativen deren Vorbereitung und anschließend die Umsetzung der ausgewählten Alternative stellt. In der ersten Phase – Suchphase oder intelligence activity – werden Veränderungen in der wirtschaftlichen, technischen, politischen und sozialen Umwelt vom Entscheidungsträger wahrgenommen. In der Entwurfsphase (design activity) werden Handlungsalternativen erstellt und das Zielsystem präzisiert. Anschließend wird in der Auswahlphase (choice activity) eine subjektiv beste Handlungsalternative durch ein Bewertungsverfahren ausgewählt. Diese Entscheidung wird dann in der Review-phase überprüft. Abbildung 3.2 zeigt in Anlehnung an Sprague Jr. und Carlson ([SC82], S. 27) die Einbettung und die Verbindung der Teilschritte durch ständiges Feedback. Die Entscheidungsfindung kann also beliebig wiederholt oder erneut gestartet werden.

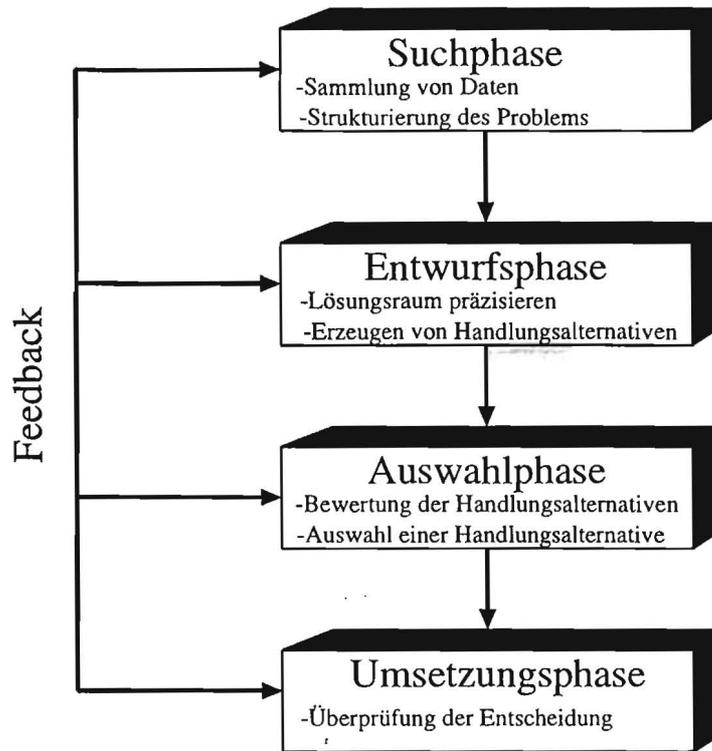


Abbildung 3.2: Der Entscheidungsprozeß als Problemlösungsprozeß

3.3 Decision Support Systeme

Hat ein Entscheidungsproblem keinerlei Struktur, also lassen sich keine Kriterien bilden, dann muß der Entscheidungsträger (in unserer Domäne der Produktmanager) ohne Unterstützung auskommen. Lassen sich jedoch größtenteils Kriterien bilden (*semistrukturierte Entscheidung = Vorzugswahl*), so hat man eine Entscheidungssituation, die einen geeigneten Ansatzpunkt für ein Decision Support System bietet. Zu diesem Zweck vereint es Informationsgeschwindigkeit des Computers und Intuition bzw. Erfahrung des Menschen und integriert dadurch Entscheidungsproblem, Computersystem und Entscheidungsträger. Die folgende Definition von Keen und Scott Morton ([KM78], S. 97) und die Abbildung 3.3 (nach McLeod, 1986 S.431, [McL86]) fassen diesen Sachverhalt zusammen.

Definition: "... *DSS are computerbased support for management decisionmakers who are dealing with semistructured problems.*"

Die Bestandteile eines DSS lassen sich in ein Dialogsubsystem, Datensubsystem und ein Modellsystem gliedern (DDM-Paradigma). Das heißt, es existieren

- Dialoge zwischen Entscheidungsträger und interaktiv genutztem Informationssystem

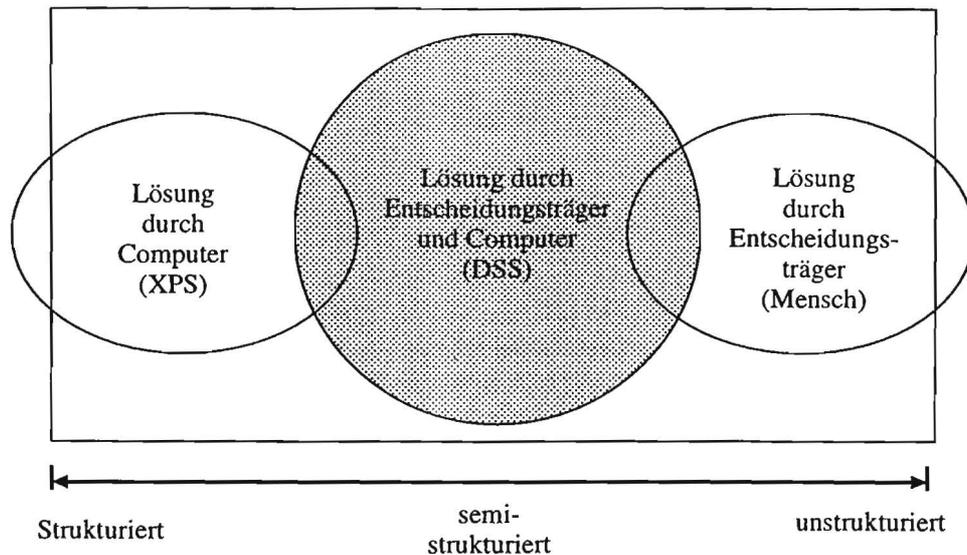


Abbildung 3.3: Die Ausrichtung von DSS auf semistrukturierte Entscheidungsprobleme

- Daten, die das System unterstützen
- Modelle, mit denen Analysen durchgeführt werden.

Abbildung 3.4 zeigt die Bestandteile eines DSS. Wie stark die einzelnen Subsysteme ausgeprägt sind und Abhängigkeiten untereinander bestehen, hängt von der Anwendung in der jeweiligen Implementierung ab. Der Anwender kommuniziert mit dem DSS über das Dialogmanagementsystem und benutzt dabei die Modellbank zur Strukturierung seines Entscheidungsproblems. Mit einer Datenmanagementkomponente greift er auf verschiedene Datenbestände zu.

Eine Erweiterung der DSS stellen die *Group Decision Support Systeme (GDSS)* und die *Distributed Decision Support Systeme (DDSS)* dar. Sie beziehen sich auf die Tatsache, daß in einem Entscheidungsprozeß des industriellen Bereichs meist mehrere Entscheidungsträger zusammen kommen. Mit einem GDSS werden mehrere Entscheidungsträger, die als Gruppe ein Unterstützungssystem (DSS) benutzen, unterstützt. Die Gruppe tritt als single-user des Systems auf. Die Mitglieder der Gruppe können bei ihren unabhängigen Einzelentscheidungen von zusätzlichen Informationssystemen unterstützt werden. Erst bei der Auswahl einer Handlungsalternative werden sie von einem GDSS unterstützt.

In einem DDSS werden mehrere DSS für ein bestimmtes Entscheidungsproblem vernetzt. Die einzelnen Entscheidungen sind voneinander abhängig. Von den Mitgliedern der Gruppe, die als multi-user das System benutzt, kann keiner allein die Gesamtentscheidung treffen, aber alle tragen gemeinsam die Verantwortung für sie. Hier steht der organisatorische Zusammenhang der Gruppe im Mittelpunkt. So können auch zeitliche Abhängigkeiten der Teilentscheidungen bestehen.

Es läßt sich jedoch nicht immer eine exakte Zuordnung eines DSS zu einer der beiden Instanzen treffen, d.h. der Übergang zwischen beiden Systemen ist fließend.

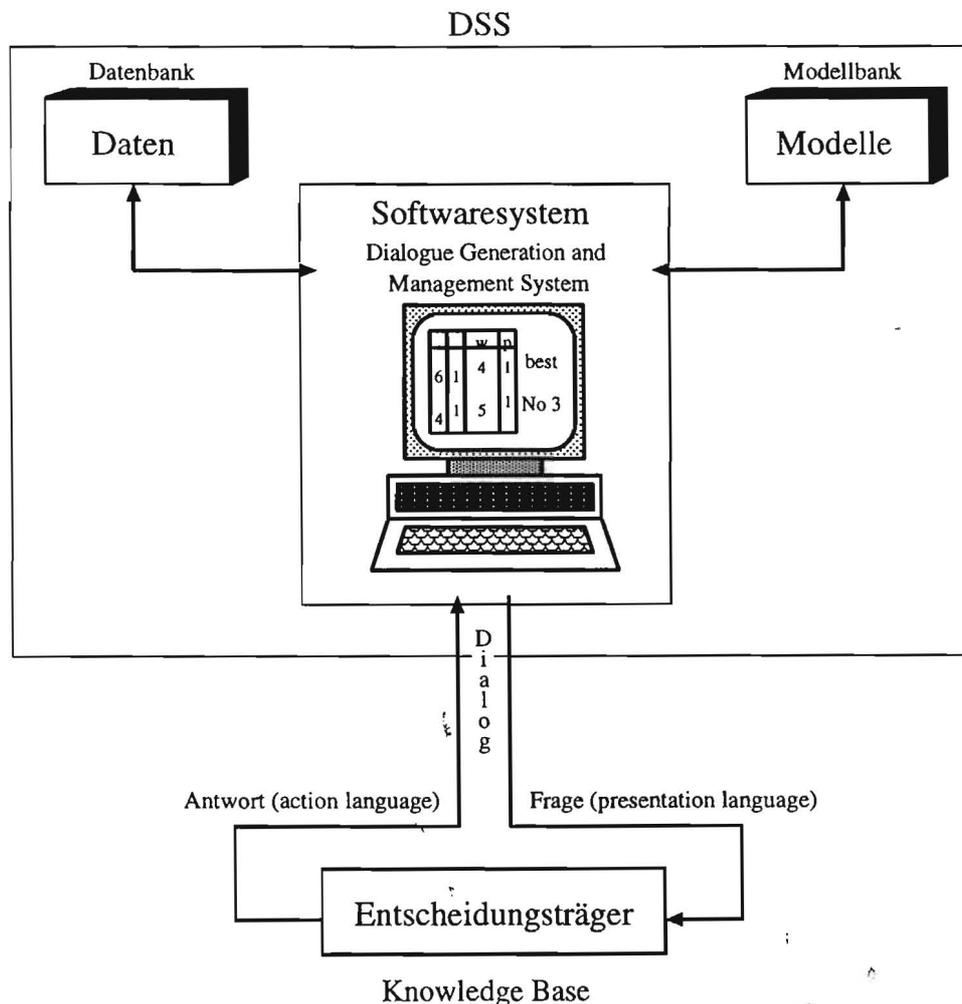


Abbildung 3.4: Das DDM-Modell eines DSS

In diesem Exkurs wurde bewußt vermieden, eine subjektive Wertung der Ansichten von Johannes Brischar zu liefern. In der Zusammenfassung in Kapitel 9 wird darauf noch Bezug genommen. Seine Arbeit, die auf umfangreicher Literaturrecherche fußt, diente in diesem Abriß der begrifflichen Einordnung der Entscheidung.

Der DESIGN-ANALYZER ist im Sinne der in diesem Kapitel erläuterten Begriffe ein Group Decision Support System, das mit Hilfe eines entscheidungstheoretischen Modells (deterministisch ökonomisch) im Sinne einer Vorzugswahl einen semistrukturierten Entscheidungsprozeß unterstützt. Die Berechnung des besten Konstruktionsvorschlags über die Maximierung einer quantitativen Nutzenfunktion ist bei mehreren Einzelzielen und Personen einfacher zu realisieren als die direkte Präferenzrelation zu erstellen, da die Bedingungen einer Ordnung über numerische Werte leichter zu erreichen sind. Die Aggregation der Zielnutzen in eine Gesamtnutzenfunktion wird in Kapitel 6 beschrieben.

Der DESIGN-ANALYZER soll zunächst einmal dem normativen Anspruch gerecht werden und die Generierung eines formalen Zielsystems unterstützen. Inwieweit Wertvorstellung bei der Erstellung des Zielsystems abbildbar sind, ist ein Modellierungsproblem und damit der Gruppe der Entscheidungsträger überlassen.

Die Architektur des DESIGN-ANALYZER ist nach dem DDM-Paradigma erstellt worden. Im zweiten Teil des Dokuments werden nun das Datensubsystem - bestehend aus der Designspezifikation zur Beschreibung der Anforderungen und der Domänenspezifikation zur Beschreibung der beteiligten Experten - , das Modellsystem mit dem eigentlichen Entscheidungsalgorithmus und das Dialogsystem realisiert in einer Benutzeroberfläche vorgestellt.

Kapitel 4

Die DesignSpezifikation

4.1 Das Pflichtenheft des Auftraggebers

Das Pflichtenheft des Auftraggebers ist zunächst einmal eine nähere Spezifikation einer Produkthanfrage an den Produktdesigner. Darin werden Daten über Material, Geometrie, Einsatzbedingungen des Produkts spezifiziert und durch Angaben über technische und wirtschaftliche Randbedingungen ergänzt. Diese Auflistung liegt den Experten der Konstruktionsfirma in textueller Version vor. Anhang A zeigt ein Beispiel einer Anforderungsliste für die Entwicklung eines Preßluftspeichers (LKW) aus Faserverbundwerkstoffen. Das Pflichtenheft enthält jedoch lediglich informelle Anforderungen an das Produkt, die aus der Bearbeitung in der Auftragsfirma und aus markttechnischen Bedingungen (Lagerung, Verkauf, Benutzerwünsche) resultieren. Sie lassen sich als *explizite Anforderungen* klassifizieren. Nicht erwähnt werden hingegen technische Richtlinien und Normen, die stillschweigend vorausgesetzt werden oder sich erst durch die Auswahl einer bestimmten Bauweise ergeben. Diese Anforderungen lassen sich der Klasse der *impliziten Anforderungen* zuordnen. Der *Technische Überwachungsverein (TÜV)* gibt zu diesem Zweck die DIN- (Deutsche Industrienorm) vor und in der betrachteten Domäne der Druckbehälter legt die *Arbeitsgemeinschaft Druckbehälter* in AD-Merkblättern Richtlinien für den Konstruktionsprozeß fest. Diese müssen bei der Bewertung beachtet werden (siehe Abschnitt 4.4).

Wenn ein Vorschlag alle Anforderungen erfüllt, kann die kreative Phase der Lösungsfindung verlassen werden. Ansonsten müssen neue Alternativen entwickelt oder bestehende Alternativen verändert werden. Für den IMCOD-Manager bedeutet das, daß er durch Backtracking die Wiederholung von Iterationsschritten im Designprozeß ermöglichen muß. Besteht der Lösungsraum aus mehreren Vorschlägen, so muß aus diesen der vermeintlich beste anhand der Kriterien selektiert werden.

4.2 Anforderungstypen

Um das größtenteils umgangssprachlich formulierte Pflichtenheft für den DESIGN-ANALYZER zugänglich zu machen, werden die Anforderungen in *Anforderungstypen* unterteilt. Von denen in der Literatur [Rod76], [Rot82] et. al. vorgeschlagenen Typisierungen qualifi-

zierte sich die Einteilung in **Festpunktforderungen**, **Bereichsforderungen**, **Zielforderungen**, da sie bei der Abbildung auf Kriterien den Zugriff auf die gleiche Datenstruktur ermöglichen (siehe 4.3).

1. Festforderungen als Punktforderungen

Bei diesem Anforderungstyp muß ein bestimmter Wert genau eingehalten werden. Dabei muß darauf geachtet werden, daß die Anforderung von einem Konstruktionsvorschlag praktisch erfüllbar ist. Zu restriktive Forderungen müssen neu formuliert werden, wenn der Lösungsraum wegen diesen leer bleibt. (z.B.: *Behältervolumen = 40.5678liter*).

2. Ungezielte Grenz-und Bereichsforderungen

Zur Erfüllung dieser Anforderung dürfen die Grenzen eines Toleranzbereichs nicht überschritten werden. Daher werden sie auch unter dem Oberbegriff Mindestforderungen zusammengefaßt. Wie nah man an die Grenzen herankommt, spielt keine Rolle. Eine Bereichsforderung besitzt eine untere und obere Grenze. Ein Konstruktionsvorschlag muß einen entsprechenden Wert innerhalb des beschriebenen Intervalls vorweisen. Grenzforderungen besitzen zunächst nur eine Grenze. Die zweite Grenze liegt entweder im "Unendlichen" oder wird implizit festgelegt (z.B.: *Volumeninhalt < 80liter*, die untere Grenze ergibt sich implizit mit 0 liter). Demnach lassen sich für die Grenzforderung auch Toleranzbereiche angeben. Auch hier muß die Forderung von einem Konstruktionsvorschlag praktisch erfüllbar sein.

3. Zielforderungen (Bereichsforderungen mit Nutzenfunktion)

Diese Anforderungen können mehr oder weniger gut erfüllt werden. Auch die Zielforderung beschreibt einen Toleranzbereich der jedoch noch um einen Zielwert innerhalb des Intervalls ergänzt wird. Dieser "beste Wert" orientiert sich dabei an einer fiktiven, idealen Konstruktionsvorschlag. Der Erfüllungsgrad eines Konstruktionsvorschlags orientiert sich an diesem Zielpunkt und kann zur Bewertung durch eine Nutzwertanalyse herangezogen werden.

Ein zusätzlicher Unterscheidungsaspekt ergibt sich durch die Einteilung in zwingend zu erfüllende und optional zu erfüllende Anforderungen (Wunschforderungen). Die Wunschforderungen kann man als optimierende Forderungen ansehen, deren Wertebereich einer der drei vorangegangenen Forderungen entspricht (siehe auch Abbildung 4.1). Sie müssen nicht notwendigerweise erfüllt sein und sollten zunächst nicht zur Nutzwertanalyse herangezogen werden (VDI 2222). Man kann sie daher in die Anforderungsliste zunächst aufnehmen oder sofort verwerfen.

Bei der Zusammenstellung der Anforderungen sollte darauf geachtet werden, daß sie nicht voneinander abhängen oder gar widersprüchlich sind. Ausserdem sollten die Toleranzbereichsgrenzen sinnvoll gewählt sein, da sie erheblichen Einfluß auf die spätere Bewertung durch die Experten (Mensch + Maschine) haben. Ein Mensch erkennt zum Beispiel sofort, daß die Forderung "Volumeninhalt zwischen 0 und 80 liter" eine zu schwache Einschränkung darstellt. Toleranzgrenzen haben in der Praxis einen außerordentlich wertbeeinflussenden Charakter. Die Festpunkt- und Bereichsforderungen dienen ausschließlich der

Einschränkung des kreativen Freiraums des innovativen Konstrukteurs. Das Gesamtziel, nämlich die Erreichung eines optimalen Designergebnisses, wird durch die Abbildung der Zielforderungen (und eventuell Wunschforderungen) auf Einzelziele bzw. Kriterien modelliert.

