

Universität Hamburg  
Fachbereich Informatik

# Orientierung und Navigation im Arbeitsprozess der Produktauswahl von komplexen Produkten

mit einem Beispiel aus der Gebäudeautomation

Diplomarbeit im Fach Informatik am Fachbereich Informatik der  
Universität Hamburg  
Betreut durch Prof. Dr. Horst Oberquelle und Dr. Andreas Günter

Vorgelegt von:  
Katharina Wolter  
email: [katharina.wolter@web.de](mailto:katharina.wolter@web.de)

27. September 2004



# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>7</b>
<b>2</b>	<b>Software-Ergonomie</b>	<b>11</b>
2.1	Grundlagen . . . . .	11
2.1.1	Perspektiven der Mensch-Computer-Interaktion . . . . .	11
2.1.2	Benutzer und ihre Aufgaben . . . . .	13
2.1.3	Mentale und konzeptuelle Modelle . . . . .	14
2.2	Orientierung und Navigation . . . . .	16
2.2.1	Was brauchen Benutzer zur Orientierung? . . . . .	18
2.2.2	Welche Navigationsmöglichkeiten brauchen Benutzer? . . . . .	20
2.3	Usability Engineering . . . . .	22
2.3.1	Usability Engineering Lifecycle nach Mayhew . . . . .	22
2.3.2	Usability Engineering nach Carroll und Rosson . . . . .	26
2.3.3	Contextual Design nach Beyer und Holtzblatt . . . . .	28
2.3.4	Discount Usability Engineering nach Nielsen . . . . .	29
<b>3</b>	<b>Konfigurierung und wissensbasierte Systeme</b>	<b>32</b>
3.1	Grundlagen . . . . .	32
3.2	Kontrolle und Konflikte . . . . .	38
3.3	encoway und EngCon . . . . .	40
3.4	Sichtweisen bei Expertensystemen . . . . .	42
3.5	Wissenserhebung . . . . .	45
<b>4</b>	<b>Produktauswahl bei komplexen variantenreichen Produkten</b>	<b>47</b>
4.1	Konfigurierung vs. Produktauswahl . . . . .	47
4.2	Benutzergruppen . . . . .	49
4.3	Arbeitsaufgabe Produktauswahl . . . . .	51
4.3.1	Produktauswahl durch Kunden . . . . .	52
4.3.2	Produktauswahl durch Vertriebsmitarbeiter . . . . .	53
4.3.3	Produktauswahl bei komplexen Produkten . . . . .	54
4.4	Mechanismen und Techniken zur Unterstützung der Produktauswahl	58

<b>5</b>	<b>Beispiel: SVEA</b>	<b>63</b>
5.1	Gebäudeautomation . . . . .	63
5.2	Das Unternehmen SVEA . . . . .	64
5.3	Benutzergruppen und Aufgaben im Anwendungsbeispiel SVEA . . . . .	66
5.3.1	Benutzerprofile: Planer und Vertriebsmitarbeiter . . . . .	69
5.3.2	Zusammenarbeit von Planern und Vertriebsmitarbeitern . . . . .	70
5.3.3	Aufgaben der Benutzergruppen . . . . .	72
5.3.4	Vorgehen bei der Erstellung einer GA-Lösung . . . . .	74
5.3.5	Weitere Benutzergruppen . . . . .	76
<b>6</b>	<b>Unterstützung der Produktauswahl</b>	<b>77</b>
6.1	Anforderungen . . . . .	77
6.1.1	Anforderungen — Orientierung und Navigation . . . . .	77
6.1.2	Anforderungen — Arbeitsaufgabe Produktauswahl . . . . .	79
6.1.3	Anforderungen — SVEA . . . . .	82
6.2	Überblick über die Unterstützungsansätze . . . . .	86
6.3	Formularbasierte oder direktmanipulative Produktauswahl . . . . .	89
6.4	Ansätze zur Unterstützung der Orientierung . . . . .	93
6.4.1	Vorstellung von der Arbeitsaufgabe . . . . .	93
6.4.2	Schemata . . . . .	102
6.4.3	Zurückgelegter Weg . . . . .	103
6.4.4	Fortschritt — Standort des Benutzers im Arbeitsprozess . . . . .	105
6.4.5	Typische Wege . . . . .	107
6.5	Ansätze zur Unterstützung der Navigation . . . . .	109
6.5.1	Flexibilität im Vorgehen — Fester Weg bis freie Navigation . . . . .	109
6.5.2	Abkürzungen . . . . .	114
6.5.3	Schrittgröße . . . . .	118
6.5.4	Undo-Mechanismen . . . . .	120
6.5.5	Direktes Ändern von Entscheidungen . . . . .	125
6.5.6	Wiederaufsetzpunkte . . . . .	126
<b>7</b>	<b>Der Prototyp</b>	<b>129</b>
7.1	Systemskizze . . . . .	129
7.2	Benutzungsschnittstelle . . . . .	130
7.3	Evaluation . . . . .	141
7.4	Wissensbasis . . . . .	144
7.5	Vorgehen . . . . .	145
7.5.1	Einarbeitung . . . . .	146
7.5.2	Benutzerprofile . . . . .	146
7.5.3	Task Analysis und Wissenserhebung . . . . .	147

7.5.4	Anforderungen . . . . .	149
7.5.5	Entwurf und Evaluation . . . . .	149
<b>8</b>	<b>Zusammenfassung und Diskussion der Ergebnisse</b>	<b>153</b>
8.1	Zusammenfassung . . . . .	153
8.2	Diskussion und Bewertung der Ergebnisse . . . . .	155
8.3	Kritischer Rückblick . . . . .	155
8.4	Weiterführende Fragen . . . . .	157
<b>A</b>	<b>Interviewleitfaden</b>	<b>158</b>
<b>B</b>	<b>Szenarien</b>	<b>161</b>
<b>C</b>	<b>Dokumente aus dem Anwendungsbeispiel SVEA</b>	<b>168</b>
<b>D</b>	<b>Ergebnisse aus dem Anwendungsfall SVEA</b>	<b>172</b>
<b>E</b>	<b>Mockups</b>	<b>177</b>
	<b>Literaturverzeichnis</b>	<b>180</b>



# Kapitel 1

## Einleitung

Die vorliegende Arbeit beschäftigt sich damit, wie Menschen bei der *Auswahl komplexer Produkte* unterstützt werden können. *Komplexe Produkte* sind Produkte, die aufgrund der individuellen Kundenanforderungen aus einer Reihe von Komponenten zusammengesetzt sind. Beispiele für solche komplexen Produkte sind Computer, Fahrzeuge oder auch Fertigungsanlagen. Um ein komplexes Produkt auszuwählen, muss eine Reihe von Komponenten zusammengestellt werden. Damit ein fehlerfreies komplexes Produkt entsteht, ist eine Vielzahl von Bedingungen zu beachten, da die Komponenten nicht beliebig kombinierbar sind. So ist zum Beispiel eine bestimmte Fahrzeugkarosserie nicht mit jedem beliebigen Motor zusammen einsetzbar.

In der Informatik wird das schrittweise Zusammenfügen von Komponenten einer Anwendungsdomäne zu einem komplexen Produkt als Konfigurierung bezeichnet. Mit Hilfe der Methoden der wissensbasierten Konfigurierung kann der Benutzer dabei unterstützt werden, ein fehlerfreies komplexes Produkt zusammenzustellen<sup>1</sup>. Dies allein ist jedoch nicht ausreichend. Darüber hinaus sollte der Benutzer komplexe Produkte auch möglichst effizient und zu seiner eigenen Zufriedenheit auswählen können. Eine solche benutzergerechte Gestaltung der Interaktion fällt in den Bereich der Software-Ergonomie.

Im Rahmen dieser Arbeit werden Ergebnisse aus diesen beiden Fachgebieten — der Software-Ergonomie und der wissensbasierten Konfigurierung — verbunden. Thema der Arbeit ist die Unterstützung von Benutzern bei der Produktauswahl von komplexen Produkten. Hierbei wird speziell die *Orientierung und Navigation im Arbeitsprozess* der Produktauswahl betrachtet. Im Fokus stehen dabei die Benutzergruppen der *Kunden* und der *Vertriebsmitarbeiter*.

---

<sup>1</sup>Beispiele für den Einsatz dieser Methoden geben der Volkswagen Konfigurator: (<http://www.volkswagen.de/konfigurator/>; zuletzt besucht 01.05.2003) und die IBM Server Configuration Tools: (<http://www.pc.ibm.com/us/eserver/xseries/library/configtools.html>; zuletzt besucht 01.05.2003)

### **Was bedeutet Orientierung im Arbeitsprozess?**

Aufgrund der Vielzahl voneinander abhängiger Entscheidungen kann der Arbeitende leicht den Überblick im Arbeitsprozess verlieren. Bei der Konfigurierung „von Hand“ kann dies zum Beispiel unvollständige komplexe Produkte zur Folge haben, wenn ein Teil der Entscheidungen vergessen wurde. Bei der Auswahl von komplexen Produkten mit Hilfe eines Konfigurators kann der Orientierungsverlust Frustration, Stress und Demotivation bei Benutzern zur Folge haben. Bei Vertriebsmitarbeitern kann dies eine Verminderung der Arbeitsleistung bewirken. Führen dagegen Kunden die Produktauswahl selbst durch, so gehen dem Anbieter hierdurch potentielle Kunden verloren. Ein Kunde, der schon eine Reihe von Entscheidungen getroffen hat, aber keine Vorstellung davon hat, wieviel Entscheidungen noch vor ihm liegen, wird die Produktauswahl früher oder später abbrechen. Die Unterstützung von Benutzern bei der Orientierung im Arbeitsprozess ist daher einerseits für die Qualität der Arbeit von Bedeutung und andererseits aus wirtschaftlichen Gesichtspunkten wichtig.

### **Was bedeutet Navigation im Arbeitsprozess?**

Bei der Auswahl eines komplexen Produktes sind unterschiedliche Vorgehensweisen möglich. Ein Konfigurator kann dem Benutzer alle prinzipiell möglichen Arbeitsprozesse oder auch nur eine Teilmenge davon ermöglichen. Beim Entwurf eines Konfigurators werden also die Navigationsmöglichkeiten des Benutzers im Arbeitsprozess festgeschrieben. Werden hierdurch für den Benutzer wichtige Vorgehensweisen verhindert, so kann er die Produktauswahl nicht effizient und zu seiner eigenen Zufriedenheit durchführen.

### **Ziele der Arbeit**

Ziel der Arbeit ist die Ermittlung einer Reihe von Methoden und Mechanismen, mit denen Benutzer bei der Orientierung und Navigation im Arbeitsprozess unterstützt werden können. Eine Auswahl dieser Unterstützungsansätze wird praktisch erprobt. Hierfür wird ein Prototyp zur Auswahl komplexer Produkte entworfen und evaluiert. Als Beispiel für komplexe Produkte werden hierbei die vom Unternehmen SVEA Building Control Systems GmbH & Co<sup>2</sup> angebotenen Gebäudeautomationslösungen verwendet. Mit Hilfe dieser Systeme können zum Beispiel Beleuchtung, Sonnenschutz und Raumtemperatur in Gebäuden geregelt werden. Der Prototyp sowie die im Rahmen der Arbeit erstellte Wissensbasis basieren auf dem Konfigurierungskern EngCon des Unternehmens encoway GmbH & Co KG<sup>3</sup>.

---

<sup>2</sup><http://www.svea.de>

<sup>3</sup><http://www.encoway.de>



Da es das Ziel ist, Menschen bei der Orientierung im Arbeitsprozess zu unterstützen, ist zunächst zu klären, unter welchen Bedingungen Menschen sich orientiert fühlen. Eine weitere Voraussetzung ist eine genaue Kenntnis der Arbeitsaufgabe. Auf der Basis dieser Ergebnisse können Anforderungen formuliert werden, die eine Unterstützung von Benutzern bei der Orientierung und Navigation im Arbeitsprozess erfüllen sollte. Die erarbeiteten Unterstützungsansätze setzen diese Anforderungen um. Der Prototyp basiert zusätzlich auf allgemeinen Anforderungen, die aus der Arbeitsaufgabe *Produktauswahl von komplexen Produkten* resultieren, sowie den Anforderungen, die im konkreten Anwendungsfall SVEA bestehen. Der auf der Basis der Unterstützungsansätze entworfene Prototyp wird durch einen Benutzertest evaluiert.

Wie bereits beschrieben, verbindet die Arbeit die Fachgebiete der Software-Ergonomie und der wissensbasierten Konfigurierung. Das Anwendungsbeispiel erweitert das Spektrum um die LON-basierte Gebäudeautomation. Mein Studenschwerpunkt ist die Software-Ergonomie. Daher erforderte die Arbeit, dass ich mich zunächst in das Gebiet der wissensbasierten Konfigurierung speziell in Verbindung mit EngCon einarbeitete und mich mit der LON-basierten Gebäudeautomation vertraut machte. Diese Einarbeitung war natürlich zeitaufwändig aber auch sehr interessant.

## Überblick

In *Kapitel 2* werden zuerst zentrale Grundlagen aus dem Bereich der Software-Ergonomie behandelt. Anschließend werden die Begriffe *Orientierung* und *Navigation* näher erläutert. Es werden die Voraussetzungen dafür beschrieben, dass Menschen sich orientiert fühlen. Diese Ergebnisse bilden die Grundlage für Abschnitt 6.1.1. Dort werden die Anforderungen definiert, die eine Unterstützung von Benutzern bei Orientierung und Navigation im Arbeitsprozess erfüllen sollte. Der letzte Abschnitt des zweiten Kapitels behandelt Methoden und Techniken des Usability-Engineering. Hierauf aufbauend wird in *Kapitel 7* das Vorgehen im Rahmen des Prototypentwurfs beschrieben und begründet.

*Kapitel 3* behandelt die notwendigen Grundlagen aus dem Gebiet der wissensbasierten Konfigurierung. Es werden Ursachen und Behandlungsmethoden für Konflikte beschrieben. Von einem Konflikt wird gesprochen, wenn eine Zusammenstellung von Komponenten nicht alle einzuhaltenden Bedingungen berücksichtigt. Es wird der Kontext in dem ich meine Diplomarbeit geschrieben habe, vorgestellt: Das Unternehmen encoway und der dort entwickelte Konfigurierungskern EngCon. Anschließend wird auf Sichtweisen bei Expertensystemen beziehungsweise wissensbasierten Systemen eingegangen. Zum Abschluss des Kapitels wird kurz das Thema Wissenserhebung beleuchtet.

Um Benutzer bei einer Arbeitsaufgabe unterstützen zu können, ist ein Verständnis dieser Arbeitsaufgabe notwendig. Daher wird in *Kapitel 4* auf die Arbeitsaufgabe Produktauswahl im allgemeinen und später speziell bei komplexen Produkten näher eingegangen. Als Grundlage für diese Beschreibung der Arbeitsaufgabe wird zunächst die Bedeutung des Begriffs *Produktauswahl* im Rahmen dieser Arbeit genau erläutert. Danach wird kurz auf typische Benutzergruppen bei der Auswahl komplexer Produkte eingegangen. Zum Abschluss des Kapitels werden Mechanismen und Techniken vorgestellt, die zur Unterstützung der Produktauswahl eingesetzt werden.

Das *Kapitel 5* stellt das Anwendungsbeispiel SVEA vor. Die von SVEA vertriebenen komplexen Produkte sind Gebäudeautomationslösungen. Daher wird zunächst kurz auf die Gebäudeautomation und das Unternehmen SVEA eingegangen. Der Hauptteil des Kapitels behandelt *Arbeitsaufgaben* und *Benutzergruppen* speziell im Beispiel SVEA. Der in *Kapitel 7* beschriebene Prototyp basiert unter anderem auf diesen Ergebnissen.

In *Kapitel 6* werden zunächst die Anforderungen beschrieben, die eine Unterstützung von Benutzern bei der Produktauswahl erfüllen soll. Dann werden die Ansätze zur Unterstützung von Benutzern bei der Orientierung und Navigation im Arbeitsprozess der Produktauswahl vorgestellt. Vor- und Nachteile der einzelnen Methoden und Mechanismen werden beleuchtet.

Der im Rahmen der Diplomarbeit erarbeitete Prototyp wird in *Kapitel 7* vorgestellt. Anschließend werden die Ergebnisse der Evaluation präsentiert und die Wissensbasis, die ich im Rahmen der Arbeit erstellt habe, wird kurz behandelt. Zum Abschluss des Kapitels wird das gewählte Vorgehen begründet, wobei auf die Vorgehensmodelle zurückgegriffen wird, die in *Kapitel 2* beschrieben wurden.

Die Arbeit schließt mit einer Zusammenfassung und der Diskussion der Ergebnisse in *Kapitel 8*.

# Kapitel 2

## Software-Ergonomie

Kapitel 2 geht zunächst auf zentrale Grundlagen der Software-Ergonomie ein. Anschließend werden die Begriffe *Orientierung* und *Navigation* näher erläutert, da es Ziel der Arbeit ist, Benutzer hierbei zu unterstützen. Der letzte Abschnitt des Kapitels behandelt Methoden und Techniken des Usability Engineering. Auf der Grundlage dieser Vorgehensmodelle erläutere ich in Kapitel 7 mein Vorgehen im Anwendungsfall SVEA.

### 2.1 Grundlagen

#### 2.1.1 Perspektiven der Mensch-Computer-Interaktion

Die im Folgenden vorgestellten Perspektiven werden in Abschnitt 3.4 wieder aufgegriffen. Dort wird auf Perspektiven bei der Entwicklung von wissensbasierten Systemen eingegangen.

Der Gestaltung der Mensch-Computer-Interaktion können ganz unterschiedliche Perspektiven zugrunde liegen. Jede Perspektive ist mit einem bestimmten Menschenbild, eigenen Optimierungszielen und einer individuellen Mensch-Computer-Beziehung verbunden. (Oberquelle 1991) beschreibt, dass Perspektiven oft im Unbewussten bleiben. Da sie aber den Entwicklungsprozess, die entstehenden Anwendungssysteme und die erforderliche Benutzerqualifikation beeinflussen, sollte deutlich gemacht werden, welche Perspektiven beim Entwurf eingenommen werden. Ziel dieses Abschnittes ist es, die von Oberquelle und Maaß beschriebenen Perspektiven vorzustellen (Oberquelle 1991; Maaß und Oberquelle 1992).

Bei der *Maschinen-Perspektive* steht die Maschine im Zentrum der Betrachtung. Das Optimierungsziel besteht in der möglichst umfassenden Automatisierung menschlicher Arbeit. Der Mensch wird als Maschinenbediener gesehen, dem

die nicht automatisierbaren Reste der Arbeit zufallen. Es ist nicht das Ziel, dass Benutzer die Systemabläufe nachvollziehen können oder Ausgaben hinterfragen.

Die *System-Perspektive* betrachtet Menschen und Computer bzw. Programme als vergleichbare Komponenten eines Systems. Die Aufgabe dieses Systems besteht in der Datenverarbeitung. Der Mensch wird also als eine informationsverarbeitende Komponente neben anderen betrachtet. Ziel ist es, den Datenaustausch zwischen diesen Komponenten möglichst sicher und effizient zu gestalten.

Von einer *Kommunikationsperspektive* wird ausgegangen, wenn Mensch und Computer Nachrichten in einer Dialogsprache austauschen. Es können die folgenden zwei Kommunikationsperspektiven unterschieden werden. Bei der *Dialogpartner-Perspektive* werden Mensch und Computer als gleichberechtigt gesehen. Das Entwicklungsziel besteht darin, den Computer möglichst menschenähnlich zu gestalten. Hierdurch wird eine Anthropomorphisierung<sup>1</sup> des Computers durch den Menschen gefördert. Dies soll bei der zweiten Kommunikationsperspektive – der *formalen Kommunikation* – vermieden werden. Die Kommunikationsregeln sollen als vom Entwickler entworfen erkannt werden. Das Optimierungsziel besteht in einem für den Menschen transparenten und durch ihn kontrollierten System.

In der *Werkstatt-Perspektive* (auch Werkzeug-Perspektive) wird der Mensch als qualifizierter Arbeitnehmer betrachtet. Der Computer wird als ein unterstützendes Werkzeug zur Bearbeitung von aufgabenrelevanten Materialien gesehen. Dieses Werkzeug sollte bei der Arbeit aus dem Bewusstsein des Benutzers verschwinden, damit er sich voll auf seine Arbeitsaufgabe konzentrieren kann. Außerdem hat der Benutzer die volle Kontrolle über das Werkzeug und erkennt die Auswirkungen seiner Handlungen sofort.

Die *Medienperspektive* schließlich betont die Zusammenarbeit und Kommunikation von Menschen. Hierbei wird über die Betrachtung der Mensch-Computer-Beziehung hinausgegangen. Es wird die Kommunikation, Koordination und Kooperation von mehreren Personen betrachtet, wobei der Computer als Medium dient.

Für die Gestaltung von menschengerechter Arbeit ist es notwendig, mehr als eine Perspektive einzunehmen. Außerdem sind nicht alle beschriebenen Perspektiven hierfür gleichermaßen geeignet, wie Maaß und Oberquelle herausstellen (Maaß und Oberquelle 1992). So stehen die Maschinen- sowie die System-Perspektive im Widerspruch zu zentralen Anforderungen an menschengerechte Arbeit wie zum Beispiel ganzheitliche Tätigkeiten, Anforderungsvielfalt und der Möglichkeit zu lernen.

---

<sup>1</sup>Vermenschlichung; Übertragung von menschlichen Eigenschaften und Verhaltensweisen auf den Computer

## 2.1.2 Benutzer und ihre Aufgaben

Die Gebrauchstauglichkeit eines interaktiven Systems kann nicht isoliert bewertet werden, sondern nur in Abhängigkeit von der Zielgruppe der Benutzer und ihren Aufgaben. Daher ist es, wie Nielsen (1993) betont, beim Usability Engineering zentral, die zukünftigen Benutzer zu kennen. Als Benutzer eines Systems sollten laut Nielsen hierbei alle Personen betrachtet werden, deren Arbeit durch dieses System beeinflusst wird.

„The concept of 'user' should be defined to include everybody whose work is affected by the product in some way, including the users of the system's end product or output even if they never see a single screen.“ (Nielsen 1993, S. 73)

(Shneiderman 2002) fordert als Grundlage für jeden Entwurf Benutzerprofile, die Angaben über Alter, Geschlecht, körperliche Fähigkeiten, Ausbildung, kulturellen und ethnischen Hintergrund, Vorkenntnisse, Motivation, Ziel und Persönlichkeit der Benutzer enthalten. Solche Benutzerprofile erlauben es, die allgemeinen Anforderungen einer bestimmten Benutzergruppe an eine Benutzungsschnittstelle zu ermitteln. Laut Nielsen (Nielsen 1993) sind die drei wichtigsten Kategorien, in denen sich die Erfahrungen der Benutzer unterscheiden können, die folgenden:

- Erfahrungen mit dem betrachteten Anwendungssystem
- Erfahrungen mit Computern im Allgemeinen
- Erfahrungen mit der betrachteten Aufgabendomäne

(Shneiderman 2002) unterscheidet die folgenden drei groben Benutzerklassen: Anfänger oder erstmalige Anwender, erfahrene, periodische Anwender und Experten durch regelmäßige Anwendung. Erstmalige Anwender sind hierbei in Abgrenzung zu Anfängern Benutzer, die die Aufgabenkonzepte gut kennen, nicht aber die Konzepte der Schnittstelle. Shneiderman bezieht in diese Klassifizierung der Benutzer also neben dem Wissen über das betrachteten Anwendungssystem auch das Wissen aus der Anwendungsdomäne mit ein. Aufgaben können nach Oberquelle hinsichtlich ihrer Formalisierbarkeit grob in drei Klassen gruppiert werden (Oberquelle 1994, S. 98):

**Algorithmische Aufgaben** sind vollständig automatisierbar. Beispiele sind das Berechnen von Formeln oder das Lösen von Gleichungssystemen.

**Stark-strukturierte Aufgaben** dagegen werden nach festen Regeln und mit vorgegebenen Mitteln erledigt. Der arbeitenden Person bleiben bei der Durchführung aber geringe Spielräume. Als Beispiele für stark-strukturierte Aufgaben nennt Oberquelle Bestell- und Buchungsvorgänge.

**Schwach-strukturierte Aufgaben** sind gekennzeichnet durch eine große Offenheit sowohl hinsichtlich der Mittel als auch der Wege. Typische Beispiele für solche Aufgaben sind Entwurfs-, Gestaltungs- und Planungsaufgaben.

Die hier beschriebenen Grundlagen zu Benutzern und ihren Aufgaben werden in Kapitel 4 wieder aufgegriffen, wo die Arbeitsaufgabe *Produktauswahl bei komplexen Produkten* näher erläutert wird.

### 2.1.3 Mentale und konzeptuelle Modelle

Als Grundlage für den nächsten Teilabschnitt wird im Folgenden kurz auf die Begriffe des mentalen Modells, des konzeptuellen Modells und der Metapher eingegangen.

Um erfolgreich mit einem interaktiven System arbeiten zu können, benötigt der Benutzer eine Vorstellung von der Funktionsweise und den Möglichkeiten dieses Systems. Solches Wissen speichert der Mensch in Form von untereinander verbundenen *mentalenen Modellen* (Oberquelle 1994). Laut Mayhew (1992, S. 80) sind mentale Modelle interne Repräsentationen, die das momentane Verständnis des Benutzers vom System umfassen. Sie bilden die Grundlage, auf der der Benutzer versucht, das Systemverhalten vorauszusehen und sich zu erklären (Norman 1983, S. 7). Norman (1983) unterscheidet unter anderem zwischen dem System selbst, dem *mentalenen Modell* des Benutzers vom System und *konzeptuellen Modellen*. Mentale Modelle entstehen bei der Interaktion des Benutzers mit dem System. So haben Anfänger ein sehr einfaches, unvollständiges und eventuell fehlerhaftes mentales Modell des Systems. Bei der Arbeit mit dem System wird das mentale Modell mit der Zeit genauer und vollständiger (Mayhew 1992, S. 81). Außerdem umfasst das mentale Modell des Benutzers auch Hypothesen über unsichtbare Teile und Prozesse des Systems und ihre Beziehung zu den sichtbaren Teilen und Prozessen. Laut Norman (1983) sind mentale Modelle typischerweise unvollständig und instabil. So vergessen Benutzer zum Beispiel Details, wenn sie ein System nicht kontinuierlich nutzen.

*Konzeptuelle Modelle* sind Repräsentationen des Systems. Sie sollen es dem Benutzer erleichtern, ein brauchbares und klares mentales Modell vom System zu entwickeln. Ein konzeptuelles Modell wird im Rahmen des Entwurfes einer Benutzungsschnittstelle erarbeitet und bildet die Grundlage für alle anderen Schritte. Nach Johnson und Henderson (2002) definiert ein konzeptuelles Modell die folgenden Aspekte einer Benutzungsschnittstelle:

- zentrale Metaphern und Analogien
- Begriffe, die in der Benutzungsschnittstelle eingesetzt werden

- Beziehungen zwischen diesen Begriffen
- Zuordnung zwischen diesen Begriffen und denen der Aufgabendomäne, in der das System eingesetzt wird.

Die Qualität eines konzeptuellen Modells ist abhängig von der Benutzergruppe, die durch dieses konzeptuelle Modell unterstützt werden soll. So bringen unterschiedliche Benutzer verschiedene mentale Modelle mit, auf denen sie aufbauen können. Kann ein Benutzer weitgehend auf bereits vorhandenen mentalen Modellen aufbauen oder leicht an sie anknüpfen, so erleichtert dies das Erlernen eines Systems. Um ein System erfolgreich einzusetzen, muss der Benutzer die technische Funktionsweise dieses Systems nicht im Detail kennen.

Auch Cooper (1995) unterscheidet die oben beschriebenen drei Modelle (System, konzeptuelles Modell und mentales Modell). Er verwendet jedoch die Begriffe *implementation model* (System), *mental model* und *manifest model* (konzeptuelles Modell). Abbildung 2.1 veranschaulicht, dass die Qualität eines konzeptuellen Modells steigt, wenn es dem mentalen Modell des Benutzers näherkommt und für ihn irrelevante Implementationsdetails verbirgt. Cooper hebt hervor, dass es Benutzern um so leichter fällt ein System zu verstehen und zu benutzen, je dichter das konzeptuelle Modell an seinem mentalen Modell liegt (siehe auch Abbildung 2.1).

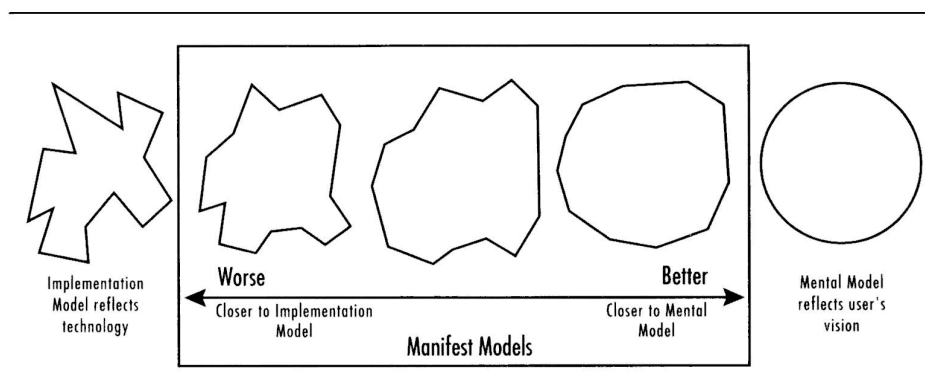


Abbildung 2.1: Drei Modelle eines Anwendungssystems nach Cooper (1995, S. 27ff)

Wie oben beschrieben, kann man Benutzern das Erlernen eines neuen Systems erleichtern, indem man es ihnen ermöglicht, bereits vorhandenes Wissen zu übertragen. Eine Möglichkeit dieses Übertragen von Wissen zu fördern und damit auch den Aufbau eines mentalen Modells, besteht im Einsatz von *Metaphern*.

Eine Metapher ist, wie Dutke (1994) beschreibt, ein Mittel, um auf eine Analogiebeziehung zwischen einem Gegenstandsbereich und einem Zielbereich hinzuweisen. Eine der bekanntesten Metaphern ist die Schreibtisch-Metapher. Sie besteht aus einer Reihe von Objekten, die der Benutzer aus seinem Arbeitsalltag kennt, wie zum Beispiel Ordern, Dokumenten und einem Papierkorb. Auf dieser Basis kann der Benutzer seine Dateien verwalten, ohne die Details der Dateiverwaltung kennen zu müssen. Es wird also von einem Teil des Systems (bzw. des implementation models) abstrahiert.

## 2.2 Orientierung und Navigation

Die Begriffe „Orientierung“ und „Navigation“ haben ihren Ursprung nicht in der Software-Ergonomie. „Orientierung“, „sich orientieren“ und „Navigation“ werden in Meyers Lexikon wie folgt definiert (Meyers Lexikonredaktion 1992):

### **orientieren** [...]

1. (sich o.) eine Richtung suchen, sich zurechtfinden;
2. informieren, unterrichten;
3. nach etwas ausrichten (einer Sache oder die eigene Meinung);
4. nach der Himmelsrichtung einstellen.

### **Orientierung** [...]

v. a. auf verschiedene Reize aus der Umwelt, aber auch auf Gedächtnisleistungen bzw. Lernvorgängen beruhende, meist zu gerichteten Bewegungen führende Reaktion bei Tieren und bei Pflanzen. [...]

### **Navigation** [...]

i.w.S. das Führen eines Wasser-, Luft- oder Raumfahrzeuges von einem Ausgangsort auf bestimmtem Wege zu einem Zielort, einschl. der dazu erforderl. Meß- und Rechenvorgänge. [...]

Die Begriffe beschreiben also ursprünglich Aspekte der Bewegung des Menschen durch die Umwelt sowie des Zurechtfindens in dieser Umwelt. Sie werden aber auch im Bereich der Software-Ergonomie häufig eingesetzt.

„The user must be able to **navigate** through the features and facilities of a complex and powerful program. He must be able to stay **oriented** [eigene Hervorhebung] in the program as he moves from screen to screen.“ (Cooper 1995, S. 507)



Hierbei wird direkt eine Analogie zwischen tatsächlichen Räumen der menschlichen Umwelt und den symbolischen „Räumen“ eines Systems aufgebaut.

„If we imagine our program as a house, we can picture each window as a separate room. The house itself is represented by the program's main window, and each room is a document window or dialog box.“ (Cooper 1995, S. 73)

„Software is not just a device with which the user interacts; it is also the **generator of a space** [eigene Hervorhebung] in which the user lives. Software design is like architecture: When an architect designs a home or an office building, a structure is being specified. More significantly, though, the patterns of life for its inhabitants are being shaped.“ (Winograd 1996, S. xvii)

Fitzgibbon und Patrick sehen eine Analogie zwischen der Orientierung in realen Räumen und den „Räumen“ eines Programms (Fitzgibbon und Patrick 1987). Benutzer finden die Navigation in einem Programm schwierig, wenn sie keine Vorstellung von dessen Struktur haben. Genauso erleichtert der Erwerb von räumlichem Wissen die zielgerichtete Bewegung im tatsächlichen Raum. Dutke nennt die *Raum-Metapher* als eine Möglichkeit, systeminterne Zusammenhänge zu visualisieren (Dutke 1994, S. 142). Hierbei werden die Darstellungsmittel der räumlichen Trennung und der Gruppierung von Zusammengehörigem eingesetzt, um Beziehungen zwischen Elementen zu verdeutlichen. Die Raum-Metapher ist laut Dutke sehr allgemein und trotzdem leicht verständlich. Da der Mensch hervorragend für die Orientierung in Räumen ausgestattet ist, kann durch den Einsatz der Raum-Metapher die Bildung mentaler Modelle unterstützt werden.

Wie Dutke (1994, S. 133) beschreibt, werden symbolische wie tatsächliche Räume durch Bewegung erfahrbar. Bewegt der Benutzer „sich selbst“ durch das System, so spricht Dutke von *Navigation*. Ziegler (1997) definiert den Begriff Navigation als „Bewegung des Benutzers durch die Sichten eines Systems“. Hierbei versteht er unter einer Sicht eine Zusammenfassung von Elementen in einem Fenster oder einer Maske. Ziegler unterscheidet zwischen Interaktion, die der Manipulation von Objekten dient, und Interaktion, die der Navigation dient. Navigationsinteraktion hat bei Zielger keine Auswirkungen auf den Zustand der Arbeitsobjekte des Benutzers.

Die obigen Beispiele betrachten die Orientierung und Navigation bezogen auf ein Programm. Im Rahmen dieser Arbeit wird dagegen die *Orientierung und Navigation im Arbeitsprozess* behandelt. Ob Menschen eine Unterstützung bei der Orientierung im Arbeitsprozess benötigen und wie diese gestaltet sein sollte, hängt von der Arbeitsaufgabe ab. Die im Rahmen dieser Arbeit betrachtete Arbeitsaufgabe *Produktauswahl bei komplexen Produkten* wird in Kapitel 4 detail-

liert analysiert. Ein Programm kann den Benutzer bei der Orientierung im Arbeitsprozess mehr oder weniger gut unterstützen. Außerdem wird beim Entwurf eines Programms festgelegt, welche Arbeitsprozesse bei der Bearbeitung einer Aufgabe möglich sind und welche nicht. Es werden also die Navigationsmöglichkeiten des Benutzers im Arbeitsprozess festgeschrieben.

Aus dem Bezug der Begriffe Orientierung und Navigation auf den Arbeitsprozess bekommt der Begriff Navigation eine zweifache Bedeutung. Die von Dutke beschriebene Navigation im System kann der Benutzer einsetzen, um eine Navigation im Arbeitsprozess zu realisieren. So kann er zum Beispiel mit Hilfe der Undo-Funktion einen zurückliegenden Bearbeitungszustand wieder herstellen. Navigation im Arbeitsprozess ist aber auch unabhängig von einem System möglich. Im Gegensatz zu Zieglers Navigationsinteraktion hat die Navigation im Arbeitsprozess jedoch immer Auswirkungen auf die Arbeitsobjekte.

### 2.2.1 Was brauchen Benutzer zur Orientierung?

Nach Nievergelt und Weydert (1979) sollte sich der Benutzer mit Hilfe der Benutzungsschnittstelle jederzeit die folgenden Fragen beantworten können:

- Wo bin ich?
- Was kann ich hier tun?
- Wie kam ich hierher?
- Wohin kann ich gehen und wie gelange ich dort hin?

Während sich die zweite Frage auf mögliche Handlungen bezieht, dient die Beantwortung der anderen Fragen der Orientierung und Navigation in der Benutzungsschnittstelle.

Da ein System von einem Benutzer zur erfolgreichen Anwendung nicht vollständig verstanden werden muss, definiert Maaß die „individuelle Durchschaubarkeit beim Gebrauch“ (Maaß 1994, S. 11). Sie ist gegeben, wenn die Benutzerin sich ein „zweckmäßiges mentales Modell vom System gemacht hat, [...] so daß sie ihre Handlungen planen und die Systemausgaben verstehen kann.“ Hierbei setzt Maaß individuelles Durchschauen mit Orientierung gleich. Orientierung erfordert nach Maaß also ein zweckmäßiges mentales Modell.

Dutke weist darauf hin, dass bei Systemen, die auf der Raum-Metapher beruhen, ähnliche Voraussetzungen für die Orientierung bestehen wie in natürlichen Räumen.

„Der Benutzer benötigt ein mentales Modell, das Wissen über Orte, Routen und Überblickswissen enthält. Erst mit der Entwicklung aller

drei Formen 'räumlichen Wissens' entsteht ein mentales Modell, das selbständiges 'Finden von Wegen', also Problemlösung ermöglicht.“ (Dutke 1994, S.143f)

Auch Dutke sieht also ein mentales Modell des Benutzers von seiner Umgebung als Voraussetzung für die Orientierung. Ist ein mentales Modell der Umgebung die einzig mögliche Grundlage für Orientierung, so bedeutet dies, dass Menschen in einer neuen Umgebung zunächst immer desorientiert sind. Der Architekt Passini befasst sich mit Orientierung und Navigation von Menschen in Gebäuden (siehe Passini (1984); Arthur und Passini (1992)). Er untersuchte die Bedingungen, unter denen Menschen das Gefühl haben, sich orientieren zu können. Dabei stellte er fest, dass Menschen sich orientiert fühlten, wenn sie

- eine adäquate mentale Repräsentation ihrer Umgebung bilden und ihren eigenen Standort in dieser Repräsentation lokalisieren konnten **oder**
- einen Plan oder eine Strategie hatten, um wichtige Ziele zu erreichen.

Passini grenzt sich von Autoren ab, die Orientierung definieren als die Fähigkeit, den eigenen Standort in einer adäquaten mentalen Repräsentation der Umgebung zu lokalisieren.

„A person who cannot establish his position in space but knows how to find his way is still oriented.“ (Passini 1984, S. 45)

Um diese Abgrenzung zu verdeutlichen, verwendet Passini den Begriff „Wayfinding“ anstelle des Begriffes „spatial orientation“, der sonst verwendet wird, um das Sich-Zurechtfinden von Menschen zu beschreiben.

Bei seinen Untersuchungen beobachtete Passini bei den Probanden zwei unterschiedliche Herangehensweisen bei der Bewältigung von Orientierungs- und Navigationsanforderungen (Passini 1984, S. 76f). Diese Gruppen unterschieden sich in der Art der Informationen, die zur Orientierung herangezogen wurden. Während die eine Gruppe sich auf Informationen der Gebäudebeschilderung stützte (zum Beispiel Richtungsschilder), verwendete die andere räumliche Informationen (wie zum Beispiele Hausbegrenzungen). Passini bezeichnet diese Herangehensweisen als linearen und räumlichen Orientierungsstil:

- Beim *linearen Orientierungsstil* basiert die Orientierung auf Wissen, wie ein Ziel erreicht werden kann. Mit Hilfe eines Leitsystems ( wie z.B. Schildern) können Orte im Gebäude erreicht werden, ohne dass eine Vorstellung vom Gebäude existiert.
- Beim *räumlichen Orientierungsstil* ist die Grundlage der Orientierung ein Verständnis der räumlichen Umgebung.

Nach Passinis Beobachtungen haben Menschen eine Präferenz für den einen oder anderen Orientierungsstil. In unbekanntem Umgebungen und Umgebungen, in denen es schwer ist, sich eine räumliche Vorstellung zu verschaffen, fallen jedoch alle Menschen auf den linearen Orientierungsstil zurück. Passini fordert immer beide Orientierungsstile zu berücksichtigen, wenn es um die Unterstützung von Menschen bei der Orientierung und Navigation in Gebäuden geht. Er geht mit seinen Forderungen also über die reine Unterstützung des Aufbaus mentaler Modelle hinaus.

Gemeinsam ist obigen Ergebnissen, dass ein zentraler Punkt bei der Orientierung eine Vorstellung von der Umgebung ist. Wer eine Vorstellung von der Umgebung hat und seine eigene Position darin kennt, kann sich auch zielgerichtet bewegen. Darüber hinaus fordert Passini zusätzlich die Unterstützung des linearen Orientierungsstils. Dieser basiert nicht auf Kenntnis der Umgebung, sondern auf einem Leitsystem. Der Benutzer kann auch ohne eine Vorstellung von der Umgebung an sein Ziel gelangen.

Meines Erachtens können Passinis Erkenntnisse zum Wayfinding auf die Orientierung im Arbeitsprozess übertragen werden. Danach findet sich eine Person in einer Arbeitsaufgabe zurecht, wenn sie **entweder** eine Vorstellung von der Arbeitsaufgabe und ihrem aktuellen Standort im Arbeitsprozess hat **oder** über einen Plan zum Erreichen ihres Arbeitszieles verfügt. Eine Vorstellung von der Arbeitsaufgabe umfasst dabei Wissen über die Teilaufgaben und Arbeitsschritte, aus denen die Arbeitsaufgabe besteht sowie Abhängigkeiten zwischen diesen.

### 2.2.2 Welche Navigationsmöglichkeiten brauchen Benutzer?

Unter *Navigation im Arbeitsprozess* wird im Rahmen dieser Arbeit die „Bewegung“ einer Person durch die Arbeitsaufgabe verstanden. Zu einem bestimmten Zeitpunkt liegen einige Arbeitsschritte noch vor ihr andere sind bereits erledigt. Die Erledigung einer Arbeitsaufgabe ist oft durch unterschiedliche Arbeitsprozesse möglich, die mehr oder weniger effizient sein können. Wie bereits oben beschrieben, wird beim Entwurf eines Programms festgelegt, welche Folgen von Handlungsschritten prinzipiell möglich sind und welche nicht. Hierdurch werden auch die Navigationsmöglichkeiten der Benutzer im Arbeitsprozess festgeschrieben. Es können zwei Interaktionsformen, die systemgeführte und die benutzergeführte Interaktion, unterschieden werden (siehe Paul (1994, S. 137ff), Koch u. a. (1991, S. 15f) und Mayhew (1992, S. 69f)).

Bei der *systemgeführten Interaktion* wird die „Auswahl und Reihenfolge der Handlungsschritte zur Erledigung einer Arbeitsaufgabe durch das interaktive System festgelegt“ (Paul 1994, S. 137). Hierfür müssen schon bei der Entwicklung eines Systems optimale Vorgehensweisen („one-best-way“) erarbeitet und festgelegt werden. Bei der *benutzergeführten Interaktion* dagegen wird die „Auswahl

und Reihenfolge der Handlungsschritte zur Erledigung einer Arbeitsaufgabe vom Benutzer frei bestimmt“ (Paul 1994, S. 142). Benutzergeführte Interaktion erweitert die Handlungs- und Entscheidungsspielräume des Benutzers, stellt aber auch höhere Anforderungen an seine Qualifikation. Abbildung 2.2 stellt die Vor- und Nachteile der Interaktionsformen einander gegenüber. Welche Interaktions-

<b>systemgeführte Interaktion</b>	<b>benutzergeführte Interaktion</b>
+ leicht erlernbar	- schwerer erlernbar
+ niedrige Ausbildungszeit	- höhere Ausbildungszeit
+ geringe Fehlerhäufigkeit	- höhere Fehlerhäufigkeit
- geringe Effizienz der Interaktion	+ hohe Effizienz der Interaktion
- geringe Flexibilität	+ hohe Flexibilität
- eingeschränkte Kontrolle durch den Benutzer	+ Kontrolle beim Benutzer

Abbildung 2.2: Bewertung von systemgeführter und benutzergeführter Interaktion nach Koch u. a. (1991, S. 15f)

form die geeignetere ist, kann nach Koch u. a. (1991) nur in Abhängigkeit vom Anwendungsfall entschieden werden. Paul hält systemgeführte Interaktion vor allem dann für sinnvoll, wenn eine Aufgabe nur selten durchgeführt wird und trotz beschränkter Sachkenntnisse Ziele erreicht werden sollen (Paul 1994, S. 142). Benutzergeführte Interaktion hält er dagegen vor allem bei Systemen für angemessen, die fester Bestandteil der täglichen Arbeit von Experten sind. Dies gilt insbesondere wenn die Benutzer unter wechselnden Bedingungen und Zielvorgaben konstruierend oder kreativ arbeiten (Paul 1994, S. 143). Nach Paul schließen sich die beiden Interaktionsformen nicht gegenseitig aus; oft sei es sogar wünschenswert, dem Benutzer die beiden Interaktionsformen als Alternativen anzubieten (Paul 1994, S. 137). Hier besteht eine Analogie zu Passinis Forderung jeweils beide Orientierungsstile zu unterstützen.

## 2.3 Usability Engineering

Usability Engineering ist eine Disziplin, die Methoden und Techniken zum Entwurf benutzbarer interaktiver Systeme bietet. In diesem Abschnitt werden vier unterschiedliche Vorgehensmodelle des Usability Engineering beschrieben: der Usability Engineering Lifecycle von Mayhew, Contextual Design von Beyer und Holtzblatt, das Scenario-based Development (SBD) Framework von Rosson und Carroll sowie das Discount Usability Engineering nach Nielsen.

### 2.3.1 Usability Engineering Lifecycle nach Mayhew

Mayhew hat die Aufgaben des Usability Engineering in einem Lifecycle vereinigt und detailliert beschrieben, wie diese Aufgaben durchgeführt werden können (Mayhew 1999). Der Lifecycle basiert stark auf dem Prinzip der Iteration (Mayhew 1999, S. 6). Mayhew untergliedert den Usability Engineering Lifecycle in die drei Phasen *Anforderungsanalyse*, *Entwurf/Testen/Entwickeln* und *Installation*. Abbildung 2.3 gibt einen grafischen Überblick über diesen Lifecycle. In diesem Überblick sind hierbei nur die häufigsten Iterationen durch Pfeile hervorgehoben. Jede der drei Phasen des Lifecycles untergliedert Mayhew jeweils in eine Reihe von Aufgaben. Für jede Aufgabe beschreibt sie unter anderem das mit der Aufgabe verbundene Ziel und die Beziehung der Aufgabe zu den anderen Aufgaben. Des Weiteren erläutert Mayhew für jede Aufgabe sehr anschaulich anhand einer ausgewählten Technik Schritt für Schritt das notwendige Vorgehen, und gibt eine grobe Einschätzung des entstehenden Aufwandes. Außerdem bietet Mayhew jeweils einen Überblick über alternative Techniken und geht dabei auch auf weniger aufwändige und genaue Techniken ein. Nielsen (1993, S. 73f) beschreibt, dass es oft schwierig ist, Zugang zu den wirklichen Benutzern zu bekommen. Mayhew liefert zu diesem Thema eine Reihe von Hinweisen, wie auch trotz dieses Problems sinnvoll vorgegangen werden kann. Aufgrund der detaillierten Beschreibungen der einzelnen Aufgaben halten Preece u. a. (2002, S. 195) den Usability Engineering Lifecycle für weniger erfahrene Personen für besonders geeignet. Im Folgenden werden die einzelnen Aufgaben des Lifecycles kurz vorgestellt. Die erste Phase des von Mayhew beschriebenen Usability Engineering Lifecycles – die Anforderungsanalyse – besteht aus den folgenden Aufgaben:

**User Profile.** Im Rahmen dieser Aufgabe werden die Merkmale der Benutzer, die entscheidend für den Entwurf der Benutzungsschnittstelle sind, beschrieben. Hierbei werden auch die Benutzergruppen ermittelt, die im Rahmen der Aufgabenanalyse (Contextual Task Analysis) betrachtet werden.

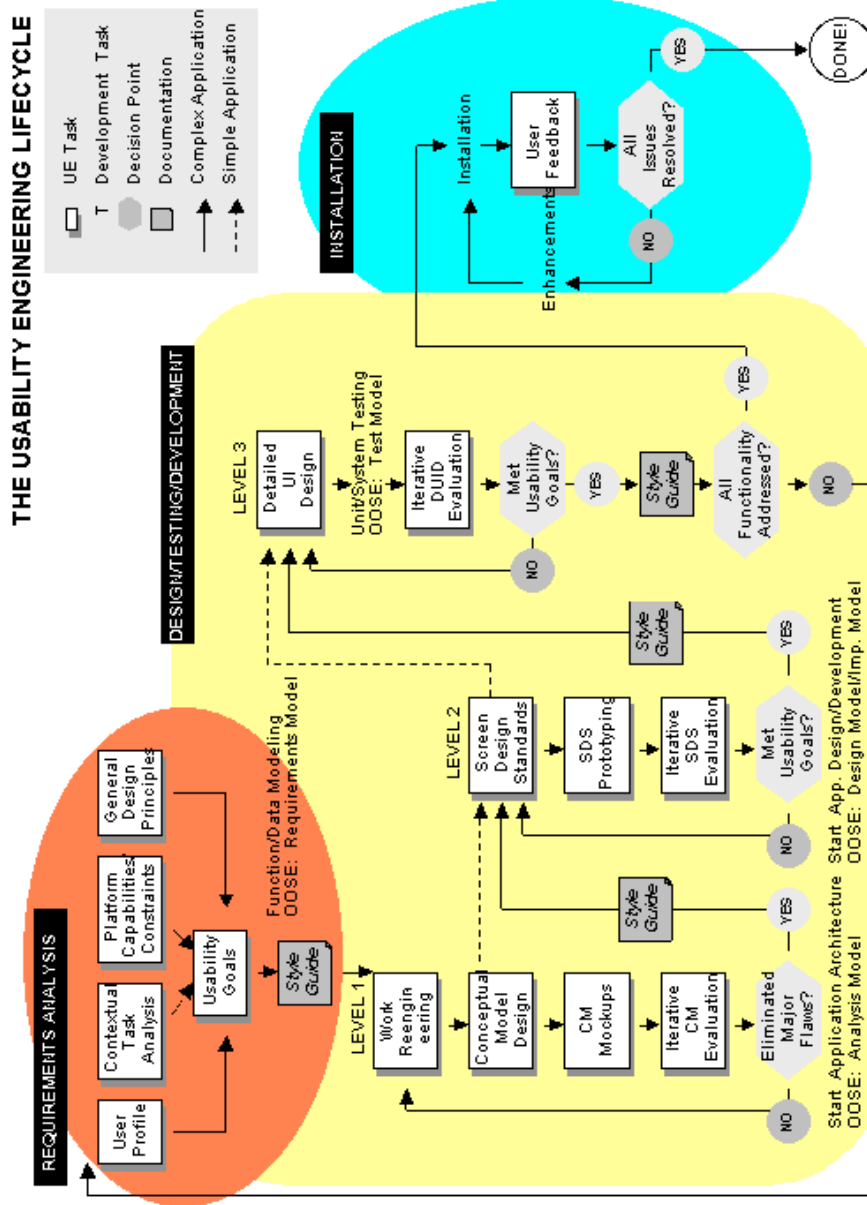


Abbildung 2.3: Usability Engineering Lifecycle nach Mayhew, Quelle: <http://drdeb.vineyard.net//index.html>; zuletzt besucht: 02.05.2003

**Contextual Task Analysis.** Ziel der Aufgabenanalyse ist es, die momentanen Aufgaben der Benutzer, die auftretenden Arbeitsabläufe sowie die der Arbeit zugrunde liegenden Ziele zu verstehen. Außerdem soll ein Verständnis der Begriffswelt der Benutzer erworben werden.

**Platform Capabilities and Constraints.** Technische Grenzen und Möglichkeiten, die das Spektrum der realisierbaren Entwürfe beeinflussen, werden im Rahmen dieser Aufgabe ermittelt.

**General Design Principles.** In dieser Aufgabe werden verfügbare Guidelines und Designprinzipien nach relevanten Punkten durchgesehen. Diese werden zusammen mit den projektspezifischen Informationen in einem Produkt-Styleguide gesammelt und bilden die Grundlage für den weiteren Entwurf.

**Usability Goal Setting.** Als Basis für die späteren Evaluationen und als Richtschnur für den Entwurf werden im Rahmen dieser Aufgabe qualitative und quantitative Usability-Ziele festlegen.

Die zentrale Phase des Lifecycles – Entwurf/Testen/Entwickeln – untergliedert Mayhew entsprechend eines top-down-Ansatzes in drei Stufen. Die oberste Stufe umfasst die folgenden vier Aufgaben:

**Work Reengineering.** Auf der Basis der Anforderungsanalyse werden die Arbeitsaufgaben der Benutzer neu organisiert und strukturiert. Hierbei werden die folgenden Ziele verfolgt: die Möglichkeiten der Automatisierung nutzen, Unternehmensziele fördern und Benutzer optimal bei ihrer Arbeit unterstützen sowie die notwendige Einarbeitung minimieren. Dies wird erreicht, indem die bisherige Arbeitspraxis nur dort verändert wird, wo die ersten beiden Ziele dies verlangen. Bei einer Veränderung der Arbeitspraxis ist jeweils zwischen diesen Zielen abzuwägen.

**Conceptual Model Design.** Im Rahmen dieser Aufgabe werden alternative konzeptuelle Modelle entworfen. Im Gegensatz zur Definition eines konzeptuellen Modells von Johnson und Henderson (2002), (siehe hierzu Seite 14), fasst Mayhew diesen Begriff wesentlich weiter. So werden im Rahmen des Conceptual Model Designs bereits Hauptdisplays und Navigationswege entworfen.

**Conceptual Model Mockups.** Die entworfenen konzeptuellen Modelle, werden in dieser Aufgabe in Form von Mockups der Benutzungsschnittstelle realisiert. Dies kann in Form eines lauffähigen Prototypen oder in Form von



*Papier-Mockups* geschehen. Papier-Mockups sind Skizzen der Benutzungsschnittstelle, wie sie zum Beispiel mit Stift und Papier oder einem Grafik-Programm erstellt werden können. Da sie mit wenig Aufwand realisiert werden, ermöglichen sie günstige und schnelle Tests. Abbildung 2.4 gibt ein Beispiel für ein Papier-Mockup.

**Iterative Conceptual Model Evaluation.** Die erstellten Mockups dienen der Evaluation des Entwurfes bzw. der Entwürfe. Mayhew sieht hierfür einen Benutzertest vor, bei dem 3 - 10 Benutzer typische Arbeitsaufgaben mit den Mockups lösen.



Abbildung 2.4: Papier-Mockup einer Benutzungsschnittstelle, Quelle: <http://www.incent.com/cd/cdp.html>; zuletzt besucht: 02.05.2003

Die vier beschriebenen Aufgaben werden in Iterationen wiederholt, bis alle grundlegenden Usability-Probleme behoben worden sind. Im Folgenden werden die vier Aufgaben der mittleren Stufe der Phase Entwurf/Testen/Entwickeln beschrieben. Das zentrale Ziel dieser Aufgaben ist das Entwickeln einer konsistenten und klaren Benutzungsschnittstelle. Die zweite Stufe untergliedert sich in die folgenden Aufgaben:

**Screen Design Standards.** In dieser Aufgabe werden produktspezifische Standards für alle Aspekte des Screendesigns festgelegt.

**Screen Design Standards Prototyping.** Auf der Grundlage des konzeptuellen Modells und der Screen Design Standards wird in dieser Aufgabe ein Teil der Funktionalität in Form eines Prototyps realisiert.

**Iterative Screen Design Standard Evaluation.** Der in der vorherigen Aufgabe erstellte Prototyp wird, zum Beispiel mit Hilfe eines Benutzertests, evaluiert. Der Prototyp und die zugrunde liegenden Standards werden so lange überarbeitet, bis die spezifizierten Usability-Ziele erreichbar scheinen, und alle großen Usability-Probleme behoben sind.

**Style Guide Development.** Das validierte konzeptuelle Modell sowie die validierten Screen Design Standards werden im Produkt-Styleguide dokumentiert. Der Produkt-Styleguide bildet die Grundlage für die vollständige Realisierung der Benutzungsschnittstelle.

In der letzten Stufe der Phase Entwurf/Testen/Entwicklung wird die Benutzungsschnittstelle schließlich vervollständigt. Diese Stufe umfasst zwei Aufgaben:

**Detailed User Interface Design.** Auf der Basis des Produkt-Styleguides wird die Benutzungsschnittstelle ganz realisiert.

**Iterative Detailed User Interface Design Evaluation.** Im Rahmen dieser Aufgabe wird die Benutzungsschnittstelle auf der Basis der Usability-Ziele evaluiert.

Die letzte Phase des Lifecycles wird von Mayhew als Installation bezeichnet und enthält nur eine Aufgabe. Diese dient der Erhebung von Feedback, nachdem ein Produkt bereits eine gewisse Zeit im Einsatz ist.

### 2.3.2 Usability Engineering nach Carroll und Rosson

Rosson und Carroll (2002) beschreiben ein weiteres Vorgehensmodell des Usability Engineering. Das von ihnen als *scenario-based development (SBD) framework* bezeichnete Vorgehensmodell beruht auf Szenarien als zentraler Technik. Szenarien definieren Rosson u.a. wie folgt:

„A scenario describes an existing or envisioned system from the perspective of one or more users and includes a narration of their goals, plans, and reactions.“ (Rosson und Carroll 2002, S. xviii)

Solche Szenarien bilden eine sehr gute Grundlage für Analyse und Diskussion, da sie von allen Projektbeteiligten gleichermaßen verstanden werden. Außerdem können sie genau wie Papier-Mockups bereits eingesetzt werden, bevor ein System realisiert wurde.

Ähnlich wie der Usability Engineering Lifecycle orientiert sich auch das SBD an den in der Software-Entwicklung üblichen Phasen. Abbildung 2.5 gibt einen Überblick über das SBD Framework. Laut Rosson und Carroll (2002, S. 28)

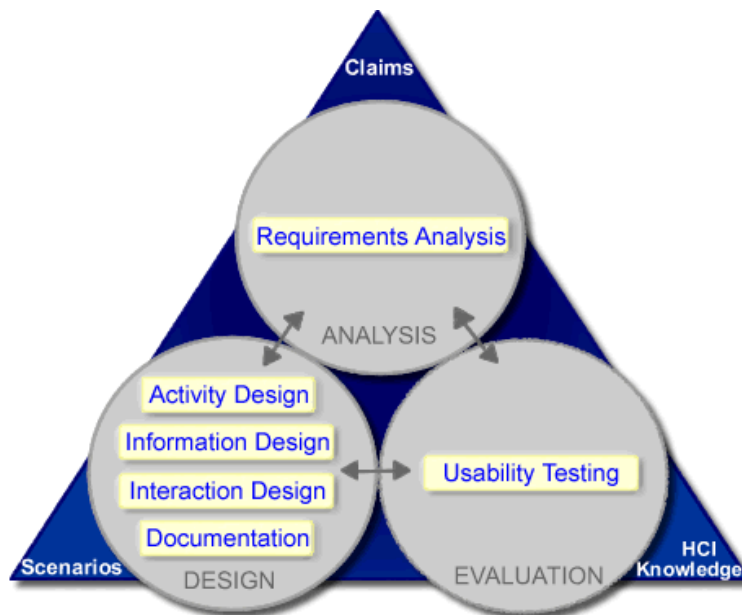


Abbildung 2.5: Der Usability Engineering Prozess nach Carroll und Rosson, Quelle: <http://ucs.cs.vt.edu>; zuletzt besucht: 02.05.2003

stimmt das Vorgehen beim SBD in weiten Teilen mit dem von Mayhew beschriebenen Lifecycle überein. Einen wesentlichen Unterschied sehen sie in der Stellung, die Szenarien im Prozess einnehmen. Mayhew setzt diese nur im Rahmen der Anforderungsanalyse ein, während Szenarien im SBD auch im Entwurf eine zentrale Rolle spielen. Einen weiteren Unterschied sehen Rosson u.a. in ihrer Betonung der im Prozess notwendigen Kompromisse. Solche abzuwägenden Alternativen werden im SBD als *Claims* und *Tradeoffs* bezeichnet. Sie werden zusammen mit einer Begründung der getroffenen Entscheidung dokumentiert. Rosson und Carroll (2002, S. 7) sehen hierin vor allem bei komplexen Systemen und umfangreichen Projekten einen entscheidenden Vorteil. Dieses systematische Bewerten von Alternativen ist in den anderen Vorgehensmodellen nicht vorgesehen. Hierin sehe ich einen erheblichen Vorteil des SBD.

Im Gegensatz zu Mayhew geben Rosson u.a. für ihr Vorgehen keine Abschätzung des Aufwandes an. Dies gilt auch für den Aufwand, der auf Seiten der zukünftigen Benutzer des Systems entsteht. Auch auf Möglichkeiten im Umgang mit dem Problem der Erreichbarkeit von Benutzern gehen Rosson u.a. nicht näher ein. Hierin sehe ich für den Einsatz in der Praxis einen erheblichen Nachteil.

### 2.3.3 Contextual Design nach Beyer und Holtzblatt

Beyer und Holtzblatt (1998) beschreiben ein weiteres Vorgehensmodell für das Usability Engineering. Im Gegensatz zum Usability Engineering Lifecycle und dem SBD orientiert sich das Contextual Design nicht direkt an den in der Software-Entwicklung verbreiteten Phasen. Abbildung 2.6 gibt einen Überblick über die Schritte des Usability Engineering Prozesses beim Contextual Design. Die

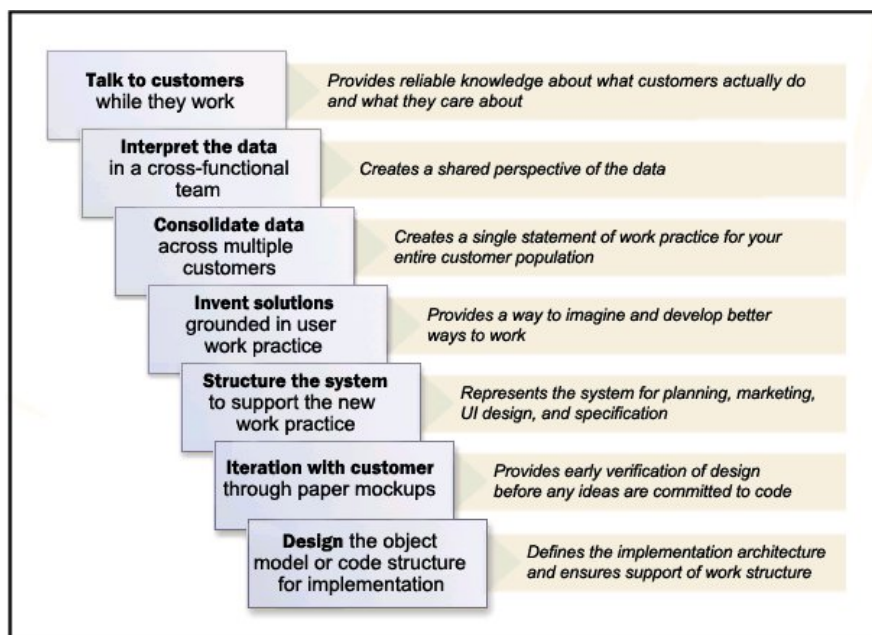


Abbildung 2.6: Der Usability Engineering Prozess nach Beyer und Holtzblatt, Quelle: <http://www.incent.com/cd/cdhow.html>; zuletzt besucht: 02.05.2003

dargestellten Schritte werden iterativ durchlaufen, was in der Abbildung nicht verdeutlicht wird. Im Rahmen des Contextual Design werden die folgenden Schritte unterschieden:

**Contextual Inquiry.** Ziel der *Contextual Inquiry* ist es, sich ein Verständnis der Benutzer und ihrer Arbeitspraxis zu verschaffen. Hierzu beobachtet ein Interviewer einen Benutzer bei der Arbeit und stellt Fragen, um Arbeitsschritte sowie zugrunde liegende Motivationen und Strategien zu verstehen.

**Interpretation Session.** Die erhobenen Daten werden in einem interdisziplinären Team interpretiert.

**Work Modeling.** Im Rahmen des *Work Modeling* werden Modelle erstellt, die die aktuelle Arbeit der einzelnen Benutzer sowie der gesamten Organisation beschreiben. Es werden fünf verschiedene Modelle verwendet, die unterschiedliche Aspekte der Arbeit fokussieren.

**Consolidation.** Ziel der *Consolidation* ist es, die aus den einzelnen Interviews entstandenen Modelle zu einem gemeinsamen Modell zu vereinigen, um die zugrunde liegende Struktur zu erarbeiten.

**Work Redesign.** Auf der Grundlage der konsolidierten Modelle werden neue Arbeitsweisen entworfen mit dem Ziel, die Arbeitspraxis der Benutzer zu verbessern. Solche Visionen werden in Form von Storyboards beschrieben.

**User Environment Design.** Im Rahmen des User Environment Designs wird eine Art Grundriss der zukünftigen Benutzungsschnittstelle erarbeitet. Dieser Grundriss untergliedert die Benutzungsschnittstelle in einzelne Bereiche. Für jeden Bereich wird beschrieben, welche Benutzeraufgaben er unterstützt, welche Funktionen hierfür vorhanden sind und wie andere Bereiche der Benutzungsschnittstelle erreicht werden können. Die im Rahmen dieses Schrittes erarbeitete Beschreibung ist dabei so abstrakt, dass sie noch keine bestimmte Benutzungsschnittstelle festlegt.

**Mockup and Test with Customers.** Mit Hilfe von Prototypen in Form von Papier-Mockups wird der Entwurf iterativ durch Benutzer getestet und verbessert.

Hierbei gehen Beyer u.a. von einem interdisziplinären Team von 8-12 Personen aus, in dem unter anderem Entwickler, Marketing, Kunden / Benutzer und Usability-Spezialisten zusammenarbeiten (Holtzblatt und Beyer 1996). In (Beyer und Holtzblatt 1999) geben Beyer und Holtzblatt einige Hinweise wie Contextual Design in unterschiedlichen Anwendungskontexten eingesetzt werden kann und unter welchen Umständen einzelne Aufgaben abgekürzt oder ausgelassen werden können.

### 2.3.4 Discount Usability Engineering nach Nielsen

Die oben beschriebenen Vorgehensweisen sind an großen Projekten orientiert in denen ein Team aus mehreren Personen arbeitet. Diese Vorgehensweisen stellen Anforderungen an eine Reihe von Ressourcen, wie beispielsweise verfügbare Zeit des Teams sowie der Benutzer, Erfahrungen der Usability-Experten oder benötigte Ausrüstung, die im benötigten Umfang nicht in allen Projekten verfügbar sind. Nielsen (1993, S. 16f) betont, dass es sinnvoller ist, weniger aufwändige

Methoden einzusetzen, auch wenn diese keine perfekten Ergebnisse erzielen, als aufgrund des nötigen Aufwandes auf Usability Engineering vollkommen zu verzichten. Auch schon mit einfacheren Methoden kann die Qualität der Design-Entscheidungen wesentlich verbessert werden. In diesem Zusammenhang stellt er das sogenannte *Discount Usability Engineering* vor (Nielsen 1993, 1990, 1989), das auf den folgenden vier Techniken beruht:

1. User and task observation
2. Scenarios
3. Simplified thinking aloud
4. Heuristic evaluation

Auch beim Discount Usability Engineering ist das frühe Einbeziehen der Benutzer eine zentrale Grundlage. Eine Möglichkeit hierfür besteht im einfachen *Beobachten der Benutzer* bei ihrer täglichen Arbeit.

Unter dem Begriff *Szenario* versteht Nielsen im Gegensatz zu anderen Autoren, wie zum Beispiel Carroll und Rosson, eine besondere Art von Prototyp. Dieser entsteht durch die Kombination von vertikalem und horizontalem Prototyp, wie Abbildung 2.7 veranschaulicht. Ein Szenario besteht also nur aus der Darstellung eines Ausschnittes der Benutzungsschnittstelle. Papier-Mockups bieten eine Möglichkeit, um ein solches Szenario zu realisieren. Diese Beschränkung

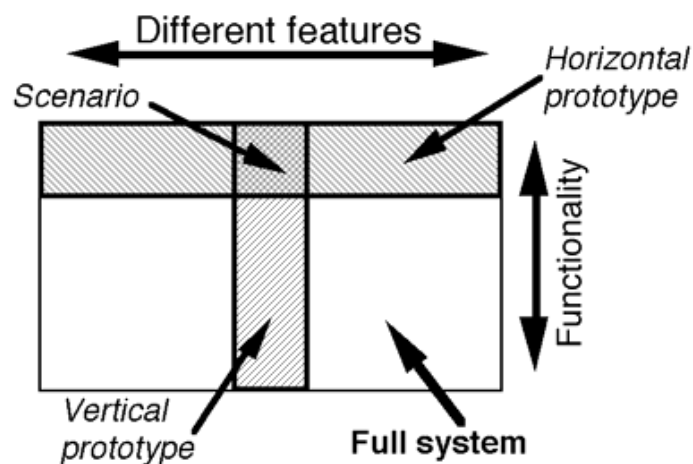


Abbildung 2.7: Horizontaler und vertikaler Prototyp sowie das Szenario bei Nielsen (1993, S. 95)

auf einen kleinen Ausschnitt der Benutzungsschnittstelle und die Verwendung von einfachen Mockups ermöglicht es schnell und häufiger Entwürfe zu testen.

Der Benutzertest des erstellten Prototyps wird beim Discount Usability Engineering mit Hilfe der Methode des *Thinking Aloud* durchgeführt. Hierbei wird jeweils ein Benutzer zur Zeit gebeten, eine bestimmte, typische Arbeitsaufgabe mit dem Prototypen zu lösen und dabei laut zu denken. Dadurch, dass Benutzer ihre Gedanken verbalisieren, wird für die Beobachter verständlich, warum Benutzer bestimmte Dinge tun und welche Elemente der Benutzungsschnittstelle Missverständnisse verursachen. Bei dieser vereinfachten Form des Thinking Aloud wird auf Videoaufnahmen verzichtet. Die Auswertung basiert stattdessen auf Notizen, die während des Test gemacht werden. Nielsen (1994a) beschreibt, dass in qualitativen Studien zum Ermitteln von Usability-Problemen auch schon eine kleine Menge von Testpersonen ausreichend sei. So würden mit drei bis fünf Benutzern schon etwa 75% der Usability-Probleme ermittelt. Spool und Schroeder (2001) beschreiben jedoch neuere Ergebnisse, nach denen mit nur fünf Benutzern teilweise ein erheblich geringerer Anteil der Usability-Problem ermittelt wurde. Sie betonen, dass die Anzahl der Benutzer nicht unabhängig von Testtechniken, der Komplexität der Aufgabe und Produktmerkmalen festgelegt werden kann. In dieser Arbeit wird eine sehr anspruchsvolle Arbeitsaufgabe betrachtet. Daher liegt die Vermutung nahe, dass eine so geringe Anzahl von Tests nicht ausreichend ist, um alle Usability-Probleme zu ermitteln.

Im Gegensatz zum Thinking Aloud ist die *Heuristic Evaluation* ein Expertentest. Bei einem solchen wird die Benutzungsschnittstelle durch einen Evaluators bewertet; typischerweise ein Usability-Experte. Nielsen (1994b) beschreibt aber, dass auch Entwickler und andere Personen eine *Heuristic Evaluation* durchführen können. Grundlage für die Bewertung der Benutzungsschnittstelle sind eine Reihe von Richtlinien. Nielsen empfiehlt, im Rahmen des Discount Usability Engineering nur einige wenige Richtlinien zur Evaluation heranzuziehen und schlägt hierfür zehn allgemeine Regeln vor (Nielsen 1993, S. 20). Da auch ein Usability-Experte normalerweise nur einen Teil der bestehenden Usability-Probleme ermittelt, sollten laut Nielsen drei bis fünf Evaluatoren unabhängig von einander die Benutzungsschnittstelle testen (Nielsen 1993, S. 156f).

Nielsen (1994b, S. 57f) beschreibt, dass Benutzertests beispielsweise durch Thinking aloud und Heuristische Evaluation eine sinnvolle Ergänzung bieten, da sie jeweils Fehler finden, die die andere Methode eher unentdeckt lässt. Ein abwechselnder Einsatz der Methoden bietet sich daher an.

Die hier beschriebenen Vorgehensweise des Usability Engineering werde ich in Abschnitt 7.5 wieder aufgreifen. Dort beschreibe ich mein Vorgehen im Anwendungsbeispiel SVEA und begründe die Auswahl der eingesetzten Methoden.

# Kapitel 3

## Konfigurierung und wissensbasierte Systeme

In der Informatik wird das schrittweise Zusammenfügen von Objekten einer Anwendungsdomäne zu einer Problemlösung beziehungsweise einem Produkt als Konfigurierung bezeichnet. Die Methoden der Konfigurierung bilden also die Basis zur Unterstützung der Produktauswahl bei komplexen Produkten. Daher werden in diesem Kapitel die notwendigen Grundlagen aus dem Bereich der Konfigurierung vorgestellt.

Auf der Basis des ersten Abschnittes wird in Abschnitt 3.2 näher auf die Steuerung der Konfigurierung — die Kontrolle — eingegangen. Außerdem werden in Abschnitt 3.2 inkonsistente Teillösungen — Konflikte — behandelt, sowie ihre Ursachen und Behandlungsmethoden. Anschließend wird der Kontext, in dem ich meine Diplomarbeit geschrieben habe, beleuchtet. Es wird das Unternehmen encoway vorgestellt und der dort entwickelte Konfigurierungskern EngCon. Sichtweisen spielen bei der Entwicklung von gebrauchstauglichen Systemen eine zentrale Rolle. Daher wird in Abschnitt 3.4 auf Sichtweisen bei wissensbasierten Systemen eingegangen. Zum Abschluss dieses Kapitels wird das Thema Wissens-erhebung kurz behandelt.

### 3.1 Grundlagen

*Komplexe Produkte* sind Produkte, die aufgrund der individuellen Kundenanforderungen aus einer Reihe von Komponenten zusammengesetzt werden müssen. Hierbei muss typischerweise eine Vielzahl von Abhängigkeiten zwischen den Komponenten beachtet werden. Beispiele für solche Produkte sind vor allem technische Systeme wie Computer, Fahrzeuge und Fertigungsanlagen, aber auch Dienstleistungen oder Softwaresysteme. Das schrittweise Zusammenfügen von



Objekten (Komponenten) einer Anwendungsdomäne zu einer Problemlösung beziehungsweise einem Produkt wird in der Informatik als *Konfigurierung* bezeichnet.

„Die Problemklasse *Planung und Konfigurierung* ist dadurch gekennzeichnet, daß eine Gesamtlösung (ein Plan, eine Konfiguration) aus Einzelkomponenten (Planschritten, Bauteilen) **konstruiert** werden soll, wobei bestimmte Randbedingungen zu berücksichtigen sind.“ (Günter 1992, S. 14)

Nach Biundo u. a. (1995) fallen sowohl Konfigurierung als auch Planung in die Klasse der Konstruktionsaufgaben. Ein Konfigurierungsproblem kann beschrieben werden durch (Günter und Kühn 1999; Günter 1995a, 1990)):

- Die Spezifikation einer *Aufgabe*, durch die die Anforderungen an die Konfiguration festgelegt werden.
- Eine Menge von *Objekten* der Anwendungsdomäne und ihre Eigenschaften. Diese werden auch als Domänenobjekte und Parameter bezeichnet.
- Eine Menge von *Relationen* und *Restriktionen* zwischen diesen Domänenobjekten.
- *Kontrollwissen* über die Vorgehensweise bei der Konfigurierung.

Der Vorgang der Konfigurierung besteht aus einer Reihe von *Konfigurierungsschritten* durch die eine initiale Teilkonfiguration in eine vollständige Konfiguration überführt werden soll. Das grundsätzliche Problem der Konfigurierung besteht darin, zu einer gegebenen Menge von Anforderungen ein aus Komponenten zusammengesetztes Lösungsobjekt zu finden, wie Abbildung 3.1 illustriert.

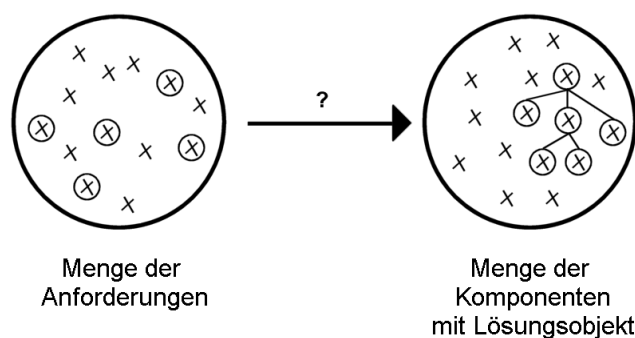


Abbildung 3.1: Struktur von Konfigurierungsproblemen nach Puppe (1991, S. 94)

Ein Beispiel für ein Konfigurierungsproblem ist die Zusammenstellung eines Computers. Anforderungen können zum Beispiel in einer hohen Speicherkapazität, einem großen Display, einer speziellen Grafikkarte, einem niedrigen Preis oder einer geringen Gerätegröße liegen. Die Domänenobjekte sind Bauteile wie Motherboard, Prozessor, RAM und Grafikkarte. Restriktionen bestehen zum Beispiel bei den Kombinationsmöglichkeiten von unterschiedlichen Prozessoren und Motherboards. Gesucht ist eine Menge von Hardware-Komponenten, die den bestehenden Anforderungen gerecht wird und dabei alle Restriktionen beachtet.

Konfigurierungsprobleme zeichnen sich meist durch einen sehr großen Lösungsraum aus (Günter 1990; Biundo u. a. 1995). Die Menge der prinzipiell möglichen Lösungen entspricht der Menge aller Kombinationen aus Komponenten und ihren Parametern. Das Ermitteln einer Lösung durch die schrittweise Weiterentwicklung einer Teilkonfiguration kann als ein Suchproblem in diesem Lösungsraum aufgefasst werden. Prinzipiell können daher Suchverfahren wie Breitensuche und Tiefensuche eingesetzt werden. Aufgrund der häufig großen Lösungsräume ist jedoch der Einsatz von reiner Breiten- oder Tiefensuche praktisch meist nicht möglich. Durch die Berücksichtigung von domänenspezifischem Wissen kann der Suchraum reduziert werden. So können im obigen Beispiel eine Reihe von potentiellen Lösungen ausgeschlossen werden, wenn berücksichtigt wird, dass ein Computer immer mindestens einen Prozessor hat. Wissensbasierte Systeme bieten die Möglichkeit, solches domänenspezifische Wissen zur Reduzierung des Suchraums einzusetzen.

Puppe (1990, S. 238) definiert ein *wissensbasiertes System* als ein „Programm zur Lösung von *Problemen*, bei dem programmtechnisch *Problemlösungsmethoden* und *Wissen* getrennt sind, was sich in Änderungsfreundlichkeit [...] und Erklärungsfähigkeit durch Angabe des zur Herleitung einer *Problemlösung* benutzten Wissens auswirkt“. Abbildung 3.2 vergleicht den Aufbau von konventionellen Programmen und wissensbasierten Systemen bzw. Expertensystemen. Ein Exper-

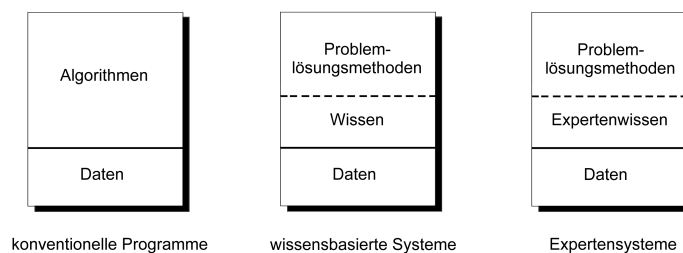


Abbildung 3.2: Aufbau von konventionellen Programmen, wissensbasierten Systemen und Expertensystemen nach Puppe (1990, S. 2)

tensystem grenzt sich nach Puppe von einem wissensbasierten System nur durch die Einschränkung auf **Expertenwissen** ab. In Abschnitt 3.4 wird näher auf die allgemeine Architektur von wissensbasierten Systemen bzw. Expertensystemen eingegangen.

### Strukturorientierte Konfigurierung

Für die Lösung von Konfigurierungsproblemen können unterschiedliche Methoden eingesetzt und kombiniert werden. Einen Überblick über Methoden zur Lösung von Konfigurierungsproblemen bieten Brinkop (1999); Günter und Kühn (1999); Tank (1993). Der Konfigurierungskern EngCon basiert auf dem Ansatz der *strukturorientierten Konfigurierung*. Da der im Rahmen dieser Diplomarbeit erstellte Prototyp auf dem Konfigurierungskern EngCon beruht, wird im Folgenden näher auf die strukturorientierte Konfigurierung eingegangen.

Die Steuerung der Suche im Lösungsraum wird als *Kontrolle* bezeichnet (Günter 1992, S. 22). In Abschnitt 3.2 wird näher auf das Thema Kontrolle eingegangen. Bei der *strukturorientierten Konfigurierung* basiert die Kontrolle auf der hierarchischen Struktur der Lösungsobjekte. Der Ansatz der strukturorientierten Konfigurierung ist weit verbreitet, und es werden unterschiedliche Varianten eingesetzt (Günter und Kühn 1999). Diese basieren auf Regeln, Bäumen, Skelettplänen oder Hierarchien. Der in Abschnitt 3.3 beschriebene Konfigurierungskern EngCon realisiert eine *begriffshierarchieorientierte Kontrolle*. Diese basiert auf einer Beschreibung der Domänenobjekte in einer *Begriffshierarchie*. Die Begriffshierarchie umfasst eine *Zerlegungshierarchie* (Partonomie) und eine *Spezialisierungshierarchie* (Taxonomie). Die Partonomie ("has-parts"-Relation) beschreibt aus welchen Elementen ein Element besteht. Die Taxonomie ("is-a"-Relation) dagegen ordnet jedem Element ein Oberkonzept und mögliche Unterkonzepte zu. Abbildung 3.3 auf Seite 36 zeigt ein Beispiel für Partonomie und Taxonomie.

In der begriffshierarchieorientierten Kontrolle sind die folgenden Konfigurierungsschritte zur Weiterentwicklung einer Teilkonfiguration möglich:

**Zerlegen.** Das Zerlegen von Komponenten entspricht einem Top-Down-Vorgehen bei der Konfigurierung. Hierbei wird eine Komponente (Aggregat) der Partonomie entsprechend in ihre Bestandteile zerlegt. Der Computer in Abbildung 3.3 kann zerlegt werden in die Komponenten Gehäuse, Tastatur, Bildschirm und so weiter.

**Integrieren.** Im Gegensatz zur Zerlegung ermöglicht die Integration ein Bottom-up-Vorgehen bei der Konfigurierung. Hierbei werden mehrere Komponenten zu einer Komponente zusammengefügt. Die Komponenten Gehäuse, Tastatur usw. in Abbildung 3.3 können zu einem Computer zusammengefügt werden.

**Spezialisieren.** Hierbei wird eine Komponente gemäß der Taxonomie durch eine Spezialisierung ersetzt. Das Ausgabemedium in Abbildung 3.3 kann zum Beispiel zu einem Drucker spezialisiert werden.

**Parametrieren.** Die obigen Konfigurierungsschritte beruhen auf Relationen zwischen den Domänenobjekten. Mit Hilfe der Parametrierung können dagegen Eigenschaften einzelner Komponenten festgelegt werden. Mögliche Parameter der Komponente Drucker sind Druckleistung und die bedruckbaren Papierformate. Die Parametrierung ermöglicht es, die Werte dieser Parameter einzuschränken oder festzulegen.

Ausgangspunkt der Konfigurierung ist eine Aufgabenstellung. Durch sie wird die initiale Teilkonfiguration definiert – ein Domänenobjekt oder mehrere. Der Prozess der Konfigurierung besteht dann aus einer Reihe der oben beschriebenen Konfigurierungsschritte. Mit Hilfe dieser Schritte soll die initiale Teilkonfiguration in eine Konfiguration überführt werden, die sowohl den Anforderungen genügt, als auch den Restriktionen gerecht wird. Kann die aktuelle Teilkonfiguration aufgrund der bisherigen Konfigurierungsschritte und den in der Wissensbasis definierten Daten weiter eingeschränkt werden, so leistet das Konfigurierungssystem dies automatisch. Soll die Lösung beispielsweise einen schwarz/weiß Tintenstrahldrucker umfassen und in der Wissensbasis ist nur ein solcher Drucker enthalten, so wird der Drucker automatisch zu diesem spezialisiert. Dieser Prozess wird als *taxonomische Inferenz* bezeichnet.

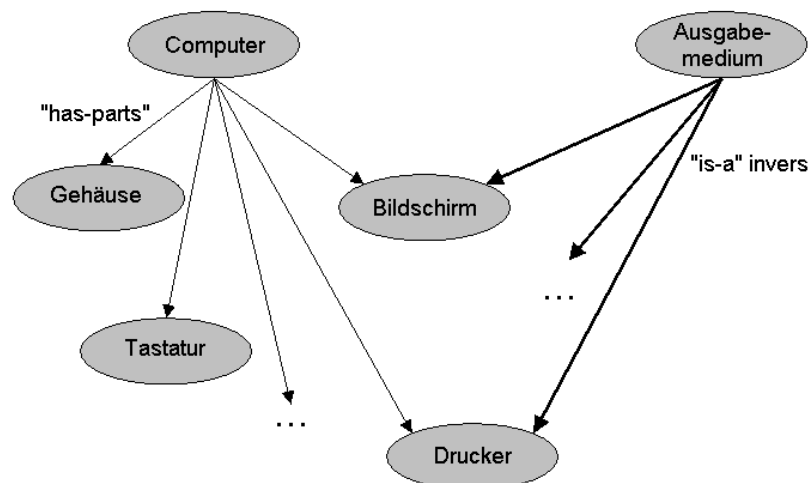


Abbildung 3.3: Beispiel für Partonomie (“has-parts“) und Taxonomie (“is-a“)

*Constraints* bieten – neben Taxonomie und Partonomie – eine weitere Möglichkeit, um Beziehungen zwischen Domänenobjekten oder ihren Parametern zu beschreiben. So kann zum Beispiel der Wertebereich eines Parameters in Abhängigkeit vom Wert eines anderen Parameters eingeschränkt werden. Die Berechnung solcher Auswirkungen wird als *Constraint-Propagation* bezeichnet. Auch die Constraint-Propagation ermöglicht eine Einschränkung des Lösungsraumes, kann jedoch aufwändige Berechnungen erfordern.

Die Weiterentwicklung einer Teilkonfiguration durch einen der obigen Konfigurierungsschritte wird als *Elaboration* bezeichnet. Die Historie des Konfigurierungsvorganges kann durch ein sogenanntes *Elaborationsnetz* repräsentiert werden. Dies hat als Ausgangspunkt die initiale Teilkonfiguration, und enthält alle bisher erzeugten (Teil-)Konfigurationen als Knoten. Die Kanten, die diese Knoten verbinden, stehen für die Konfigurierungsschritte, durch die die jeweilige Teilkonfiguration erzeugt wurde. In Abbildung 3.4 wird ein solches Elaborationsnetz zur Veranschaulichung skizziert.

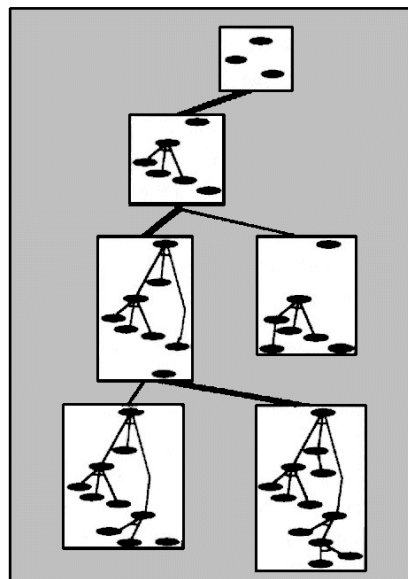


Abbildung 3.4: Schematische Darstellung eines Elaborationsnetzes nach Günter (1992, S. 105)

Aufgabe der Kontrolle ist es jeweils festzulegen, welcher der aktuell möglichen Konfigurierungsschritte durchgeführt werden soll. Hierauf wird im nächsten Abschnitt näher eingegangen.

## 3.2 Kontrolle und Konflikte

Wie oben beschrieben, wird die Steuerung der Suche im Lösungsraum als *Kontrolle* bezeichnet. Wegen der Größe des Lösungsraumes kommt ihr eine erhebliche Bedeutung zu. Durch die Kontrolle wird also die Reihenfolge bei der Abarbeitung der Konfigurierungsschritte festgelegt. Hierbei können, wie Günter (1992, S. 89) beschreibt, unterschiedliche Kontrolltypen zum Einsatz kommen:

**Strukturorientiert.** Die Kontrolle orientiert sich an der Struktur der Domänenobjekte.

**Benutzerorientiert.** Die Kontrollentscheidungen werden durch den Benutzer getroffen.

**Datenorientiert.** Die Kontrolle orientiert sich am aktuellen Zustand der Problemlösung.

**Fallorientiert.** Die Kontrolle basiert auf früheren Konfigurierungsvorgängen.

**Starr vorgehen.** Es existiert ein fest vorgegebener Konfigurierungsvorgang.

**Programmierbar.** Die Kontrollentscheidungen basieren auf domänenabhängigem Kontrollwissen, welches zuvor spezifiziert wurde.

Günter (1992) betont, dass konkrete Kontrollmechanismen meist mehrere dieser Kontrolltypen verbinden. Die oben beschriebene begriffshierarchieorientierte Kontrolle, ist primär dem strukturorientierten Kontrolltyp zuzuordnen.

Ausgehend von einer bestimmten Teilkonfiguration sind meist eine Reihe von Konfigurierungsschritten möglich. Diese werden in der *Agenda* zusammengefasst. Eine wesentliche Aufgabe der Kontrolle besteht in der Auswahl eines Konfigurierungsschrittes aus dieser Agenda. Diese Auswahl wird durch sogenannte *Agendaauswahlkriterien* festgelegt. Ein zentrales Ziel bei der Festlegung von Agendaauswahlkriterien ist es, eine möglichst erfolgsversprechende Reihenfolge für die Konfigurierungsschritte zu definieren. Werden zum Beispiel Entscheidungen, die den Lösungsraum sehr stark einschränken erst spät getroffen, so treten oft Konflikte auf.

Ein anhand der Agendaauswahlkriterien bestimmter Konfigurierungsschritt kann durch unterschiedliche *Bearbeitungsverfahren* ausgeführt werden. So kann zum Beispiel der Benutzer die entsprechende Entscheidung treffen. Andere Bearbeitungsverfahren bestehen in der Übernahme eines Default-Wertes oder Auswertung einer Berechnungsvorschrift. Auf der Grundlage der getroffenen Entscheidung wird eine neue Teilkonfiguration erzeugt. Eine solche Entscheidung kann Auswirkungen auf andere Komponenten und ihre Parameter haben. So können

zum Beispiel einzelne Spezialisierungen ausgeschlossen werden oder der Wertebereich von Parametern eingeschränkt werden. Durch die Ermittlung der Auswirkungen einer Entscheidung können Konflikte erkannt werden.

Von einem *Konflikt* spricht man, wenn die aktuelle Teilkonfiguration die bestehenden Restriktionen oder Anforderungen nicht vollständig einhält. Ein solcher Konflikt kann unterschiedliche Ursachen haben (siehe Günter (1993)). Eine mögliche Ursache besteht in *heuristischen Entscheidungen*, die während der Konfigurierung getroffen wurden. Solche auf unsicherem Wissen beruhenden Entscheidungen sind in vielen Anwendungsdomänen notwendig, um den Lösungsraum zu reduzieren. Sie können sich aber im Verlauf der Konfigurierung als falsch herausstellen. Weitere Ursachen für Konflikte sind Benutzerentscheidungen oder unlösbare Aufgabenstellungen. Außerdem kann die aktuelle Teilkonfiguration vom Benutzer als nicht geeignet bewertet werden.

Von Günter und Hotz werden drei verschiedene Vorgehensweisen zur Auflösung von Konflikten beschrieben (Günter 1993; Günter und Hotz 1995):

**Rücknahme von Konfigurierungsschritten.** Als Backtracking wird die Rücknahme von einer Entscheidung oder mehreren bezeichnet. Die Backtracking-Verfahren unterscheiden sich im Wesentlichen darin, wie die Entscheidung ermittelt wird, zu der zurückgesprungen wird. Beim *chronologischen Backtracking* wird jeweils die letzte Entscheidung zurückgenommen. Stattdessen wird bei *abhängigkeitsgesteuertem Backtracking* und bei *wissensbasiertem Backtracking* zu einer Entscheidung zurückgesprungen, die am Konflikt beteiligt ist. Diese letzten Verfahren unterscheiden sich in dem Wissen, das zur Ermittlung dieser Entscheidung herangezogen wird. Während beim abhängigkeitsbasierten Backtracking nur domänenunabhängiges Wissen herangezogen wird, arbeitet das wissensbasierte Backtracking auf domänenspezifischem Konfliktlösungswissen. Beim *interaktiven Backtracking* wird die Entscheidung, zu der zurückgesprungen wird, durch den Benutzer bestimmt. Allen bisher beschriebenen Backtracking-Verfahren ist ein Merkmal gemeinsam. Alle Entscheidungen, die nach der Entscheidung getroffen wurden, zu der zurückgesprungen wird, gehen verloren. Dies erfordert das wiederholte Ausführen von Konfigurierungsschritten. *Backtracking mit Datenübernahme* verhindert diese Ineffizienz, indem alle Entscheidungen, die nicht am Konflikt beteiligt sind, in die neue Teilkonfiguration übernommen werden.

**Reparatur der Teilkonfiguration.** Bei der Reparatur wird die inkonsistente Teilkonfiguration so modifiziert, dass eine konsistente Teilkonfiguration entsteht. Hierbei werden keine Konfigurierungsschritte zurückgenommen. Es kann zwischen der automatischen Reparatur einer Teilkonfiguration durch

das System und der interaktiven Reparatur durch den Benutzer unterschieden werden.

**Akzeptieren eines Konfliktes.** Das Akzeptieren eines Konfliktes kann aus unterschiedlichen Gründen sinnvoll sein. So kann der Benutzer einen bestehenden Konflikt akzeptieren, wenn er eine Lösung des aktuellen Konfigurationsproblems für unmöglich hält. Ist das vom System verwendete Domänenwissen nicht vollständig und korrekt, so kann ein Konflikt erkannt werden, obwohl die Teilkonfiguration tatsächlich nicht inkonsistent ist. Durch das Akzeptieren eines Konfliktes kann der Benutzer in solchen Situationen den Konfigurator „überstimmen“. Außerdem können neben den bisher genannten Restriktionen auch solche verwendet werden, die nur weiche Rahmenbedingungen beschreiben. Konflikte, die solche weichen Restriktionen verletzen, können dann toleriert werden.

Konfliktbehandlungsmethoden sollten nicht nur eine konsistente Teilkonfiguration zurückliefern. Bei der Konfliktauflösung sollte ein für den Benutzer intuitives und sinnvolles Ergebnis entstehen (TZI 2001, S. 48).

### 3.3 encoway und EngCon

Wie bereits beschrieben, habe ich diese Arbeit, auf der Grundlage des Konfigurierungskerns EngCon des Unternehmens encoway GmbH & Co KG<sup>1</sup>, erstellt. Im Folgenden wird dieser Konfigurierungskern kurz vorgestellt.

Der Konfigurierungskern EngCon ist auf der Grundlage der Erfahrungen mit dem Konfigurierungssystem KONWERK sowie dem Expertensystemkern für Planungs- und Konfigurierungsaufgaben PLAKON entwickelt worden. Diese beiden Systeme wurden im Rahmen von BMFT-Projekten im universitären Umfeld entwickelt. Detaillierte Beschreibungen dieser Systeme bieten Günter (1995b) sowie Cunis u. a. (1991). Der Konfigurierungskern EngCon ist im Gegensatz zu diesen Systemen in Java implementiert (Hollmann u. a. 2000). Er realisiert, genau wie die oben genannten Systeme, den strukturorientierten Konfigurierungsansatz. Zur Beschreibung der Domänenobjekte stehen neben Parametern und den Relationen „is-a“ (Taxonomie) und „has-parts“ (Partonomie) auch benutzerdefinierte Relationen zur Verfügung. Weitere Abhängigkeiten zwischen den Domänenobjekten und ihren Parametern werden in Form von Constraints beschrieben. Die Kontrolle in EngCon ist agendabasiert. Eine Konfigurierungsentscheidung kann hierbei die folgenden Ursachen haben (Hollmann u. a. 2000):

---

<sup>1</sup><http://www.encoway.de>



- Benutzerinteraktion,
- Übernahme eines Defaults,
- Berechnung eines dynamischen Defaults (wie zum Beispiel Minimum oder Maximum),
- Berechnungsfunktionen,
- Constraints,
- Taxonomische Inferenzen.

Ein auf EngCon basierender Konfigurator umfasst typischerweise die folgenden drei Komponenten: eine Benutzungsschnittstelle, den Konfigurierungskern EngCon und eine Wissensbasis (siehe auch Abbildung 3.5). Die Wissensrepräsentation ist XML-basiert und das in der Wissensbasis spezifizierte Wissen lässt sich in drei Kategorien untergliedern: Wissen über die Domänenobjekte, Constraint-Wissen und Kontrollwissen (Hollmann u. a. 2000; Ranze u. a. 2002). Mit Hilfe des in der Wissensbasis spezifizierten Kontrollwissens können eine systemgeführte Interaktion, eine benutzergeführte Interaktion oder beliebige Abstufungen zwischen diesen bei der Konfigurierung realisiert werden. EngCon stellt also eine sehr flexible und daher gut geeignete Grundlage für die in Kapitel 6 erläuterten Unterstützungsansätze dar. Günter (1992, S. 172ff) beschreibt detailliert die unterschiedlichen Interaktionsmöglichkeiten, die mit KONWERK realisiert werden können.

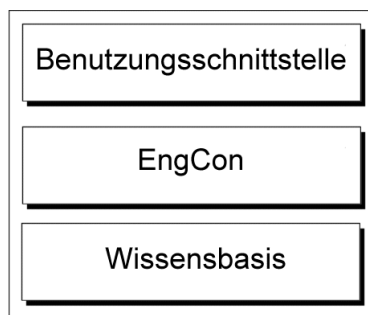


Abbildung 3.5: Allgemeine Architektur von EngCon-Konfiguratoren

Der von Ranze u. a. (2002) beschriebene *Drive Solution Designer* ist ein Beispiel für einen auf EngCon basierenden Konfigurator. Der Drive Solution Designer wird von den Vertriebsmitarbeitern der Lenze AG zur Auslegung von Antriebssystemen eingesetzt.

### 3.4 Sichtweisen bei Expertensystemen

Dem Einsatz wissensbasierter Systeme können unterschiedliche Sichtweisen zugrunde liegen. Aufbauend auf den in Abschnitt 2.1.1 beschriebenen Perspektiven wird im Folgenden auf verbreitete Sichtweisen bei Expertensystemen bzw. wissensbasierten Systemen eingegangen. Als Grundlage hierfür wird zunächst kurz auf die allgemeine Architektur von Expertensystemen eingegangen.

Die Trennung von Problemlösungsmethoden und dem zur Lösung herangezogenen Wissen ist ein zentrales Merkmal von wissensbasierten Systemen. Der Teil eines wissensbasierten Systems, in dem dieses Wissen abgelegt ist, wird als *Wissensbasis* bezeichnet. Dem gegenüber steht das Steuersystem, welches die Benutzungsschnittstelle und die Problemlösungskomponente umfasst. Abbildung 3.6 zeigt die allgemeine Architektur von Expertensystemen.

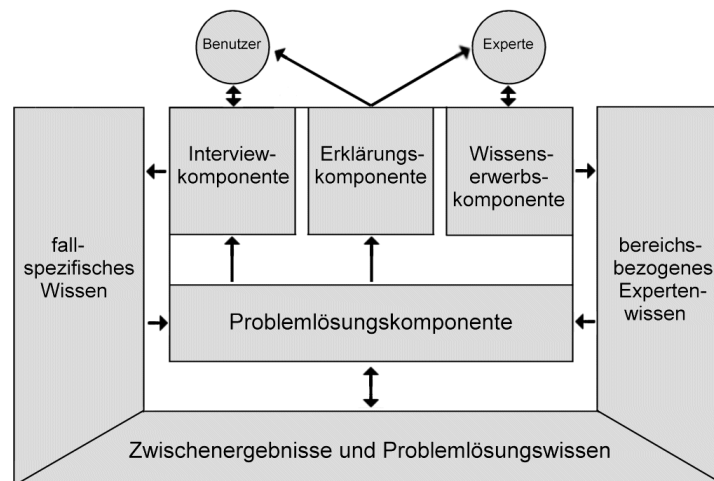


Abbildung 3.6: Allgemeine Architektur von Expertensystemen nach Puppe (1991, S. 13)

Die Benutzungsschnittstelle untergliedert sich hierbei, laut Puppe (1991, S. 12), in drei relativ eigenständige Teile: eine Interviewkomponente, eine Erklärungskomponente und eine Wissenserwerbskomponente. Die Aufgaben dieser Komponenten können wie folgt beschrieben werden:

**Interviewkomponente.** „Die Interviewkomponente führt den Dialog mit dem Benutzer und / oder liest automatisch erhobene Meßdaten ein“ (Puppe 1991, S. 12). Sie dient laut Meyer-Fujara u. a. (1995) dazu „um fallspezifisches Wissen vom Endbenutzer zu erfragen, das in die dynamische Wissensbasis eingetragen wird.“

**Erklärungskomponente.** „Die Erklärungskomponente macht die Vorgehensweise des Expertensystems transparent.“ (Puppe 1991, S. 12). Hierfür stellt sie „für den Endbenutzer das Lösungsergebnis und den Lösungsweg dar und begründet evtl. die gefundene Lösung.“ (Meyer-Fujara u. a. 1995).

**Wissenserwerbskomponente.** „Die Wissenserwerbskomponente dient dem Experten bzw. dem Wissensingenieur dazu, das Expertenwissen in die [...] Wissensbasis einzubringen.“ (Meyer-Fujara u. a. 1995).

Das Leitbild früher Expertensysteme bestand in der möglichst umfassenden Ersetzung des Experten durch ein Expertensystem. Rahmer beschreibt die Perspektive, die der Entwicklung *klassischer Expertensysteme* zugrunde liegt, wie folgt:

„Die klassische Vorstellung von einem Expertensystem ist es, daß das System vom Benutzer eine Aufgabe erhält und diese vollständig selbst bearbeitet. Der Mensch wird allenfalls zur Behebung von Konflikten in den Lösungsprozeß eingeschaltet und soll so dem System durch das Bereitstellen von zusätzlicher Information assistieren.“ (Rahmer 1999, S.1)

Typischerweise spezifiziert der Benutzer bei einem klassischen Expertensystem zunächst vollständig die Aufgabe. Die hierfür notwendige Interaktion ist systemgeführt. In einer zweiten Phase berechnet das System eine entsprechende Lösung. Abschließend kann sich der Benutzer vom System die von diesem berechnete Lösung erklären lassen. Der Entwicklung eines solchen Systems liegen die Maschinen-Perspektive und die System-Perspektive sowie die Dialogpartner-Perspektive zugrunde (zur Erläuterung der Perspektiven siehe Abschnitt 2.1.1, Seite 11).

Der Benutzer dient der Interviewkomponente als eine Datenquelle neben anderen. Er gibt Daten ein, die das System nicht anders ermitteln kann. Die Bezeichnung Interviewkomponente legt eine Befragung des Benutzers durch das System nahe. Ein solches Vorgehen wird von Cooper (1995, S. 144) als „interrogation mode“ bezeichnet. Statt Antworten vom Benutzer zu verlangen, sollte ein Programm laut Cooper Auswahlmöglichkeiten anbieten, und den Benutzer durch Rückmeldungen unterstützen.

Dem Einsatz einer Erklärungskomponente, die dem Benutzer die Ergebnisse erläutert, liegt die Dialogpartner-Perspektive zugrunde. Hierdurch wird, wie in Abschnitt 2.1.1 beschrieben, eine Anthropomorphisierung des Systems durch den Menschen gefördert. Dies ist insbesondere deshalb problematisch, weil hierdurch die Tatsache verdeckt wird, dass nur der Mensch Verantwortung für die Richtigkeit der erstellten Lösung übernehmen kann. Busch u. a. (1994, S. 45ff) fordern

daher statt einer klassischen Erklärungskomponente Explorationswerkzeuge anzubieten, durch deren Einsatz sich der Benutzer selbständig Ergebnisse erklären kann. Die oben beschriebene Erklärungskomponente sieht Begründungen nur für Lösungen, nicht aber für Zwischenergebnisse vor, was im Widerspruch steht zur software-ergonomischen Forderung, dem Benutzer kontinuierlich Feedback zu den Auswirkungen seiner Handlungen zu geben (siehe Nielsen (1993, S. 134ff), Shneiderman (2002, S. 100), Normenausschuss Ergonomie im DIN (1998, S. 6f)).

Expertensysteme, die aus der Maschinen-Perspektive bzw. System-Perspektive gestaltet sind, können nach Busch u.a. eine Dequalifizierung der Benutzer zur Folge haben, da diesen die Möglichkeit zum Sammeln von Erfahrungen genommen wird (Busch u. a. 1994, S.6). Außerdem könne der oftmals geringe Handlungsspielraum Stress auslösen. Bei einer Gestaltung, die den Stärken und Bedürfnissen der Benutzer Rechnung trägt, könne dagegen eine Qualifizierung und Leistungsverdichtung erreicht werden.

Rahmer (1999) grenzt *interaktive Konfigurierungssysteme* und *Assistenzsysteme* von klassischen Expertensystemen ab. Systeme zum interaktiven Konfigurieren charakterisiert Rahmer wie folgt:

„Beim interaktiven Konfigurieren muß der Benutzer seine Anforderungen nicht mehr (unter Umständen in einer vorgegebenen Reihenfolge) vollständig spezifizieren, sondern das Spezifizieren sowie das Lösen des Konfigurationsproblems können sich überlappen, und insbesondere wird dem Benutzer ein größerer Freiraum bei der Konfiguration eingeräumt. Der Konfigurator übernimmt dabei die Konsistenzprüfung der aktuellen Konfiguration, die Abhängigkeitsverwaltung und das Anzeigen der verbleibenden Möglichkeiten. Wesentlich für das interaktive Konfigurieren ist, daß es dem Benutzer eine nicht-prozedurale Vorgehensweise erlaubt, das heißt, er darf seine Konfiguration mit einer beliebigen Entscheidung beginnen und zu jedem Zeitpunkt des Konfigurationsprozesses eine beliebige Entscheidung treffen sowie eine früher getroffene Entscheidung modifizieren oder revidieren.“ (Rahmer 1999, S. 3)

Assistenzsysteme ermöglichen laut Rahmer darüber hinaus ein exploratives Vorgehen, welches ein Generieren, Evaluieren und Integrieren von Alternativen unterstützt. Zusätzlich bieten sie dem Benutzer Entlastungsmöglichkeiten bei sich wiederholenden Tätigkeiten. Hierbei arbeitet Rahmer mit der Metapher des Assistenten, als einer untergeordneten Person, an die Aufgaben delegiert werden können.

Auch Richter (1996, S. 9) stellt den klassischen Expertensystemen sogenannte *Assistenzsysteme* gegenüber, bei denen eine explizite Verteilung der Aufgaben zwischen Mensch und Maschine erfolgen soll. Hierbei werden die Stärken des

Menschen in Intuition, Erfahrung und Hintergrundwissen gesehen. Dem System kommt dagegen die Aufgabe zu, die Korrektheit von Lösungen sicherzustellen und Informationen bereitzustellen. Richter betrachtet hierbei mehrere Akteure, wobei ein Akteur ein Mensch oder ein System sein kann. Das Assistenzsystem dient diesen Akteuren als Mittel der Kooperation und Koordinierung. Dies entspricht der in Abschnitt 2.1.1 beschriebenen Medien-Perspektive.

Auf der Grundlage des Konfigurierungskerns EngCon ist sowohl ein klassisches Expertensystem als auch ein interaktives Konfigurierungssystem bzw. ein Assistenzsystem im obigen Sinne realisierbar (Hollmann u. a. 2000). Die hier erörterten Sichtweisen bei Expertensystemen sowie die in Abschnitt 2.1.1 beschriebenen Perspektiven werden in Abschnitt 6.1.3 wieder aufgegriffen. Dort werden die Perspektiven beschrieben, die ich beim Prototypentwurf im Rahmen der Diplomarbeit eingenommen habe.

### 3.5 Wissenserhebung

Die Erstellung einer Wissensbasis ist Gegenstand der *Wissensaquisition*. Auf das Thema Wissensacquisition soll hier nur kurz eingegangen werden, da die Erstellung der Wissensbasis nicht den Schwerpunkt meiner Arbeit darstellt. Mit Hilfe von Methoden zur *Wissenserhebung* werden zunächst Daten erhoben, die dann analysiert, interpretiert und formalisiert werden (Meyer-Fujara u. a. 1995). Einen ausführlichen Überblick über Methoden zur Wissenserhebung gibt Burge (1998).

Sollen durch ein wissensbasiertes System Menschen bei ihrer Arbeit unterstützt werden, so sind nicht nur in der Wissensbasis abzulegende Fakten oder Regeln von Bedeutung. Auch Ziele, Motive, Arbeitsverfahren und Vorgehensweisen spielen dann eine zentrale Rolle. Hacker hält daher den Begriff *Wissenserhebung* für weniger geeignet und spricht stattdessen vom *Ermitteln von Leistungsvoraussetzungen* (Hacker 1992, S. 12). Hacker betont, dass Menschen einen ausschlaggebenden Teil der Leistungsvoraussetzungen unbewusst nutzen. „Das betrifft hauptsächlich Vorgehensweisen wie beispielsweise Algorithmen und Heuristiken“ (Hacker 1992, S. 36). Diese unbewusst genutzten Leistungsvoraussetzungen seien zum Teil sogar bereits unbewusst erworben worden. Solche nicht aussagbaren Leistungsvoraussetzungen stellen vor allem bei geistigen Tätigkeiten ein Problem dar, da sie sich zusätzlich der Beobachtung entziehen (Hacker 1992, S. 11). Daher kann auf verbalisierende Verfahren bei der Ermittlung von Leistungsvoraussetzungen nicht verzichtet werden. Hacker (1992) beschreibt eine Methodenbatterie zur Ermittlung von Leistungsvoraussetzungen.

Im Rahmen des Anwendungsbeispiels SVEA habe ich selbst die Erfahrung gemacht, dass die befragten Vertriebsmitarbeiter Regeln oder Beziehungen zwar auf Nachfrage bestätigten und als wichtig bezeichneten, diese aber nicht von sich

aus formulieren konnten. Mein Vorgehen im Rahmen des Prototypenwurfes wird in Abschnitt 7.5 detailliert beschrieben. Dort wird das Thema Wissenserhebung bzw. Leistungsvoraussetzungen wieder aufgegriffen.

Die in den letzten beiden Kapiteln beschriebenen Grundlagen aus den Bereichen Software-Ergonomie und wissensbasierte Konfigurierung werden in den folgenden Kapiteln wieder aufgegriffen und zusammengeführt. In Kapitel 4 wird die Arbeitsaufgabe *Produktauswahl bei komplexen Produkten* allgemein beschrieben. Kapitel 5 geht dann auf das im Rahmen der Arbeit betrachtete Anwendungsbeispiel ein.

# Kapitel 4

## Produktauswahl bei komplexen variantenreichen Produkten

Die Kenntnis der zu unterstützenden Arbeitsaufgabe ist eine zentrale Grundlage beim Usability Engineering. Daher ist es das Ziel dieses Kapitels, die Arbeitsaufgabe *Produktauswahl bei komplexen Produkten* näher zu analysieren. Dies erfolgt in Abschnitt 4.3 während die Abschnitte 4.1 und 4.2 die hierfür notwendigen Voraussetzungen schaffen. In Abschnitt 4.1 wird das Verständnis des Begriffs *Produktauswahl* in dieser Arbeit näher erläutert. Anschließend wird im Abschnitt 4.2 kurz auf *typische Benutzergruppen* bei der Auswahl komplexer Produkte eingegangen. Eine wesentliche Quelle für die Ergebnisse dieses Kapitels sind neben der zitierten Literatur auch meine Gespräche mit Experten aus dem Gebiet der Konfigurierung und mit den Mitarbeitern von SVEA.

Zur Unterstützung von Benutzern bei der Auswahl von Produkten im WWW werden unterschiedliche *Mechanismen* und *Techniken* eingesetzt. Den Abschluss dieses Kapitels bildet ein kurzer Überblick über diese Mechanismen und Techniken.

### 4.1 Konfigurierung vs. Produktauswahl

In Kapitel 3 wurde beschrieben, dass das schrittweise Zusammenfügen eines Produktes aus einer Reihe von Komponenten in der Informatik als Konfigurierung bezeichnet wird. In den einzelnen Domänen, in denen eine solche Tätigkeit vorkommt, ist der Begriff Konfigurierung dagegen meist ungebräuchlich oder hat eine abweichende Bedeutung. Die Arbeitsaufgabe, ein Produkt aus einer Reihe von Komponenten zusammenzustellen, wird stattdessen mit verschiedensten Begriffen wie zum Beispiel Auslegung, Aggregation oder parametrisierte Produktsu-

che bezeichnet<sup>1</sup>. Daher halte ich den Begriff Konfigurierung zur Beschreibung der Arbeitsaufgabe aus Benutzersicht nicht für sinnvoll. Der Begriff wird im Rahmen dieser Arbeit ausschließlich verwendet, um auf die Methoden der Informatik zu verweisen, mit denen diese Arbeitsaufgabe unterstützt werden kann. Außerdem legt die Verwendung des Begriffes Konfigurierung einen Arbeitsprozess nahe, bei dem ein Konfigurator zur Unterstützung eingesetzt wird. Im Rahmen dieses Kapitels wird jedoch zunächst die grundsätzliche Arbeitsaufgabe näher beschrieben. Eine Einschränkung auf die Arbeit mit einem Konfigurator ist hierbei nicht gewollt.

Ziel des Arbeitenden ist es, ein *Produkt zu ermitteln*, das die bestehenden Anforderungen erfüllt, beziehungsweise eine *Lösung für das bestehende Anwendungsproblem* zu finden. Daher werden im Rahmen dieser Arbeit die Begriffe *Produktauswahl* und *Lösungsfindung* zur Beschreibung der Arbeitsaufgabe verwendet. Je nach auszuwählendem Produkt sind hierfür jedoch unterschiedliche Arbeitsprozesse notwendig. Es können drei Kategorien von Produkten unterschieden werden: fest definierte Produkte, individualisierbare Produkte und komplexe Produkte. Bei *fest definierten Produkten* wählt der Kunde entsprechend seinen Kriterien ein Produkt aus. Ein Beispiel hierfür sind Bücher oder CDs. Auch bei *individualisierbaren Produkten* wählt der Benutzer zunächst eine vordefinierte Lösung. Anschließend passt er diese Lösung seinen Kriterien durch Auswahl einzelner Parameter an. Beispiele hierfür sind Taschen<sup>2</sup> und Uhren<sup>3</sup>, für die der Kunde Farbe und Material auswählen kann. *Komplexe Produkte* werden entsprechend den bestehenden Anforderungen aus einer Reihe von Komponenten zusammengesetzt. Beispiele für solche Produkte sind technische Systeme wie Fertigungsanlagen, Computer<sup>4</sup> und Fahrzeuge<sup>5</sup>, aber auch Büroeinrichtungen<sup>6</sup> oder Fertighäuser<sup>7</sup>. Im Gegensatz zur Produktauswahl bei fest definierten oder individualisierbaren Produkten existiert hier also zunächst noch kein Produkt. Es wird erst im Rahmen des Arbeitsprozesses aus Komponenten zusammengestellt. Im engeren Sinne kann daher nicht von Produktauswahl, sondern eher von Produktzusammenstellung gesprochen werden. Da für den Arbeitenden jedoch meistens das Ermitteln des komplexen Produktes im Vordergrund steht, halte ich den Begriff *Produktauswahl* zur allgemeinen Beschreibung der Arbeitsaufgabe durchaus

<sup>1</sup>siehe auch: <http://www-agki.tzi.de/konfiguration2002/resultate/Vehring062002.pdf>

<sup>2</sup><http://www.timbuk2.com/byob/byob.html>, Zuletzt besucht: 02.05.2003

<sup>3</sup><http://www.idtown.com>, Zuletzt besucht: 11.03.2002

<sup>4</sup><http://www.pc.ibm.com/us/eserver/xseries/library/configtools.html>; zuletzt besucht: 02.05.03

<sup>5</sup>Der VW-Konfigurator: (<http://www.volkswagen.de/konfigurator>; zuletzt besucht: 02.05.03) und der BMW-Konfigurator: (<http://www.bmwusa.com/virtual/byo/build.html>; zuletzt besucht: 02.05.2003)

<sup>6</sup>[http://www.officeshop.de/web\\_d/default.htm](http://www.officeshop.de/web_d/default.htm); zuletzt besucht: 02.05.2003

<sup>7</sup><http://www.streif.de/>; zuletzt besucht: 02.05.2003



für treffend. Die Begriffe *Produktauswahl bei komplexen Produkten* und *Lösungsfindung* werden im Folgenden verwendet, um die Zusammenstellung eines komplexen Produktes aus Komponenten zu beschreiben.

Günter u. a. (2001) prägen für die oben beschriebenen drei Klassen von Arbeitsprozessen die Begriffe *Click&Buy*, *Customize&Buy* und *Configure&Buy*. Im Zentrum dieser Arbeit steht der Arbeitsprozess beim *Configure&Buy*.

## 4.2 Benutzergruppen

Laut Nielsen sollten alle Personen, deren Arbeit durch ein Anwendungssystem beeinflusst wird, auch als Benutzer dieses Systems verstanden werden (siehe hierzu Abschnitt 2.1.2). Ein Anwendungssystem, das die Auswahl von komplexen Produkten unterstützen soll, kann daher eine Vielzahl unterschiedlicher Benutzergruppen haben. Diese können nicht allgemein, sondern nur im Kontext eines entsprechenden Anwendungsfalls, beschrieben werden. Unabhängig von einem speziellen Anwendungsfall können jedoch zwei allgemeine Benutzergruppen von zentraler Bedeutung unterschieden werden. Die eine Gruppe bilden Personen, deren Aufgabe im Verkauf der komplexen Produkte besteht. Diese werden im Folgenden als *Vertriebsmitarbeiter* bezeichnet. Üblicherweise kann die Benutzergruppe der Vertriebsmitarbeiter untergliedert werden in *Außendienstmitarbeiter* und *Innendienstmitarbeiter*. Die zweite zentrale Benutzergruppe — als *Kunden* bezeichnet — besteht aus Personen, die komplexe Produkte erwerben. Auf diese beiden allgemeinen Benutzergruppen wird in diesem Abschnitt kurz eingegangen. Die Beschreibung dieser beiden Gruppen, unabhängig von einem konkreten Anwendungsfall, ist natürlich stark verallgemeinernd. Als Grundlage für die Beschreibung der Arbeitsaufgabe ist sie aber ausreichend. In einzelnen Anwendungsgebieten können die Merkmale der Benutzergruppen von den hier beschriebenen abweichen. Die speziellen Benutzergruppen, die im Anwendungsbeispiel SVEA von Bedeutung sind, werden in Kapitel 5 detailliert beschrieben.

Gemeinsam ist den beiden allgemeinen Benutzergruppen das Ziel, entsprechend den Kundenanforderungen ein komplexes Produkt bzw. eine Lösung auszuwählen. Typische Ziele, die Vertriebsmitarbeiter darüber hinaus verfolgen, sind:

- Auswahl eines Produktes, das den Kunden überzeugt.
- Auswahl eines Produktes, mit dem der Kunde auch langfristig zufrieden ist.
- Auswahl eines Produktes, das möglichst dicht am oberen Preislimit des Kunden liegt.
- Auswahl eines Produktes, das aus Komponenten mit möglichst hoher Provision besteht.

- Auswahl von vertrauten Komponenten, mit dem bereits gute Erfahrungen gemacht wurden.

Darüber hinaus werden Vertriebsmitarbeiter oft leistungsbezogen entlohnt und sind daher an einer schnellen Produktauswahl interessiert. Für Kunden stehen dagegen typischerweise die folgenden Ziele im Vordergrund:

- Auswahl eines besonders guten Produktes beziehungsweise einer Lösung, die die bestehenden Anforderungen bestens erfüllt.
- Auswahl eines möglichst günstigen Produktes.
- Auswahl eines Produktes, das innerhalb eines bestimmten Zeitraums verfügbar ist.
- Auswahl eines erweiterbaren, ausbaufähigen Produktes.

Ein weiterer, entscheidender Punkt, in dem sich Vertriebsmitarbeiter und Kunden normalerweise unterscheiden, liegt in ihrem Vorwissen und ihren Erfahrungen. Entsprechend den von Nielsen beschriebenen Kategorien (siehe hierzu Abschnitt 2.1.2) kann unterschieden werden zwischen: allgemeinem Computerwissen, Wissen über das Anwendungssystem und Domänenwissen. Das Anwendungssystem ist in diesem Zusammenhang ein Konfigurator mit dessen Hilfe die komplexen Produkte zusammengestellt werden können. Das für die Arbeitsaufgabe relevante Domänenwissen umfasst typischerweise Wissen über die einzelnen Komponenten und ihre Kombinationsmöglichkeiten, sowie Wissen über das Vorgehen bei der Zusammenstellung der komplexen Produkte und Wissen über die Anwendung oder den Einsatz der Produkte. Kunden haben meist nur Wissen über die Anwendung der komplexen Produkte. Vertriebsmitarbeiter dagegen haben normalerweise ein umfangreiches Wissen über Komponenten und ihre Kombinationsmöglichkeiten. Abbildung 4.1 a) gibt einen Überblick über den typischen Umfang des Domänenwissens in den beiden Benutzergruppen. Während das Domänenwissen der meisten Kunden gering ist, können Vertriebsmitarbeiter im Allgemeinen auf ein umfangreiches Domänenwissen zurückgreifen. Oft haben Vertriebsmitarbeiter in einem speziellen Teilbereich ein besonders tiefgehendes Domänenwissen.

Abbildung 4.1 b) zeigt den typischen Umfang des Wissens über den eingesetzten Konfigurator in den beiden Benutzergruppen. Wird ein Konfigurator erstmalig zur Unterstützung von Benutzern bei der Produktauswahl eingesetzt, so kann zunächst keine der Benutzergruppen auf Erfahrungen mit dem System zurückgreifen. Aufgrund des unterschiedlich umfangreichen Domänenwissens können Kunden anfangs als Anfänger und Vertriebsmitarbeiter als erstmalige Anwender

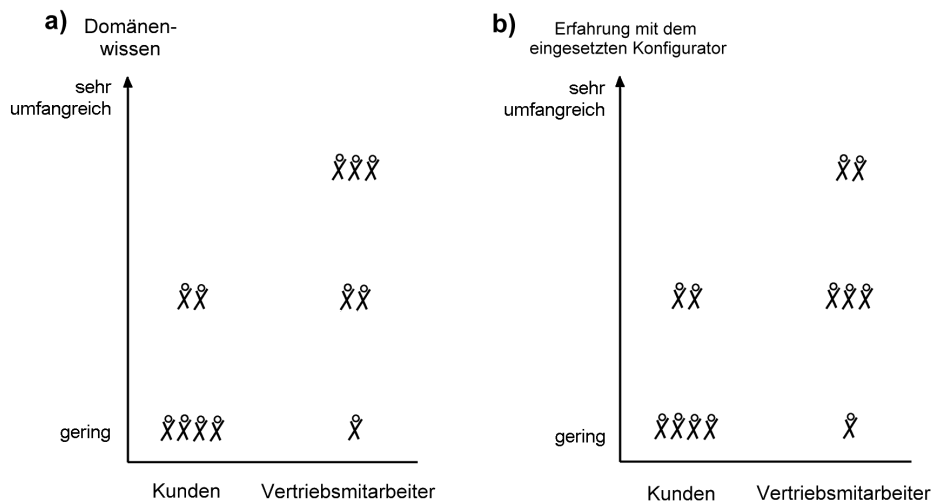


Abbildung 4.1: Menge der Kunden / Vertriebsmitarbeiter mit (a) geringem, mittlerem bzw. umfangreichem Domänenwissen und (b) geringem, mittlerem und umfangreichem Wissen über den eingesetzten Konfigurator

bezeichnet werden (zur Erläuterung der Begriffe siehe Abschnitt 2.1.2). Da Vertriebsmitarbeiter aber normalerweise regelmäßig Produkte zusammenstellen, werden sie schnell zu erfahrenen Anwendern bzw. Experten. Kunden dagegen bleiben oft Anfänger.

Auf der Basis dieser kurzen allgemeinen Beschreibung der beiden Benutzergruppen Kunden und Vertriebsmitarbeiter wird im Folgenden näher auf die Arbeitsaufgabe Produktauswahl eingegangen.

### 4.3 Arbeitsaufgabe Produktauswahl

Die ersten beiden Teile dieses Abschnittes gehen zunächst allgemein auf die Arbeitsaufgabe Produktauswahl ein. Es werden unterschiedliche Modelle vorgestellt, die diese Arbeitsaufgabe beschreiben. Anschließend wird begründet, warum klassische Expertensysteme zur Unterstützung der Arbeitsaufgabe nur bedingt geeignet sind. Im dritten Teil dieses Abschnittes wird dann die Aufgabe der Produktauswahl speziell bei komplexen Produkten eingehend beschrieben. Es wird auf die Unterschiede eingegangen, die zwischen der Auswahl fest definierter Produkte und der Auswahl komplexer Produkte bestehen.

### 4.3.1 Produktauswahl durch Kunden

O’Keefe und McEachern (1998) beschreiben ein Modell für das Kundenverhalten im e-Commerce. Sie untergliedern den Entscheidungsprozess des Kunden in die folgenden fünf Phasen:

**Need recognition.** Der Kunde erkennt bei sich einen Bedarf .

**Information search.** Der Kunde sucht nach Angeboten, die seinen Bedarf decken könnten. Um eine Wahl treffen zu können, muss er eine bestimmte Informationsgrundlage haben oder sich verschaffen.

**Evaluation.** Der Kunde nutzt Heuristiken, um Produkte zu bewerten und zu vergleichen.

**Purchase.** Der Kunde bestellt und bezahlt das ausgewählte Produkt.

**After purchase evaluation.** Der Kunde bewertet den Kauf in Hinblick auf zukünftige Entscheidungen.

Die Autoren betonen, dass es sich hierbei um einen nur teilweise strukturierten Prozess handelt, der nicht auf einen Algorithmus reduziert werden kann. So entspricht der tatsächliche Ablauf nicht unbedingt der linearen Struktur der obigen Darstellung der Phasen.