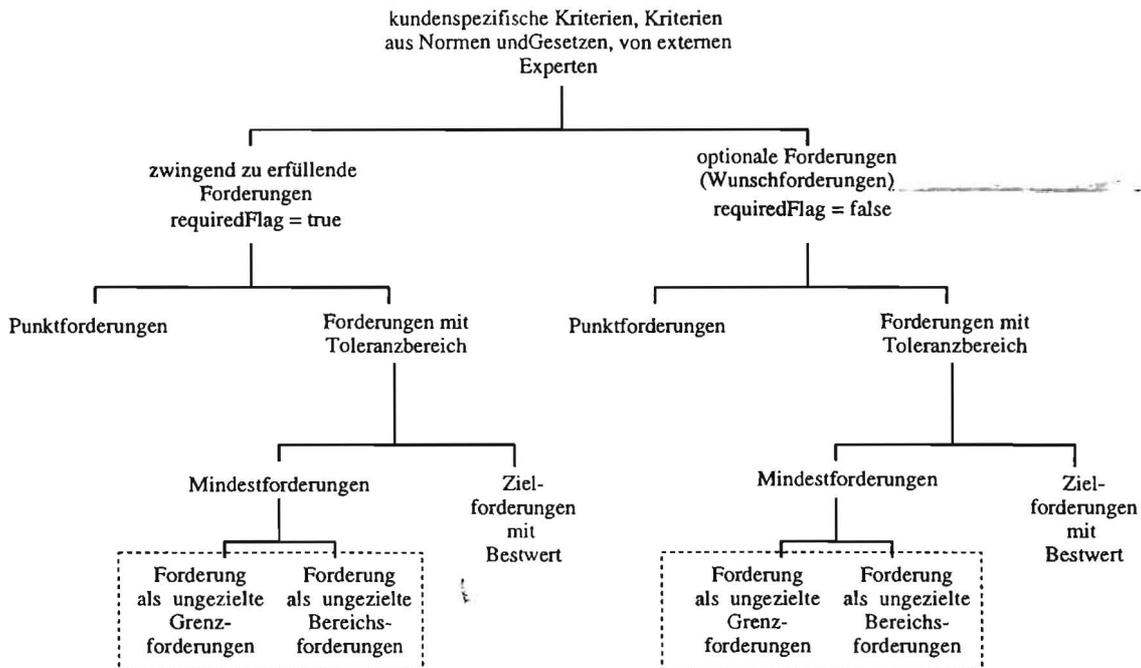


Abbildung 4.1: Anforderungstypen

4.3 Die Abbildung auf Kriterien

Zur Abbildung auf Kriterien werden die Forderungen in einer Datenstruktur formalisiert. Dies ermöglicht eine einheitliche Verwaltung der Kriterien und das Arbeiten mit der Kommunikationsschnittstelle der integrierten Expertensysteme. Die Datenstruktur hat (unabhängig von einer bestimmten Programmiersprache) folgende Slots.

```

<attribute> <best-value> <min-value> <max-value>
            <type>
            <required-flag>
            <weight>
            <category>
  
```

attribute ist ein Wort (STRING), das das Kriterium identifiziert.

best-value gibt den zu erfüllenden Wert für die Festpunktforderungen an. Für die Zielforderungen ist es der optimale Wert. Bei einer Mindestforderung bleibt dieser Slot unbesetzt.

min-value und **max-value** definieren ein Intervall für die Mindestforderungen und die Grenzen für die Zielforderung. Bei der Zielforderung muß der **best-value** innerhalb des definierten Intervalls liegen, kann aber auch mit einem der Grenzwerte identisch sein. Er gibt damit den optimalen Zielwert bzw., wenn optimaler Wert und Grenzwert im Unendlichen liegen, die Tendenz an.

type bestimmt den Typ der Werte des Wertebereichs. Das kann zum Beispiel *STRING* oder *REAL* sein, aber auch speziellere Typen wie zum Beispiel Maßeinheiten (*mm*, *kg*, *N*). Dieser Slot kann auch unbesetzt bleiben. Bei einer Anfrage an ein angeschlossenes Expertensystem muß der Typ jedoch korrekt angegeben sein, damit ein-korrekt arbeiten mit den Begriffsontologien der Schnittstellenbibliothek möglich ist.

required-flag ordnet das Kriterium im Falle *required-flag = true* den zwingend zu erfüllenden Kriterien und im Falle *required-flag = false* den Wunschkriterien zu.

weight wird nur für die Zielkriterien belegt und repräsentiert die Wichtigkeit für das Produktdesign. Mögliche Belegungen sind: *very-low*, *low*, *medium*, *high*, *very-high*.

category ordnet das Kriterium einem bestimmten Expertisenbereich zu. Dieser Slot dient einerseits als Ordnungskriterium aus dokumentationstechnischer Sicht, andererseits als Zuordnungsmöglichkeit zu einem am Entscheidungsprozeß beteiligten Experten.

Die Gewichtung fließt später in die Nutzwertberechnung ein und repräsentiert die Wichtigkeit des Kriteriums für das Produktdesign. Während ein menschlicher Experte dieses Gewicht selbst bestimmt, kann ein XPS diese Zuordnung aufgrund des fehlenden globalen Zusammenhangs (fehlendes Domänenwissen) nicht treffen. Daher muß von der Firmenvertretung (Auftragsfirma) eine Gewichtung vorgenommen werden, die die Präferenz des Kriteriums für die Firma wiedergibt und auf die das XPS zurückgreifen kann. Gleichzeitig kann dieses Gewicht als Orientierungswert für die menschlichen Experten dienen.

Zum *best-value*-Slot muß noch erwähnt werden, daß auch diskrete Mengen bei Festpunktforderungen eingesetzt werden können. Ein Beispiel wäre die Widerstandsfähigkeit des Lackes gegen Mineralöl, Seifenlaugen und UV-Strahlung:

```
Bestaendigkeiten: (Mineraloel, Seifenlaugen, UV-Strahlungen, Lacke)
    <type>
    <min-value>
    <max-value>
    true
    <weight>
    category: Stoffgroessen/chemisch
```

Ein umfassenderes Beispiel findet sich in Anhang B. Die dort abgebildete Designspezifikation ist die formale Abbildung der umgangssprachlich formulierten Anforderungen des Pflichtenhefts aus Anhang A.

4.4 Der domänenspezifische Kriterienkatalog

Wie schon zu Anfang dieses Kapitels erwähnt, ist das Pflichtenheft des Auftraggebers bezüglich der Anforderungen an einen Konstruktionsplan unvollständig. Zusätzliche domänenspezifische Anforderungen können bei einer vergleichenden Bewertung entscheidende Gründe gegen einen Konstruktionsalternative und damit für andere Alternativen liefern, daß heißt einen Vorschlag aus dem Bewertungsprozeß ausschließen. Ein recht einfaches aber verdeutlichendes Beispiel eines solchen Kriteriums ist, daß Druckbehälter undurchlässig gegenüber den Stoffen sein müssen, mit denen sie gefüllt werden. Diese Anforderungen werden als allgemein bekannt angesehen und stehen daher weder im Pflichtenheft noch in den Normensammlungen der Arbeitsgemeinschaft Druckbehälter.

Der DESIGN-ANALYZER bietet während des Formalisierungsvorgangs die Möglichkeit, die das Pflichtenheft repräsentierende Designspezifikation durch domänenspezifische Kriterien zu ergänzen, die Normen und andere fachspezifische Kriterien der Domäne umfassen. Diese Anforderungen sind formalisiert in einem Kriterienkatalog zusammengestellt, aus dem der Benutzer die für die Aufgabe relevanten auswählen kann und das Kriterium sofort editieren kann. Die Slots *< category >*, *< weight >*, *< requiredFlag >* und *< type >* sind schon mit Defaultwerten belegt. So haben zum Beispiel Kriterien, die sich mit Sicherheitsnormen beschäftigen, ein (zwingend) hohes Gewicht.

Beispiel:

```
Temperaturbestaendigkeit: <best-value> <min-value> <max-value>
                          Grad Celsius
                          true
                          high
                          Guete/Sicherheit
```

4.4.1 Mögliche Erweiterung durch MIKADO

Am DFKI wurde im Laufe des ARC-TEC-Projektes die fallorientierte Wissensakquisitionsmethode COKAM+ (Case Oriented Knowledge Acquisition Method from Text) entwickelt, die im MIKADO-Tool-Kit (Modellbasierte, interaktive Wissensakquisition und Dokumentation von Domänenwissen) eine Weiterentwicklung erfährt. Die Erhebung von Expertenwissen wird dabei von der Formalisierungskomponente INFOCOM (Interaktive Formalisation Component) [Pet93] unterstützt, die entsprechend einer vorgegebenen Zielrepräsentationssprache eine natürlichsprachliche Wissensseinheit formalisiert. INFOCOM schlägt zunächst für die Formalisierung eine Menge von Schablonen (abstrakte Templates) der Zielrepräsentationssprache vor. Nach der Auswahl eines Templates wird dieses anhand des Textes teilweise gefüllt und kann dann - falls notwendig - noch vollständig editiert werden. Zum Füllen der Schablonen wird ein Domänenexperte benötigt, der eventuell noch durch einen Knowledge Engineer unterstützt wird. Weiterhin wird das nun formalisierte Textwissen mit der Originaltextstelle verknüpft, sodaß diese stets einsehbar ist, wenn sich Fragen oder Schwierigkeiten ergeben.

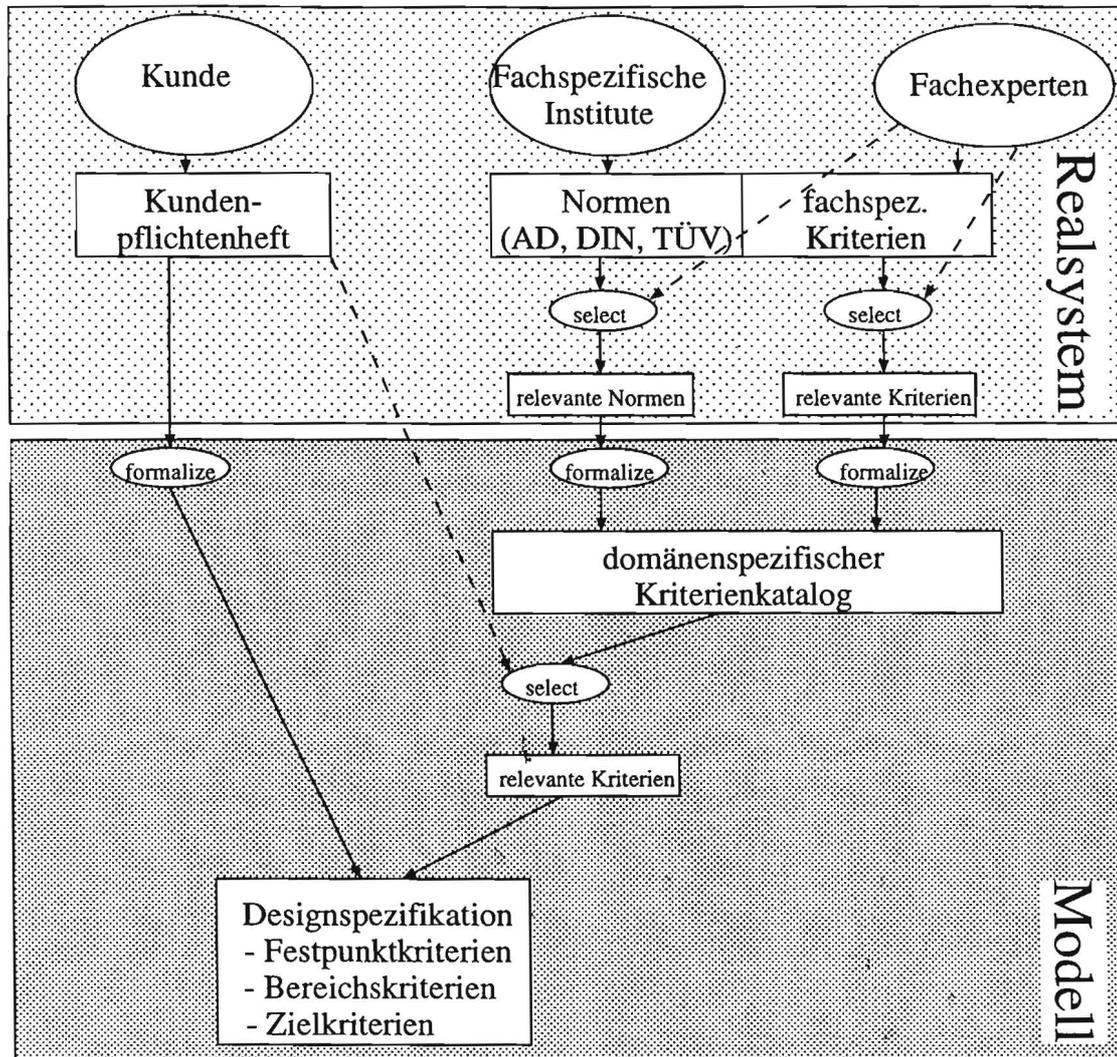


Abbildung 4.2: Die Abbildung relevanter Umweltausschnitte auf Kriterien

Der DESIGN-ANALYZER ist in *Smalltalk/VisualWorks* [Par] implementiert. Da CO-KAM+ in Smalltalk reimplementiert wird (ursprünglich Allegro CommonLisp), bietet sich mit INFOCOM eine sehr gutes und einfach zu integrierendes Tool zur interaktiven Formalisierung des Pflichtenhefts an. Als Ergebnis der möglichen Templates erhält man hier verschiedenen Kriterienarten: Festpunktkriterium, Bereichskriterium und Zielkriterium. Die zweite Einsatzmöglichkeit von COKAM+ findet sich in den zusätzlichen domänenspezifischen Kriterien, die durch verschiedenen Wissensseinheiten wie TÜV-Normen und AD-Merkblätter gebildet werden. Hier können die relevanten Normen für die Designaufgabe selektiert und formalisiert werden. Zusammen mit den übrigen domänenspezifischen Kriterien ergibt sich die *domänenspezifische Designspezifikation*. Abbildung 4.2 zeigt die Abbildung der Anforderungen an das Produkt von dem Realsystem auf die Kriterien des

Modells.

Ein Beispiel:

Knowledge-Unit

”Druckbehälter aus glasfaserverstärktem Kunststoff müssen auf der Innenseite mit einer geeigneten Schutzschicht versehen sein, die das Eindringen der Beschickung in den Verbundwerkstoff verhindert (PVC, Metall, gleiches oder ähnliches Harz). Sie müssen eine Mindestwandstärke von 4 mm aufweisen.”

Filled-Template

```
precondition(Kunststoff-Druckbehaelter),material(glasfaserverstaerkt)
-> Innenseite: (PVC or Metall or Epoxyd-Harz)
    <min-value>
    <max-value>
    plastic
    true
    <weight>
    Guete/Sicherheit
```

und

```
precondition(Kunststoff-Druckbehaelter),material(glasfaserverstaerkt)
-> Wanddicke: 4
    4
    10
    mm
    true
    very-high
    Geometrie
```

Kontextprobleme, die durch die Formalisierung der Kriterien entstehen, könnten durch die Referenz zu den Textstellen in den entsprechenden Wissenseinheiten beseitigt werden.

Kapitel 5

Die Domänenspezifikation

Bei der Erarbeitung eines umfassenden Konzepts für den DESIGN-ANALYZER wurde deutlich, daß die Repräsentation der Domäne mit den angeschlossenen Experten unabhängig von der Domäne der Druckbehälter sein sollte, um die wissenschaftliche Verwertbarkeit der Arbeitsergebnisse zumindest auf den konstruktionstechnischen Bereich des Maschinenbaus auszudehnen. Diese Bedingung führte zwingend zu einer flexibel gestaltbaren Spezifikation des Domänenmodells. Die Tests der entwickelten Software wurden in der Anwendungsdomäne 'Druckbehälter aus Faserverbundwerkstoffen' vollzogen, da diese auch Ausgangspunkt des IMCOD-Projekts durch die Zusammenarbeit mit dem Institut für Verbundwerkstoffe (IVW) war.

5.1 Basisstruktur

Der grundsätzliche Aufbau des Domänenmodells ist durch folgende Datenstruktur der Klasse `ImcodDomain` beschrieben:

```
<domainname> <designspecification>
               <experts>
               <interfacebibliothek>
```

domainname bezeichnet die Domäne.

designspecification ist eine Liste von typischen domänenspezifischen Auswahlkriterien, die sich einem Konstruktionsvorschlag der Domäne zuordnen lassen. Sie wurde bereits in Kapitel 4 beschrieben.

experts ist eine Liste von Experten, die ein gewisses Know-how bezüglich der Domäne vorweisen können und in 5.2 noch näher beschrieben werden.

interfacebibliothek enthält die Computerexperten zusammen mit den Schnittstellenfunktionen, die sie zur Verfügung stellen. Sie können eine Teilmenge der Domänenexperten sein, die ja auch die menschlichen Experten umfassen; sie können aber

auch noch durch zusätzliche Expertensysteme erweitert sein, die als Wissens- und Datenbank während des Entscheidungsprozesses fungieren.

Die Domänenspezifikation ermöglicht erst das Arbeiten mit dem GDSTool und die Kommunikation mit den integrierten XPS.

5.2 Die Experten

Die Liste der Experten vereint menschliche Experten und computergesteuerte Expertise. Im folgenden wird die erste Gattung mit HXP (human expert) und die zweite mit XPS (Expertensystem) abgekürzt, um eine Differenzierung zu erleichtern. Sind beide Gruppen gemeint, werden sie als Experten bezeichnet.

Ein Experte ist ein Objekt der Klasse *ExpertModel* mit den Slots *expertname* und *knowHowList*. Durch die *knowHowList* wird jedem Experten eine Liste von Wissensbereichen zugeordnet, in denen er *kompetent* ist, d.h. Antworten zu Fragen bezüglich dieses Wissensbereichs geben kann. Der einzelne Kompetenzbereich wird durch ein Attribut-Wert-Paar beschrieben: Know-How(*category*, *status*)

Der DESIGN-ANALYZER unterstützt zwei Kompetenzstufen: *status = high* und *status = low*. Experten der zweiten Stufe werden im Entscheidungsprozeß nur dann berücksichtigt, wenn sich zu einem Entscheidungskriterium kein Experte der ersten Stufe findet oder dieser (als einziger der Stufe 'high') keine oder keine sinnvolle Antwort auf eine Anfrage zurückgibt.

Durch diese Kompetenzzuweisung ist es möglich, daß mehrere Experten eine Bewertung zu einem Kriterium der Designspezifikation abgeben können. Diese Fähigkeit charakterisiert ein 'Group Decision Support System'. HXP und XPS agieren auf parallelen Hierarchieebenen und erlauben so die horizontale Integration mehrerer Unterstützungssysteme.

Die Bezeichnung der Kompetenzbereiche findet sich in den Kategorien der Kriterien wieder. Durch die Zuordnung der Kriterien zu einer bestimmten Kategorie, bestimmt der User, welcher Experte zu welchem Kriterium Stellung nehmen darf. Bild 5.1 verdeutlicht diese Zuordnung in Abhängigkeit der Kompetenzbereiche. Um auf sinnvolle Anfragen

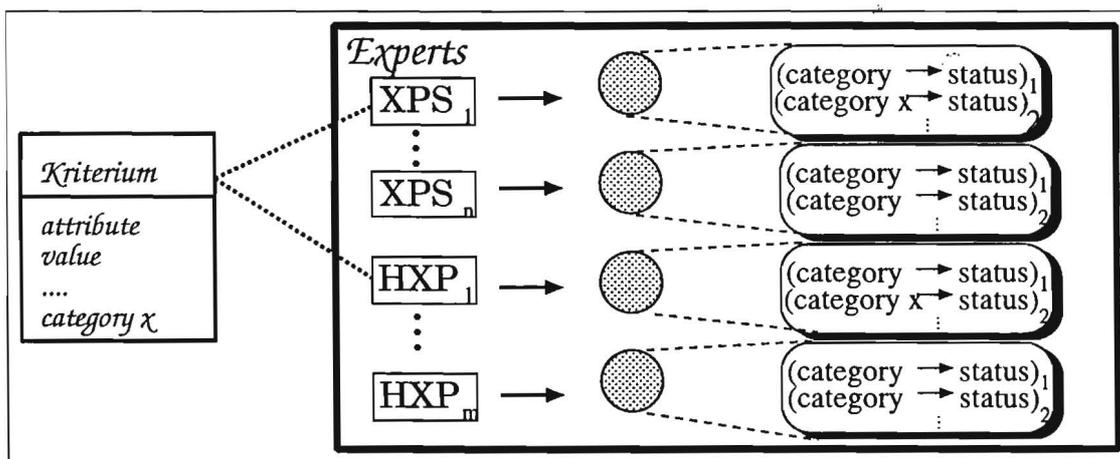


Abbildung 5.1: Kategorienzugehörigkeit der Kriterien und Experten

sinnvolle Antworten liefern zu können, müssen natürlich bei den Vertretern der XPS entsprechende Schnittstellenfunktionen implementiert sein. Sinnvoll heißt in unserem Fall, daß die Antwort vom DESIGN-ANALYZER verwertet werden kann (siehe auch Kapitel 6).