Ein weiteres Modell für das Kundenverhalten beschreiben Maes u. a. (1999). Sie untergliedern den Prozess in die Phasen *“need identification“*, *“product brokering“*, *„merchant brokering“*, *„negotiation“*, *„purchase and delivery“* sowie *„product service and evaluation“*. Auch Maes u. a. (1999) betonen, dass diese Phasen oft in einer iterativen und nicht linearen Weise durchlaufen werden.

Miles u. a. (2000) beschreiben die Aktivitäten von Kunden in den ersten Phasen dieser Modelle detaillierter. Hierbei beschränken sie sich auf den Bereich des B2C-e-Commerce<sup>8</sup>. Sie unterscheiden die folgenden drei zielgerichteten Aktivitäten:

**Suche nach Produkten.** Nachdem der Kunde seinen Bedarf erkannt hat, verfügt er über eine Reihe von mehr oder weniger genauen Kriterien, die das Produkt erfüllen soll. Auf der Grundlage dieser Kriterien sucht der Kunde nach passenden Produkten. Irgendwann, zum Beispiel als Reaktion auf neue Informationen oder eine eigene Entscheidung, kann sich der Kunde entschließen zu einer anderen Aktivitäten über zu wechseln.

---

<sup>8</sup>Business-to-Consumer-e-Commerce in Abgrenzung zu Business-to-Business-e-Commerce. Ein Großteil der komplexen Produkte fällt in den Bereich des B2B-e-Commerce.

**Management der Suchkriterien.** Neue Informationen können den Benutzer zur Änderung seiner Suchkriterien veranlassen. Diese Änderungen können sowohl in leichten Anpassungen als auch in der Formulierung vollständig neuer Kriterien bestehen.

**Vergleich von Produkten.** Um aus einer Reihe von potentiellen Produkten zu wählen, muss der Kunde diese vergleichen können.

Miles u. a. (2000) betonen, dass die Produktsuche und das Management der Suchkriterien normalerweise sich abwechselnde Tätigkeiten sind. Der Benutzer kann also üblicherweise nicht schon zu Beginn seine Suchkriterien oder Produkthanforderungen genau und vollständig beschreiben.

Pu und Faltings (2000) kritisieren, dass in den meisten Systemen zur Produktauswahl die beiden Phasen *“need identification“* und *“product brokering“* getrennt werden (Phasen nach Maes u. a. (1999)). Dies halten die Autoren für falsch, da ihrer Meinung nach Anforderungen oft anhand des Angebotes festgelegt werden.

„We believe that in most cases, needs define themselves as a result of the products being offered. For example, originally we might have decided that a 300MHz processor was what we needed for our new PC. When we find out that we can get a 366MHz and a CD-Rom drive at almost the same price, these might suddenly become part of our needs as well.“ (Pu und Faltings 2000, S. 289)

Auch Stolpmann (2001, S. 204) beschreibt den typischen Prozess der Produktauswahl als iterativen Prozess, in dem eine Anfrage schrittweise verfeinert wird.

### 4.3.2 Produktauswahl durch Vertriebsmitarbeiter

Rosewitz (2001) beschreibt den Prozess der Beratung von Kunden bei Kaufentscheidungen. Er untergliedert den Prozess in die drei Phasen *Anforderungsanalyse*, *Produktbewertung* und *Ergebnispräsentation*. Im Rahmen der Anforderungsanalyse befragt der Berater zunächst den Kunden und ermittelt so ein Anforderungsprofil, welches er zur Rückversicherung dem Kunden präsentiert. Dem Anforderungsprofil entsprechend wählt der Berater eine Vorauswahl an Produkten aus dem Sortiment aus. Er erläutert dem Kunden den Grad ihrer Übereinstimmung mit dem Anforderungsprofil. Im Rahmen der Ergebnispräsentation stellt der Berater dem Kunden einzelne Produkte vor, verweist eventuell auf Substitutions- und Komplementärprodukte und vergleicht die vorgestellten Produkte. Hierdurch soll der Kunde in die Lage versetzt werden, eine sinnvolle Kaufentscheidung zu treffen. Ist der Kunde nicht mit dem vorgestellten Anforderungsprofil oder den

präsentierten Produkten zufrieden, so wird laut Rosewitz zu der entsprechenden Phase zurückgesprungen.

Emde u. a. (1997) schildern die Entwicklung eines wissensbasierten Systems zur Unterstützung von Vertriebsmitarbeitern bei der Zusammenstellung von Telekommunikationssystemen. In diesem Zusammenhang äußern die Autoren, dass Kunden normalerweise anfangs ihre Anforderungen nicht vollständig und endgültig beschreiben können. Daher kann es im Laufe der Produktauswahl notwendig sein, Anforderungen zu verfeinern oder aufzuheben.

Sowohl der hier vorgestellte Prozess der Kundenberatung als auch die Modelle des Kundenverhaltens aus Abschnitt 4.3.1 charakterisieren die Produktauswahl als einen iterativen Prozess, in dem die Anforderungen an das Produkt schrittweise verfeinert und erweitert werden. Es wird jeweils ein Prozess beschrieben, in dem sich die Definition von Anforderungen und die Bewertung und Auswahl von Produkten überschneiden und abwechseln. Dies steht im Widerspruch zum Aufbau klassischer Expertensysteme, wie sie in Abschnitt 3.4 (Seite 42ff) vorgestellt wurden. Hier wird vorausgesetzt, dass der Benutzer zu Beginn der Arbeit bereits über eine vollständige Anforderungsspezifikation verfügt. Die von Rahmer beschriebenen Systeme zum interaktiven Konfigurieren (siehe Zitat Seite 44) verbinden dagegen die Phasen der Anforderungsspezifikation und der Lösung des Konfigurierungsproblems. Darüber hinaus bieten sie dem Benutzer einen wesentlich größeren Handlungsspielraum als klassische Expertensysteme.

### 4.3.3 Produktauswahl bei komplexen Produkten

Die in den beiden vorigen Abschnitten vorgestellten Modelle der Produktauswahl gehen von fest definierten Produkten aus. Im Folgenden wird erläutert, welche Unterschiede zwischen der Auswahl fest definierter und komplexer Produkte bestehen. Es wird begründet, warum die Auswahl komplexer Produkte aufwändiger und anspruchsvoller ist als die Auswahl fest definierter Produkte.

Die von Miles u. a. (2000) beschriebenen Aktivitäten — Suche nach Produkten, Management der Suchkriterien, Vergleich von Produkten — sind für die Auswahl komplexer Produkte nicht ganz zutreffend. Die Auswahl eines komplexen Produktes umfasst die Auswahl einer ganzen Reihe von Komponenten (fest definierten Produkten). Daher kann im engeren Sinne nicht von der „*Suche nach Produkten*“, sondern eher von der Zusammenstellung eines Produktes gesprochen werden. Es muss also nicht nur eine Entscheidung für ein bestimmtes fest definiertes Produkt getroffen werden, sondern eine Vielzahl von Entscheidungen für die einzelnen Komponenten, aus denen das komplexe Produkt zusammengesetzt ist. Im Vergleich zur Auswahl von fest definierten oder individualisierbaren Produkten ist die Auswahl also viel *aufwändiger*. Für diese Entscheidungen gibt es normalerweise keine feste Bearbeitungsreihenfolge. So werden sinnvolle Vorge-

hensweisen teilweise in Schulungen vermittelt, Experten weichen jedoch bei der Arbeit von diesen ab.

Die Auswahl komplexer Produkte kann daher, entsprechend den in Abschnitt 2.1.2 vorgestellten Aufgabenkategorien, als schwach-strukturierte Aufgabe bezeichnet werden. In Abhängigkeit von den getroffenen Entscheidungen können zusätzliche Entscheidungen notwendig werden. Andererseits können auch einzelne Entscheidungen durch bereits getroffene Entscheidungen entfallen. Die Anzahl und Art der notwendigen Entscheidungen ist somit vor Arbeitsbeginn unbekannt und verändert sich während der Arbeit laufend. Im Gegensatz zur räumlichen Umgebung, die Passini betrachtet, verändert sich die „Umgebung“ der aktuell notwendigen Entscheidungen also ständig.

Um eine korrekte Lösung sicherzustellen, sind neben den bestehenden Anforderungen auch eine Reihe von weiteren Restriktionen bei der Produktzusammensetzung zu beachten. Eine korrekte Lösung entspricht dabei im Sinne von Kapitel 3 einer konsistenten und vollständigen Konfiguration. Zur Überprüfung, ob die bestehenden Restriktionen eingehalten werden, können umfangreiche, teilweise routinemäßige Berechnungen notwendig sein. Darüber hinaus kann die Zusammenstellung eines komplexen Produktes das Erarbeiten neuer Informationen erfordern. Die Auswahl eines komplexen Produktes ist also wesentlich *anspruchsvoller* als die eines fest definierten Produktes. Auch Rosewitz (2001, S. 8) betont, dass „die Auswahl konfigurierbarer Produkte [...] in der Regel deutlich komplexer als die von Standardprodukten [...]“ ist.

Neben diesen hohen Denkanforderungen ist die Auswahl eines komplexen Produktes auch mit erheblichen Behaltensanforderungen<sup>9</sup> verbunden, da neben wichtigen Zwischenergebnissen auch alle Restriktionen, die für die aktuelle Teillösung zu beachten sind, behalten werden müssen. Werden einzelne Restriktionen nicht beachtet, so kann dies zu überteuerten oder sogar inkorrekten Lösungen führen. Dass solche mangelhaften Lösungen im Vertrieb komplexer Produkte tatsächlich ein Problem darstellen, beschreiben Schreier (2001) und Hüllenkremer (2003).

Auch bei der zweiten von Miles u. a. (2000) beschriebenen Aktivität zur Produktauswahl — dem „*Vergleich von Produkten*“ — besteht ein deutlicher Unterschied zwischen fest definierten Produkten und komplexen Produkten. Bei komplexen Produkten ist nicht nur der Vergleich kompletter Produkte sinnvoll, sondern auch der Vergleich von Teillösungen oder einzelnen Komponenten. Der Vergleich von (Teil-)Lösungen wird durch die Tatsache erschwert, dass verschiedene Lösungen sich auch in der Struktur unterscheiden können. Ein direkter Vergleich

---

<sup>9</sup>Geistige Tätigkeiten stellen unterschiedliche Anforderungen an den Arbeitenden. Grundsätzlich können nach Hacker *Verarbeitungsanforderungen* (z.B. Denkanforderungen) und *Behaltensanforderungen* unterschieden werden (Hacker 1994).

auf Komponentenbasis ist daher nicht zwangsläufig möglich.

Genau wie bei fest definierten Produkten ist jedoch der Ausgangspunkt für die Produktauswahl eine Menge von Anforderungen. Diese Anforderungen können, genau wie bei fest definierten Produkten, unvollständig und sogar widersprüchlich sein. Daher kann es im Laufe der Produktauswahl notwendig werden, Anforderungen zu ändern. Beim „*Management von Suchkriterien*“ besteht der wesentliche Unterschied zwischen der Auswahl von fest definierten und komplexen Produkten im Umfang der festzulegenden Anforderungen bzw. Entscheidungen.

Je nach „Qualität“ der Anforderungen entspricht die Produktauswahl eher einem linearen Prozess oder eher einer *iterativen Annäherung* an die Lösung. Hierzu passend beschreibt Hacker (1992, S. 34f), dass Experten nur bei wenig komplexen, begrenzt algorithmischen Entwurfsvorgängen ein systematisches Vorgehen einsetzen. Bei komplexen und schlecht definierten Problemen vom Typ des Entwerfens wird dagegen erst dann systematisch vorgegangen, wenn mit Hilfe eines *opportunistischen Vorgehens* eine vollständige Problemzerlegung erarbeitet wurde. Dieses opportunistische Vorgehen ist laut Hacker gekennzeichnet durch einen unregelmäßigen Wechsel zwischen Problemanalyse, Verfeinerung der Anforderungen und einzelnen Lösungsschritten. Solche Wechsel im Vorgehen werden, wie Hacker beschreibt, vor allem durch neu erarbeitetes Wissen ausgelöst.

Passend zu Hackers Beschreibung eines opportunistischen Vorgehens bis zur vollständigen Problemzerlegung stellt Rahmer (1999, S. 19) die These auf, dass Konfigurierung eine explorative Entwurfstätigkeit ist. Er grenzt die explorative Konfigurierung von idealen Konfigurierungsaufgaben ab, bei denen ein wohl definiertes Problem gegeben ist. Bei der explorativen Konfigurierung werde dagegen von einer unvollständigen Problemspezifikation ausgegangen, die in eine vollständige Problemspezifikation und deren Lösung überführt werde; dies veranschaulicht Abbildung 4.2. Nach Rahmer (1999, S. 23) ist explorative Konfigurierung

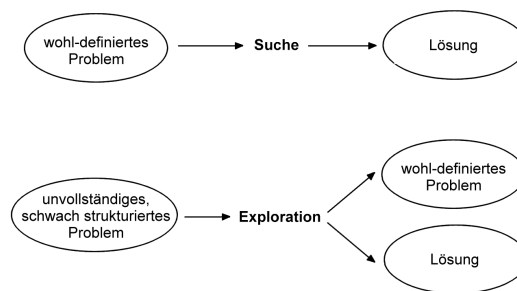


Abbildung 4.2: Gegenüberstellung von Konfigurierung als Suche und Exploration nach Rahmer (1999, S. 18)



„ein interaktiver und inkrementeller Prozeß zur Konstruktion einer Konfiguration, der durch die Generierung und Auswertung von Alternativen geleitet wird.“

Zusammenfassend charakterisiere ich die Produktauswahl bei komplexen Produkten als einen mehr oder weniger stark iterativen Prozess, in dem:

- Anforderungen ermittelt oder verfeinert werden,
- Anforderungen verworfen werden,
- Informationen gesucht und genutzt werden,
- Entscheidungen getroffen werden,
- Entscheidungen zurückgenommen werden,
- Komponenten verglichen werden,
- (Teil-)Lösungen bewertet und verglichen werden,
- die Einhaltung bestehender Restriktionen überprüft wird.

Aufbauend auf dieser Charakterisierung der Arbeitsaufgabe wird im Folgenden darauf eingegangen, warum die Orientierung in diesem Arbeitsprozess problematisch ist. Wie oben beschrieben, ist die Anzahl und Art der Entscheidungen, die im Rahmen der Produktauswahl zu treffen sind, variabel und verändert sich fortlaufend bei der Arbeit. Darüber hinaus verändern sich die notwendigen Entscheidungen auch durch zurückgenommene Entscheidungen. Auch die Struktur des Arbeitsergebnisses – des komplexen Produktes – ist zunächst unbekannt und kann sich während der Arbeit verändern. Die Orientierung bezüglich des Arbeitsprozesses und der Struktur des Produktes zu erhalten, stellt daher hohe Anforderungen an den Arbeitenden. Der Verlust der Orientierung kann Demotivation, Stress und Frustration beim Benutzer auslösen. Diese negativen Auswirkungen können zum Beispiel zu einer verminderten Arbeitsleistung bei Vertriebsmitarbeitern führen. Bei Kunden dagegen kann hierdurch auch der komplette Abbruch der Produktauswahl ausgelöst werden.

Ziel der von mir erarbeiteten Unterstützungsansätze ist es, Benutzer bei der Orientierung im Prozess der Produktauswahl zu unterstützen. Darüber hinaus wird untersucht, welche unterschiedlichen Navigationsmöglichkeiten im Arbeitsprozess der Produktauswahl möglich und sinnvoll sind. Im nächsten Abschnitt werden kurz Mechanismen und Techniken zur Unterstützung der Produktauswahl vorgestellt.

## 4.4 Mechanismen und Techniken zur Unterstützung der Produktauswahl

Der Handel von Produkten über das Internet hat eine stetig wachsende Bedeutung. Um Benutzer bei der Auswahl von Produkten im WWW zu unterstützen, werden unterschiedliche Mechanismen und Techniken eingesetzt und kombiniert. Im Folgenden wird ein kurzer Überblick über diese Mechanismen und Techniken sowie die Einordnung der Produktkonfigurierung gegeben.

Nach Stolze (1999) können Mechanismen zur Unterstützung einer analytischen Produktauswahl in drei Kategorien untergliedert werden:

**Filtermechanismen** erlauben es dem Benutzer, ihn interessierende Teilmengen der angebotenen Produktmenge zu ermitteln. Hierfür können dem Benutzer unterschiedliche Filtertechniken angeboten werden, wie zum Beispiel Produkthierarchien oder Produktkonfiguratoren. Auf Filtertechniken wird weiter unten näher eingegangen.

**Visualisierungsmechanismen** dienen der Darstellung von Produktinformationen. Eine bekannte Technik aus diesem Bereich sind Tabellen, die den direkten Vergleich von Produkten ermöglichen. Als Beispiel für eine Verbindung von Filter- und Visualisierungsmechanismen nennt Stolze den Dynamic HomeFinder von Williamson und Shneiderman (vgl. Williamson und Shneiderman (1992)). Dieser ermöglicht dem Benutzer die Formulierung von Anfragen mit Hilfe von Schiebereglern und stellt Ergebnisse sofort dar.

**Evaluationsmechanismen** bieten über die Ermittlung einer Produktteilmenge hinaus auch eine Bewertung der einzelnen Produkte. Auf der Grundlage einer Bewertungsfunktion wird für jedes Produkt ein numerischer Wert bestimmt, der ein Ranking der Produkte erlaubt.

Systeme zur Produktauswahl bieten dem Benutzer normalerweise eine Kombination dieser Mechanismen. Laut Stolze ist das Angebot von Visualisierungs- und Evaluationsmechanismen bei höherpreisigen Produkten besonders wichtig, da sie dem Benutzer die Möglichkeit bieten, sich von der Richtigkeit seiner Entscheidung zu überzeugen.

Pu und Faltings (2000) unterscheiden die folgenden vier Techniken zur Produktauswahl:

**Hierarchien** erlauben die Auswahl von Teilmengen des Produktspektrums anhand einer vorgegeben Kategorisierung der Produkte. Ein Beispiel für den Einsatz dieser Technik ist in Abbildung 4.3 gegeben.

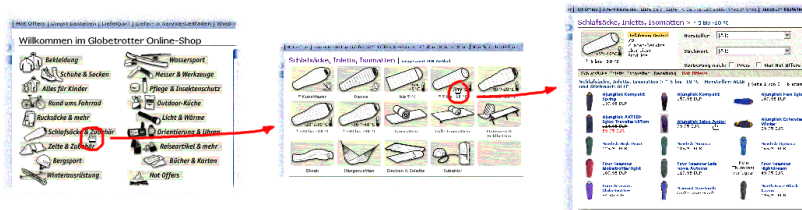


Abbildung 4.3: Produktauswahl über eine Produkthierarchie am Beispiel Globetrotter Ausrüstung (<http://www.globetrotter.de>; zuletzt besucht: 26.04.2003)

**Filter** werden genutzt, um anhand von mehr oder weniger umfangreichen Anfragen eine Teilmenge des Produktspektrums zu beschreiben. Die einfachste Form dieser Technik ist die Angabe eines Stichwortes zur Produktsuche, wie sie in Abbildung 4.4 gezeigt wird.



Abbildung 4.4: Produktauswahl über einen Filter am Beispiel Globetrotter Ausrüstung (<http://www.globetrotter.de>; zuletzt besucht: 26.04.2003)

**Konfigurierung** erlaubt es dem Benutzer, komplexe Produkte aus Komponenten zusammensetzen; ein Beispiel zeigt Abbildung 4.5. Genau wie die beiden vorherigen Techniken wird die Konfigurierung den Filtermechanismen zugerechnet. Hierbei wird aus der Menge aller möglichen Kombinationen von Komponenten ein technisch korrektes und vollständiges, komplexes Produkt ermittelt, das den Anforderungen des Benutzers gerecht wird.



Abbildung 4.5: Produktauswahl über einen Konfigurator: der VW-Konfigurator (<http://www.volkswagen.de/konfigurator>; zuletzt besucht: 03.02.2003)

**Präferenzen** bieten die Möglichkeit, unscharfe Anforderungen an ein Produkt zu definieren. So können auch Produkte ermittelt werden, die die beschriebenen Anforderungen nicht vollständig erfüllen. Die ermittelten Produkte werden entsprechend ihrer Übereinstimmung mit den Anforderungen bewertet, wofür ein System die Anforderungen gewichten können muss. Während die ersten drei Techniken den Filtermechanismen zuzuordnen sind, werden hierbei Filter- und Evaluationsmechanismen verbunden.

Abbildung 4.6 gibt einen Überblick über die vorgestellten Techniken für die Produktauswahl.

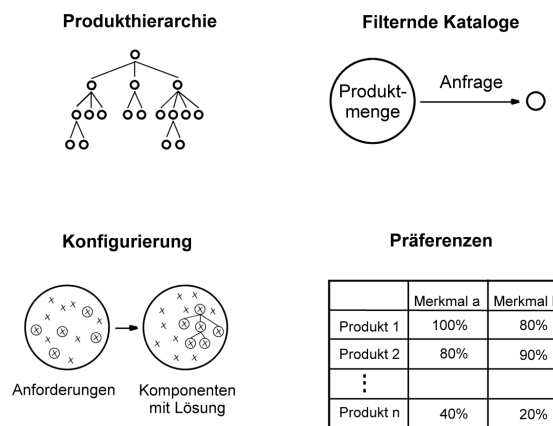


Abbildung 4.6: Techniken für die Produktauswahl

Ein weiteres Beispiel für die Verbindung von Filter- und Evaluationsmechanismen sind interaktive Tabellen wie FOCUS und die Weiterentwicklung InfoZoom, die von Spence u. a. (1996) sowie Callahan und Koenemann (2000) beschrieben werden. Hierbei wird von einer potentiell sehr großen Tabelle ausgegangen, die alle Produkte umfasst. Durch die Angabe von Auswahlkriterien kann der Benutzer diese Tabelle sukzessive einschränken bis zur Auswahl eines einzelnen Produktes. Alle getroffenen Einschränkungen werden dargestellt und können jederzeit wieder aufgehoben werden. Diese Technik eignet sich nach Meinung der Autoren für Tabellen mit bis zu einigen Hundert Zeilen und Spalten. Abbildung 4.7 zeigt ein Beispiel für eine solche Tabelle.

Eine weitere Technik, die auf dem Mechanismus des Filterns beruht, ist die *opportunistische Exploration*, wie sie von Bryan und Gershman (1999) beschrieben wird. Diese beruht auf der von den Autoren entwickelten Aquarium-Metapher.

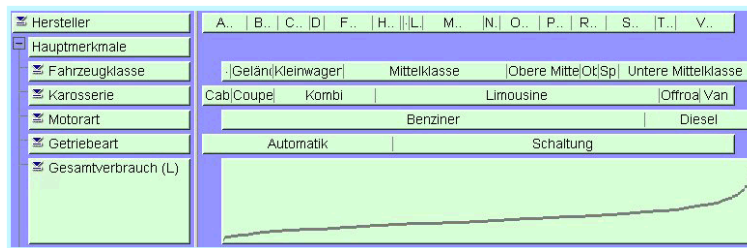


Abbildung 4.7: Interaktive Tabelle für die Produktauswahl: InfoZoom, [http://www.humanit.de/de/produkte\\_loesungen/products/iz/finding/index.html](http://www.humanit.de/de/produkte_loesungen/products/iz/finding/index.html); zuletzt besucht: 02.05.2003

Dem Benutzer werden jeweils etwa ein Dutzend Produkte präsentiert. Diese bewegen sich langsam wie Fische in einem Aquarium. Mit der Zeit wechselt die angezeigte Produktauswahl. Der Benutzer kann entweder passiv die „vorbeiziehenden“ Produkte betrachten oder die angezeigte Auswahl durch Operationen („more like this“, „less like this“) beeinflussen.

Ein zentrales Merkmal von Systemen zur Produktauswahl ist die Art, in der der Benutzer seine Anforderungen formulieren kann. Felix u. a. (2001) unterscheiden hierbei zwischen „*feature-oriented product access*“ und „*need-oriented product access*“. Beim Ersten gibt der Benutzer zum Beispiel bei der Auswahl einer Digitalkamera die gewünschte Auflösung und Speicherkapazität an. Dies kann jedoch für Personen mit geringem Domänenwissen schwer zu spezifizieren sein. Daher halten die Autoren für Personen mit geringem Fachwissen die zweite Variante für angemessener.

„Instead of asking shoppers about desired features of a product, these catalogues elicit shoppers’ needs and the way shoppers intend to use the desired product.“ (Felix u. a. 2001, S. 400)

Diese beiden Varianten können auch als Produktauswahl aus *Produktsicht* oder aus *Anwendungssicht* bezeichnet werden. Eine Produktauswahl aus Anwendungssicht setzt voraus, dass die funktionalen Anforderungen des Benutzers in Produktmerkmale übersetzt werden. Eine solche Übersetzung erfordert bei komplexen Produkten eine Wissensbasis, die neben einem Produktmodell auch ein Anwendungsmodell umfasst.

Abgesehen von der Konfigurierung gehen alle vorgestellten Techniken zur Produktauswahl von der Menge aller angebotenen Produkte aus. Aufgrund des großen Produktspektrums wird bei komplexen Produkten stattdessen ein Produkt schrittweise zusammengestellt. Hieraus resultieren für den Benutzer einige Nachteile im Vergleich zu den anderen Techniken zur Produktauswahl. So ist es nicht

einfach möglich, sich einen Überblick über das angebotene Produktspektrum zu verschaffen, wie es zum Beispiel durch eine Produkthierarchie möglich wäre. Die Tatsache, dass jeweils nur ein komplexes Produkt im Mittelpunkt steht, reduziert die Einsatzmöglichkeiten von Visualisierungs- und Evaluationsmechanismen. Möchte der Benutzer zum Beispiel Alternativen vergleichen, so muss er diese zunächst alle erarbeiten. An diesem Punkt setzt das von Rahmer (1999) beschriebene Unterstützungsmodell für das explorative Konfigurieren an. Es basiert auf Agenten, die parallel zum Benutzer den Lösungsraum explorieren, und dem Benutzer Alternativen zu seiner Lösung vorschlagen. Darüber hinaus bieten sie dem Benutzer die Möglichkeit, Routineaufgaben zu delegieren.

Nachdem in diesem Kapitel die Arbeitsaufgabe Produktauswahl analysiert wurde, werden im nächsten Kapitel Benutzergruppen und Arbeitsaufgaben in Anwendungsbeispiel SVEA näher beleuchtet.

# Kapitel 5

## Beispiel: SVEA

In diesem Kapitel wird auf Arbeitsaufgaben und Benutzergruppen im Rahmen des Anwendungsbeispiels SVEA eingegangen. SVEA ist ein Anbieter von Gebäudeautomationslösungen auf der Basis von LON (**L**okal **O**perating **N**etwork). Diese Gebäudeautomationslösungen sind ein Beispiel für komplexe Produkte. Zum besseren Verständnis dieses Abschnittes wird zunächst kurz auf die Gebäudeautomation und LON eingegangen. Anschließend wird das Unternehmen SVEA vorgestellt. Der letzte Abschnitt dieses Kapitels geht dann schließlich näher auf Benutzer und Arbeitsaufgaben im Anwendungsfall SVEA ein. Dies ist der Kontext, in dem ich die Unterstützung von Benutzern bei der Auswahl von komplexen Produkten praktisch erprobt habe.

### 5.1 Gebäudeautomation

Die Gebäudeautomation (kurz: GA) umfasst Maßnahmen zur Regelung, Steuerung und Überwachung von Gebäudefunktionen. Je nach Art des Gebäudes können unterschiedliche Anlagen automatisiert werden<sup>1</sup> (siehe auch LON Nutzer Organisation e.V. (2000, S. 1-8)):

- Beleuchtungsanlagen
- Sonnenschutzanlagen
- Wärme- und Kälteerzeugungsanlagen
- Wärme- und Kälteverteilanlagen
- Heizungs-, Lüftungs- und Klimaanlage
- Sanitäreanlagen
- Elektrizitätsverteilanlagen

---

<sup>1</sup>Weiterführende Informationen zur LON-basierten Gebäudeautomation bietet die LON Nutzer Organisation (LNO): <http://www.lno.de/ak/ga/>; zuletzt besucht: 21.04.2003

- Aufzug- und Förderanlagen
- Fernmeldeanlagen
- Sicherheitsanlagen

Traditionell werden in Gebäuden die Anlagen der einzelnen Gewerke getrennt gesteuert. Hierdurch ergibt sich ein sehr hoher Verdrahtungsaufwand, und ein Informationsaustausch zwischen den Systemen ist nur schwer realisierbar.

Im Gegensatz zur konventionellen Technik kann mit LON ein gewerkeübergreifendes System realisiert werden. Es können also Geräte aus den Bereichen Beleuchtung, Sonnenschutz, HLK<sup>2</sup>, EDV usw. über einen gemeinsamen Bus gesteuert werden. LON (**Lokal Operating Network**) ist ein Netzwerk, das auf dem Protokoll LonTalk<sup>3</sup> basiert. Die Gebäudeautomation ist eines der Anwendungsgebiete von LON. Da LON ein dezentral organisiertes System ist, kommunizieren die im Gebäude verteilten Geräte über den gemeinsamen Bus direkt miteinander. Außerdem ist LON ein herstellerunabhängiges System. Es gibt unterschiedliche Anbieter von LON-Geräten für die Gebäudeautomation<sup>4</sup>. Die Geräte dieser Anbieter können in einem gemeinsamen Gebäudeautomationssystem kombiniert werden.

## 5.2 Das Unternehmen SVEA

Die 1995 gegründete SVEA Building Control Systems GmbH & Co<sup>5</sup> ist ein Anbieter von Gebäudeautomationslösungen auf LON-Basis. SVEA vertreibt zur Zeit rund 260 Produkte zur Beleuchtungssteuerung, Sonnenschutzsteuerung, Einzelraumregelung, Zutrittskontrolle, Energieverbrauchserfassung sowie zum Zeitmanagement und für allgemeine Schaltfunktionen. Aus diesen Geräten werden die Gebäudeautomationslösungen – das komplexe Produkt – zusammengestellt. Abbildungen 5.1 und 5.2 zeigen beispielhaft einige Produkte. Über diese Produkte hinaus bietet SVEA auch Dienstleistungen, wie Planung, Inbetriebnahme und Projektmanagement im Bereich der Gebäudeautomation.

Abbildung 5.3 auf Seite 66 verdeutlicht die Stellung von SVEA im Rahmen eines Bauvorhabens. Ein Bauherr beauftragt einen Planer und / oder Architekten mit Planung und Entwurf eines Bauvorhabens. Diese vergeben wiederum Aufträge an Baugewerbe und Handwerk. Um die vergebenen Aufträge erfüllen zu können, schreiben die mit der Ausführung beauftragten Unternehmen die benötigten Geräte zur Gebäudeautomation aus. Produkthersteller wie zum Beispiel

---

<sup>2</sup>Heizung, Lüftung, Klima

<sup>3</sup>siehe auch: <http://www.lno.de/technologie/>; zuletzt besucht: 21.04.2003

<sup>4</sup>Die LON Nutzer Organisation bietet eine herstellerübergreifende Produktdatenbank für LON-Geräte: <http://www.lno-db.de/>; zuletzt besucht: 21.04.2003

<sup>5</sup><http://www.svea.de/>; zuletzt besucht: 21.04.2003





Abbildung 5.1: Bedienelemente aus dem Produktspektrum von SVEA



Abbildung 5.2: Aktoren und Sensoren aus dem Produktspektrum von SVEA

SVEA geben hierzu Angebote ab. Eine Zusammenarbeit zwischen Planung und Produktherstellern (in Abbildung 5.3 durch einen grauen Pfeil gekennzeichnet) ist bei diesem Ablauf eigentlich nicht vorgesehen. Hierauf wird im nächsten Abschnitt eingegangen.

SVEA hat zur Zeit 22 Mitarbeiter, wovon fünf im Vertrieb arbeiten. Neben vier Vertriebsmitarbeitern im Außendienst ist ein Vertriebsmitarbeiter im Innendienst tätig.

Auf der Grundlage dieser kurzen allgemeinen Beschreibung des Unternehmens SVEA werden im Folgenden die relevanten Benutzergruppen und die Arbeitsaufgaben im Anwendungsbeispiel SVEA näher beschrieben.

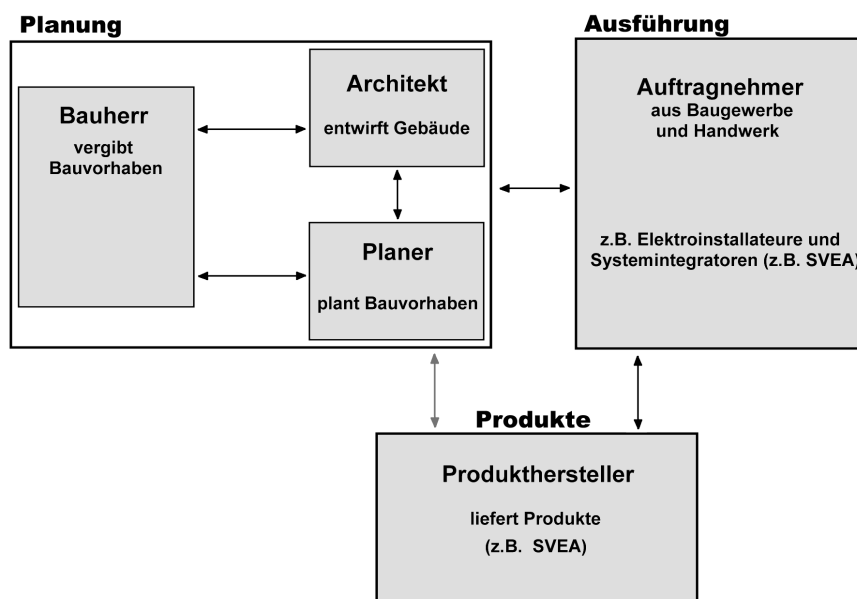


Abbildung 5.3: Informationsfluss im Rahmen von Bauvorhaben. In Anlehnung an: ([http://www.gaeb2000.de/grafik/g2\\_inf.gif](http://www.gaeb2000.de/grafik/g2_inf.gif); zuletzt besucht: 09.01.2003)

### 5.3 Benutzergruppen und Aufgaben im Anwendungsbeispiel SVEA

Wie in Abschnitt 2.1.2 beschrieben wurde, ist die Kenntnis von Benutzergruppen und ihren Aufgaben eine zentrale Grundlage beim Usability Engineering. Daher werden im Folgenden Benutzergruppen und Aufgaben im Anwendungsbeispiel SVEA näher beschrieben. Die hier geschilderten Ergebnisse habe ich im Rahmen einer Reihe von Interviews mit zwei Vertriebsmitarbeitern und einem Planer ermittelt. Das von mir in diesem Zusammenhang eingesetzte Vorgehen wird in Abschnitt 7.5 vorgestellt.

Wie üblich, kann auch bei SVEA die Benutzergruppe der Vertriebsmitarbeiter in Innen- und Außendienstmitarbeiter untergliedert werden. Daneben muss im Beispiel SVEA auch die in Abschnitt 4.2 beschriebene Benutzergruppe der Kunden weiter untergliedert werden:

**Bauherren** sind die eigentlichen Endkunden der Geräte, die SVEA vertreibt. Sie treten aber nicht direkt als Käufer dieser Produkte auf.

**Architekten** haben früher die einzelnen am Bau beteiligten Gewerke koordiniert. Heute sind sie meist auf den baulichen Teil beschränkt.

**Planer** koordinieren die am Bau beteiligten Gewerke. Sie entwerfen entsprechend den Vorstellungen des Bauherren Konzepte für die Gebäudeautomation, schreiben diese aus und vergeben die Umsetzung. Ein Konzept für die Gebäudeautomation beschreibt die zu realisierenden Raumfunktionen und die dafür notwendigen Geräte. Im Folgenden wird ein solches Konzept als *Gebäudeautomationslösung* bezeichnet (kurz: GA-Lösung). Aufgrund des umfangreichen Arbeitsfeldes der Planer haben diese oft nicht die notwendigen Kenntnisse, um LON-basierte GA-Lösungen selbst zu entwerfen. Daher werden sie hierbei durch die Außendienstmitarbeiter von SVEA unterstützt. Diese Zusammenarbeit von Planern und Vertriebsmitarbeitern wurde bereits in Abbildung 5.3 durch einen grauen Pfeil angedeutet.

**Ausführende Unternehmen** sind beispielsweise Elektroinstallateure, die für den Einbau der Geräte in das Gebäude verantwortlich sind.

**Systemintegratoren** setzen die im Gebäude eingebauten Geräte entsprechend der Funktionsvorgaben zu einer funktionierenden Anlage zusammen (Dietrich und Fischer 2001, S. 50). Die Arbeit des Systemintegrators kann verglichen werden mit der Arbeit eines Netzwerkadministrators (Dietrich und Fischer 2001, S. 133ff). Es sind Hard- und Software unterschiedlicher Anbieter zu einem funktionierenden System zusammenzusetzen. Der Systemintegrator weist den Geräten logische Adressen zu. Außerdem werden die Geräte mit Programmen versehen. Um eine konkrete Raumfunktionalität zu realisieren, müssen diese Programme entsprechend parametrisiert werden.

Lediglich die ausführenden Unternehmen und gelegentlich auch die Systemintegratoren sind direkte Käufer der von SVEA angebotenen Geräte. Planer und vereinzelt auch Architekten werden dagegen bei der Ausschreibung dieser Geräte unterstützt, sie treten aber nicht als Käufer auf. Abbildung 5.4 auf Seite 68 erweitert Abbildung 5.3 von Seite 66 um die ermittelten Benutzergruppen und ihre Beziehungen zueinander. Diese werden im Folgenden näher beschrieben.

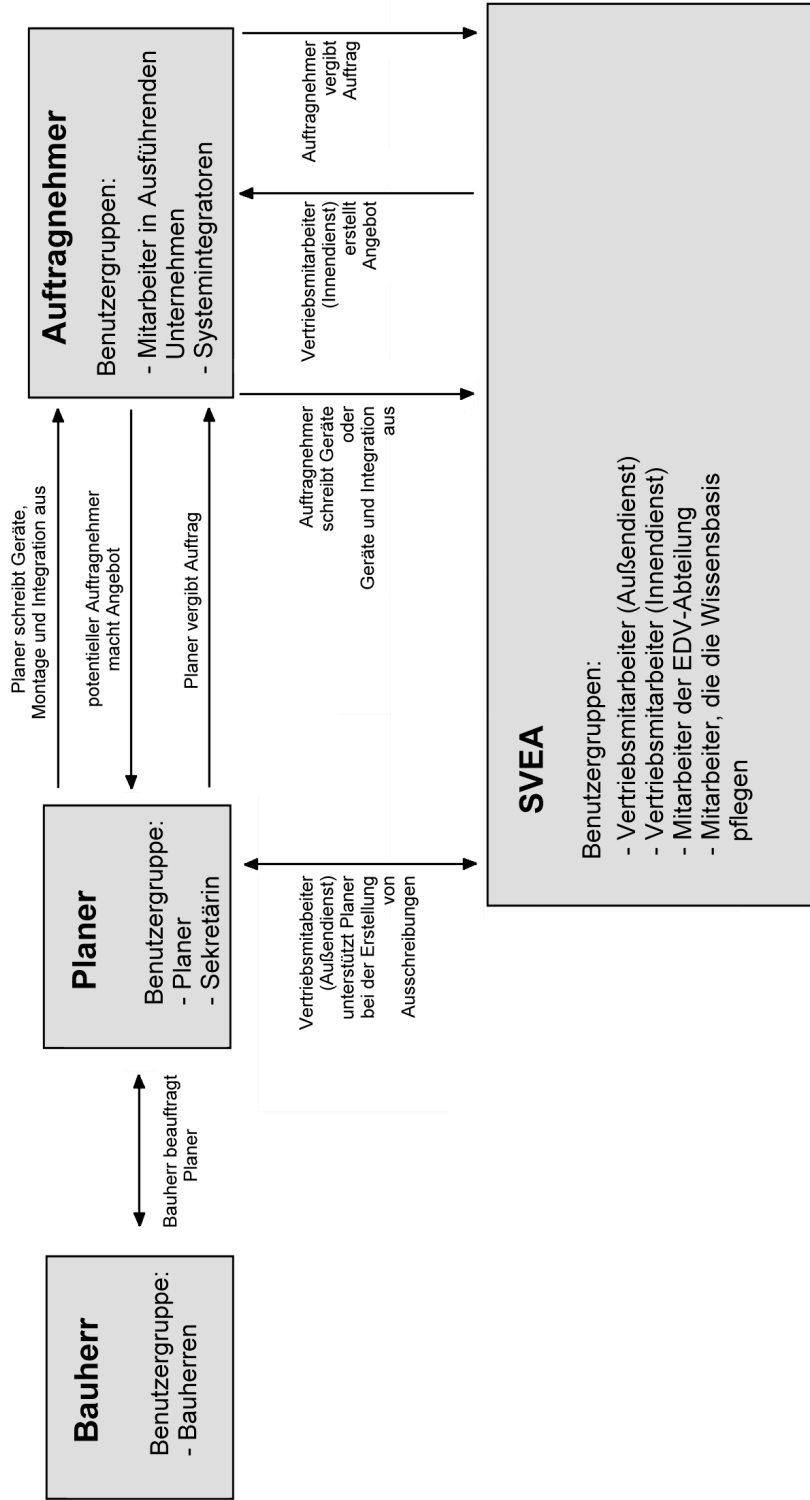


Abbildung 5.4: Benutzergruppen im Anwendungsbeispiel SVEA

### 5.3.1 Benutzerprofile: Planer und Vertriebsmitarbeiter

Alle **Vertriebsmitarbeiter** bei SVEA haben ein technisches oder naturwissenschaftliches Diplom. Es gibt keine Vertriebsmitarbeiter mit einer kaufmännischen Ausbildung. Im Bereich der Gebäudeautomationstechnik haben alle Vertriebsmitarbeiter bereits mehrere Jahre Berufserfahrung. Die Vertriebsmitarbeiter kennen das Produktspektrum von SVEA und teilweise auch das von Konkurrenzanbietern. Außerdem sind sie sachkundig auf dem Gebiet LON und mehr oder weniger umfangreich auch im Bereich anderer busbasierter GA-Systeme. Sie können die Produkte (Komponenten) zu GA-Lösungen (komplexen Produkten) zusammensetzen. Sie arbeiten täglich am Computer und können als geübte Computernutzer bezeichnet werden. Erfahrungen mit einer Konfigurierungsanwendung aus einem beliebigen Einsatzbereich können nicht vorausgesetzt werden. Das Wissen zu Konfigurierungsanwendungen und das Domänenwissen der Vertriebsmitarbeiter wird in Abbildung 5.5 dargestellt.

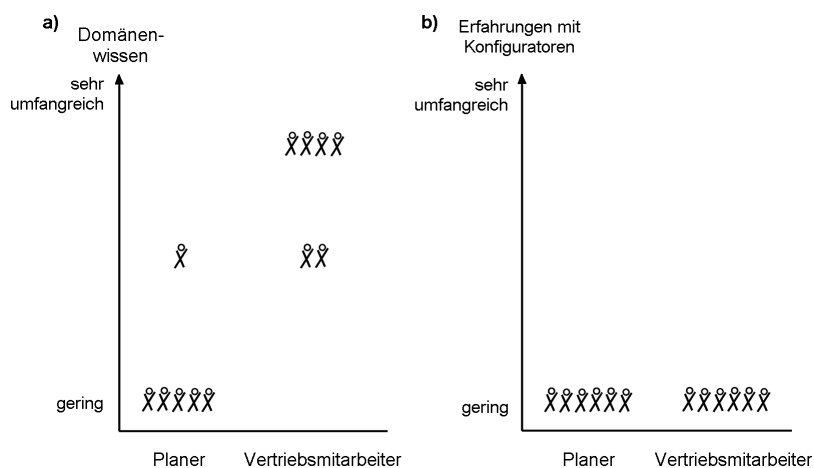


Abbildung 5.5: Menge der Planer / Vertriebsmitarbeiter mit (a) geringem, mittlerem bzw. umfangreichem Domänenwissen und (b) geringer, mittlerer und umfangreicher Erfahrung mit Konfiguratoren

Das oberste Ziel der Vertriebsmitarbeiter bei der Arbeit besteht im Absatz der Produkte. Hierfür müssen optimale GA-Lösungen erarbeitet werden, die den Kunden überzeugen und auch langfristig zufriedenstellen. Aufgrund der stetigen Produktweiterentwicklung ist dazu eine ständige Weiterbildung notwendig. Die Vertriebsmitarbeiter bringen ein großes Engagement für ihre Arbeit mit und sind motiviert, diese möglichst effizient zu gestalten.

**Planer** sind vorwiegend Ingenieure. Da aber für Planer keine bestimmte Ausbildung vorgeschrieben ist, kann dies nicht vorausgesetzt werden. Die Erstellung

von LON-basierten GA-Lösungen macht nur einen kleinen Teil des Aufgabenspektrums der meisten Planer aus. Im Gegensatz zu den Vertriebsmitarbeitern von SVEA haben die meisten Planer kaum oder gar kein Wissen über LON und die Produkte von SVEA. Ihnen fehlt also das notwendige Wissen, um selbständig LON-basierte GA-Lösungen zusammenzustellen. Genau wie die Vertriebsmitarbeiter können Planer als geübte Computernutzer bezeichnet werden und es kann nicht davon ausgegangen werden, dass sie bereits Erfahrungen mit einer Konfigurierungsanwendung gesammelt haben. In Abbildung 5.5 auf Seite 69 wird das Wissen der Planer dem der Vertriebsmitarbeiter von SVEA gegenübergestellt.

Das oberste Ziel der Planer besteht darin, eine optimale GA-Lösung zu ermitteln. Da Planer leistungsbezogen bezahlt werden, spielt für sie außerdem eine möglichst schnelle Lösungsfindung eine zentrale Rolle.

### 5.3.2 Zusammenarbeit von Planern und Vertriebsmitarbeitern

Die im Folgenden beschriebene Zusammenarbeit von Planern und Vertriebsmitarbeitern ist schematisch auch in Abbildung 5.4 auf Seite 68 zusammengefasst.

Bevor Planer eine konkrete GA-Lösung ausschreiben können, müssen sie sich eine Vorstellung von den entstehenden Kosten verschaffen. Zu diesem Zweck werden *Kostenschätzungen* erstellt. Eine Kostenschätzung umfasst die für eine bestimmte GA-Lösung benötigten Geräte sowie ihre Preise. Abbildung 5.6 gibt ein Beispiel für eine Kostenschätzung.

Da Planern, wie oben beschrieben, oft das nötige Wissen zur Erstellung von LON-basierten GA-Lösung fehlt, wenden sie sich zu diesem Zweck an einen


Kostenschätzung								
Pos.	Pos.-Bezeichnung	Fabrikat	Artikel-Bezeichnung	Artikel-Nr.	Stück	Listenpreis	Komponenten	
							EP	GP
	pro Klassenraum							
Anteil 0,5	Dimmer	SVEA	LON DALI-Controller REG-S 4DIM	36236 - 128	1	407,00	407,00	203,50
Anteil 0,25	Digital-Ausgang 4f	SVEA	LON I/O-Modul REG-M 4S	32333 - 067	0	238,00	238,00	59,50
Anteil 1	Digital-Eingang 4f	SVEA	LON I/O-Modul UP 4DI	31312 - 195	1	105,00	105,00	105,00
Anteil 1	Multisensor	SVEA	LON Multisensor LA-21	42320 - 104	1	192,00	192,00	192,00
Anteil 1	Busankoppler m. Modul	SVEA	LON Busankoppler m. Raumtemp -panel	63315 - 166	1	155,00	155,00	155,00
Anteil 1	Stellantrieb	SVEA	LON Stellantrieb SA-11	62301 - 118	1	172,00	172,00	172,00
								<b>887,00</b>
	<b>für 26 Klassenräume</b>							
	Dimmer	SVEA	LON DALI-Controller REG-S 4DIM	36236 - 128	13	407,00	407,00	5.291,00
	Digital-Ausgang 4f	SVEA	LON I/O-Modul REG-M 4S	32333 - 067	7	238,00	238,00	1.666,00
	Digital-Eingang 4f	SVEA	LON I/O-Modul UP 4DI	31312 - 195	26	105,00	105,00	2.730,00
	Multisensor	SVEA	LON Multisensor LA-21	42320 - 104	26	192,00	192,00	4.992,00
	Busankoppler m. Modul	SVEA	LON Busankoppler m. Raumtemp -panel	63315 - 166	26	155,00	155,00	4.030,00
	Stellantrieb	SVEA	LON Stellantrieb SA-11	62301 - 118	26	172,00	172,00	4.472,00
	Spg-vers m. Router	SVEA	LON Spannungsversorgung m. Router LPS/RTR-W	11231 - 006	3	842,00	842,00	2.526,00
	Systemuhr	SVEA	LON Systemuhr REG 4 DCF	41334 - 087	1	325,00	325,00	260,00
	Zubehör Systemuhr	SVEA	DCF-77 Empfänger	41021 - 089	1	61,00	61,00	48,80
								<b>26.015,80</b>
						Summe		

Abbildung 5.6: Kostenschätzung für eine GA-Lösung

Vertriebsmitarbeiter. Sie schildern einem Außendienstmitarbeiter die im Bauvorhaben bestehenden funktionalen Anforderungen. Der Außendienstmitarbeiter erstellt auf der Grundlage dieser Informationen eine Kostenschätzung und eine Ausschreibung. Im Mittel umfassen GA-Lösungen bei SVEA etwa 500 Geräte. Es treten aber auch Lösungen mit mehr als 5000 Geräten auf. Theoretisch sind der Anzahl der Geräte keine Grenzen gesetzt. Eine Ausschreibung ist meist ein *Massenleistungsverzeichnis*. Dies umfasst neben der Nennung der einzelnen Geräte und der benötigten Geräteanzahl auch deren nähere Beschreibung. Ein Beispiel für eine solche Gerätebeschreibung ist im Anhang enthalten (siehe Seite 168). Neben den Geräten wird auch deren Montage und Integration ausgeschrieben. Im Anhang ist ein Auszug aus einem Massenleistungsverzeichnis abgebildet (siehe Seite 169).

Sowohl einer Kostenschätzung als auch einem Massenleistungsverzeichnis liegt also eine konkrete GA-Lösung zugrunde. Bei der *funktionalen Ausschreibung*, einer weiteren Form der Ausschreibung, ist dies nicht der Fall, da sie nur die zu realisierenden Raumfunktionen beschreibt. Auf die hierfür notwendigen Geräte sowie Montage und Integration der Geräte wird dabei nicht eingegangen. Ein Beispiel für eine funktionale Ausschreibung ist in Abbildung 5.7 gegeben.

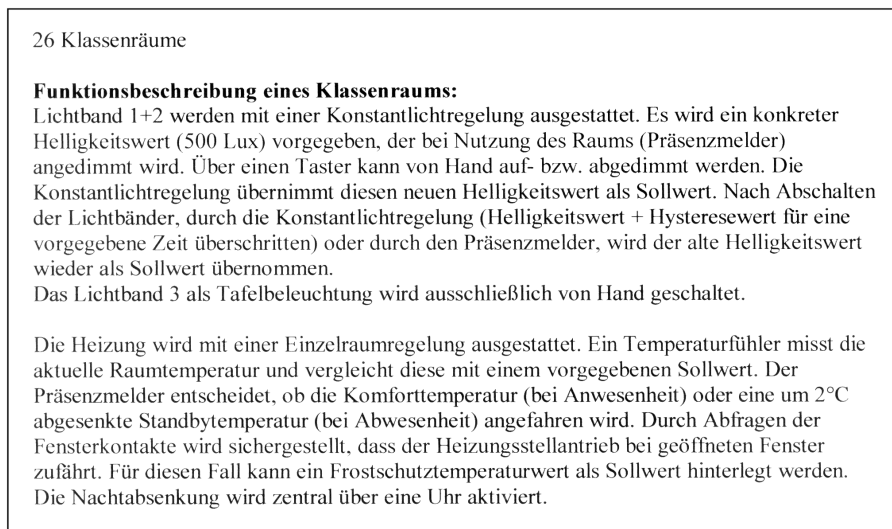


Abbildung 5.7: Funktionale Ausschreibung einer GA-Lösung

Die so erstellte Ausschreibung richtet sich an ausführende Unternehmen wie zum Beispiel Elektroinstallateure. Damit ein solches Unternehmen ein Angebot abgeben kann, muss es zunächst selbst ein Angebot für die benötigten Geräte beim Hersteller einholen. Solche Angebote über Geräte und deren Integration werden bei SVEA fast ausschließlich im Innendienst erstellt.

### 5.3.3 Aufgaben der Benutzergruppen

Im Folgenden wird näher auf die Aufgabenbereiche, das Arbeitsumfeld und die verfügbaren Hilfsmittel von Planern sowie von Vertriebsmitarbeitern eingegangen.

#### Planer

Der Aufgabenbereich des Planers erstreckt sich über den gesamten Zeitraum eines Bauvorhabens. Beginnend mit dem Vorentwurf und der Grundlagenplanung über die Ausführungsplanung, Ausschreibung und Vergabe bis hin zu Objektbetreuung und -übergabe sowie dem Kontakt zum Bauherren und Betreiber. Eine Beschreibung der einzelnen Planungsschritte wird in (Dietrich und Fischer 2001, S. 95f) gegeben. Die von Planern hierbei eingesetzte Software wird als AVA-Software (Ausschreibung, Vergabe, Abrechnung) bezeichnet.

Als *Hilfsmittel* zur Erstellung von LON-basierten GA-Lösungen stehen dem Planer die Mitarbeiter der Gerätehersteller (z.B. SVEA) zur Verfügung. Außerdem greifen Planer bei der Erstellung von Ausschreibungen oft auf alte Ausschreibungen zurück. Im Gegensatz zu Vertriebsmitarbeitern erstellen sie Ausschreibungen nicht neu, sondern passen eher bestehende Ausschreibungen an. Für die Ausschreibung von LON-basierten GA-Lösungen ist dieses Vorgehen jedoch nicht möglich. Dies liegt an der stetigen Weiterentwicklung der Produkte und der Tatsache, dass die meisten Planer nur selten LON-basierte GA-Lösungen ausschreiben.

Mögliche Fehler bei der Erstellung von GA-Lösungen bestehen im Vergessen von einzelnen Geräten oder notwendigem Zubehör. Durch solche Fehler kann ein Bauvorhaben wesentlich kostspieliger werden als vom Planer prognostiziert. Daher stellen sie ein gravierendes Risiko für den Planer dar.

#### Vertriebsaußendienst

Der Vertriebsaußendienst arbeitet vor allem mit Planern zusammen. Die wichtigste Aufgabe des Vertriebsaußendienstes ist die Akquisition von Kunden. In diesen Bereich fällt auch die Unterstützung von Planern bei der Erstellung von Ausschreibungen. Außerdem gehört es zum Aufgabengebiet der Außendienstmitarbeiter, Planern Hintergrundwissen zum Thema LON zu vermitteln. Hierzu gehört auch das Organisieren von Vorträgen und Workshops.

Außendienstmitarbeiter arbeiten etwa zu gleichen Anteilen beim Kunden und im Home Office. Ein Kundenbesuch dauert zwischen ein und drei Stunden. Zur Erstellung von GA-Lösungen ermittelt der Außendienstmitarbeiter im Gespräch mit dem Planer die zu realisierenden Raumfunktionen. Auf der Basis dieser In-



formationen erstellt er im Home Office Kostenschätzung und Ausschreibung und lässt sie dem Planer innerhalb einer Woche zukommen.

### **Vertriebsinnendienst**

Die Hauptaufgabe des Innendienstes besteht im Erstellen von Angeboten (etwa 20-30% der Arbeitszeit). Hierfür arbeitet er sowohl mit ausführenden Unternehmen als auch mit Planern zusammen. Grundlage für die Erstellung eines Angebotes ist eine Ausschreibung. Momentan sind 20% dieser Ausschreibungen funktionale Ausschreibungen und etwa 50% Massenleistungsverzeichnisse. 30% der Ausschreibungen basieren auf einem anderen Bussystem als LON. Zur Erstellung von Angeboten muss der Vertriebsinnendienst oft den zuständigen Planer kontaktieren.

Funktionale Ausschreibungen müssen sehr detailliert sein, damit auf ihrer Grundlage die notwendigen Geräte ermittelt werden können. Dies ist oft nicht der Fall. Die beiden anderen Klassen von Ausschreibungen enthalten dagegen nur die Beschreibung und Anzahl der einzusetzenden Geräte. Diese Geräteangaben beziehen sich jedoch teilweise auf Geräte anderer Hersteller und müssen dann zur Angebotserstellung in entsprechende Geräte von SVEA übersetzt werden. Geräte bieten aber üblicherweise eine Reihe unterschiedlicher Funktionen. In einer konkreten GA-Lösung wird oft nur ein Teil dieser Gerätefunktionen genutzt. Um ein möglichst günstiges Angebot machen zu können, ist es notwendig, die genutzten Gerätefunktionen zu kennen. Diese Informationen sind jedoch in Ausschreibungen nicht enthalten. Darüber hinaus sind Ausschreibungen oft nicht strukturiert, wodurch die Angebotserstellung zusätzlich erschwert wird. Aufgrund dieses Mangels an benötigten Informationen ist der Innendienst oft gezwungen, zur Angebotserstellung telefonisch weitere Informationen beim Planer zu erfragen. Ein weiteres Problem stellen ungünstige oder fehlerhafte GA-Lösungen dar. Werden solche GA-Lösungen vom Planer ausgeschrieben, so muss der Innendienst zunächst eine entsprechende Änderung beim Bauherren oder Planer erwirken, bevor ein Angebot erstellt werden kann.

Ein fertiges Angebot besteht jeweils aus einer Stückliste mit Preisen für Geräte und Dienstleistungen, einem Standardtext sowie Erläuterungen zum Angebot. Die Erstellung eines Angebotes dauert zwischen ein und fünf Tagen. Eine weitere wichtige Aufgabe des Innendienstes ist der Kundenkontakt im Rahmen von Angeboten und die allgemeine Pflege der Kundenbeziehungen. Zusätzlich wird der Außendienst telefonisch beraten. 30-50% der Arbeitszeit besteht aus Telefonaten. Die Arbeit wird also sehr häufig durch Anrufe unterbrochen.

Die Vertriebsmitarbeiter setzen bei der Erstellung von GA-Lösungen eine Reihe von *Hilfsmitteln* ein. Die Produktdaten werden bei SVEA mit Excel verwaltet. Außerdem gibt es zu jedem Produkt eine Produktdokumentation und einen Aus-

schreibungstext. Beispiele für diese Dokumente sind im Anhang enthalten (siehe S. 168ff). Zur Kommunikation mit Kunden werden Telefon und E-Mail eingesetzt. Neben einer Adressverwaltung wird auch ein Vertriebsinformationssystem eingesetzt, mit dessen Hilfe allen Vertriebsmitarbeitern jederzeit Informationen über aktuelle und abgeschlossene Projekte zugänglich sind. Kataloge und das WWW werden genutzt, um Informationen zu Produkten anderer Hersteller zu ermitteln. Darüber hinaus stehen sich die Vertriebsmitarbeiter gegenseitig mit Auskünften zur Seite.

### 5.3.4 Vorgehen bei der Erstellung einer GA-Lösung

Größere Gebäude wie Büro- und Verwaltungsgebäude oder Krankenhäuser bestehen meist aus einer Vielzahl von einheitlichen Räumen. Entscheidend sind hierbei Daten über die Anzahl der Fenster, Beleuchtungsgruppen, Heizkörper und Jalousien im Raum. Zur Erstellung einer GA-Lösung werden solche einheitlichen Räume in Gruppen zusammengefasst (zum Beispiel Büro, Labor und 2-Bettzimmer). Eine Lösung wird zunächst für einen einzelnen Raum erstellt. Es werden die Geräte ermittelt, die zur Realisierung der geplanten Raumfunktionen benötigt werden. Diese Geräte werden anschließend entsprechend der Häufigkeit des Raumes im Gebäude hochgerechnet. Bei der Auswahl der Geräte sind die Montagemöglichkeiten für Geräte im Raum zu beachten. Sind für die Haupträume die Geräte festgelegt, werden spezielle Räume wie Konferenzräume, Flure oder Teeküchen betrachtet.

Eine GA-Lösung umfasst im Mittel etwa 4 unterschiedliche Raumarten. Mehr als 10 unterschiedliche Raumarten kommen nicht vor. Die Aufgabe, eine Lösung für einen Raum zu ermitteln, kann weiter untergliedert werden in die Festlegung der:

- Beleuchtungssteuerung,
- Einzelraumregelung (Heizungssteuerung),
- Sonnenschutzsteuerung und
- Bedienung der Raumfunktionen.

Je nach geplanten Raumfunktionen müssen nicht immer alle diese Teilaufgaben tatsächlich bearbeitet werden. So kann in einem Raum auch ausschließlich eine Beleuchtungssteuerung vorgesehen sein, wodurch die Festlegung von Sonnenschutzsteuerung und Einzelraumregelung entfallen. Neben den Geräten, die in den einzelnen Räumen eingesetzt werden, sind weitere Geräte zur Realisierung des Netzwerkes festzulegen. Abbildung 5.8 zeigt beispielhaft die Struktur der Arbeitsaufgabe.

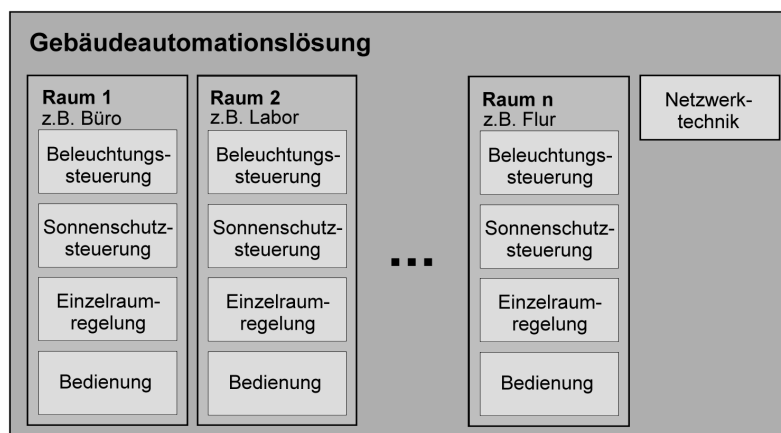


Abbildung 5.8: Struktur der Arbeitsaufgabe

Einige der Teilaufgaben sind abhängig von einander oder beeinflussen sich gegenseitig. So können beispielsweise einige Geräte von Beleuchtungssteuerung und Jalousiensteuerung gemeinsam genutzt werden. Eine Uhr oder Wetterstation wird jeweils nur einmal für die gesamte GA-Lösung vorgesehen. Die Festlegung der Bedienung umfasst zum Beispiel die Auswahl von Tastern. Über diese Taster werden die Raumfunktionen gesteuert. Daher kann die Bedienung erst festgelegt werden, wenn die gewünschten Raumfunktionen bestimmt wurden. Die notwendige Netzwerktechnik ist abhängig von der Anzahl und Art der eingesetzten Geräte. Sie kann daher immer erst zum Abschluss bestimmt werden. Über diese Abhängigkeiten hinaus gibt es jedoch keine feste Reihenfolge, in der die einzelnen Teilaufgaben zur Erstellung einer GA-Lösung bearbeitet werden. Außerdem werden die einzelnen Teilaufgaben nicht immer nacheinander jeweils abschließend bearbeitet. Um eine optimale Lösung zu finden, werden stattdessen oft einzelne Teile geändert. Anschließend müssen die Auswirkungen auf andere Teile der GA-Lösung überprüft werden, um eine korrekte Lösung sicherzustellen.

Änderungen von GA-Lösungen kommen nicht nur während der Erstellung vor. Häufig äußern Kunden nachträglich Änderungswünsche oder wünschen gleich die Erstellung von alternativen Lösungen. Wie bereits beschrieben, ist die Erstellung einer GA-Lösung die Basis für die Erstellung einer Kostenschätzung, eines Massenleistungsverzeichnisses und eines Angebotes. Szenarien, die diese Arbeitsabläufe näher beschreiben, befinden sich im Anhang ab Seite 161. Die hier beschriebenen Ergebnisse zur Arbeitsaufgabe *Erstellung einer GA-Lösung* bei SVEA decken sich mit Ergebnissen zur Arbeitsaufgabe *Produktauswahl*, die in Kapitel 4 beschrieben wurden. Auch hier kann nicht von einer linearen Bearbeitung der einzelnen Teilaufgaben gesprochen werden.

In diesem Abschnitt wurde umfassend auf die zentralen Benutzergruppen im Rahmen dieser Arbeiten eingegangen – Planer und Vertriebsmitarbeiter. Im nächsten Abschnitt werden kurz weitere Benutzergruppen vorgestellt.

### 5.3.5 Weitere Benutzergruppen

Neben den beiden zentralen Benutzergruppen – Planer und Vertriebsmitarbeiter – existieren im Anwendungsbeispiel SVEA weitere Benutzergruppen. Auf diese Benutzergruppen wird nur sehr kurz eingegangen, da ich mich im Rahmen der Arbeit auf die oben beschriebenen Benutzergruppen konzentriert habe.

*Bauherren* erhalten vom Planer eine Kopie der erstellten Ausschreibung und sind daher als Benutzergruppe zu betrachten. Um Ausschreibungen ein einheitliches Layout zu geben, werden diese gegebenenfalls im *Sekretariat des Planungsbüros* überarbeitet. Daher sind auch die Mitarbeiter dort als Benutzergruppe anzusehen. Darüber hinaus sind, wie beschrieben, *ausführende Unternehmen* und *Systemintegratoren* Adressaten der Ausschreibung. Neben Planern übernehmen auch *Architekten* gelegentlich die Planung der Gebäudeautomation. Daher sind auch sie als Benutzergruppe einzustufen. Neben den Vertriebsmitarbeitern bestehen auch bei SVEA weitere Benutzergruppen. So ist die *EDV-Abteilung* betroffen, wenn ein Programm zur Unterstützung von Vertriebsmitarbeitern eingesetzt wird oder Planern über die Homepage ein entsprechendes Angebot gemacht wird. Beim Einsatz eines wissensbasierten Systems kommt außerdem die Wartung und Pflege der Wissensbasis hinzu.

Die hier beschriebenen Ergebnisse, zu Arbeitsaufgaben und Benutzergruppen, bilden die Grundlage für den von mir im Rahmen der Arbeit entworfenen Prototyp. Dieser wird in Kapitel 7 vorgestellt. Im nächsten Kapitel werden die von mir erarbeiteten Ansätze zur Unterstützung von Benutzern bei der Auswahl komplexer Produkte vorgestellt. Eine Auswahl dieser Unterstützungsansätze wird im erarbeiteten Prototyp eingesetzt.

# Kapitel 6

## Unterstützung der Produktauswahl

Kapitel 6 erläutert die entwickelten Ansätze zur Unterstützung von Benutzern bei Orientierung und Navigation in der Produktauswahl komplexer Produkte. Der erste Abschnitt dieses Kapitels beschreibt die Anforderungen, die den Unterstützungsansätzen zugrunde liegen, gefolgt von einem Überblick über die Unterstützungsansätze. Als Grundlage für die Unterstützungsansätze wird in Abschnitt 6.3 auf die Unterschiede zwischen einer formularbasierten und einer direktmanipulativen Produktauswahl eingegangen. Im Abschnitt 6.4 werden dann die Ansätze zur Unterstützung der Orientierung vorgestellt, gefolgt von den Ansätzen zur Unterstützung der Navigation in Abschnitt 6.5.

### 6.1 Anforderungen

#### 6.1.1 Anforderungen — Orientierung und Navigation

Im Folgenden werden zunächst Anforderungen beschrieben, die speziell zur Unterstützung von Orientierung und Navigation im Arbeitsprozess umgesetzt werden sollten.

Die Arbeitsaufgabe Produktauswahl stellt eine komplexe Umgebung dar, in der sich der Arbeitende orientieren muss. Als Gründe für die Komplexität dieser Umgebung habe ich in Kapitel 4 die große Menge an Entscheidungen, die dynamische Veränderung der notwendigen Entscheidungen und die Abhängigkeiten zwischen den einzelnen Entscheidungen genannt. In dieser Umgebung muss sich der Arbeitende zurechtfinden, unabhängig davon, ob er das komplexe Produkt mit oder ohne Hilfe eines Konfigurators zusammenstellt. Beim Einsatz eines Konfigurators können dem Benutzer zusätzliche Umgebungsinformationen geboten werden, wodurch die Orientierung in der Arbeitsaufgabe unterstützt werden kann. Durch den Einsatz eines Konfigurators wird es auch möglich, dass Perso-

nen sich erst während der Arbeit näher mit der Arbeitsaufgabe vertraut machen. Aus der Beschreibung der Arbeitsaufgabe Produktauswahl in Kapitel 4 folgt auch, dass eine flexible Navigation im Arbeitsprozess notwendig ist, da nicht davon ausgegangen werden kann, dass die Produktauswahl einem linearen Ablauf folgt. Während ein Konfigurator das Erarbeiten einer Lösung unterstützt, erfordert die Arbeitsaufgabe Produktauswahl auch das Erforschen von alternativen (Teil-)Lösungen.

Ziel der von mir erarbeiteten Unterstützungsansätze ist es, Orientierung und Navigation im Arbeitsprozess der Produktauswahl zu erleichtern. Dies soll auch für Benutzer möglich sein, die zunächst noch kein fundiertes mentales Modell von der Arbeitsaufgabe haben. Um auch ihnen ein effizientes Arbeiten zu ermöglichen, sollen die Unterstützungsansätze zusätzlich den Auf- bzw. Ausbau eines mentalen Modells von der Arbeitsaufgabe fördern. Schließlich sollen die Ansätze dem Benutzer auf der Basis der Konfigurierung eine flexible Navigation durch die Arbeitsaufgabe ermöglichen. Diese grundlegenden Ziele sind jedoch sehr allgemein. Im Folgenden werden konkretere Anforderungen beschrieben, die durch die Unterstützungsansätze umgesetzt werden sollen.

In Anlehnung an Passinis Begriff des *Wayfinding* wird im Rahmen dieser Arbeit davon ausgegangen, dass sich Menschen auch in unbekannter Umgebung orientieren können (siehe hierzu Abschnitt 2.2.1). Dies ist allerdings nur möglich, wenn Personen, die keine Vorstellung von der Arbeitsaufgabe haben, auf den linearen Orientierungsstil zurückfallen können. Den Benutzern sollte daher ein **Leitsystem** angeboten werden.

Der iterative Charakter der Arbeitsaufgabe erfordert **Flexibilität im Vorgehen** genauso, wie die Tatsache, dass unterschiedliche Personen meist verschieden bei der Produktauswahl vorgehen. Schließlich ist ein flexibles Vorgehen auch notwendig, damit der Vertriebsmitarbeiter bei der Produktauswahl im Kundengespräch entsprechend den Prioritäten des Kunden vorgehen kann. Ein solches flexibles und an die Situation angepasstes Vorgehen ist aber nur auf der Grundlage des räumlichen Orientierungsstils möglich. Neben dem Angebot eines Leitsystems sollte daher auch der **räumliche Orientierungsstil** unterstützt werden.

Da der räumliche Orientierungsstil ein **mentales Modell der Arbeitsaufgabe** voraussetzt, sollten Benutzer dabei unterstützt werden, dieses auf- bzw. auszubauen. Der Benutzer sollte sich also eine Vorstellung von der Arbeitsaufgabe als Ganzes machen können. Zu diesem Zweck sollte er sich anhand der Benutzungsschnittstelle jederzeit die folgenden Fragen beantworten können:

1. Aus welchen Teilaufgaben / Schritten besteht die Arbeitsaufgabe?
2. Wo im Arbeitsprozess stehe ich?
3. Welche Entscheidungen habe ich bereits getroffen?

4. Welche Entscheidungen wurden vom System getroffen?
5. Welche Entscheidungen sind noch offen?
6. Habe ich die Arbeitsaufgabe erfolgreich abgeschlossen?

Unabhängig vom Orientierungsstil sollte der Benutzer anhand der Benutzungsschnittstelle seinen eigenen **Fortschritt bei der Arbeit** einschätzen können.

Aus dem iterativen Charakter der Arbeitsaufgabe folgt außerdem, dass der Benutzer bereits getroffene Entscheidungen einfach **widerrufen** und **ändern** können sollte. Hacker fordert zur Unterstützung des opportunistischen Vorgehens den **schnellen Wechsel zwischen Bereichen** eines Anwendungssystems zu ermöglichen (Hacker 1994, S. 36). Das Navigieren zwischen Bereichen sollte nach Hacker hierbei durch Gedächtnis- und Veranschaulichungshilfen unterstützt werden.

Während Personen ohne eine genaue Vorstellung von der Arbeitsaufgabe überschaubare Schritte benötigen, steht für Experten die Effizienz bei der Produktauswahl im Vordergrund. Insbesondere die Auswahl von „Standardprodukten“ sollte für Experten sehr schnell möglich sein. Während also für Experten **schnelle Wege** im Vordergrund stehen, benötigen Anfänger vor allem **einfache Wege**.

### 6.1.2 Anforderungen — Arbeitsaufgabe Produktauswahl

Die Unterstützung von Benutzern bei Orientierung und Navigation ist nur eine Anforderung, der ein System zur Auswahl komplexer Produkte gerecht werden muss. Aus der Arbeitsaufgabe *Produktauswahl bei komplexen Produkten* resultiert noch eine Menge weiterer Anforderungen. Auf besonders wichtige wird hier eingegangen.

Sowohl Kunden als auch Vertriebsmitarbeiter verfolgen bei der Produktauswahl das Ziel, entsprechend den Kundenanforderungen ein komplexes Produkt auszuwählen. Der Vertriebsmitarbeiter versucht den Kunden im Gespräch von einem bestimmten Produkt zu überzeugen. Wählt der Kunde selbständig ein Produkt aus, so fällt es in seine Verantwortung, die richtige Wahl zu treffen. Der Vergleich verschiedener komplexer Produkte spielt im zweiten Fall eine noch größere Rolle als im ersten. Durch den Vergleich von mehreren Alternativen kann sich der Benutzer von der Qualität seiner Wahl überzeugen. Dass der **komfortable Vergleich von Produkten** eine wesentliche Bedeutung im e-Commerce spielt, wird von vielen Autoren betont (Emde u. a. 1997; Liang und Chen 2000; Miles u. a. 2000; Tilson u. a. 2002).

„Nothing empowers a shopper more than getting the best deal. In both B2B and B2C studies, we’ve seen users again and again ask for ways to compare items side by side.“ (Nielsen und Tahir 2002)

Bei einem solchen Produktvergleich sollten die Unterschiede klar hervorgehoben werden, damit sie mit einem Blick erfasst werden können.

Um verschiedene komplexe Produkte vergleichen zu können, müssen zunächst mehrere komplexe Produkte zusammengestellt werden. Der Benutzer sollte daher einfach und **effizient Alternativen zusammenstellen** können. Damit der Benutzer nicht jede Alternative vollständig erarbeiten muss, sollte das Ändern und Widerrufen von Entscheidungen leicht möglich sein. Die Anforderung, effizient Alternativen erstellen zu können, stellt also wiederum Anforderungen an die Navigationsmöglichkeiten im Arbeitsprozess. Bei aktuellen Konfiguratoren steht das Erarbeiten einer korrekten Lösung im Vordergrund. Das effiziente Erarbeiten von Alternativen und der Vergleich von Produkten wird meist nicht unterstützt. Dies ist aber wesentlich, um den Benutzer bei der Auswahl eines komplexen Produktes zu unterstützen. Hier sehen auch Pu und Faltings einen Mangel aktueller Konfiguratoren (Pu und Faltings 2000).

Eine zentrale Anforderung, die wissensbasierte Systeme grundsätzlich erfüllen sollen, ist nach Görz (1993, S.730) die **Nachvollziehbarkeit der Lösung**. Der Benutzer sollte sowohl die Lösung als auch den Lösungsweg nachvollziehen können. Wählt der Vertriebsmitarbeiter im Kundengespräch ein Produkt aus, so sollte auch der Kunde die Lösung nachvollziehen können. Hierbei kann der Vertriebsmitarbeiter den Kunden unterstützen. Dies ist nicht der Fall, wenn der Kunde das Produkt allein wählt. In diesem Fall muss er sich (Teil-)Lösungen mit Hilfe des Systems selbständig erklären können. Der klassische Ansatz, um die Nachvollziehbarkeit von Lösungen in wissensbasierten Systemen zu gewährleisten, ist die Erklärungskomponente, der die Dialogpartnerperspektive zugrunde liegt (zu Perspektiven siehe Abschnitt 2.1.1 auf Seite 11ff). Durch das Bild vom System als einem gleichberechtigten Gegenüber wird die Tatsache verschleiert, dass nur ein Mensch die Verantwortung für die Lösung übernehmen kann. Daher sollten dem Benutzer **Explorationswerkzeuge** angeboten werden, mit deren Hilfe er selbständig (Teil-)Lösungen nachvollziehen kann. Dies bedeutet allerdings auch einen erhöhten Aufwand für den Benutzer, wie Busch u. a. (1994, S. 50) feststellen.

Eine weitere zentrale Anforderung, die wissensbasierte Systeme erfüllen sollten, ist **sofortiges Feedback** zu den vom Benutzer getroffenen Entscheidungen. Es sollten alle bisher getroffenen Entscheidungen und die aktuelle (Teil-)Lösung angezeigt werden. Dies fordern auch Rahmer (1999) und Emde u. a. (1997). Darüber hinaus sollten für den Benutzer wichtige Zwischenergebnisse dargestellt werden und Rückmeldungen zur Korrektheit der aktuellen (Teil-)Lösung gegeben werden.

Meines Erachtens leistet sofortiges Feedback einen erheblichen Beitrag zur Nachvollziehbarkeit von (Teil-)Lösungen, da hierdurch die Auswirkungen von Entscheidungen deutlich werden. Durch die kontinuierliche Anzeige wichtiger



Ergebnisse wird auch das **Arbeitsgedächtnis des Benutzers entlastet**. Außerdem kann der Benutzer so seinen eigenen Fortschritt bei der Arbeit fortwährend nachvollziehen.

Um den Benutzer bei der Produktauswahl zu unterstützen, sollte auch erkennbar sein, welche Produkte nicht angeboten werden, wie Nielsen u.a. für die Auswahl fest definierter Produkte fordern (Nielsen und Tahir 2002). Für die Auswahl komplexer Produkte erfordert dies, dass für jede Entscheidung alle prinzipiell möglichen Alternativen angezeigt werden. Werden nur die Alternativen angezeigt, die aufgrund der aktuellen (Teil-)Lösung noch möglich sind, so ist für den Benutzer das **angebotene Produktspektrum** nicht ersichtlich. Der Benutzer sollte sich mit Hilfe der Benutzungsschnittstelle die folgenden Fragen beantworten können:

1. Welche Alternativen werden prinzipiell angeboten?
2. Welche Alternativen sind mit der aktuellen (Teil-)Lösung möglich?
3. Wieso kann ich Alternative X nicht wählen?
4. Was muss ich ändern, um X doch wählen zu können?

In Abschnitt 4.4 auf Seite 58ff wurden zwei Sichtweisen unterschieden, aus denen der Benutzer seine Anforderungen an das Produkt formulieren kann: **Produktsicht** und **Anwendungssicht**. Das Wissen von Benutzern über die einzelnen Komponenten beziehungsweise über die Anwendung der komplexen Produkte kann sich stark unterscheiden. Daher sollten dem Benutzer diese beiden Sichtweisen als Alternativen angeboten werden.

Wie bereits beschrieben, haben nicht alle Benutzer das notwendige Wissen, um komplexe Produkte ohne Unterstützung zusammenzustellen. Ein Beispiel hierfür ist die Benutzergruppe der Planer im Anwendungsbeispiel SVEA. Felix u. a. (2001) halten es für sinnvoll, die Produktauswahl von Benutzern mit geringem Domänenwissen als einen **Lernprozess** aufzufassen. Hierbei sollte das Ziel darin bestehen, diesen Lernprozess zu unterstützen. Auch Liang und Chen (2000) fordern, dem Benutzer bei der Produktauswahl weiterführende Informationen zu bieten. Ein Beispiel hierfür besteht bei der Auswahl komplexer Produkte darin, dem Benutzer den Aufbau eines mentalen Modells der Arbeitsaufgabe zu erleichtern. Auch die Produktauswahl aus Anwendungssicht ist ein Beispiel. Darüber hinaus sollten dem Benutzer zu jeder Entscheidung und den bestehenden Alternativen Informationen angeboten werden, die die Bedeutung der Entscheidung und ihre Auswirkungen beschreiben. Außerdem sollten Informationen zu den angebotenen Komponenten verfügbar sein.

Während die Benutzergruppe der Vertriebsmitarbeiter meist für den Einsatz eines Systems zur Produktauswahl geschult werden kann, ist dies bei Kunden

normalerweise nicht der Fall. Die Systembenutzung darf daher nur mit einem **minimalen Lernaufwand** verbunden sein. Da ein geringer Lernaufwand laut Hacker die Akzeptanz und Nützlichkeit von Systemen steigert, ist dies für alle Benutzergruppen eine wichtige Anforderung (Hacker 1994).

Dass die Auswahl komplexer Produkte sehr aufwändig ist, kann bei Kunden und insbesondere bei Erstbenutzern ein Problem darstellen. Führt die Produktauswahl nicht schnell zu erkennbaren Ergebnissen, sind sie eventuell nicht bereit, weitere Zeit in die Auswahl zu investieren. Daher sollten solchen Benutzern **schnell erste Arbeitsergebnisse** ermöglicht werden. Außerdem werden Kunden normalerweise nicht bereit sein, umfangreiche Berechnungen im Rahmen der Produktauswahl durchzuführen. Auch die Arbeit der Vertriebsmitarbeiter kann durch die **Entlastung von Routineberechnungen** wesentlich effizienter werden.

Wie bereits oben beschrieben sollte der Benutzer mit Hilfe des Systems die Korrektheit seiner Lösung überprüfen und sicherstellen können. Zusätzlich sollte aber auch die Möglichkeit bestehen, das System zu „überstimmen“ – die letzte Entscheidung über die Lösung sollte beim Benutzer liegen. Dies ermöglicht es dem Benutzer zum Beispiel, gezielt spezielle Komponenten einzusetzen oder aus der Lösung zu entfernen. Hat der Kunde mit einer bestimmten Komponente gute Erfahrungen gemacht, möchte er diese eventuell auch in weiteren komplexen Produkten einsetzen. Der Vertriebsmitarbeiter kann durch die Möglichkeit das System zu „überstimmen“ bei der Auswahl auch Gewinn und Provision der einzelnen Komponenten berücksichtigen. Benutzer sollten auch in der Lage sein, bewusst eine unvollständige oder inkonsistente Lösung zu erstellen. Dies bietet dem Kunden die Möglichkeit, eine bereits vorhandene Komponente einzusetzen. In diesem Fall möchte er zwar ein komplexes Produkt mit dieser Komponente zusammensetzen, die spezielle Komponente selbst aber nicht erwerben.

Die Bedeutung der oben beschriebenen allgemeinen Anforderungen ist im konkreten Anwendungsfall jeweils zu gewichten und gegebenenfalls um weitere Anforderungen zu erweitern. Im nächsten Abschnitt wird konkret auf Anforderungen im Anwendungsbeispiel SVEA eingegangen.

### 6.1.3 Anforderungen — SVEA

Im Folgenden werde ich kurz auf die Perspektiven eingehen, die ich im Rahmen des Entwurfes eingenommen habe (zu Perspektiven siehe Abschnitt 2.1.1). Anschließend werden die daraus resultierenden Anforderungen im Anwendungsbeispiel SVEA beschrieben. Die bereits in den vorherigen Abschnitten beschriebenen Anforderungen werden in diesem Abschnitt nicht wiederholt aufgegriffen.

## Systemvision

Ziel des Entwurfes ist es, sowohl Planern als auch Außendienstmitarbeitern ein *Werkzeug* zur Verfügung zu stellen, mit dem sie konsistente und vollständige GA-Lösungen (komplexe Produkte) zusammenstellen können. Außendienstmitarbeiter sollten mit diesem Werkzeug im gemeinsamen Gespräch mit dem Planer Ausschreibungen erstellen können. Die Ergebnisse stehen dem Planer so unmittelbar zur Verfügung. Außerdem soll der Planer mit diesem Werkzeug auch selbständig Ausschreibungen erstellen können.

Überdies dient das System als *Medium* zur Kommunikation zwischen Vertriebsmitarbeitern und Planern, mit dem Ziel diese zu fördern. So wird zum Beispiel der Planer dabei unterstützt, Ausschreibungen zu erstellen, die alle vom Innendienst benötigten Informationen in übersichtlich strukturierter Weise enthalten. Planer und Vertriebsmitarbeiter können über dieses Medium leicht (Teil-)Lösungen sowie Kommentare und Fragen hierzu austauschen. Auch die Fertigstellung einer vom Planer begonnenen Lösung durch einen Vertriebsmitarbeiter ist denkbar.

Außerdem bietet das System Planern die Möglichkeit, während der Arbeit ihr Domänenwissen zu erweitern. Der Planer kann also die Erstellung einer Ausschreibung auch als *Lernprozess* nutzen.

Schließlich soll das System mit minimaler Einarbeitung sowohl von Vertriebsmitarbeitern als auch von Planern eingesetzt werden können. Daher beruht die Interaktion mit dem System auf der Fachsprache, die Vertriebsmitarbeiter und Planer teilen. Das konzeptuelle Modell des Systems baut in möglichst hohem Maß auf Wissen auf, das bereits bei allen Benutzern vorhanden ist.

Aus der oben beschriebenen Systemvision ergeben sich eine Menge von Anforderungen, die im Folgenden erläutert werden.

## Anforderungen

Planer sollen mit Hilfe des Systems selbständig vollständige und korrekte Ausschreibungen erstellen können. Aufgrund ihres meist geringen Domänenwissens im Bereich der LON-basierten GA-Lösungen können Planer die **Produktauswahl ausschließlich aus Anwendungssicht** durchführen. Die in Abschnitt 4.4 vorgestellten Sichtweisen bei der Produktauswahl – Produktsicht und Anwendungssicht – können im Anwendungsbeispiel SVEA wie folgt veranschaulicht werden: Aus *Produktsicht* trifft der Benutzer eine Entscheidung zwischen unterschiedlichen Geräten oder Gerätekategorien. So entscheidet er sich für den Einsatz eines Digital-Ausgangs, eines Steuerausgangs oder eines Dimmer-Ausgangs. Aus *Anwendungssicht* legt der Benutzer dagegen fest, ob die Beleuchtung geschaltet oder gedimmt werden soll und welche Art von Leuchten eingesetzt werden.

In Abschnitt 6.1.2 wurde die Anforderung aufgestellt, dass auch der Kunde die vom Vertriebsmitarbeiter im Kundengespräch erstellte Lösung nachvollziehen können sollte. Daher sollte auch der Vertriebsmitarbeiter die Produktauswahl aus Anwendungssicht durchführen.

Auf der Grundlage der vom Benutzer erstellten GA-Lösung sollte das System als **Arbeitsergebnisse** eine **Kostenschätzung**, ein **Massenleistungsverzeichnis** und eine **funktionale Ausschreibung** zur Verfügung stellen. Beispiele für diese Arbeitsergebnisse werden auf den Seiten 70, 71 und 169 gegeben. Die funktionale Ausschreibung kann als eine textuelle Beschreibung der erstellten GA-Lösung aus Anwendungssicht bezeichnet werden, während das Massenleistungsverzeichnis eine Produktsicht auf die Lösung bietet.

Planer können die Korrektheit von GA-Lösungen aufgrund ihres geringen Domänenwissens meist nicht beurteilen. Dies gilt auch für die momentan vom Außendienst erstellten GA-Lösungen. Durch die **Verbindung von funktionaler Ausschreibung und Massenleistungsverzeichnis** (Verbindung von Produktsicht und Anwendungssicht) ist es jedoch den Vertriebsmitarbeitern möglich, die vom Planer erstellte Ausschreibung zu überprüfen. Dies ist allein auf der Grundlage eines Massenleistungsverzeichnisses nicht möglich. Daher sollten die erstellten Ausschreibungen jeweils beide Arbeitsergebnisse umfassen.

Die Erstellung von Angeboten im Innendienst ist vielfach mit Rückfragen an Planer verbunden, da Ausschreibungen oft nicht die benötigten Angaben enthalten. Das System sollte den Benutzer darin unterstützen, Ausschreibungen zu erstellen, die alle vom Innendienst benötigten **Informationen in übersichtlich strukturierter Form** enthalten. Auch hierfür sollten Ausschreibungen jeweils ein Massenleistungsverzeichnis und eine funktionale Ausschreibung umfassen. Die Informationen sollten entsprechend der Struktur der Arbeitsaufgabe gegliedert sein (siehe Seite 75), weil hierdurch Lesbarkeit und Übersichtlichkeit verbessert werden.

Zusätzlich sollte das System Planer dabei unterstützen, ihr **Domänenwissen** zu **erweitern**. Auch hierfür ist die Verbindung von funktionaler Ausschreibung und Massenleistungsverzeichnis sinnvoll. So können Planer Zusammenhänge zwischen Produktsicht und Anwendungssicht erkennen. Nicht nur die Arbeitsergebnisse sollten Produktsicht und Anwendungssicht verbinden, sondern auch die Benutzungsschnittstelle sollte beide Sichtweisen auf die Lösung bieten. Außerdem sollten sich Planer mit dem System einfach einen Überblick über mögliche LON-basierte GA-Lösungen verschaffen können. Da Planer oft keine Vorstellung von LON-Geräten besitzen, sollten auch Informationen zu den einzelnen Geräte angeboten werden.

Die interviewten Vertriebsmitarbeiter und Planer haben das Erstellen einer GA-Lösung alle als eine **Konstruktion oder Zusammenstellung eines Systems aus Geräten** beschrieben. Das konzeptuelle Modell des Systems sollte diese Vor-

stellung aufgreifen. Außerdem sollten nur **Fachbegriffe** verwendet werden, die sowohl Vertriebsmitarbeitern als auch Planern bereits bekannt sind.

Im Gegensatz zu Vertriebsmitarbeitern erstellen Planer Lösungen meist nicht neu, sondern ändern bereits bestehende Lösungen. Um diese Arbeitsweise auch weiterhin zu ermöglichen, sollten dem Benutzer eine Menge von **typischen GA-Lösungen als Beispiele** zur Verfügung gestellt werden. Diese sollten vom Benutzer durch Änderungen leicht an die bestehenden Anforderungen angepasst werden können. Das einfache und effiziente Ändern von (Teil-)Lösungen wurde bereits in Abschnitt 6.1.1 als Anforderung genannt und bekommt so zusätzliche Bedeutung.

Von sehr hoher Bedeutung ist die **Zufriedenheit der Planer** mit dem System. Planer werden das System nur einsetzen, wenn sie mit dessen Einfachheit, Effizienz und den erzeugten Arbeitsergebnissen zufrieden sind und / oder Spaß an der Arbeit mit dem System haben. Ansonsten können sie sich zur Erstellung von Ausschreibungen auch weiterhin an die Hersteller von Geräten zur Gebäudeautomation wenden.

Für den Austausch von Daten im Rahmen von Bauvorhaben existiert ein Standard, das **GAEB-Format**<sup>1</sup>. Dies umfasst eine Reihe von Dateiformaten, die den einzelnen Datenaustauschphasen im Rahmen von Bauvorhaben zugeordnet sind. Damit Planer Kostenschätzung und Massenleistung unmittelbar mit der von ihnen eingesetzten AVA-Software<sup>2</sup> weiterverarbeiten können, sollten diese bereits GAEB-Format haben. Hierdurch kann für Planer eine deutliche Zeitersparnis erreicht werden.

Das vom Planer erstellte Massenleistungsverzeichnis umfasst normalerweise nicht nur die Ausschreibung der Gebäudeautomation. Um für das gesamte Massenleistungsverzeichnis auf einfache Weise ein einheitliches Layout zu ermöglichen, sollte der Benutzer die **gewünschte Formatierung** angeben können.

Eine Erweiterung des Systems könnte die Arbeit des Systemintegrators unterstützen (siehe hierzu Seite 67). Der Systemintegrator parametriert die Software für die einzelnen Geräte der GA-Lösung. Aufbauend auf der vom Planer erstellten GA-Lösung könnte der Systemintegrator die notwendige Software mit Hilfe des Systems parametrieren. Auf diesen Aspekt wird im Folgenden nicht weiter eingegangen.

---

<sup>1</sup>Festgelegt vom **Gemeinsamen Ausschuss Elektronik im Bauwesen**. (<http://www.gaeb.de>; zuletzt besucht: 02.05.2003)

<sup>2</sup>Ausschreibung, Vergabe, Abrechnung

## 6.2 Überblick über die Unterstützungsansätze

Eine klare Unterscheidung von Ansätzen, die die Orientierung unterstützen und solchen Ansätzen, die der Navigation dienen, ist nicht möglich. Dies liegt an der engen Beziehung zwischen Orientierung und Navigation. Navigation ist nur auf der Grundlage von Orientierung möglich. Durch die Navigation werden dann wiederum neue Umgebungsinformationen wahrgenommen, die für die Orientierung hilfreich sind. Trotz dieser engen Beziehung überwiegt bei den erarbeiteten Unterstützungsansätzen jedoch meist einer der beiden Aspekte. Im Folgenden werden zunächst kurz die Ansätze vorgestellt, die die Orientierung fördern. Dann folgt ein Überblick über die Ansätze zur Unterstützung der Navigation.

Damit der Benutzer sich im Arbeitsprozess orientieren kann, muss er sich eine *Vorstellung von der Arbeitsaufgabe* machen können. Mit Hilfe der Raum-Metapher kann die Beziehung von einzelnen Entscheidungen zur Gesamtaufgabe vermittelt werden. So kann der Benutzer ein mentales Modell der Arbeitsaufgabe aufbauen.

Wie in Kapitel 4 beschrieben, ist die große Anzahl der Entscheidungen ein Grund für die Komplexität der Arbeitsaufgabe. Indem der Benutzer die Produktauswahl mit der Auswahl einer Teillösung – einem Schema – beginnt, kann die Anzahl der weiteren Entscheidungen reduziert werden. So ist es leichter, einen Überblick über die Aufgabe zu bekommen.

Durch eine Interaktionshistorie oder einen Interaktionsgraphen kann der Benutzer den bereits *zurückgelegten Weg erforschen*. Zusätzlich können Interaktionshistorie und -graph als Basis für bestimmte Undo-Mechanismen und Wiederaufsetzpunkte dienen.

Wenn der Benutzer eine Vorstellung von der Arbeitsaufgabe hat und seinen eigenen Standort im Arbeitsprozess kennt, kann er seinen Fortschritt selbst einschätzen. Personen ohne ein fundiertes mentales Modell von der Arbeitsaufgabe können dies hingegen nicht selbständig. Mit einem Überblick über die Aufgabe kann der *Fortschritt bei der Arbeit vermittelt* werden.

Mit Hilfe einer Szenario-Maschine oder einer animation machine können Anfänger die Produktauswahl mit dem System kennenlernen. Sie bieten die Möglichkeit, Anfängern *typische Wege zu vermitteln*, Vorgehensweisen, die sich als günstig herausgestellt haben.

Beim Entwurf eines Konfigurators wird festgelegt, welche Wege durch die Aufgabe für den Benutzer zukünftig möglich sind – die Navigationsmöglichkeiten werden definiert. Mit einem *festen Weg* wird dem Benutzer ein Vorgehen vorgeschrieben, dass sich in vielen Fällen als sinnvoll erwiesen hat. Das *Leitsystem* und der *freie Weg* ermöglichen dagegen ein flexibles Vorgehen.

Für Experten ist eine effiziente Produktauswahl von großer Bedeutung. Durch

*Abkürzungen* können Experten häufig wiederkehrende Teilaufgaben schneller erledigen. Für Anfänger sind dagegen solche Abkürzungen von besonderer Bedeutung, die die Arbeit vereinfachen.

Während Anfänger ein kleinschrittiges Vorgehen benötigen, steht für Experten ein effizientes Arbeiten im Vordergrund. Da Systemrückmeldungen sehr aufwändige Rechengänge erfordern können (z.B. Constraint-Propagation), kann es für Experten sinnvoll sein, nach einzelnen Schritten auf Systemrückmeldungen zu verzichten. Hier stellen Anfänger und Experten also gegenläufige Anforderungen an die *Schrittgrößen* bei der Arbeit. Benutzerdefinierte Konfigurierungseinheiten sind ein Ansatz, um diese Anforderungen zu erfüllen.

Die Möglichkeit, Entscheidungen durch ein *Undo* zu stornieren oder *direkt zu ändern*, unterstützen ein flexibles Vorgehen bei der Produktauswahl. Es werden verschiedene Undo-Mechanismen vorgestellt und bezüglich ihres Einsatzes in der Produktauswahl bewertet. Es wird begründet, warum zusätzlich zu einer Undo-Funktion auch das direkte Ändern von Entscheidungen angeboten werden sollte.

Durch *Wiederaufsetzpunkte* können Benutzer im Arbeitsprozess vor und zurück springen und alternative Lösungen erforschen. So können sie sich selbständig von der Qualität einer Produktauswahl überzeugen.

Ein exploratives Arbeiten wird vor allem unterstützt durch die Erforschbarkeit des zurückgelegten Weges, die Sichtbarkeit aller Entscheidungen, eine Undo-Funktion, die Möglichkeit Entscheidungen direkt zu ändern, Wiederaufsetzpunkte und ein flexibles Vorgehen. Exploratives Arbeiten erfüllt erstens die Anforderungen der Arbeitsaufgabe Produktauswahl und zweitens fördert es den Aufbau mentaler Modelle (Dutke 1994; Hacker 1994). Deshalb ist diese Arbeitsform geeignet, um Benutzer bei der Orientierung zu unterstützen. Schließlich hilft die Exploration Benutzern bei der Klärung von Fragen und bietet ihnen so die Möglichkeit, ihr Domänenwissen zu erweitern (Busch u. a. 1994, S. 172ff). Daher kommt den oben genannten Unterstützungsansätzen eine besondere Bedeutung zu.

Abbildung 6.1 auf Seite 88 gibt einen grafischen Überblick über die Unterstützungsansätze in Abschnitt 6.4 und 6.5. In der Software-Ergonomie werden unter anderem die *direkte Manipulation* und *Formulare* als Interaktionstechniken unterschieden. Auch für die Produktauswahl kann zwischen diesen Techniken gewählt werden. Die Entscheidung zwischen diesen Interaktionstechniken hat einen erheblichen Einfluss darauf, welche Auswahl von Unterstützungsansätzen sinnvoll ist. So erfordert die direkte Manipulation zum Beispiel einen freien Weg durch die Aufgabe und das direkte Ändern von Entscheidungen. Daher kann die Wahl zwischen diesen Techniken als Basis für eine Zusammenstellung von Unterstützungsansätzen gesehen werden. Andererseits kann die Entscheidung für eine der Techniken auch als die Auswahl eines ersten Unterstützungsansatzes betrachtet werden. Während die direkte Manipulation den Benutzer vor allem bei der Orientierung unterstützt, kann mit Formularen eher ein erfolgversprechendes Vorgehen

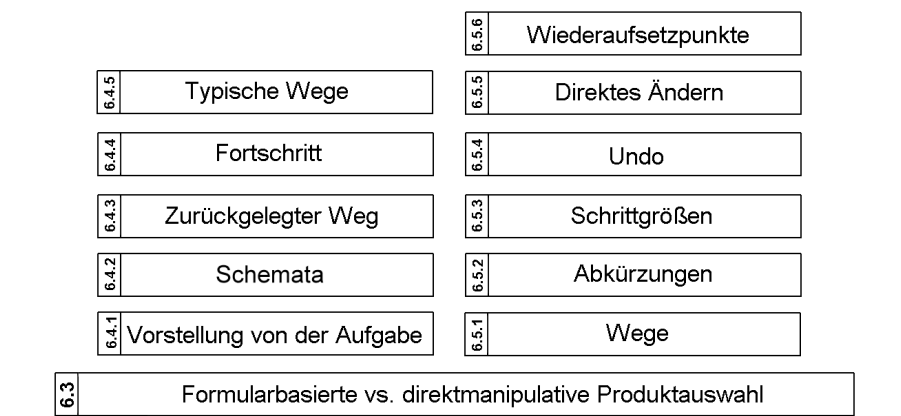


Abbildung 6.1: Überblick über die Unterstützungsansätze in Abschnitt 6.4 und 6.5

vermittelt werden. Da die Wahl zwischen diesen Interaktionstechniken eine übergeordnete Rolle spielt, wird hierauf im nächsten Abschnitt eingegangen, bevor anschließend die weiteren Unterstützungsansätze behandelt werden.

Für die erarbeiteten Unterstützungsansätze werden jeweils Beispiele genannt sowie Vor- und Nachteile beschrieben. Außerdem wird darauf eingegangen, für welche Benutzergruppen die Unterstützungsansätze besonders geeignet sind. Die hier beschriebenen Ansätze verstehen sich als ein Baukasten von Methoden und Mechanismen zur Unterstützung von Orientierung und Navigation. Eine allgemeingültige Auswahl und Zusammenstellung dieser Methoden kann nicht beschrieben werden. Sie muss immer in Abhängigkeit von Benutzergruppen, der konkreten Arbeitsaufgabe und der Domäne getroffen werden. Die Bewertung der einzelnen Ansätze dient hierbei als Hilfe.



### 6.3 Formularbasierte oder direktmanipulative Produktauswahl

Die bei Konfiguratoren vorherrschende Interaktionstechnik sind Formulare. Über sie werden die notwendigen Entscheidungen in textueller Form festgelegt. Eine auf der direkten Manipulation beruhende Benutzungsoberfläche würde stattdessen die wesentlichen Objekte der Anwendungsdomäne darstellen, und der Benutzer würde aus ihnen das komplexe Produkt direkt zusammenstellen. Anstatt zum Beispiel die Anzahl der Beleuchtungsgruppen in einem Feld anzugeben, fügt er zwei entsprechende Objekte in die Lösung ein. Die direkte Manipulation kann sowohl in Verbindung mit der *Produktsicht* als auch in Verbindung mit der *Anwendungssicht* eingesetzt werden. Abbildung 6.2 skizziert den Einsatz der direkten Manipulation für das Anwendungsbeispiel SVEA. Der Planer kann hier aus Anwendungssicht die LON-basierte GA-Lösung zusammenstellen. Im Anwen-

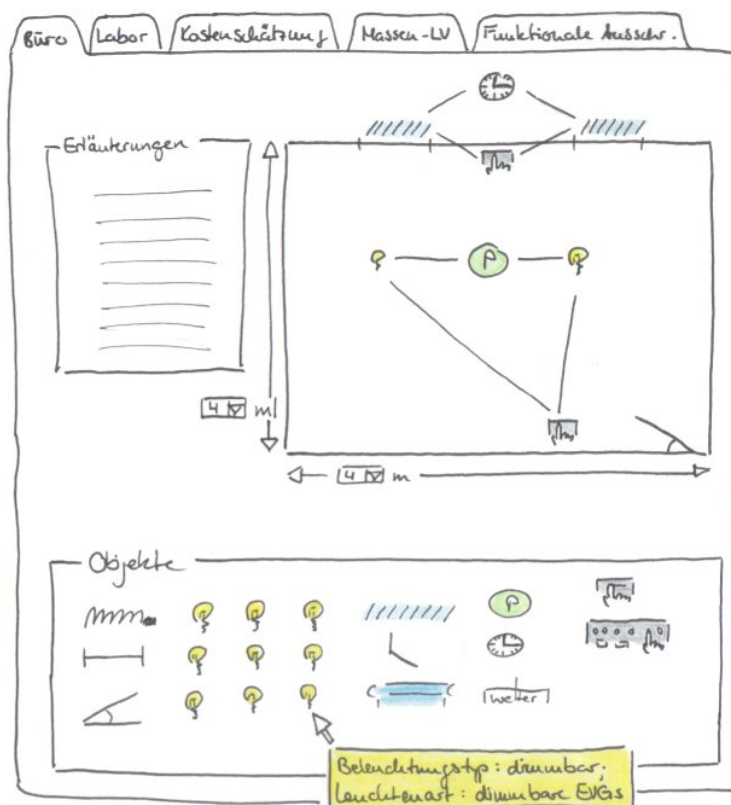


Abbildung 6.2: Direktmanipulative Oberfläche zur Zusammenstellung eines komplexen Produktes aus Anwendungssicht

dungsfall SVEA erscheint der Einsatz der direkten Manipulation naheliegend, da alle befragten Benutzer die Arbeitsaufgabe als ein *Zusammenfügen eines Systems aus Komponenten* beschrieben haben. Diese Vorstellung von der Produktauswahl wird durch die direkte Manipulation viel stärker aufgegriffen, als es in der formularbasierten Interaktion möglich ist. Auch unabhängig vom Anwendungsfall scheint die direkte Manipulation ein sinnvoller Ansatz zu sein, um Benutzer bei der Orientierung zu unterstützen. So fördert die direkte Manipulation den Aufbau mentaler Modelle und unterstützt so Benutzer bei der Orientierung (Maaß 1994). Darüber hinaus bieten direktmanipulative Oberflächen Antworten auf die Nievergeltschen Fragen (siehe Abschnitt 2.2.1) und ermöglichen durch ständiges grafisches Feedback eine visuelle Erfolgskontrolle (Oberquelle 1994, S. 134). Da direktmanipulative Oberflächen außerdem ein exploratives Arbeiten fördern, kann mit ihnen gut eine flexible Navigation in der Arbeitsaufgabe realisiert werden. Shneiderman (1998, S. 72) beschreibt, dass die direkte Manipulation die Zufriedenheit der Benutzer steigert.

Betrachtet man die Produktauswahl komplexer Produkte, so stehen diesen Vorteilen jedoch auch bedeutende Nachteile gegenüber. Wie bereits im letzten Abschnitt erwähnt, ist die direkte Manipulation immer mit einem freien Weg durch die Aufgabe verbunden. Einerseits ermöglicht ein freier Weg das explorative Arbeiten, andererseits vermittelt er aber keine günstigen Vorgehensweisen. Da auf den freien Weg in Abschnitt 6.5.1 näher eingegangen wird, sollen hier nur einige seiner Nachteile vorweggenommen werden. So kann es passieren, dass Benutzer mit geringem Domänenwissen durch ungünstige Vorgehensweisen nur unbefriedigende Lösungen erarbeiten oder dass häufig Konflikte auftreten.

Ein weiteres Problem kann durch die grafische Darstellung der Objekte entstehen, ein zentrales Merkmal der direkten Manipulation. Werden die komplexen Produkte zum Beispiel aus Produktsicht zusammengestellt, so sind diese Objekte die einzelnen Komponenten. Die Menge der Objekte kann also je nach Anwendungsdomäne äußerst groß sein. Für diese Objekte leicht erkennbare und unterscheidbare Repräsentationen zu finden, ist dann gar nicht oder nur schwer möglich. Da komplexe Produkte vor der Einführung eines Konfigurators auf der Basis von Listen und Katalogen zusammengestellt werden, sind den Benutzern eher Bezeichnungen für die Komponenten bekannt als grafische Repräsentationen. Dies gilt vor allem für die Benutzergruppe der Vertriebsmitarbeiter, die mit den Produktbezeichnungen normalerweise gut vertraut sind. Grafische Repräsentationen für Komponenten müssten dagegen zunächst von den Benutzern erlernt werden. Aus diesen Gründen kann eine vollständig direktmanipulative Oberfläche schwer oder gar nicht realisierbar sein.

Um trotz einer großen Menge von Objekten die Vorteile der direkten Manipulation zu nutzen, kann die Interaktionstechnik mit Formularen kombiniert werden. Durch eine solche Kombination kann auf die Darstellung aller Komponenten ver-

zichtet werden. Stattdessen werden ganze Klassen von Komponenten durch ein gemeinsames Objekt in der Benutzungsschnittstelle repräsentiert. Die Objekte werden also auf einer höheren Abstraktionsebene angesiedelt, wie Abbildung 6.3 anhand einer Taxonomie veranschaulicht. Während in Abbildung 6.3 a) alle Objekte auf der Blattebene der Taxonomie liegen, sind die Objekte in Abbildung 6.3 b) auf höheren Ebenen angesiedelt.

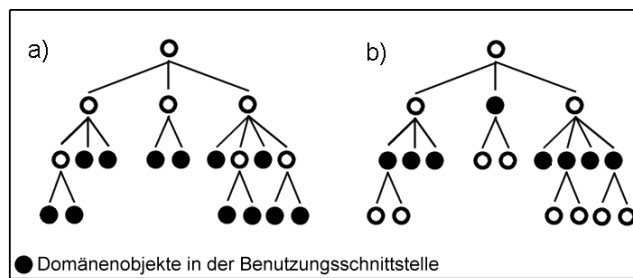


Abbildung 6.3: Ansiedlung der Objekte auf verschiedenen Ebenen der Taxonomie

Im Anwendungsfall SVEA könnten beispielsweise alle Beleuchtungsgruppen durch nur ein Objekt dargestellt werden. Die genaueren Anforderungen an das Objekt Beleuchtungsgruppe spezifiziert der Benutzer dann über ein Formular, wie in Abbildung 6.4 skizziert. Durch diese Verbindung der Interaktionstechniken

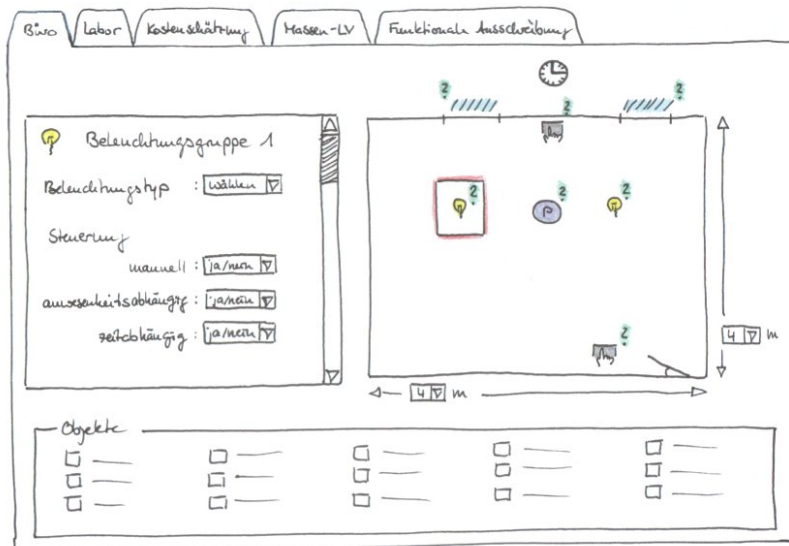


Abbildung 6.4: Verbindung von Formular und direkter Manipulation

muss der Benutzer nur eine geringe Anzahl von grafischen Repräsentationen erlernen. Die ständige Darstellung der aktuellen Lösung unterstützt die Orientierung.

Da die direkte Manipulation, eher als Formulare, zum „Herumspielen“ mit dem System anregt, kann sie eingesetzt werden, um Benutzer zum Ausprobieren des Systems anzuspornen. Außerdem kann sich der Benutzer gut bei der Arbeit mit dem System vertraut machen. Mayhew und Shneiderman halten die direkte Manipulation besonders für Anfänger und für gelegentliche Benutzer geeignet (Mayhew 1992, S. 365ff), (Shneiderman 1998, S. 73). Handelt es sich um Anfänger bezüglich des Anwendungssystems gilt dies auch bei der Auswahl komplexer Produkte. Für Anfänger bezüglich der Anwendungsdomäne gilt dies jedoch nur bedingt. Sind spezielle Vorgehensweisen notwendig, um eine gute Lösung zu erarbeiten, können diese durch einen festen Weg (siehe Abschnitt 6.5.1) vermittelt werden. Ein fester Weg setzt jedoch eine formularbasierte Oberfläche voraus. Für besonders geeignet halte ich die direkte Manipulation oder die oben beschriebene Verbindung der Interaktionstechniken, wenn Vertriebsmitarbeiter und Kunden die Produktauswahl gemeinsam durchführen. Hier kann der Vertriebsmitarbeiter sein Wissen über Vorgehensweisen einbringen, und durch die grafische Darstellung der (Teil-)Lösung kann der Kunde gut in die Auswahl einbezogen werden. Steht vor allem die Effizienz im Vordergrund, ist eine rein formularbasierte Produktauswahl geeigneter. Sie ermöglicht geübten Tastaturbenutzern eine schnellere Produktzusammenstellung.

## 6.4 Ansätze zur Unterstützung der Orientierung

### 6.4.1 Vorstellung von der Arbeitsaufgabe

Eine flexible Navigation durch die Arbeitsaufgabe ist für den Benutzer nur möglich, wenn er ein mentales Modell von der Aufgabe hat. Er kann zum Beispiel nur dann eine bestimmte Entscheidung gezielt treffen, wenn er weiß, dass diese Entscheidung prinzipiell möglich ist, dass diese Entscheidung auch im System getroffen werden kann und wo dies möglich ist. Hat der Benutzer eine Vorstellung von der Aufgabe, so kann er auch seinen Fortschritt selbständig einschätzen.

Wie leicht der Benutzer sich eine solche Vorstellung von der Arbeitsaufgabe verschaffen kann, wird auch durch die Wahl zwischen einer direktmanipulativen und einer formularbasierten Oberfläche bestimmt. Bei einer direktmanipulativen Oberfläche kann sich der Benutzer leicht ein Bild davon machen, wie die Produktauswahl mit dem System aussieht. Bei der formularbasierten Produktauswahl ist dies nicht so einfach möglich. Daher werden im Folgenden Unterstützungsansätze vorgestellt, die dem Benutzer vor allem in der formularbasierten Produktauswahl dabei helfen, sich eine Vorstellung von der Aufgabe zu verschaffen. Zuerst wird der Einsatz der *Raum-Metapher* erläutert, anschließend wird auf das Thema *Sichtbarkeit von Entscheidungen* eingegangen.

#### Raum-Metapher

Bei vielen formularbasierten Konfiguratoren besteht die Produktauswahl aus einer Folge von Formularen, über die der Benutzer die Entscheidungen trifft. Eine solche Folge ist in Abbildung 6.5 a) skizziert. Aus folgenden Gründen ist es dem

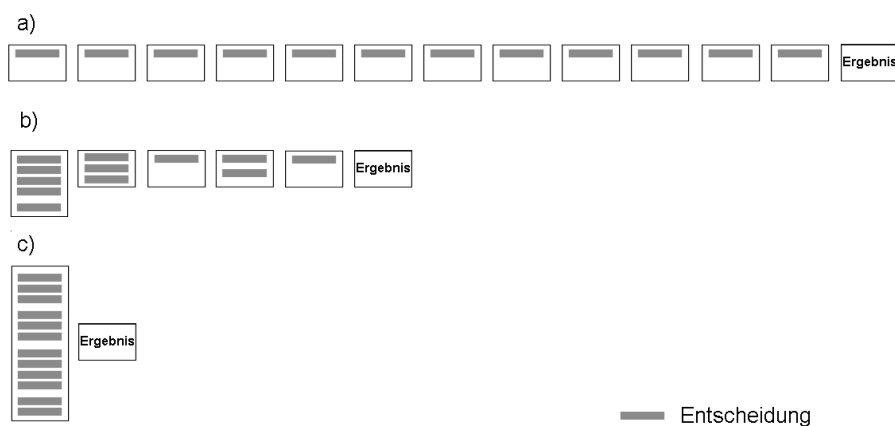


Abbildung 6.5: Gruppierung von Entscheidungen

Benutzer kaum möglich, sich ausschließlich auf der Grundlage einer Folge von Entscheidungen eine Vorstellung von der Aufgabe zu verschaffen: Erstens ist es schwierig, sich eine große Menge von Entscheidungen zu merken. Zweitens kann der Benutzer die Beziehung einzelner Entscheidungen zur Gesamtaufgabe nicht erkennen. Schließlich kann jeder Arbeitsdurchlauf aus einer anderen Menge von Entscheidungen bestehen. Dadurch ist es auch bei wiederholter Produktauswahl schwierig, einen Überblick zu gewinnen. Eine solche reine Folge von Formularen kann verglichen werden mit einem Labyrinth, in dem keine Kreuzungen vorhanden sind. Man kann sich nicht verlaufen und kommt sicher ans Ziel. Aufgrund der fehlenden Umgebungsinformation ist es aber unmöglich, sich ein Bild vom zurückgelegten Weg zu machen. Lange Wege auf denen der Benutzer seinen Fortschritt nicht erkennen kann, können Frustration auslösen. Fleming spricht in diesem Zusammenhang von dem „Are We There Yet?“-Syndrom (Fleming 1998, S.19ff).

Passini beschreibt, dass komplexe Umgebungen einen *Strukturierungsprozess* erfordern, um vom Benutzer erfassbar zu sein (Passini 1984, S. 186). Meines Erachtens kann diese Aussage auch auf die Gestaltung von Benutzungsschnittstellen übertragen werden. Kann der Benutzer z.B. die einzelnen Entscheidungen in eine Struktur „einsortieren“ und deren Aufbau verstehen, so kann er sich eher eine Vorstellung vom Ganzen machen. Eine solche Strukturierung der „Umgebung“ ist in Abbildung 6.5 b) auf Seite 93 angedeutet. Die in Abschnitt 2.2.1 beschriebene Raum-Metapher kann meines Erachtens gut eingesetzt werden, um eine solche Struktur zu realisieren und zu vermitteln. Im Folgenden wird erläutert, wie die Raum-Metapher eingesetzt werden kann, um Benutzer bei der Orientierung in der formularbasierten Produktauswahl zu unterstützen. Beim Einsatz der Raum-Metapher benötigt der Benutzer Wissen über Orte und Routen sowie Überblickswissen (siehe Abschnitt 2.2).

**Orte.** Im Programm können Orte zum Beispiel durch unterschiedliche Fenster oder Reiter realisiert werden. Eine Gruppe von Entscheidungen, die in der formularbasierten Produktauswahl an einem Ort angezeigt wird, bezeichne ich im Folgenden als *Navigationseinheit*. Durch eine solche Gruppierung von Entscheidungen kann deren Zusammengehörigkeit ausgedrückt werden. Kann der Benutzer die Gruppierung nachvollziehen, so kann er sich mit ihrer Hilfe leichter eine Vorstellung von der Aufgabe machen. Haben die Benutzer bereits eine Vorstellung von der Aufgabe, sollte diese in jedem Fall bei der Festlegung der Navigationseinheiten aufgegriffen werden, damit Benutzer an bereits Bekanntes anknüpfen können, wodurch der Aufbau eines mentalen Modells erleichtert wird. Dass eine solche benutzerorientierte Gruppierung von enormer Bedeutung ist, wird auch durch eine Studie gestützt, deren Ergebnisse Nielsen (2000, S. 198f) beschreibt. Nielsen untersuchte die Auswirkungen einer herstellerorientierten gegenüber einer benutzerorientierten Navigationsstruktur auf die Erfolgsrate bei der Auswahl

von fest definierten Produkten. Sein Vergleich ergab eine Erfolgsrate von 80% bei einer Struktur, die dem mentalen Modell der meisten Benutzer entsprach. Bei einer Struktur, die nach dem mentalen Modell der Hersteller gestaltet war, sank die Erfolgsrate dagegen auf nur 9%.

Im Anwendungsfall SVEA habe ich in Zusammenarbeit mit den Vertriebsmitarbeitern die Struktur der Arbeitsaufgabe erarbeitet. Sie beschreibt, aus welchen Teilaufgaben die Zusammenstellung eines GA-Systems nach Auffassung der Vertriebsmitarbeiter besteht. Diese Struktur wird in Abbildung 5.8 (siehe Seite 75) gezeigt. Auf der Basis dieser Struktur habe ich die Navigationseinheiten im Prototyp entworfen. Eine Navigationseinheit umfasst also jeweils die Entscheidungen, die zu einer Teilaufgabe gehören.

Die Navigationseinheiten eines Systems können verschieden groß sein. Vereinfacht können drei Fälle unterschieden werden:

- Jede Entscheidung bildet eine eigene Navigationseinheit.
- Eine bis mehrere Entscheidung(en) bilden eine Navigationseinheit.
- Alle Entscheidungen liegen in einer Navigationseinheit.

Diese Fälle wurden einander schon in Abbildung 6.5 auf Seite 93 gegenübergestellt. Navigationseinheiten aus mehr als einer Entscheidung erleichtern nicht nur den Überblick über die Aufgabe, sondern bieten auch die Möglichkeit die Auswirkungen von Entscheidungen zu verfolgen. Außerdem wird es leichter, Entscheidungen in Abhängigkeit von anderen Entscheidungen zu treffen. Bei angemessener Größe der Navigationseinheiten ist auch weniger Navigationsinteraktion notwendig. Diese Vorteile gehen allerdings durch unüberschaubar große Navigationseinheiten wieder verloren.

Damit der Benutzer durch die Orte bei der Orientierung unterstützt wird, muss er diese wiedererkennen. Orte sollten daher nicht identisch gestaltet sein, sondern sich durch Orientierungspunkte unterscheiden.

„In the same way that sailors navigate by reference to shorelines or stars, users navigate by reference to permanent objects placed in the program’s user interface.“ (Cooper 1995, S. 508)

Beispiele für solche Orientierungspunkte sind Symbole, Bilder oder andere Erkennungszeichen. Die einzelnen Orte können auch durch verschiedene Farben gekennzeichnet werden. Überdies ist die Anzeige von Namen für einzelne Orte eine Unterstützungsmöglichkeit, wie Abbildung 6.6 auf Seite 96 zeigt. Dies entspricht den Identifikationsschildern, die in Gebäuden und Städten verwendet werden. Damit der Benutzer die Orientierungspunkte nutzen kann, dürfen sie nicht modifiziert oder an unterschiedlicher Stelle angezeigt werden – ein konsistenter Einsatz

der Orientierungspunkte ist also wichtig. Weitere Möglichkeiten, das Wiedererkennen von Orten zu unterstützen, entstehen durch den Einsatz eines Überblicks. Hierauf wird weiter unten eingegangen.

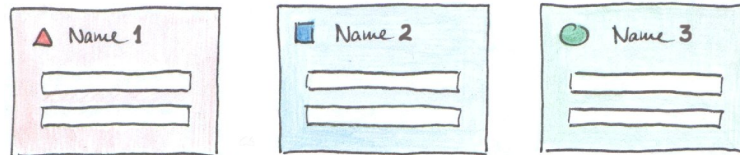


Abbildung 6.6: Wiedererkennen von Orten durch visuelle Merkmale

**Routen.** Verbindungen zwischen den Orten (Navigationseinheiten) des Systems werden als Routen bezeichnet. Die Routen im System sollten die möglichen Wege durch die Arbeitsaufgabe klar vermitteln. Mögliche Arbeitsabläufe und ihre Vor- und Nachteile werden in Abschnitt 6.5.1 beleuchtet. Hat der Benutzer zum Beispiel einen *festen Weg* durch die Arbeitsaufgabe gewählt, so benötigt er andere Routen, als wenn er sich für die Unterstützung durch ein *Leitsystem* entschieden hat.

Bei der in Abschnitt 6.5.1 als *Leitsystem* beschriebenen Alternative sollte der Benutzer eine Route angeboten bekommen, mit der er die Entscheidungen in der festgelegten Reihenfolge treffen kann. Außerdem sollte er aber auch beliebige Navigationseinheiten direkt erreichen können. Hierdurch kann er einfach von der festgelegten Reihenfolge abweichen und sich jederzeit einen Überblick über die Arbeitsaufgabe verschaffen.

Ist dagegen kein Leitsystem vorgesehen (*freier Weg*), so ist es auch nicht sinnvoll, eine bestimmte Route zwischen den Navigationseinheiten hervorzuheben. Stattdessen sollte der Benutzer von jedem Ort aus alle anderen Orte schnell erreichen können. Hierfür bietet die Navigation über den Überblick eine sinnvolle Möglichkeit. Dies wird im nächsten Abschnitt kurz behandelt.

Ist schließlich eine bestimmte Reihenfolge für die Entscheidungen festgelegt (*fester Weg*), so sollte dies auch durch eine vorgeschriebene Route zwischen den Navigationseinheiten verdeutlicht werden. Kann der Benutzer nur auf der bisher zurückgelegten Route hin- und herwandern, so wird klar vermittelt, wo die nächste Entscheidung möglich ist. Könnte sich der Benutzer dagegen beliebig zwischen den Navigationseinheiten hin- und herbewegen, so würde dies suggerieren, dass die Entscheidungen eben auch in beliebiger Reihenfolge getroffen werden können.

Damit sich Benutzer erfolgreich im System bewegen können, müssen mögliche Routen klar erkennbar sein. Dutke spricht in diesem Zusammenhang von „Wegweisern durch das System“ (Dutke 1994, S. 116). Beispiele für solche Weg-



weiser sind das Menü, Buttons, Links und Reiter, aber auch der im nächsten Abschnitt beschriebene Überblick. Abbildung 6.7 gibt ein Beispiel, wie mit Hilfe von Buttons ein festgelegter Weg erkennbar gemacht werden kann. Durch die

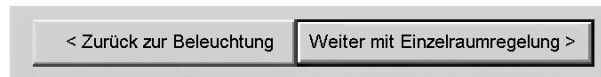


Abbildung 6.7: Weg über Button verdeutlichen

Kontextinformation (Weiter mit Einzelraumregelung >) wird das Ziel der Bewegung für den Benutzer erkennbar. Solche Buttons entsprechen Richtungsschildern in Gebäuden oder Städten. Neben diesen bestehen im System jedoch noch weitere Möglichkeiten, um den Benutzer auf Routen aufmerksam zu machen. Ein Beispiel hierfür ist der von Benest und Potok beschriebene *Eye Mover*; ein animierter Pfeil, der die Aufmerksamkeit des Benutzers auf den Menüpunkt lenken soll, der als nächstes besonders sinnvoll ist (Benest und Potok 1984). Die Autoren bezeichnen solche Hilfen als Signposting-Techniken. Ähnlich einem Eye Mover kann der Button zur nächsten Navigationseinheit besonders hervorgehoben werden, wenn alle Entscheidungen der aktuellen Navigationseinheit getroffen wurden. Dies muss nicht durch einen Pfeil geschehen. Auch eine minimal vergrößerte Darstellung des Buttons, ein kurzes Aufflackern oder intensivere Farben können die Aufmerksamkeit des Benutzers auf die Route zum nächsten Ort lenken. Genauso kann auf noch offene Entscheidungen hingewiesen werden.

Die Oberfläche sollte aber nicht nur auf mögliche Routen aufmerksam machen, sondern auch vermitteln wo keine Routen bestehen. So betont Abbildung 6.8 a) eine bestimmte Route, während Abbildung 6.8 b) unterschiedliche Routen vermuten lässt. Auch anhand eines Überblicks können mögliche Routen vermittelt werden.