Dazu ein Beispiel:

Bereichskriterium A: (Volumen, Interval [45 , 52],
liter, nil, Geometrie, true)

Zielkriterium B: (Volumen, 50 (best) in [48 , 52],
liter, high, Geometrie, true)

Betrachten wir die zwei Experten CAD-System (XPS) und Forschungsabteilungsleiter für Berechnung und Konstruktion (HXP). Beiden wurde im Bereich Geometrie der Status 'high' zuerkannt. Es könnte nun sein, daß die Implementation der Schnittstelle zwar das tatsächliche Volumen eines KV berechnen kann, aber damit nur über 'erfüllt' oder 'nicht erfüllt' für Bereichskriterium A entscheiden kann. Eine Bewertung des Erfüllungsgrades von Zielkriterium B ist jedoch nicht vorgesehen. In diesem Fall würde der HXP die Bewertung alleine vornehmen.

Die Schnittstellenfunktionen der XPSs sind in der Interfacebibliothek gesammelt.

5.3 Die Interface-Bibliothek

Die Interfacebibliothek ist Teil eines weiteren Tools, das im IMCOD-Projekt implementiert wurde, um die statische Definition der Kommunikationsschicht für ein beliebiges XPS zu ermöglichen [Dri94]. Die Beschreibung der Schnittstellenfunktionen zum Wissensaustausch mehrerer Expertensysteme werden von diesem Tool in einer Bibliothek gesammelt. Diese ist identisch mit der Interfacebibliothek des Domänenmodells und ermöglicht dem IMCOD-Manager durch eine einheitliche Strukturierung der Funktionen einen einheitlichen und geordneten Zugriff auf die XPSs. Die möglichen Nachrichten sind in die Operatoren ASK, TELL, RETRACT, COMMAND, und COMPOSE eingeteilt. Die semantische und syntaktische Beschreibung findet sich in [Dri94]. Hier sei nur der ASK-Operators dargestellt, da er für Anfragen über Kriterien die entsprechende Struktur bestimmt:

Closed-Ask	{	language:	PROLOG
		ontology:	PLASTIC-Ontology
		documentation:	"Asks wether a specific kind of plastic is recycable."
		number-of-answer	1
		world-assumption	CWA
		destructive:	false
		receiver:	Material Expert
		content:	recycable(?x:plastic)

Hinter dieser statischen Definition steckt eine dynamische Definition in einem Modul, das den semantischen und syntaktischen Abgleich zu den dynamischen Schnittstellenfunk-

tionen des XPS durchführt. Der *receiver* kann zu diesem Zweck auch direkt eine TCP/IP-Adresse sein. In der Abbildung 5.2 werden diese Module mit $to - L_i$ bezeichnet.

Der Name des Expertensystems dient in der Interfacebibliothek als Schlüssel, mit dem

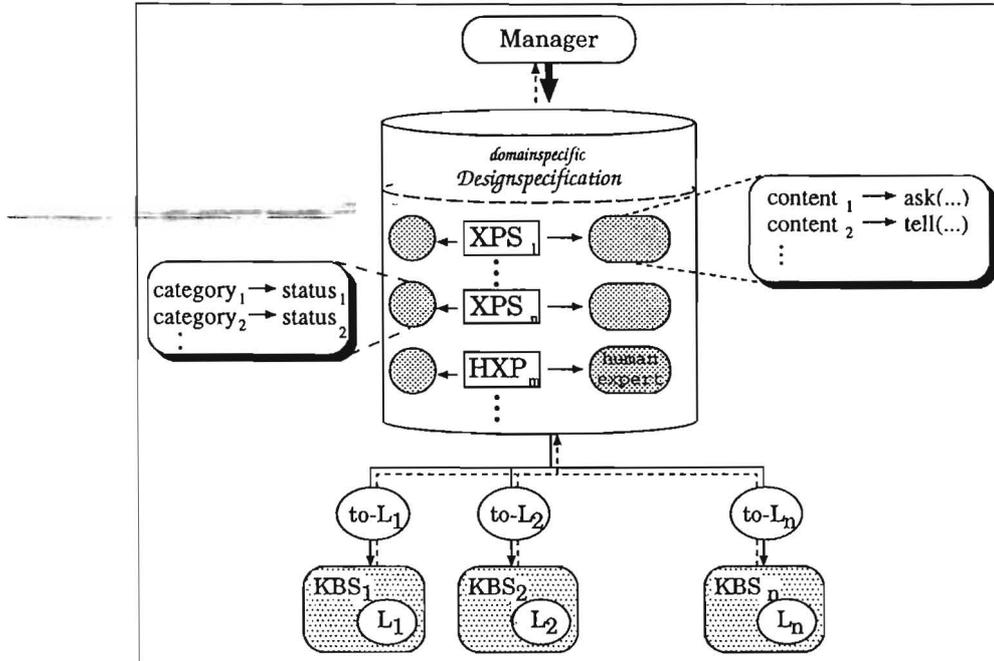


Abbildung 5.2: Die Architektur der Domänenspezifikation

man auf seine Operatoren zugreifen kann. Der Manager kann nun während seiner Arbeitsphase durch eine Anfrage an die Bibliothek herausfinden, welches XPS eine bestimmte Funktionalität leistet, seine Nachricht korrekt aufbauen und die Antwort des XPS korrekt interpretieren. Abbildung 5.2 zeigt die Architektur der Domänenspezifikation mit der Knowhow-Tabelle, den Schnittstellenfunktionen und der Designspezifikation mit dem domänenspezifischen Kriterienkatalog.

Kapitel 6

Der Entscheidungsalgorithmus

In diesem Kapitel soll zunächst detailliert auf den Entscheidungsprozeß und die die Definition der notwendigen Arbeitsumgebung eingegangen werden. Im zweiten Abschnitt wird das Verfahren dann als eingebundener Iterationsschritt in den kompletten Produktentwurfsprozeß betrachtet.

6.1 Isolierte Betrachtung

Der Entscheidungsprozeß des DESIGN-ANALYZERS läßt sich in sechs Einzelschritte einteilen:

1. Modellierung der Domänenspezifikation
2. Abbildung des Pflichtenheftes in die kundenspezifische Designspezifikation
3. Erweiterung der Designspezifikation um domänenspezifische Kriterien
4. Laden der Produktmodelle der zu bewertenden Konstruktionsvorschläge [KV]
5. Multi-criteria-decision
6. Bestimmung des favorisierten Konstruktionsvorschlags

Abbildung 6.1 zeigt die elementaren Schritte des Entscheidungsalgorithmus. Die Domänenspezifikation muß modelliert werden, um mit dem Designmanager arbeiten zu können. Dazu werden den Experten Kompetenzbereiche und speziell den XPSen noch Schnittstellenfunktionen zugeordnet. Da die Kompetenzzuweisung darüber entscheidet, wer zu welchem Kriterium seine Stimme abgeben darf, ist hier Fingerspitzengefühl und Wissen über die Organisation der Entwicklungs- bzw. Konstruktionsabteilung gefragt. An das Expertenmodell wird noch ein domänenspezifischer Kriterienkatalog angehängt, der auch schon im Vorfeld in einer Datenbank abgelegt sein kann.

Mit dem Design-Manager kann dann ein "domänenkundiger" Maschinenbauexperte die

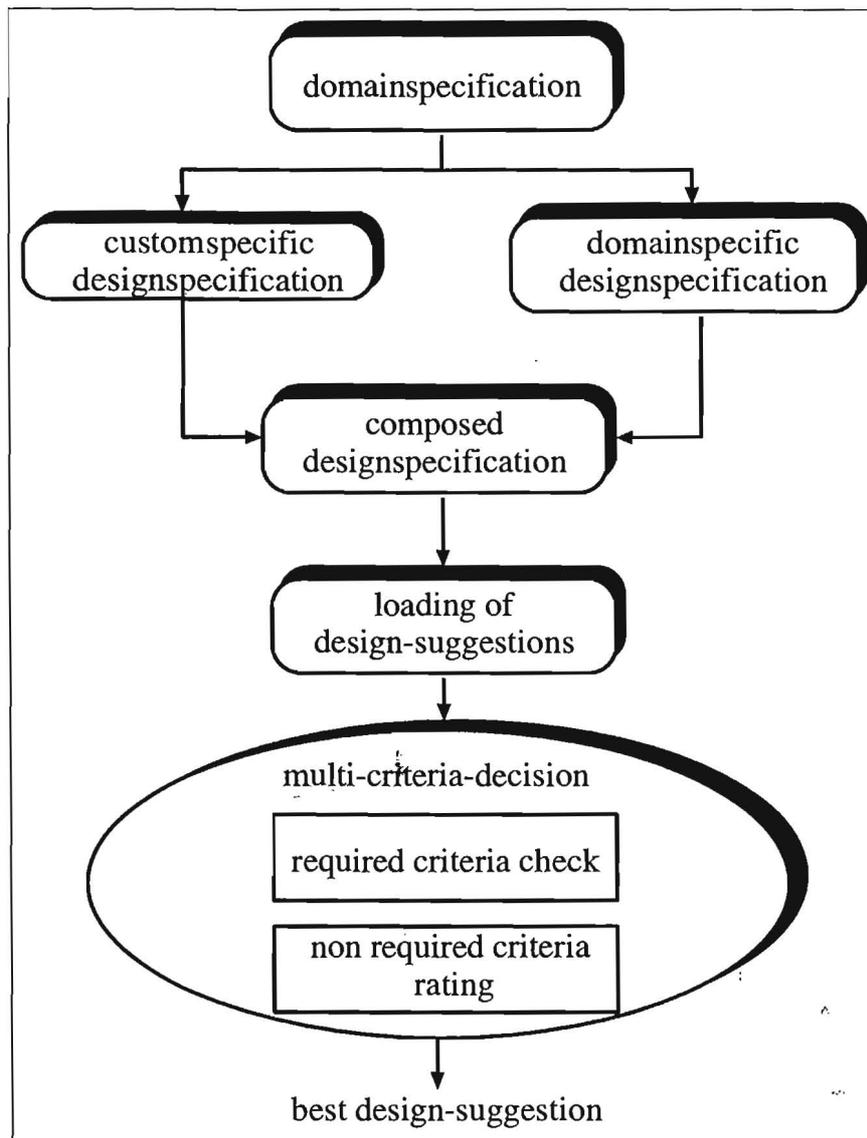


Abbildung 6.1: Der Entscheidungsalgorithmus des Design-Analyzers

Kriterien für die Bewertung zusammenstellen und editieren. Dabei orientiert er sich hauptsächlich an dem Pflichtenheft des Auftraggebers. Hier ist es sinnvoll, daß er von einem Knowledge-Engineer in dieser Aufgabe unterstützt wird, wenn der Maschinenbau-Ingenieur kein Computer-Experte ist. In diesem Schritt muß versucht werden, die Anforderungen an das Produkt möglichst realitätsnah auf die Kriterien abzubilden und dabei gleichzeitig die vorhandene Expertise zu beachten, die in der Domänenspezifikation beschrieben wurde. Es macht keinen Sinn ein Kriterium einer Kategorie zuzuordnen, für die es keinen Experten gibt, der es bezüglich der Konstruktionsvorschläge überprüfen kann.

Zusätzlich ist noch die Mitarbeit eines Vertreters der Kundenfirma zu empfehlen, um die Kriterien bezüglich der Firmenpräferenzen mit Gewichten zu belegen, Kriterien als 'optional' einzustufen und um eventuelle Mißverständnisse der Anforderungsliste zu beseitigen. Den Benutzern stehen für ihre Arbeit zwei Quellen zur Verfügung:

- a) Das Pflichtenheft des Auftraggebers, das von Hand formalisiert werden muß.
- b) Die domänenspezifische Designspezifikation, die schon formalisiert nur noch editiert werden muß.

Die zweite Quelle kann gleich zu Anfang komplett übernommen und dann gekürzt werden oder auch nach dem Formalisieren des Pflichtenhefts mit diesem verglichen werden, um eventuell fehlende Kriterien zu ergänzen. Im nächsten Schritt müssen nun die entscheidungsrelevanten Daten aus dem dreiteiligen Datensubsystem (Domänenspezifikation, Designspezifikation und Konstruktionsvorschläge) gewonnen werden.

6.1.1 Verknüpfung der entscheidungsrelevanten Daten

Die komplette Designspezifikation mit den durch Kategorien geordneten Kriterien bestimmt nun durch den Abgleich mit den Kompetenzbereichen der Experten des Domänenmodells Kriterium-Experte-Paare. Die resultierende Liste teilt sich auf in Paare mit Festpunkt- oder Bereichskriterien (*non rating criteria*) und mit Zielkriterien (*rating criteria*). Zunächst werden bei der Paarbildung jedoch nur Experten mit dem Status *high* der jeweiligen Kategorie berücksichtigt. Nur wenn sich kein Experte dieser Kompetenzstufe finden läßt, werden auch Experten der niedrigeren Kompetenzstufe *low* als Entscheidungsträger eingesetzt.

Wenn der Benutzer die zu bewertenden Konstruktionsvorschläge durch das anschließende Laden der Produktmodell-Dateien aus der Datenbank bestimmt hat, wird jeder KV mit den Experte-Kriterium-Listen für die Zielkriterien als auch für die Festpunkt- und Bereichskriterien zugeordnet. Durch die Verknüpfung entstehen also für jeden KV zwei Datensätze, deren Einträge durch Hinzufügen weiterer Slots folgende Struktur haben:

Non Rating Criteria	Rating Criteria
<designIndex>	<designIndex>
<criterium>	<criterium>
<expert>	<expert>
<success>	<weight>
	<value>

Während bei den Rating Criteria Gewicht und Erfüllungsgrad in den Slots *< weight >* und *< value >* notiert werden, reicht für die Non Rating Criteria ein Slot *< success >* für das 'Erfülltsein' des Kriteriums aus. Die Zuordnung zu den optionalen oder den zwingend zu erfüllenden Kriterien erfolgt durch das **requiredFlag** des Slots *<criterium>*. Die Vorgehensweise der direkten Kriterienzuweisung läßt sich als *statische Kompetenzüberprüfung* bezeichnen, da zu diesem Zeitpunkt noch nicht feststeht, ob ein Experte überhaupt zu jedem der Kriterien eine sinnvolle Antwort geben kann.

Der Design-Manager sieht auch noch eine **dynamische Kompetenzüberprüfung** vor. Sollte nämlich ein Experte zu einem Kriterium und einem KV keine Aussage machen können, da zum Beispiel notwendige Daten im Produktmodell fehlen oder mit freien Variablen belegt sind, und auch kein anderer Experte mehr im Datensatz zu dieser KV-Kriterium-Kombination verbleibt, dann wird sofort nach Experten der niedrigeren Kompetenzstufe gesucht und diese zu dem Datensatz hinzugefügt. Sollte sich auch in dieser Stufe niemand finden, so wird das Kriterium aus der Kriterienliste genommen. Dies wirkt sich bei Zielkriterien nicht nachteilig auf den Bewertungsprozeß aus, führt allerdings zu einer Meldung durch den DESIGN-ANALYZER, da letztendlich die Vergleichbarkeit der KVs nicht mehr gewährleistet ist und eventuelle nicht erfüllte Kriterien im Bewertungsvorgang nicht berücksichtigt werden.

Mögliche Belegungen des Slots *< success >* sind *succeed* für erfülltes und *failed* für nicht erfülltes Festpunkt- oder Bereichskriterium. Wird einem Kriterium eines KVs ein *failed* zugewiesen und ist dieses Kriterium zwingend zu erfüllen (*requiredFlag = true*), so wird zunächst überprüft, ob bezüglich dieser Kriterium-KV-Kombination andere Experten auch dieser Meinung sind. Wenn ja, wird der KV komplett, d.h. mit all seinen Einträgen im Entscheidungsprozeß, herausgenommen. Daher spricht man bei diesen Kriterien auch von Killerkriterien. Stehen jedoch noch Experten-Aussagen aus oder lauten diese anders, wird das KV-Kriterium-Experte Tripel zunächst in eine separate Liste geschrieben. Dadurch werden voreilige Ausschlüsse aus der Lösungsmenge vermieden.

Für die Zielkriterien haben die HXPs die Möglichkeit das Gewicht des Kriteriums zu bestimmen, die XPSs greifen auf die Gewichtung der Kundenfirma zurück. Mögliche Belegungen des Slots *< weight >* sind *very low, low, medium, high, very high*. Die Belegungen spiegeln die Wichtigkeit bzw. die Priorität eines Ziels gegenüber einem anderen Ziel in Bezug auf das Oberziel 'bester Konstruktionsvorschlag' wieder. Die Belegungen des Slots *< value >* umfassen *failed* für Fälle außerhalb des Toleranzbereichs und *worst, poor, fair, good, best* im Falle der Erfüllung (Default-Belegung ist *worst*), Sie besagen, wie nahe ein Konstruktionsvorschlag der optimalen Zielerfüllung ist. Auch die Zielkriterien haben - wenn sie nicht als optional eingestuft sind - den Charakter der Killerkriterien und werden entsprechend den non rating criteria gehandhabt.

Im Gegensatz dazu bedingen die optionalen Kriterien im Falle des Nicht-Erfülltseins nicht den Ausschluß des KVs aus der Lösungsmenge. Sie werden zunächst weder bei der Bestimmung des möglichen Lösungsraums, noch bei der Nutzwertberechnung berücksichtigt, sondern kommen erst dann zum Zuge, wenn die Entscheidung für den weiter zu bearbeitenden KV zweifelhaft erscheint. In diesem Fall kann man zusätzlich die Anzahl der erfüllten optionalen Kriterien zur Rate ziehen. Auf eine weitere Gewichtung der optionalen Kriterien bezüglich des Oberziels wird verzichtet, da die Optionalität der Kriterien diese "überdeckt", das bedeutet eine durch Gewichtung entstehende (zweite) Ordnung wäre nicht aussagekräftig im Vergleich zu den durch die erste Nutzwertkalkulation gewonnenen Ergebnisse. Stattdessen wird für jeden KV die Anzahl der erfüllten, optionalen Kriterien bestimmt.¹ Tabelle 6.2 stellt nochmals die Bedeutung und Bewertungsmöglichkeiten der

¹Die optionalen Zielkriterien werden hier trotzdem zunächst bewertet, da ein "Switchen" des required-Flags zu späterem Zeitpunkt zwecks Simulation veränderter Parameter stets möglich sein soll (siehe auch

verschiedenen Kriterientypen zusammen.

		Bewertungs- möglichkeit	zusätzliche Be- wertungsparameter	requiredFlag	Einfluss auf die Entscheidung
Festpunktkriterium		failed, succeed orientiert an der Punktforderung	keine	true	nicht erfüllt (failed) führt zwingend zum Ausschluss aus der Lösungsmenge
				false	führt nicht zum Ausschluss; kann als zusätzliche Entscheidungs- grundlage in Pattsituationen genutzt werden
Kriterium mit Toleranz- bereich	unge- zieltes Grenz- und Bereichs- kriterium	failed, succeed orientiert am Toleranzbereich	keine	true	nicht erfüllt (failed) führt zwingend zum Ausschluss aus der Lösungsmenge
				false	führt nicht zum Ausschluss; kann als zusätzliche Entscheidungs- grundlage in Pattsituationen genutzt werden
	Ziel- kriterium mit Bestwert	failed + Bewertungs- skala orientiert am Bestwert und Toleranz- bereich	Gewichtung des Kriteriums orientiert am Gesamtziel	true	nicht erfüllt (failed) führt zwingend zum Ausschluss aus der Lösungsmen- ge; erfüllt (succeed) -> Bewertungs- parameter sind Grundlage für Nutz- wertkalkulation
				false	führt nicht zum Ausschluss; kann als zusätzliche Entscheidungs- grundlage in Pattsituationen genutzt werden

Abbildung 6.2: Bedeutung der Kriterientypen für die Entscheidung

6.1.2 Die Nutzwertkalkulation

Mit den beiden Belegungen $\langle weight \rangle$ und $\langle value \rangle$ der nicht-optionalen Zielkriterien läßt sich der Nutzen eines Kriteriums für den Designprozeß modellieren, nachdem der mögliche Lösungsraum durch die Funktion der Killerkriterien eingegrenzt wurde. Da mehr als ein Zielkriterium ausschlaggebend für das übergeordnete Ziel eines optimalen Designergebnisses ist, müssen die einzelnen Nutzenfunktionen aggregiert werden. Dazu ist es sinnvoll den umgangssprachlichen Bewertungen numerische Werte zuzuordnen. Die Bedingungen einer Ordnungsrelation können so weiterhin gewährleistet werden.

Die numerischen Werte für Gewicht und Wert können die Integerwerte von 0 bis 4 – also 4 für *very high* bzw. *best* und 0 für *very low* bzw. *worst* – annehmen. Dadurch läßt sich eine numerische Berechnungsvorschrift für den Gesamtnutzen eines KVs aufstellen, die sogenannte Nutzwertanalyse. Das Besondere dieser Nutzwertkalkulation ist – im Vergleich zu den in der Maschinenbau-Fachliteratur beschriebenen Vorgehensweisen –, daß die Berechnungsparameter von mehreren Experten bestimmt werden. Diesem Sachverhalt wird

in der Nutzwertanalyse des DESIGN-ANALYZERS Rechnung getragen.

Gewichtung der Kriterien

Wie schon mehrfach angesprochen bestimmen die menschlichen Experten das Gewicht jedes Kriteriums, das ihnen zugeordnet wurde, selbst. Allerdings sollten sie dabei die Gewichtung des Kriteriums durch die Kundenfirma berücksichtigen, da sie als Konstruktionsexperten der Herstellerfirma ihren ausgewählten Designvorschlag der Kundenfirma präsentieren müssen. Da die Expertensysteme nicht die Möglichkeit der subjektiven Einstufung des Kriteriums in das Gesamtziel des optimalen Designergebnisses besitzen, übernehmen sie die Gewichte der Kundenfirma. Bildet man den Mittelwert der Gewichte aller n stimmberechtigten Experten für ein Kriterium j , so entspricht der gewonnene Wert dem Gewicht eines hypothetischen Experten für das betrachtete Kriterium.²

$$\text{WeightHypo}_j = \underbrace{\left(\frac{1}{n} \sum_{i=1}^n \text{weight}_{j,i} \right)}_{\text{Durchschnittsgewicht}}$$

Einzelbewertung der Kriterien

Ebenso wie bei der Gewichtung wird auch hier der Mittelwert aus allen Einzelbewertungen der n Experten gebildet, so daß sich eine Bewertung des KVs bezüglich des betrachteten Kriteriums j durch einen hypothetischen Experten ergibt.

$$\text{ValueHypo}_j = \underbrace{\left(\frac{1}{n} \sum_{i=1}^n \text{value}_{j,i} \right)}_{\text{Durchschnittsbewertung}}$$

In diesem Fall sollten die menschlichen Experten ihre Bewertung unabhängig von der Gewichtung der Kriterien vornehmen, da der Erfüllungsgrad sich nur am tatsächlichen Wert des einzelnen Konstruktionsvorschlags in Beziehung zu einem (fiktiven) Idealwert dieses speziellen Kriteriums orientiert. Die XPS werden diese semantische Trennung von Gewicht und Bewertung ohnehin wahren.

Diese Vorgehensweise entspricht den Überlegungen eines Ingenieurs, der einerseits beurteilt, wie gut eine Teilfunktion oder eine Forderung des Pflichtenhefts erfüllt ist, andererseits gewichtet er den daraus entstandenen Vor- oder Nachteil für den Konstruktionsvorschlag in Bezug zu den übrigen Forderungen.

Die eigentliche Bewertung des Kriteriums j für den Konstruktionsvorschlag d ergibt sich

²Die in diesem Abschnitt verwendeten Indizes sind wie folgt zuzuordnen:

- Experten i mit $i=1, \dots, n$
- Konstruktionsvorschlag d
- Kriterium j mit $j=1, \dots, m$

nun aus dem Produkt des hypothetischen Gewichts und der hypothetischen Einzelbewertung. Würde das arithmetische Mittel aus den Einzelprodukten statt dem Produkt der arithmetischen Mittel gebildet, wäre die Bewertung des Kriteriums für den Experten auch durch einen Slot mit der Skala 0 bis 16 ausreichend und die semantische Trennung der beiden Parameter nicht gewährleistet.

$$\text{Rate}_{d,j} = \text{ValueHypo}_{d,j} * \text{WeightHypo}_{d,j}$$

Gesamtbewertung des Konstruktionsvorschlags

Zur Berechnung des Gesamtergebnisses müssen jetzt noch alle Kriterienbewertungen eines KV's addiert werden. Das Ergebnis wird noch durch die Summe der hypothetischen Gewichte aller Zielkriterien des betrachteten KV's dividiert, um es auf die Skala von 0 - 4 zu normieren:

$$W_d = \frac{\sum_{j=1}^m \text{Rate}_{d,j}}{\sum_{j=1}^m \text{WeightHypo}_{d,j}}$$

oder die komplette Gleichung, d.h. ohne Einsetzen der Zwischenergebnisse:

$$W_d = \frac{\sum_{j=1}^m \left[\left(\frac{1}{n} \sum_{i=1}^n \text{weight}_{j,i,d} \right) * \left(\frac{1}{n} \sum_{i=1}^n \text{value}_{j,i,d} \right) \right]}{\sum_{j=1}^m \frac{1}{n} \sum_{i=1}^n \text{weight}_{j,i,d}}$$

Der KV mit dem höchsten Wert W_d wird als KV vom DESIGN-ANALYZER vorgeschlagen. Die maximal zu erreichende Punktzahl ist 4. Dieses Maximum wird dann erreicht, wenn allen zu bewertenden Kriterien eines KV's der Erfüllungsgrad *best* von allen stimmberechtigten Experten zugeordnet wird.

Ein mit der minimalen Punktzahl 0 bewerteter KV würde voraussetzen, daß alle Kriterien mit *worst* bewertet werden. Das quantitative Endergebnis der Nutzwertkalkulation könnte also wieder in eine qualitative Bewertung umgerechnet werden (z.B. very-low bis very-high), allerdings erkennt man an diesen Ergebnissen nicht mehr die tatsächlichen Abstände zwischen den konkurrierenden Konstruktionsalternativen. Sie sind die Diskussionsgrundlage für die abschließende Designentscheidung. Die schwache Relation ' \leq ' definiert aufgrund der quantitativen Bewertung eine Ordnung auf den Einzelergebnissen der KV's. Das bedeutet, daß die Interpretation der Ergebnisse erst durch einen Vergleich der Einzelergebnisse sinnvoll wird. Extremwerte im unteren und oberen Bereich sollten als Warnung verstanden werden. Die Werte können auf schlecht differenzierte bzw. unvollständige Bewertungen oder fehlende Experten für angesprochene Kategorien zurückgehen. Aussagen über Ergebnisintervalle kann man hier jedoch nicht tätigen, da die konkreten Werte stets vom Pflichtenheft bzw. der Designspezifikation abhängen.

Zusammenfassend wird in Abbildung 6.3 dargestellt, wie die am Entscheidungsprozeß beteiligten Experten die Einflußparameter der Nutzwertkalkulation bestimmen können und in welcher Phase das geschieht. Die Nutzwertberechnung wird natürlich auch manipuliert, wenn ein ehemals als optional eingestuftes Zielkriterium als nicht optional eingestuft wird oder umgekehrt. Dies zeigt wiederum, daß die Aufgabenformulierungsphase als Ganzes

entscheidend für die Aussagekraft der Nutzwertberechnung ist. Anschließend folgt ein Rechenbeispiel für die Bewertung eines Zielkriteriums durch die Nutzwertkalkulation.

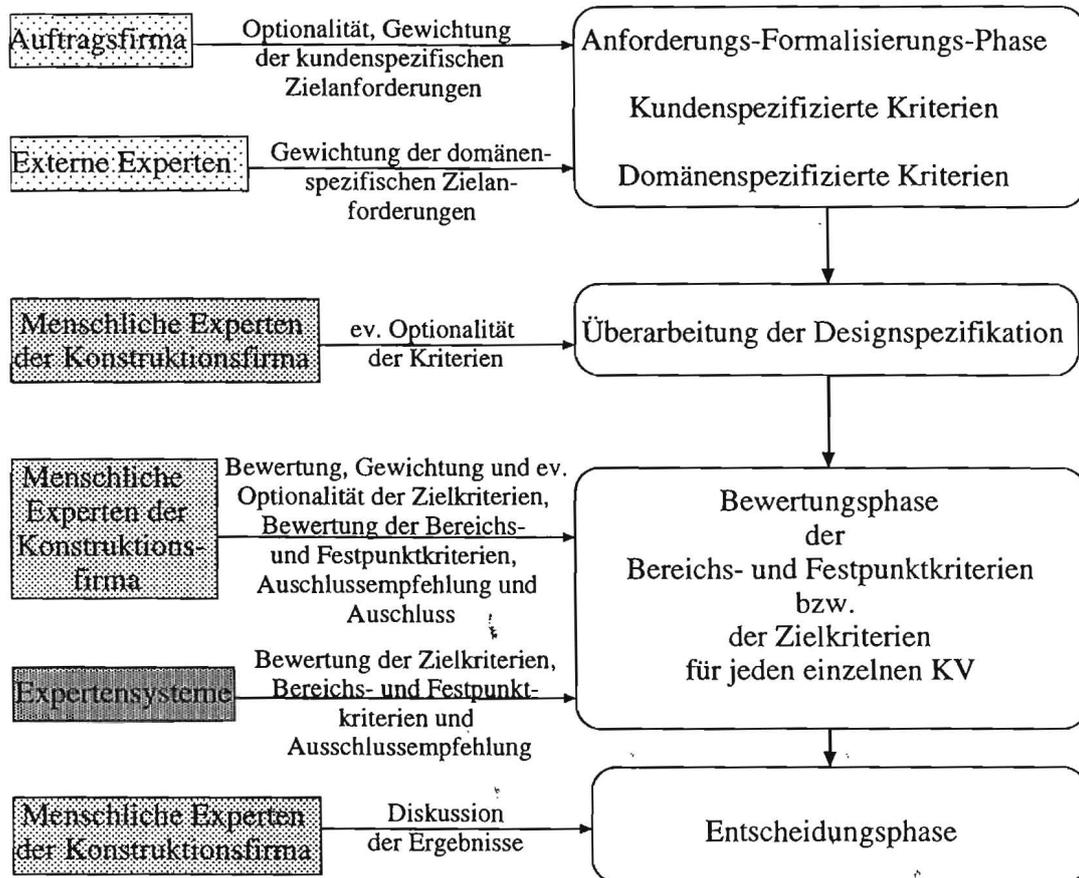


Abbildung 6.3: Einflußparameter des Entscheidungsprozesses

Ein Beispiel:

Betrachtet wird das Zielkriterium:

```

'Recyclierfaehigkeit': 100 (best) in: [80 , 100 ] %
  category: 'Stoffgroessen/chemisch'
  defaultweight: 'high'
  required: true
  
```

Zwei Experten, ein Verarbeitungstechniker und ein Expertensystem für Materialien, besitzen den Status *high* und sind somit stimmberechtigt. Der menschliche Experte verändert die Gewichtsgröße auf *medium* und bewertet die Recyclierfähigkeit von 89 Prozent mit Blick auf die Werbewirksamkeit mit *best*, da dieser Anteil für Werbezwecke vollkommen

ausreicht und totales Recycling sowieso nur theoretisch möglich ist. Die Entscheidung des Expertensystems basiert auf einer Materialontologie. Es berechnet 84 Prozent Recyclerfähigkeit und bewertet das Kriterium mit *poor*; die Gewichtung übernimmt es vom Auftraggeber.

Die Berechnungsvorschrift ergibt demnach für dieses Kriterium:

$$Rate_{d,j} = \underbrace{\left[\frac{1}{2} * (4 + 0) \right]}_{\text{Durchschnittswert}} * \underbrace{\left[\frac{1}{2} * (2 + 3) \right]}_{\text{Durchschnittsgewicht}} = 2 * 2.5 = 5$$

Hätte man die Bewertung für jeden Experten einzeln berechnet, so hätte das Expertensystem dem Kriterium $0 * 4 = 0$ Punkte zugeordnet, der Verarbeitungstechniker $4 * 2 = 8$ Punkte, also im Mittel 4 Punkte. Auch wenn der Unterschied nicht sehr groß ist, geschieht die Durchschnittswertberechnung des DESIGN-ANALYZERS an bezüglich der Semantik geeigneterer Stelle. Eventuell nachteilig wirkt sich allerdings in beiden Varianten aus, daß durch die Mittelwertbildung stark gegensätzliche Bewertungen in einer Art Kompromissbewertung 'verwischt' werden. Allerdings sind die individuellen Bewertungen jederzeit durch den DESIGN-ANALYZER einsehbar und können daher anschließend noch diskutiert und eventuell korrigiert werden.

Die lineare Skalierung von Gewichten und Werten muß nicht beibehalten werden. Sie kann wie auch der Typ (hier Cardinal) geändert werden und zu einer neuen Nutzwertkalkulation führen. Weiteres dazu in Kapitel 7.2 und in der Zusammenfassung Kapitel 9.2.

Die Entscheidung über den endgültigen Ausschluß eines KVs wird in der Runde der menschlichen Experten diskutiert und beschlossen. Diese letzte Sicherungseinrichtung begründet sich daraus, daß oftmals gute Ideen durch einen kleinen, leicht zu korrigierenden Fehler aus dem Lösungsraum fallen, was wiederum nicht Sinn und Zweck der Nutzwertanalyse ist.

Da die Nutzwertanalyse des DESIGN-ANALYZER auch als entscheidungsunterstützend anzusehen ist, können die HXPs zum Beispiel bei geringen Unterschieden im Punktestand auch einen anderen als den vorgeschlagenen KV favorisieren und für den zweiten Schritt im Konstruktionsprozeß auswählen. Sie werden in dieser Situation des Zweifelns durch die Informationen über die optionalen Kriterien unterstützt. Hier kommen Aspekte einer freien Entscheidung der menschlichen Experten zum Tragen, die so auch die nicht formalisierten Wertvorstellungen im Entscheidungsprozeß geltend machen können.

6.2 Integration in den Produktentwurfsprozeß

Der DESIGN-ANALYZER ist als entscheidungsunterstützendes System in den IMCOD-Design-Manager integriert. In dem umfassenden Produktionsprozeß von der Idee zur Fertigung hat er im wesentlichen vier verschiedene Aufgaben zu erfüllen.

Aufgabenstrukturierung und -formulierung

Durch die Formalisierung und Gewichtung einzelner Kriterien in der Designspezifikation

können die Anforderungen an das Produkt genauer spezifiziert werden. Das oft düftig strukturierte Pflichtenheft alleine bietet in diesem Sinne keine Basis für Entscheidungssituationen im Konstruktionsprozeß. Gleichzeitig kann bei frühzeitiger Erstellung der Designspezifikation diese genutzt werden, um den kreativen Freiraum für die Konstrukteure exakt zu bestimmen.

Überprüfung der Erfüllbarkeit des Pflichtenhefts

Es besteht die Möglichkeit, daß die Konstruktionsvorschläge den Anforderungen eines Pflichtenhefts zusammen mit den domänenspezifischen Normen nicht genügen. Die ersten Entwürfe scheitern also an den Killerkriterien des DESIGN-ANALYZERS. Das Expertenteam hat dann zwei Möglichkeiten zur Auswahl. Es kann die Konstruktionsentwürfe noch einmal an die Design-Experten verteilen und ihnen die entsprechenden Gründe für die Ablehnung mitliefern.

Die andere Möglichkeit ist die Kontaktierung der Auftragsfirma und die Umformulierung der harten Constraints, die als Killerkriterien wirkten.

Verwendung der Bewertungsergebnisse in der zweiten Konstruktionsphase

Wird letztlich ein Designvorschlag für die Constructional-Layout-Phase zur weiteren Optimierung ausgewählt, so können die Ergebnisse des Entscheidungsalgorithmus herangezogen werden. Insbesondere bieten sich die Zielkriterien an, die mögliche Schwachstellen des Vorschlags aufzeigen.

Wiederverwendung der Daten

Während des Entscheidungsprozesses für die konkrete Designaufgabe fallen eine große Menge Daten an, hinter denen sich Expertenwissen der Unternehmung verbirgt. Diese Informationen könnten jedoch auch für folgende Aufträge von Bedeutung sein. Daher werden die anfallenden Daten gesichert. Dies sind im einzelnen

- Die Produktmodellaten der Konstruktionsvorschläge zeigen oft intelligente Lösungen für Teilfunktionen einer Konstruktion, auch wenn der Vorschlag nicht weiter verfolgt wird.
- Die Domänenspezifikation kann als Modell des DSS zusammen mit der schon definierten XPS-Interface-Bibliothek für weitere Sitzungen der F&E-Abteilung verwendet werden.
- Zu dem gerade erwähnten Punkt gehört sicherlich auch der domänenspezifische Kriterienkatalog, der sich mehr und mehr vervollständigen läßt.
- Die entscheidenden Gründe für die Ablehnung eines Vorschlags könnten als Falldatenbasis eines Cased-Based-Reasoning-Tools fungieren.

Die Wiederverwendung der Daten von Expertenentscheidungen ist Teil der Kernidee des IMCOD-Projekts. Die gesammelte Expertise wird nicht nur für konkrete Fälle genutzt, sondern auch bei nachfolgenden Problemen referenziert.

Die verschiedenen Aufgabenbereiche des DESIGN-ANALYZERS in der ersten Layout-Phase des Produktionsprozesses sind noch einmal graphisch in Abbildung 6.4 dargestellt.