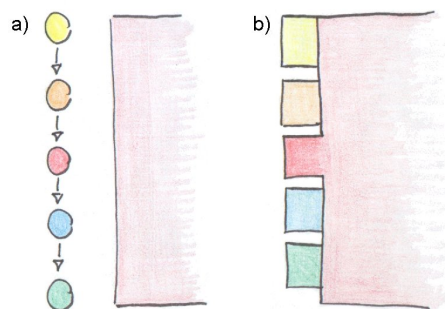


Abbildung 6.8: Mögliche und unmögliche Routen verdeutlichen

**Überblickswissen.** Überblickswissen ermöglicht sowohl *Distanzschätzungen* als auch das *Erkennen alternativer Routen* (Dutke 1994, S. 113f, 118ff). Daher ist Überblickswissen meines Erachtens auch sehr wichtig, um ein exploratives Arbeiten zu ermöglichen. Auf das Thema Distanzschätzungen wird in Abschnitt 6.4.4 näher eingegangen.

Überblicksinformationen können dem Benutzer in Form von Plänen oder Karten angeboten werden. Neben dem *Erkennen von Routen* erleichtern Karten auch das *Verständnis der Umgebung* (Passini 1984, S. 189ff). Um diese Funktionen zu erfüllen sollten sie:

- einen knappen und statischen Überblick bieten,
- die Beziehung zwischen der Karte und ihrer Umgebung verdeutlichen,
- das Finden von Zielorten ermöglichen,
- den Standort des Benutzers anzeigen und
- die Hauptrouten klar hervorheben.

Nur wenn die Karte so *knapp* ist, dass der Benutzer sie im Ganzen erfassen kann, bietet sie eine gute Unterstützung. Überladung sollte daher vermieden werden. Außerdem sollten Karten einen *statischen* Überblick anbieten. Dass ein sich ständig verändernder Überblick die Orientierung erschwert, ist unmittelbar einsichtig: Wird eine Karte zum Beispiel durch Expandieren einzelner Teile verändert, muss sich der Benutzer jeweils neu mit der Karte vertraut machen. Diese Anforderung an den Überblick wird auch durch Nielsen gestützt, der für Sitemaps fordert, dass sie einen statischen Überblick über die Struktur einer Website geben sollten (Nielsen 2002). Bei der formularbasierten Produktauswahl kann der Überblick daher weder die aktuell notwendigen Entscheidungen noch alle Entscheidungen anzeigen. Stattdessen sollte die Karte einen Überblick über die einzelnen Orte des Programms bieten. Wurden die Entscheidungen nach den Vorstellungen der Benutzer gruppiert, bietet der Überblick so eine Anzeige der Teilaufgabe, die die Benutzer beschrieben haben.

Die *Beziehung zwischen der Karte und ihrer Umgebung* kann verdeutlicht werden, indem die Orientierungspunkte der Umgebung in der Karte aufgegriffen werden, also zum Beispiel Farben, Symbole oder die Namen der Orte. Auch für das *Finden von Zielorten* spielen die Namen eine zentrale Rolle. Bei vielen Orten muss hier abgewogen werden zwischen den Informationen, die der Benutzer benötigt und einer knappen Darstellung, ohne die der Überblick nutzlos wird.

Der *aktuelle Standort des Benutzers* kann durch verschiedene Techniken im Überblick vermittelt werden. So kann er zum Beispiel durch Farbe, Form oder Größe hervorgehoben werden, oder durch einen Pfeil markiert sein. Abbildung

6.9 gibt für diese Techniken Beispiele. Eine weitere Möglichkeit besteht darin, den aktuellen Standort im Überblick als „unclickable“ zu realisieren.

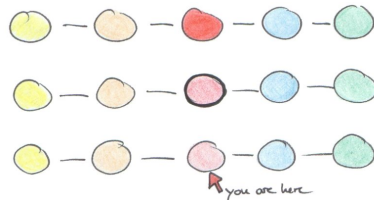


Abbildung 6.9: Aktuellen Standort im Überblick anzeigen

Schließlich sollte der Überblick Hauptwege hervorheben. Dies kann einerseits durch *Linien* oder *Pfeile* zwischen den Orten realisiert werden. Wenn ein Ort mit einigen anderen Orten durch Linien verbunden ist, wird die Verbindung zwischen diesen Orten im Gegensatz zu anderen betont. Aber auch schon allein durch die räumliche Anordnung können Wege vermittelt werden. Die in Abbildung 6.10 a) dargestellten Punkte legen aufgrund unserer Lesegewohnheiten einen Weg von links nach rechts nahe. Ist nur ein Weg zwischen den Orten möglich, so kann dieser klar durch Pfeile und eine Numerierung der Orte verdeutlicht werden. Abbildung 6.10 gibt einige Beispiele für die Vermittlung von Wegen im Überblick.

Die Raum-Metapher stellt meines Erachtens für alle Benutzergruppen eine große Unterstützung dar, und sollte daher bei einer formularbasierten Produktauswahl in jedem Fall eingesetzt werden. Anfänger profitieren, weil sie so ein mentales Modell von der Aufgabe aufbauen können. Außerdem kann bei einer benutzerorientierten Gruppierung der Entscheidungen gelegentliche Benutzer und Experten auf bereits Bekanntem aufbauen und finden sich so schneller zurecht. Der benutzerorientierten Gruppierung kommt beim Einsatz der Raum-Metapher daher eine große Bedeutung zu. Der Benutzer kann die Auswirkungen einer Entscheidung auf die anderen Entscheidungen am gleichen Ort leicht untersuchen und so Entscheidungen in Abhängigkeit voneinander treffen.

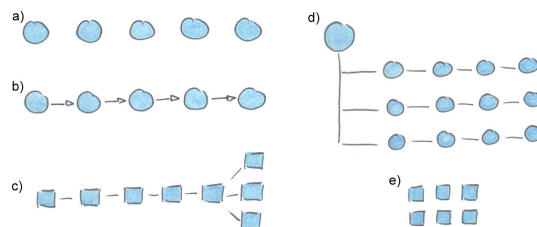


Abbildung 6.10: Wege im Überblick verdeutlichen

Karten, die einen Überblick über einen virtuellen Raum bieten, können im Gegensatz zu Stadtplänen oder Landkarten nicht nur das Erkennen von Wegen ermöglichen. Sie können auch „Sprünge“ zu den angezeigten Zielen ermöglichen. So kann Benutzern der schnelle Wechsel zwischen Bereichen geboten werden, den Hacker (1994, S. 36) zur Unterstützung von Experten bei Entwurfstätigkeiten fordert. Auch deshalb sollte der Überblick ständig angezeigt werden. Zusätzlich kann über den Überblick auch Feedback zu bereits erledigten und noch offenen Teilaufgaben gegeben werden. Hierauf wird in Abschnitt 6.4.4 eingegangen.

Der Einsatz der Raum-Metapher erfordert eine Gruppierung von Konfigurierungsschritten. Dies ist auf der Basis von EngCon möglich: Es können beliebige, in der Agenda enthaltene Konfigurierungsschritte gruppiert werden. Eine benutzerorientierte Gruppierung ist nur durch die frühzeitige Zusammenarbeit mit den zukünftigen Benutzern realisierbar. Wenn sowohl Vertriebsmitarbeiter als auch Kunden mit dem System arbeiten, sollten auch beide Benutzergruppen einbezogen werden. Bei der Verwendung einer benutzerorientierten Gruppierung entsteht ein Problem, wenn Benutzer oder Benutzergruppen sehr unterschiedliche Vorstellungen von der Arbeitsaufgabe haben. Eine relativ aufwändige Lösung besteht in der Verwendung von unterschiedlichen Gruppierungen für die einzelnen Benutzergruppen. Hierdurch wird aber auch die Kommunikation zwischen diesen Benutzergruppen erschwert. Eine sinnvolle Alternative sehe ich in diesem Fall darin, eine Gruppierung zu erarbeiten, die für alle Benutzer relativ leicht erlernbar ist. Ein solcher „Kompromiss“ kann dann auch den Benutzern als Grundlage für die Kommunikation über Arbeitsergebnisse dienen. Dabei wird das System als *Medium zur Unterstützung der Kommunikation* zwischen Benutzern eingesetzt. Die klare Vermittlung der zugrunde liegenden Struktur macht es den Benutzern möglich, Diskrepanzen zwischen dem konzeptuellen Modell und ihrem mentalen Modell zu erkennen.

Auch in der direktmanipulativen Produktauswahl ist der Einsatz der Raum-Metapher sinnvoll, wenn nicht die gesamte Teillösung gleichzeitig angezeigt werden kann. So wird zum Beispiel in Abbildung 6.4 auf Seite 91 für jeden Raum, den die GA-Lösung umfasst, ein eigener Reiter verwendet.

### **Sichtbarkeit von Entscheidungen**

Um dem Benutzer die Orientierung zu erleichtern, sollte er sich einen Überblick über die prinzipiell möglichen Entscheidungen verschaffen können. Dieser Unterstützungsansatz ist vergleichbar mit der Sichtbarkeit aller Menüpunkte im Menü. Dort werden jeweils alle Funktionen angezeigt und nicht nur die aktuell wählbaren, wodurch eine ständige Veränderung der „Umgebung“ vermieden wird.

Sind alle Entscheidungen sichtbar, so kann sich der Benutzer einen Überblick über die gesamte Aufgabe verschaffen; er kann die möglichen Entscheidungen er-

forschen, wodurch der Aufbau eines mentalen Modells unterstützt wird. Es sollte auch vermittelt werden, welche Entscheidungen aktuell notwendig sind und welche nicht. Dies kann ähnlich wie bei Menüpunkten durch Hervorheben geschehen, wie Abbildung 6.11 zeigt. Durch die Gruppierung von Entscheidungen und

The image shows a user interface window titled "Beleuchtung" (Lighting). At the top left is a lightbulb icon. Below the title, the text reads: "Das Raumkonzept soll eine Beleuchtungssteuerung [i] umfassen:". There are two radio button options: "ja [i]" (selected) and "nein [i]". Below this is a horizontal separator line. Underneath, it says "Anzahl der Beleuchtungsgruppen [i] : 1" with a dropdown arrow. Another horizontal separator line is at the bottom.

Abbildung 6.11: Sichtbarkeit aller Entscheidungen und Hervorhebung der aktuell notwendigen

deren ständige Sichtbarkeit werden außerdem Beziehungen zwischen einzelnen Entscheidungen offenkundig. Wählt der Benutzer bei Entscheidung X die Alternative A, so wird Entscheidung Z notwendig. Diesen Vorteilen steht das Problem der Überladung gegenüber. Aufgrund des großen Lösungsraums ist es bei der Auswahl komplexer Produkte oft nicht sinnvoll, alle prinzipiell möglichen Entscheidungen anzuzeigen. Umfasst eine Navigationseinheit eine zu große Menge von Entscheidungen, so kann der Benutzer trotz Sichtbarkeit aller Entscheidungen kaum einen Überblick gewinnen. Hier muss jeweils ein sinnvolles Mittelmaß gefunden werden, bei dem die wichtigsten Entscheidungen jederzeit sichtbar sind, das Problem der Überladung aber vermieden wird.

Genau wie die Sichtbarkeit von Entscheidungen unterstützt auch die Sichtbarkeit aller prinzipiell bestehenden Auswahlalternativen die Orientierung. So kann der Benutzer erkennen, welche Alternativen prinzipiell angeboten werden und welche nicht. Das Produktspektrum wird also für den Benutzer erkennbar, wie von Nielsen gefordert (siehe Abschnitt 6.1.2). Dabei sollten jeweils die Auswahlalternativen hervorgehoben werden, die aufgrund der aktuellen (Teil-)Lösung gewählt werden können.

Ganz besonders wichtig ist die Sichtbarkeit aller Entscheidungen und Wahlmöglichkeiten für Anfänger und gelegentliche Benutzer, die so einen Überblick

gewinnen können. Durch diesen Ansatz wird die Konsistenz zwischen verschiedenen Arbeitsdurchläufen erhöht, was auch für Experten hilfreich, da sie so bestimmte Handlungsabläufe automatisieren können.

Um die Sichtbarkeit aller zentralen Entscheidungen zu ermöglichen, ist eine angemessene Modellierung notwendig. Durch sie kann sichergestellt werden, dass zentrale Entscheidungen prinzipiell in der Agenda enthalten sind.

### 6.4.2 Schemata

Die große Menge an Entscheidungen erschwert die Orientierung in der Produktauswahl. Durch eine Reduzierung der notwendigen Entscheidungen, also eine „Reduzierung der Umgebung“, kann die Orientierung erleichtert werden. Eine solche Reduzierung kann durch das Angebot von Teillösungen für typische Produktklassen erreicht werden. Als Ausgangspunkt für die Produktauswahl wählt der Benutzer ein passendes Schema aus. Ein solches Schema beschreibt für die jeweilige Produktklasse die Struktur der Produkte. Dadurch wird nicht nur die Menge der notwendigen Entscheidungen, sondern auch die Menge der erreichbaren Lösungen reduziert, wie Abbildung 6.12 verdeutlicht. Durch das Angebot einer Reihe von Schemata kann dem Benutzer für die meisten Lösungen eine solche einfachere „Umgebung“ angeboten werden.

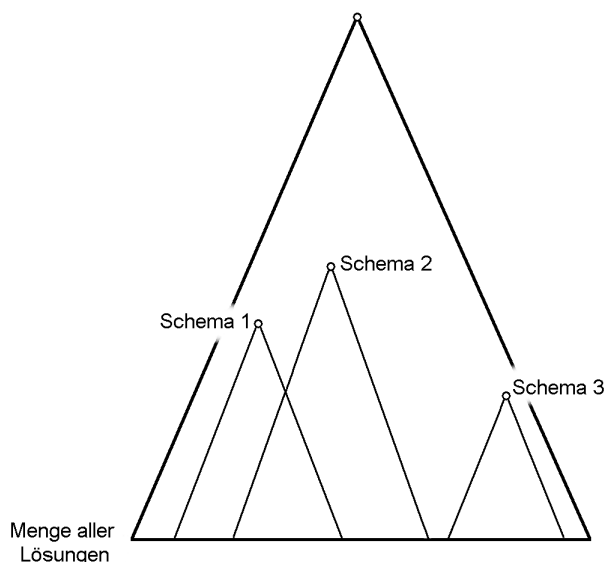


Abbildung 6.12: Schemata zur „Reduzierung der Umgebung“

Im Anwendungsbeispiel SVEA könnten dies Schemata für Bürogebäude, für Schulen oder Krankenhäuser sein. Bei der Auswahl eines PKWs bieten sich Kategorien wie „Familienwagen“, „sportlich“ oder „sicher“ an. Eine solche Vorauswahl wird unter anderem durch den BMW-Konfigurator geboten. Hier kann der Benutzer zwischen sechs unterschiedlichen „driving preferences“ wählen<sup>3</sup>.

Das Angebot von Schemata ist ein Unterstützungsansatz der sowohl in der formularbasierten als auch in der direktmanipulativen Produktauswahl sinnvoll eingesetzt werden kann. In der formularbasierten Produktauswahl können dem Benutzer zusätzlich auf die Produktklasse zugeschnittene feste Wege angeboten werden. Auf diesen Aspekt wird in Abschnitt 6.5.1 eingegangen.

Voraussetzung dafür, dass der Benutzer die Schemata erfolgreich nutzen kann, sind erwartungskonforme Bezeichnungen; die Benennung der Schemata muss den jeweiligen Lösungsraum verdeutlichen. Anhand der Bezeichnungen muss sich der Benutzer also die Fragen beantworten können: „Mit welchem Schema kann ich mein Ziel erreichen?“ und „Welche Lösungen kann ich mit Schema A nicht erreichen?“.

Da Schemata nicht nur die Orientierung erleichtern, sondern zusätzlich eine *Abkürzung* durch die Arbeitsaufgabe bieten, stellen sie für alle Benutzergruppen eine sinnvolle Unterstützung dar. Schemata entsprechen Teilkonfigurationen, die als Ausgangspunkt der Produktauswahl dienen. Dieser Unterstützungsansatz ist nicht nur mit EngCon realisierbar, sondern wird auch schon in der Praxis eingesetzt. Der in Abschnitt 3.3 genannte *Drive Solution Designer* bietet dem Benutzer eine Reihe von Schemata zur Auswahl an.

### 6.4.3 Zurückgelegter Weg

Den bisher im Rahmen der Produktzusammenstellung zurückgelegten Weg kann der Benutzer mit Hilfe einer Interaktionshistorie oder einem Interaktionsgraphen erforschen. Während die *Interaktionshistorie* die Form eines Protokolls hat, bietet der *Interaktionsgraph* eine Visualisierung durch einen gerichteten Graphen. Beide veranschaulichen das bisherige Vorgehen und unterstützen so die Orientierung. Paul (1994, S. 161) beschreibt, dass sie dem Benutzer die Nievergeltsche Frage „Wie kam ich hierhin?“ beantworten. Umfassen Historie und Graph nur die Schritte des Benutzers, stimmt diese Aussage im Rahmen der Konfigurierung nicht. Die aktuelle Teilkonfiguration beruht nicht nur auf den Entscheidungen des Benutzers, sondern auch auf den Auswirkungen dieser Entscheidungen, die vom System berechnet wurden. Durch taxonomische Inferenz und Constraint-Propagation kann die Lösungsmenge gegebenenfalls weiter eingeschränkt wer-

---

<sup>3</sup>BMW-Konfigurator: (<http://www.bmwusa.com/virtual/overview/build.html>; zuletzt besucht: 12.03.2003)

den. Damit der Benutzer sich mit Hilfe der Historie die oben genannte Frage beantworten kann, müssen auch diese „Folgeentscheidungen“ für ihn sichtbar gemacht werden. Zur Unterstützung von Benutzern bei der Produktauswahl sollte die Historie meines Erachtens daher sowohl die Benutzerentscheidungen als auch vom System berechnete „Folgeentscheidungen“ enthalten. Die Anzeige aller Auswirkungen kann jedoch wieder das Problem der Überladung aufwerfen. Hat eine Entscheidung Auswirkungen auf hunderte von Parametern, so ist die Anzeige aller Parameter nicht sinnvoll. Die Auswirkungen einer Entscheidung auf andere zentrale Entscheidungen sollte jedoch in jedem Fall sichtbar sein. Außerdem sollte für den Benutzer jederzeit erkennbar sein, welche der Entscheidungen nicht von ihm selbst festgelegt wurden. Eine solche Historie gibt dem Benutzer auch die Möglichkeit, sich die folgenden Fragen zu beantworten: „Welche Entscheidungen sind bereits erledigt?“ und „Welche der Entscheidungen wurden vom System ermittelt?“. Es wird also vermittelt, dass die Produktzusammenstellung nicht allein auf den Entscheidungen des Benutzers beruht. Dies halte ich vor allem zur Unterstützung von Anfängern für sehr sinnvoll.

Eine Historie kann dem Benutzer in textueller Form auch über eine Combobox angeboten werden. Durch den sehr geringen Platzverbrauch dieser Variante, kann sie jederzeit parallel zur Arbeit angeboten werden. Durch Wiederaufsetzpunkte (siehe Abschnitt 6.5.6) und einzelne Undo-Mechanismen (siehe Abschnitt 6.5.4) können jedoch Verzweigungen in der Historie entstehen, die in Listenform nicht mehr darstellbar sind. Mit einem Interaktionsgraphen kann dagegen auch bei Verzweigungen ein Überblick über alle zurückliegenden Schritte gegeben werden, was jedoch mit einem wesentlich höherem Platzverbrauch erkauft wird. Ein Interaktionsgraph kann daher nur in einem separaten Fenster angeboten werden. Untersucht der Benutzer im Rahmen der Produktauswahl eine Reihe von Alternativen, so kann der Interaktionsgraph sehr umfangreich werden. Um solche Graphen überschaubar zu machen, schlägt Paul *Filter* vor (Paul 1994, S. 164ff). Mit Hilfe von Filtern können unterschiedliche Sichten auf die Historie bzw. den Graphen dargestellt werden. Paul unterscheidet hervorhebende, reduzierende, gruppierende und optimierende Filter.

Neben der Möglichkeit, den zurückgelegten Weg zu erkunden, sind die Historie bzw. der Interaktionsgraph auch wirkungsvolle Hilfsmittel bei der Stornierung von Handlungsschritten. Mit ihrer Hilfe kann der Benutzer die zu stornierenden Schritte auswählen und so auch mehrere Schritte auf einmal widerrufen. Im Gegensatz zu einem einfachen Undo-Button kann der Benutzer im Voraus erkennen welchen Schritt bzw. welche Schritte er storniert. Auch Yang (1990) fordert eine Interaktionshistorie, die separat alle ausgeführten und stornierten Handlungsschritte enthält. Auf das Thema Stornierung wird in Abschnitt 6.5.4 eingegangen.

Interaktionsgraph bzw. Interaktionshistorie sind für alle Benutzergruppen ein wirkungsvolles Hilfsmittel bei der Produktauswahl und unterstützen bei der Ori-



entierung, dadurch dass sie den zurückliegenden Weg vermitteln.

Bei der formularbasierten Produktauswahl bietet sich das Angebot einer Historie an, da der Benutzer die Entscheidungen in textueller Form trifft. Die im Formular verwendeten Begriffe können so in der Historie aufgegriffen werden. Bei der direktmanipulativen Produktauswahl können dagegen die grafischen Darstellungen der einzelnen Objekte gut im Interaktionsgraphen aufgegriffen werden. Eine textuelle Historie würde dagegen vom Benutzer verlangen, dass er die von ihm durchgeführten Aktionen in die Beschreibungen der Historie übersetzt.

Die Realisierung einer Interaktionshistorie oder eines Interaktionsgraphen erfordert ein Elaborationsnetz (TZI 2001, S. 65), wenn alle „Folgeentscheidungen“ angezeigt werden sollen. Ein solches Elaborationsnetz ist auf der Basis von EngCon realisierbar, wird aber momentan noch nicht eingesetzt. Eine Historie der Benutzerentscheidungen ist auch ohne ein Elaborationsnetz realisierbar. Sie kann auch erweitert werden um Auswirkungen, die die Benutzerentscheidungen auf eine Reihe von anderen zentralen Entscheidungen haben.

#### **6.4.4 Fortschritt — Standort des Benutzers im Arbeitsprozess**

Benutzer, die eine klare Vorstellung von der Aufgabe Produktauswahl haben, können ihren Fortschritt selbst beurteilen. Sie können das, was sie bereits erledigt haben, in Bezug setzen zu dem, was noch vor ihnen liegt, vergleichbar mit einer Person, die einen alltäglichen Weg geht oder fährt. Sie fragt sich nicht, wann sie endlich da ist. Sie weiß es.

Benutzer, die keine so genaue Vorstellung von der Aufgabe haben, können ihren Fortschritt dagegen nicht so gut beurteilen. Bei einer formularbasierten Produktauswahl, die nur aus einer langen Folge von Entscheidungen besteht und erst am Ende die Ergebnisse präsentiert, kann so leicht das „Are We There Yet?“-Syndrom ausgelöst werden. In der direktmanipulativen Produktauswahl bekommt der Benutzer ständig Feedback und kann so seinen Fortschritt verfolgen. Daher kann die *direkte Manipulation* als ein Unterstützungsansatz gesehen werden, der Benutzern ihren Fortschritt vermittelt.

Auch in der formularbasierten Produktauswahl kann dem Benutzer das fortschreitende „Wachstum“ *des Produktes vermittelt* werden. Hierfür kann das bisher zusammengestellte Produkt grafisch visualisiert werden. Eine einfachere Möglichkeit besteht in der textuellen Anzeige der bisher ermittelten Komponenten. Durch das „Wachstum des Produktes“ kann der Benutzer zwar seinen Fortschritt bei der Arbeit erkennen, aber nicht unbedingt ablesen, wie viel noch vor ihm liegt.

Eine nahe liegende Möglichkeit, um dem Benutzer in der formularbasierten Produktauswahl zu vermitteln, wie viel noch vor ihm liegt, wäre das Angebot eines Fortschrittsbalkens, wie er zum Beispiel bei Kopiervorgängen häufig angezeigt wird. Bei der Zusammenstellung eines komplexen Produktes ist dies jedoch

nicht sinnvoll realisierbar. Während bei einem Kopiervorgang der Umfang der Datei feststeht (z.B. 100 KB), ist die Anzahl der notwendigen Entscheidungen für die Produktauswahl zunächst unbekannt, da auch die Struktur des Produktes noch nicht feststeht. Die Geschwindigkeit eines Kopiervorgangs kann zwar variieren und damit auch die verbleibende Restzeit, der bereits kopierte Anteil der Daten kann jedoch nicht abnehmen. Bei der Produktauswahl kann die Anzahl der Entscheidungen anfangs höchstens sehr grob geschätzt werden. Im Laufe der Produktzusammenstellung kann diese Schätzung dann erheblich schwanken, wodurch im Fortschrittsbalken eventuell ein „Rückschritt“ angezeigt werden müsste. Darüber hinaus hat der Fortschrittsbalken für die Produktauswahl einen weiteren Nachteil. Bei einem Kopiervorgang ist es für den Benutzer unerheblich, welche der Datenblöcke bereits kopiert wurden. Welche der Entscheidungen im Arbeitsprozess bereits festgelegt wurden, ist für den Benutzer aber eine wichtige Information. Insbesondere bei einem flexiblen Vorgehen muss der Benutzer erkennen können, wo noch Entscheidungen offen sind. Eine sinnvolle Lösung besteht daher darin, den *Fortschritt auf der Ebene von Teilaufgaben* zu vermitteln. Dieser Ansatz basiert auf dem Angebot eines Überblicks über die Aufgabe (siehe hierzu Abschnitt 6.4.1 über die Raum-Metapher). Beruhen die Orte auf einer benutzerorientierten Gruppierung der Entscheidungen, so steht jeder Ort für eine Teilaufgabe. Im Überblick kann dann für diese Teilaufgaben angezeigt werden, ob sie bereits vollständig bearbeitet wurden. Dieser Ansatz wird an einem Beispiel in Abbildung 6.13 veranschaulicht. Alternativ können auch die Bereiche gekennzeichnet werden, in denen noch offene Entscheidungen bestehen. Dieser Ansatz wird in Abbildung 6.4 auf Seite 91 eingesetzt. Objekte, die noch genauer festgelegt werden müssen, sind dort durch ein Fragezeichen gekennzeichnet. Durch die Abnahme der Fragezeichen kann der Benutzer seinen Fortschritt erkennen und abschätzen, wie viel noch vor ihm liegt.

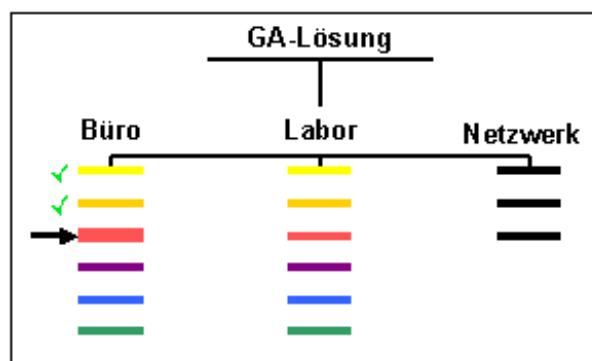


Abbildung 6.13: Fortschritt anhand der erledigten Teilaufgaben vermitteln

Ein wesentlicher Vorteil der direktmanipulativen Produktauswahl besteht in der Tatsache, dass der Benutzer durch das ständige Feedback seinen Fortschritt jederzeit erkennen kann. Die in Abschnitt 6.1.2 genannte Anforderung, dass der Benutzer sofortiges Feedback zu den getroffenen Entscheidungen bekommen sollte, wird durch die direktmanipulative Produktauswahl also erfüllt. In der formularbasierten Produktauswahl sollte das oberste Ziel darin bestehen, Benutzern eine Vorstellung von der gesamten Aufgabe zu vermitteln. So können sie ihren Fortschritt auch selbst einschätzen. Das *Wachstum des Produktes* und der *Fortschritt auf der Ebene von Teilaufgaben* sollten zusätzlich angeboten werden. Das Wachstum des Produktes bietet auch das oben angesprochene sofortige Feedback zu den getroffenen Entscheidungen. Bei einem flexiblen Vorgehen (siehe hierzu Abschnitt 6.5.1) muss der Fortschritt unbedingt über den Überblick angezeigt werden, da Benutzer ansonsten gezwungen sind, sich offene Entscheidungen zu merken oder nach ihnen zu suchen.

Sowohl das Wachstum des Produktes als auch die bereits erledigten Teilaufgaben können anhand der aktuellen Teilkonfiguration ermittelt werden.

### 6.4.5 Typische Wege

Bei der Produktkonfigurierung spielen Vorgehensweisen eine wichtige Rolle. Bei ungünstigem Vorgehen kann es sein, dass überhaupt keine Lösung oder nur eine sehr schlechte gefunden wird und die Wahrscheinlichkeit von Konflikten ist eventuell wesentlich höher. Personen mit geringem Domänenwissen sind solche günstigen Vorgehensweisen meist nicht bekannt. Im Folgenden werden zwei Ansätze beschrieben, mit denen Anfänger sich mit typischen Wegen durch die Produktauswahl vertraut machen können – die *animation machine* und die *Szenario-Maschine*.

**Animation machine.** Howes und Payne (1990) haben die *animation machine* in Anlehnung an Videospiele entworfen. Videospiele zeigen oft einzelne Spielsequenzen, wenn das System nicht genutzt wird. So soll das Interesse der Spieler erweckt werden; der Spieler bekommt erste grundlegende Informationen und kann Strategien erkennen. Die *animation machine* bietet dem Benutzer eine Reihe von „Filmen“ zur Auswahl an. Sie zeigen die Bearbeitung einzelner Aufgaben, wobei die Benutzeraktionen und das Feedback des Systems veranschaulicht werden.

Der Ansatz der *animation machine* kann in der Produktauswahl sinnvoll eingesetzt werden. Anfänger können sich anhand eines Films einen ersten Überblick über die Produktauswahl mit dem System verschaffen und es können Vorgehensweisen vermittelt werden, die in den meisten Fällen sinnvoll sind. Filme können zum Beispiel für typische Produktklassen oder Teillösungen angeboten werden. Im Anwendungsfall SVEA bieten sich unter anderem Filme zur Beleuchtungssteuerung, Sonnenschutzsteuerung und Einzelraumregelung an.

**Szenario-Maschine.** Während die animation machine dem Benutzer die Rolle eines passiven Zuschauers gibt, ermöglicht die Szenario-Maschine die Arbeit mit dem System. Die von Paul (1994, S. 170ff) beschriebene Szenario-Maschine bietet dem Benutzer eine Auswahl von Szenarien an; jedes Szenario beschreibt die Lösung einer Teilaufgabe. Hierfür ist jeweils eine feste Folge von Schritten vorgesehen („one-best-way“). Ein Szenario beschreibt also einen Lösungsweg für eine konkrete (Teil-)Aufgabe. Der Benutzer führt die Schritte selbst aus, kann aber von der vorgesehenen Folge nicht abweichen. Da das Durchlaufen eines Szenarios der normalen Interaktion möglichst nahe kommen soll, laufen Szenarien im System selbst ab. Entgegen der normalen Interaktion ist aber immer nur der Button oder Menüpunkt aktiv, der für den nächsten Schritt benötigt wird. Kurze Texte erläutern die Auswirkungen des durchgeführten Schrittes sowie den jeweils nächsten Schritt. Der Benutzer spielt hierbei also einen konkreten Konfigurationsdurchgang nach, wodurch er eine Beziehung zwischen Systemfunktionen und Arbeitsaufgaben herstellen kann.

Da der Benutzer die Schritte in der Szenario-Maschine selbst durchführt, ist damit zu rechnen, dass sie besser erinnert werden. Die im Rahmen eines Szenarios erstellte Lösung sollte der Benutzer meines Erachtens anschließend weiter bearbeiten können. Dazu kann der Benutzer zum Abschluss eines Szenarios wählen zwischen „Dieses Produkt weiter bearbeiten“, „Neues Szenario starten“ und „Produktauswahl starten“. So kann der Benutzer zum Beispiel die erarbeitete Beleuchtungssteuerung um eine Sonnenschutzsteuerung erweitern oder einzelne Entscheidungen zur Beleuchtungssteuerung gezielt ändern. Für das Ändern solcher Lösungen ist eine Undo-Funktion beziehungsweise das direkte Ändern von Lösungen notwendig. Diese Themen werden in den Abschnitten 6.5.4 und 6.5.5 behandelt.

Die hier beschriebenen Ansätze sind speziell zur Unterstützung von Anfängern und gelegentlichen Benutzern geeignet. Neben typischen Vorgehensweisen kann mit diesen Ansätzen auch die Nutzung von einzelnen Systemfunktionen vorgestellt werden. So kann zum Beispiel der Einsatz von Wiederaufsetzpunkten (siehe Abschnitt 6.5.6) zum Erstellen von Alternativen vorgestellt werden.

Die animation machine kann vollständig unabhängig vom Konfigurationssystem realisiert werden. Benötigt werden nur einzelne Screenshots als Material für die Filme. Dies ist ein Vorteil gegenüber der Szenario-Maschine. Die Realisierung der Szenario-Maschine ist etwas aufwändiger, da die Szenarien im System selbst ablaufen. Für die einzelnen Szenarien muss die Benutzungsschnittstelle um die erläuternden Texte erweitert werden und jeweils alle Funktionen außer der gerade erlaubten Funktion gesperrt werden. Dies kann durch eine zusätzliche Schicht zwischen Benutzungsschnittstelle und Konfigurierungskern erreicht werden, die das Vorgehen gemäß einer vorgegebenen Konfiguration sicherstellt.

## 6.5 Ansätze zur Unterstützung der Navigation

### 6.5.1 Flexibilität im Vorgehen — Fester Weg bis freie Navigation

Aufgrund der Abhängigkeiten zwischen den Entscheidungen können diese nicht in beliebiger Reihenfolge getroffen werden. So kann zum Beispiel im Anwendungsfall SVEA die Netzwerktechnik nicht festgelegt werden, bevor nicht alle anderen Entscheidungen getroffen wurden. Abgesehen von diesen Einschränkungen kann der Benutzer die Reihenfolge der Entscheidungen frei wählen, wenn er ein komplexes Produkt ohne Unterstützung durch einen Konfigurator zusammenstellt. Der Benutzer kann sich also frei durch die Arbeitsaufgabe bewegen. Durch den Einsatz eines Konfigurators kann die Reihenfolge, in der die Entscheidungen getroffen werden können, weiter eingeschränkt werden. Die Interaktion kann also eher systemgeführt oder eher benutzergeführt realisiert werden. Vor- und Nachteile dieser Interaktionsformen wurden einander in Abbildung 2.2 auf Seite 21 gegenübergestellt. Die hier behandelte Flexibilität im Vorgehen verbindet das Thema der *system- bzw. benutzergeführten Interaktion* mit dem Thema der *Kontrolle in der Konfigurierung* (siehe hierzu Abschnitt 3.2 auf Seite 38). Im Folgenden werden vier Alternativen beschrieben und bewertet – der feste Weg, alternative feste Wege, das Leitsystem und der freie Weg. Zwischen einem komplett freiem Vorgehen und einem vollständig festen Weg sind beliebige Abstufungen denkbar. Die hier von mir vorgestellten Alternativen sollen jedoch die bestehenden Vor- und Nachteile veranschaulichen.

#### Reihenfolge der Entscheidungen durch System festgelegt – Fester Weg

Ist die Reihenfolge der Entscheidungen durch das System festgelegt, so kann der Benutzer zu einem Zeitpunkt immer nur genau eine Entscheidung treffen; die Interaktion ist vollständig systemgesteuert. Alle auf Seite 38 beschriebenen Kontrolltypen außer der benutzerorientierten Kontrolle realisieren eine solche systemgesteuerte Interaktion.

Eine rein systemgesteuerte Interaktion steht im klaren Widerspruch zu den Merkmalen der Arbeitsaufgabe Produktauswahl, da ein iteratives Vorgehen nicht möglich ist. Außerdem kann der Benutzer die Entscheidungen nicht entsprechend seinen eigenen Prioritäten treffen. Ist ihm beispielsweise der Einsatz eines bestimmten Bedienelementes am wichtigsten, so kann er diese Entscheidung trotzdem nicht zuerst treffen und alle weiteren an diese anpassen.

Andererseits können durch einen solchen Weg auch Personen mit geringem Domänenwissen ohne umfangreiche Einarbeitung schnell ein komplexes Produkt zusammenstellen. Die Reihenfolge der Entscheidungen kann hierbei so festgelegt

werden, dass Konflikte möglichst unwahrscheinlich sind. Außerdem kann der Benutzer sich hierdurch voll auf den Inhalt der Entscheidungen konzentrieren, ohne durch die Planung des weiteren Vorgehens von der eigentlichen Aufgabe abgelenkt zu werden. Benest und Potok (1984) betonen, dass der Benutzer durch die Planung der nächsten Handlungen in einem innovativen Designprozess unterbrochen werden kann. Die von den Autoren entwickelten Signposting-Techniken sollen den Benutzer so durch das System leiten, dass er sich dabei voll auf die Arbeitsaufgabe konzentrieren kann. Ein Beispiel für diese Signposting-Techniken ist der *Eye Mover* (siehe Abschnitt 6.4.1). Die Signposting-Techniken schränken jedoch den Handlungsspielraum des Benutzers nicht ein. Hierin unterscheiden sie sich wesentlich von einer durch das System festgelegten Reihenfolge von Entscheidungen.

### **Benutzer wählt eine vom System festgelegte Reihenfolge – Alternative feste Wege**

Der Handlungsspielraum kann minimal erweitert werden, indem der Benutzer zwischen einer Reihe von verschiedenen festgelegten Reihenfolgen wählen kann. Dies könnte zum Beispiel durch eine fallorientierte Kontrolle realisiert werden: Der Benutzer wählt zunächst einen Fall aus und geht anschließend analog zu diesem vor. Dies entspricht einer Verbindung von benutzerorientierter und fallorientierter Kontrolle.

Diese Erweiterung des Handlungsspielraumes verlangt vom Benutzer ein umfangreicheres Wissen, damit er vor Arbeitsbeginn eine angemessene Vorgehensweise auswählen kann. Sinnvoll ist der Einsatz von alternativen festen Wegen in Verbindung mit dem Angebot von Schemata (siehe hierzu Abschnitt 6.4.2). Hier wählt der Benutzer also nicht direkt eine spezielle Vorgehensweise, sondern eine Teillösung, die seinem Arbeitsziel entspricht. Für jedes Schema kann dann jedoch auch ein angepasster fester Weg angeboten werden.

### **Benutzer kann von festgelegter Reihenfolge abweichen – Leitsystem**

Eine andere Möglichkeit zur Erweiterung des Handlungsspielraumes besteht darin, zwar eine feste Reihenfolge für die Entscheidungen festzulegen (one-best-way), dem Benutzer aber die Möglichkeit zu geben, jederzeit von dieser abzuweichen. Der Benutzer hat so einen festen Weg zur Verfügung, kann aber sein Vorgehen stets der Situation anpassen.

Hierdurch bleiben die Vorteile eines festen Wegs erhalten. So kann der Benutzer auch ohne umfangreiches Domänenwissen und aufwändige Einarbeitung schnell ein komplexes Produkt zusammenstellen. Durch den festen Weg kann eine Vorgehensweise betont werden, die in den meisten Fällen sinnvoll ist. Er-

fordern es aber die Prioritäten des Benutzers, so kann er von dieser festgelegten Reihenfolge abweichen. Ansonsten kann er dem festen Weg folgen und sich so voll auf die Inhalte der Entscheidungen konzentrieren.

### **Benutzer wählt Reihenfolge der Entscheidungen selbst – Freie Navigation ohne Leitsystem**

Schließlich kann auch ganz auf einen festen Weg verzichtet werden. Eine solche vollständig benutzergeführte Interaktion basiert ausschließlich auf dem benutzerorientierten Kontrolltyp. Nach jeder Entscheidung legt der Benutzer fest, welche Entscheidung er als nächstes treffen möchte.

Auch hier kann der Benutzer natürlich entsprechend seinen Prioritäten vorgehen. Durch die aktiv getroffenen Navigationsentscheidungen wird zudem der Aufbau eines mentalen Modells der Arbeitsaufgabe gefördert. Dies beschreibt auch Schenck (1997, S. 38). Wählt der Mensch seinen Weg zum Beispiel durch eine Stadt oder ein Programm selbständig, so erinnert er sich besser, als wenn er passiv einer Führung gefolgt ist. Ein solches aktives Navigieren durch die Arbeitsaufgabe ist aber nicht ohne grundlegendes Wissen möglich. Hat der Benutzer keine Vorstellung von den Auswirkungen seiner Entscheidungen, so trifft er sie eventuell in einer ungünstigen Reihenfolge. Hierdurch kann es zu eigentlich vermeidbaren Konflikten kommen. Außerdem ist es möglich, dass der Benutzer gar keine Lösung findet, oder nur solche, die seinen Anforderungen nicht gerecht werden.

### **Vergleich und Bewertung**

Die Tabelle auf Seite 112 fasst die Vor- und Nachteile der oben beschriebenen Alternativen zusammen.

**Fester Weg.** Wie bereits oben beschrieben, steht der feste Weg im Widerspruch zu den Merkmalen der Arbeitsaufgabe Produktauswahl. Daher halte ich diese eingeschränkten Navigationsmöglichkeiten zur Unterstützung von gelegentlichen Benutzern und Experten nicht für sinnvoll. Personen, die die Aufgabe äußerst selten durchführen und nur über ein geringes Domänenwissen verfügen, können aber sinnvoll durch einen festen Weg unterstützt werden. Für sie bietet er eine Möglichkeit, das Ziel überhaupt zu erreichen. Auch Paul hält die systemgesteuerte Interaktion für eine Interaktionsform, „die vor allem dann angestrebt werden sollte, wenn Arbeitsschritte nur selten durchgeführt werden, trotz fehlender Expertise in einem Anwendungsbereich Handlungsziele erreicht werden sollen oder die Konzentration nicht von anderen Vorgängen abgelenkt werden darf“ (Paul 1994, S. 142). Meines Erachtens kann ein fester Weg Anfängern einen schnellen Einstieg ermöglichen. Sie können hierdurch sofort in der zunächst un-

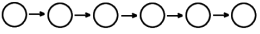
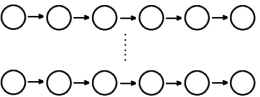
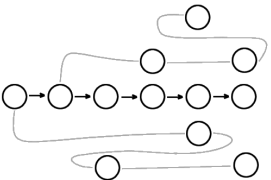
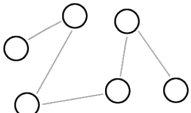
Navigationsmöglichkeiten durch die Arbeitsaufgabe	Vorteile	Nachteile
<p>Fester Weg</p> 	<p>Ziel kann ohne umfangreiches Domänenwissen schnell erreicht werden</p> <p>Hebt für den Standardfall günstigen Weg hervor</p> <p>Keine Einarbeitung notwendig</p> <p>Benutzer wird nicht durch Navigationsentscheidungen belastet</p>	<p>Benutzer kann nicht entsprechend seiner Prioritäten vorgehen</p> <p>Steht im Widerspruch zum Vorgehen bei der Produktauswahl</p> <p>Kein Handlungsspielraum</p>
<p>Alternative feste Wege</p> 	<p>Ziel kann ohne umfangreiches Domänenwissen schnell erreicht werden</p> <p>Hebt für den Standardfall günstigen Weg hervor</p> <p>Kaum Einarbeitung nötig</p> <p>Benutzer wird nicht durch Navigationsentscheidungen belastet</p>	<p>Benutzer kann nicht entsprechend seiner Prioritäten vorgehen</p> <p>Steht im Widerspruch zum Vorgehen bei der Produktauswahl</p> <p>Kaum Handlungsspielraum</p> <p>Erfordert Auswahl einer passenden Vorgehensweise vor Arbeitsbeginn</p>
<p>Leitsystem</p> 	<p>Ziel kann ohne umfangreiches Domänenwissen schnell erreicht werden</p> <p>Hebt für den Standardfall günstigen Weg hervor</p> <p>Kaum Einarbeitung nötig</p> <p>Benutzer wird nicht durch Navigationsentscheidungen belastet</p> <p>Benutzer kann entsprechend seiner Prioritäten vorgehen</p> <p>Handlungsspielraum</p> <p>Wird der Arbeitsaufgabe Produktauswahl gerecht</p>	<p>Beim Abweichen vom festen Weg bestehen die gleichen Nachteile wie im freien Weg</p>
<p>Freier Weg ohne Leitsystem</p> 	<p>Benutzer kann entsprechend seiner Prioritäten vorgehen</p> <p>Handlungsspielraum</p> <p>Wird der Arbeitsaufgabe Produktauswahl gerecht</p>	<p>setzt mentales Modell der Arbeitsaufgabe voraus</p> <p>Einarbeitung erforderlich</p>

Abbildung 6.14: Vor- und Nachteile der alternativen Navigationsmöglichkeiten durch die Arbeitsaufgabe



bekannten Umgebung navigieren und sich dabei mit der Umgebung vertraut machen. Hierfür sollte der feste Weg mit weiteren Unterstützungsansätzen kombiniert werden. Vor allem Ansätze, die den Aufbau eines mentalen Modells der Arbeitsaufgabe fördern, sind hier sinnvoll zu sein. Benutzern, die die Arbeitsaufgabe häufiger durchführen, kann so schnell der räumliche Orientierungsstil ermöglicht werden. Eine sinnvolle Alternative oder Ergänzung zu einem festen Weg sind Ansätze, mit denen sich der Benutzer mit typische Wegen vertraut machen kann (siehe Abschnitt 6.4.5).

**Alternative feste Wege.** Auch die Auswahl eines alternativen festen Weges wird den Anforderungen der Arbeitsaufgabe nicht voll gerecht. Daher halte ich diese Variante auch für nicht geeignet, um Benutzer mit fundiertem Domänenwissen bei der Arbeit zu unterstützen. Für Anfänger kann es hingegen schwierig sein, vor Arbeitsbeginn eine angemessene Vorgehensweise auszuwählen. Erfolgt die Auswahl dagegen über das Ziel der Produktauswahl, wie es bei der Auswahl eines Schemas der Fall ist, so ist diese Alternative eine sinnvolle Unterstützung für Personen mit geringem Domänenwissen.

**Leitsystem.** Das Angebot eines „ausgeschilderten Weges“ und die Möglichkeit, jederzeit von diesem abzuweichen entspricht ausgeschilderten Routen in der realen Welt. Es bietet eine Hilfestellung, ohne dass der Handlungsspielraum des Benutzers eingeschränkt wird. Ich halte daher die Bezeichnung Leitsystem für diese Alternative für passend. Das Leitsystem verbindet die Vorteile von systemgeführter und benutzergeführter Interaktion. Den Anforderungen der Arbeitsaufgabe wird es gerecht, weil der Benutzer sein Vorgehen jederzeit der Situation anpassen kann. Außerdem unterstützt es beide von Passini beschriebenen Orientierungsstile. Deshalb halte ich das Leitsystem zur Unterstützung von gelegentlichen Benutzern und Experten für die beste Lösung. Diese Bewertung wird auch durch Paul gestützt. Auch er hält es oft für sinnvoll, beide Interaktionsformen zu entwickeln und den Benutzer über die für seine Arbeit passende Form entscheiden zu lassen (Paul 1994, S. 137). Passini spricht in diesem Zusammenhang von *wayfinding conditions* (Passini 1984, S. 164ff). Abhängig von diesen braucht der Benutzer unterschiedliche Unterstützung bei der Navigation. Beim Abweichen vom festgelegten Weg bestehen im Leitsystem jedoch die gleichen Probleme, wie bei der freien Navigation: Bei ungünstigem Vorgehen ist es möglich, dass der Benutzer keine oder nur unbefriedigende Lösungen findet.

In Abschnitt 6.4.1 wurde beschrieben, wie festgelegte Wege mit Hilfe der Raum-Metapher vermittelt werden können.

**Freier Weg.** Die letzte Alternative bietet dem Benutzer ausschließlich den freien Weg ohne Leitsystem an. Für Benutzer mit geringem Domänenwissen stellt diese Alternative einen erheblichen Nachteil dar, da sie nicht auf den linearen Orientierungsstil zurückfallen können, wie es von Passini gefordert wird. Im Widerspruch zu Passinis Forderungen wird nur der räumliche Orientierungs-

stil unterstützt. Auch für Benutzer mit umfangreichem Domänenwissen bietet das Leitsystem gegenüber dem freien Weg Effizienzvorteile. Im Leitsystem können sie einfach dem festgelegten Weg folgen und sich voll auf die Entscheidungen konzentrieren, wenn die Aufgabe kein anderes Vorgehen erfordert. Aus diesem Grund halte ich die Verbindung von freier Navigation und „Wegweisern“, wie sie das Leitsystem bietet, für sinnvoller. Wie bereits in Abschnitt 6.3 beschrieben, ist die direktmanipulative Produktauswahl immer mit einem freien Weg verbunden. Die *animation machine* und die *Szenario-Maschine* (siehe Abschnitt 6.4.5) bieten hier eine sinnvolle Möglichkeit, um dem Benutzer meist erfolgversprechende Vorgehensweisen zur Verfügung zu stellen.

Wie bereits in Abschnitt 3.3 beschrieben, bietet EngCon große Flexibilität bei der Festlegung der möglichen Abläufe. Alle hier erläuterten Alternativen und auch Mischformen können mit Hilfe des Kontrollwissens realisiert werden, das in der Wissensbasis spezifiziert wird.

## 6.5.2 Abkürzungen

Experten können durch Abkürzungen häufig wiederkehrende Teilaufgaben effizienter bearbeiten. Anfängern können Abkürzungen dagegen einfachere Wege durch die Produktauswahl bieten. Im Folgenden werden verschiedene Ansätze erläutert, mit denen Benutzern Abkürzungen angeboten werden können. Grob lassen sich *Beispiellösungen* und *Teillösungen* unterscheiden.

### Beispiellösungen

Bereits vollständig erstellte *Beispiellösungen* bieten dem Benutzer eine Abkürzung direkt zu den Arbeitsergebnissen. Ihr Angebot bietet sich für typische Standardlösungen an. Für jedes Beispiel sollten nicht nur die Arbeitsergebnisse zugänglich sein, sondern auch der Weg durch die Arbeitsaufgabe, der zu diesem Ergebnis geführt hat. Wird ein Beispiel geöffnet, so befindet sich das System im gleichen Zustand, als hätte der Benutzer das Produkt gerade selbst zusammengestellt. Auf diese Weise kann der Benutzer auch den *Lösungsweg erforschen*, wenn Unterstützungsansätze wie eine Historie und ein Undo angeboten werden. Im Gegensatz zu den Szenarien der *Szenario-Maschine* startet der Benutzer hier am Ziel, und der Handlungsspielraum entspricht dem der normalen Interaktion. Aus Sicht der Konfigurierung entspricht eine solche Beispiellösung einer vollständigen und konsistenten Konfiguration.

Besonders sinnvoll ist dieser Unterstützungsansatz in Verbindung mit dem Angebot, Lösungen direkt zu ändern (siehe hierzu Abschnitt 6.5.5) beziehungsweise in Verbindung mit einem referentiellen Undo (siehe hierzu Abschnitt 6.5.4). So

kann der Benutzer nicht nur die Arbeitsergebnisse selbst nutzen und den Lösungsweg erforschen, sondern die Lösung auch leicht variierenden Anforderungen anpassen.

Dieser Ansatz kann auch zur Unterstützung der Zusammenarbeit zwischen Vertriebsmitarbeitern und Kunden eingesetzt werden. Der Vertriebsmitarbeiter erstellt eine Beispiellösung für einen Kunden, der das Arbeitsergebnis untersucht und dann eventuell einzelne Änderungen an der Lösung durchführen kann. Bei Bedarf überprüft der Vertriebsmitarbeiter die vom Kunden modifizierte Lösung anschließend. Eine Kombination von Produktsicht und Anwendungssicht macht eine solche Überprüfung möglich.

Durch das Angebot von Beispiellösungen wird dem Benutzer der kürzeste Weg zu den Arbeitsergebnissen geboten. Der Weg ist noch schneller und einfacher als der feste Weg (siehe Abschnitt 6.5.1) oder die Wege, die durch ein Szenario angeboten werden. Den Vorteilen dieses Ansatzes stehen die Probleme, die beim Einsatz des referentiellen Undos oder des direkten Änderns auftreten können, gegenüber. Darüber hinaus entstehen durch diesen Ansatz aufgrund der Vorgehensweise keine innovativen Lösungen. Beispiellösungen können sowohl in der formularbasierten als auch in der direktmanipulativen Produktauswahl angeboten werden.

### **Teillösungen**

Mit Hilfe einer Teillösung kann der Benutzer eine Menge von Entscheidungen wiederholt bei der Produktauswahl nutzen.

**Schemata.** Neben Schemata, die bereits in Abschnitt 6.4.2 beschrieben wurden, können Teillösungen auch durch den Benutzer definiert werden.

**Benutzerdefinierte Teillösungen.** Häufig sind Vertriebsmitarbeiter auf eine spezielle Produktklasse spezialisiert. Treffen sie bei der Produktauswahl regelmäßig die gleichen Entscheidungen, so ist es sinnvoll, diese einmal festzulegen und anschließend immer wieder darauf zurückzugreifen. Während ein Schema die Struktur einer Produktklasse beschreibt, kann eine benutzerdefinierte Teillösung alle Entscheidungen betreffen. Benutzerdefinierte Teillösungen können wie eine normale Produktzusammenstellung erstellt werden. Dabei legt der Benutzer aber nur die Entscheidungen fest, die jeweils identisch sind. Die so definierte Teillösung wird gespeichert und kann anschließend immer wieder als Ausgangspunkt für die Produktauswahl verwendet werden. Aus Sicht der Konfigurierung definiert der Benutzer hierdurch eine Teilkonfiguration, die als Aufgabenstellung für zukünftige Produktzusammenstellungen dient. Auch benutzerdefinierte Teillösungen sind in der formularbasierten und in der direktmanipulativen Produktauswahl einsetzbar.

In der formularbasierten Produktauswahl können beim Einsatz von benutzer-

definierten Teillösungen zwei Varianten unterschieden werden. Erstens können die verbleibenden Entscheidungen dem Benutzer genauso präsentiert werden, als hätte er die Teillösung gerade erstellt. Beim Einsatz der Raum-Metapher (siehe Abschnitt 6.4.1) sind die restlichen Entscheidungen dann eventuell über die einzelnen Orte verstreut. Zweites können alle verbleibenden Entscheidungen auch an einem Ort angezeigt werden. Durch die Definition einer Teillösung schafft der Benutzer dann eine neue Navigationseinheit. Im Folgenden werden solche Navigationseinheiten als *Formulare* bezeichnet. Diese Variante hat den Vorteil, dass zur Produktauswahl wesentlich weniger Navigationsinteraktion notwendig ist. Die Formularerstellung kann um weitere Möglichkeiten ergänzt werden. Der Benutzer kann zusätzlich auch *Defaultwerte* für einzelne Entscheidungen definieren oder eine spezielle Sortierung für die Entscheidungen angeben. Außerdem kann ein solches Formular sinnvoll mit *benutzerdefinierten Konfigurierungseinheiten* kombiniert werden (siehe Abschnitt 6.5.3).

**Copy & Paste.** Schemata und benutzerdefinierte Teillösungen sind präventive Maßnahmen zur Wiederverwendung von Entscheidungen. Mit „Copy & Paste“ kann der Benutzer bereits getroffene Entscheidungen nachträglich wiederholt nutzen, ohne dies vorher festzulegen. Copy & Paste von Entscheidungen bietet sich vor allem in der direktmanipulativen Produktauswahl an. Hier kann der Benutzer eine Menge von Objekten auswählen und anschließend kopieren.

Im Anwendungsfall SVEA kann ein GA-System beispielsweise eine Reihe von identischen Beleuchtungsgruppen enthalten. Daher bietet es sich an, die Entscheidungen nicht wiederholt festzulegen, sondern sie einfach zu kopieren. Dies ist auch in der Kombination von direktmanipulativer und formularbasierter Produktauswahl einfach möglich (siehe Abbildung 6.4 auf Seite 91). In der rein formularbasierten Produktauswahl ist das „Recycling“ von Entscheidungen dagegen weniger anschaulich. Um dem Benutzer ein wiederholtes Festlegen von identischen Beleuchtungsgruppen zu ersparen, kann hier ein spezieller Button angeboten werden. Hat der Benutzer bereits eine Beleuchtungsgruppe festgelegt, kann er über diesen Button identische Beleuchtungsgruppen erzeugen.

### **Bewertung und Vergleich**

Zur Unterstützung von Anfängern und gelegentlichen Benutzern sind besonders Beispiellösungen und Schemata geeignet, weil sie den Weg durch die Produktauswahl vereinfachen. Den einfachsten Weg bieten Beispiellösungen. Kunden kann eine erste Vorstellung von Lösungsmöglichkeiten vermittelt werden, wobei innovative Lösungen nicht im Vordergrund stehen. Darüber hinaus können Beispiellösungen als Kommunikationsgrundlage zwischen Vertriebsmitarbeitern und Kunden eingesetzt werden.

Für Experten kann ein effizienterer Weg durch die Produktauswahl geboten

werden durch das Copy & Paste von Entscheidungen, durch benutzerdefinierte Teillösungen und durch Schemata. Benutzerdefinierte Teillösungen können auch in der Zusammenarbeit von Vertriebsmitarbeitern und Kunden eingesetzt werden. Benötigt ein Kunde regelmäßig ähnliche Produkte, so kann ihm der Vertriebsmitarbeiter für diese ein speziell zugeschnittenes Formular erstellen.

Auf die Realisierung von *Schemata* auf der Grundlage von EngCon wurde bereits in Abschnitt 6.4.2 eingegangen.

**Beispiellösungen.** Das Angebot von Beispiellösungen ist ein Ansatz, der dem *fallbasierten Konfigurieren* zuzurechnen ist<sup>4</sup>. Das Angebot einer kleinen Menge von Beispiellösungen kann sehr einfach durch eine Menge von vollständigen und konsistenten Konfigurationen realisiert werden. Bei nur wenigen Beispielen ist für die Auswahl kein Ähnlichkeitsmaß erforderlich. Für den Anfänger stellen sie trotzdem eine sinnvolle Hilfe dar, weil sie typische Lösungen und Lösungswege vermitteln. Soll dagegen eine große Menge von Beispielen angeboten werden, ist zur Auswahl ein Ähnlichkeitsmaß erforderlich.

**Benutzerdefinierte Teillösungen.** Das Speichern von Teilkonfigurationen und deren spätere Nutzung bei neuen Produktzusammenstellungen ist mit EngCon ohne weiteres möglich. Auch die Zusammenfassung der verbleibenden Entscheidungen in einem Formular kann realisiert werden. Sie erfordert nur eine abweichende Anzeige der Entscheidungen. Dazu ist eine entsprechende Erweiterung der Benutzungsschnittstelle notwendig. Diese wird jedoch unabhängig vom Konfigurierungskern EngCon realisiert (siehe Abbildung 3.5 auf Seite 41).

**Copy & Paste.** Beim Copy & Paste von Teillösungen bzw. Entscheidungen wird ein Top-Down-Vorgehen mit einem Bottom-up-Vorgehen kombiniert. Hierbei entstehen eine Reihe von Problemen. Das freie Kopieren einer beliebigen Teillösung in eine Teilkonfiguration ist unabhängig vom eingesetzten Konfigurierungskern nicht möglich, da die Teillösung zur Struktur der Teilkonfiguration passen muss. Selbst wenn dies durch eine Überprüfung sichergestellt wird, können durch das Einfügen Konflikte ausgelöst werden. Die Übernahme von strukturgleichen Teillösungen kann jedoch an einzelnen Punkten ermöglicht werden. Der oben beschriebene Button zum Erzeugen einer identischen Beleuchtungsgruppe ist durch eine Erweiterung mit EngCon realisierbar.

---

<sup>4</sup>Beim fallbasierten Konfigurieren wird zur Lösung des Konfigurierungsproblems auf bereits gelöste Aufgaben zurückgegriffen. Aus einer Fallbibliothek wird zunächst ein ähnlicher Fall ausgewählt und anschließend angepasst. Dazu wird ein Ähnlichkeitsmaß benötigt und eine geeignete Vorgehensweise zum Anpassen der Fälle. Für beides geeignete Methoden zu finden, macht die Schwierigkeit der fallbasierten Konfigurierung aus.

### 6.5.3 Schrittgröße

Benutzergruppen unterscheiden sich in ihren Anforderungen an die „Schrittgröße“ bei der Produktauswahl. Anfänger benötigen ein kleinschrittiges Vorgehen, während Experten für ein möglichst effizientes Vorgehen auf größere Schritte angewiesen sind. Vom Konfigurierungssystem werden Entscheidungen ermittelt, die von den bereits getroffenen Entscheidungen durch taxonomische Inferenz und Constraint-Propagation eindeutig ableitbar sind. Eine Menge von Entscheidungen, nach der diese Auswirkungen der Benutzerentscheidungen berechnet und angezeigt werden, bezeichne ich im Folgenden als *Konfigurierungseinheit*. Um der Anforderung nach sofortigem Feedback gerecht zu werden (siehe Abschnitt 6.1.2), müssen solche Folgeentscheidungen nach jeder Entscheidung des Benutzers sofort ermittelt und angezeigt werden.

Günter (1992, S. 157) beschreibt, dass die Propagation relativ aufwändig sein kann. Daher kann es für Benutzer in einzelnen Fällen sinnvoll sein, auf sofortiges Feedback zu verzichten, wenn sie eine genaue Vorstellung von den Auswirkungen ihrer Entscheidungen haben. Prinzipiell verringern kleine Konfigurierungseinheiten die Wahrscheinlichkeit von Konflikten und liefern schnelles Feedback, während, größere Konfigurierungseinheiten Effizienzvorteile bieten.

Bei der Definition der Konfigurierungseinheiten können drei Alternativen unterschieden werden. Diese werden im Folgenden vorgestellt und bezüglich ihrer Eignung für unterschiedliche Benutzergruppen bewertet.

**Konfigurierungseinheit entspricht einer Entscheidung.** Sobald der Benutzer eine Entscheidung getroffen hat, werden die Auswirkungen vom System berechnet und angezeigt. Diese Alternative bietet dem Benutzer ein schnelles Feedback zu den Auswirkungen seiner Entscheidungen und minimiert die Wahrscheinlichkeit von Konflikten. Bei Navigationseinheiten von mehr als einer Entscheidung können so auch Entscheidungen experimentell in Abhängigkeit voneinander getroffen werden. Außerdem wird keine Interaktion zum Auslösen der Konfigurierung notwendig. Damit der Benutzer seine Entscheidung als Ursache für die Änderung von Werten wahrnimmt, müssen allerdings kurze Antwortzeiten sichergestellt werden. Laut Nielsen (1993, S. 135) sind hier Werte von unter einer Sekunde, optimaler Weise unter 0,1 Sekunde notwendig. Dies kann zum Beispiel bei einer Konfigurierung über das Internet nicht sichergestellt werden.

**Konfigurierungseinheiten werden durch den Benutzer definiert.** Die Auswirkungen der Benutzerentscheidungen werden erst dann berechnet und angezeigt, wenn der Benutzer dies durch einen Button auslöst. So kann der Benutzer die Konsistenz seiner Lösung gezielt nur an einzelnen Schlüsselpunkten kontrollieren. Er kann die Größe seiner Schritte also flexibel anpassen. Im Extremfall setzt der Benutzer bei dieser Alternative den Konfigurator nur zur abschließenden Konsistenzprüfung ein, nachdem er die gesamte Lösung definiert hat. Benutzer-

definierte Konfigurierungseinheiten sind nicht effizient, wenn kleine Konfigurierungseinheiten realisiert werden sollen. Außerdem besteht die Gefahr, dass der Benutzer eine ganze Reihe von Entscheidungen vergeblich trifft, weil Konflikte mit Verzögerung aufgedeckt werden. Dies kann passieren, wenn der Benutzer vergisst, die Konfigurierung auszulösen oder die Auswirkungen seiner Entscheidungen falsch einschätzt. Der Einsatz von benutzerdefinierten Konfigurierungseinheiten ist also besonders dort sinnvoll, wo der Benutzer die Auswirkungen seiner Entscheidungen sehr gut einschätzen kann, was bei den in Abschnitt 6.5.2 beschriebenen Formularen gegeben ist. Eine Kombination dieser beiden Unterstützungsansätze bietet sich daher an.