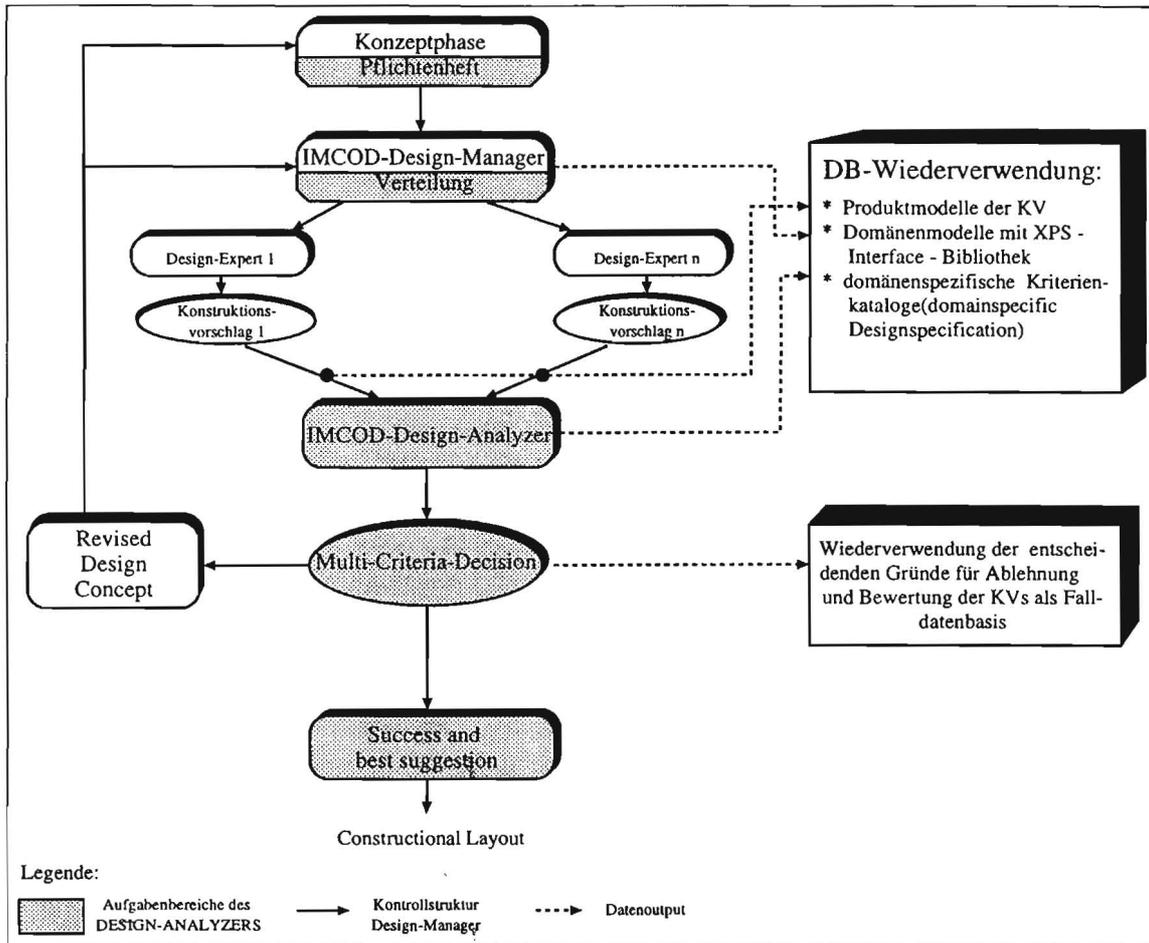


Abbildung 6.4: Der Design-Analyser integriert in den Produktionsprozeß

6.2.1 Darstellung der Konstruktionsvorschläge

Die Konstruktionsvorschläge werden laut Annahme (siehe Kapitel 1) mit Hilfe derselben Ontologie bzw. desselben Produktmodells repräsentiert. Das dient nicht nur der besseren Bewertung der KVs, sondern auch der besseren Darstellung. Durch das Laden der Domäne kann gleichzeitig ein verwendetes CAD-System als Experte festgelegt werden, mit dem dann auch die jeweiligen Vorschläge dargestellt werden (in Wort und Bild) oder sogar direkt Veränderungen an KVs durchgeführt werden können.

Der DESIGN-ANALYZER hat über den IMCOD-MANAGER Zugriff auf ein System zur Editierung von Konstruktionsvorschlägen für die Druckbehälterdomäne mit Hilfe eines Produktmodells, das von Dipl.-Inform. Ansgar Bernardi und Dipl.-Ing. Markus Steffens in Anlehnung an das STEP-Format am DFKI entwickelt wurde ([SB95]). Das Datenmodell umfaßt Geometriedaten, die sich auf ein definiertes Koordinatensystem beziehen, ein Topologiemodell, um Nachbarschaftsbeziehungen zwischen Elementen des Geometriemo-

dells zu beschreiben, ein Produktstruktur- und Materialmodell, ein Oberflächenmodell und schließlich ein Toleranzmodell.

Kapitel 7

Implementierung

Der DESIGN-ANALYZER wurde in der objekt-orientierten Sprache Smalltalk mit Visual-Works2.0 von ParcPlace Systems implementiert ([Par]). Im folgenden soll kurz auf Aspekte der Datenverarbeitung eingegangen werden. Insbesondere wird die Methode **testNonRating**, die die Expertenliste, die Kriterienliste und die Konstruktionsvorschlagsliste für die Bereichs- und Festpunktkriterien verknüpft, und die Methode **getRankingDictionary** zur Nutzwertberechnung vorgestellt, da sie die eigentliche Synthese von Mensch und Computer in dieser Arbeit vollziehen.

7.1 Daten

7.1.1 Datenspeicherung

Die Datenbanken des DESIGN-ANALYZERS werden durch Files repräsentiert. Das Smalltalk-System sieht zwar auch einen sogenannten Database-Modeler für das Arbeiten mit Datenbanken (mit SQL) vor, der jedoch leider nicht einsetzbar war. Der Einsatz eines Datenbanksystems wäre auch insoweit sinnvoll gewesen, weil die Zugriffsmethoden auf Teilmengen der Kriterienlisten, der Expertenmenge und der Produktmodelle während des Entscheidungsalgorithmus durch SQL-Anfragen hätten realisiert werden können. Die entsprechenden Methoden sind nun zusätzlich implementiert und finden sich zum Beispiel für die Klasse **RatingList** (Liste von Kriterien) in den Methoden der Kategorie **filtering**. Im folgenden werden nun die Pfadnamen mit der Endung (*.Endung) der jeweiligen Datei aufgelistet, unter denen die Daten abgelegt sind. Sollte das System unter einem anderen HOME-Directory installiert werden, müssen die Pfade der jeweiligen LOAD- oder SAVE-Methoden geändert werden. Diese Methoden sind immer unter der Methodenkategorie *fileIn/Out* zusammengefaßt.

Domänenspez. Designspezifikation:	/home/tautges/small/databases/domain/*.typ
Domänenspezifikationen:	/home/tautges/small/databases/domain/*.domain
Designspezifikationen:	/home/tautges/small/databases/formal/*.form
Produktmodelle:	/home/tautges/small/databases/construct/*.cad
Bewertungsergebnisse:	/home/tautges/small/databases/ratings/*.rate

Das Speichern und Laden der Daten wird mit dem Binary Object Storage System (BOSS) des SmallTalk-Systems durchgeführt. BOSS ermöglicht das schnelle Speichern der Objekte und die automatische Generierung der abgespeicherten Instanzen beim Laden. Ein Parser ist nicht nötig. Eine Ausnahme bilden die Produktmodelle, die während der Testläufe verwendet wurden. Sie sind im Gegensatz zu den anderen Daten auch als File vollständig lesbar, benötigen aber einen Parser und einen Compiler, mit dem die Instanzen beim Laden im Smalltalk-System generiert werden. Der Parser und der zugehörige Compiler wurden schon vorher im IMCOD-Projekt erstellt.

7.1.2 Datenverknüpfungen aus DB-Sicht

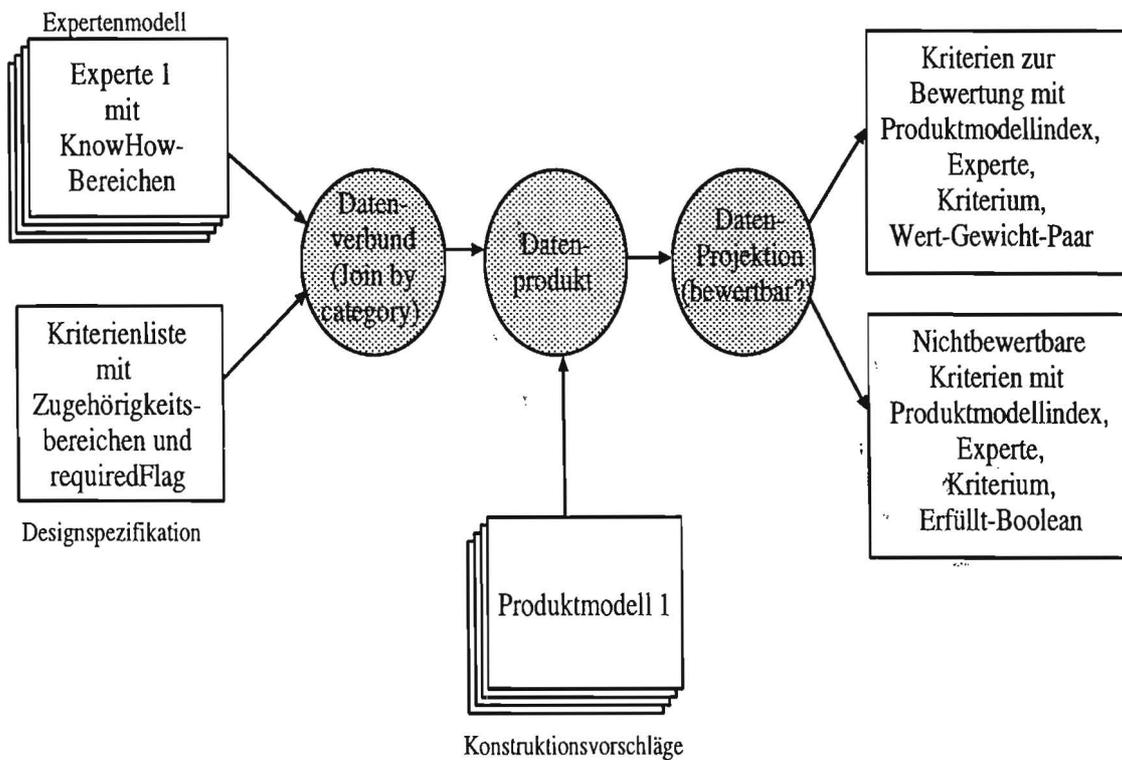


Abbildung 7.1: Die Verknüpfung der Kriterien, Experten und Konstruktionsvorschläge aus Datenbanksicht

Man kann die Verknüpfung der Kriterien aus der Designspezifikation, dem Expertenmodell aus der Domänenspezifikation und den Konstruktionsvorschlägen mit der Verknüpfung von Daten einer relationalen Datenbank vergleichen. Die Kriterienliste und die Expertenliste werden durch einen natürlichen Verbund (Data-join) unter der Bedingung der gleichen Kategoriezugehörigkeit zusammengeführt. Der resultierende Datensatz wird dann jeweils

mit den Indizes der Produktmodelle verknüpft, was dem kartesischen Produkt in der Datenbankdomäne entspricht. Da man in der Designspezifikation die Zielkriterien und die Bereichskriterien separat bearbeiten kann, teilt man die Daten entsprechend einem der zwei Datensätze mit den Slots *weight* und *value* bzw. *success* zu.

Die Zusammenstellung der Kriterium-Experte-KV-Tripel in zwei Datensätze ist auch deshalb zu favorisieren, da das SmallTalk-System bei der Generierung der Benutzeroberfläche die Klasse *DataSet* zur Darstellung von Datensätzen in Tabellenform zur Verfügung stellt. Die Datensätze können dann durch Notebooks – indizierte "Wörterbücher" – katalogisiert nach den einzelnen Slots dargestellt werden. Die Indexierung erfolgt über einen oder zwei Schlüssel (*minorKeys* und *majorKeys*), so daß zum Beispiel nur der Kriteriendatensatz eines bestimmten Experten und eines KVs dargestellt wird und der komplette Datensatz anhand dieser Schlüssel katalogisiert ist. Mit diesem Feature wird auch bei der Verwaltung der Expertenliste und der Designspezifikation gearbeitet (siehe dazu auch die Fensterdarstellung in Kapitel 8).

Im folgenden wird die Methode **testNonRating** als Quellcode dargestellt. Durch sie wird für die Festpunkt- und Bereichskriterien der Kriteriendatensatz zusammengestellt (*testRating* analog für Zielkriterien). Die Aufteilung der Verknüpfungsmethoden für Zielkriterien und Bereichs- oder Festpunktkriterien ermöglicht die separate Durchführung des Entscheidungsalgorithmus für die Kriterientypen. Die do-Schleifen beginnen bei der Kriterienliste, laufen dann über die Expertenliste und schließlich über die Liste der Konstruktionsvorschläge. Dabei werden die Experte_i mit Kompetenzbereich *low* der *nonRatingListLow* zugefügt.

Im zweiten Teil der Methode wird überprüft, ob jedes Kriterium durch einen Experten vertreten ist. Wenn nicht, werden die entsprechenden Experten der *nonRatingListLow* ausgewählt und aus der bisherigen Liste gestrichen. Die verbleibenden "Low-Experten" werden zur weiteren Verwendung jedoch am Ende der Methode mit zurückgegeben. Sie werden benötigt, wenn ein vorher zugeordneter Experte zu einem Kriterium wider Erwarten keine Aussage machen kann.

testNonRating

*"test the required requests for every workpiece. First the method creates a criteriaList for experts with 'high' knowledge-status and 'low'-knowledgestatus and then move low-knowledge-criterias to 'nonRatingListHigh' if there is no expert in that criteria with a high-knowledge-status.
If the workpiece fails in one request, the workpiece is suggested to be removed from the workpieceList"*

| nonRatingListHigh nonRatingListLow requiredList crits removeList |

```

nonRatingListHigh := RatingList new.
nonRatingListLow := RatingList new.
requiredList := DesignSpecList requiredRequests.
requiredList do:
  [:request | Domain experts do:
    [:expert |
      (expert knowHowCategories includes: request category) ifTrue:
        [WorkpieceList keysDo:
          [:workpiece |
            (expert statusIsHigh: request category)
            ifTrue:
              [nonRatingListHigh add:
                (NonRatingCriteria designIndex: workpiece
                  criterium: request attribute
                  expert: expert
                  success: 'succeed')]
            ifFalse:
              [nonRatingListLow add:
                (NonRatingCriteria designIndex: workpiece
                  criterium: request attribute
                  expert: expert
                  success: 'succeed')]]]]].

crits := nonRatingListHigh criteria.
removeList := nonRatingListLow copyWithout: nil.
requiredList do: [:request |
  (crits includes: request) ifFalse:
    [nonRatingListLow do:
      [:crit |
        (crit criterium = request attribute) ifTrue:
          [nonRatingListHigh add: crit.
            removeList remove: crit]]].
  ^ (nonRatingListHigh -> removeList)

```

7.2 Nutzwertkalkulation

Die Berechnungsvorschrift wurde im Kapitel 6 schon beschrieben. Hier wird die entsprechende Smalltalkmethode beschrieben, die einige wichtige Feinheiten enthält. Sie wird stets aufgerufen, wenn ein entscheidungsbeeinflussender Parameter sich ändert. Zunächst wird aus der gesamten Kriterienmenge eines KVs (*ratings*) die Untermenge bestimmt, die jedes Kriterium-Attribut nur einmal enthält (*crits*) und deren Größe bestimmt (*critCap*). Den Variablen *totalPoints*, *singleWeight* und *singlePoint* wird eine Instanz der Klasse **RatingValue** zugewiesen, deren Instanzmethoden auch arithmetische Operationen

wie Addition und Division umfassen. In der jetzigen Version des DESIGN-ANALYZERS wird die Arithmetik der Gleitkommazahlen benutzt. Die zusätzliche Klassendefinition ermöglicht es auch andere Wertzuweisungen für die Variablen *weight* und *value* zu benutzen, indem die entsprechenden Konstanten in der Klasse **RatingValue** geändert werden. Wenn der Typ der Variablen sich ändern sollte – zum Beispiel FUZZY-Zahlen oder negative Integer – so muß eventuell auch die Arithmetik angepaßt werden. Mao-Jiun J. Wang und Tien-Chien Chang [WTC95] schlagen zum Beispiel die Belegung der Werte mit FUZZY-Sets vor (siehe auch Kapitel 9.2).

Die Methode liefert eine nach erzielten Punkten geordnete Liste zurück.

getRankingDictionary

“compute the new ranking positions and the points for each workpiece”

```
| li crits singlePoint singleWeight num totalPoints ratings critCap totalWeight |
li := SortedCollection new.
self indexes
  do:
    [:in |
      ratings := self for: in. “RatingCriteria of workpiece: in”
      crits := ratings requiredCriteriaSet. “CriteriaSet of workpiece: in”
      totalPoints := RatingValue new.
      totalWeight := 0.
      critCap := crits size.
      crits
        do:
          [:crit |
            singlePoint := RatingValue new.
            singleWeight := RatingValue new.
            num := 0.
            (ratings ofRequiredCriterium: crit attribute)
              do:
                [:singleCrit |
                  singlePoint := singlePoint +
                    (RatingValue newR: singleCrit value).
                  singleWeight := singleWeight +
                    (RatingValue newW: singleCrit weight).
                  num := num + 1].
            totalWeight := totalWeight + (singleWeight / num).
            totalPoints := totalPoints +
              (singleWeight * singlePoint / num / num)].
```

```
critCap = 0 ifTrue: [li add: (MyAssociation key: in value: 0.0)]
              ifFalse: [li add: (MyAssociation key: in value: (totalPoints /
totalWeight ))]].
      ^li reverse)
```

Kapitel 8

Die Benutzeroberfläche

Die Oberfläche besteht hauptsächlich aus drei Fenstern (Domain Specification Window, Design Specification Window, Rating and Ranking Window), die von zusätzlichen Editoren (Value Editor, Integration in IMCOD), oder Auswahlfenstern (Category Selection Window, Checking Window) unterstützt werden. Der DESIGN-ANALYZER wird von einer Konsole aus gesteuert. Die Fenster werden mit ihren wichtigsten Features im folgenden kurz vorgestellt. Weitergehende Informationen können in den ersten drei Fenstern durch die HELP-Funktion aus dem Menü entnommen werden. Alle Buttons eines Fensters sind auch über die Menüleisten (falls vorhanden) aktivierbar.

8.1 IMCOD-Console

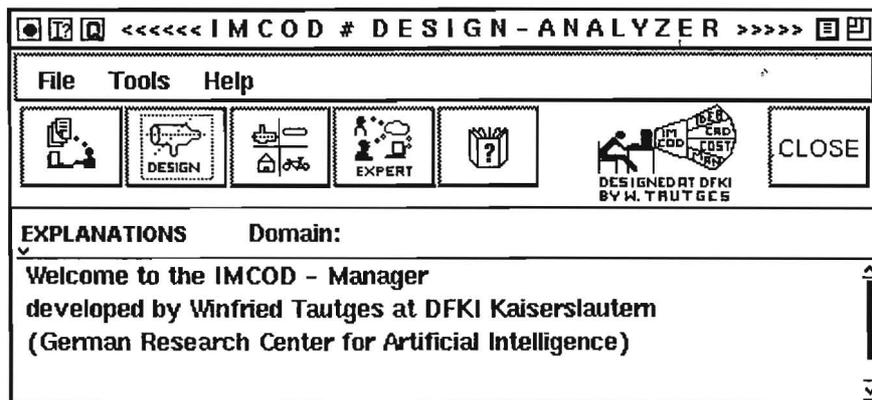


Abbildung 8.1: Konsole des DESIGN-ANALYZERS

Die Konsole des DESIGN-ANALYZERS hat zwei Funktionen:

- Start- und Ladefunktionen
- Transscript-Fenster zur Übermittlung von Nachrichten an den Benutzer

Die Startfunktionen umfassen das Domain Specification Window, das Design Specification Window, ein CAD-Tool und ein Hilfeprogramm. Diese Tools sind sowohl vom Menü als auch über Picture-Buttons unterhalb der Menüleiste aktivierbar.

Um mit dem DESIGN-ANALYZER fachgerecht arbeiten zu können, sollte man hier gleich zu Beginn die gewünschte Domäne laden.

Die Konsole fungiert als sogenannter Master der übrigen Fenster. Das bedeutet, wenn das Master-Fenster geschlossen oder als Item dargestellt wird, so geschieht dies auch mit allen Slave-Windows. Diese Option ist besonders nützlich, wenn man zwischendurch andere Aufgaben erledigen will oder die Fenstervielfalt des Smalltalksystems zu groß wird.

8.2 DOMAIN-SPEC-Window

Im Domain Specification Window wird wie auch im Rating and Ranking Window mit den bereits in Kapitel 7 erwähnten Notebooks gearbeitet, die es ermöglichen, wie in einem Wörterbuch mit verschiedenen Indizes zu blättern. Das erhöht die Übersicht und reduziert die Anzahl der offenen Fenster während einer Sitzung. Die sogenannten Major-Keys an der rechten Seite des Notebooks zeigen die integrierten Experten an. Durch Anklicken erhält man die entsprechenden Einträge in der Interfacebibliothek und die Liste mit den Kategorie-Status-Paaren. Der Add-Button bringt das Category Selection Window auf den Bildschirm, mit dem man dem ausgewählten Experten in einer Multi-Selektions-Liste gleich mehrere bereits definierte oder neue Kategorien zuordnen kann. Die Liste wird für die Sitzung um die neu definierten Kategorien ergänzt.

Die Interface-Bibliothek läßt sich durch die Betätigung der **Edit Interface BIB**-Taste editieren. Das aufgerufene Fenster verschafft dem Benutzer Zugang zu der von Fredj Dridi implementierten Benutzeroberfläche zur Beschreibung der Schnittstellenfunktionen. Eine genaue Beschreibung von 'Integration in IMCOD' findet sich in [Dri94].

Der Status eines Experten bezüglich eines Kompetenzbereichs läßt sich durch das Anklicken des entsprechenden Feldes und die anschließende Auswahl aus einem Pull-Down-Menü ändern. Der domänenspezifische Kriterienkatalog wird durch den Pfadnamen der Datei repräsentiert. Dies ermöglicht die mehrfache Benutzung der Kriterien, so daß sie bei veränderten Domänenmodellen nicht stets wieder neu selektiert und formalisiert werden müssen. Experten oder ihre Kompetenzbereiche lassen sich natürlich auch wieder löschen. Der Menüeintrag **Update Domainspec** paßt das geladene Domänenmodell an die Veränderungen an.

8.3 DESIGN-SPEC-Window

Das Design Specification Window zeigt den Datensatz der Kriterien. Beim erstmaligen Aufruf ist dies die domänenspezifische Designspezifikation, sonst die bearbeitete Designspezifikation. Die Liste läßt sich durch Menüpunkte unter **Edit** nach Kategorien oder nach der Einteilung in 'required' (Kästchen ausgefüllt) oder 'not required' ordnen. Einzelne Kriterien können hinzugefügt, neu editiert oder gelöscht werden. Man selektiert Zeilen in dem Datensatz durch das Anklicken der Zeile. Dies gilt auch für die Kompe-

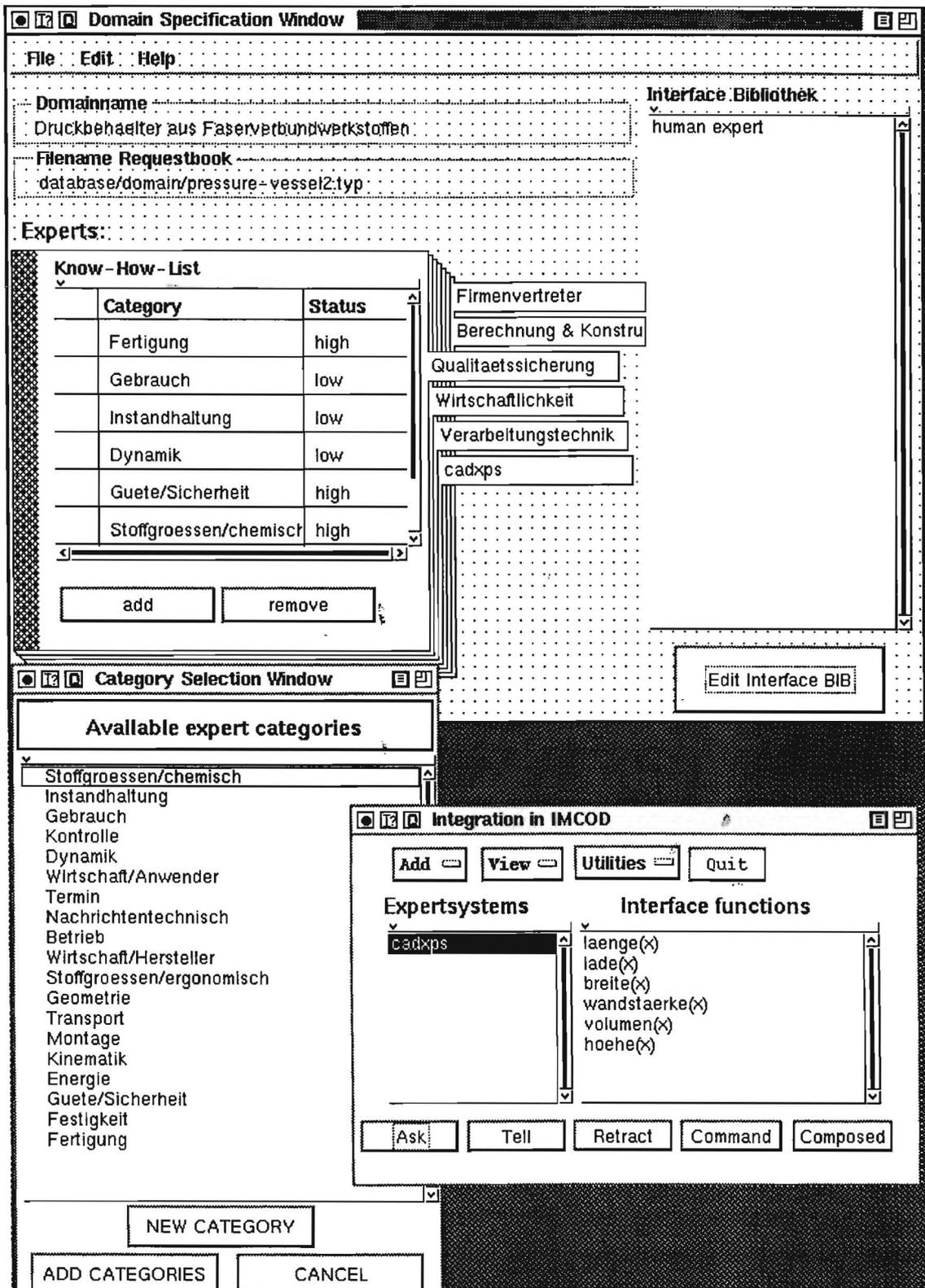


Abbildung 8.2: Domain Specification Window mit Kategorie-Auswahl-Fenster und Interface-Bibliothek-Editor

Request	Value	Weight	Category	Required
Durchmesser	Interval [89 , 91] cm		Stoffgroessen/chemisc	<input checked="" type="checkbox"/>
Temperaturbestaendigkeit	Interval [-15 , +40] Grad Celsius		Stoffgroessen/chemisc	<input checked="" type="checkbox"/>
Recyclierfaehigkeit	80 (best) in: [40 , 100] %	medium	Stoffgroessen/chemisc	<input type="checkbox"/>
Grenztueckkosten	0 (best) in: [0 , 70] DM/Stck	high	Wirtschaft/Hersteller	<input type="checkbox"/>
Stossfestigkeit	70 (best) in: [50 , 70] km/h	high	Stoffgroessen/ergonom	<input type="checkbox"/>
Lackhaftung	komplett (fix)		Gebrauch	<input checked="" type="checkbox"/>
Volumen	Interval [67 , 70] liter		Geometrie	<input checked="" type="checkbox"/>
Wandstaerke	1.1 (best) in: [1.0 , 1.2] cm	very high	Geometrie	<input type="checkbox"/>
Pruefdruck	17 (fix) bar		Guete/Sicherheit	<input checked="" type="checkbox"/>
Abnahme TUV	DIN ISO 9000 (fix)		Guete/Sicherheit	<input checked="" type="checkbox"/>
Produktionsanzahl	8000 (best) in: [7000 , 8000] Stck	medium	Fertigung	<input type="checkbox"/>

Abbildung 8.3: Designspecification-Window des DESIGN-ANALYZERS

Attribute	Weight	Category
Durchmesser		Stoffgroessen/chemisch
Recyclierfaehigkeit	mediu	Stoffgroessen/chemisch
Grenztueckkosten	high	Wirtschaft/Hersteller
Pruefdruck		Guete/Sicherheit
Abnahme TUV		Guete/Sicherheit

Buttons: Add selection, Cancel

Abbildung 8.4: Checking-Window mit fehlenden domänenspezifischen Kriterien

tenzbereiche eines Experten im Domain Specification Window, den Kriterienkatalog des Checking-Windows und die Einträge im Rating and Ranking Window. Soll ein Kriterium editiert werden, wird der Value-Editor aufgerufen (siehe nächster Abschnitt).

Eine Designspezifikation kann geladen, gespeichert und zum Drucken an den lokalen Drucker gesendet werden. Bevor man den Bewertungsalgorithmus startet, sollte man noch die Checking-Option verwenden, mit dem man die domänenspezifischen Kriterien im Checking-Window aufgelistet bekommt (Abbildung 8.4). Auch diese können selbstverständlich noch editiert werden. Danach kann man durch **Load Workpieces** die Produktmodelle aus der entsprechenden Datenbank laden und schließlich das Rating and Ranking Window durch den **RANKING**-Button aktivieren. Die Schritte werden bei erfolgreicher Ausführung in der Konsole dokumentiert. Bei mißlungenen Aktionen oder fehlenden Schritten werden Dialogfenster mit Fehlermeldungen aufgerufen.

8.4 Value-Editor

Fixpoint Interval Interval with tendency

Request: Temperaturbestaendigkeit

Category: Stoffgroessen/chemisch

Minimum: -15

Best:

Maximum: +40

Default-Weight:

Type / Unit: Grad Celsius

Required Request

Accept Cancel

Abbildung 8.5: ValueEditor zum editieren der Kriterien

Der Value-Editor zeigt die notwendigen Slots der drei Kriterienarten. Durch die Radio-Buttons am oberen Fensterrand kann man das Kriterium den Festpunktkriterien (Fixpoint), den Bereichskriterien (Interval) oder den Zielkriterien (Interval with tendencies) zuzuordnen. Entsprechend der Auswahl werden die editierbaren Eingabefelder aktiviert. Nicht

editierbare Felder (im Abbildung 8.5 'Best' und 'Default-Weight') werden gesperrt und blass gezeichnet. Der Benutzer kann jederzeit den Vorgang durch **Cancel** abbrechen, jedoch erst bei vollständig ausgefüllten Eingabefeldern das Kriterium der Designspezifikation hinzufügen, sodaß der spätere Bewertungsvorgang auch korrekt ablaufen kann. Semantische Korrektheit wird jedoch nicht überprüft (zum Beispiel die Zuordnung Type='liter' und Request='Temperaturbeständigkeit'). Um den Slot Category zu belegen, kann man einen Bereich aus einer Liste auswählen, die die Kompetenzbereiche enthält, die durch Experten im Domänenmodell abgedeckt sind. Dadurch wird die Anzahl der nicht bewertbaren Kriterien klein gehalten. Das Gewicht oder Defaultgewicht wird ebenfalls aus einer Liste ausgewählt.

8.5 RatingAndRanking-Window

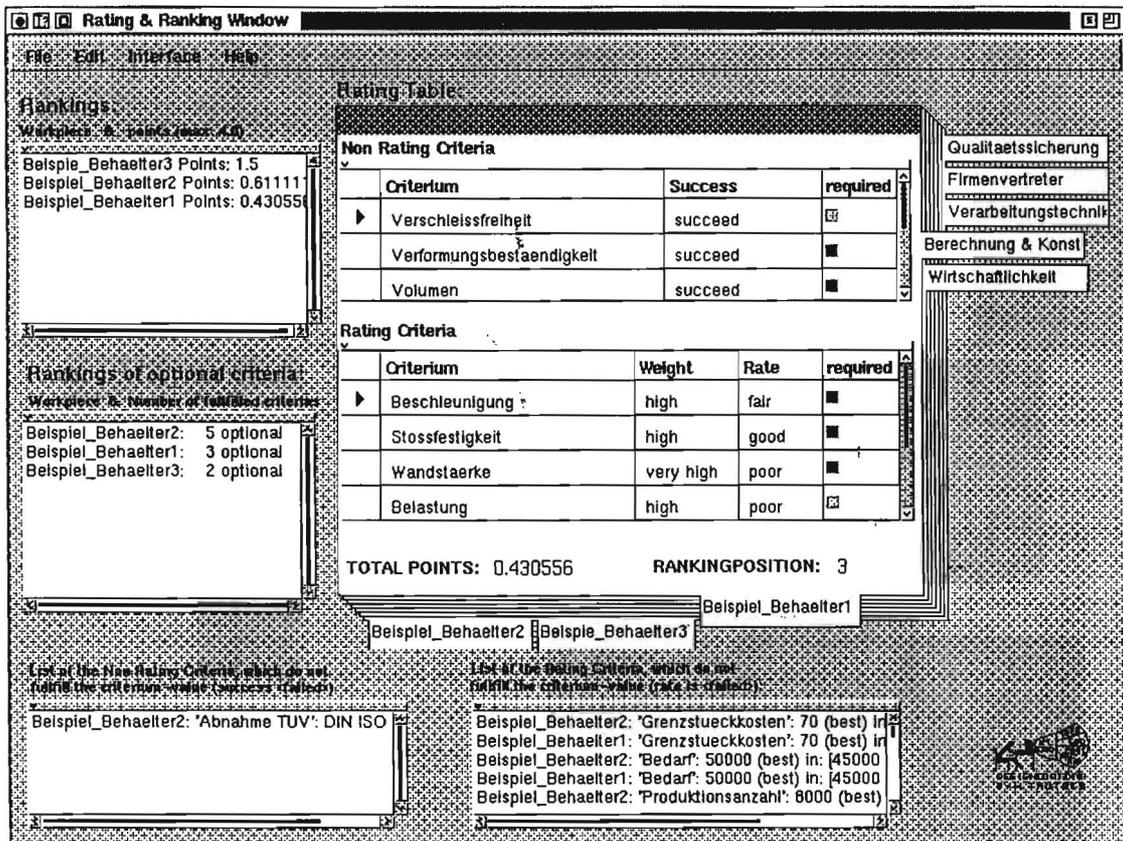


Abbildung 8.6: Rating-Window des DESIGN-ANALYZERS

Das RatingAndRanking-Window faßt den Input und den Output des Entscheidungs-Algorithmus in einem Fenster zusammen. Veränderungen des Inputs werden im Output sofort berücksichtigt. Das Fenster zeigt rechts ein Notebook mit den Experte-KV-Kombinationen, links zwei Ranglisten für bewertete Kriterien und die Anzahl der erfüllten

optionalen Kriterien und unten zwei Listen mit den KVs, die einzelne Anforderungen nicht erfüllen. Durch Anklicken eines Konstruktionsvorschlags und eines Experten werden alle Non Rating Criteria und alle Rating Criteria dieser Kombination angezeigt. Die Werte für Success, Weight und Rate sind Default-Werte und können durch Auswahl aus einem Pull-down-Menü verändert werden (entsprechendes Feld anklicken). Der DESIGN-ANALYZER berechnet daraufhin sofort die neuen Punktestände bzw. Rangfolgen und zeigt diese in den entsprechenden Listen auf der linken Seite, bzw. am unteren Rand des Notebooks für den angewählten Vorschlag an. Mithilfe des Mausclicks kann man jederzeit in den rechten Spalten der Kriterientypen das Requiredflag von true auf false (und umgekehrt) "switchen" und damit die Entscheidung beeinflussen.

In der linken unteren Liste werden alle Konstruktionsvorschläge aufgelistet, die in einem Non Rating Criteria ein 'failed' zugewiesen bekamen. Entsprechend werden im rechten unteren Listenfeld die Vorschläge aufgelistet, denen in einem Zielkriterium der Wert 'failed' zugeordnet wurde. In diesen Fällen kann nun diskutiert werden, ob der Vorschlag abgelehnt oder überarbeitet zu dem Bewertungsalgorithmus zugelassen werden soll. Vorschläge können wie auch Experten oder einzelne Kriterien aus dem Notebook mithilfe des Menüpunkts **Edit** gelöscht werden.

Die Ergebnisse können natürlich gespeichert bzw. wieder geladen werden. Außerdem ist auch noch eine Druckoption im Menü vorhanden. Mit dem Menüpunkt **Interface** wird eine Schnittstelle für zusätzliche Anfragen an die angeschlossenen Expertensysteme zur Verfügung gestellt. Hier kann auch ein integriertes CAD-System durch **Show Workpiece** aufgerufen werden, um den selektierten Konstruktionsvorschlag darzustellen.

8.5.1 Algorithmus einer Beispielsitzung

Im folgenden wird ein vereinfachter Algorithmus angegeben, der den Ablauf einer Beispielsitzung beschreibt. Die erwähnten Buttons und Menüfelder sind umrandet.

- A Aufruf des DESIGN-ANALYZERS durch *'ImcodManager open'*.
- B Falls die Domänenspezifikation für konkreten Fall existiert, diese durch Load Domain ... im File-Menü oder mittleren Button aus der Domänenmodell-DB laden, - > C;
sonst Domänenspezifikation für die Beispielsitzung modellieren - > B.1
 - B.1 Mit EXPERT-Button oder DomainSpecs im Tools-Menü das Domain Specification Window aufrufen.
 - B.2 Domänenname eintragen
 - B.3 Pfad der domänenspezifischen Designspezifikation angeben. Falls diese noch nicht existiert, muß sie noch editiert werden - > C
 - B.4 Durch Add New Expert im Edit-Menü Experten spezifizieren und durch den add-Button Kategorien aus dem Kategorie-Auswahl-Fenster zuordnen.