**Konfigurierungseinheit entspricht einer Navigationseinheit.** Bildet jede Navigationseinheit eine Konfigurierungseinheit, so kann der Benutzer den Wechsel zur nächsten Navigationseinheit und das Feedback zu seinen Entscheidungen über einen gemeinsamen Button auslösen. Werden Konfigurierung und Navigation über einen gemeinsamen Button ausgelöst, so ist ein Vergessen der Konfigurierung nicht möglich. Neben einer Reduzierung der notwendigen Interaktion erfordert diese Alternative auch nicht so geringe Antwortzeiten wie die erste Alternative. Diesen Vorteilen stehen jedoch erhebliche Nachteile gegenüber. Ein gezieltes Experimentieren mit Entscheidungen ist nicht effizient möglich, weil Auswirkungen von Entscheidungen jeweils erst in der nächsten Navigationseinheit sichtbar werden. Auch ist der Zusammenhang zwischen einer Entscheidung und ihren Auswirkungen nicht klar erkennbar. Bei größeren Navigationseinheiten steigt zusätzlich die Wahrscheinlichkeit von Konflikten.

Für Personen mit geringem Domänenwissen sind Konfigurierungseinheiten von einer Entscheidung notwendig, damit sie die Zusammenhänge, die in der Domäne bestehen, erkennen können, und die Wahrscheinlichkeit von Konflikten möglichst gering ist. Bei Experten kann es ,zugunsten eines effizienteren Vorgehens, sinnvoll sein, die Anforderung des sofortigen Feedbacks zu vernachlässigen. Die letzte Variante halte ich insbesondere für Anfänger und gelegentliche Benutzer für weniger geeignet, da die Zusammenhänge zwischen Entscheidungen und ihren Auswirkungen nicht erkennbar sind. Der Benutzer kann sich so kein neues Wissen erschließen. Gegen diese Alternative spricht auch die erhöhte Wahrscheinlichkeit von Konflikten bei größeren Navigationseinheiten. Werden in einer Navigationseinheit nur voneinander unabhängige Entscheidungen gruppiert, so kann dieses Risiko reduziert werden. Eine solche Gruppierung kann jedoch einer benutzerorientierten Gruppierung der Entscheidungen entgegenstehen. Eine Realisierung der beschriebenen Konfigurierungseinheiten ist mit EngCon möglich.

### 6.5.4 Undo-Mechanismen

Zum Abschluss dieses Kapitels werden in den folgenden Abschnitten drei Ansätze beschrieben, mit denen der Benutzer im Arbeitsprozess zurückwandern kann oder zurückliegende Entscheidungen ändern kann: *Undo-Mechanismen*, das *direkte Ändern von Entscheidungen* und *Wiederaufsetzpunkte*. Aus technischer Sicht stellen diese Ansätze ähnliche Anforderungen und bieten vergleichbare Möglichkeiten zum Ändern des aktuellen Teilproduktes. Aus Benutzersicht ist es aber sinnvoll, diese Ansätze zu unterscheiden. In diesem Abschnitt wird zunächst auf Undo-Mechanismen eingegangen.

Die Bedeutung einer universellen Undo-Funktion wird von vielen Autoren betont (Nielsen 1993, S. 138ff), (Cooper 1995, S. 465ff), (Mayhew 1992, S. 529), (Paul 1994, S. 174ff). Durch die Möglichkeit, Handlungsschritte zu stornieren, wird der Benutzer beim explorativen Arbeiten unterstützt (Nielsen 1993, S. 138), (Cooper 1995, S. 467), (Paul 1994, S. 174ff). Er arbeitet schneller und ist dabei weniger gestresst (Mayhew 1992, S. 529). Nach Cooper sollte die Undo-Funktion als *Werkzeug zur Exploration* und nicht als *Mittel zur Fehlerbehebung* gesehen werden. Aufgrund der Merkmale der Arbeitsaufgabe Produktauswahl halte ich diesen Punkt für sehr wichtig. Der Benutzer sollte also durch einen entsprechenden Undo-Mechanismus beim explorativen Arbeiten unterstützt werden.

In der Literatur werden eine Vielzahl unterschiedlicher Undo-Mechanismen beschrieben. Im Folgenden werden einige wichtige Undo-Mechanismen kurz vorgestellt – das *retract undo*, das *travel undo*, das *group multiple undo* und das *referentielle Undo*. Anschließend werden sie bezüglich der Unterstützung, die sie Benutzern bei der Auswahl komplexer Produkte bieten, bewertet.

**Retract Undo.** Durch das retract undo wird der Systemzustand vor der Ausführung des letzten Handlungsschrittes wieder hergestellt (Herczeg 1986, S. 130). Es kann mit dem chronologischen Backtracking verglichen werden. Die Historie wird um den jeweiligen Schritt gekürzt, wie in Abbildung 6.15 veranschaulicht wird. Da nach dem Undo keine Informationen über den stornierten Schritt mehr verfügbar sind, kann das Undo selbst nicht rückgängig gemacht werden – es ist irreversibel. Mehrere Schritte können in umgekehrter Ausführungsreihenfolge storniert werden.

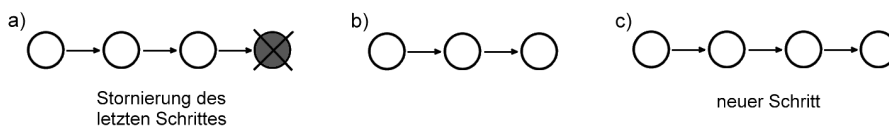


Abbildung 6.15: retract undo



**Travel Undo.** Im Gegensatz zum retract undo ist das travel undo reversibel, es kann also rückgängig gemacht werden (Herczeg 1986, S. 130). Undo und Redo beruhen auf einer LIFO-Struktur. Hierdurch kann beim Redo ein Problem entstehen: Ein stornierter Schritt kann in einem anderen Systemzustand als dem der ursprünglichen Durchführung wieder durchgeführt werden (Redo). Es besteht das gleiche Problem wie beim Backtracking mit Datenübernahme. Es kann nicht vorausgesetzt werden, dass der Schritt vom neuen Zustand aus durchführbar ist, wie in Abbildung 6.16 d) angedeutet wird.

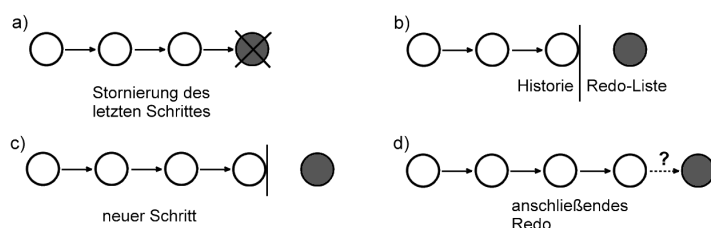


Abbildung 6.16: travel undo

**Group multiple Undo.** Das group multiple undo ist eine Abwandlung des travel undo und wird von Cooper beschrieben (Cooper 1995, S. 479). Es wird zum Beispiel in MS Word für Windows 6.0 eingesetzt, und erlaubt auch das Stornieren mehrerer Aktionen in einem Schritt. Hierfür ist eine Interaktionshistorie erforderlich. Das oben beschriebene Problem des travel undo wird beim group multiple undo durch ein Löschen der Redo-Liste gelöst. Nach dem Stornieren von einem oder mehreren Schritten sind diese in der Redo-Liste aufgeführt. Über die Redo-Liste kann die Stornierung der Schritte rückgängig gemacht werden. Wird jedoch ein neuer Handlungsschritt durchgeführt, so wird die Redo-Liste gelöscht. Die Stornierung der Schritte kann nicht länger widerrufen werden. Hierdurch wird verhindert, dass Schritte in einem anderen als dem ursprünglichen Systemzustand durchgeführt werden. Dies wird in Abbildung 6.17 veranschaulicht. Dieses Ver-

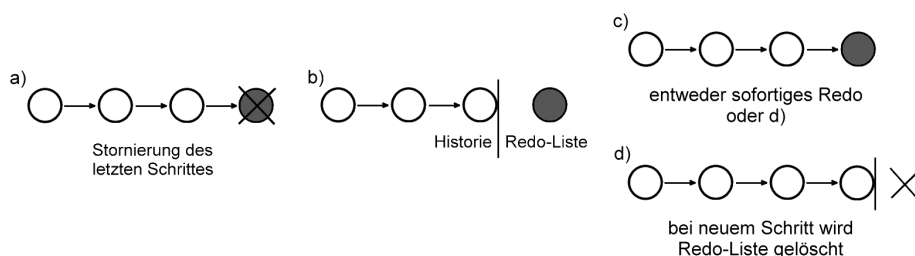


Abbildung 6.17: group multiple undo

fahren liegt auch dem Undo-Mechanismus des *Drive Solution Designer* (DSD) zugrunde. Dort wird jedoch auf das Angebot einer Interaktionshistorie verzichtet, so dass nur das Undo bzw. Redo des jeweils letzten Schrittes möglich ist.

**Referentielles Undo.** Alle bisher beschriebenen Undo-Mechanismen haben einen erheblichen Nachteil: Soll ein weiter zurückliegender Handlungsschritt storniert werden, so müssen auch alle zeitlich nachfolgenden Schritte widerrufen werden. Im Gegensatz zu diesen Undo-Mechanismen erlaubt das referentielle Undo, beliebige zurückliegende Schritte separat zu stornieren (Paul 1994, S. 176), wie Abbildung 6.18 verdeutlicht. Das referentielle Undo kann mit der Reparatur einer Teilkonfiguration verglichen werden (siehe hierzu Abschnitt 3.2). Damit der

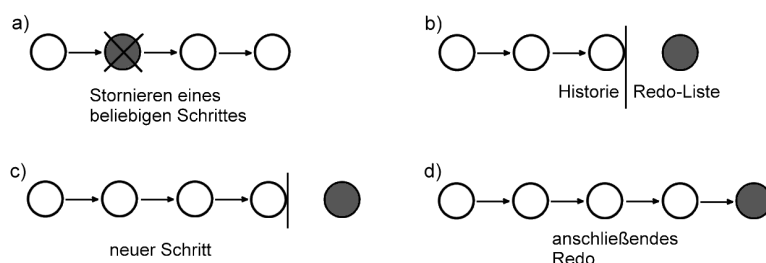


Abbildung 6.18: Referentielles Undo

Benutzer den zu stornierenden Schritt auswählen kann, benötigt er eine Interaktionshistorie oder einen Interaktionsgraphen. Außerdem entsteht durch das referentielle Undo das gleiche Problem, das schon beim travel undo erläutert wurde. Wird ein zurückliegender Schritt separat storniert, so müssen die nachfolgenden Schritte in einem anderen als dem ursprünglichen Systemzustand durchgeführt werden. Wie mit Schritten zu verfahren ist, die auf dem stornierten Schritt aufbauen, kann laut Paul nicht pauschal entschieden werden. Dies sei nur in Abhängigkeit von Arbeitsaufgabe und Funktion entscheidbar (Paul 1994, S. 177). Das gleiche Problem besteht beim Redo solcher stornierten Schritte.

### Vergleich und Bewertung

Anhand einer Reihe von Merkmalen kann begründet werden, wie gut die einzelnen Undo-Mechanismen zur Unterstützung der Produktauswahl geeignet sind.

Um eine explorative Produktauswahl zu ermöglichen, sollte der Benutzer den zurückgelegten Weg wiederholt durchwandern können. Dies ist nur durchführbar, wenn auch ein Redo angeboten wird – das Undo also *reversibel* ist. Das retract undo ist daher ungeeignet.

Für ein effizientes Arbeiten sollten auch mehrere Schritte mit einer Aktion storniert werden können. Diese Forderung wird auch durch Yang (1990) gestützt.

Voraussetzung für das gleichzeitige Stornieren mehrerer Schritte ist eine Interaktionshistorie oder ein Interaktionsgraph. Werden weder Interaktionshistorie noch -graph angeboten, so spricht Cooper von einem *blind undo* (Cooper 1995, S. 470). Als blind undo können realisiert werden: das retract undo, das travel undo und die einfachere Form des group multiple undo, wie sie im DSD eingesetzt wird. Diese letzte Variante halte ich dabei für die beste Wahl: Das travel undo ist ohne eine Anzeige von Undo-Liste und Redo-Liste für den Benutzer nicht durchschaubar. Das retract undo ist ungeeignet, da es selbst nicht rückgängig gemacht werden kann.

Eine optimale Unterstützung der explorativen Produktauswahl ist nur gegeben, wenn auch weiter zurückliegende Schritte separat storniert werden können, wie es das referentielle Undo ermöglicht. Das referentielle Undo wird auch von vielen anderen Autoren als optimale Unterstützung genannt (Berlange 1994), (Cooper 1995, S. 467), (Paul 1994, S. 176), (Yang 1988). Die Autoren verwenden jedoch jeweils ein eigene Bezeichnung für diesen Undo-Mechanismus.

Dem bedeutenden Vorteil des referentiellen Undos stehen jedoch einige Nachteile gegenüber. Durch das Undo wird nicht zu einer alten Teilkonfiguration zurückgesprungen, sondern eine neue Teilkonfiguration definiert. Das separate Stornieren eines zurückliegenden Schrittes kann einige der nachfolgenden Schritte unmöglich machen oder nicht mehr sinnvoll. Storniert der Benutzer zum Beispiel die Entscheidung, dass das GA-System eine Beleuchtungssteuerung umfassen soll, so werden alle Entscheidungen, die sich auf diese Beleuchtungssteuerung beziehen, unmöglich. Andere Entscheidungen sind vielleicht weiterhin möglich, im neuen Kontext aber nicht mehr sinnvoll. Wie mit Schritten zu verfahren ist, die auf dem stornierten Schritt aufbauen, kann laut Paul nicht pauschal entschieden werden (Paul 1994, S. 177). Für die Auswahl komplexer Produkte halte ich die folgenden Alternativen für sinnvoll:

1. Die nicht mehr möglichen Schritte werden ebenfalls storniert und sind in der Redo-Liste enthalten.
2. Der Benutzer bekommt die nicht mehr möglichen Schritt präsentiert und kann entscheiden, ob er die Stornierung aller Schritte durchführen will oder nicht.

Durch das Redo können einzelne zurückliegende Entscheidungen geändert werden, wodurch auch Konflikte auftreten können. Das referentielle Undo kann daher als ein sehr mächtiger Undo-Mechanismus bezeichnet werden, der aber nur mit fundiertem Domänenwissen sinnvoll genutzt werden kann.

Ein weiteres wichtiges Merkmal ist die Menge der stornierbaren Entscheidungen. So kann unterschieden werden zwischen Undo-Mechanismen, die das Stornieren aller Benutzerentscheidungen erlauben und solchen, die zusätzlich auch

das Stornieren der vom System ermittelten Auswirkungen erlauben. Da solche Auswirkungen aus den Entscheidungen des Benutzers folgen, können sie nicht separat storniert werden. Es müssen immer auch die Benutzerentscheidungen widerrufen werden, die die Ursache dieser Folgeentscheidungen sind. Ist das direkte Stornieren von Folgeentscheidungen nicht möglich, so muss der Benutzer, wenn eine Folgeentscheidung seinen Anforderungen widerspricht, zunächst die ursächlichen Entscheidungen ermitteln, um diese dann zu ändern. Dies kann bei komplexen Abhängigkeiten zwischen Entscheidungen aufwändig und frustrierend sein. Stattdessen sollte der Benutzer beim Stornieren einer angezeigten Folgeentscheidung die ursächlichen Entscheidungen direkt aufgelistet bekommen und die Option erhalten, eine oder mehrere dieser Entscheidungen aufzuheben oder die Folgeentscheidung bestehen zu lassen. Die Stornierung von Folgeentscheidungen, ist ein sehr wirkungsvoller Unterstützungsansatz, der allerdings nur in Verbindung mit einer Interaktionshistorie oder einem Interaktionsgraphen sinnvoll ist.

Welcher Undo-Mechanismus am angebrachtensten ist, hängt auch von den Navigationsmöglichkeiten des Benutzers ab. Beim Angebot eines festen Weges oder alternativer fester Wege sollte ein *group multiple undo* eingesetzt werden. Ein referentielles Undo ist hier unpassend, da es die festgelegte Reihenfolge für die Entscheidungen wieder aufheben würde. Im Leitsystem und im freien Weg bietet das referentielle Undo dem Benutzer dagegen die weitest reichende Unterstützung.

Das *group multiple undo* kann auf der Basis von EngCon realisiert werden und wird in der vereinfachten Form ohne Interaktionshistorie bereits im DSD eingesetzt. Auch das referentielle Undo kann realisiert werden. Wird eine Entscheidung storniert, so müssen auch alle Auswirkungen dieser Entscheidung wieder rückgängig gemacht werden. Hierfür gibt es zwei Lösungen: Alle nicht stornierten Benutzerentscheidungen können für einen neuen Konfigurierungsdurchlauf eingesetzt werden. Da so alle Auswirkungen neu berechnet werden, ist die Konsistenz der Lösung sichergestellt. Die andere Lösung setzt ein Reasoning Maintenance System (RMS) voraus. Ein solches RMS verwaltet die Abhängigkeiten zwischen Entscheidungen; so können die Auswirkungen der stornierten Entscheidung direkt ermittelt und aufgehoben werden. Dies kann jedoch mit einem erheblichen Rechenaufwand verbunden sein. Abhängigkeiten werden von EngCon bisher nicht verwaltet.

Um das oben beschriebene direkte Stornieren von angezeigten Folgeentscheidungen zu ermöglichen, müssen die bestehenden Abhängigkeiten genauso bekannt sein. Um den Unterstützungsansatz zu realisieren, müsste EngCon also mit einem RMS zur Verwaltung der Abhängigkeiten kombiniert werden.

### 6.5.5 Direktes Ändern von Entscheidungen

Mit Hilfe des direkten Ändern kann eine bereits getroffene Entscheidung unmittelbar neu getroffen werden. Dieser Ansatz ist genau wie eine Undo-Funktion sowohl in der direktmanipulativen als auch in der formularbasierten Produktauswahl sinnvoll. In der formularbasierten Produktauswahl kann der Benutzer beispielsweise die Entscheidung „Beleuchtungstyp = dimmbar“ direkt ändern in „Beleuchtungstyp = schaltbar“. In der direktmanipulativen Produktauswahl löscht der Benutzer zunächst das Objekt „dimmbare Beleuchtungsgruppe“ und fügt anschließend ein Objekt „schaltbare Beleuchtungsgruppe“ ein.

Der Ansatz des direkten Ändern von Entscheidungen ermöglicht das effiziente Ändern von Entscheidungen und fördert das Experimentieren mit Alternativen. Technisch gesehen ist das direkte Ändern vergleichbar mit dem referentiellen Undo. Die oben beschriebene Änderung kann der Benutzer auch mit einer Undo-Funktion durchführen. Warum ist dann der Ansatz des direkten Ändern trotzdem eine sinnvolle Unterstützung? Das direkte Ändern von Entscheidungen ist *intuitiver* und *effizienter*, wenn der Benutzer weiß, wie er eine Entscheidung treffen möchte.

Der Benutzer hat beispielsweise in der formularbasierten Auswahl die Anzahl der Beleuchtungsgruppen zunächst auf zwei festgelegt und möchte sie jetzt auf drei ändern. Mit Hilfe des referentiellen Undos müsste er zuerst die entsprechende Entscheidung stornieren. Hierbei würden auch alle Entscheidungen zu diesen Beleuchtungsgruppen widerrufen werden, und anschließend könnte der Benutzer die Anzahl der Beleuchtungsgruppen auf drei festlegen. Um nicht alle seine Entscheidungen zu den ersten beiden Beleuchtungsgruppen erneut treffen zu müssen, kann er diese durch ein Redo zurückholen. Durch ein direktes Ändern ist dies in einem Schritt möglich. Darüber hinaus ist es intuitiver eine Entscheidung sofort zu ändern und nicht zunächst zu stornieren, wenn bereits klar ist, wie sie getroffen werden soll.

Das direkte Ändern von Entscheidungen stellt jedoch keinen Ersatz für eine Undo-Funktion dar, da es nicht die Option bietet Entscheidungen nur zu stornieren und dann zunächst offen zu lassen.

Die direktmanipulative Produktauswahl beruht auf dem direkten Ändern von Entscheidungen. Sie kann nicht ohne diesen Unterstützungsansatz realisiert werden. Aber auch in der formularbasierten Produktauswahl sollte dieser Ansatz eingesetzt werden. Welche Entscheidungen der Benutzer direkt ändern können sollte, ist hier von der Flexibilität im Vorgehen abhängig. Hat der Benutzer einen festen Weg durch die Aufgabe gewählt, so sollte auch nur die letzte Entscheidung direkt änderbar sein. Im Leitsystem oder im freien Weg sollten dagegen alle bereits getroffenen Entscheidungen änderbar sein. Hierbei können, genau wie beim referentiellen Undo, auch Konflikte ausgelöst werden. Um das direkte Ändern von

Entscheidungen erfolgreich zu nutzen, muss der Benutzer also die Auswirkungen seiner Änderungen einschätzen können.

Durch das direkte Ändern definiert der Benutzer eine neue Teilkonfiguration. Die Konsistenz dieser Teilkonfiguration kann, genau wie beim referentiellen Undo, überprüft werden, indem sie als neue Aufgabenstellung interpretiert wird.

### 6.5.6 Wiederaufsetzpunkte

Von Paul (1994, S. 179) ist der Wiederaufsetzpunkt definiert als ein „gesicherter Zustand eines interaktiven Systems, zu dem nach jedem Handlungsschritt beliebig oft zurückgesprungen werden kann.“ Wiederaufsetzpunkte sind ein Ansatz, mit dem das explorative Arbeiten unterstützt werden kann. Im Gegensatz zur Stornierung und dem Ändern von Entscheidungen stellen Wiederaufsetzpunkte eine präventive Maßnahme dar. Bevor der Benutzer die Exploration startet, sichert er den aktuellen Zustand mit einem Wiederaufsetzpunkt. Das Aufrufen eines Wiederaufsetzpunktes ist verwandt mit dem interaktiven Backtracking (siehe Abschnitt 3.2). Im Gegensatz zu diesem wird der Rücksprungpunkt aber im Voraus festgelegt.

Wiederaufsetzpunkte bieten die Möglichkeit, mit einem Schritt eine ganze Reihe von Schritten zurückzunehmen, was auch beim Auftreten eines Konfliktes sinnvoll genutzt werden kann. Zusätzlich fördern sie das Erarbeiten von Alternativen. Hierfür sichert der Benutzer zunächst die Ausgangslage durch einen Wiederaufsetzpunkt und untersucht anschließend die erste Alternative. Erscheint diese erwägenswert, so kann er sie durch einen weiteren Wiederaufsetzpunkt sichern. Nachfolgend erforscht er weitere Alternativen und verfährt hierbei analog. Hat der Benutzer alle Alternativen untersucht, kann er durch Auswahl des entsprechenden Wiederaufsetzpunktes die beste Lösung weiterverfolgen. Dieses Vorgehen wird in Abbildung 6.19 veranschaulicht.

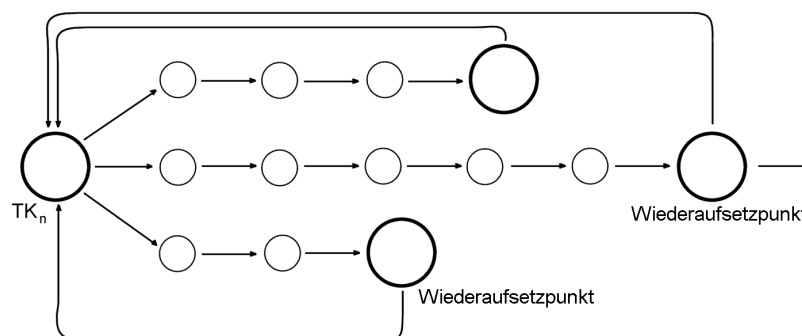


Abbildung 6.19: Erforschen von Alternativen durch Wiederaufsetzpunkte

Zum Rücksprung sieht Paul die Auswahl eines Wiederaufsetzpunktes aus dem Interaktionsgraphen vor. Diese Variante hat meines Erachtens einige Nachteile. Einerseits erfordert sie eine relativ aufwändige Realisierung; andererseits ist auch die Nutzung der Wiederaufsetzpunkte selbst sehr aufwändig: Der potentiell große Umfang von Interaktionsgraphen erschwert ihre Nutzung. Dies beschreibt auch Paul (1994, S.182). Außer Filtern hält er daher auch Navigationshilfen und Editorfunktionen für den Interaktionsgraphen für notwendig. Sie sollen die Orientierung in den Graphen ermöglichen.

Da Wiederaufsetzpunkte im Gegensatz zu Backtrackingpunkten präventiv gesetzt werden, sind sie auch ohne einen Interaktionsgraphen nutzbar. Eine von mir entworfene Alternative realisiert Wiederaufsetzpunkte mit Hilfe des Überblicks. Hierzu wird der Überblick um „Speicherplätze“ für Wiederaufsetzpunkte erweitert, wie Abbildung 6.20 zeigt. Setzt der Benutzer einen Wiederaufsetzpunkt, so

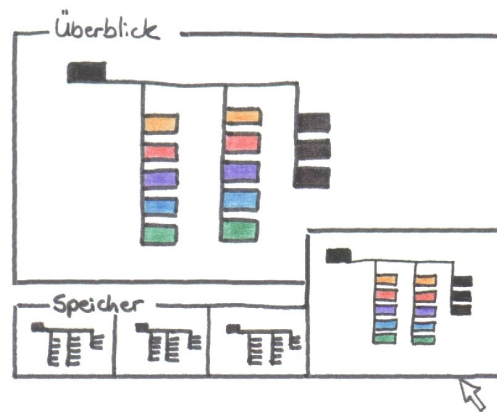


Abbildung 6.20: Über den Überblick realisierte Wiederaufsetzpunkte

„wandert“ eine Kopie des aktuellen Überblicks in einen der Speicherplätze. Durch die Kennzeichnung des aktuellen Standortes sowie der bereits erledigten und der noch offenen Teilaufgaben kann der Benutzer die einzelnen Wiederaufsetzpunkte unterscheiden. Um die Wiederaufsetzpunkte im Speicher erkennbar zu machen, sollten sie vergrößert dargestellt werden, wenn der Mauszeiger über der Grafik positioniert ist (siehe Abbildung 6.20). Diese Lösung bietet zwar nur eine begrenzte Anzahl von Wiederaufsetzpunkten, sie ist dafür aber wesentlich einfacher benutzbar und realisierbar.

Im Leitsystem kann es auch sinnvoll sein, einen Wiederaufsetzpunkt automatisch zu setzen. Verlässt der Benutzer den „ausgeschilderten Weg“, so wird der aktuelle Zustand gesichert. Treten bei der freien Navigation später Probleme auf, so kann der Benutzer zu diesem Punkt zurückspringen.

Unabhängig von der konkreten Realisierung halte ich Wiederaufsetzpunkte für einen wirkungsvollen Ansatz, um Benutzer beim Erarbeiten von Alternativen zum Produktvergleich zu unterstützen. Die von Paul beschriebene Variante erfordert eine gewisse Einarbeitung, stellt aber auch ein sehr mächtiges Werkzeug dar. Für Experten bietet sie die am weitesten reichende Unterstützung. Zur Unterstützung von gelegentlichen Benutzern halte ich dagegen die von mir beschriebene Abwandlung für sinnvoller. Diese hat außerdem den Vorteil, auch ohne einen Interaktionsgraphen realisierbar zu sein. Setzt der Benutzer einen Wiederaufsetzpunkt, so wird die aktuelle Teilkonfiguration gesichert. Beim Angebot von fünf „Speicherplätzen“ müssten also maximal fünf Teilkonfigurationen gespeichert werden. So können dem Benutzer schon mit sehr geringem Aufwand Wiederaufsetzpunkte geboten werden.

Im folgenden Kapitel wird der Einsatz einer Auswahl von Unterstützungsansätzen im Anwendungsfall SVEA beschrieben.



# Kapitel 7

## Der Prototyp

Aufbauend auf der Skizze des Gesamtsystems in Abschnitt 7.1 wird im zweiten Abschnitt detailliert die entworfene Benutzungsschnittstelle vorgestellt. In Abschnitt 7.3 werden dann die Ergebnisse des Benutzertests präsentiert und anschließend wird kurz auf die erstellte Wissensbasis eingegangen. Zum Abschluss dieses Kapitels schildere und begründe ich mein Vorgehen im Rahmen des Prototypentwurfes.

### 7.1 Systemskizze

Ziel des Systems ist das interaktive Erstellen von Kostenschätzungen und Ausschreibungen für LON-basierte GA-Systeme. Die Architektur des Systems entspricht dem allgemeinen Aufbau von EngCon-basierten Konfiguratoren (siehe auch Abschnitt 3.3): Auf die *Benutzungsschnittstelle* wird im nächsten Abschnitt genauer eingegangen. Die Erstellung fehlerfreier GA-Systeme wird durch den Konfigurierungskern *EngCon* sichergestellt. Das hierfür notwendige Domänenwissen ist in einer *Wissensbasis* spezifiziert.

Mit dem System können GA-Systeme zusammengestellt werden, die 1-10 Raumkonzepte umfassen. Jedes der Raumkonzepte kann eine Beleuchtungssteuerung, eine Sonnenschutzsteuerung und eine Einzelraumregelung sowie die Bedienung der Raumfunktionen umfassen. Eine Beleuchtungssteuerung kann dabei aus bis zu 10 Beleuchtungsgruppen bestehen. Die erforderliche Netzwerktechnik wird vom System berechnet.

Um ein GA-System zusammenzustellen, gibt der Benutzer die funktionalen Anforderungen an, die das GA-System erfüllen soll (z.B. Anzahl und Art der Jalousien, Präsenzerfassung über Taster oder Sensor, etc.). Einen Überblick über die im Prototyp vorgesehenen Entscheidungen gibt Abbildung D.3 im Anhang auf Seite 175. Das System bietet drei Arbeitsergebnisse:

- **Kostenschätzung:** Gerätestückliste mit Preisangaben,
- **Massenleistungsverzeichnis:** Gerätestückliste mit detaillierten Beschreibungen der Geräte und ohne Preisangaben,
- **Funktionale Beschreibung des GA-Systems:** Gibt die vom Benutzer spezifizierten Anforderungen in Textform wieder.

Die funktionale Beschreibung hat ein Textformat (.rtf oder .doc) und die Kostenschätzung und das Massenleistungsverzeichnis haben das aktuelle GAEB-Format (GAEB2000). GAEB2000 ist wie die von EngCon verwendete Wissensbasis XML-basiert<sup>1</sup>.

Das System sollte auf Standard-Windows-PCs lokal installiert werden können, da ein Internetzugang nicht bei allen Benutzern vorausgesetzt werden kann. Aus dem gleichen Grund sollte das System neben einer Distribution über die SVEA-Homepage auch auf CD angeboten werden.

## 7.2 Benutzungsschnittstelle

Bei dem entworfenen Prototyp handelt es sich um Papier-Mockups, die die Benutzungsschnittstelle des Systems beschreiben. Die hier abgebildeten Screens wurden auch im Benutzertest eingesetzt. Sie sind aus einer ganzen Reihe von einzelnen Teilen zusammengesetzt, was in den folgenden Abbildungen nicht direkt zu erkennen ist.

### Formularbasierte oder direktmanipulative Produktauswahl

Da alle befragten Benutzer die Arbeitsaufgabe Produktauswahl als ein *Zusammenfügen eines Systems aus Komponenten* beschrieben haben, liegt der Einsatz einer direktmanipulativen Produktauswahl nahe, die diese Vorstellung gut aufgreifen kann. Gegen eine direktmanipulative Produktzusammenstellung spricht nicht nur die große Anzahl der zu unterscheidenden Objekte, für die die Benutzer bisher keine grafischen Repräsentationen kennen. Auch die Tatsache, dass diese Objekte oft sehr ähnlich sind, erschwert eine geeignete grafische Präsentation. So wird zum Beispiel zwischen schaltbaren und dimmbaren Beleuchtungsgruppen und Beleuchtungsgruppen mit Konstantlichtregelung unterschieden. Für dimmbare Beleuchtungsgruppen und die Konstantlichtregelung ist weiterhin zwischen

---

<sup>1</sup>Zu weiteren Informationen zum GAEB-Format siehe auch:  
<http://www.gaeb.de>; zuletzt besucht: 02.05.2003,  
<http://www.mwm.de/gaeb/gaeb1.html>; zuletzt besucht: 02.05.2003,  
<http://www.mwm.de/presse/pr0901.html>; zuletzt besucht: 02.05.2003.

sechs unterschiedlichen Leuchtenarten zu differenzieren. Aus diesen Gründen beruht die Produktauswahl im Prototyp überwiegend auf einer *formularbasierten Oberfläche*.

Die in Abschnitt 6.3 beschriebene Kombination der Interaktionstechniken halte ich im Anwendungsfall SVEA auch für eine sinnvolle Lösung. Da ich diesen Ansatz jedoch erst nach dem letzten Benutzertest entwickelt habe, konnte er in den Prototypentwurf nicht einfließen.

Einer der wesentlichen Vorteile der direkten Manipulation – schnelle, inkrementelle und leicht umkehrbare Operationen, deren Effekt sofort sichtbar wird – lässt sich auch mit einer formularbasierten Oberfläche realisieren. Um dieses Ziel zu erreichen, spielen mehrere Faktoren eine wichtige Rolle: Durch *Konfigurierungseinheiten* von jeweils einer Entscheidung wird schnelles Feedback möglich. Die Auswirkungen von Entscheidungen werden im Prototyp zum Beispiel über die Anzeige der bisherigen (Teil-)Lösung verdeutlicht. Durch einen *Undo-Mechanismus* können alle Operationen einfach storniert werden. Auf diese Aspekte wird später genauer eingegangen.

### **Navigationsmöglichkeiten durch die Arbeitsaufgabe**

Da für das Zusammenstellen einer LON-basierten GA-Lösung die Definition einer festen Vorgehensweise nicht sinnvoll ist, bietet der Prototyp ein Leitsystem an. Dieses Leitsystem wird im Prototyp als *freier Weg* bezeichnet und bietet die Flexibilität im Vorgehen, die die Arbeitsaufgabe erfordert. Zusätzlich kann zum Einstieg auch ein *fester Weg* durch die Arbeitsaufgabe gewählt werden. Nach dem Aufruf des Programms kann der Benutzer sich zwischen diesen Alternativen entscheiden. Abbildungen 7.1 und 7.2 auf Seite 132 veranschaulichen dies.

Der „ausgeschilderte“ Weg im Leitsystem ist identisch mit dem Vorgehen im festen Weg. Auf diesem Weg legt der Benutzer zunächst allgemeine Projektdaten fest (z.B. die Anzahl der Raumkonzepte) und anschließend werden die Raumkonzepte nacheinander definiert. Für jedes Raumkonzept startet der Benutzer mit der Angabe der einzusetzenden Gerätetypen und legt dann Beleuchtungssteuerung, Sonnenschutzsteuerung, Einzelraumregelung und Bedienung fest. Die benötigte Netzwerktechnik wird vom System automatisch berechnet. Sind an einem Ort gerade mehrere Entscheidungen möglich, so kann der Benutzer auch im festen Weg die Reihenfolge für diese Entscheidungen frei wählen. Abbildungen 7.3 und 7.4 auf Seite 133 zeigen beispielhaft je einen Screen aus dem Leitsystem und dem festen Weg.

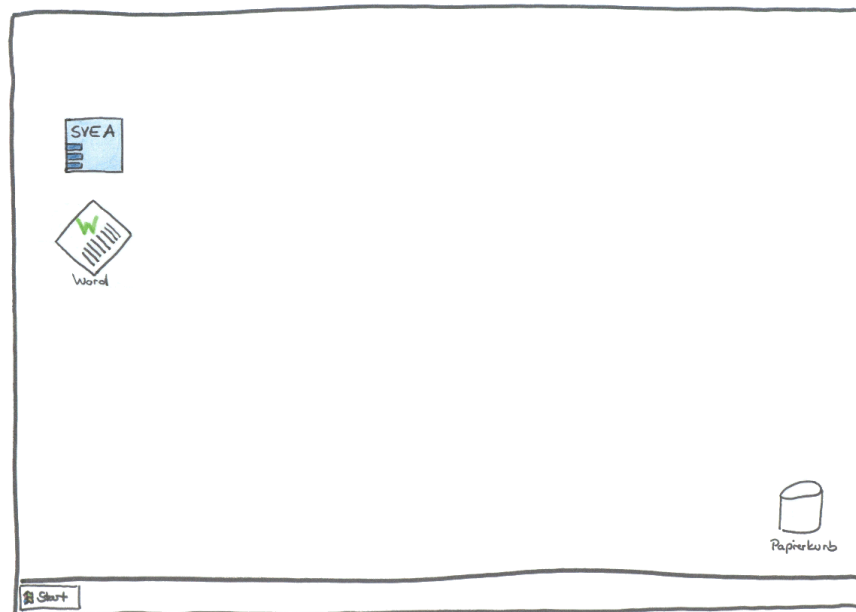


Abbildung 7.1: Der Desktop als Ausgangspunkt

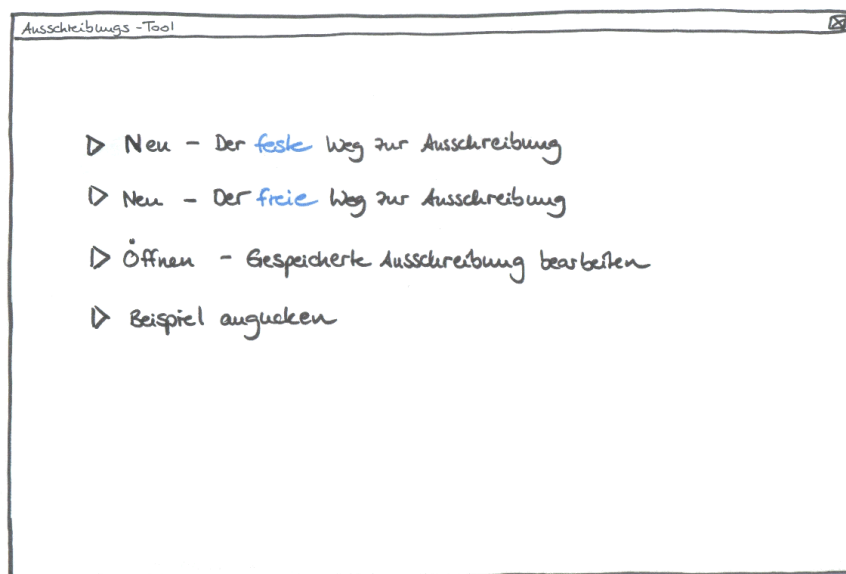


Abbildung 7.2: Fester Weg und Leitsystem (hier als freier Weg bezeichnet) als Alternativen

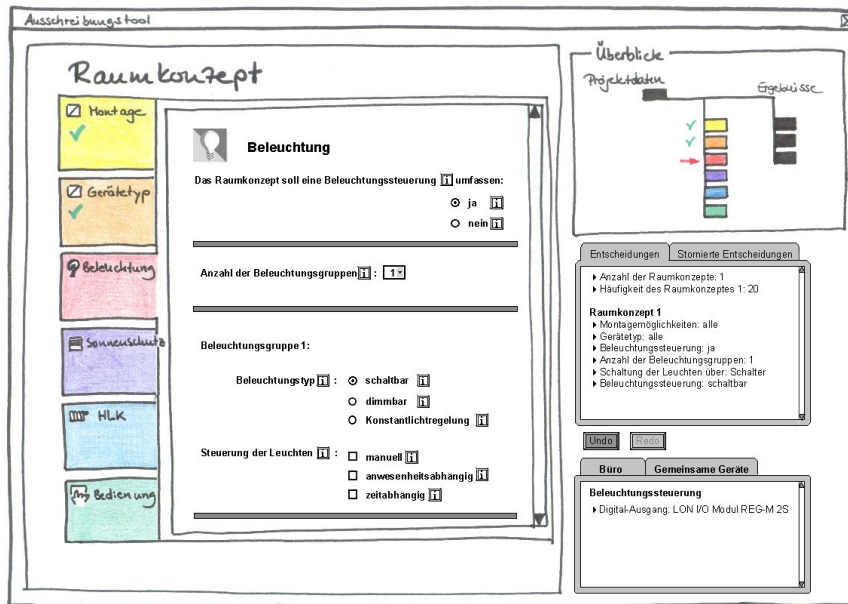


Abbildung 7.3: Beispielscreen im Leitsystem

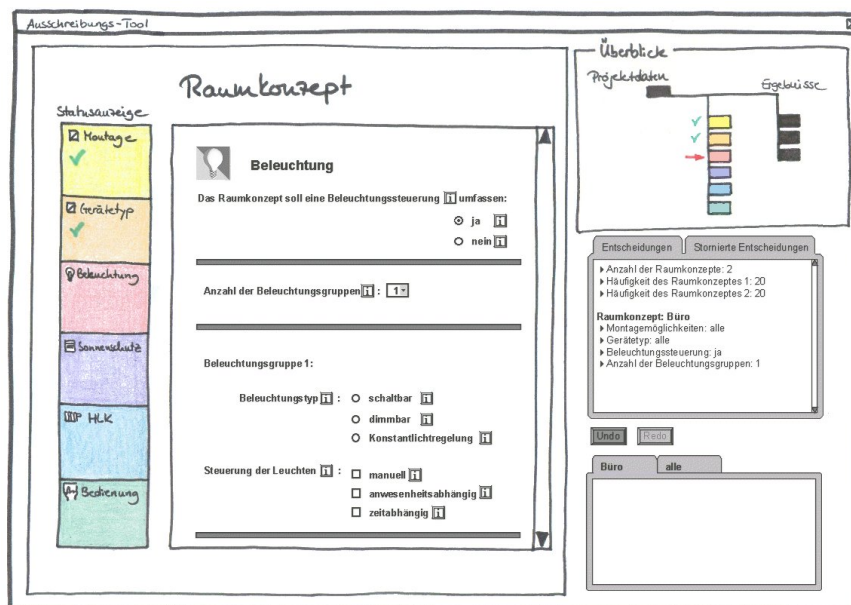


Abbildung 7.4: Beispielscreen im festen Weg

### Produktsicht und Anwendungssicht

Um sowohl Vertriebsmitarbeitern als auch Planern die Arbeit mit dem System zu ermöglichen, erfolgt die Anforderungsdefinition bei der Produktauswahl aus *Anwendungssicht*. So können Planer auch unabhängig von Vertriebsmitarbeitern selbständig Ausschreibungen erstellen. Die Produktauswahl aus Anwendungssicht unterstützt auch den Vertriebsmitarbeiter, da er im Gespräch mit dem Planer nicht gezwungen ist, zwischen Produktsicht und der Anwendungssicht des Planers zu übersetzen. Der Benutzer wählt also nicht den gewünschten Dimmer-Ausgang oder Steuerausgang direkt aus. Stattdessen entscheidet er, ob die Beleuchtung gedimmt werden soll und welche Art von Leuchten eingesetzt wird.

Der Prototyp bietet aber nicht nur die Anwendungssicht, sondern verbindet Anwendungssicht und Produktsicht. Während der Anforderungsdefinition und der funktionalen Ausschreibung eine Anwendungssicht zugrunde liegt, bieten die Kostenschätzung, das Massen-LV und die Ergebnisanzeige eine Produktsicht. Auf die Ergebnisanzeige wird weiter unten eingegangen. Diese *Kombination von Produkt- und Anwendungssicht* ist sehr sinnvoll, da sie das Erkennen von Zusammenhängen zwischen diesen Sichtweisen ermöglicht und so das aufgabenbezogene Lernen unterstützt. Außerdem können durch diese Kombination von anderen Personen erstellte Lösungen nachvollzogen und geprüft werden. So kann zum Beispiel ein Vertriebsmitarbeiter anhand der funktionalen Ausschreibung nachvollziehen, warum einzelne Geräte im Massen-LV enthalten sind. Dies stellt eine erhebliche Verbesserung dar, weil viele der Rückfragen an den Planer unnötig werden.


### Konzeptuelles Modell

Da Planer oft nur geringes Domänenwissen im Bereich LON-basierter GA-Lösungen haben, ist es besonders wichtig, eine Vorstellung von der Arbeitsaufgabe zu vermitteln. Daher wird im Prototyp die Struktur der Arbeitsaufgabe mit Hilfe der *Raum-Metapher* vermittelt. Als Grundlage hierfür habe ich in Zusammenarbeit mit den Vertriebsmitarbeitern, eine *benutzerorientierte Struktur* erarbeitet. Problematisch war in dieser Beziehung, dass eine Zusammenarbeit mit dem Planer erst spät möglich war. Dies stellte in gleicher Weise ein Problem für das Aufgreifen der Fachsprache der Benutzer dar. So konnte ich auch das Glossar der Fachbegriffe ausschließlich in Zusammenarbeit mit den Vertriebsmitarbeitern erstellen.

Soll der Benutzer durch die Raum-Metapher unterstützt werden, so benötigt er Wissen über Orte, Routen und Überblickswissen.

**Orte.** Für jedes Raumkonzept, das der Benutzer erstellt (z.B. Büro, Labor, etc.), umfasst der Prototyp die folgenden Orte: Gerätetypen, Beleuchtungssteue-

rung, Sonnenschutzsteuerung, Einzelraumregelung und Bedienung. Eine Übersicht über die *Orte* des Prototyps (Navigationseinheiten) befindet sich im Anhang auf Seite 175. Abbildung 7.5 hingegen zeigt beispielhaft Entscheidungen der Navigationseinheit Beleuchtungssteuerung. Als *Erkennungszeichen* für die einzelnen Orte habe ich Symbole verwendet, die größtenteils bereits von SVEA eingesetzt werden. Außerdem sind die Orte durch unterschiedliche Farben gekennzeichnet und jeder Ort ist mit einem Namen bezeichnet. Abbildung 7.6 auf Seite 136 gibt einen Überblick über die eingesetzten Orientierungspunkte. Durch sie kann der Benutzer Orte wiedererkennen und so ein mentales Modell der Umgebung aufbauen.

 **Beleuchtung**

Das Raumkonzept soll eine Beleuchtungssteuerung  umfassen:

ja

nein

---

Anzahl der Beleuchtungsgruppen  :

---

Beleuchtungsgruppe 1:

Beleuchtungstyp  :  schaltbar

dimmbar

Konstantlichtregelung

Steuerung der Leuchten  :  manuell

anwesenheitsabhängig

zeitabhängig

---

Leuchtenart  :  DALI-Geräte

Dimmbare EVGs

Elektronische Transformatoren

Glühlampen

HV-Halogenlampen

Dimmbare gewickelte Transformatoren

---

Bedienung der Beleuchtungsgruppe  :  Vorort

Zentral

---

Anwesenheit feststellen über  :  Präsenztaster

Präsenzmelder

---

Raummaße  :  Meter X  Meter

---

Abbildung 7.5: Entscheidungen der Navigationseinheit Beleuchtungssteuerung



Abbildung 7.6: Erkennungszeichen der Orte

Da eine Beleuchtungssteuerung beispielsweise bis zu 10 Beleuchtungsgruppen umfassen kann, ist es nicht sinnvoll, alle an einem Ort möglichen Entscheidungen *sichtbar* zu machen. Dies würde den Ort unüberschaubar groß machen. Trotzdem sollten alle möglichen Entscheidungen erforschbar sein. Daher werden alle Entscheidungen angezeigt, die für eine einzelne Beleuchtungsgruppe getroffen werden können. Bei einer höheren Anzahl von Beleuchtungsgruppen muss der Ort dann um die entsprechenden Entscheidungen erweitert werden. Die aktuell nicht notwendigen Entscheidungen werden durch grauen Text kenntlich gemacht.

Wichtig bei der Präsentation der Entscheidungen ist auch die Wahl zwischen Checkboxen bzw. Radiobuttons. Üblicherweise werden ja/nein-Entscheidungen durch Checkboxen repräsentiert. Dies ist jedoch bei der formularbasierten Produktauswahl nicht sinnvoll, weil jeweils drei Zustände unterschieden werden müssen. Am Beispiel der ersten Entscheidung in Abbildung 7.5 wären dies die folgenden Zustände: „Ja, das Raumkonzept soll eine Beleuchtungssteuerung umfassen.“, „Nein, das Raumkonzept soll keine Beleuchtungssteuerung umfassen.“ und „Die Entscheidung wurde noch nicht getroffen.“. Der dritte Zustand ist für die Anfangsbelegung und besonders nach dem Stornieren von Entscheidungen wichtig. Kann dieser Zustand nicht dargestellt werden, so würde aus dem Stornieren der Entscheidung „Ja ...“ automatisch die Entscheidung „Nein ...“ folgen, was nicht der Intention des Benutzers entspricht. Durch die Verwendung von Radiobuttons oder Comboboxen ist es möglich, diese drei Zustände zu unterscheiden, daher sollten sie zur Präsentation von ja/nein-Entscheidungen eingesetzt werden. Dies wird in Abbildung 7.5 jedoch noch nicht vollständig befolgt.

Ein weiteres Problem ist der große Platzbedarf, der schon bei den in Abbildung 7.5 gezeigten Entscheidungen entsteht. Dieser Platzbedarf kann durch die



Verwendung von Comboboxen erheblich reduziert werden. Obwohl hierdurch der Überblick über Entscheidungsalternativen erschwert wird, ist diese Variante sehr sinnvoll, da sie die Übersicht über den Ort ermöglicht. Im Prototyp habe ich keine Comboboxen eingesetzt, weil die Interaktion mit Comboboxen auf der Basis eines Papier-Prototyps nur umständlich simuliert werden kann.

**Routen.** Je nach Wahl von Leitsystem oder festem Weg bestehen unterschiedliche *Routen* zwischen den Orten im System. Im festen Weg ist eine Navigation zwischen den Orten nur entsprechend der festgelegten Vorgehensweise möglich. Die jeweils nächste Navigationseinheit ist erst erreichbar, wenn alle Entscheidungen der aktuellen Navigationseinheit getroffen wurden. Bereits besuchte Orte können über den zurückgelegten Weg aufgesucht werden. Im Leitsystem kann der Benutzer zusätzlich beliebige Orte direkt erreichen. Der Wechsel zwischen Orten ist hier auch über den Überblick und über die Reiter möglich.

**Überblick.** Abbildung 7.7 zeigt den Überblick, der im festen Weg und im Leitsystem eingesetzt wird. Der Standort des Benutzers ist durch einen Pfeil ge-

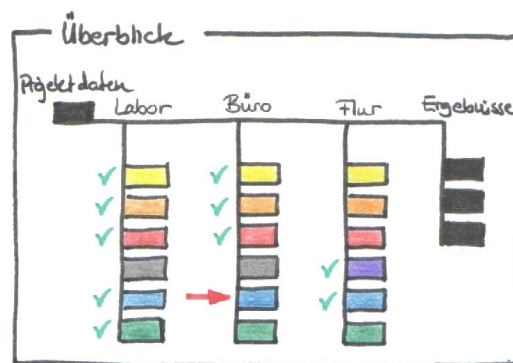


Abbildung 7.7: Überblick über die Orte

kennzeichnet. Die Beziehung zwischen dem Überblick und der Umgebung wird durch Aufgreifen der Farben und durch die Namen der einzelnen Raumkonzepte verdeutlicht. Zusätzlich habe ich für die Reiter im Leitsystem eine seitliche Anordnung gewählt. Auch dies verdeutlicht die Beziehung zwischen Überblick und Umgebung, da die Anordnung der Orte im Überblick der Anordnung in der Umgebung entspricht. Im Überblick hätten auch die Namen der Orte und die in Abbildung 7.6 abgebildeten Erkennungszeichen wieder aufgegriffen werden können, wodurch das Finden von Zielorten erleichtert worden wäre. Aus Platzgründen habe ich auf diese Unterstützung verzichtet. Die Namen der einzelnen Orte können stattdessen über Tooltips zugänglich gemacht werden, wie in Abbildung 6.13 auf Seite 106 angedeutet. Die Namen der Raumkonzepte sind in der Übersicht enthalten, da sie sich bei jeder Produktzusammenstellung ändern. Daher sind sie

für die Orientierung von besonderer Bedeutung und auch für Benutzer, die die Struktur der Aufgabe bereits kennen, eine wichtige Unterstützung. Der Überblick gibt durch die räumliche Gruppierung der Orte die Struktur der Aufgabe gut wieder. Erledigte Subaufgaben (Navigationseinheiten) werden in diesem Überblick durch Haken gekennzeichnet. Subaufgaben, die in der aktuellen Aufgabe nicht enthalten sind (z. B. Sonnenschutzsteuerung), werden grau dargestellt, wie Abbildung 7.7 veranschaulicht. So kann der Benutzer seinen *Fortschritt bei der Arbeit* verfolgen. Die Verwendung eines identischen Überblicks in Leitsystem und festem Weg ist nicht sinnvoll, da nicht verdeutlicht wird, welche unterschiedlichen Routen möglich sind. Eine überarbeitete Version des Prototyps sollte daher im festen Weg einen Überblick einsetzen, der die festgelegte Route klarer vermittelt.

Bei einer GA-Lösung mit 10 Raumkonzepten entsteht ein nicht geringer Platzbedarf für den Überblick. Dieser kann durch eine Art Fisheye-View auf den Überblick reduziert werden. Dadurch, dass immer nur die Orte des aktuellen Raumkonzeptes in voller Größe dargestellt werden, wird der Platzbedarf verringert. Eine solche Lösung erscheint mir sinnvoll, auch wenn sie Niensens Forderung nach einem statischen Überblick nicht ganz gerecht wird. Da nur die Größenverhältnisse der Darstellung verändert werden, halte ich diesen Ansatz für vertretbar. Die vergrößerte Darstellung des aktuellen Raumkonzeptes hat zusätzlich den Vorteil, dass der aktuelle Standort des Benutzers hervorgehoben wird.

Auch die Arbeitsergebnisse sind entsprechend der Struktur der Arbeitsaufgabe gegliedert. Ein Beispiel hierfür gibt Abbildung 7.8. Durch diese sinnvolle

Kostenschätzung Projekt: Krankenhaus XY

Gesamtsumme \_\_\_\_\_

Raumkonzept 1: Büro

\_\_\_\_\_

\_\_\_\_\_

Netzwerktechnik Büro

\_\_\_\_\_

\_\_\_\_\_

Summe \_\_\_\_\_

Raumkonzept 2: Labor

\_\_\_\_\_

\_\_\_\_\_

Netzwerktechnik Labor

\_\_\_\_\_

\_\_\_\_\_

Summe \_\_\_\_\_

Gesamtsumme \_\_\_\_\_

Abbildung 7.8: Kostenschätzung für eine GA-Lösung

und einheitliche Strukturierung der Ausschreibungen werden sie wesentlich verständlicher und leichter lesbar. So können Benutzer die von anderen Personen erstellten Ausschreibungen leichter nachvollziehen als es momentan der Fall ist.

### **Historie, Undo und Ändern von Entscheidungen**

Unter dem Überblick befindet sich eine *Interaktionshistorie* und eine Redo-Liste (siehe Abbildungen 7.3 und 7.4 auf Seite 133). Sie enthalten jeweils alle Entscheidungen, die der Benutzer bereits getroffen bzw. storniert hat. Um eine flexible Produktauswahl zu unterstützen, wird die Historie ständig angezeigt. Darüber hinaus ermöglicht sie jederzeit eine Übersicht über die bereits getroffenen Entscheidungen. Wichtig ist, dass der Benutzer die Entscheidungen in der Historie den einzelnen Raumkonzepten zuordnen kann. Im festen Weg ist dies relativ einfach, da alle Entscheidungen zu einem Raumkonzept nacheinander getroffen werden müssen. Im Leitsystem ist es jedoch notwendig, die Entscheidungen zu kennzeichnen, da sie in beliebiger Reihenfolge getroffen werden können. So kann in der Historie für jede Entscheidung der Name des zugehörigen Raumkonzeptes aufgeführt werden. Zusätzlich kann ein gruppierender Filter für die Entscheidungen angeboten werden. Diese Funktionalität ist im Prototyp jedoch nicht vorgesehen. Vollständig getrennte Historien für die einzelnen Raumkonzepte sind dagegen weniger sinnvoll, da sie die zeitliche Folge der Entscheidungen nicht verdeutlichen und ein Raumkonzepte übergreifendes Arbeiten erschweren.

Die Interaktionshistorie ist auch die Basis für das *Stornieren von Entscheidungen*. Abhängig von der Wahl des Benutzers zwischen festem Weg und Leitsystem sind unterschiedliche Undo-Mechanismen vorgesehen. Im festen Weg ist eine Reihenfolge für die Entscheidungen vorgeschrieben. Daher ist auch ein Widerruf beliebiger Entscheidungen nicht möglich. Mit Hilfe des `group multiple undo` kann der Benutzer jedoch in einem Schritt auch mehrere Entscheidungen stornieren. Das Leitsystem bietet hingegen mit dem referentiellen Undo einen Undo-Mechanismus, der den Anforderungen der Arbeitsaufgabe voll gerecht wird.

Ein ähnlicher Unterschied besteht zwischen festem Weg und Leitsystem beim *Ändern von Entscheidungen*. Während im festen Weg nur die Entscheidungen am aktuellen Ort geändert werden können, ermöglicht das Leitsystem das direkte Ändern beliebiger Entscheidungen.

Außerdem umfasst der Prototyp eine Anzeige der bisherigen Teillösung (siehe Abbildungen 7.3 und 7.4). Diese Ergebnisanzeige ermöglicht es dem Benutzer, die Auswirkungen seiner Entscheidungen zu verfolgen. So kann er eine Beziehung herstellen zwischen den aus Anwendungssicht getroffenen Entscheidungen und der Zusammenstellung der Lösung aus Produktsicht.

Zusätzlich vermittelt die kontinuierliche Erweiterung des komplexen Produktes den *Fortschritt bei der Arbeit*. Schließlich kann über diese Ergebnisanzeige

auch der Austausch von Geräten ermöglicht werden. Diese Funktion ist im Prototyp jedoch noch nicht vorgesehen.

Bei der Ergebnisanzeige kann zwischen einer textuellen und einer grafischen Darstellung der aktuellen Lösung unterschieden werden. Während eine textuelle Anzeige einen geringeren Platzbedarf hat, einfacher zu realisieren ist und die Produktbezeichnungen aufgreift, die zumindest die Vertriebsmitarbeiter kennen, hat die grafische Anzeige den Vorteil, dass auch die Beziehungen zwischen den Geräten dargestellt werden können. Um eine ständige Anzeige der aktuellen Teillösung zu ermöglichen, ist die Beschränkung auf eine textuelle Anzeige notwendig. Um einen Überblick über die Lösung zu erleichtern und ein Verständnis der Lösung zu unterstützen, sind die Geräte entsprechend den Raumkonzepten sortiert.

### **Exploration**

Der Prototyp umfasst mehrere Unterstützungsansätze, die eine explorative Produktauswahl fördern: Auf die Sichtbarkeit der Entscheidungen an einem Ort, die Interaktionshistorie, die Undo-Funktion und das direkte Ändern von Entscheidungen wurde bereits oben eingegangen. Auch der Handlungsspielraum im Leitsystem unterstützt ein exploratives Arbeiten.

Obwohl *Wiederaufsetzpunkte* eine sehr wirkungsvolle Unterstützung bieten, umfasst der Prototyp diesen Ansatz bisher nicht. Gegen den Einsatz sprach die Notwendigkeit eines Interaktionsgraphen zum Setzen und Aufrufen von Wiederaufsetzpunkten. Ein solcher Graph kann sehr umfangreich und von komplexer Struktur sein. Daher halte ich ihn zur Unterstützung von gelegentlichen Benutzern nicht für besonders geeignet. Die von mir erarbeitete Alternative, die Wiederaufsetzpunkte auf der Basis des Überblicks ermöglicht, ist erst später entstanden und konnte daher nicht im Prototyp eingesetzt werden. Diese Lösung halte ich jedoch für eine sehr sinnvolle Erweiterung.

Auch *Beispiellösungen* und eine *Szenario-Maschine* oder *animation machine* würden den Prototyp sinnvoll erweitern. Beispiellösungen sind in der Startauswahl des Prototyps genannt (siehe Abbildung 7.2). Einzelne Beispiele wurden aus Zeitgründen nicht im Detail ausgearbeitet. Da Beispiellösungen die von Planern gewohnte Vorgehensweise bei der Erstellung von Ausschreibungen aufgreifen, sollten sie in jedem Fall angeboten werden. Auf den Entwurf einer Szenario-Maschine bzw. einer *animation machine* musste ich ebenfalls aus Zeitgründen verzichten.

Zusätzlich sieht der Prototyp zu jeder Entscheidung und jeder Alternative ein Info-Fenster vor, das die Auswirkungen der Entscheidung beschreibt und mögliche Beziehungen zu anderen Entscheidungen verdeutlicht. Info-Fenster (siehe Skizze in Abbildung 7.9) kann der Benutzer parallel zum Hauptfenster öffnen, so dass sie beliebig lange erreichbar bleiben.

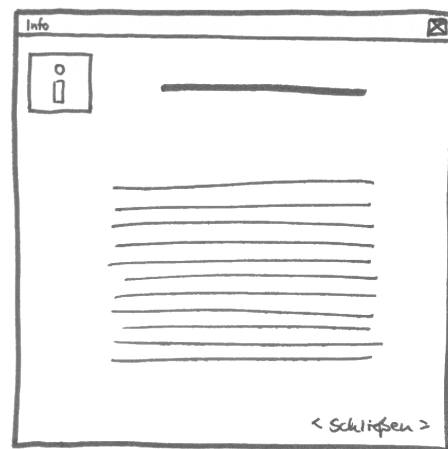


Abbildung 7.9: Info-Fenster bieten Informationen zu den Auswirkungen einer Entscheidung und Beziehungen zu anderen Entscheidungen.

Da ich mich im Prototyp auf den Kernarbeitsablauf konzentriert habe, sind keine benutzerdefinierten Navigations- und Konfigurierungseinheiten vorgesehen.

Die wesentlichen Komponenten der Benutzungsschnittstelle sind oben besprochen worden: Der Bereich zur Festlegung von Entscheidungen, der Überblick, die Historie und die Ergebnisanzeige. Abbildung D.4 im Anhang (Seite 176) fasst die Funktionen dieser Komponenten für das Leitsystem zusammen. Die Anordnung dieser Komponenten im Fenster entspricht nicht der in vielen anderen Programmen gängigen Aufteilung. So wird zum Beispiel beim Windows Explorer und in vielen Web-Sites die Hierarchie links angeordnet, während der Überblick im Prototyp rechts liegt. Die im Prototyp gewählte Anordnung orientiert sich an der Aufmerksamkeitsverteilung auf dem Monitor. Ob eine solche Orientierung an der allgemeinen Aufmerksamkeitsverteilung sinnvoll ist, oder ob es sinnvoller ist, eventuelle Gewohnheiten der Benutzer aufzugreifen, kann nur ein vergleichender Benutzertest ergeben.

## 7.3 Evaluation

Im Rahmen der Arbeit war nur der Benutzertest mit einem Planer möglich. Das von mir eingesetzte Vorgehen wird in Abschnitt 7.5 geschildert und begründet. Eine Bewertung eines Entwurfes ist auf der Grundlage eines Tests nicht möglich. Hierfür sollten weitere Tests durchgeführt werden. Ziel des Testes war es, Schwachstellen und Probleme im Entwurf aufzudecken. Im Folgenden werden die wichtigsten Ergebnisse geschildert und mögliche Verbesserungen erläutert.