- B.5 Für jede Kategorie den Status durch Auswahl aus dem Pull-down-Menü bestimmen.
- B.6 Für die Expertensysteme durch Betätigen des `Edit Interface BIB`-Buttons die Schnittstellen-Bibliothek editieren.
- B.7 Im File-Menü mit `Save As ...` die Domänenspezifikation abspeichern und durch `Update Domain` die Änderungen für die Beispielsitzung aktualisieren.
- B.8 Domain Specification Window mit `Quit` verlassen.
- C Mit dem linken Button der Konsole oder `DesignSpecs`-Button des Tools-Menüs das Design Specification Window aufrufen.
- D Das Fenster zeigt die domänenspezifischen Kriterien. Entweder diese einzeln anwählen und durch `Edit Selected` im Edit-Menü editieren, löschen `Remove Row` oder neue Kriterien hinzufügen `Add Row` und editieren. Jeder Editiervorgang ruft den Value-Editor auf – > D.1
- D.1 Der Value-Editor zeigt das neu zu editierende Kriterium. Durch Anklicken einer der drei Radio-Buttons in der obersten Reihe werden die entsprechenden Eingabefelder der Maske zur Bearbeitung der gewählten Kriterienart freigegeben.
- D.2 Angaben in die Felder eintragen bzw. für die Slots `< Category >` und `Default-Weight >` aus den Pull-down-Menüs auswählen.
- D.3 Mit `Accept` läßt sich das Kriterium in die Designspezifikation einfügen, wenn alle notwendigen Eintragungen erfolgt sind. Sollte schon ein Kriterium mit gleichem Namen existieren, wird dieses überschrieben. Mit Cancel läßt sich der Vorgang jederzeit abbrechen.
- E Mit dem `Check`-Button Checking-Window aufrufen, um eventuell fehlende Kriterien aus der Auflistung mithilfe automatisch aufgerufenen Value-Editors zu ergänzen – > D.1. Sonst weiter mit – > F
- F Mit `Save ...` die Designspezifikation abspeichern und anschließend mit `Use As DesignSpecification` die Änderungen akzeptieren.
- G Mit `Load Workpieces ...` aus dem Decision-Menü die zu bewertenden Konstruktionsvorschläge aus der Produktmodell-DB laden.
- H Mit dem `Ranking`-Button das RatingAndRanking Window starten. Die Bewertung der Kriterien durch die Expertensysteme erfolgt automatisch und muß nicht manuell gestartet werden. Dadurch kann sich der Fensteraufbau verzögern. Sollte kein Werkstück die Anforderungen der Festpunkt- und Bereichskriterien erfüllen, dann – > D
- I Durch die Pull-down-Menüs der Success-Spalte bzw. der Weight- und Rate-Spalte die Bewertung noch nicht bewerteter Kriterien für jeden Konstruktionsvorschlag

vornehmen. Bei Änderungen wird die Nutzwertkalkulation stets erneut durchgeführt und die Ergebnisse angezeigt.

J Mithilfe der **Remove x...**-Befehle des Edit-Menüs können Experten, Konstruktionsvorschläge oder einzelne Kriterien bei Bedarf durch vorheriges Anklicken gelöscht werden.

K Nach vollständiger Bewertung die Bewertungsliste mit **Save** abspeichern, den DESIGN-ANALYZER verlassen und im Produktionsprozeß mit dem ausgewählten Konstruktionsvorschlag fortfahren.

Kapitel 9

Zu guter Letzt

"Die Nutzenkalkulation expliziert nur das, was in die Datenbasis als Information und Vorentscheidung schon eingegangen ist. Die Handlung ist genauso rational oder irrational, genauso gelungen oder mißlungen, wie die Daten angemessen sind, auf deren Grundlage die Kalkulation vorgenommen wird." (Ottfried Höffe)

9.1 Berechnung statt Entscheidung?

Es fragt sich am Ende der Arbeit, ob das Konzept der Verknüpfung menschlicher Expertise und wissensverarbeitenden Expertensystemen überhaupt zu einer echten Entscheidung im Designproblem führt. Oder ist es nur eine schlichte, deterministische Nutzenkalkulation, die dem (scheinbar) besten Konstruktionsvorschlag die höchste Punktzahl zuordnet?

Ein entscheidungsunterstützendes System sollte wohl beide Elemente enthalten, um dem Wortfeld "Unterstützung" und "Entscheidung" gleichsam gerecht zu werden. Einerseits soll es den menschlichen Experten durch das Bereitstellen strukturierten lokalen Wissens die Entscheidungsfindung erleichtern; andererseits soll es mehr sein als ein schnelles Informationssystem, indem es eindeutig formalisierbare Kriterien der Vorzugswahl bewertet, die keinen globalen Einblick in das Realsystem der Entscheidungssituation verlangen. Das Überprüfen des Volumeninhalts mit Hilfe der Geometriedaten ist sicherlich einem formalen System anzuvertrauen, vor allem da es eine simple Berechnung darstellt. Schwieriger wird hingegen die Programmierung der dynamischen Schnittstelle eines integrierten Expertensystems sein, wenn es zusätzlich zu den Bereichs- oder Festpunktkriterien auch den Erfüllungsgrad der Zielkriterien festlegen muß. Dies entspricht der Umwandlung einer einmaligen Entscheidung in einen beliebig oft wiederholbaren Berechnungsvorgang.

Eine wichtige Forderung an entscheidungsunterstützende Systeme diesbezüglich ist deren Einsatz als Werkzeug. Das so oft benutzte englische Wort 'tool' impliziert das Wissen des Benutzers über den Umgang und die Fähigkeiten seines Werkzeugs. Dann besteht nämlich die Möglichkeit, das verwendete Entscheidungsmodell zu durchschauen und einschätzen zu können und letztlich echte Entscheidungen (Fehlen von entscheidenden Gründen gegen oder für eine Handlungsalternative) nicht aus der Hand zu geben.

Sogar wenn eine Designspezifikation nur aus berechenbaren Kriterien bestände, würden in der Aufgabenformulierungs- und strukturierungsphase genügend echte Entscheidungen

getroffen werden, die dem Entscheidungsträger eine Identifikation mit der letztlich getroffenen (Vorzugs-)Wahl ermöglichen. Hier liegt vielleicht der Unterschied zwischen Wissen und Können. Bezogen auf die Synthese menschlicher und computerunterstützter Expertise im DESIGN-ANALYZER kann man dem menschlichen Experten eher ein globaleres "Können" (know-how) und den Expertensystemen ein lokales "Wissen" (know-what) zuordnen.

Erstere könnten mehr, wenn sie wüßten; die anderen wüßten noch mehr, wenn sie könnten.

9.2 Ausbaumöglichkeiten und andere Groupware-Ansätze

Der Design-Manager des IMCOD-Projekts bietet durch das Konzept des zentralen Managements die Möglichkeit den Entscheidungsprozeß durch den Einsatz weiterer Software zu perfektionieren. Vorgesehen ist der schon in Kapitel 4 beschriebene Einsatz des COKAM+-Systems zur Wissensakquisition.

Die Software ist als Ein-Platz-System implementiert. In dem Zusammenhang wäre ein Mehr-Platz-System bei einer Expertensitzung wünschenswert, das dann durch entsprechende Zugriffskontrollen die Bearbeitung der Kriterien regelt. Natürlich sollte der Rechnerinsatz nicht die Kommunikation durch gesprochene Sprache behindern. Die Designspezifikation des DESIGN-ANALYZER könnte als Ausgangspunkt der Diskussion fungieren und den Entscheidungsprozeß dokumentierend begleiten. Die durch Zielfunktionen ausgewählte Alternative könnte noch einmal mit den Wertvorstellungen der Experten verglichen werden, die in einer Diskussion wieder stärker mit eingehen als die punktbewerteten Einzelziele.

Interessant wäre in diesem Zusammenhang auch der Einsatz einer Simulationskomponente mit Sensitivitätsanalyse. Durch die Veränderung der entscheidungsbeeinflussenden Parameter - Kompetenzstufen, Kriterien mit Gewichten und Bewertungen und Requiredflag - könnte eine Kompromißlösung erarbeitet werden, wenn das zieleorientiert gewonnene Entscheidungsergebnis aufgrund abweichender (Wert-)Vorstellungen der menschlichen Experten nicht von der Gruppe akzeptiert wird. Das "gute Gefühl" bei einer Entscheidung durch Experten sollte diesen nicht genommen werden. Die Sensitivitätsanalyse stellt die Parameter heraus, die das Ergebnis am stärksten in negativer oder positiver Richtung beeinflussen. In diesem Zusammenhang wäre eine graphische Representation der Ergebnisse hilfreich, da diese die Auswertung mehrerer Parameter für das menschliche Auge erleichtert.

Die Dokumentationsfunktion des DESIGN-ANALYZERS ist allerdings nur eine von mehreren Möglichkeiten des Einsatzes eines Group Decision Support Systems. Eine intensive Weiterentwicklung wird derzeit im betriebswirtschaftlichen Bereich betrieben, wo das GDSS stärker von den menschlichen Experten isoliert ist und eher als Koordinationswerkzeug fungiert. Das Anwendungsszenario erstreckt sich hier von Verhandlungsdokumentation bis zur 'hitzigen' Debatte über den Cyberspace. Die Funktionsbereiche der Systeme unterscheiden sich dabei in ihrem Interventionsgrad.

Eine unterstützende Rolle mit niedrigem Interventionsgrad übernimmt zum Beispiel das GDSS MEDIATOR von Jarke et al. [JJS87]. Das System ordnet jeder Handlungsalterna-

tive für jeden Entscheidungsträger seine eigene Nutzenfunktion zu ohne die Alternative explizit zu 'kennen'. Durch Veränderung der Zielgrößen können die Präferenzlisten der Alternativen verändert werden, Handlungsalternativen hinzugenommen oder weggelassen werden. Die Entscheidungsträger versuchen durch Anpassung der Listen zu einem Kompromißvorschlag zu kommen.

Ein Verfahren von Korhonen et al. [KWZ82] berechnet die beste Alternative mithilfe linearer Programmierung. Hierzu werden alle Zielfunktionen aller Entscheidungsträger aggregiert. Wenn eine beste Lösung gefunden ist, werden die nächstliegenden Funktionen verglichen und versucht deren Zielfunktionseckpunkte mit in die bis dahin beste Lösung einzubinden. Das Verfahren ist beendet, wenn sich keine Verbesserungsmöglichkeit mehr bietet.

Mit zunehmendem Interventionsgrad des GDSS steigt auch die Vermittlerrolle des Systems und drängt die Experten eher zu einem Kompromißvorschlag. Das Verfahren von Isermann [Ise85] basiert auf Mehrheitsentscheidung, d.h. wenn mindestens die Hälfte der Entscheidungsträger sich über denselben Vorschlag einig sind, wird dieser vom System favorisiert. Das System versucht eine Mehrheitskoalition zu bilden, indem es die vorläufigen Vorschläge der Verhandlungspartner vergleicht und eine Kompromißlösung generiert.

Im Maschinenbau wird häufig mit Konstruktionskatalogen gearbeitet ([Rot82],[Wü88]), in denen Konstruktionsvarianten für Teilfunktionen einer komplexen Designaufgabe katalogisiert sind. Sie werden den Anforderungen entsprechend ausgewählt, und die entstandenen Kompositionen können nach einer Gewichtung der Teilfunktionen in einem morphologischen Kasten bewertet werden. Die Schwierigkeit liegt hier in der Verknüpfung und der Kompatibilität der Teillösungen.

Eine weitere Variationsmöglichkeit liegt in der Berechnungsvorschrift für die beste Lösung. Dieser Aspekt wurde auch schon bei der Implementierung des DESIGN-ANALYZERS beachtet. Mao-Jiun J. Wang und Tien-Chien Chang [WTC95] schlagen zum Beispiel die Belegung der Werte und Gewichte mit FUZZY-Sets vor, um die Umwandlung qualitativer Angaben der Experten (zum Beispiel "bezüglich dieses Kriteriums nicht schlecht" oder "fair" beim DESIGN-ANALYZER) in quantitative Angaben zu verbessern. Während bei der reinen Mittelwertbildung die Abweichung der Kriterienbewertung nicht zum Tragen kommt, können bei der Berechnung mit FUZZY-Sets Unterschiede und Tendenzen festgestellt werden. Eine diesbezügliche Erweiterung des DESIGN-ANALYZERS wird in Betracht gezogen, da es die Entscheidungsunterstützung besonders bei "knappen" Entscheidungen verbessern würde.

Obwohl die Stufen der Nutzenbewertung beliebig ausgebaut werden können, wird davon abgeraten mehr als neun Level zu definieren, da dies die generelle Grenze des menschlichen Differenzierungsvermögens ist. Der DESIGN-ANALYZER bietet durch die Kombination von Wert und Gewicht zwar $5 * 5 = 25$ Möglichkeiten der Bewertung für ein Kriterium, erleichtert dem Experten jedoch durch die inhaltliche Trennung der beiden Slots die Bewertungsaufgabe.

9.3 Zusammenfassung

Um Fehlentscheidungen in einem Designprozeß zu vermeiden, müssen alle relevanten Informationen schon während der Entwicklungsphase zusammengetragen und ausgewertet werden. Die wichtigste Informationsquelle eines Unternehmens ist dabei die die Erfahrungen der Fachexperten der Unternehmung und fachspartenspezifische Informationssysteme. Die vorliegende Arbeit beschäftigt sich daher hauptsächlich mit zwei Zielen:

- Die Vervollständigung und Formalisierung der Anforderungen an ein Produkt
- Die Konzeptualisierung des Unterstützungsvorgangs durch Expertensysteme in einer Gruppenentscheidung menschlicher Experten.

Aufbauend auf Konzepten der klassischen Entscheidungstheorie und dem DDM-Paradigma des Decision Supports wurde die Aufgabenformulierungs- und strukturierungsphase beschrieben, die im wahrsten Sinne des Wortes **entscheidend** für die weitere Arbeit mit dem DESIGN-ANALYZER ist. Die resultierende Designspezifikation beinhaltet sowohl optionale als auch unbedingt erforderliche Kriterien, die sich in Festpunkt-, Bereichs- und Zielkriterien einteilen lassen. Anschließend wurde die flexibel gestaltbare Domänenspezifikation des Designprozesses mit den entscheidungsbefugten lokalen Experten beschrieben. Diese Spezifikationen und die Produktmodellaten der Konstruktionsalternativen bilden die drei Teile des Datensubsystem, welche als Eingabe des Modellsystems – beschrieben durch den Entscheidungsalgorithmus – fungieren. Der Lösungsraum wird zunächst durch die Festpunkt- und Bereichskriterien eingeschränkt, um dann durch eine Nutzwertanalyse bezüglich der Zielkriterien eine vorläufige Rangordnung der verbliebenen Konstruktionsvorschläge zu erhalten. Die abschließend beschriebene Benutzeroberfläche mit interaktiver Parametereingabe implementiert das Dialogsubsystem und dokumentiert die Ergebnisse der Gruppenentscheidung.

Die Implementierung als praktische Realisierung der entwickelten Konzepte ordnet sich in weitere geplante oder bereits implementierte Tools des zentralen IMCOD-MANAGERS ein.

Anhang A

Pflichtenheft

Pflichtenheft für die Entwicklung eines Preßluftspeichers (LKW) aus Faserverbundwerkstoffen

Volumen:	80 liter
max. Durchmesser:	350 mm
Drücke:	8,5 bar im Betrieb +/- 2 bar max. Schwankdruck 17 bar max. Prüfdruck Beim Abstellen des Fahrzeugs fällt der Druck auf 0 bar ab.
Form:	Zylindrisch, längs eingebaut und mechanisch am Fahrzeugaufbau befestigt, aber demontierbar zu Reperaturzwecken.
Stückzahl:	Geschätzter Bedarf von 45000 Stück pro Jahr
Anschlüsse:	2 stirnseitig, 1 mittig für Druckverschraubungen Innengewinde Rp 3/8 Für Überdruckventil und zur Entwässerung; letztere manuell bedienbar.
max. Beschleunigung:	Durch Frontalaufprall bei 100 km/h
max. Belastung:	Seitlicher Aufprall z.B. eines Motorradfahrers
Beständigkeiten:	gegen Mineralöl, Seifenlaugen, UV-Strahlung und Lacke (Tank muß lackierbar sein)

Einsatzbedingungen

Der Behälter soll in Leichtbauweise ausgeführt werden, die Dauerfestigkeit muß gewährleistet und die Verformungen sollen gering sein. Das Material darf nur eine geringe Wasseraufnahme besitzen und muß von -40 bis +40 Grad Celsius temperaturbeständig sein. Die Einsatzorte sind Baustellen (Steinschlag) und Straßen weltweit (Klima).

Technische Randbedingungen

Der Tank darf keine scharfen Kanten haben und soll eine saubere, glatte Oberfläche aufweisen (Sichtteil). Es muß eine Bauteilabnahme durch den TÜV möglich sein, d.h. die ISO 9000 bis 9004 gesichert werden. Sie müssen zur Montage in die Fahrzeuge zum Automobilhersteller transportiert werden können. Sämtliche Anschlüsse sollen verschleißfrei sein, der Tank selbst muß wartungsfrei und austauschbar sein, das bedeutet aber nicht instandsetzbar. Nach Möglichkeit ist eine Wiederverwendung anzustreben, die Tanks müssen in jedem Fall wiederverwertbar sein (recyclierfähiges Material und recyclinggerechte Konstruktion).

Wirtschaftliche Randbedingungen

Grenzkosten:	ca. 100 DM/Stück
erwarteter Gewinn:	ca. 30 Prozent pro Stück
max. Investitionsvolumen:	5 Mio. DM
Amortisationszeit:	5 Jahre
Ausstattung:	Werkhalle, keine Fertigungsmaschinen

Wie vereinbart, soll in 12 Monaten der erste Prototyp des Tanks von Ihnen geliefert werden.

Anhang B

Designspezifikation

Designspecification with requestslots for
'Pressluftspeicher LKW aus Faserverbundwerkstoffen'

'Bestaendigkeiten': Mineraloel, Seifenlaugen, UV-Strahlungen, Lacke (fix)
category: 'Stoffgroessen/chemisch'
required: true

'Temperaturbestaendigkeit': Interval [-40 , +120] Grad Celsius
category: 'Stoffgroessen/chemisch'
required: true

'Verschleissfreiheit': Anschlusse (fix)
category: 'Festigkeit'
required: true

'Verformungsbestaendigkeit': Steinschlag, Strassen weltweit (fix)
category: 'Kinematik'
required: true

'Lackhaftung': komplett (fix)
category: 'Gebrauch'
required: false

'Volumen': Interval [67 , 70] liter
category: 'Geometrie'
required: true

'Durchmesser': 350 (fix) mm
category: 'Geometrie'
required: true

- 'Form': zylindrisch (fix)
category: 'Geometrie'
required: true
- 'Anschluesse': (2 stirnseitig Entwaesserung, 1 mittig Ueberdruckventil) (fix)
category: 'Geometrie'
required: true
- 'Transportabilitaet': LKW (fix)
category: 'Montage'
required: false
- 'Abnahme TUV': DIN ISO 9000 (fix)
category: 'Guete/Sicherheit'
required: true
- 'Pruefdruck': 17 (best) in: [0 , 17] bar
category: 'Dynamik'
defaultweight: 'very high'
required: true
- 'Recyclierfaehigkeit': 90 (best) in: [80 , 100] %
category: 'Stoffgroessen/chemisch'
defaultweight: 'high'
required: false
- 'Grenzstueckkosten': 70 (best) in: [0 , 100] DM/Stck
category: 'Wirtschaft/Hersteller'
defaultweight: 'medium'
required: true
- 'Beschleunigung': 200 (best) in: [180 , nil]
category: 'Kinematik'
defaultweight: 'high'
required: true
- 'Stossfestigkeit': 70 (best) in: [50 , 70] km/h
category: 'Stoffgroessen/ergonomisch'
defaultweight: 'high'
required: false
- 'Wandstaerke': 1.1 (best) in: [1.0 , 1.2] cm
category: 'Geometrie'
defaultweight: 'very high'

required: false

'Bedarf': 50000 (best) in: [45000 , 50000]
category: 'Betrieb'
defaultweight: 'high'
required: true

'Belastung': 4000 (best) in: [0 , 4000] N
category: 'Guete/Sicherheit'
defaultweight: 'high'
required: true

'Produktionsanzahl': 8000 (best) in: [7000 , 8000] Stck
category: 'Fertigung'
defaultweight: 'medium'
required: true

'Gewicht': 200 (best) in: [0 , 1500] g
category: 'Fertigung'
defaultweight: 'medium'
required: false

Literaturverzeichnis

- [AD886] AD-Merkbl/ätter. Technischer Bericht, Vereinigung der Technischen Überwachungs-Vereine e.V. Essen, Dezember 1986.
- [BBS96] Ansgar Bernardi, Bernd Bachmann, Anna-Maria Schoeller. The IMCOD Project: Integrating Heterogeneous Experts into a Homogeneous Design Assistant. In Detlev Ruland (Hrsg.), *CAD '96*, 1996.
- [Bri93] Johannes Brischar. Berechnung statt Entscheidung? Kritik des Einsatzes von Formalen Systemen zur Entscheidungsunterstützung, 1993.
- [CH88] K. Conrads, V. Hornung. Neue Möglichkeiten durch objektorientierte Arbeitstechniken bei der Arbeit mit CAD-Systemen. In VDI-Gesellschaft Entwicklung Konstruktion Vertrieb (Hrsg.), *VDI-Berichte 700.2, Datenverarbeitung in der Konstruktion*, 1988.
- [Dri94] Fredj Dridi. Konzeption und Implementierung einer generischen Schnittstelle zum Wissensaustausch. Diplomarbeit, November 1994.
- [GR85] Adele Goldberg, David Robson. *Smalltalk-80: The Language and its Implementation*. Addison-Wesley, 1985.
- [Hä93] Michael Hänle. *Systeme zur Unterstützung von Gruppenentscheidungen*. Verlag Josef Eul, Bergisch Gladbach Köln, 1993.
- [Ise85] H. Isermann. Interactive Group Decision Making by Conditions. In A. P. Wierzbicki M. Grauer (Hrsg.), *Interactive Decision Analysis*. Springer-Verlag, Berlin, 1985.
- [JJS87] M. Jarke, M.T. Jelassi, M.F. Shakun. MEDIATOR: Toward a Negotiation Support System. In *Evolutionary Systems Design: Policy Making Under Complexity*, S. S. 151–181. San Francisco: Holden Day, 1987.
- [Kir83] A. Kirchgäßner. Vergleich von Verfahren zur Lösung von Entscheidungsproblemen bei mehrfacher Zielsetzung. In *Europäische Hochschulzeitschriften, Reihe V, Volks- und Betriebswirtschaft*, Band Bd. 424. Peter Lang, 1983.
- [KM78] P.G.W. Keen, M. S. Scott Morton. *Decision Support Systems: An Organizational Perspective*. Reading, MA: Addison Wesley, 1978.