Schon im Interview hatte der Planer geäußert, dass Ausschreibungen üblicherweise nicht neu erstellt werden, sondern bereits bestehende geändert würden. Beim Betrachten der Startauswahl äußerte er erneut, dass er hier am liebsten eine alte Ausschreibung öffnen würde. Eine überarbeitete Version sollte daher auch Beispiellösungen anbieten.

Anschließend wählte der Planer jedoch den *festen Weg zur Ausschreibung*. Der Unterschied zwischen den beiden Alternativen war ihm dabei klar und er wählte den festen Weg, mit der Begründung: „... weil ich das erste Projekt mache und geh’ davon aus, dass das Programm den Weg definiert und darauf achtet, dass ich nichts vergesse.“ Nachdem er am ersten Ort alle Entscheidungen getroffen hatte, erwartete der Planer zunächst, dass er automatisch am nächsten Ort landen würde. Erst nachdem dies nicht geschah, entdeckte er den Weiterbutton. Um diesem Missverständnis vorzubeugen, sollte der Weiterbutton im festen Weg zunächst nur „ausgegraut“ dargestellt werden. Hierdurch kann der Button, nachdem alle Entscheidungen getroffen wurden, hervorgehoben werden. Dies hat erstens den Vorteil, dass die Aufmerksamkeit des Benutzers auf den Weg zum nächsten Ort gelenkt wird, und es verdeutlicht zusätzlich, dass dieser Ort erst erreichbar ist, nachdem alle Entscheidungen getroffen wurden. Ein vom System ausgelöster Sprung an den nächsten Ort ist dagegen nicht sinnvoll. Er kann zur Desorientierung des Benutzers führen und unterstützt nicht den Aufbau eines mentalen Modells der Umgebung. Aus der Tatsache, dass der Benutzer die letzte Entscheidung an einem Ort getroffen hat, kann nicht geschlossen werden, dass er den nächsten Ort aufsuchen möchte. Auch die Statusanzeige könnte das Bemerkens des Weiterbuttons erschwert haben, weil sie im festen Weg sehr dominant ist. Sie sollte in jedem Fall Platz sparer gestaltet werden und die festgelegte Reihenfolge der Orte deutlicher betonen. Ein Beispiel hierfür gibt Abbildung 6.8 a) auf Seite 97.

Nachdem der Planer die erste Ausschreibung erstellt hatte, wollte er einzelne Entscheidungen direkt am entsprechenden Ort ändern. Dies ist jedoch im festen Weg nicht möglich. Auch das Aufsuchen der Orte über den festgelegten Weg empfand der Planer als zu aufwändig: „Direkt hin- und herspringen ist notwendig, um flexibler und schneller zu arbeiten“. Abschließend beurteilte er den festen Weg als geeigneter zum anfänglichen Verstehen. Nach dem ersten Durchlauf würde man aber seiner Meinung nach nur noch mit dem freien Weg arbeiten. Um das Ändern einer Lösung, die über den festen Weg erstellt wurde, zu ermöglichen, sollte dem Benutzer ein Übergang vom festen Weg in das Leitsystem angeboten werden. Darüber hinaus ist es fraglich, ob das Angebot eines festen Weges überhaupt notwendig ist, wenn dieser tatsächlich nur einmal benutzt wird. Alternativ könnte dem Benutzer der Einstieg durch eine Szenario-Maschine oder animation machine erleichtert werden. Dies hätte den Vorteil, dass die Einschränkung des Handlungsspielraums den Benutzer nicht bei der tatsächlichen Arbeit behindert. Außerdem können zum Beispiel in einem Szenario noch weitergehende Hilfen

beim Einstieg gegeben werden, als es im festen Weg möglich ist. Ein Nachteil dieser Lösung könnte darin bestehen, dass der Benutzer nicht sofort an der Lösung seiner eigenen Aufgabe arbeiten kann. Dies könnte eventuell die Motivation der Benutzer, sich mit dem System auseinanderzusetzen, vermindern.

Die verwendete Struktur der Arbeitsaufgabe und die eingesetzten Fachbegriffe wurden vom Planer weitestgehend verstanden und verursachten keine Missverständnisse. Eine Ausnahme stellten hier die beiden Navigationseinheiten *Gerätetyp* und *Montagemöglichkeiten* dar. Welche Entscheidungen unter dem Begriff *Gerätetyp* möglich sein würden, konnte sich der Planer nicht sofort vorstellen. Für den Begriff *Montagemöglichkeiten* wird in der Statusanzeige die Abkürzung *Montage* verwendet. Dies war für den Planer verwirrend, da er den Begriff mit dem Einbau der Geräte in das Gebäude verbunden hat. Da diese beiden Navigationseinheiten zusätzlich auch sehr klein und relativ eng miteinander verbunden sind, erscheint es sinnvoll, sie zu einer Einheit zusammenzulegen. Ob die Bezeichnung *Gerätetypen* für diesen Ort verständlich ist oder welche anderen Begriffe eventuell intuitiver sind, können nur weitere Benutzertests ergeben.

Bei der Festlegung der Beleuchtungssteuerung vermisste der Planer eine Möglichkeit, die Anforderung „Leuchtmittelüberwachung“ angeben zu können. Außerdem wollte er für das Massen-LV die gewünschten Formatierungen angeben. Schließlich sollte die Ausschreibung nach seinen Wünschen alle notwendigen Geräte und etwaiges Zubehör umfassen und nicht nur die von SVEA angebotenen Geräte. Dies sei wichtig, um das Vergessen von Komponenten zu verhindern.

Der Prototyp ist beim Planer auf große Akzeptanz gestoßen. Er hat geäußert, dass ein solches System eine hilfreiche Unterstützung beim Erstellen von Ausschreibungen wäre. Die vom Prototyp vorgesehenen Arbeitsergebnisse hat er im Benutzertest als „Rundum-Sorglos-Paket“ bezeichnet. Abschließend hat der Planer ein solches System zum Erstellen von Ausschreibungen für LON-basierte GA-Systeme als erstrebenswert bezeichnet. Daher halte ich die Weiterentwicklung und zusätzliche Tests für sinnvoll.

## 7.4 Wissensbasis

Neben dem Prototyp habe ich im Rahmen der Arbeit auch eine prototypische Wissensbasis für den Anwendungsfall SVEA erstellt. Hierfür habe ich mit K-Build gearbeitet, einem Tool von encoway, das eine grafische Benutzungsschnittstelle zur Modellierung bietet. Die Wissensbasis umfasst Partonomie und Taxonomie zur Beschreibung der Domänenobjekte. Sie enthält über 100 Konzepte und einige Constraints. Um sicherzustellen, dass nur fehlerfreie GA-Systeme konfiguriert werden können, müsste die Menge der Constraints noch erheblich erweitert werden. Kontrollwissen umfasst die Wissensbasis bisher nicht. Abbildungen 7.11 und 7.10 zeigen Auszüge aus der Wissensbasis.

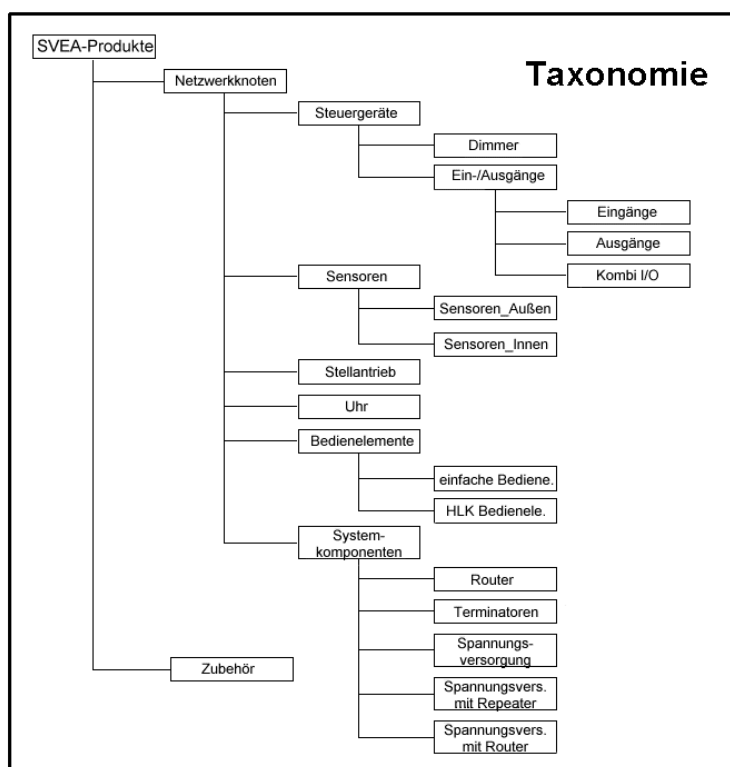


Abbildung 7.10: Auszug aus der Wissensbasis

Da der Prototyp eine Produktauswahl aus Anwendungssicht ermöglichen soll, habe ich auch in der Wissensbasis sowohl ein Produktmodell als auch ein Anwendungsmodell erstellt. In Abbildung 7.11 sind die Konzepte grau gekennzeichnet, die dem Anwendungsmodell zuzuordnen sind.



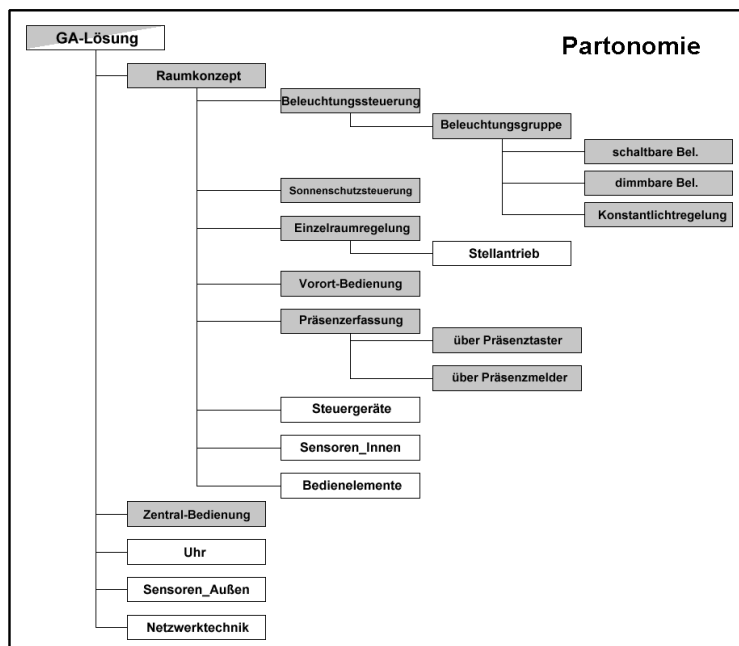


Abbildung 7.11: Auszug aus der Wissensbasis. Die grau gekennzeichneten Konzepte beschreiben das Anwendungsmodell.

## 7.5 Vorgehen

Im Folgenden gehe ich auf meine Vorgehensweise beim Prototypentwurf ein, und greife dabei zurück auf die in Abschnitt 2.3 beschriebenen Vorgehensmodelle beim Usability Engineering und den Abschnitt 3.5 zum Thema Wissenserhebung.

Ich habe mich zunächst dazu entschlossen, mich an dem von Mayhew beschriebenen Usability Engineering Lifecycle zu orientieren. Neben der detaillierten Erläuterung des Vorgehens und der Abschätzung des Aufwandes für Entwickler und Benutzer sprachen vor allem die Hinweise auf mögliche Shortcuts dafür. Mayhews Lifecycle ist eigentlich auf größere Projekte ausgerichtet. Sie geht von einem Team aus mehreren Personen mit unterschiedlichen Rollen – Usability Engineer, User Interface Designer und User Interface Developer – aus. Für eine einzelne Person würde das beschriebene Vorgehen eine Projektdauer von über einem Jahr erfordern. Bei den daher notwendigen Abweichungen wollte ich mich an den von Mayhew beschriebenen Shortcuts und dem Discount Usability Engineering von Nielsen orientieren. Im Laufe der Arbeit haben sich jedoch auch weitere Abweichungen ergeben, auf die ich im Folgenden eingehen werde.

### 7.5.1 Einarbeitung

Zur Einarbeitung in das Gebiet der Konfigurierung habe ich neben Gesprächen mit Experten des Gebietes und dem Literaturstudium den Workshop „Konfiguration in der wissenschaftlichen Praxis“ vom 05. bis 06. Juli am TZI in Bremen besucht. Zusätzlich habe ich eine Reihe unterschiedlicher Konfiguratoren untersucht.

Um mich mit dem Gebiet der LON-basierten Gebäudeautomation vertraut zu machen, habe ich ein Gespräch mit einem Praktikanten bei SVEA geführt und dann mein Wissen anhand der Literatur erweitert. Außerdem habe ich mich mit dem Produktspektrum von SVEA vertraut gemacht.

### 7.5.2 Benutzerprofile

Alle Personen, deren Arbeit durch die Einführung eines Systems beeinflusst wird, sollten auch als Benutzer des Systems betrachtet werden. Aufgrund des begrenzten Rahmens einer Diplomarbeit habe ich mich jedoch auf zwei zentrale Benutzergruppen – Vertriebsmitarbeiter und Kunden – konzentriert. Diese beiden Gruppen haben sich in Gesprächen mit Experten als bedeutsam herausgestellt. Sie spielen als primäre Benutzer des Systems eine wesentliche Rolle. Die Unterscheidung dieser beiden Gruppen ist sinnvoll, da sie meist über verschiedenartiges Vorwissen verfügen und bei der Produktauswahl unterschiedliche Ziele verfolgen (siehe hierzu 4.2).

Im Rahmen der Arbeit konnte ich mit zwei der fünf Vertriebsmitarbeiter von SVEA zusammenarbeiten, dem Innendienstmitarbeiter sowie einem Außendienstmitarbeiter. Eine Zusammenarbeit mit einem Planer, also einem Kunden von SVEA, war erst für den Benutzertest möglich. Das Benutzerprofil der Planer konnte ich daher zunächst nur auf Informationen der Vertriebsmitarbeiter aufbauen.

Mayhew sieht zur Datenerhebung für die Benutzerprofile Fragebögen vor. Der Einsatz eines Fragebogens ist jedoch bei einer Gruppe von nur zwei Benutzern nicht sinnvoll. Der Vorteil von Fragebögen, einfach quantitative Daten erheben zu können, kommt hier nicht zum Tragen. Für Interviews sprach hingegen deren größere Flexibilität. Als Shortcut schlägt auch Mayhew Interviews mit Personen vor, welche die Benutzergruppe gut kennen (Mayhew 1999, S. 39). Da die Vertriebsmitarbeiter regelmäßig mit Planern zusammenarbeiten, habe ich sie auch zur Benutzergruppe der Planer befragt, um für diese Gruppe überhaupt ein Benutzerprofil erstellen zu können. Dies geschah in Form eines Interviews. Aus Zeitgründen habe ich dieses Interview sowohl zur Erhebung von Benutzerdaten, als auch zur Task Analysis und zur Wissenserhebung genutzt.

### 7.5.3 Task Analysis und Wissenserhebung

Im Folgenden beschreibe ich zunächst mein Vorgehen zur Task Analysis, Wissenserhebung und Erstellung der Benutzerprofile. Anschließend begründe ich die einzelnen Aspekte.

Ausgangspunkt für Benutzerprofile, Task Analysis und Wissenserhebung waren Interviews, die ich mit den beiden Vertriebsmitarbeitern geführt habe. Der Interviewleitfaden, den ich hierbei eingesetzt habe, ist im Anhang enthalten (siehe Seite 158ff). Die Fragen des Leitfadens habe ich nicht alle und nicht wortwörtlich im Rahmen der Interviews gestellt. Der Leitfaden diente mir vielmehr als Überblick über die wichtigen Themengebiete. Im Zentrum meines Interesses stand der Arbeitsprozess der Produktzusammenstellung. Im Rahmen der Interviews habe ich die Vertriebsmitarbeiter gebeten, die Zusammenstellung einer konkreten GA-Lösung zu simulieren. Zusätzlich habe ich sie aufgefordert, dabei laut zu denken und ich habe selbst wiederholt Nachfragen gestellt. Die Beschreibung von konkreten Regeln, die bei der Produktzusammenstellung zu beachten sind, fiel den Vertriebsmitarbeitern sehr schwer. Hacker weist in diesem Zusammenhang darauf hin, dass vor allem Wissen über Vorgehensweisen oft psychisch automatisiert ist und nur unbewusst genutzt wird (Hacker 1992, S. 36). Dies ist problematisch, weil die auf dem Wissen aufbauenden Denkprozesse darüber hinaus nicht beobachtet werden können. Die in Abschnitt 5.3 beschriebenen wechselseitigen Abhängigkeiten zwischen Teilaufgaben habe ich zum Großteil erst in weiteren Gesprächen sowie durch Nachfragen und das Aufstellen von Hypothesen ermitteln können.

Das Interview mit dem Innendienstmitarbeiter konnte ich bei SVEA durchführen, während das Interview mit dem Außendienstmitarbeiter in einem Restaurant stattgefunden hat. Während dieser zwei- bis dreistündigen Gespräche habe ich umfangreiche Feldnotizen angefertigt und Beispiele für wichtige Arbeitsergebnisse erhalten. Anschließend habe ich auf ihrer Grundlage sofort ausführliche Gedächtnisprotokolle angefertigt. Auf der Basis der Interviews habe ich eine Reihe von Szenarien verfasst, die die Produktauswahl näher beschreiben, und diese mit den Vertriebsmitarbeitern durchgesprochen. Diese Szenarien sind im Anhang enthalten (siehe Seite 161ff). Außerdem habe ich eine Liste mit wichtigen Fachbegriffen erstellt und kontinuierlich weiterentwickelt (siehe Anhang 172ff). Neben diesen Ergebnissen habe ich die Struktur der Arbeitsaufgabe *Erstellung einer LON-basierten GA-Lösung* in Grafiken zusammengefasst und diese mit den Vertriebsmitarbeitern diskutiert und weiterentwickelt. Diesbezügliche Resultate befinden sich in Kapitel 5 und im Anhang auf Seite 174f. Die Ergebnisse der Task Analysis sowie die Benutzerprofile werden in Kapitel 5 geschildert.

Das hier beschriebene Vorgehen weicht von Mayhews Vorgehen im Usability Engineering Lifecycle ab. Es sieht als Grundlage der Task Analysis eine Kombination von Beobachtung und Interview von Benutzern bei ihrer Arbeit vor. Nach

Mayhew sollten etwa drei bis sechs Benutzer jeder Benutzergruppe für mehrere Stunden begleitet werden. Zur Datenerhebung sieht sie dabei vor allem freie Notizen sowie Kopien der wichtigen Artefakte und Arbeitsergebnisse vor aber auch Fotos sowie Audio- und Videoaufzeichnungen. Anschließend werden für jede Benutzergruppe häufige Aufgaben in Form von Szenarien beschrieben. Diese werden bei Mayhew im Team mit einigen Benutzern erstellt (Mayhew 1999, S. 95). Außerdem wird ein *Current User Task Organization Model* erarbeitet, das die Beziehungen zwischen den unterschiedlichen Benutzeraufgaben beschreibt.

Von dieser Vorgehensweise bin ich aus folgenden Gründen abgewichen: Es war nicht möglich die Vertriebsmitarbeiter während der Arbeit zu beobachten oder zu interviewen. Eine Zusammenarbeit mit mehr als zwei Vertriebsmitarbeitern war aus organisatorischen Gründen nicht durchführbar. Auch konnte ich zu diesem Zeitpunkt, wie oben beschrieben, noch nicht mit Planern zusammenarbeiten. Daher habe ich mich für ein Interview mit den Vertriebsmitarbeitern entschieden. Dies schlägt auch Mayhew als möglichen Shortcut zur Task Analysis vor (Mayhew 1999, S. 103f). Die Erstellung einer LON-basierten GA-Lösung besteht aber vor allem aus geistiger Arbeit. Beyer und Holtzblatt (1998, S.75f) betonen, dass der Interviewer bei mentalen Prozessen direkt anwesend sein muss, da der Benutzer zu einem späteren Zeitpunkt nur wenig erinnert. Daher habe ich die Vertriebsmitarbeiter im Rahmen des Interviews gebeten, die Zusammenstellung einer konkreten GA-Lösung zu simulieren. Auch Rosson und Carroll (2002, S. 54) beschreiben, dass es notwendig sein kann, Benutzer Aktivitäten simulieren zu lassen, zum Beispiel wenn diese nur selten vorkommen. Da sich mentale Prozesse der Beobachtung entziehen, habe ich die Vertriebsmitarbeiter zusätzlich gebeten, bei der Simulation der Arbeit laut zu denken. Da diese Methode im Rahmen von Benutzertests eingesetzt wird, um Aufschluss über die Denkprozesse des Benutzers zu bekommen, erscheint mir der Einsatz auch hier sinnvoll. Um den Überblick über die wichtigen Themengebiete im Interview nicht zu verlieren, hielt ich einen Leitfaden für unverzichtbar. Meuser und Nagel betonen, dass sich für Experteninterviews vor allem eine offene, leitfadenorientierte Vorgehensweise eigne:

„Wenn es aber um handlungsleitende Regeln jenseits von Verordnungen, um ungeschriebene Gesetze des ExpertInnenhandels, um tacit knowing und Relevanzaspekte geht, gibt es zu offenen ExpertInneninterviews keine Alternative.“ (Meuser und Nagel 1991, S. 449)

Dabei garantiere gerade der Leitfaden laut Meuser u.a. die notwendige Offenheit, wenn er nicht als Ablaufmodell gesehen werde. Auf eine Audio- oder Videoaufzeichnung der Interviews habe ich einerseits wegen des erheblichen Zeitaufwandes für deren Auswertung verzichtet, andererseits sprach für mich auch das Ziel einer möglichst natürlichen Gesprächssituation gegen solche Aufnahmen. Der

Gefahr des Verlustes wichtiger Informationen habe ich versucht durch die sofortige Anfertigung von Protokollen auf der Grundlage der Feldnotizen vorzubeugen.

Das im Lifecycle vorgesehene „Current User Task Organization Model“ beschreibt die Beziehung zwischen unterschiedlichen Benutzeraufgaben. Da mein Fokus jedoch auf einer konkreten Arbeitsaufgabe liegt, erschien es mir sinnvoller, die Beziehungen zwischen einzelnen Teilaufgaben dieser Aufgabe zu ermitteln. Ein *Current User Task Organization Model* habe ich daher nicht erarbeitet.

Mayhew benennt das Lernen der Benutzersprache zwar als ein Ziel der Task Analysis (Mayhew 1999, S. 79), sie sieht zu diesem Zwecke jedoch kein Glossar vor. Mir erschien dies jedoch aufgrund der fremden Domäne notwendig.

Neben den Benutzerprofilen und der Task Analysis sieht Mayhew als Grundlage für die Anforderungen auch noch die Ermittlung von *Platform Capabilities and Constraints* sowie *General Design Principles* vor. Da mein Fokus auf einer speziellen Arbeitsaufgabe liegt, habe ich diese Punkte ausgeklammert.

#### 7.5.4 Anforderungen

Usability-Anforderungen werden von Mayhew als *Usability Goals* bezeichnet. Hierbei unterscheidet sie zwischen qualitativen und quantitativen Anforderungen. Während die qualitativen Anforderungen vor allem als Richtlinien für den Entwurf dienen, werden die quantitativen Anforderungen später als Evaluationskriterien verwendet. Da eine solche summative Evaluation im Rahmen meiner Arbeit nicht möglich war, habe ich mich auf die Festlegung von qualitativen Anforderungen beschränkt. Diese wurden im Abschnitt 6.1 beschrieben.

Anforderungen werden bei Mayhew direkt aus den *User Profiles* und den Ergebnissen der *Task Analysis* sowie allgemeinen Unternehmenszielen abgeleitet (Mayhew 1999, S. 133ff). Meines Erachtens sind die aufgestellten Anforderungen jedoch auch von den zugrunde gelegten Perspektiven abhängig. Dieser Aspekt wird jedoch von Mayhew nicht thematisiert. Beyer und Holtzblatt machen diesen Aspekt im Schritt *Creating a Vision* explizit: „The range of solutions a design team considers depends on who is on the team and the *perspectives* [eigene Hervorhebung] they take [... ]“ (Beyer und Holtzblatt 1998, S. 273ff). Die meinem Entwurf zugrunde liegenden Perspektiven habe ich in Abschnitt 6.1.3 beschrieben.

#### 7.5.5 Entwurf und Evaluation

Im Folgenden erläutere ich zunächst mein Vorgehen im Rahmen des Prototypentwurfes. Anschließend begründe ich die einzelnen Aspekte und gehe auf Abweichungen vom Usability Engineering Lifecycle bei Mayhew ein.

Auf der Grundlage der für die Arbeitsaufgabe notwendigen Arbeitsabläufe, des Glossars und der Struktur der Arbeitsaufgabe habe ich einen ersten Entwurf erstellt. Abbildungen einzelner Entwürfe sind im Anhang enthalten (siehe Seite 177ff). Ziel der Evaluationen war vor allem ein Test des konzeptuellen Modells, also zum Beispiel der zugrunde liegenden Struktur der Arbeitsaufgabe, der verwendeten Fachbegriffe sowie möglicher Arbeitsabläufe. Für die erste Evaluation habe ich einen Mockup-Prototyp in Form von PowerPoint-Folien erstellt. Diese zeigen alle Screens, die zur Bearbeitung einer konkreten, typischen Arbeitsaufgabe notwendig sind. Abbildung E.2 auf Seite 178 im Anhang zeigt beispielhaft einen solchen Screen. Anhand dieser Screens habe ich mit dem Innendienstmitarbeiter die Arbeitsaufgabe am Notebook „durchgespielt“. Wir haben einzelne Aspekte erörtert und ich habe mir Notizen zu Anmerkungen und Fehlern in der Benutzungsschnittstelle gemacht.

Auf der Grundlage dieser Ergebnisse habe ich den Entwurf überarbeitet und neue Mockups erstellt. Im Gegensatz zum ersten Prototyp habe ich dabei echte Papier-Mockups angefertigt. Diesen zweiten Prototyp habe ich mit einer Gruppe von Experten – drei Informatikstudenten aus dem Gebiet der Software-Ergonomie – diskutiert. Da sie keine Kenntnisse über die Anwendungsdomäne hatten, habe ich zunächst kurz die Arbeitsaufgabe und die betrachteten Benutzergruppen vorgestellt. Auf dieser Basis haben wir wiederum anhand der Mockups eine typische Arbeitsaufgabe durchlaufen, wobei ich die einzelnen Screens jeweils aus Komponenten zusammengestellt und mit Folienstift ergänzt habe. Abschließend habe ich Problempunkte geschildert und wir haben Design-Alternativen diskutiert.

Diese Papier-Mockups wurden von mir zusätzlich für eine weitere Evaluation mit dem Außendienstmitarbeiter genutzt. Bei dieser Evaluation bin ich im Prinzip wie in der Evaluation mit dem Innendienstmitarbeiter vorgegangen. Der einzige Unterschied bestand in der Verwendung der Papier-Mockups.

Die so ermittelten Ergebnisse habe ich für eine weitere Überarbeitung des Prototyps genutzt, wobei ich wieder Papier-Mockups eingesetzt habe. Diesen letzten Prototyp habe ich in einem Benutzertest mit einem Planer getestet. Da dies der erste Termin mit einem Planer war, habe ich ihn auch für ein kurzes Interview genutzt. Dabei wollte ich vor allem Informationen darüber bekommen, wie der Planer die Arbeitsaufgabe *Erstellung einer LON-basierten GA-Lösung* sieht. Das Interview hat etwa 45 Minuten gedauert und fand in den Räumen des Planungsbüros statt. Nach dem Interview habe ich einen Benutzertest in Form eines *simplified thinking aloud test* nach Nielsen gemacht. Der Planer hat eine typische Arbeitsaufgabe mit Hilfe des Prototyps gelöst. Hierfür habe ich mit Hilfe der Mockups das System simuliert. Da dies meine volle Aufmerksamkeit erforderte, habe ich eine Informatikstudentin gebeten mich zu unterstützen. Sie hat während des Interviews und des Benutzertests Notizen angefertigt. Auf der Basis dieser Notizen, habe ich anschließend sofort ein ausführliches Gedächtnisprotokoll verfasst.

Diesen Prototyp habe ich zusätzlich einem der Geschäftsführer von SVEA vorgestellt. Auch hier haben wir den Prototyp anhand einer Arbeitsaufgabe durchlaufen und ich habe mir Anmerkungen notiert und Notizen zu Fehlern gemacht. Die Ergebnisse aller beschriebenen Termine habe ich genutzt, um das Glossar, die von mir festgelegten Navigationseinheiten, den Prototyp und die Wissensbasis zu überarbeiten. Die Navigationseinheiten sind im Anhang abgebildet (siehe Seite 175).

In Mayhews Usability Engineering Lifecycle besteht die Phase *Design/Testing/Development* aus drei Stufen:

1. *Work Reengineering* und *Conceptual Model Design and Evaluation*
2. *Screen Design Standards* festlegen und *Screen Design Standard Evaluation*
3. *Detailed User Interface Design and Evaluation*

Die Stufen umfassen einen zunehmend detaillierteren Entwurf der Benutzungsschnittstelle. Für kleinere Projekte nennt Mayhew die Möglichkeit, die drei Stufen zu einer zu verbinden und diese gemeinsam iterativ zu durchlaufen. Ich bin nach dieser Methode vorgegangen, wobei die letzte Stufe, die die vollständige Entwicklung der Benutzungsschnittstelle umfasst, bei mir entfiel. Im Rahmen des *Work Reengineering* beschreibt Mayhew vor allem die Überarbeitung des *Current User Task Organization Model* vor, also die Umstrukturierung der gesamten Aufgabemenge. Die Umgestaltung einzelner Arbeitsabläufe ist zwar auch Inhalt dieses Schrittes, hierauf geht Mayhew jedoch nur am Rande ein (Mayhew 1999, S. 179). Beyer und Holtzblatt behandeln die Überarbeitung von einzelnen Arbeitsabläufen hingegen genauer (Beyer und Holtzblatt 1998, S. 256ff). Da im Rahmen meiner Arbeit speziell das Work Reengineering der Arbeitsaufgabe Produktauswahl im Mittelpunkt steht, habe ich mich in diesem Bereich am Vorgehen von Beyer u.a. orientiert.

Mayhew validiert die überarbeiteten Arbeitsabläufe, indem sie sie mit mehreren repräsentativen Benutzern durchläuft (Mayhew 1999, S. 179). Für kleinere Projekte schlägt Mayhew als Shortcut vor, die Angemessenheit der Arbeitsabläufe im Rahmen der Prototypevaluation zu testen (Mayhew 1999, S. 182). Da es sich in meinem Projekt um nur eine Arbeitsaufgabe handelt, hielt ich diese Variante für sinnvoll. Als Grundlage für die Evaluation sieht Mayhew Papier-Mockups oder lauffähige Prototypen vor. Sie beschreibt, dass diese beiden Typen zum Ermitteln von Usability-Problemen gleichwertig sind. Da Papier-Mockups schneller entwickelt und geändert werden können, habe ich mich für diese Variante entschieden.

Zur Evaluation des Prototyps sieht Mayhew in jeder Iteration einen formalen Usability-Test vor. Dieser soll jeweils 3-10 repräsentative Benutzer umfassen, die

mehrere realistische Aufgaben bearbeiten und dabei laut denken. Ein solcher Test war wegen des hohen Zeitaufwandes für die Benutzer und für mich nicht möglich. Als Alternative wollte ich statt dessen drei Methoden kombinieren. Einen Benutzertest mit einem Planer nach der Methode des *simplified thinking aloud*, einen Austausch mit Experten und eine vereinfachte Form des *pluralistic walkthrough* (Bias 1991, 1994), mit den Vertriebsmitarbeitern und Experten aus dem Gebiet der Konfigurierung. Für den pluralistic walkthrough sprach die Möglichkeit die Sichtweisen unterschiedlicher Personengruppen auf den Entwurf in einem Treffen zu ermitteln. Außerdem kann mit dieser Methode in nur einem Termin Feedback von einer ganzen Gruppe von Personen ermittelt werden. Aus organisatorischen Gründen war ein solcher gemeinsamer Termin jedoch leider nicht möglich. Die daher notwendig gewordenen mehrfachen Termine habe ich dann als einzelne Iterationen genutzt.

Da der Prototyp ausschließlich in Zusammenarbeit mit den Vertriebsmitarbeitern entworfen worden war, wollte ich bei der Evaluation mit dem Planer nicht darauf verzichten, dass dieser die Aufgabe selbst durchführt. Ich hoffte, so mehr Ergebnisse zur Erwartungskonformität des Prototyps mit den Vorstellungen und dem Vorwissen des Planers zu erhalten. Der Austausch mit anderen Experten hingegen war mir besonders wichtig, da ich ansonsten in einem „1-Personen-Team“ gearbeitet habe.

Für die Verwendung von PowerPoint-Folien beim ersten Prototyp sprach die Tatsache, dass das Zusammenstellen der einzelnen Screens während eines Treffens sowie das Simulieren des Systemfeedbacks zeitaufwendig und kompliziert sind. Zusätzlich erfordert es viel Aufmerksamkeit, da im Prinzip die Arbeit des wissensbasierten Systems simuliert wird. Da ein wirklicher Papier-Prototyp aber das Experimentieren mit Alternativen erleichtert und auch eher dazu auffordert, habe ich trotz dieser Nachteile spätere Prototypen aus Papier realisiert.

Parallel zum Usability Engineering habe ich eine prototypische Wissensbasis für den Anwendungsfall SVEA realisiert. Dazu habe ich im Rahmen jedes Benutzertreffens Rückfragen gestellt sowie bisher Gelerntes wiedergegeben und um eventuell nötige Korrekturen gebeten.



# Kapitel 8

## Zusammenfassung und Diskussion der Ergebnisse

### 8.1 Zusammenfassung

In der vorliegenden Arbeit wurden Methoden und Mechanismen erarbeitet, beschrieben und bewertet, die Benutzer bei der Orientierung und Navigation im Prozess der Produktauswahl komplexer Produkte unterstützen.

Als Grundlage für die Unterstützungsansätze wurde zunächst erörtert, welche Voraussetzungen erfüllt sein müssen, damit Menschen sich orientiert fühlen. Die Arbeit stützt sich hier unter anderem auf die Ergebnisse des Architekten Pardini. Dieser beschreibt, dass ein erfolgreiches *Wayfinding* auch ohne fundierte Kenntnisse der Umgebung möglich ist. Als weitere Grundlage wurde auf die unterschiedlichen Navigationsmöglichkeiten – systemgeführte und benutzergeführte Interaktion – eingegangen, die in Programmen realisiert werden können.

Da es das Ziel ist, Benutzer bei einer speziellen Arbeitsaufgabe, der Auswahl eines komplexen Produktes, zu unterstützen, wurde diese Arbeitsaufgabe näher analysiert. Die Arbeitsaufgabe wird als ein meist iterativer Prozess beschrieben, in dem Anforderungen an das Produkt ermittelt, verfeinert aber auch verworfen werden. Der Prozess besteht aus einer Vielzahl voneinander abhängiger Entscheidungen. Die Arbeitsaufgabe Produktauswahl stellt daher eine komplizierte Umgebung dar, in der sich der Benutzer zurechtfinden muss. Aus der Charakterisierung der Produktauswahl als einem iterativen Prozess folgt außerdem, dass der Benutzer flexible Navigationsmöglichkeiten in der Arbeitsaufgabe benötigt.

Unabhängig davon, ob der Benutzer ein komplexes Produkt mit Hilfe eines Programms zusammenstellt oder nicht, muss er sich in dieser Arbeitsaufgabe zurechtfinden. Durch die Unterstützung der Arbeitsaufgabe mittels eines Programms, können dem Benutzer zusätzliche Umgebungsinformationen angeboten

werden. So kann er bei der Orientierung unterstützt werden. Durch die Methoden der Konfigurierung kann bei der Zusammenstellung eines komplexen Produktes ein vollständiges und konsistentes Produkt sichergestellt werden.

Die erarbeiteten Unterstützungsansätze verwirklichen die folgenden Ziele:

- Orientierung und Navigation im Prozess der Produktauswahl erleichtern, auch für Benutzer ohne ein fundiertes mentales Modell von der Arbeitsaufgabe,
- Auf- und Ausbau eines mentalen Modells der Arbeitsaufgabe fördern,
- Flexible Navigation durch die Arbeitsaufgabe auf der Basis der Konfigurierung ermöglichen.

Wegen der engen Beziehung zwischen Orientierung und Navigation ist eine klare Unterscheidung von Ansätzen, die die Orientierung fördern und solchen, die die Navigation unterstützen, nicht möglich. Es überwiegt jedoch meist einer der Aspekte. Zur Unterstützung der Orientierung wurden die nachstehenden Ansätze beschrieben:

- Vorstellung von der Arbeitsaufgabe vermitteln,
- Schemata anbieten,
- Zurückgelegten Weg erforschbar machen,
- Fortschritt verdeutlichen und
- Typische Wege vermitteln.

Die Navigation kann dagegen durch die folgenden Ansätze unterstützt werden:

- Flexibilität im Vorgehen ermöglichen,
- Abkürzungen anbieten,
- Benutzergerechte Schrittgrößen realisieren,
- Undo-Funktion offerieren,
- Direktes Ändern von Entscheidungen erlauben,
- Wiederaufsetzpunkte zur Verfügung stellen.

Die erarbeiteten Unterstützungsansätze wurden detailliert beschrieben. Vor- und Nachteile der Ansätze wurden erläutert. Sie sind als ein Baukasten von Unterstützungsmechanismen zu sehen. Entsprechend den konkreten Anforderungen in einem Anwendungsfall können geeignete Ansätze ausgewählt werden.

Die benutzer-, aufgaben- und domänenabhängige Auswahl und der Einsatz der Unterstützungsansätze wurde an einem Anwendungsbeispiel verdeutlicht. Auf der Basis von Task Analysis und Benutzerprofilen wurde eine Systemvision skizziert und Usability-Anforderungen beschrieben. Diese bildeten die Grundlage für den Einsatz der Unterstützungsansätze und den Entwurf des Prototypen. In Zusammenarbeit mit den Vertriebsmitarbeitern wurde der Prototyp wiederholt überarbeitet und in einem Benutzertest mit einem Planer evaluiert.

## 8.2 Diskussion und Bewertung der Ergebnisse

Zentrales Ziel der Arbeit war es, Benutzer bei der Orientierung und Navigation im Arbeitsprozess der Produktauswahl komplexer Produkte zu unterstützen. Dieses Ziel kann durch die in Kapitel 6 beschriebenen Unterstützungsansätze verwirklicht werden. Das Ziel der Arbeit wurde damit erreicht. Die Unterstützungsansätze stellen einen umfangreichen Baukasten von Mechanismen dar. Den jeweiligen Anforderungen in einem konkreten Projekt entsprechend, sind aus diesem Baukasten die geeigneten Unterstützungsansätze auszuwählen und zu kombinieren. Eine allgemeine Lösung ist dagegen nicht möglich, da sie abhängig ist von Merkmalen der Benutzer, der Aufgabe und der Anwendungsdomäne. Die in Kapitel 6 enthaltenen Bewertungen der Ansätze bilden eine solide Grundlage für die Auswahl und Zusammenstellung im jeweiligen Anwendungsfall. Notwendige Voraussetzung für eine geeignete Auswahl ist das Wissen über die Benutzer und ihre Aufgaben im Anwendungsfall, also ein benutzerorientiertes Vorgehen im jeweiligen Projekt. Ein weiteres Ergebnis der Arbeit ist der in Kapitel 7 beschriebene Prototyp. Er setzt eine Auswahl der Unterstützungsansätze um und veranschaulicht so ihren Einsatz.

Dem Prozess der Konfigurierung liegt ein lineares Vorgehen zugrunde. Die Analyse der Arbeitsaufgabe Produktauswahl bei komplexen Produkten hat gezeigt, dass sie meist einem iterativen Prozess entspricht. Mit Hilfe der erarbeiteten Unterstützungsansätze kann dem Benutzer auf der Basis der Konfigurierung ein flexibles Vorgehen ermöglicht werden. Eine flexible, aufgabenangemessene Navigation durch den Arbeitsprozess ist aber nur möglich, wenn der Benutzer orientiert ist und eine Vorstellung von der Arbeitsaufgabe als Ganzes hat. Auch dies kann mit Hilfe der erarbeiteten Unterstützungsansätze ermöglicht werden. Mit ihrer Hilfe kann der Benutzer sofort mit der Arbeit beginnen und während der Arbeit sein mentales Modell von der Arbeitsaufgabe auf- bzw. ausbauen.

Die erarbeiteten Unterstützungsansätze bilden ein wirkungsvolles Hilfsmittel, damit Benutzer komplexe Produkte effizient und zu ihrer eigenen Zufriedenheit auswählen können.

## 8.3 Kritischer Rückblick

Die in Kapitel 2 vorgestellten Vorgehensmodelle beschreiben ein optimales Vorgehen. In der Praxis ist es oft schwer, das beschriebene Vorgehen vollständig umzusetzen. Mögliche Gründe hierfür liegen in den begrenzten Ressourcen, der mangelnden Erreichbarkeit von Benutzern, den Vorbehalten von Projektbeteiligten oder den organisatorischen Rahmenbedingungen. So bedeutet zum Beispiel ein Interview bei der Arbeit in einem Mehrpersonenbüro eine erhebliche Störung.

Aufgrund dieser Problematik fand ich es besonders herausfordernd und interessant, die Diplomarbeit außerhalb des universitären Rahmens zu schreiben.

Der zeitliche Rahmen einer Diplomarbeit ermöglicht es nicht, das von Mayhew beschriebene Vorgehen vollständig einzusetzen. Für mich als Einzelperson hätte dies allein eine Projektdauer von 14 Monaten bedeutet. Die Einarbeitung in das Gebiet der Konfigurierung sowie den Anwendungsfall SVEA waren sehr interessant, haben aber natürlich auch zusätzliche Zeit beansprucht.

Eine noch intensivere Benutzerbeteiligung und vor allem eine frühere Beteiligung der Planer wäre wünschenswert gewesen. Die zeitlichen Möglichkeiten der Benutzer standen dem jedoch entgegen. Die frühe Beteiligung der Planer hätte es ermöglicht, parallel zu den Gesprächen mit den Vertriebsmitarbeitern auch mit den Planern die Struktur der Arbeitsaufgabe und das Fachvokabular zu erarbeiten. Obwohl dies nicht möglich war, sind beim Benutzertest mit dem Planer kaum Missverständnisse aufgrund der verwendeten Begriffe aufgetreten. Dies zeigt, dass die Vertriebsmitarbeiter die bei den Planern bekannten Fachbegriffe gut einschätzen können.

Zur Weiterentwicklung des Prototyps sollten zusätzliche Benutzertests durchgeführt werden. Nielsen (1994a) hält hier drei bis fünf Benutzer für ausreichend um 75% der Usability-Probleme aufzudecken. Neuere Ergebnisse zeigen jedoch, dass diese Anzahl insbesondere bei komplexen e-Commerce-Angeboten nicht ausreichend ist (Spool und Schroeder 2001; Perfetti und Landesman 2003). Systeme zur Auswahl komplexer Produkte sind meist anspruchsvoller und umfangreicher als Systeme zur Auswahl fest definierter Produkte. Auch dies stärkt die Bedeutung weiterer Tests.

Da viele Methoden des Usability-Engineering ein Entwicklungsteam voraussetzen, wäre es positiv gewesen, das Projekt im Team und nicht als Einzelperson durchzuführen. Auch für die Modellierung halte ich ein interdisziplinäres Team für sinnvoller, da ich im Rahmen der Arbeit die Erfahrung gemacht habe, dass es zu Missverständnissen kommen kann, wenn die Fachexperten die Möglichkeiten der Modellierung nicht kennen. Im Rahmen der Diplomarbeit hat mir die Modellierung jedoch die Möglichkeit gegeben, mich auch praktisch in das Gebiet der Konfigurierung einzuarbeiten.

Der Einsatz von Papier-Mockups hat sich sehr bewährt, da sie schnell und einfach zu überarbeiten sind. Darüber hinaus bieten sie die Möglichkeit, im Gespräch alternative Entwürfe zu diskutieren. Diesen bedeutenden Vorteilen stehen jedoch beim Entwurf eines wissensbasierten Systems auch Nachteile gegenüber. Sollen Benutzer Arbeitsaufgaben mit Papier-Mockups erledigen, so muss zumindest ein Teil der Leistung des wissensbasierten Systems simuliert werden. Dies erfordert viel Aufmerksamkeit und ein Verständnis der Domäne. Im frühen Entwurfsstadium halte ich Papier-Mockups trotzdem auch bei wissensbasierten Systemen für sehr wertvoll.

## 8.4 Weiterführende Fragen

Wie in Abschnitt 6 beschrieben wurde, können bei einigen Undo-Mechanismen durch das Undo oder Redo Konflikte ausgelöst werden. Dies gilt auch beim direkten Ändern von Entscheidungen. Ob diese Tatsache in der Arbeitspraxis von Experten aber tatsächlich ein Problem darstellt, bedarf eines Testes. Da gerade das referentielle Undo und das direkte Ändern von Entscheidungen sehr wirkungsvolle Unterstützungsansätze sind, halte ich einen solchen Test für sinnvoll.

Wiederaufsetzpunkte bieten auch beim Auftreten von Konflikten eine Unterstützung indem sie den Sprung in einen konfliktfreien Zustand erlauben. Aus Benutzersicht kann es jedoch ebenfalls sinnvoll sein, trotz eines Konfliktes zunächst mit der Produktauswahl fortzufahren, wie es auch bei der Arbeit ohne einen Konfigurator möglich ist. Durch den Einsatz eines Konfigurators kann der Benutzer dabei jederzeit erkennen, ob seine Lösung fehlerfrei ist oder nicht. Ist es möglich die Ursache eines Konfliktes zu vermitteln, so können Benutzer Konflikte nutzen, um ihr Domänenwissen zu erweitern. Hier stellt sich die Frage nach geeigneten Methoden, um die komplexen Abhängigkeiten darzustellen.

Eine weitere offene Frage betrifft den Vergleich von komplexen Produkten. Fest definierte oder individualisierbare Produkte können mit Hilfe von Tabellen einfach verglichen werden. Dies ist bei komplexen Produkten oder Teillösungen nicht immer gegeben. Alternative Lösungen können hier in ihrer Struktur sehr verschieden sein und Teillösungen können außerdem unterschiedlich weit entwickelt sein. Dies macht einen einfachen Vergleich auf Komponentenbasis unmöglich. Daher stellt sich die Frage, welche Filter- und Evaluationsmechanismen beim Vergleich von komplexen Produkten eingesetzt werden können.

Die Kombination von formularbasierter und direktmanipulativer Oberfläche erscheint mir besonders sinnvoll, um Personen zum Ausprobieren eines Systems anzuregen. Welche Auswirkungen der Einsatz der direkten Manipulation auf die Motivation und Zufriedenheit von Benutzern und die Effizienz der Produktauswahl hat, sollte daher genauer untersucht werden.

# Anhang A

## Interviewleitfaden

Vor Start des Interviews:

- Vorstellung
- Projekt das durch die Beobachtung unterstützt werden soll
- Spezielle Ziele dieses Interviews
- Erwartungen an das Interview
- Interviewdauer
- Bei Verwendung eines Diktiergerätes um Erlaubnis fragen. Zweck: Gedächtnisstütze, Verbesserungsmöglichkeiten

Allgemeines:

- Welchen Beruf üben Sie aus und wie lange schon?
- Wie lange arbeiten Sie bereits in Ihrem Unternehmen?
- Welche Ausbildung haben Sie?
- Welche unterschiedlichen Aufgaben haben Sie?
- Welche Aufgaben sind davon besonders wichtig?
- Welche dieser Aufgaben sind häufig / selten?
- Welche Interaktion gibt es zwischen Ihnen und anderen Mitarbeitern? Mit welchen Personen arbeiten Sie zusammen?
- Welche Hilfsmittel (z.b. Software, Kataloge, Listen, Formulare, Personen anrufen, Formelsammlung, Regelwerk) werden eingesetzt?
- Was finden Sie gut / schlecht an den (Software)-Hilfsmitteln?
- Wo arbeiten Sie? (bei welchem Kunden: ausführende Unternehmen, Planer, ...)

Arbeitsprozess:

- Wo werden die „Produktkonfigurationen“ erstellt (beim Kunden)?
- Wie oft erstellen Sie „Produktkonfigurationen“ (Auslegungen, Berechnungen, Parametrierungen)?

- Aus welchen Schritten besteht die „Produktkonfigurierung“ (der Auslegungsprozess)?
- Welche Schritte gehören zusammen?
- Welche Informationen sind bei der „Konfigurierung“ zentral?
- Wie merken Sie sich was sie wie entschieden haben? Was schreiben sie auf?
- Werden „Konfigurationen“ bei der Erstellung oder danach noch einmal verändert?
- Simulieren sie einen typischen Ablauf einer „Produktkonfigurierung“. (think aloud)
- Haben Sie eine feste Vorgehensweise bei der „Produktkonfigurierung“?
- Gibt es Ausnahmen von dieser Vorgehensweise?
- Wie lange benötigt man für das Erstellen einer „Konfiguration“?
- Wodurch werden Sie bei der „Konfigurierung“ zusätzlich gefordert?

## Input:

- Welche Angaben benötigen Sie? (Womit fangen Sie an?)
- Woher bekommen Sie die Informationen für die „Produktkonfigurierung“? (Kennen die Kunden alle benötigten Angaben?) Was fragen Sie den Kunden?

## Output:

- Welche Ergebnisse werden bei der „Produktkonfigurierung“ erarbeitet?
- Wie werden diese Informationen genutzt?

## Verbesserungspotential:

- Wo / Wie könnte man der Arbeitsprozess zusätzlich unterstützen?
- Was ist an der „Konfigurierung“ besonders zeitaufwändig?
- Welche Tatsachen erschweren Ihre Arbeit?
- Gibt es immer wieder auftretende Arbeitsschritte / Dinge die Sie stören?
- Welche Probleme treten bei der „Produktkonfigurierung“ auf? Wie begegnen Sie diesen Problemen?
- Komplexe Aufgabe, wie machen Sie das? Was geht manchmal daneben? Welche Fehler treten auf?
- Wie erkennen und beheben Sie die Fehler?

## Ändern und Vergleichen:

- Wieviel unterschiedliche Lösungen erstellen Sie in einem Kundengespräch? Ändern Sie oder erstellen Sie jedesmal ganz neu?
- Sind die Lösungen, die Sie erstellen oft sehr ähnlich?
- Passiert es manchmal, dass Auslegungen noch einmal verändert werden müssen z.B. weil der Kunde nicht alle Angaben gemacht hat, etwas ver-

gessen wurde, Lösung nicht günstig genug ist?

- Kommt es vor, dass Auslegungen nicht ganz stimmig sind? Wann fällt das auf und was wird dann gemacht?
- Als was würden Sie die „Konfigurierung“ beschreiben? z.B. Konstruktion eines Produktes aus Komponenten, Auswahl eines Produktes aus einer sehr großen Produktmenge, ...

Unterstützungsmöglichkeiten:

- Welche Dinge würden Sie am liebsten ändern?
- Welche Visualisierungen wären zur Unterstützung hilfreich?
- Brauchen Sie die Möglichkeit unterschiedliche „Konfigurationen“ zu vergleichen?
- Wäre es sinnvoll alle möglichen „Konfigurationen“ (z.B. in einer Tabelle) anzuzeigen, wenn nur noch eine überschaubare Menge von Lösungen möglich ist?
- Welche Verbesserungsmöglichkeiten sehen Sie? (z.B. Automatische Berechnung der noch möglichen Knotenzahl?)

Erfahrung mit Konfigurierungs-Tools:

- Haben Sie schon einmal ein Konfigurierungs-Tool benutzt (z.B. Internet)? (Wenn ja was für ein Tool? Was wurde konfiguriert?)
- Was war gut an dem Tool?
- Was war nicht so überzeugend?
- Was könnte / sollte man anders machen?

Kunden:

- Haben sie einen festen Kundenstamm den sie betreuen?
- Kundenziele vs. Vertrieblerziele?
- Wie oft sind sie beim gleichen Kunden?
- Wie lange dauert ein Kundenbesuch?
- Welche Ausbildung haben Kunden?
- Was ist für Kunden in ihrem Beruf besonders wichtig?
- Wodurch unterscheidet sich das Wissen der Kunden von dem von Vertriebsmitarbeitern (allg. Computererfahrung, Wissen über die Domäne und Wissen über Konfigurierungsanwendungen)?
- Was ist für Kunden bei der „Produktkonfigurierung“ besonders wichtig?
- Welche Probleme haben Kunden bei der „Konfigurierung“ von Produkten (Ursachen)?
- Wie könnten sie unterstützt werden?

Bedanken für die investierte Zeit!



# Anhang B

## Szenarien

### Szenario 1

Benutzer: Planer und Vertriebsmitarbeiter (Außendienst)

Aufgabe: Anforderungsprofil beim Kunden (Planer) ermitteln

#### Aufgabenkontext

Ziel des Planers ist es eine Ausschreibung für die Gebäudeautomation in einem Krankenhaus herauszugeben. Ziel des Außendienstmitarbeiters ist es den Planer vom LON-Einsatz zu überzeugen und ihn bei der Erstellung einer Ausschreibung zu unterstützen. Im Gespräch vermittelt der Vertriebsmitarbeiter zunächst die Möglichkeiten, die sich mit LON-Einsatz bieten. Auf dieser Grundlage versucht der Vertriebsmitarbeiter durch gezielte Fragen zum Gebäude und den dort zu realisierenden Funktionen die Angaben zu ermitteln, die er für das Erstellen eines Konzeptes für die Gebäudeautomation braucht.

#### Arbeitsablauf

- Der Vertriebsmitarbeiter fragt den Planer nach den Daten eines typischen Krankenzimmers (Größe des Raumes, Anzahl der Betten, Anzahl der Fenster, Anzahl der Beleuchtungsgruppen, Anzahl der Heizkörper, Jalousien vorhanden?). Nach diesen Angaben fertigt der Vertriebsmitarbeiter Notizen zum Raum an:
  - 2 Betten
  - 2 Fenster
  - 2 Beleuchtungsgruppen
  - 1 Lesebeleuchtung je Bett (2 Beleuchtungsgruppen)
  - 1 Heizkörper
  - keine Jalousien
- Der Vertriebsmitarbeiter erläutert dem Planer die unterschiedlichen Möglichkeiten der Beleuchtungssteuerung: schalten, dimmen, Konstantlichtregelung.
- Der Vertriebsmitarbeiter notiert sich die Anforderungen des Planers (zwei Beleuchtungsbänder an der Decke: Konstantlichtregelung, Lesebeleuch-

tung je Bett nur schaltbar).

- Der Vertriebsmitarbeiter nennt die Möglichkeiten die Beleuchtung über Präsenzmelde oder/und Schalter zu schalten.
- Der Vertriebsmitarbeiter notiert sich, dass Deckenbeleuchtung und Lesebeleuchtung geschaltet werden sollen.
- Der Vertriebsmitarbeiter erkundigt sich wie die Heizung geregelt werden soll: Nachtabsenkung der Raumtemperatur, Absenkung der Raumtemperatur in nicht belegten Zimmern, Energiesparmodus bei geöffnetem Fenster.
- Der Vertriebsmitarbeiter notiert sich die Anforderungen des Planers (Absenkung der Raumtemperatur in nicht belegten Zimmern und Energiesparmodus bei geöffnetem Fenster).
- Der Vertriebsmitarbeiter erkundigt sich von welchen Orten aus Beleuchtung und Heizung bedient werden sollen.
- Der Vertriebsmitarbeiter vermerkt die Orte in seiner Skizze und notiert sich die Anzahl der benötigten Bedienelemente.
- Der Vertriebsmitarbeiter zeigt dem Planer unterschiedliche Bedienelemente, die für die Bedienung in Frage kommen und notiert sich die Vorstellungen des Planers.
- Für die spätere Festlegung der Netzwerktechnik erkundigt sich der Vertriebsmitarbeiter wieviel identische Krankenzimmer auf einer Station vorhanden sind und wieviel Stationen das Gebäude enthält. Er notiert sich die Informationen.
- Der Vertriebsmitarbeiter erkundigt sich welche weiteren Räume zu berücksichtigen sind (Stationszimmer, Küche, Aufenthaltsraum, etc.) und notiert sich diese Angaben.

**Zeitbedarf für diese Aufgabe:** ca. 1-2 Stunden

## Szenario 2

Benutzer: Planer und Vertriebsmitarbeiter (Außendienst)

Aufgabe: Erstellen einer Kostenschätzung

### Aufgabenkontext

Im Gespräch mit dem Kunden (Planer) hat der Außendienstmitarbeiter ein Anforderungsprofil ermittelt. Dies setzt er im Home Office in eine Kostenschätzung um. Nach der Erstellung schickt er die Kostenschätzung zusammen mit einer Funktionsbeschreibung dem Planer. Der Planer erstellt auf der Basis dieser Dokumente eine Ausschreibung.

### Arbeitsablauf

- Der Vertriebsmitarbeiter nimmt die Notizen aus dem Kundengespräch hervor und sieht sie durch:
- Für die Steuerung der beiden Beleuchtungsgruppen an der Decke wählt der Vertriebsmitarbeiter den DALI-Controller REG-S 4 DIM.

- Da mit diesem Steuergerät aber jeweils 4 Beleuchtungsgruppen gesteuert werden können, fasst er zwei Zimmer als kleinste Planungseinheit zusammen.
- Für die gewünschte Konstantlichtregelung wird ein Multisensor (Multisensor LA-21) benötigt. Für die Planungseinheit (2 Zimmer) notiert sich der Vertriebsmitarbeiter 2 Multisensoren.
- Für die Steuerung der Lesebeleuchtung am Bett notiert sich der Vertriebsmitarbeiter das Gerät I/O-Modul REG-M 4S einmal, da es auch 4 Beleuchtungsgruppen steuert.
- Der Vertriebsmitarbeiter notiert sich, dass für die Heizung ein Stellantrieb benötigt wird.
- Für die Absenkung der Raumtemperatur bei Abwesenheit wird ein Multisensor benötigt. Da dieser aber schon für die Konstantlichtregelung vorhanden ist, notiert sich der Vertriebsmitarbeiter kein weiteres Gerät.
- Für die Planungseinheit notiert sich der Vertriebsmitarbeiter zwei Busankoppler mit Raumtemperaturpanel für die Heizung.
- Für die Deckenbeleuchtung wird pro Zimmer ein Busankoppler mit 2fach Taster benötigt. Daher notiert sich der Vertriebsmitarbeiter für die Planungseinheit 2 Busankoppler mit 2fach Taster.
- Für die Lesebeleuchtung am Bett wird je Bett ein Busankoppler mit 1fach Taster benötigt. Daher notiert sich der Vertriebsmitarbeiter vier 1fach Taster.
- Der Vertriebsmitarbeiter gibt die gewählten Geräte in eine Excel-Anwendung ein.
- Die Summe der ausgewählten Geräte sowie ihre mittlere Leistungsaufnahme in LPU kann der Vertriebsmitarbeiter der Anwendung entnehmen.
- Aufgrund dieser Informationen entscheidet der Vertriebsmitarbeiter, dass 2 Spannungsversorgungen mit Router (Spannungsversorgung mit Router LPS/RTR-W) benötigt werden.
- Der Vertriebsmitarbeiter fügt diese Geräte der erstellten Liste hinzu.
- Die Stückliste der Geräte sowie deren Preise bilden die Kostenschätzung für den Planer.

**Zeitbedarf für diese Aufgabe:** ca. 1,5 Stunden

### **Szenario 3**

Benutzer: Planer und Vertriebsmitarbeiter (Außendienst)

Aufgabe: Erstellen einer Funktionsbeschreibung

#### **Aufgabenkontext**

Im Gespräch mit dem Kunden (Planer) hat der Außendienstmitarbeiter ein Anforderungsprofil ermittelt. Dies setzt er im Home Office in eine Funktionsbeschreibung um. Nach der Erstellung schickt er die Funktionsbeschreibung zusammen mit einer Kostenschätzung dem Planer. Der Planer erstellt auf der Basis dieser

Dokumente eine Ausschreibung.

#### **Arbeitsablauf**

- Der Vertriebsmitarbeiter nimmt die Notizen aus dem Kundengespräch hervor und sieht sie durch.
- Auf der Grundlage seiner Notizen beschreibt der Vertriebsmitarbeiter in einem frei formulierten Text die Funktionen der Beleuchtungssteuerung.
- Entsprechend seiner Notizen beschreibt der Vertriebsmitarbeiter die Funktionen der Sonnenschutzsteuerung.
- Der Vertriebsmitarbeiter stellt die Funktionen der Einzelraumregelung dar.
- Entsprechend seiner Notizen beschreibt der Vertriebsmitarbeiter die vorge-sehene Bedienung.
- Der Vertriebsmitarbeiter notiert die Anzahl gleicher Räume.
- Der Vertriebsmitarbeiter korrigiert eventuelle Rechtschreibfehler.
- Der Vertriebsmitarbeiter nimmt Veränderungen am Layout des Textes vor.
- Der Vertriebsmitarbeiter liest den geschriebenen Text einmal durch.

**Zeitbedarf für diese Aufgabe:** ca. 0,5 Stunden

#### **Szenario 4**

Benutzer: Planer und Vertriebsmitarbeiter (Außendienst)

Aufgabe: Vermitteln der LON-Vorteile und Einsatzmöglichkeiten

#### **Aufgabenkontext**

Im Gespräch mit dem Planer stellt der Außendienstmitarbeiter dem Planer die Konzepte und Vorteile von LON vor (z.B. Energieeinsparung, Flexibilität in der Gebäudenutzung, gewerkeübergreifende Integration, usw.). Er schildert unterschiedliche Einsatzmöglichkeiten und Potentiale von LON und stellt die Vorteile gegenüber herkömmlichen Einzelsystemen heraus.

**Zeitbedarf für diese Aufgabe:** ca. 1 - 1,5 Stunden

#### **Szenario 5**

Benutzer: Vertriebsmitarbeiter (Innendienst)

Aufgabe: Erstellen eines Angebotes auf Basis einer funktionalen Ausschreibung

#### **Aufgabenkontext**

Die meisten Angebote werden im Innendienst erstellt. Als Grundlage für die Angebotserstellung kommen unterschiedliche Ausschreibungstypen vor: funktionale Ausschreibungen (20%), Massen-Leistungsverzeichnisse (50%), andere Bussysteme (30%). Das erstellte Angebote besteht aus einem Standardtext, einer Stückliste mit Preisen sowie Erläuterungen zum Angebot.

#### **Arbeitsablauf**

- Der Vertriebsmitarbeiter nimmt die funktionale Ausschreibung hervor und liest sie.

- Beim Lesen notiert er sich die wichtigsten Angaben aus dem Text (Anzahl Standardräume, Anzahl Fenster, Anzahl Sonderräume, zu realisierende Funktionen in den Bereichen Beleuchtungssteuerung, Sonnenschutzsteuerung, Heizungs- und Klimasteuerung, etc.).
- Der Vertriebsmitarbeiter legt die Geräte fest, die für die Beleuchtungssteuerung in den Standardräumen benötigt werden.
- Das Telefon klingelt. Ein Außendienstmitarbeiter benötigt spezielle Informationen zu einem Produkt eines Fremdherstellers.
- Der Vertriebsmitarbeiter schlägt die gesuchte Information im Katalog dieses Herstellers nach und teilt sie dem Außendienstmitarbeiter mit.
- Der Vertriebsmitarbeiter liest seine Notizen zur geforderten Sonnenschutzsteuerung.
- Für die Sonnenschutzsteuerung wird ein Multisensor benötigt. Der Vertriebsmitarbeiter prüft, ob er für die Beleuchtungssteuerung bereits einen Multisensor vorgesehen hat.
- Der Vertriebsmitarbeiter legt die Geräte fest, die für die Sonnenschutzsteuerung in den Standardräumen benötigt werden.
- Der Vertriebsmitarbeiter überprüft, ob ein Kombi-Gerät zur gemeinsamen Steuerung von Beleuchtung und Jalousien eingesetzt werden kann, um eine günstigere Lösung zu erhalten.
- Der Vertriebsmitarbeiter liest seine Notizen zur geforderten Heizungs- und Klimasteuerung.
- Der Vertriebsmitarbeiter legt die Geräte fest, die für die Heizungs- und Klimasteuerung in den Standardräumen benötigt werden.
- Der Vertriebsmitarbeiter liest seine Notizen zur geforderten Bedienung der Raumfunktionen und legt die notwendigen Bedienelemente fest.
- Der Vertriebsmitarbeiter überlegt sich welche Geräte außerhalb der Standardräume benötigt werden, wie z.B. physikalische Sensoren.
- Der Vertriebsmitarbeiter liest seine Notizen zu den Sonderräumen (Konferenzraum und Flure).
- Der Vertriebsmitarbeiter ist gerade dabei die benötigten Geräte zu ermitteln, als das Telefon klingelt.
- Ein Planer hat eine Nachfrage zur Beleuchtungssteuerung in einem Angebot.
- Der Vertriebsmitarbeiter beantwortet die Fragen des Planers.
- Der Planer bittet um ein alternatives Angebot mit einer veränderten Funktionalität (ausschließlich schaltbare Beleuchtung anstelle einer Konstantlichtregelung).
- Der Vertriebsmitarbeiter sichert zu, das Angebot in den nächsten Tagen zu erstellen und verabschiedet sich.
- Er macht sich Notizen zu den Angaben des Planers.

- Der Vertriebsmitarbeiter sieht nochmals seine Notizen zu den Sonderräumen durch und ermittelt die benötigten Geräte.
- Der Vertriebsmitarbeiter betrachtet die Anzahl der verwendeten Geräte und deren Leistungsaufnahme.
- Um die benötigte Netzwerktechnik festlegen zu können, sucht der Vertriebsmitarbeiter nach Angaben zur räumlichen Ausdehnung des geplanten GA-Systems. Da die funktionale Ausschreibung hierzu keine Angaben enthält ruft der Vertriebsmitarbeiter den zuständigen Planer an.
- Der Planer gibt dem Vertriebsmitarbeiter die notwendigen Angaben zur Größe der Räume.
- Aufgrund dieser Angaben sowie der Anzahl der Geräte und deren Leistungsaufnahme ermittelt der Vertriebsmitarbeiter die Anzahl der Router und Terminatoren, die gebraucht werden.
- Der Vertriebsmitarbeiter geht die Stückliste der Geräte noch einmal durch.
- Zum Schluss formuliert der Vertriebsmitarbeiter die Erläuterungen zum Angebot.

**Zeitbedarf für diese Aufgabe:** ca. 5 Stunden

### **Szenario 6**

Benutzer: Vertriebsmitarbeiter (Innendienst)

Aufgabe: Erstellen eines Angebotes auf der Grundlage eines Massen-LV

#### **Aufgabenkontext**

Das zu erstellende Angebot besteht aus einem Standardtext, einer Stückliste und Erläuterungen zum Angebot. Das Angebot wird auf der Grundlage einer Ausschreibung erstellt. Es können drei Klassen von Ausschreibungen unterschieden werden:

- funktionale Ausschreibungen (20%)
- Massen-Leistungsverzeichnisse (50%)
- Ausschreibungen die sich auf andere Bussysteme beziehen (30%)

Im Massen-Leistungsverzeichnis sind im Gegensatz zur funktionalen Ausschreibung keine Angaben enthalten, die Beschreiben welche Funktionen mit den Geräten realisiert werden sollen. Dies macht Rückfragen beim Planer notwendig. Die im Massen-LV aufgeführten Geräte können auch von Fremdherstellern sein und müssen dann in entsprechende Geräte von SVEA übersetzt werden.

### **Szenario 7**

Benutzer: Vertriebsmitarbeiter (Innendienst)

Aufgabe: Umsetzen von Änderungswünschen an einem Angebot

#### **Aufgabenkontext**

Kunden äußern teilweise sogar mehrfach Änderungswünsche am Angeboten.

Beim Umsetzen dieser Änderungen können weitere Änderungen notwendig werden. Die Stückliste muss daher vom Vertriebsmitarbeiter noch einmal kontrolliert werden, um solche Folgeänderungen nicht zu übersehen.

**Zeitbedarf für diese Aufgabe:** einige Minuten bis Stunden

## Szenario 8

Benutzer: Kundenbetreuung

Aufgabe: Vertriebsmitarbeiter (Innendienst)

### Aufgabenkontext

SVEA bietet Ihren Kunden im WWW eine Produktdatenbank an. Hier haben Kunden die Möglichkeit Angebotsanfragen zu erstellen.

### Arbeitsablauf

- Das Telefon klingelt. Ein Kunde (Planer) interessiert sich für die Möglichkeiten der Beleuchtungssteuerung in einem konkreten Projekt.
- Der Vertriebsmitarbeiter erläutert die unterschiedlichen Varianten und erkundigt sich nach den Rahmenbedingungen (Art und Anzahl der Räume, Anzahl der Beleuchtungsgruppen).
- Der Kunde erkundigt sich nach den Kosten für die Realisierung der unterschiedlichen Varianten. Er hat den Produktkatalog im WWW schon besucht, um sich einen Kostenüberblick zu verschaffen, wusste aber nicht welche Geräte in Frage kommen.
- Der Vertriebsmitarbeiter schlägt dem Kunden vor, gemeinsam in der Online-Produktdatenbank eine Angebotsanfrage zu erstellen.
- Vertriebsmitarbeiter und Kunde rufen die Online-Produktdatenbank auf.
- Der Vertriebsmitarbeiter erläutert dem Kunden welche Geräte benötigt werden und in welcher Anzahl.
- Beide legen die entsprechenden Geräte in den Warenkorb der Produktdatenbank.
- Der Vertriebsmitarbeiter läßt sich den Inhalt des Warenkorbs anzeigen und bittet den Kunden gleiches zu tun.
- Der Vertriebsmitarbeiter nennt dem Kunden zur Kontrolle noch einmal Art und Anzahl der Gerät im Warenkorb.
- Der Vertriebsmitarbeiter weist den Kunden auf die Möglichkeit hin, eine Angebotsanfrage zu stellen.
- Der Kunde gibt seine Daten im Online-Formular an um ein Angebot zu erhalten.
- Der Vertriebsmitarbeiter bittet den Kunden, sich bei Nachfragen an ihn zu wenden und verabschiedet sich.

**Zeitbedarf für diese Aufgabe:** ca. 0,5 Stunden

# Anhang C

## Dokumente aus dem Anwendungsbeispiel SVEA

Pos.	Leistungsbeschreibung	Menge	Preis je Einheit	Gesamt-betrag
	<p><b>LON Busankoppler mit Raumtemperaturpanel und Taster 4fach</b></p> <p>Raumbedieneinheit für LON Netzwerke bestehend aus Raumtemperaturpanel, Taster 4fach, Rahmen 2fach sowie 2 Stück LON Busankopplern.</p> <p>Raumtemperaturpanel als Bedieneinheit für LON-Einzelraumregler mit integriertem Temperatursensor. Anzeige von bis zu 5 Betriebszuständen über unterschiedliche LED's möglich, wie z.B. Komfort, Standby, Nacht, Frost-/ Hitzeschutz und Taupunktalarm. Mit Präsenztaster z.B. zum Umschalten des Reglers zwischen Standby und Komfort-Betrieb. Mit Stellrad zur manuellen Sollwertverschiebung mit konfigurierbarem Verstellbereich. Mit Software-Applikation gemäß LonMark-Profil "Thermostat (8060)"</p> <p>Taster 4fach mit acht Tastflächen zum Abrufen der über die Softwareapplikation zugeordneten Funktionen. Die Betriebsbereitschaft wird durch LED's angezeigt. Für jedes Tastenpaar gibt es eine gemeinsame Status-Anzeige. Mit Software-Applikationen zum Dimmen und Schalten von Beleuchtung, zur Jalousie- oder Raumszenensteuerung gemäß den geltenden LonMark-Profilen.</p> <p>Busanschluß: 2 x LON Link-Power-Transceiver (LPT-10) Bedienelemente: Sollwertversteller und Präsenztaste 8 Bedientasten Anzeige: 5 LED's zur Anzeige des Regler-Betriebszustandes 8 Betriebsanzeigen für den Taster 1 Statusanzeige je Tastenpaar Abmessungen: Einbautiefe jeweils ca. 32mm in 60mm Einbaudose Farbe: polarweiß Design: Merten ARTEC Schutzart: IP20 Zul. Luftfeuchtigkeit: 0 bis 95% , keine Betauung Zul. Umgebungstemp.: -5 °C bis +45 °C</p> <p><b>Fabrikat: SVEA</b> <b>Artikel-Nr. 63315-170</b> oder gleichwertig</p>			

Abbildung C.1: Beispiel für eine Gerätebeschreibung im Massen-LV: Ausschreibungstext für ein LON Busankoppler mit Raumbedienpanel und Taster 4fach



Seite 119

---

Pos.-Nr.:	Menge	Einheit	Beschreibung	EP	GP
-----------	-------	---------	--------------	----	----


**Titel 4.16                      Sonnenschutzanlage**



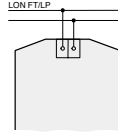
**Vorbemerkungen Sonnenschutzanlage**  
Auf der Südsseite des Gebäudes werden die einzelnen Räume mit einzelnen außenliegenden Jalousieanlagen ausgestattet. Die busfähige Anlage besteht aus folgenden Komponenten:  
- Steuerzentrale zur Einzel- und Gruppensteuerung  
- Motorsteuergeräte  
- Jalousietaster  
- Sonnensensor  
- Windsensor  
- Verteiler  
- Leitungsnetz  
Die Bedientaster vor Ort werden im Bereich der Außenfassade montiert. Ein Wind- und Sonnenwächter sorgt für eine licht- und windabhängige Steuerung über einen Zentralbefehl. Weiterhin ist beim Hausmeister ein Bedientableau vorgesehen, das die gruppenweise Ansteuerung der Jalousieanlagen ermöglicht. Die Anlage ist als Komplettanlage funktionsfertig zu liefern und zu montieren. Benötigte und nicht aufgeführte Komponenten sind anzuzeigen und in die Einheitspreise einzurechnen. Im Leistungsumfang nicht enthalten sind die Raffstore. Die Anlage basiert auf einer dezentrale Anordnung der Anlagenbausteine, der Datenaustausch erfolgt über ein Bussystem. Die elektro-mechanischen Schaltelemente (Motorsteuergeräte) befinden sich im Deckenbereich in der Nähe des Antriebes.

4.16.10	15,00	St	<b>Verbindungsdose, a.P/FR</b>  Allgemeine Angaben: Verbindungsdose DIN VDE 0605 aus Isolierstoff, als Abzweigdose in Aufputzausführung.  Besondere Angaben: keine  Technische Daten: Grundfläche : 80*80 Schutzart : IP 54  Planer                                              Bieter  Fabrikat:  Typ:  oder gleichwertig, komplett mit erforderlichem Zubehör liefern und betriebsfertig montieren.		
			Übertrag	Euro	

Abbildung C.2: Ausschnitt aus einem Massenleistungsverzeichnis

## LON Busankoppler mit Raumtemperaturpanel und Taster 4fach 63315-170



### 1 Artikelbeschreibung

- Kombination aus Bedieneinheit für Temperaturregler und Taster 4fach im Design Merten-ARTEC komplett mit zwei LON-Busankopplern und Rahmen 2fach
- Bedieneinheit für LON-Einzelraumregler (z.B. REG-T KDS, Art.-Nr. 61232-043) mit integriertem Temperatursensor
- Anzeige von Betriebszuständen wie Komfort, Standby, Nacht, Frost-/Hitzeschutz und Reglersperre über LED's
- Präsenztaster z.B. zur Umschaltung zwischen Standby- und Komfort-Betrieb
- Stellrad zur Sollwertverschiebung
- Softwareapplikation gemäß LonMark-Profil "Space Comfort Controller Module (8090)"
- Taster mit acht Tastflächen zur individuellen Funktionszuordnung
- Anzeige der Betriebsbereitschaft über Lichtkreise um die Bedientasten
- mit einer Status-Anzeige je Tastenpaar
- mit Softwareapplikationen zur Umsetzung der Signale der Bedieneinheit in Telegramme zur Licht-, Jalousie- oder Raumszenensteuerung gemäß den geltenden LonMark-Richtlinien
- Schutzart: IP 20
- Farbe: polarweiß

NEU!
LPT-10
Disk

### 2 Funktion

<p>Steuereinheit für LON-Netzwerke bestehend aus Bedieneinheit für Einzelraumregler mit integriertem Temperatursensor, Universal 4fach Taster, zwei LON-BCU® und 2fach Rahmen.</p> <p>Das Bedienungs-Anwendungsmodul besitzt eine Präsenztaste und ein Stellrad zur Sollwertverschiebung. Der Regler-Status wird über LEDs angezeigt.</p> <p>Das Taster-Anwendungsmodul besitzt 8 Tastflächen. Die Betriebsbereitschaft und</p>	<p>der Status werden über Lichtkreise um den Bedientasten angezeigt.</p> <p>Über die Tastflächen können die Applikationsbedingten Funktionen abgerufen werden. Es stehen Applikationen zur Licht- und Jalousiesteuerung, sowie zur Szenensteuerung zur Verfügung.</p> <p>Der LON-Busankoppler (LON-BCU®) ist das Basismodul für UP-Anwendungsmodule mit steckbarer topoliger EIB-kompatibler</p>	<p>Anwendungsschnittstelle (AST).</p> <p>Der Busankoppler stellt die mechanische, elektrische und datentechnische Verbindung zwischen einem LonWorks FT/LP-Netzwerk und dem Anwendungsmodul her. Die vom Bus empfangenen Telegramme werden ausgewertet und als Befehl über die AST an das Anwendungsmodul weitergeleitet. In Gegenrichtung werden Signale des Anwendungsmoduls in Telegramme gewandelt und auf den Bus gesendet.</p>
---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 3 Montage

<p>Der LON-Busankoppler ist für die Montage in Untertafelgehäusen geeignet. Er wird mit Schrauben an der Installationsdose befestigt. Eine Befestigung mittels Krallen ist nicht möglich. Das Anwendungsmodul wird auf den LON-Busankoppler aufgesteckt.</p>	<p>Der Netzwerkananschluß erfolgt über eine zweipolige Busanschlußklemme, die eine Anschlußmöglichkeit für bis zu 4 Adernpaare bietet. Es können Adern mit einem Querschnitt von 0,6 bis 0,8 mm<sup>2</sup> angeschlossen werden.</p> <p>Durch Drücken der Servicetaste wird der Busankoppler veranlaßt, seine Neuron-ID zu</p>	<p>senden. Die Service-LED zeigt den Programmierstatus an.</p> <p>Für den Betrieb des Gerätes ist ein Applikationsprogramm erforderlich. Das Applikationsprogramm ist bei SVEA separat zu bestellen und mit einem Netzwerk-Managementtool in das Gerät zu laden.</p>
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 4 Anmerkungen


<p>Einbau und Montage elektrischer Geräte dürfen nur durch eine Elektrofachkraft erfolgen.</p> <p>Bei der Planung und Errichtung von elektrischen Anlagen sind die einschlägigen Normen, Richtlinien, Vorschriften und Bestimmungen des jeweiligen Landes zu beachten. Darüber hinaus sind die</p>	<p>Gerätespezifikationen einzuhalten. Für die Projektierung, Montage und Inbetriebnahme werden detaillierte Fachkenntnisse der LonWorks-Technologie vorausgesetzt.</p> <p>Die Funktion des Gerätes ist softwareabhängig. Mittels eines Netzwerk-Managementtools kann das Gerät konfiguriert</p>	<p>werden. Es dürfen nur Applikationsprogramme geladen werden, die von SVEA für dieses Gerät freigegeben sind.</p> <p>Der Anlagenerrichter hat dafür Sorge zu tragen, daß die Beschaltung des Gerätes der gewählten Konfiguration entspricht.</p>
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Bearbeitungsstand: 19.06.2001  
Änderungen vorbehalten.

**LON Busankoppler mit Raumtemperaturpanel und Taster 4fach (63315-170)**

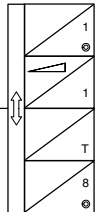
Seite 1

Abbildung C.3: Beispiel für eine Produktdokumentation; Teil 1



## 5 Technische Daten

<b>Versorgung</b>	
Leistungsaufnahme aus dem Netzwerk:	ca. 400 mW (200 mW je LON-BCU) bei DC 42,8 V Busspannung
<b>Netzwerkinterface</b>	
Transceivertyp:	LON Link-Power-Transceiver (LPT-10)
<b>Ausgänge LON-BCU</b>	
Anzahl:	2 (1 je LON-BCU)
Typ:	10polige EIB-kompatible Anwenderschnittstelle (AST)
Nennspannungen:	DC 5 V, DC 24 V (4 mA, 2 mA je LON-BCU)
Nennleistung:	50 mW
<b>Bedienelemente LON-BCU</b>	
Servicetaster:	zum Senden der Neuron-ID
<b>Bedienelemente Anwendungsmodule</b>	
Präsenztaste:	zum Wechseln des Belegt-Status
Stellrad:	zur Sollwertverschiebung
<b>Anzeigeelemente LON-BCU</b>	
Service-LED (rot):	AN: Fehler im Netzwerkzugriff BLINKT: Modul unkonfiguriert
<b>Anzeigeelemente Anwendungsmodule</b>	
5 Status-LEDs (grün):	zur Anzeige des Reglerstatus
<b>Anschlüsse LON-BCU</b>	
Bus:	2poliger steckbarer Klemmschluß für Querschnitte 0,6...0,8 mm <sup>2</sup> (Massivadern), je Pol sind 4 Busleitungen anschließbar
Anwenderschnittstelle (AST):	10polige Buchsenleiste
<b>Gehäuse LON-BCU</b>	
Abmessungen:	Einbautiefe 32 mm
Schutzart:	IP20 nach DIN 40050 / IEC 144
Befestigungsart:	Einbau in Unterputzdosen durch Schraubbefestigung (60 mm)
<b>Umgebungsbedingungen</b>	
Lagertemperatur:	-40 °C ... +55 °C
Betriebstemperatur:	-5 °C ... +40 °C
relative Luftfeuchte:	5 ... 93 % ohne Betauung
<b>EMV-Spezifikation</b>	
Störfestigkeit:	gemäß EN 50082-2
8 Bedientasten:	zum Aufrufen der applikationsbedingten Funktion
Lichtkreise um den Bedientasten:	GRÜN: betriebsbereit / AUS-Zustand ROT: EIN-Zustand



Bearbeitungsstand: 19.06.2001  
 Änderungen vorbehalten.

**LON Busankoppler mit Raumtemperaturpanel und Taster 4fach (63315-170)**

Seite 3

Abbildung C.4: Beispiel für eine Produktdokumentation; Teil 2

# Anhang D

## Ergebnisse aus dem Anwendungsfall SVEA

### Liste wichtiger Fachbegriffe

<b>Begriffe</b>	<b>Erläuterungen</b>
GA-Lösung	Alternative: GA-System
Raumkonzept	
Raumfunktion	
Gerätetypen	REG (Reiheneinbaugeräte), AP (Aufputz), UP (Unterputz)
Geräte mit Steckverbindingssystem	z.B. gesis
Zwischendecke	Montageort
Hohlboden	Montageort
Verteilerschrank	Montageort, Alternativ: Reiheneinbau
Beleuchtungssteuerung	
Beleuchtungsgruppe	Eine oder mehrere Beleuchtungseinheiten; gemeinsam gesteuert.
Beleuchtungseinheit	Eine Leuchte
schaltbar	Beleuchtungstyp
dimmbar	Beleuchtungstyp
Konstantlichtregelung	Beleuchtungstyp
Leuchtenart	ist entscheidend bei dimmbarer Beleuchtung oder Konstantlichtregelung
DALI-Geräte	Leuchtenart
dimmbare EVGs	Leuchtenart
elektronische Transformatoren	Leuchtenart
Glühlampen	Leuchtenart
wird fortgesetzt	

<b>Begriffe</b>	<b>Erläuterungen</b>
HV-Hologenlampen	Leuchtenart
dimmbare gewickelte Transformatoren	Leuchtenart
Sonnenschutzsteuerung	
Jalousien	
Jalousienmotoren	Über Jalousienmotoren werden auch Fenster, Leinwände, Rolllöre und Sonnensegel gesteuert.
Einzelraumregelung	
HLK-Steuerung	Steuerung von Heizung, Lüftung, Klima
Heizkörper	
Nachtabsenkung	
Energiesparmodus	
anwesenheitsabhängig	z.B. Standbytemperatur; Präsenzerfassung über Taster oder Melder
außentemperaturabhängig	
helligkeitsabhängig	
innentemperaturabhängig	
windabhängig	z.B. Jalousienschutz
zeitabhängig	z.B. Nachtabsenkung
Bedienung	
manuell-Vorort	
manuell-Zentral	
Bedienung über LON-Taster	
konventioneller Taster / Schalter	
Netzwerktechnik	
Kostenschätzung	Arbeitsergebnis
Massen-LV	Arbeitsergebnis
Funktionale Ausschreibung	Arbeitsergebnis
Sensoren (Montageort: Innen)	Geräteklasse
Sensoren (Montageort: Außen)	Geräteklasse
Präsenzmelder	Geräteklasse
Digital-Eingänge	Geräteklasse
Digital-Ausgänge	Geräteklasse
Kombi-Ein-/Ausgänge	Geräteklasse
Dimmer	Geräteklasse
Steuerausgänge	Geräteklasse; Alternativ: 1-10V-Dimmer
Stellantrieb	Geräteklasse
Uhr	Geräteklasse
Bedienelemente	Geräteklasse
HLK-Bedienelemente	Geräteklasse

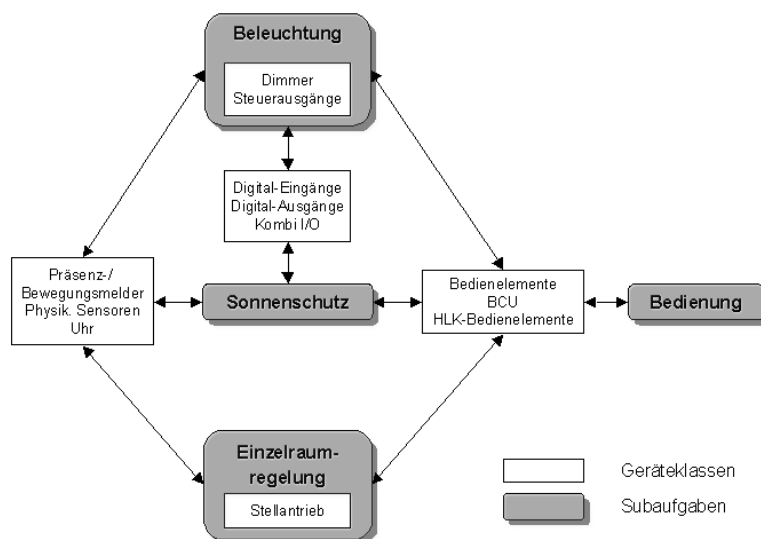


Abbildung D.1: Abhängigkeiten zwischen Subaufgaben und Geräteklassen beim Zusammenstellen von LON-basierten GA-Systemen, Teil 1

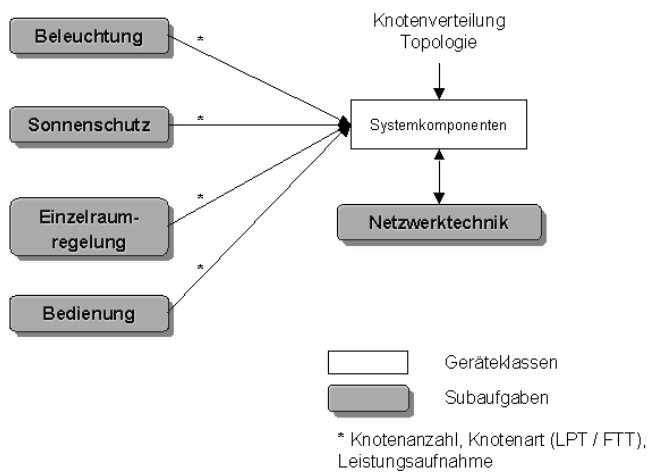


Abbildung D.2: Abhängigkeiten zwischen Subaufgaben und Geräteklassen beim Zusammenstellen von LON-basierten GA-Systemen, Teil 2

## Navigationseinheiten des Prototyps:

**Projektdaten:** Projektname + Anzahl der Raumkonzepte  
für jedes Raumkonzept:  
Namen + Häufigkeit des Raumes

## für jedes Raumkonzept:

**Gerätetypen:** alle / nur Steckverbindingssysteme  
**Montagebedingungen:** alle, Verteilerschrank, Hohlboden, Zwischendecke

**Beleuchtungssteuerung:** ja / nein

**Anzahl der Beleuchtungsgruppen im Raum:** 1- x

für jede Beleuchtungsgruppe:

**Beleuchtungstyp:** schaltbar / dimmbar / Konstantlichtregelung

wenn dimmbar, Konstantlicht:

**Leuchtenart:** DALI-Geräte / Dimmbare EVGs / ... / Glühlampen

**Steuerung:** manuell-Vorort, manuell-Zentral, anwesenheitsabhängig, zeitabhängig

wenn manuell-Vorort:

**Schalterart:** LON / Konventionell

wenn manuell-Zentral:

**Schalterart:** LON / Konventionell

wenn anwesenheitsabhängig:

**Präsenzerfassung:** über Präsenztaster / über Präsenzmelder

wenn Präsenzmelder:

**Raummaße**

**Sonnenschutzsteuerung:** ja / nein

**Anzahl der Jalousien im Raum:** 1 - x

**Steuerung:** manuell-Vorort, manuell-Zentral, anwesenheitsabhängig, helligkeitsabhängig, innentemperaturabhängig, außentemperaturabhängig, windabhängig, helligkeitsabhängig, zeitabhängig

wenn manuell-Vorort:

**Schalterart:** LON / Konventionell

wenn manuell-Zentral:

**Schalterart:** LON / Konventionell

wenn anwesenheitsabhängig:

**Präsenzerfassung:** über Präsenztaster / über Präsenzmelder

wenn Präsenzmelder:

**Raummaße**

**Einzelraumregelung:** ja / nein

**Anzahl der Heizkörper im Raum:** 1- x

**Steuerung:** manuell-Vorort, manuell-Zentral, anwesenheitsabhängig, zeitabhängig, Energiesparmodus

wenn manuell-Vorort:

**Schalterart:** LON / Konventionell

wenn manuell-Zentral:

**Schalterart:** LON / Konventionell

wenn anwesenheitsabhängig:

**Präsenzerfassung:** über Präsenztaster / über Präsenzmelder

wenn Präsenzmelder:

**Raummaße**

**Bedienung:**

**Tastermodell:** nur BCU, RCP-80, RCP-141, RCP-241, Glastaster 1fach, ....

Abbildung D.3: Überblick über die Navigationseinheiten des Prototyps

### Komponenten der Benutzungsschnittstelle und ihre Funktionen

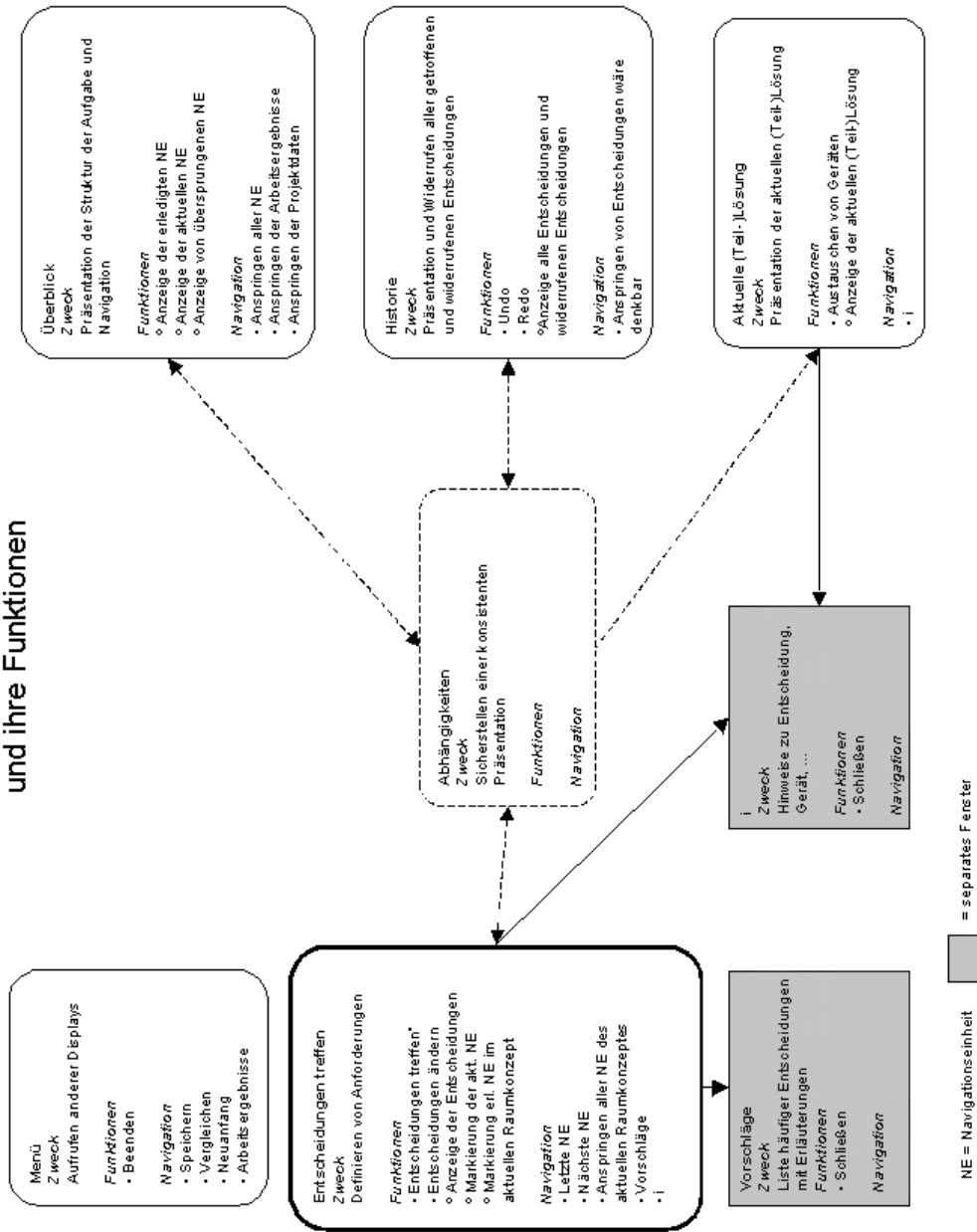


Abbildung D.4: Funktionen der einzelnen Komponenten des Prototyps im Leitsystem.



# Anhang E

## Mockups

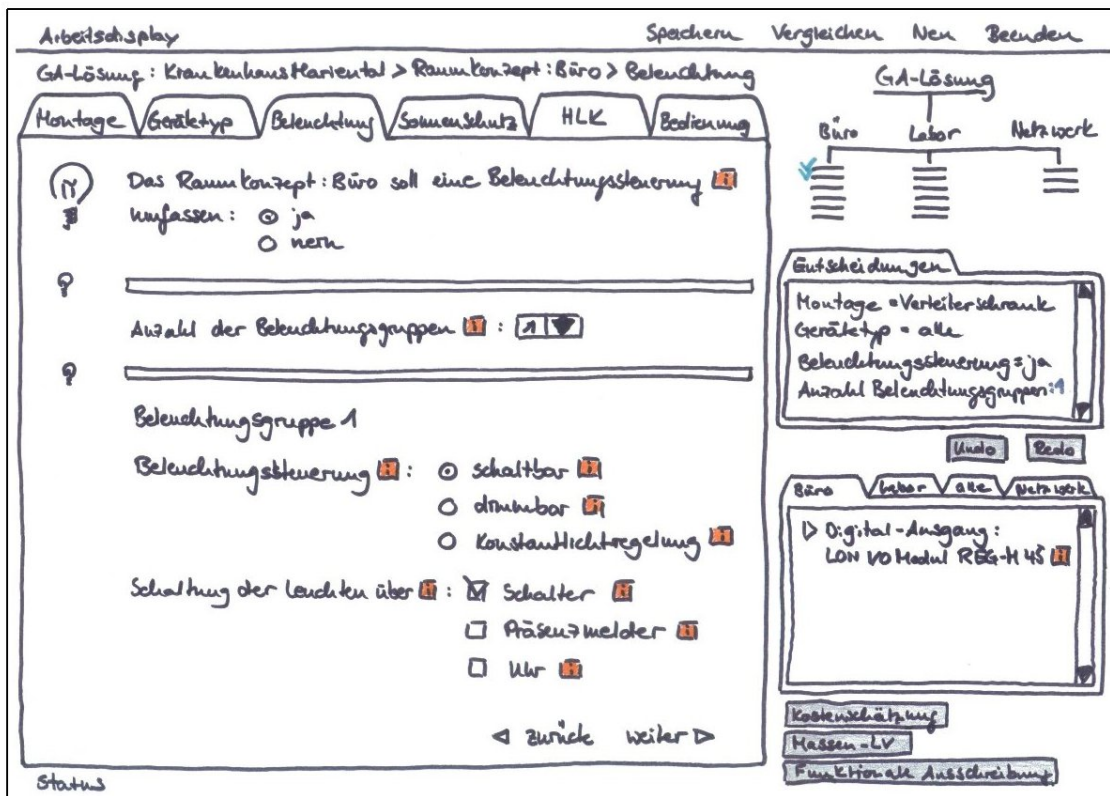


Abbildung E.1: Skizze des Prototyps.

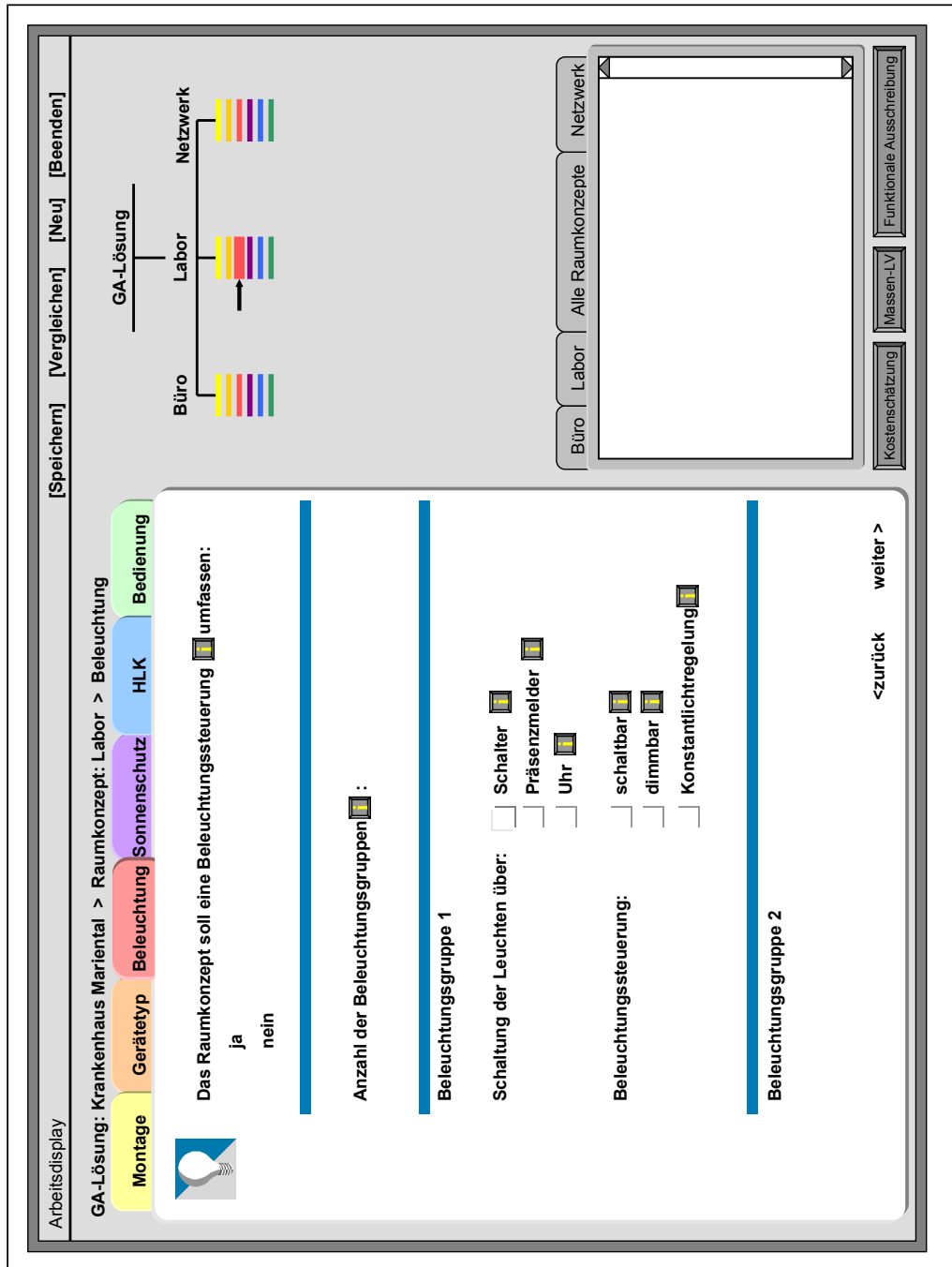


Abbildung E.2: Entwurf mit Reitern oben.

Arbeitsdisplay
[Speichern]
[Vergleichen]
[Neuanfang]
[Beenden]

GA-Lösung: Krankenhaus Mariental > Raumkonzept: Büro > Beleuchtung

**Montage**

**Gerätetyp**

**Beleuchtung**

**Sonnenschutz**

**HLK**

**Bedienung**

Das Raumkonzept soll eine Beleuchtungssteuerung umfassen:

ja  
 nein

Anzahl der Beleuchtungsgruppen:

Beleuchtungsgruppe 1:

Schaltung der Leuchten über:

Schalter  
 Präsenzmelder  
 Uhr

Steuerung der Leuchten:

schaltbar  
 dimmbar  
 Konstantlichtregelung

GA-Lösung

Büro

Labor

Netzwerk

Entscheidungen

- ▶ Anzahl der Raumkonzepte: 2
- ▶ Häufigkeit des Raumkonzeptes 1: 20
- ▶ Häufigkeit des Raumkonzeptes 2: 20

**Raumkonzept: Büro**

- ▶ Montagemöglichkeiten: alle
- ▶ Gerätetyp: alle
- ▶ Beleuchtungssteuerung: ja
- ▶ Anzahl der Beleuchtungsgruppen: 1
- ▶ Schaltung der Leuchten über: Schalter
- ▶ Beleuchtungssteuerung: schaltbar

Büro

Alle Raumkonzepte

Labor

Alle Raumkonzepte

Netzwerk

Alle Raumkonzepte

**Beleuchtungssteuerung**

- ▶ Digital-Ausgang: LON I/O Modul REG-M 2S

Abbildung E.3: Entwurf mit Reitern seitlich.

# Literaturverzeichnis

- Arthur und Passini 1992** ARTHUR, P. ; PASSINI, R.: *Wayfinding: People, Signs, and Architecture*. McGraw-Hill Ryerson Limited, 1992
- Benest und Potok 1984** BENEST, I. D. ; POTOK, M. H. N.: Wayfinding: an approach using signposting techniques. In: *Behaviour and Information Technology* 3 (1984), Nr. 2, S. 99 – 107
- Berlange 1994** BERLANGE, Thomas: A Selective Undo Mechanism for Graphical User Interfaces Based on Command Objects. In: *ACM Transaction on Computer-Human Interaction* 1 (1994), September, Nr. 3, S. 269 – 294
- Beyer und Holtzblatt 1998** BEYER, Hugh ; HOLTZBLATT, Karen: *Contextual Design: defining customer-centered systems*. Morgan Kaufmann Publishers, 1998
- Beyer und Holtzblatt 1999** BEYER, Hugh ; HOLTZBLATT, Karen: Contextual Design. In: *interactions* 6 (1999), Nr. 1, S. 32 – 42
- Bias 1991** BIAS, Randolph G.: Walkthroughs: Efficient Collaborative Testing. In: *IEEE Software* 8 (1991), Nr. 5, S. 94 – 95
- Bias 1994** BIAS, Randolph G.: *The Pluralistic Walkthrough: Coordinated Empathies*. S. 63 – 103. In: NIELSEN, Jakob (Hrsg.) ; MACK, Robert L. (Hrsg.): *Usability Inspection Methodes*, John Wiley and Sons, 1994
- Biundo u. a. 1995** BIUNDO, Susanne ; GÜNTER, Andreas ; HERTZBERG, Joachim ; SCHNEEBERGER, Josef ; TANK, Wolfgang: *Planen und Konfigurieren*. S. 754 – 798. In: GÖRZ, Günther (Hrsg.): *Einführung in die künstliche Intelligenz*, Addison Wesley, 1995
- Brinkop 1999** BRINKOP, Axel: *Variantenkonstruktion durch Auswertung der Abhängigkeiten zwischen den Konstruktionsbausteinen*. infix, 1999 (Dissertationen zur künstlichen Intelligenz)
- Bryan und Gershman 1999** BRYAN, Doug ; GERSHMAN, Anatole: Opportunistic exploration of large consumer product spaces. In: *Proceedings of the first ACM conference on Electronic commerce*, ACM Press, 1999, S. 41–47

- Burge 1998** BURGE, Janet E.: *Knowledge Elicitation for Design Task Sequencing Knowledge*, Worcester Polytechnic Institute, Diplomarbeit, Dezember 1998
- Busch u. a. 1994** BUSCH, Bodo ; HERRMANN, Thomas ; JUST, Katharina ; RITTENBRUCH, Markus: *Systeme für Experten statt Expertensysteme: von der Folgenforschung zur kompetenzförderlichen Gestaltung wissensbasierter Technik*. Infix, 1994
- Callahan und Koenemann 2000** CALLAHAN, Ewa ; KOENEMANN, Jürgen: A comparative usability evaluation of user interfaces for online product catalogs. In: *Proceedings of the 2nd ACM conference on Electronic commerce*, ACM Press, 2000, S. 197–206
- Cooper 1995** COOPER, Alan: *About Face: The Essentials of User Interface Design*. IDG Books Worldwide, 1995
- Cunis u. a. 1991** CUNIS, Roman (Hrsg.) ; GÜNTER, Andreas (Hrsg.) ; STRECKER, Helmut (Hrsg.): *Das PLAKON-Buch, Ein Expertensystemkern für Planungs- und Konfigurierungsaufgaben in technischen Domänen*. Springer, 1991
- Dietrich und Fischer 2001** DIETRICH, Dietmar (Hrsg.) ; FISCHER, Peter (Hrsg.): *LonWorks-Planerhandbuch: für Planer, Architekten und Betreiber*. VDE Verlag, 2001
- Dutke 1994** DUTKE, Stefan ; FRESE, Michael (Hrsg.) ; OBERQUELLE, Horst (Hrsg.): *Mentale Modelle: Konstrukte des Wissens und Verstehens. Kognitionspsychologische Grundlagen für die Software-Ergonomie*. Verlag für Angewandte Psychologie, 1994 (Arbeit und Technik: Praxisorientierte Beiträge aus Psychologie und Informatik)
- Emde u. a. 1997** EMDE, Werner ; RAHMER, Jörg ; VOSS, Angi ; BEILKEN, Christian ; BÖRDING, Josef ; ORTH, Wolfgang ; PETERSEN, Ulrike ; SCHAAF, Jörg ; SPENKE, Michael ; WROBEL, Stefan: Interactive configuration in KIKon. In: MERTENS, Peter (Hrsg.) ; VOSS, Hans (Hrsg.): *Expertensysteme 97: Beiträge zur 4. Deutschen Tagung wissensbasierter Systeme (XPS-97)*, 1997, S. 79–91
- Felix u. a. 2001** FELIX, Daniel ; NIEDERBERGER, Christoph ; STEIGER, Patrick ; STOLZE, Markus: Feature-oriented vs. Needs-oriented Product Access for Non-Expert Online Shoppers. In: BEAT SCHMID, Volker T. (Hrsg.): *Towards the E-Society : E-commerce, E-business, and E-government ; the first IFIP conference on E-commerce, E-business, E-government (I3E 2001)*, Kluwer Academic Publishers, 2001, S. 399–406. – <http://www.zurich.ibm.com/~mrs/papers.html>; zuletzt besucht: 07.01.2003
- Fitzgibbon und Patrick 1987** FITZGIBBON, L. ; PATRICK, J.: The use of structural displays to facilitate learning. In: H. J. BULLINGER, B. S. (Hrsg.): *Human-Computer Interaction - INTERACT '87*, Elsevier Science, 1987, S. 611 – 616
- Fleming 1998** FLEMING, Jennifer ; KOMAN, Richard (Hrsg.): *Web Navigation: Designing the User Experience*. O'Reilly, 1998

- Günter 1990** GÜNTER, Andras: Expertensysteme für Konstruktionsaufgaben. In: *Künstliche Intelligenz* 4 (1990), Nr. 3, S. 19
- Günter 1992** GÜNTER, Andreas: *Flexible Kontrolle in Expertensystemen zur Planung und Konfigurierung in technischen Domänen*, Universität Hamburg, Dissertation, 1992
- Günter 1993** GÜNTER, Andreas: Verfahren zur Auflösung von Konfigurationskonflikten in Expertensystemen. In: *Künstliche Intelligenz* (1993), Nr. 1, S. 16 – 23
- Günter 1995a** GÜNTER, Andreas: *Das Projekt PROKON im Überblick*. S. 3 – 10. In: GÜNTER, Andreas (Hrsg.): *Wissensbasiertes Konfigurieren: Ergebnisse aus dem Projekt PROKON*, Infix, 1995
- Günter 1995b** GÜNTER, Andreas (Hrsg.): *Wissensbasiertes Konfigurieren: Ergebnisse aus dem Projekt PROKON*. Infix, 1995
- Günter u. a. 2001** GÜNTER, Andreas ; HOLLMAN, Oliver ; RANZE, K. C. ; WAGNER, Thomas: Wissensbasierte Konfiguration von komplexen variantenreichen Produkten in internetbasierten Vertriebsszenarien. In: *Künstliche Intelligenz* (2001), Nr. 1, S. 33–36
- Günter und Hotz 1995** GÜNTER, Andreas ; HOTZ, Lothar: *Auflösung von Konfigurationskonflikten mit wissensbasiertem Backtracking und Reparaturanweisungen*. S. 181 – 191. In: GÜNTER, Andreas (Hrsg.): *Wissensbasiertes Konfigurieren: Ergebnisse aus dem Projekt PROKON*, Infix, 1995
- Günter und Kühn 1999** GÜNTER, Andreas ; KÜHN, Christian: Knowledge-Based Configuration - Survey and Future Directions -. In: PUPPE, Frank (Hrsg.): *Knowledge based systems: survey and future directions; Proceedings XPS-99*, Springer, 1999 (Lecture notes in artificial intelligence), S. 47 – 66
- Görz 1993** GÖRZ, Günther (Hrsg.): *Einführung in die künstliche Intelligenz*. 1. Auflage. Addison Wesley, 1993
- Hacker 1992** HACKER, Winfried ; FRESE, Michael (Hrsg.) ; OBERQUELLE, Horst (Hrsg.): *Expertenkönnen: Erkennen und Vermitteln*. Verlag für Angewandte Psychologie, 1992 (Reihe: Arbeit und Technik: Praxisorientierte Beiträge aus Psychologie und Informatik)
- Hacker 1994** HACKER, Winfried: *Arbeits- und organisationspsychologische Grundlagen der Software-Ergonomie*. Kap. 2, S. 53–93. In: EBERLEH, Edmund (Hrsg.): *Einführung in die Software-Ergonomie*, de Gruyter, 1994
- Herczeg 1986** HERCZEG, Michael: *Eine objektorientierte Architektur für wissensbasierte Benutzerschnittstellen*, Universität Stuttgart, Dissertation, 1986
- Hüllenkremer 2003** HÜLLENKREMER, Michael: Erfolgreiche Unternehmen arbeiten mit Produktkonfiguratoren. In: *Industrie Management* 19 (2003), Nr. 1, S. 37 – 40

- Hollmann u. a. 2000** HOLLMANN, Oliver ; WAGNER, Thomas ; GÜNTER, Andreas: EngCon: A Flexible Domain-Independent Configuration Engine. In: *Configuration: Papers from the Workshop at ECAI 2000*, AAAI Press, 2000, S. 94 – 96. – <http://www.hitec-hh.de/ueberuns/home/aguenter/literatur/literatur.html>; zuletzt besucht: 17.12.2002
- Holtzblatt und Beyer 1996** HOLTZBLATT, Karen ; BEYER, Hugh: *Conceptual Design: Principles and Practice*. S. 301 – 333. In: WIXON, Dennis (Hrsg.) ; RAMEY, Judith (Hrsg.): *Field methods casebook for software design*, John Wiley & Sons, 1996
- Howes und Payne 1990** HOWES, Andrew ; PAYNE, Stephen J.: Supporting Exploratory Learning. In: DIAPER, D. (Hrsg.) ; GILMORE, D. (Hrsg.) ; COCKTON, G. (Hrsg.) ; SHACKEL, B. (Hrsg.): *Human-Computer Interaction — INTERACT '90*, Elsevier Science, 1990, S. S. 881 – 885
- Johnson und Henderson 2002** JOHNSON, Jeff ; HENDERSON, Austin: Conceptual Models: Begin by Designing What to Design. In: *Interactions* (2002), Januar und Februar, S. 25 – 32
- Koch u. a. 1991** KOCH, Manfred ; REITERER, Harald ; TJOA, A. M. ; SCHAUER, Helmut (Hrsg.): *Software Ergonomie: Gestaltung von EDV-Systemen - Kriterien, Methoden und Werkzeuge*. Springer, 1991 (Angewandte Informatik)
- Liang und Chen 2000** LIANG, Ting-Peng ; CHEN, Niam-Shin: *Design of Electronic Stores*. S. 215–232. In: SHAW, Michael (Hrsg.) ; BLAUNING, Robert (Hrsg.) ; STRADER, Tray (Hrsg.): *Handbook on electronic commerce*, Springer, 2000
- LON Nutzer Organisation e.V. 2000** LON NUTZER ORGANISATION E.V. (Hrsg.): *LonWorks-Installationshandbuch: LonWorks-Praxis für Elektrotechniker*. VDE Verlag, 2000
- Maß 1994** MAASS, Susanne: Transparenz - Eine zentrale Software-ergonomische Forderung / Fachbereich Informatik, Universität Hamburg. 1994 (170). – Forschungsbericht
- Maß und Oberquelle 1992** MAASS, Susanne ; OBERQUELLE, Horst: *Persectives and Metaphors for Human-Computer Interaction*. S. 233 – 251. In: FLOYD, Christiane (Hrsg.) ; ZÜLLIGHOVEN, Heinz (Hrsg.) ; BUDDE, Reinhard (Hrsg.) ; KEIL-SLAWIK, Reinhard (Hrsg.): *Software development and reality construction*, Springer, 1992
- Maes u. a. 1999** MAES, Pathie ; GUTTMAN, Robert H. ; MOUKAS, Alexandros G.: Agents That Buy and Sell. In: *Communications of the ACM* 42 (1999), März, Nr. 3, S. 81–91
- Mayhew 1992** MAYHEW, Deborah: *Principles and Guidelines in Software User Interface Design*. Prentice Hall, 1992

- Mayhew 1999** MAYHEW, Deborah: *The Usability-Engineering Lifecycle: A practitioner's handbook for user interface design*. Morgan Kaufmann Publishers, 1999
- Meuser und Nagel 1991** MEUSER, Michael ; NAGEL, Ulrike: *ExpertInneninterviews - vielfach erprobt, wenig bedacht. Ein Beitrag zur qualitativen Methodenforschung*. S. 441 – 471. In: GARZ, Detlef (Hrsg.) ; KRAIMER, Klaus (Hrsg.): *Qualitative empirische Sozialforschung: Konzepte, Methoden, Analysen*, Westdeutscher Verlag, 1991
- Meyer-Fujara u. a. 1995** MEYER-FUJARA, Josef ; PUPPE, Frank ; WACHSMUTH, Ipke: *Expertensysteme und Wissensmodellierung*. S. 705 – 753. In: GÖRZ, Günther (Hrsg.): *Einführung in die künstliche Intelligenz*, Addison Wesley, 1995
- Meiers Lexikonredaktion 1992** MEYERS LEXIKONREDAKTION (Hrsg.): *Meiers großes Taschenlexikon: in 24 Bänden*. BI-Taschenbuchverlag, 1992
- Miles u. a. 2000** MILES, Gareth E. ; HOWES, Andrew ; DAVIES, Anthony: A framework for understanding human factors in web-based electronic commerce. In: *International Journal of Human Computer Studies* 52 (2000), S. 131–163
- Nielsen 1989** NIELSEN, Jakob: *Usability engineering at a Discount*. S. 394–401. In: SALVENDY, G. (Hrsg.) ; SMITH, M. J. (Hrsg.): *Designing and Using Human-Computer Interfaces and Knowledge Based Systems*, Elsevier Science Publishers B.V., 1989
- Nielsen 1990** NIELSEN, Jakob: Big paybacks from 'discount' usability engineering. In: *IEEE Software* 7 (1990), Nr. 3, S. 107–108
- Nielsen 1993** NIELSEN, Jakob: *Usability Engineering*. AP Professional, 1993
- Nielsen 1994a** NIELSEN, Jakob: Estimating the number of subjects needed for a thinking aloud test. In: *International Journal of Human Computer Studies* 41 (1994), S. 385 – 397
- Nielsen 1994b** NIELSEN, Jakob ; NIELSEN, Jakob (Hrsg.) ; MACK, Robert L. (Hrsg.): *Usability Inspection Methods*. John Wiley & Sons, 1994
- Nielsen 2000** NIELSEN, Jakob: *Erfolg des Einfachen: Jakob Nielsen's Web Design*. Markt + Technik Verlag, 2000
- Nielsen 2002** NIELSEN, Jakob: *Site Map Usability*. WWW. 2002. – <http://www.useit.com/alertbox/20020106.html>; zuletzt besucht: 05.05.2003
- Nielsen und Tahir 2002** NIELSEN, Jakob ; TAHIR, Marie: *Building Web Sites With Depth*. WWW. 2002. – <http://www.webtechniques.com/archives/2001/02/nielsen>; zuletzt besucht: 26.02.2002



- Nievergelt und Weydert 1979** NIEVERGELT, J. ; WEYDERT, J.: Sites, modes, and trails: telling the user of an interactive system where he is, what he can do, and how to get to places. In: GUEDJ ET AL. (Hrsg.): *IFIP Workshop on Methodology of interaction*, North-holland publishing company, 1979, S. 327–338
- Norman 1983** NORMAN, Donald A.: *Some Observations on Mental Models*. S. 7 – 14. In: GENTNER, Dedre (Hrsg.) ; STEVENS, Albert L. (Hrsg.): *Mental Models*, Lawrence Erlbaum, 1983 (Cognitive Science)
- Normenausschuss Ergonomie im DIN 1998** NORMENAUSSCHUSS ERGONOMIE IM DIN: *ISO 9241-13:1998*. 1998
- Oberquelle 1991** OBERQUELLE, Horst: *Perspektiven der Mensch-Computer-Interaktion und kooperativen Arbeit*. S. 45 – 56. In: FRESE, Michael (Hrsg.) ; KASTEN, Christoph (Hrsg.) ; SKARPELIS, Constantin (Hrsg.) ; ZANG-SCHEUCHER, Birgit (Hrsg.): *Software für die Arbeit von morgen. Bilanz und Perspektiven anwendungsorientierter Forschung*, Springer, 1991
- Oberquelle 1994** OBERQUELLE, Horst: *Formen der Mensch-Computer-Interaktion*. S. 95 – 143. In: EBERLEH, Edmund (Hrsg.) ; OBERQUELLE, Horst (Hrsg.) ; OPPERMAN, Reinhard (Hrsg.): *Einführung in die Software-Ergonomie: Gestaltung graphisch-interaktiver Systeme: Prinzipien, Werkzeuge, Lösungen*, de Gruyter, 1994
- O’Keefe und McEachern 1998** O’KEEFE, Robert M. ; MCEACHERN, Tim: Web-based Customer Decision Support Systems. In: *Communications of the ACM* 41 (1998), März, Nr. 3, S. 71 – 78
- Passini 1984** PASSINI, Romedi: *Wayfinding in Architecture*. Van Nostrand Reinhold Company Inc., 1984
- Paul 1994** PAUL, Hansjürgen: *Exploratives Agieren: ein Beitrag zur ergonomischen Gestaltung interaktiver Systeme*. Europäischer Verlag der Wissenschaften, 1994 (Europäische Hochschulschriften: Reihe 41, Informatik 16)
- Perfetti und Landesman 2003** PERFETTI, Christine ; LANDESMAN, Lori: *Eight Is Not Enough*. WWW. 2003. – [http://www.uie.com/Articles/eight\\_is\\_not\\_enough.htm](http://www.uie.com/Articles/eight_is_not_enough.htm); zuletzt besucht:23.04.03
- Preece u. a. 2002** PREECE, Jennifer ; ROGERS, Yvonne ; SHARP, Helen: *Interaction design: beyond human-computer interaction*. John Wiley & Sons, 2002
- Pu und Faltings 2000** PU, Pearl ; FALTINGS, Boi: Enriching buyer’s experiences: the SmartClient approach. In: *CHI*, 2000, S. 289–296
- Puppe 1990** PUPPE, Frank: *Problemlösungsmethoden in Expertensystemen*. Springer, 1990 (Studienreihe Informatik)

- Puppe 1991** PUPPE, Frank ; BRAUER, W. (Hrsg.) ; GOOS, G. (Hrsg.): *Einführung in Expertensysteme*. 2. Auflage. Springer, 1991
- Rahmer 1999** RAHMER, Jörg: *Exploratives Konfigurieren: Über die Unterstützung des Benutzers bei computergestützten explorativen Tätigkeiten*, Universität Kaiserslautern, Dissertation, 1999
- Ranze u. a. 2002** RANZE, Christoph ; SCHOLZ, Thorsten ; WAGNER, Thomas ; GÜNTER, Andreas ; HERZOG, Otthein ; HOLLMANN, Oliver ; SCHLIEDER, Christoph ; ARLT, Volter: *A Structure Based Configuration Tool: Drive Solution Designer - DSD*. 2002. – <http://www.hitec-hh.de/ueberuns/home/aguentner/literatur/literatur.html>; zuletzt besucht: 17.12.2002
- Richter 1996** RICHTER, Michael M.: Neue Aufgaben beim Planen und Konfigurieren. In: SAUER, Jürgen (Hrsg.) ; GÜNTER, Andreas (Hrsg.) ; HERTZBERG, Joachim (Hrsg.): *Planen und Konfigurieren 96: Beiträge zum 10. Workshop "Planen und Konfigurieren"(PuK-96)*, Infix, 1996, S. 3–15
- Rosewitz 2001** ROSEWITZ, Martin: *PBK-Editor: Ein Werkzeug zur Erstellung von WWW-gestützten Produktberatungskomponenten*, Friedrich-Alexander-Universität Erlangen-Nürnberg, Dissertation, Oktober 2001
- Rosson und Carroll 2002** ROSSON, Mary B. ; CARROLL, John M.: *Usability Engineering: Scenario-based Development of Human-Computer Interaction*. Morgan Kaufmann Publishers, 2002
- Schenck 1997** SCHENCK, Jutta: *Navigation und Orientierung in Interfacedesign: Übertragung eines Orientierungsmodells aus der Architektur am Beispiel einer elektronischen Fahrplanauskunft*. 1997
- Schreier 2001** SCHREIER, Jürgen: Alles passt von Anfang an: Automatische Produktkonfiguration senkt die Folgekosten in Konstruktion und Produktion. In: *Maschinenmarkt* (2001), Oktober, Nr. 40, S. 18–19
- Shneiderman 1998** SHNEIDERMAN, Ben: *Designing the user interface: strategies for effective human-computer interaction*. Addison Wesley Longman, 1998
- Shneiderman 2002** SHNEIDERMAN, Ben: *User Interface Design*. mitp-Verlag, 2002. – deutsche Ausgabe
- Spenke u. a. 1996** SPENKE, Michael ; BEILKEN, Christian ; BERLAGE, Thomas: FOCUS: The Interactive Table for Product Comparison and Selection. In: *UIST'96 Proceedings of the ACM Symposium on User Interface Software and Technology*, 1996, S. 41–50

- Spool und Schroeder 2001** SPOOL, Jared ; SCHROEDER, Will: Testing web sites: five users is nowhere near enough. In: *CHI '01 extended abstracts on Human factors in computer systems*, ACM Press, 2001, S. 285–286. – <http://www.winwriters.com/download/chi01-spool.pdf>, zuletzt besucht: 17.04.03
- Stolpmann 2001** STOLPMANN, Markus: *Service und Support im Internet, Intelligente Dienstleistungen - effizient zum Erfolg*. Galileo Business, 2001
- Stolze 1999** STOLZE, Markus: Comparative Study of Analytical Product Selection Support Mechanisms. In: SASSE, M. A. (Hrsg.) ; JOHNSON, Chris (Hrsg.): *Human-Computer Interaction: Interact'99*, IOS Press, 1999, S. 45–53
- Tank 1993** TANK, Wolfgang: Wissensbasiertes Konfigurieren: Ein Überblick. In: *Künstliche Intelligenz* 7 (1993), Nr. 1, S. 7 – 10
- Tilson u. a. 2002** TILSON, Roger ; DONG, Jianming ; MARTIN, Shirley ; KIEKE, Eric: *Factors and Principles Affecting the Usability of Four E-commerce Sites*. WWW. 2002. – <http://research.att.com/conf/hfweb/proceedings/tilson/index.html>; zuletzt besucht: 12.04.2002
- TZI 2001** TZI, Universität Bremen: *Datenstrukturen und Algorithmen zur Konfliktkontrolle in EngCon*. Dezember 2001. – unveröffentlicht
- Williamson und Shneiderman 1992** WILLIAMSON, Christopher ; SHNEIDERMAN, Ben: The Dynamic HomeFinder: Evaluating Dynamic Queries in a Real-Estate Information Exploration System. In: BELKIN, Nicholas (Hrsg.) ; INGWERSEN, Peter (Hrsg.) ; PEJTERSEN, Annelise M. (Hrsg.): *Proceedings of the Fifteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, ACM Press, 1992, S. 338 – 346
- Winograd 1996** WINOGRAD, Terry ; WINOGRAD, Terry (Hrsg.): *Bringing Design to Software*. ACM press books, 1996
- Yang 1988** YANG, Yiya: Undo support models. In: *International Journal of Man-Maschine Studies* 28 (1988), S. 457 – 481
- Yang 1990** YANG, Yiya: Current Approaches and new Guidelines for Undo Support Design. In: D. DIAPER ET AL. (Hrsg.): *Human-Computer Interaction - INTERACT'90*, Elsevier Science, 1990, S. 543 – 548
- Ziegler 1997** ZIEGLER, Jürgen: ViewNet - Konzeptionelle Gestaltung und Modellierung von Navigationsstrukturen. In: RÜDIGER LISKOWSKY, Wolfgang W. (Hrsg.): *Software-Ergonomie '97: Usability Engineering: Integration von Mensch-Computer-Interaktion und Software-Entwicklung*, Teubner, 1997 (Berichte des German Chapter of the ACM 49), S. S. 343 – 350