- [Kol85] R. Koller. *Konstruktionslehre für den Maschinenbau - Grundlagen des methodischen Konstruierens*. Springer Verlag, Berlin - Heidelberg - New York - Tokyo, 1985.
- [Kup88] H. Kupper. Produktdatenmodelle in CAD-Systemen. In VDI-Gesellschaft Entwicklung Konstruktion Vertrieb (Hrsg.), *VDI-Berichte 700.2, Datenverarbeitung in der Konstruktion*, 1988.
- [KWZ82] P. Korhonen, J. Wallenius, S. Zionts. Two Interactive Procedures for Multi-criterion Optimization with Multiple Decision Makers. Technischer Bericht, Universität of Jyvaskyla, Finland, 1982.
- [Lü71] Hermann Lübbe. *Theorie und Entscheidung: Studien zum Primat der praktischen Vernunft*. Freiburg: Rombach, 1971.
- [McL86] R. Jr McLeod. *Management Information Systems*. Chicago et. al.: Science Research Associates, 1986.
- [Par] ParcPlace Systems, 999 E. Arques Avenue, Sunnyvale, CA 94086-4593. *Visual-Works Users's Guide Release 1.0*, 1992 edition.
- [Pet93] Frank Peters. Unterstützung des Experten bei der Formalisierung von Textwissen. Document D-93-26, DFKI Kaiserslautern, Dezember 1993.
- [RBB92] M. M. Richter, Bernd Bachmann, Ansgar Bernardi. Projektantrag IMCOD. unpublished, Oktober 1992.
- [RBK91] Michael M. Richter, Ansgar Bernardi, Christoph Klauck. Akquisition und Repräsentation von technischem Wissen für Planungsaufgaben im Bereich der Fertigungstechnik. Research Report RR-91-23, DFKI Kaiserslautern, Juli 1991.
- [Ric89] Michael M. Richter. *Prinzipien der künstlichen Intelligenz*. B.G. Teubner Stuttgart, 1989.
- [Rod76] W. G. Rodenacher. *Methodisches Konstruieren*. Springer Verlag, Berlin - Heidelberg - New York - Tokyo, 1976.
- [Rot82] K. Roth. *Konstruieren mit Konstruktionskatalogen*. Springer-Verlag, Berlin Heidelberg New York, 1982.
- [SB95] Markus Steffens, Ansgar Bernardi. Integriertes Produktmodell für Behälter aus Faserverbundwerkstoffen. Document D-95-06, DFKI GmbH, 1995.
- [SC82] R.H. Sprague, E.D. Carlson. *Building Effective Decision Support Systems*. Englewood Cliffs, NJ: Prentice Hall, 1982.
- [Ste94] Markus Steffens. Wissenserhebung und Analyse um Entwicklungsprozess eines Druckbehälters aus Faserverbundwerkstoff. Technischer Bericht 2, Deutsches Forschungszentrum für Künstliche Intelligenz GmbH, Februar 1994.

- [WTC95] Mao-Jiun W. Wang, Chang Tien-Chien. Tool steel materials selection under fuzzy environment. *Fuzzy sets and systems*, 1995.
- [Wü88] D. Wüstenberg. Konstruktionslehre. Skriptum, Lehrstuhl für Konstruktion im Maschinen- und Apparatebau Kaiserslautern, 1988.



Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

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

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

Veröffentlichungen des DFKI

Die folgenden DFKI Veröffentlichungen sowie die aktuelle Liste von allen bisher erschienenen Publikationen können von der oben angegebenen Adresse oder (so sie als per ftp erhältlich angemerkt sind) per anonymous ftp von ftp.dfki.uni-kl.de (131.246.241.100) im Verzeichnis pub/Publications bezogen werden. Die Berichte werden, wenn nicht anders gekennzeichnet, kostenlos abgegeben.

DFKI Publications

The following DFKI publications or the list of all published papers so far are obtainable from the above address or (if they are marked as obtainable by ftp) by anonymous ftp from ftp.dfki.uni-kl.de (131.246.241.100) in the directory pub/Publications.

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

DFKI Research Reports

1996

RR-96-06

Claus Sengler
Case Studies of Non-Freely Generated Data Types
200 pages

RR-96-05

Stephan Busemann
Best-First Surface Realization
11 pages

RR-96-04

Christoph G. Jung, Klaus Fischer, Alastair Burt
Multi-Agent Planning
Using an *Abductive*
EVENT CALCULUS
114 pages

RR-96-03

Günter Neumann
Interleaving
Natural Language Parsing and Generation
Through Uniform Processing
51 pages

RR-96-02

E.André, J. Müller, T.Rist:
PPP-Persona: Ein objektorientierter Multimedia-Präsentationsagent
14 Seiten

RR-96-01

Claus Sengler
Induction on Non-Freely Generated Data Types
188 pages

1995

RR-95-20

Hans-Ulrich Krieger
Typed Feature Structures, Definite Equivalences,
Greatest Model Semantics, and Nonmonotonicity
27 pages

RR-95-19

Abdel Kader Diagne, Walter Kasper, Hans-Ulrich Krieger
Distributed Parsing With HPSG Grammar
20 pages

RR-95-18

Hans-Ulrich Krieger, Ulrich Schäfer
Efficient Parameterizable Type Expansion for Typed
Feature Formalisms
19 pages

RR-95-17

Hans-Ulrich Krieger
Classification and Representation of Types in TDL
17 pages

RR-95-16*Martin Müller, Tobias Van Roy*

Title not set

0 pages

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

RR-95-15*Joachim Niehren, Tobias Van Roy*

Title not set

0 pages

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

RR-95-14*Joachim Niehren*

Functional Computation as Concurrent Computation

50 pages

RR-95-13*Werner Stephan, Susanne Biundo*

Deduction-based Refinement Planning

14 pages

RR-95-12*Walter Hower, Winfried H. Graf*

Research in Constraint-Based Layout, Visualization, CAD, and Related Topics: A Bibliographical Survey

33 pages

RR-95-11*Anne Kilger, Wolfgang Finkler*

Incremental Generation for Real-Time Applications

47 pages

RR-95-10*Gert Smolka*

The Oz Programming Model

23 pages

RR-95-09*M. Buchheit, F. M. Donini, W. Nutt, A. Schaerf*

A Refined Architecture for Terminological Systems: Terminology = Schema + Views

71 pages

RR-95-08*Michael Mehl, Ralf Scheidhauer, Christian Schulte*

An Abstract Machine for Oz

23 pages

RR-95-07*Francesco M. Donini, Maurizio Lenzerini, Daniele Nardi, Werner Nutt*

The Complexity of Concept Languages

57 pages

RR-95-06*Bernd Kiefer, Thomas Fettig*

FEGRAMED

An interactive Graphics Editor for Feature Structures

37 pages

RR-95-05*Rolf Backofen, James Rogers, K. Vijay-Shanker*

A First-Order Axiomatization of the Theory of Finite Trees

35 pages

RR-95-04*M. Buchheit, H.-J. Bürckert, B. Hollunder, A. Laux, W. Nutt,**M. Wójcik*

Task Acquisition with a Description Logic Reasoner

17 pages

RR-95-03*Stephan Baumann, Michael Malburg, Hans-Guenther**Hein, Rainer Hoch,**Thomas Kieninger, Norbert Kuhn*

Document Analysis at DFKI

Part 2: Information Extraction

40 pages

RR-95-02*Majdi Ben Hadj Ali, Frank Fein, Frank Hoenes, Thorsten**Jaeger,**Achim Weigel*

Document Analysis at DFKI

Part 1: Image Analysis and Text Recognition

69 pages

RR-95-01*Klaus Fischer, Jörg P. Müller, Markus Pischel*

Cooperative Transportation Scheduling

an application Domain for DAI

31 pages

1994**RR-94-39***Hans-Ulrich Krieger*

Typed Feature Formalisms as a Common Basis for Linguistic Specification.

21 pages

RR-94-38*Hans Uszkoreit, Rolf Backofen, Stephan Busemann, Abdel**Kader Diagne,**Elizabeth A. Hinkelman, Walter Kasper, Bernd Kiefer,**Hans-Ulrich Krieger,**Klaus Netter, Günter Neumann, Stephan Oepen, Stephen**P. Spackman.*

DISCO—An HPSG-based NLP System and its Application for Appointment Scheduling.

13 pages

- RR-94-37**
Hans-Ulrich Krieger, Ulrich Schäfer
 TDL - A Type Description Language for HPSG, Part 1: Overview.
 54 pages
- RR-94-36**
Manfred Meyer
 Issues in Concurrent Knowledge Engineering. Knowledge Base and Knowledge Share Evolution.
 17 pages
- RR-94-35**
Rolf Backofen
 A Complete Axiomatization of a Theory with Feature and Arity Constraints
 49 pages
- RR-94-34**
Stephan Busemann, Stephan Oepen, Elizabeth A. Hinkelmann, Günter Neumann, Hans Uszkoreit
 COSMA - Multi-Participant NL Interaction for Appointment Scheduling
 80 pages
- RR-94-33**
Franz Baader, Armin Laux
 Terminological Logics with Modal Operators
 29 pages
- RR-94-31**
Otto Kühn, Volker Becker, Georg Lohse, Philipp Neumann
 Integrated Knowledge Utilization and Evolution for the Conservation of Corporate Know-How
 17 pages
- RR-94-23**
Gert Smolka
 The Definition of Kernel Oz
 53 pages
- RR-94-20**
Christian Schulte, Gert Smolka, Jörg Würtz
 Encapsulated Search and Constraint Programming in Oz
 21 pages
- RR-94-19**
Rainer Hoch
 Using IR Techniques for Text Classification in Document Analysis
 16 pages
- RR-94-18**
Rolf Backofen, Ralf Treinen
 How to Win a Game with Features
 18 pages
- RR-94-17**
Georg Struth
 Philosophical Logics—A Survey and a Bibliography
 58 pages
- RR-94-16**
Gert Smolka
 A Foundation for Higher-order Concurrent Constraint Programming
 26 pages
- RR-94-15**
Winfried H. Graf, Stefan Neurohr
 Using Graphical Style and Visibility Constraints for a Meaningful Layout in Visual Programming Interfaces
 20 pages
- RR-94-14**
Harold Boley, Ulrich Buhrmann, Christof Kremer
 Towards a Sharable Knowledge Base on Recyclable Plastics
 14 pages
- RR-94-13**
Jana Koehler
 Planning from Second Principles—A Logic-based Approach
 49 pages
- RR-94-12**
Hubert Comon, Ralf Treinen
 Ordering Constraints on Trees
 34 pages
- RR-94-11**
Knut Hinkelmann
 A Consequence Finding Approach for Feature Recognition in CAPP
 18 pages
- RR-94-10**
Knut Hinkelmann, Helge Hintze
 Computing Cost Estimates for Proof Strategies
 22 pages
- RR-94-08**
Otto Kühn, Björn Höfling
 Conserving Corporate Knowledge for Crankshaft Design
 17 pages
- RR-94-07**
Harold Boley
 Finite Domains and Exclusions as First-Class Citizens
 25 pages
- RR-94-06**
Dietmar Dengler
 An Adaptive Deductive Planning System
 17 pages

D-96-05

Martin Schaaf

Ein Framework zur Erstellung verteilter Anwendungen
94 pages

D-96-04

*Franz Baader, Hans-Jürgen Bürckert, Andreas Günter,
Werner Nutt (Hrsg.)*

Proceedings of the Workshop on Knowledge Representation and Configuration WRKP'96
83 pages

D-96-03

Winfried Tautges

Der DESIGN-ANALYZER - Decision Support im Designprozess
75 Seiten

D-96-01

Klaus Fischer, Darius Schier

Ein Multiagentenansatz zum Lösen von Fleet-Scheduling-Problemen
Seiten

1995

D-95-12

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

Working Notes of the KP95 Workshop:
KRDB-95 - Reasoning about Structured Objects:
Knowledge Representation Meets Databases
61 pages

D-95-11

Stephan Busemann, Iris Merget

Eine Untersuchung kommerzieller Terminverwaltungssoftware im Hinblick auf die Kopplung mit natürlich-sprachlichen Systemen
32 Seiten

D-95-10

Volker Ehresmann

Integration ressourcen-orientierter Techniken in das wissensbasierte Konfigurierungssystem TOOCON
108 Seiten

D-95-09

Antonio Krüger

PROXIMA: Ein System zur Generierung graphischer Abstraktionen
120 Seiten

D-95-08

Technical Staff

DFKI Jahresbericht 1994
63 Seiten

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

D-95-07

Ottmar Lutzy

Morphic - Plus

Ein morphologisches Analyseprogramm für die deutsche Flexionsmorphologie und Komposita-Analyse
74 pages

D-95-06

Markus Steffens, Ansgar Bernardi

Integriertes Produktmodell für Behälter aus Faserverbundwerkstoffen
48 Seiten

D-95-05

Georg Schneider

Eine Werkbank zur Erzeugung von 3D-Illustrationen
157 Seiten

D-95-04

Victoria Hall

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

D-95-03

Christoph Endres, Lars Klein, Markus Meyer

Implementierung und Erweiterung der Sprache *ALCP*
110 Seiten

D-95-02

Andreas Butz

BETTY

Ein System zur Planung und Generierung informativer Animationssequenzen
95 Seiten

D-95-01

Susanne Biundo, Wolfgang Tank (Hrsg.)

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

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

1994

D-94-15

Stephan Oepen

German Nominal Syntax in HPSG

— On Syntactic Categories and Syntagmatic Relations

—
80 pages

D-94-14

Hans-Ulrich Krieger, Ulrich Schäfer

TDL - A Type Description Language for HPSG, Part 2: User Guide.
72 pages

RR-94-05

Franz Schmalhofer, J. Stuart Aitken, Lyle E. Bourne jr.
 Beyond the Knowledge Level: Descriptions of Rational
 Behavior for Sharing and Reuse
 81 pages

RR-94-03

Gert Smolka
 A Calculus for Higher-Order Concurrent Constraint
 Programming with Deep Guards
 34 pages

RR-94-02

Elisabeth André, Thomas Rist
 Von Textgeneratoren zu Intellimedia-Präsentationssystemen
 22 Seiten

RR-94-01

Elisabeth André, Thomas Rist
 Multimedia Presentations: The Support of Passive and
 Active Viewing
 15 pages

DFKI Technical Memos**1996****TM-96-02**

Harold Boley
 Knowledge Bases in the World Wide Web:
 A Challenge for Logic Programming
 8 pages

TM-96-01

Gerd Kamp, Holger Wache
 CTL — a description Logic with expressive concrete domains
 19 pages

1995**TM-95-04**

Klaus Schmid
 Creative Problem Solving
 and
 Automated Discovery
 — An Analysis of Psychological and AI Research —
 152 pages

TM-95-03

Andreas Abecker, Harold Boley, Knut Hinkelmann, Holger Wache, Franz Schmalhofer
 An Environment for Exploring and Validating Declarative Knowledge
 11 pages

TM-95-02

Michael Sintek
 FLIP: Functional-plus-Logic Programming
 on an Integrated Platform
 106 pages

TM-95-01

Martin Buchheit, Rüdiger Klein, Werner Nutt
 Constructive Problem Solving: A Model Construction
 Approach towards Configuration
 34 pages

1994**TM-94-05**

Klaus Fischer, Jörg P. Müller, Markus Pischel
 Unifying Control in a Layered Agent Architecture
 27 pages

TM-94-04

Cornelia Fischer
 PAntUDE — An Anti-Unification Algorithm for Expressing Refined Generalizations
 22 pages

TM-94-03

Victoria Hall
 Uncertainty-Valued Horn Clauses
 31 pages

TM-94-02

Rainer Bleisinger, Berthold Kröll
 Representation of Non-Convex Time Intervals and
 Propagation of Non-Convex Relations
 11 pages

TM-94-01

Rainer Bleisinger, Klaus-Peter Gores
 Text Skimming as a Part in Paper Document Understanding
 14 pages

DFKI Documents**1996****D-96-07**

Technical Staff
 DFKI Jahresbericht 1995
 55 Seiten

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

D-94-12

Arthur Sehn, Serge Autexier (Hrsg.)

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

D-94-11

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

Working Notes of the KI'94 Workshop: KRDB'94 - Reasoning about Structured Objects: Knowledge Representation Meets Databases
65 pages

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

D-94-10

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

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

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

D-94-09

Technical Staff

DFKI Wissenschaftlich-Technischer Jahresbericht 1993
145 Seiten

D-94-08

Harald Feibel

IGLOO 1.0 - Eine grafikunterstützte Beweisentwicklungsumgebung
58 Seiten

D-94-07

Claudia Wenzel, Rainer Hoch

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

D-94-06

Ulrich Buhrmann

Erstellung einer deklarativen Wissensbasis über recyclingrelevante Materialien
117 Seiten

D-94-04

Franz Schmalhofer, Ludger van Elst

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

D-94-03

Franz Schmalhofer

Maschinelles Lernen: Eine kognitionswissenschaftliche Betrachtung
54 Seiten

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

D-94-02

Markus Steffens

Wissenserhebung und Analyse zum Entwicklungsprozeß eines Druckbehälters aus Faserverbundstoff
90 pages

D-94-01

Josua Boon (Ed.)

DFKI-Publications: The First Four Years 1990 - 1993
75 pages

Der DESIGN-ANALYZER – Decision Support im Designprozeß

Winfried Tautges

D-96-03
Document